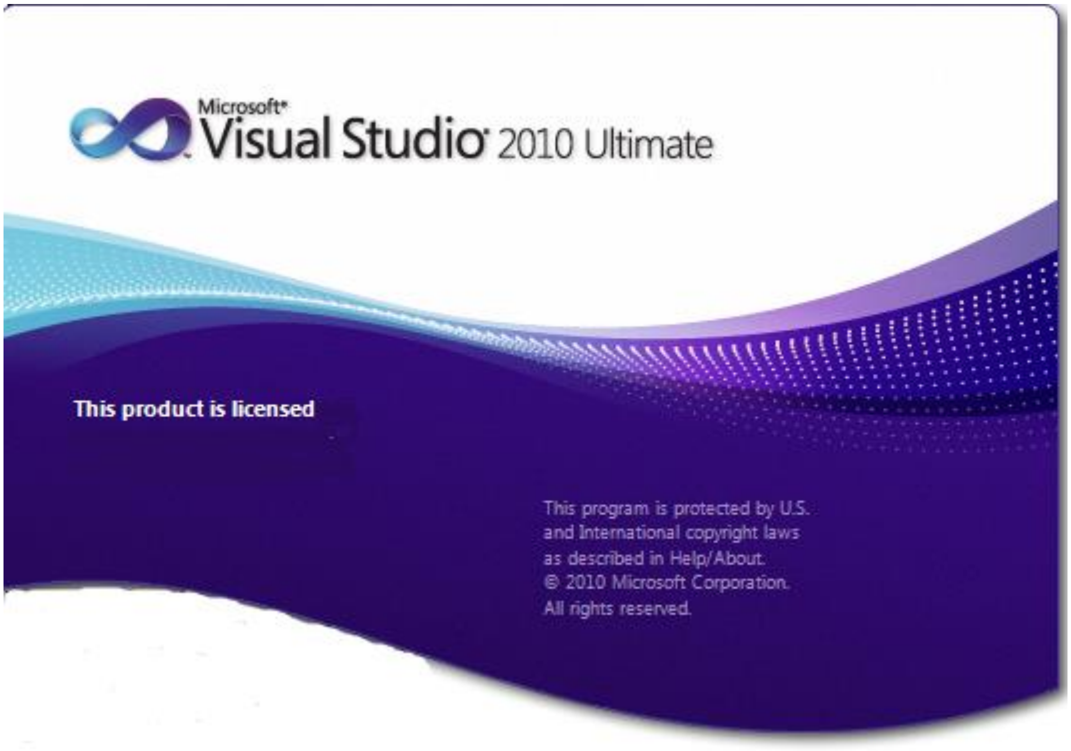
VSTS/TFS for Requirements Management



This product is licensed

This program is protected by U.S. and International copyright laws as described in Help/About. © 2010 Microsoft Corporation. All rights reserved.

A guide for Business Systems Analysts in XYZZY Corporation

Contents

Contents 2
Introduction 4
Overview and Definitions 4
Components of VSTS 6
Connecting to a Team Project 8
VSTS Client – What’s What 9
Work Items 11
  Types of Work Items 11
Entering Requirements 13
  Stating Requirements as Atomic Entities 13
  Recommended Process 13
  Multiple Regions 14
  Requirement Types 14
  Requirement Work Item 16
More on Work Items 22
  Hierarchy of Work Items 22
  Work Item Data Fields 22
  Creating a New Work Item 22
  Saving a Work Item 24
  Editing a Work Item 24
  Work Item States 26
  Linking Work Items 27
    To Link Parent-Child 27
    To Delete a Link 33
    Linking Across Projects 33
  Reporting Work Item Links 35
    Reporting Links 35
    Querying For Links – Parent-Child 36
    Querying For Links – All or Selected 37
    Creating a Requirements Traceability Matrix Query 39
  Including an External Document 41
    Attach a Document 41
    Link to Document 42
Queries 44
  Running Queries 45
  Creating Queries 46
    Additional Notes on Creating Queries 54
  Editing Queries 55
Importing and Exporting Work Items 56
  Export to Excel, Starting in VSTS 56
  Export to Excel, Starting in Excel 57
  Working with Data in Excel 58
  Publish to TFS (Import to TFS from Excel) 61
Reports 63
Areas and Iterations 64
  Defining Areas and Iterations 65
    Defining Areas 66
    Defining Iterations 69

Migrating Requirements from Other Documents 70
BRD to Business Objective 70
SRS to Features and Requirements 70
Migrating from Excel 71
Migrating from Word 71
Using VSTS/TFS Within a Software Development Process 72
Peer Reviews 72
Change Requests 74
Appendix A – TFS Web Access 76
Appendix B – Work Item States and Transitions 77
TFS 2010 77
TFS 2012 78
State Transition Diagrams 78
Requirement State Transition Diagram 79
Change Request State Transition Diagram 80
Issue State Transition Diagram 81
Documentation Defect State Transition Diagram 82
Defect State Transition Diagram 83
Test Case State Transition Diagram 84
Task State Transition Diagram 85

Introduction

Requirements management is the foundation of good software development. If you don't have good requirements, you don't know what you need to build and test. VSTS/TFS provides an automated tool for recording, managing and reporting on requirements.

A requirements management tool needs to do two kinds of things:

- Track requirements as individual items, including the following:
    - Traceability: the requirement's relationship to other things like feature requests, other requirements, code, test cases, etc.
    - State: Whether the requirement is proposed, accepted, completed, etc.
    - Scope: What release of the software the requirement is in.
    - Owner: Who is responsible for the requirement.
- Communicate all this to human readers in the form of such artifacts as the following:
    - BRD and SRS documents, which can include explanatory text, charts and diagrams that are not part of any requirements, but provide context for understanding them.
    - Reports such as Traceability Matrix showing requirement traceability, In-Scope/Out-of-Scope showing what is included in a given software release, etc.

VSTS/TFS in an excellent tool for tracking requirements. It is not as good as other mechanisms, such as Microsoft Word, at communicating context, and has varying capabilities for reporting. Some kinds of reports are easy; others require some SQL programming.

XYZZY has mandated the use of VSTS/TFS for requirements management. This document tells requirements analysts how to use it for that purpose.

> **Note** – *In order to use VSTS/TFS fully you must use Microsoft Excel 2007 or later as well. Some functions can be done only by exporting data to Excel, manipulating the data there, and importing it back into VSTS/TFS.*

Overview and Definitions

VSTS stands for Microsoft Visual Studio Team Suite, which includes a set of collaboration and development tools for a software development process. It gives development teams the ability to create, track, manage, document, link and report on requirements, bugs, change requests, reviews and other work items.

Visual Studio Team Explorer is the workstation client user interface component of all Visual Studio team editions. In addition, the various team system editions include several development and collaboration tools for various team roles. Requirements analysts will use the Team Explorer, but not necessarily the other features, for instance those that enable developers to check in and check out code.

TFS is the server component of VSTS. TFS manages the central repository of the team data, including requirements, and enables communication between team members.

TFS Web Access is a web-based interface to TFS that enables the user to do many of the same things that the workstation client (Visual Studio Team Explorer) does. See Appendix A – TFS Web Access, page 76.

Any given Team Foundation Server contains one or more Team Projects, each of which consists of Visual Studio solutions, configuration files for Team Build and Team Load Test Agents, and a single SharePoint repository containing the pertinent documents for the project. A team project contains the user-defined work items, source branches, and reports that are to be managed by TFS. Every set of requirements is housed in a team project.

> Note – a team project is not the same as an IT project defined in the project management system. A team project is a bunch of work items, code, etc. It can include many IT projects, defined as Iterations within the team project. See Areas and Iterations, page 64.

Every team project has an associated team website hosted on the Team Foundation Server. Team members can use this SharePoint project portal to save and retrieve documents, view reports, exchange information by posting messages, and use other SharePoint collaborative features like calendars and lists.

All TFS instances, and hence all Team Projects are physically housed in a Collection, which is a server or a set of servers acting as one. You can link requirements between team projects only if they are all 1n the same collection.

A work item is an object that has a pre-defined workflow and set of fields, and maps to the Software Development Lifecycle (SDLC). VSTS/TFS as implemented at XYZZY contains many different types of work items. The most important one for requirements analysts is Requirement, defined as a condition or capability that must be met by software needed by a user to solve a problem or achieve an objective.

[Some pages have been omitted.]

Work Items

A work item is an object stored in the database that has a pre-defined workflow and set of fields, and maps to the Software Development Lifecycle (SDLC) defined in the template (agile or traditional waterfall). Each work item type has its own set of fields, including values for selection boxes and state transitions, and a visual representation for data entry in the workstation client and in web access. Most of your work will be done with work items: creating new ones, entering and updating data in them, and linking them to other work items. Anyone can create a work item of any type, but process recommendations restrict creation to certain roles for different types.

To create a new work item, right-click Work Items in the Team Explorer pane and select Add Work Item, or select Team from the application menu, then Add Work Item. See Creating a New Work Item, page 22.

To find a work item to view or edit, open Work Items in the Team Explorer pane, then Team Queries. Double-click a query and then select one of the work items.

Types of Work Items

All work items have many fields in common, such as Title, Classification and Description, and each has its unique fields as well. Here is a list of the types of work items in the Traditional template. The most important ones for business analysts are Requirement and Change Request.

- **Change Request**
  The Change Request work item is used to record requests to change code, requirements or design that have already been approved. It may be entered by anybody. For process guidance, please refer to Change Requests, page 74.

- **Defect** – Should be entered by a tester or developer. This is for code defects, not defects in documents such as the SRS or TDS. Use Documentation Defect to enter defects in documents.

- **Documentation Defect** – Should be entered by a tester or developer. This is for defects in documents such as the SRS or TDS, not code defects. Select the document type in the pull-down list in the Document Type field.

- **Issue**
  This a catch-all for various kinds of information. The “Issue or Risk” field allows you to document the following:
  - Issue
  - Assumption
  - Constraint
  - Dependency
  - Question
  - Risk

  This is what corresponds to the Assumptions, Dependencies and Constraints section of the traditional SRS document.

- **Iteration**
  This is not the same as Iteration Path as a project attribute (See Areas and IterationsAreas and Iterations, page 64.) This is for Agile-type projects. Business Analysts should not enter a work item of this type.

- **Requirement**
  This is where you enter requirements of all types: Business Objective, Business Rule, Feature and Requirement. Please see the detailed discussion under Entering Requirements, page 13.

- **Review**
  This work item is to record what items are to be covered in a peer review or inspection, and the results of the review.

  Should be created by whomever is conducting the review, which might be a BA or a developer.

  For more details on how to use this work item type, see Peer Reviews, page 72.

- **Risk** – This is to be used by the project manager to record project risks.

- **Shared Steps** – To be used by the test team to record shared steps of test cases.

- **Task**
  The task work item is the lowest-level work item. It does not have any further subdivision or category. It is used to assign work to people. You can import tasks from Microsoft Project.

  You could assign tasks to yourself as a personal to-do list. Check with the project manager! It might export out to Microsoft Project.

  Development tasks should be linked to one or more requirements.   Let the development team do that.

  Should be entered by the Project manager or the Dev Lead. The Dev Lead would assign coding tasks to developers.

- **Test Case** – To be used only by the test team.

Entering Requirements

The task of the business analyst is to enter requirements in VSTS/TFS and link them to each other. Linking allows traceability from business request to feature to requirement or business rule. Since each requirement is an atomic (individual) entity in the tool, you can easily keep track of which requirements are to be implemented in which project. The down side is that you lose the context that you have in a Word document. The requirements are not visually linked to use cases or context diagrams.

Stating Requirements as Atomic Entities

They are not linked to each other visually either. You will have problems if you do this:

REQ_010 System shall receive a request for service tags

REQ_020 If service tags have already been issued, system shall do this.

REQ_030 If service tags have not already been issued, system shall do that.

The problem with these requirements is that in VSTS you only see one requirement at a time; they are not displayed in sequence. If you only look at REQ_020, you would have an incomplete picture. And, more importantly, the developers and testers will have an incomplete picture. Instead, you will have to state the requirements like this:

REQ_010 System shall receive a request for service tags

REQ_020 After receiving a request for service tags, if service tags have already been issued, system shall do this.

REQ_030 After receiving a request for service tags, if service tags have not already been issued, system shall do that.

Recommended Process

Wait until your SRS is finalized and approved before entering the requirements in VSTS. Your requirements will be housed in the same place as code and test cases. Developers sometimes like to get a head start on their design, so if you put your requirements in VSTS before they are fully cooked, the designers might start designing something that won't work. To head off this problem, follow this process:

1. Start with the traditional Word document SRS
   - Or an Excel spreadsheet SRS and a Word document SRS Supplement

2. Get sign-off on the SRS

3. Enter the requirements into TFS
   - Copy and paste by hand into TFS, or
   - Copy and paste into Excel and upload (faster and easier)
     - (See Importing and Exporting Work Items, page 56.)

Note – You must enter requirements for a particular application in the Team Project for that application. Make sure you select the appropriate Team Project in the Team Explorer tab. If your SRS contains requirements for more than one application, you must enter them into different Team Projects.

4. Pick a requirement and link the Word document to it
   - (See Linking Work Items, page 27).
   - Select All Links tab
   - Click Link To
   - Select Hyperlink
   - Enter URL
   - Click OK

- If there is an example screen shot or a diagram, the requirement text should indicate that it is in the Word SRS, and the reader must open the Word document to see it.

- For each change request, enter it as a Change Request work item and link it to the affected requirements using the “affected by” link type.

- If a change request changes a diagram or screen shot, update it in the Word SRS. You do not have to update the requirement text in Word, but you may do so if you want to.

Multiple Regions

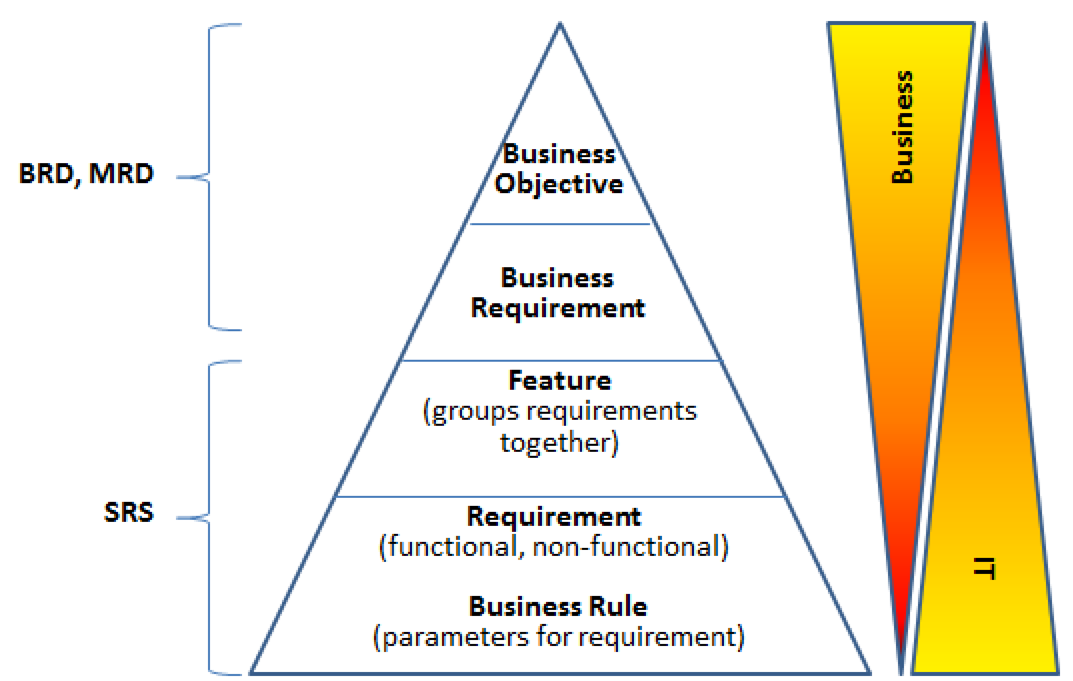
If your application has different instances in different regions and the same requirements apply to more than one region, you will have to enter your requirements more than once. We are putting requirements into the same TFS team project as the code and test cases. In the case of IRIA and OA, for instance, there is a team project for IRIA-AM, another for IRIA-EMEA and another for OA-APJ. So requirements for handling messages to and from vendors have to put separately into each team project. Using Excel to import requirements makes this easier than having to type them or copy and paste them over and over again.

With all this in mind, let’s take a look at how to enter requirements in VSTS.

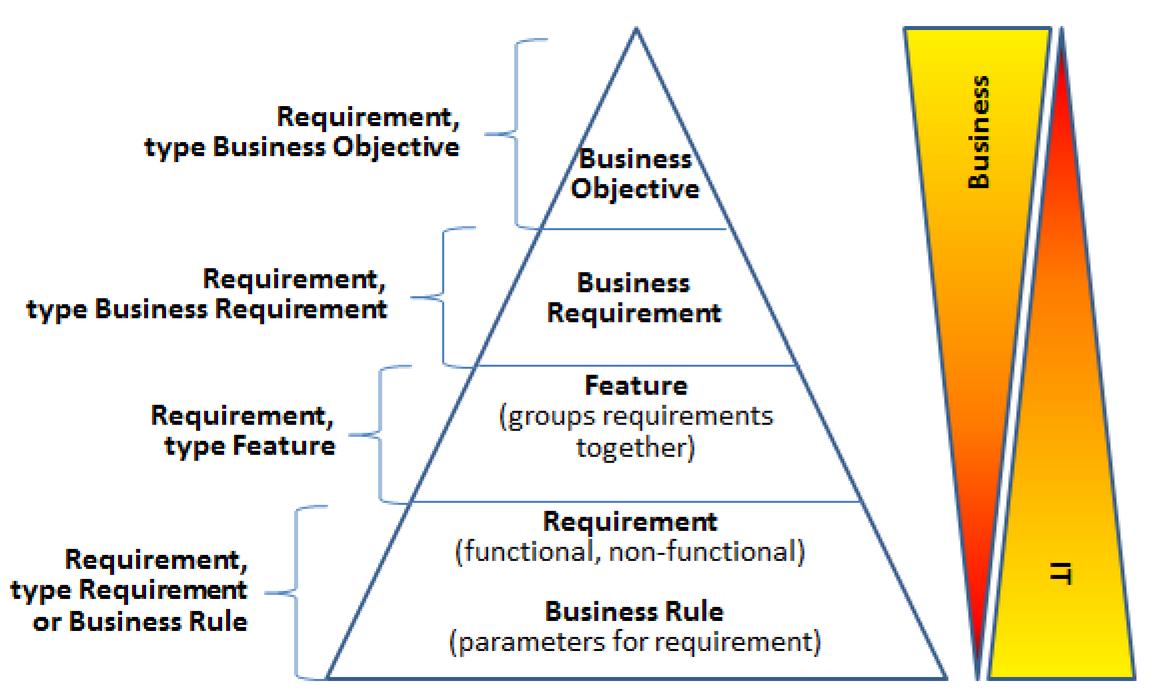
Requirement Types

The Requirement work item has five types: Business Objective, Business Requirement, Feature, Requirement and Business Rule. Here is how each of these maps to information in the traditional BRD (Business Requirements Document) or MRD (Marketing Requirements Document) and SRS (Software Requirements Specification). The higher-level types have more business-related information and less IT-related information. The reverse is true for the lower-level types.

XYZZY Documents



VSTS/TFS


Requirement, type Business Objective
Requirement, type Business Requirement
Requirement, type Feature
Requirement, type Requirement or Business Rule
Business Objective
Business Requirement
Feature
(groups requirements together)
Requirement
(functional, non-functional)
Business Rule
(parameters for requirement)
Business
IT

Please refer to the discussion of Requirements Work Item Type below for more information about each of these types.