

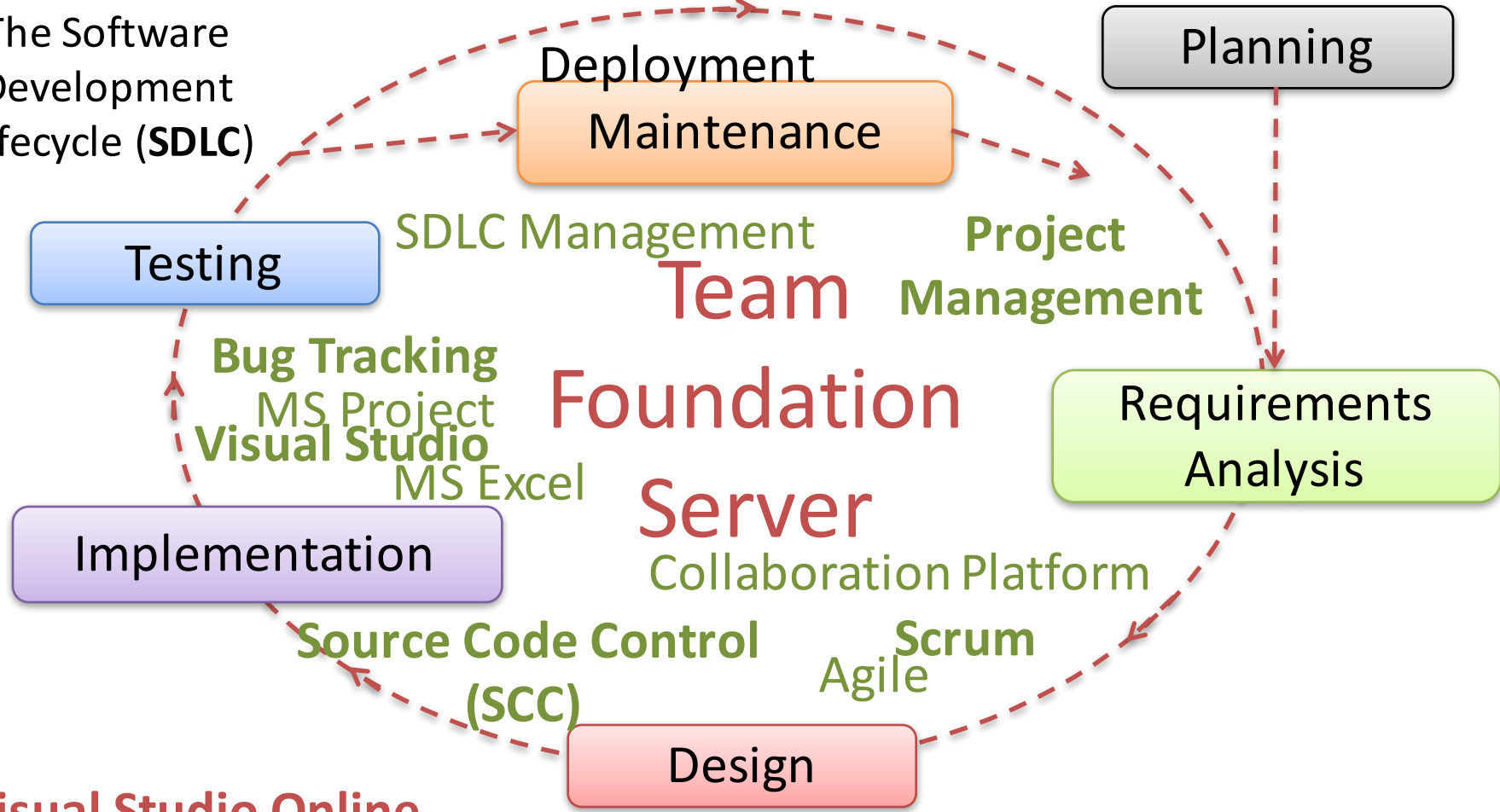


# Team Foundation Server Visual Studio Team Services

Hans-Petter Halvorsen, M.Sc.

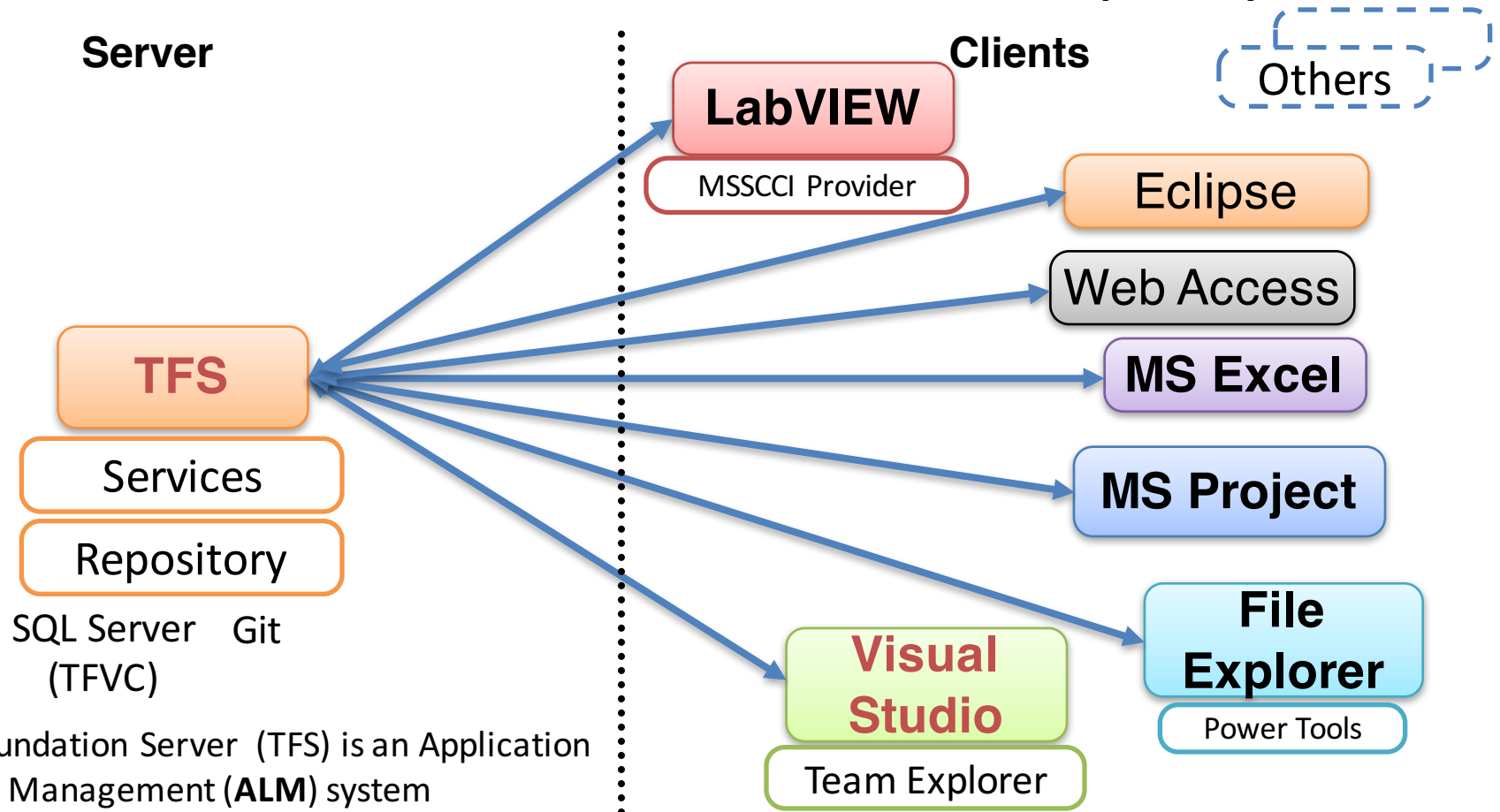
Team Foundation Server (TFS) is an Application Lifecycle Management (**ALM**) system

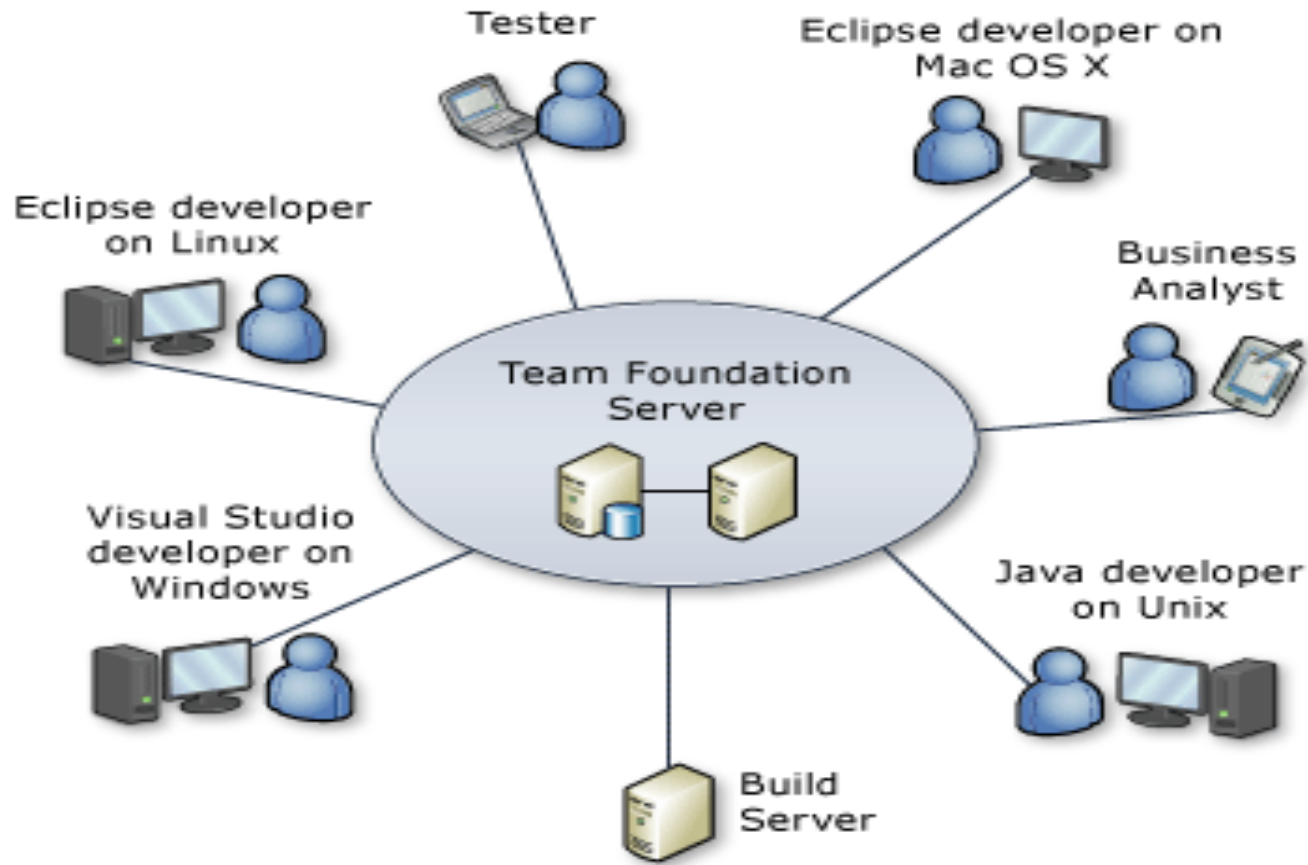
The Software  
Development  
Lifecycle (**SDLC**)



Visual Studio Online

# Team Foundation Server (TFS)





The diagram demonstrates that team members on different platforms can collaborate by using Team Foundation Server

# What is TFS?

- Team Foundation Server (TFS) is an Application Lifecycle Management (**ALM**) system,
  - i.e., the system takes care of all aspects in software development
  - from planning, requirements, coding, testing, deployment and maintenance.
- TFS is a Source Code Control (**SCC**), Bug Tracking, Project Management, and Team Collaboration platform
- Tightly integrated with **Visual Studio** as Microsoft is the vendor of both Visual Studio and TFS
- Cloud based edition (Hosting Service): “**Visual Studio Online**” (former: “Team Foundation Service”)

# Application Lifecycle Management, ALM

- ALM Tools: Tools that facilitate and integrate:
  - Requirements Management,
  - Architecture,
  - Coding,
  - Testing,
  - (Bug) Tracking,
  - And Release Management (-> Software Development!)
- There exist a lot of such tools (TFS, Jira, etc...)
- We will use Team Foundation Server (TFS) from Microsoft, since it is tightly integrated with Visual Studio

# Visual Studio Team Services

- TFS is a Source Code Control (**SCC**), Bug Tracking, Project Management, and Team Collaboration platform
- Integrated with Visual Studio
- TFS in the Cloud (This means you don't need to host the server yourself)
- Free for up to 5 users

[www.visualstudio.com](https://www.visualstudio.com)

## Account Creation

**Identity Provider**

**Account URL**

Create Account

# Team Foundation Server

- **SDLC** Management (SDLC – Software Development Life Cycle)
- Software **Team** Collaboration
- **Source Code** Management
- Supports **Agile, Scrum**, CMMI
- **Bug Tracking**
- Integrated **Test** Tools
- Automated Builds
- Built in Team Foundation Version Control (TSVC) + Support for Git repositories
- Built-in support for TFS in Visual Studio (Team Explorer)
- Plug-in for Eclipse (Team Explorer Everywhere)
- MSSCCI Provider for other IDEs like LabVIEW, etc.



# TFS Technical Overview

It uses a 3 tier/layer architecture

- Client tier (Web Services (used by Visual Studio, etc.))
- Application tier
  - Web Services
  - Web Portal, Team Web Access (TWA)
  - Document repository using SharePoint Services
- Data tier (SQL Server)

# TFS Client Tools

- **Team Explorer** (integrated with Visual Studio)
- **MS Excel Add-in** (part of Team Explorer installation)
- **MS Project Add-in** (part of Team Explorer installation)
- **Windows Explorer Integration** (part of Power Tools)  
Very useful when working with Team Foundation Server outside of Visual Studio or Eclipse
- **TFS MSSCCI Provider** (makes it possible to use TFS within tools that do not support TFS, such as e.g., LabVIEW, SQL Server, etc.)
- Team Explorer Everywhere
- 3.party software that integrates with TFS (using available web services)

# Source Code Control/Revision Control

With TFS you may use different source code repositories:

- **TFVC**
  - Team Foundation Version Control (TFVC) uses a single, centralized server repository to track and version files. Local changes are always checked in to the central server where other developers can get the latest changes.
- **Git** (“open source” alternative)
  - Git is a Distributed Version Control System (DVCS) that uses a local repository to track and version files. Changes are shared with other developers by pushing and pulling changes through a remote, shared repository.

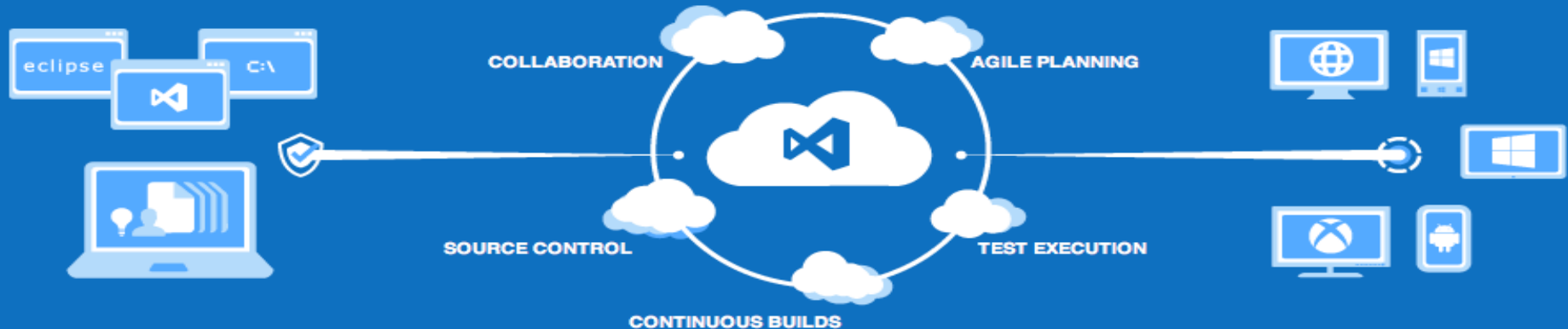


# Visual Studio Team Services (VSTS)

Hans-Petter Halvorsen, M.Sc.

# Visual Studio Team Services

- TFS in the Cloud (This means you don't need to host the server yourself)
- Free for up to 5 users
- [www.visualstudio.com](http://www.visualstudio.com)

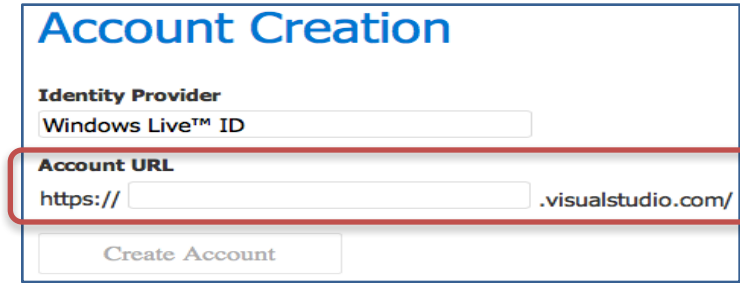


# We will learn to

- Create an Visual Studio Team Services Account
- Create Projects and Add Team Members
- Create a good Folder structure
- Create **Areas** and **Iterations**
- Create **Work Items** (Bug, Task, ...)
- Create **Queries**
- Use TFS inside Visual Studio

# VSTS- Getting Started

1. Create an Account
2. Create a New Team Project
3. You are ready to start
  1. Connect to Visual Studio Team Services from Visual Studio (similar as connection to a Team Foundation Server)
  2. Or use the Web-based interface provided (except for SCC)
4. Assign **Team Members**
5. Add **Areas, Iterations**, etc.
6. Add your Source Code
7. Check-in/Check-out

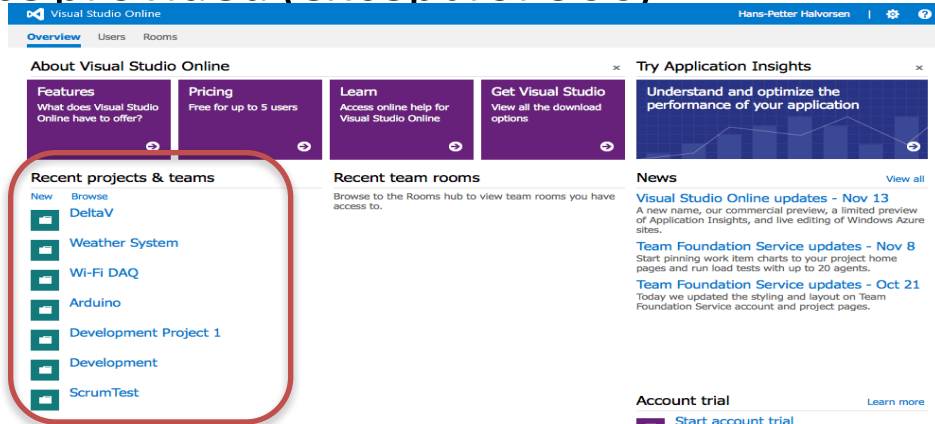


**Account Creation**

**Identity Provider**  
Windows Live™ ID

**Account URL**  
https:// [ ] .visualstudio.com/

Create Account



# Create New Team Project

Select what kind of development process you want to follow (Agile, **Scrum**, ...)

Select **TFVC** or Git Repository

CREATE NEW TEAM PROJECT

Project name

Note: You cannot change the name of your project after you have created it !!!

Description

Process template

- Microsoft Visual Studio Scrum 2013
- Microsoft Visual Studio Scrum 2013
- MSF for Agile Software Development 2013
- MSF for CMMI Process Improvement 2013

Version control



Team Foundation Version Control



Team Foundation Version Control (TFVC) uses a single, centralized server repository to track and version files. Local changes are always checked in to the central server where other developers can get the latest changes.

Create project Cancel




# The Project Start Page

 Visual Studio Online / **Development Project 1** 


Hans-Petter Halvorsen |  


HOME CODE WORK BUILD TEST


Search work items 


**Overview**

### How to

**Work**  
Create and manage  
your product backlog in  
minutes 

**Code**  
Learn about working  
with your code 


**Build**  
Learn about builds in  
the cloud 

**Test**  
Learn more about Test  
Manager and test  
planning 


### Pinned Items

42  
Example Pinned Items

12




### Work


Create new 


Backlog  
Task board  
Queries

### Team rooms


 **Machintosh Room**  
0 users in room


### Members

Manage... 



### Visual Studio

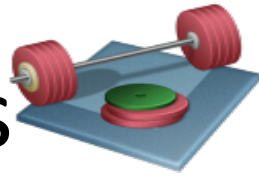
 **Open in Visual Studio**  
Requires Visual Studio 2013

 **Get Visual Studio**  
See Visual Studio downloads

### Other links

Request feedback  
Configure schedule and iterations...  
Configure work areas...

17



# Exercise – Visual Studio Team Services

- Sign up for Visual Studio Team Services  
[www.visualstudio.com](http://www.visualstudio.com)
- You need a **Windows Live ID**
- Create a proper **URL**
- Create a **New Project**
- Add **Team Members**
- Give **Access** to the Supervisor

## Account Creation

**Identity Provider**

Windows Live™ ID

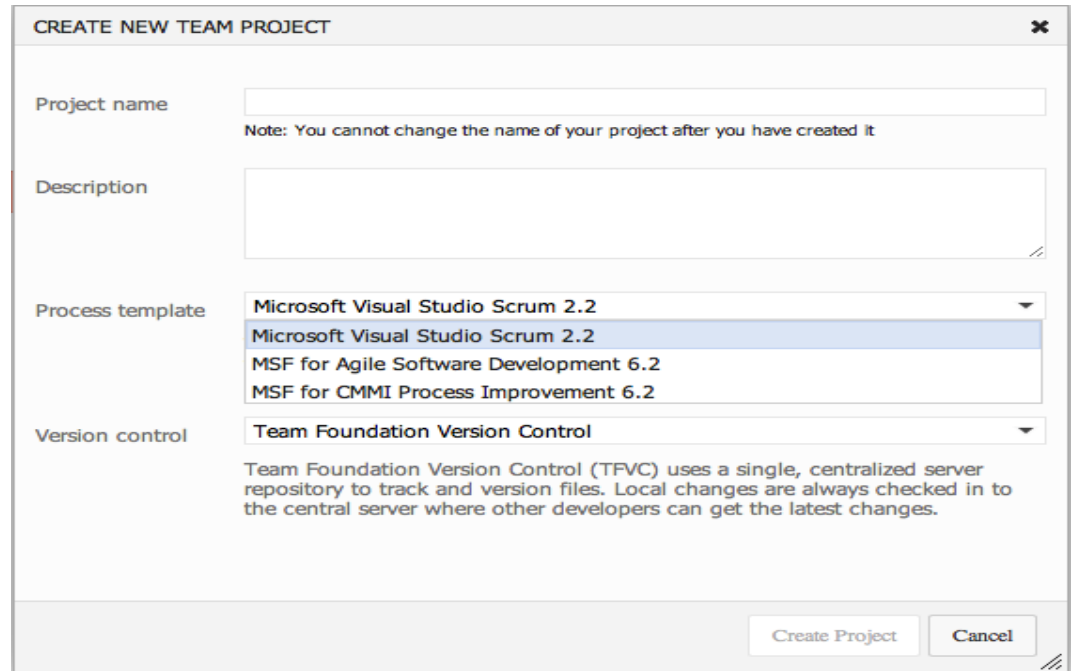
**Account URL**

https://  .visualstudio.com/

Create Account

# Team Projects

- Different templates exists
  - Agile
  - Scrum
  - etc.



The screenshot shows the 'CREATE NEW TEAM PROJECT' dialog box. It has a title bar with a close button (X). The dialog contains the following fields and options:

- Project name:** A text input field. Below it, a note states: 'Note: You cannot change the name of your project after you have created it'.
- Description:** A large text area for entering a project description.
- Process template:** A dropdown menu with the following options:
  - Microsoft Visual Studio Scrum 2.2 (highlighted)
  - Microsoft Visual Studio Scrum 2.2
  - MSF for Agile Software Development 6.2
  - MSF for CMMI Process Improvement 6.2
- Version control:** A dropdown menu with the option: Team Foundation Version Control.

Below the 'Version control' dropdown, there is a descriptive text: 'Team Foundation Version Control (TFVC) uses a single, centralized server repository to track and version files. Local changes are always checked in to the central server where other developers can get the latest changes.'

At the bottom right, there are two buttons: 'Create Project' and 'Cancel'.

# Source Code Control

- Select between
  - **Team Foundation Version Control (TFVC)**
  - Git
- Visual Studio: **Team Explorer**
- From Team Explorer: **Source Control Explorer**

# Create a good structure

- Open the Source Code Explorer in Visual Studio

• D

The screenshot shows the Visual Studio Source Code Explorer interface for a Team Foundation Service project named "Development Project 1". The interface is divided into three main sections: a left sidebar for navigation, a top header, and a main content area.

**Header:** The top blue bar displays "Team Foundation Service / Development Project 1" and the user name "Hans-Petter Halvorsen". Below this, a navigation bar includes tabs for "HOME", "CODE" (selected), "WORK", "BUILD\*", and "TEST\*". The "explorer" sub-tab is also selected.

**Left Sidebar:** This section shows the project structure. It includes "My favorites", "Team favorites", and a tree view of the project. The tree view shows the following structure:

- Development Project 1
  - BuildProcessTemplates
  - Code (selected)
    - Desktop (No items in this folder.)
    - Mobile
    - Server
      - Database
    - Web
  - Documents (No items in this folder.)

**Main Content Area:** This section displays the contents of the selected "Code" folder. The path is "\$/Development Project 1/Code". The "contents" tab is selected, showing a list of items with columns for "Name", "Last Change", and "Comments".

Name	Last Change	Comments
↑ [..]	3 minutes ago	
Desktop	3 minutes ago	
Mobile	3 minutes ago	
Server	3 minutes ago	
Web	3 minutes ago	

# Areas

The different software modules could be divided into different Areas

Control Panel > DefaultCollection > Development Project 1

Hans-Petter Halvorsen ▾

overview

iterations

areas

security

alerts

version control

services

## Areas

[areas](#) show expand all

Select the areas your team owns. Selected areas will determine what shows up on your team's backlog and what work items your team is responsible for.

New

New child

Areas

☒ Development Project 1 default area sub-areas are included

☐ Desktop

☐ Mobile

☐ Server-side

- ☐ Business Logic
- ☐ Database
  - ☐ Stored Proc...
- ☐ Web

# Iterations

Create Iterations for the different releases, milestones (internal and external); e.g., Alpha, Beta, RC, RTM

The screenshot shows the 'Iterations' page in Jira for 'Development Project 1'. The breadcrumb trail is 'Control Panel > DefaultCollection > Development Project 1'. The user 'Hans-Petter Halvorsen' is logged in. The 'iterations' tab is selected in the top navigation bar. Below the navigation bar, the page title is 'Iterations'. There is a link to 'iterations' and buttons for 'show' and 'expand all'. A descriptive text states: 'Select the iterations you want to use for iteration planning (sprint planning). Selected iterations will appear in your backlog view as iterations available for planning.' Below this, there are links for 'New' and 'New child'. A table lists the iterations with columns for 'Iterations', 'Start Date', and 'End Date'. The table has a header row and a body row for 'Development Project 1' with a 'Set dates' link. Below the table, there is a list of iterations: Alpha, Beta, Beta1, Beta2, RC, and RTM, each with a checkbox.

Control Panel > DefaultCollection > Development Project 1 Hans-Petter Halvorsen

overview iterations areas security alerts version control services

### Iterations

[iterations](#) show expand all

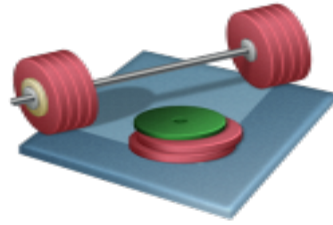
Select the iterations you want to use for iteration planning (sprint planning).  
Selected iterations will appear in your backlog view as iterations available for planning.

New New child

Iterations	Start Date	End Date
Development Project 1 <a href="#">Set dates</a>		Backlog iteration for this team

- ☐ Alpha
- ☐ Beta
  - ☐ Beta1
  - ☐ Beta2
- ☐ RC
- ☐ RTM

# Exercise – TFS



Create the following for your Project:

- Team Project
- Assign Team Members
- Areas
- Iterations
- Create a good folder structure for your Documents and Source Code



# Work Items

In TFS you can create different Work Items, such as:

- **Task**
- **Bug**
- **Feature**
- Scenario
- Issue
- **User Story**
- Test Case
- etc.

These Work Items will be used at different level in your development cycle. When the Testers reports bugs, they will use the “Bug” Work Item, etc.

# Work Items – New Bug

New Bug 1\*: WS is not working

      Copy template URL

Tags [Add...](#)

WS is not working

## STATUS

Assigned To [<No one>](#)  
State [Active](#)  
Reason [New](#)

## CLASSIFICATION

Area [Development Project 1\Desktop](#)  
Iteration [Development Project 1\Beta](#)

## PLANNING

Stack Rank [<None>](#)  
Priority [2](#)  
Severity [3 - Medium](#)

## REPRO STEPS

SYSTEM INFO

TEST CASES

[B](#) [/](#) [U](#)         

## HISTORY

ALL LINKS

ATTACHMENTS

[B](#) [/](#) [U](#)        

## DISCUSSION ONLY

ALL CHANGES

[No entries with comments]

# Queries

- Used to find existing Work Items
- You may create different Queries to make it easy to find the Work Items you need
- Queries may be personal or visible for everybody in the project (Team Queries)

New Query 1

5 work items (1 selected)

results editor

Column Options

Type of Query

Flat List of Work Items

Work Items and Direct Links

Tree of Work Items

Filters for top level work items

	And/Or	Field	Operator	Value
<div><div></div><div></div></div>		Team Project	=	@Project
<div><div></div><div></div></div>	And	Work Item Type	=	[Any]
<div><div></div><div></div></div>	And	State	=	[Any]

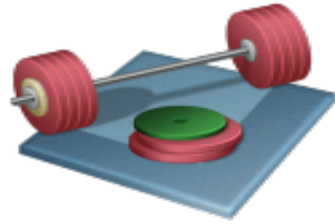
Add new clause

Save query

Column Options

ID	Work Item Type	Title	Assigned To	State	Tags
1	Bug	Database Error	Hans-Pett...	Active	
2	Task	Add Web functionality		New	
4	Test Case	Test Empty Fields	Hans-Pett...	Design	
3	Test Case	Test Web Service	Hans-Pett...	Design	
5	Bug	WS is not working		Active	

# Exercise – TFS Work Items



- Create different kind of **Work Items** for your Project (Task, Bug, Feature)
- Create different **Queries** to make it easy to find your Work Items of interest at any given time.

# Task Board

The screenshot displays the Microsoft Team Foundation Server (TFS) Task Board interface. The top navigation bar includes links for 'home', 'boards', 'backlogs', 'feedback', 'work items', 'source', and 'build'. The 'boards' tab is selected, and the 'board' sub-tab is active. The interface shows a list of tasks organized into two main sections: 'Sprint Planning - Capacity by discipline' and 'Product Backlog: Insert at current location'.

**Sprint Planning - Capacity by discipline**

- QA: Automation: Add activity validation on existing tests** (4 hours, Testing) - Aaron Patterson
- QA: Exploratory testing of interaction between capacity page and capacity** (5 hours, Testing) - Mehmet Argun AL...
- QA: Exploratory testing of discipline assignment on capacity page** (1 hour, Testing) - David Gorena Eliz...
- QA: Exploratory testing of discipline rollout on capacity page** (3 hours, Testing) - Mehmet Argun AL...
- Update server to save capacity by activity** (Aaron Patterson)
- QA: Exploratory testing of interaction between sprint backlog and capacity** (Mehmet Argun Alpa...)
- Data provider for activity** (Aaron Patterson)
- Add activity to capacity page** (Aaron Patterson)
- FieldAggregator client changes** (Aaron Patterson)
- FieldAggregator server changes** (Aaron Patterson)

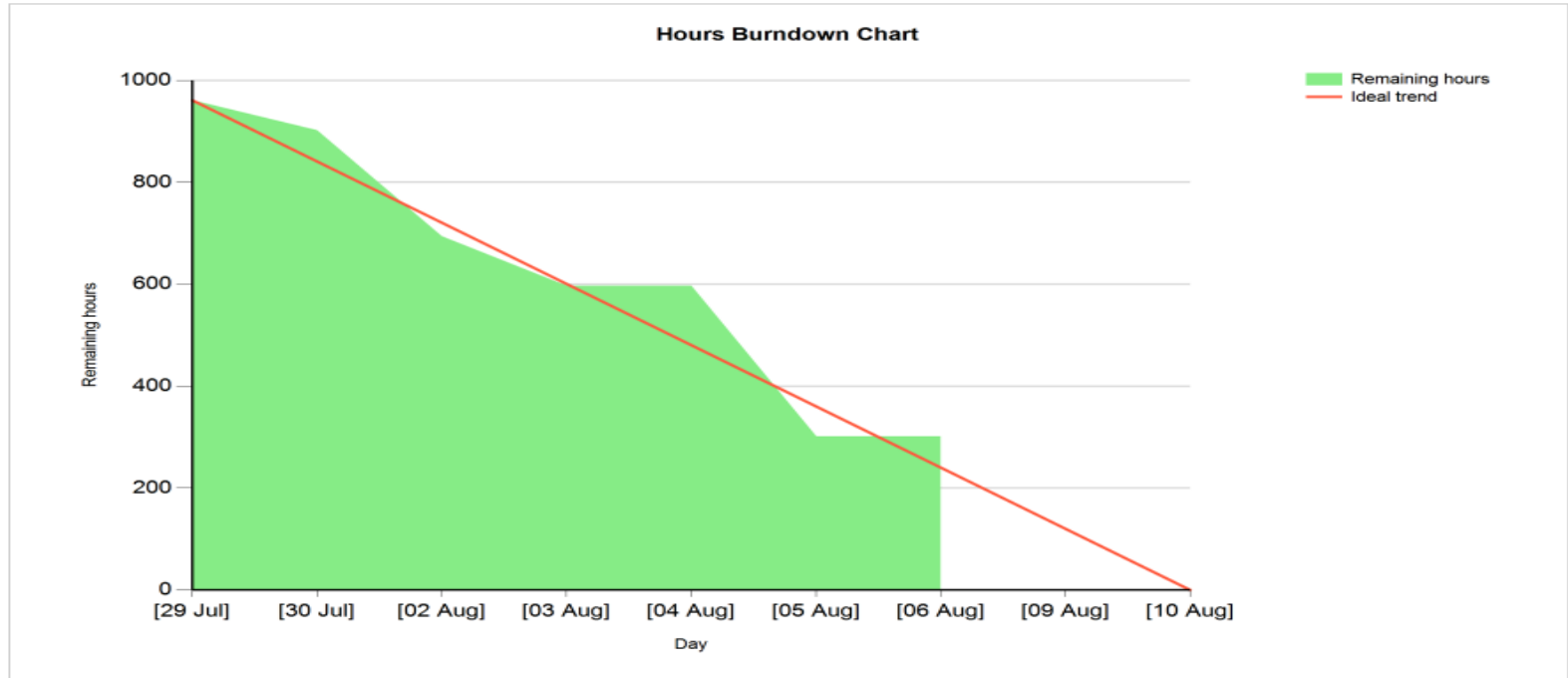
**Product Backlog: Insert at current location**

- Exploratory testing** (4 hours, Testing) - Aaron Patterson
- Modify Add Panel to add at current location** (5 hours, Development) - Aaron Patterson
- Refactor Grid to remove order usage** (5 hours, Development) - Aaron Patterson
- Style changes to Product backlog grid to highlight insertion location** (5 hours, Development) - Aaron Patterson
- Add new and fix existing automation** (12 hours, Testing) - Aaron Patterson

The interface also includes a search bar, a 'PERSON: ALL' filter, and a footer with the text 'Team Foundation Server Web App | About' and '© Microsoft Corporation. All rights reserved.'

# Burndown Chart

Burndown (Sprint 2010.07.27)





# Team Foundation Server

## Client Tools

Hans-Petter Halvorsen, M.Sc.

# Team Foundation Server (TFS)

**Server**

**Clients**

Others

**TFS**

Services

Repository

SQL Server Git  
(TFVC)

Team Foundation Server (TFS) is an  
Application Lifecycle Management (ALM)  
system

**LabVIEW**

MSSCCI Provider

**Eclipse**

**Web Access**

**MS Excel**

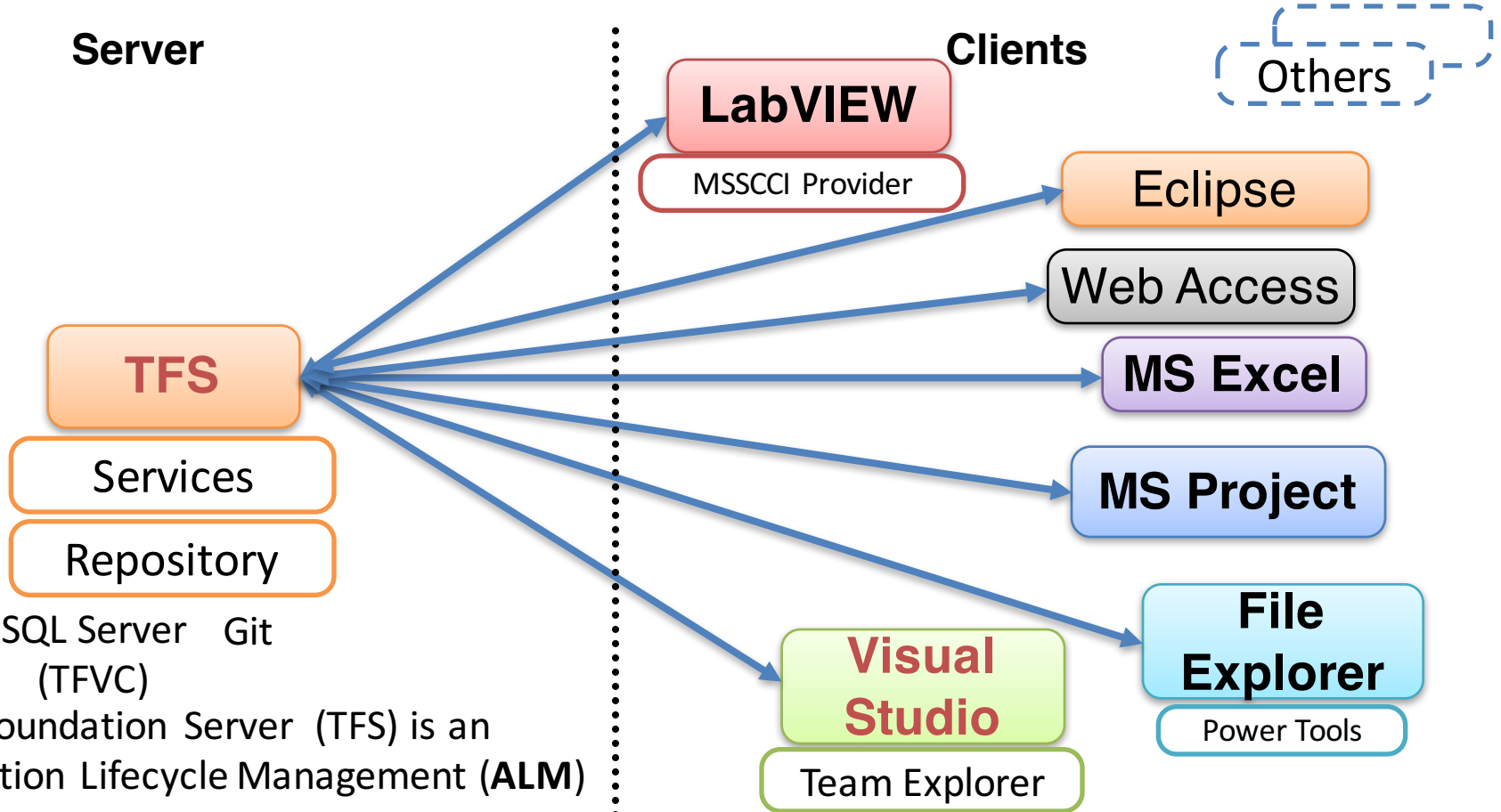
**MS Project**

**Visual  
Studio**

Team Explorer

**File  
Explorer**

Power Tools





# TFS Client Tools

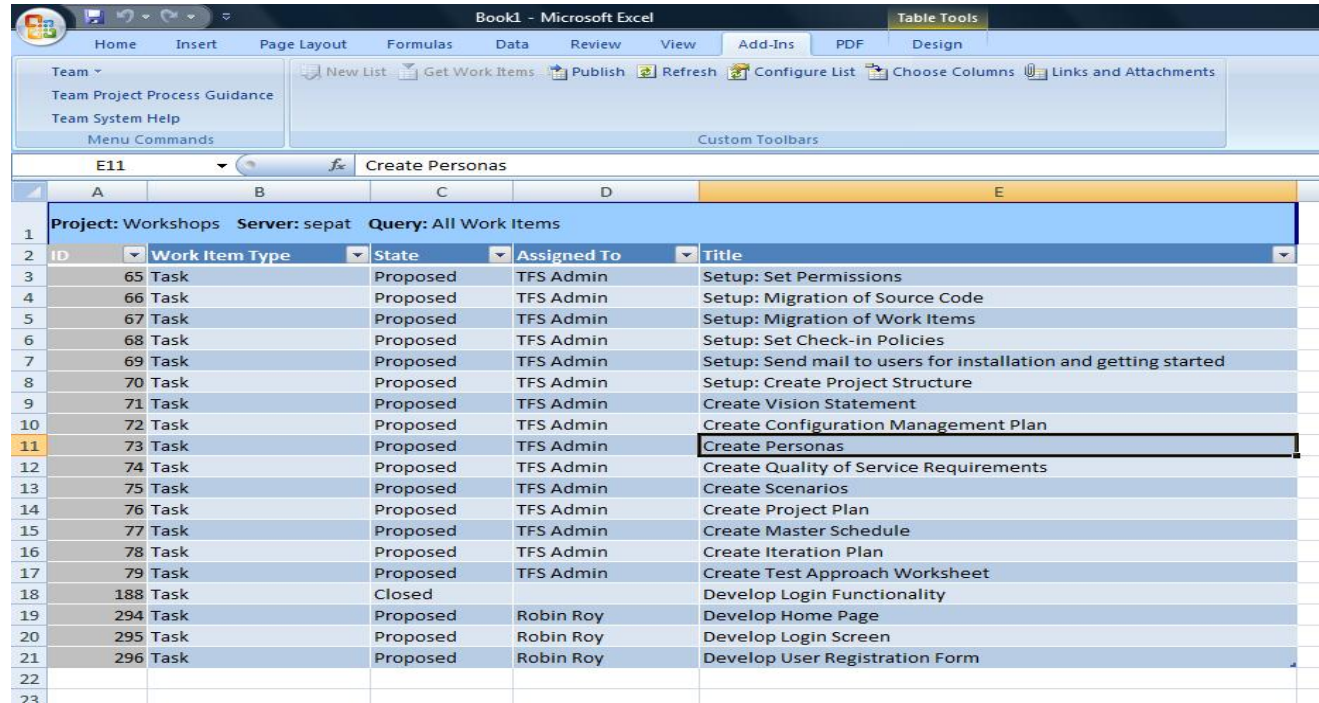
- **Team Explorer** (integrated with Visual Studio)
- **MS Excel Add-in** (part of Team Explorer installation)
- **MS Project Add-in** (part of Team Explorer installation)
- **Windows Explorer Integration** (part of Power Tools)  
Very useful when working with Team Foundation Server outside of Visual Studio or Eclipse
- **TFS MSSCCI Provider** (makes it possible to use TFS within tools that do not support TFS, such as e.g., LabVIEW, SQL Server, etc.)
- Team Explorer Everywhere
- 3.party software that integrates with TFS (using available web services)

# Team Explorer

- The main tool for communication with TFS
- Integrated with Visual Studio
- Add **Source Code**
- Add **Areas** and **Iterations**
- Add **Work Items** (Bugs, Tasks, User Stories, etc)
- Add, Edit and Use **Queries**

# MS Excel Add-in

- Export a work Items (Tasks, Bugs, etc.) to Excel
- Publish Work Items from Excel to TFS



Book1 - Microsoft Excel

Table Tools

Home Insert Page Layout Formulas Data Review View Add-Ins PDF Design

Team >  
Team Project Process Guidance  
Team System Help  
Menu Commands

New List Get Work Items Publish Refresh Configure List Choose Columns Links and Attachments

Custom Toolbars

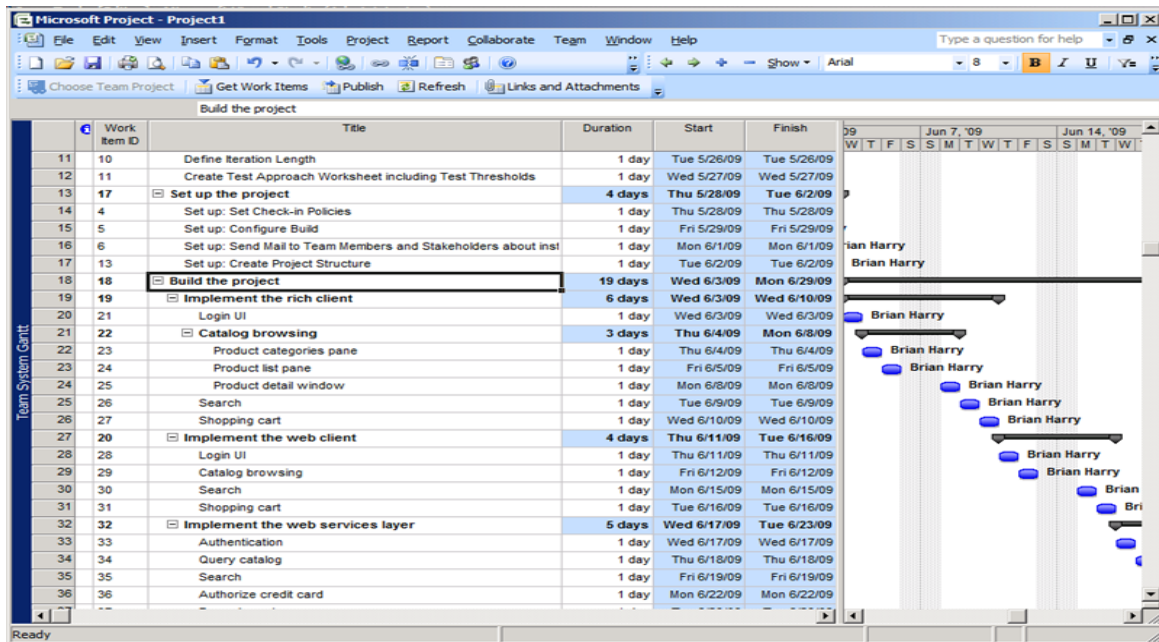
E11 fx Create Personas

Project: Workshops Server: sepat Query: All Work Items

ID	Work Item Type	State	Assigned To	Title
65	Task	Proposed	TFS Admin	Setup: Set Permissions
66	Task	Proposed	TFS Admin	Setup: Migration of Source Code
67	Task	Proposed	TFS Admin	Setup: Migration of Work Items
68	Task	Proposed	TFS Admin	Setup: Set Check-in Policies
69	Task	Proposed	TFS Admin	Setup: Send mail to users for installation and getting started
70	Task	Proposed	TFS Admin	Setup: Create Project Structure
71	Task	Proposed	TFS Admin	Create Vision Statement
72	Task	Proposed	TFS Admin	Create Configuration Management Plan
73	Task	Proposed	TFS Admin	Create Personas
74	Task	Proposed	TFS Admin	Create Quality of Service Requirements
75	Task	Proposed	TFS Admin	Create Scenarios
76	Task	Proposed	TFS Admin	Create Project Plan
77	Task	Proposed	TFS Admin	Create Master Schedule
78	Task	Proposed	TFS Admin	Create Iteration Plan
79	Task	Proposed	TFS Admin	Create Test Approach Worksheet
188	Task	Closed		Develop Login Functionality
294	Task	Proposed	Robin Roy	Develop Home Page
295	Task	Proposed	Robin Roy	Develop Login Screen
296	Task	Proposed	Robin Roy	Develop User Registration Form

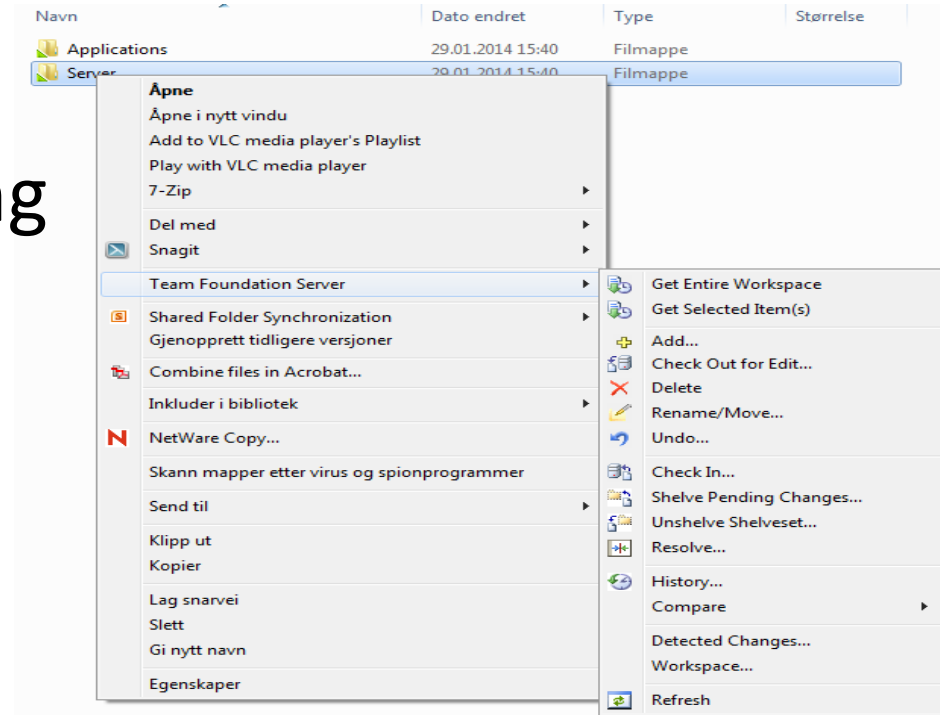
# MS Project Add-in

- Combine Gantt Charts and TFS
- The Tasks in the Gantt Chart will be created/updated in TFS as well



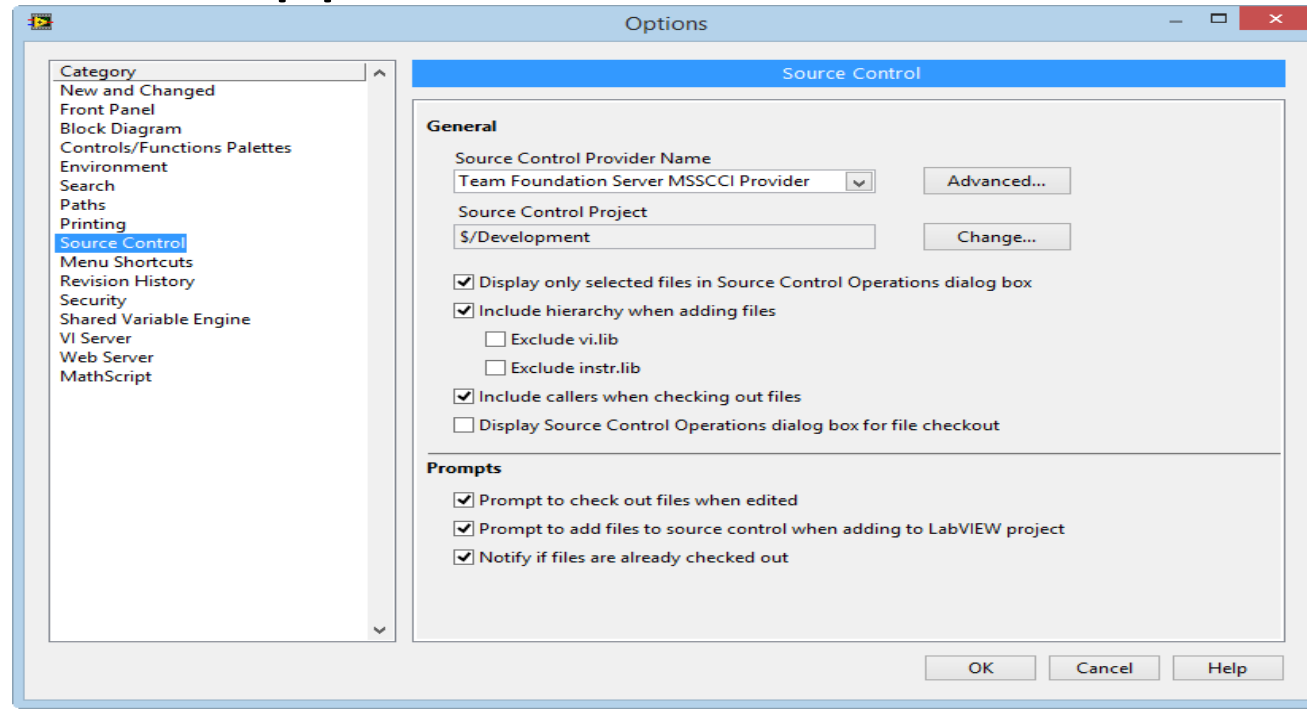
# Windows Explorer Integration

- Part of the “TFS Power Tools” package
- Very useful when working with Team Foundation Server outside of Visual Studio or Eclipse (using tools with no TFS integration)

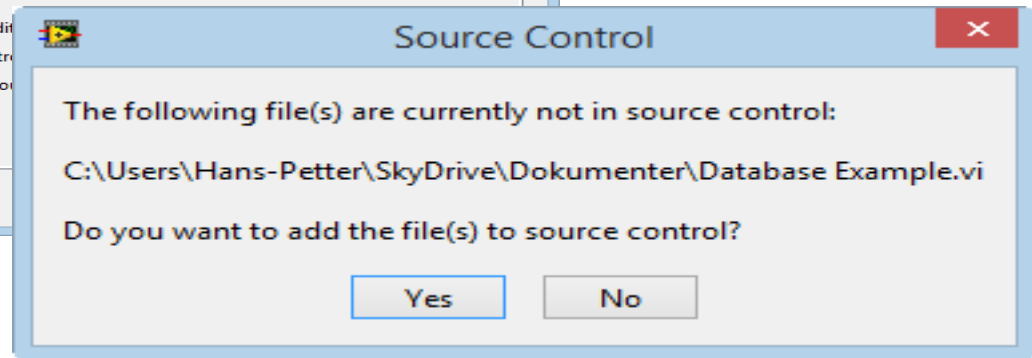
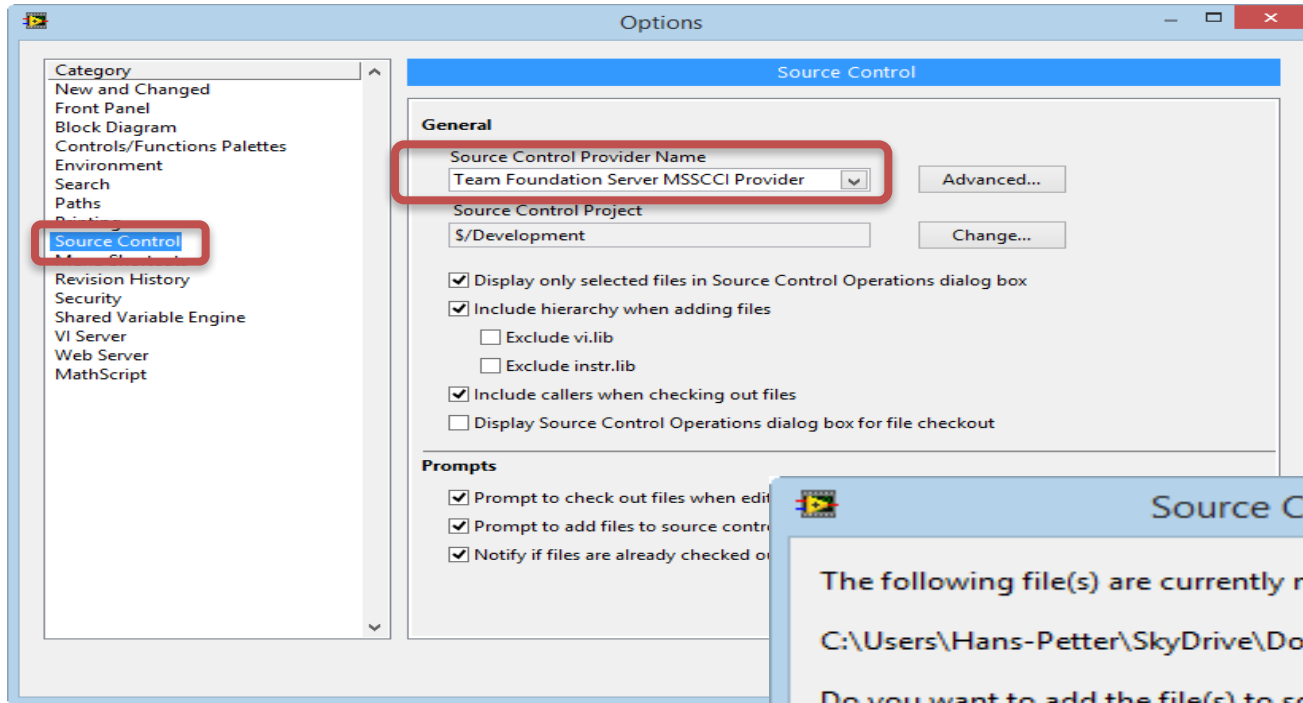


# TFS MSSCCI Provider

- Use TFS in e.g. LabVIEW, SQL Server, etc. that dont have built-in support for TFS

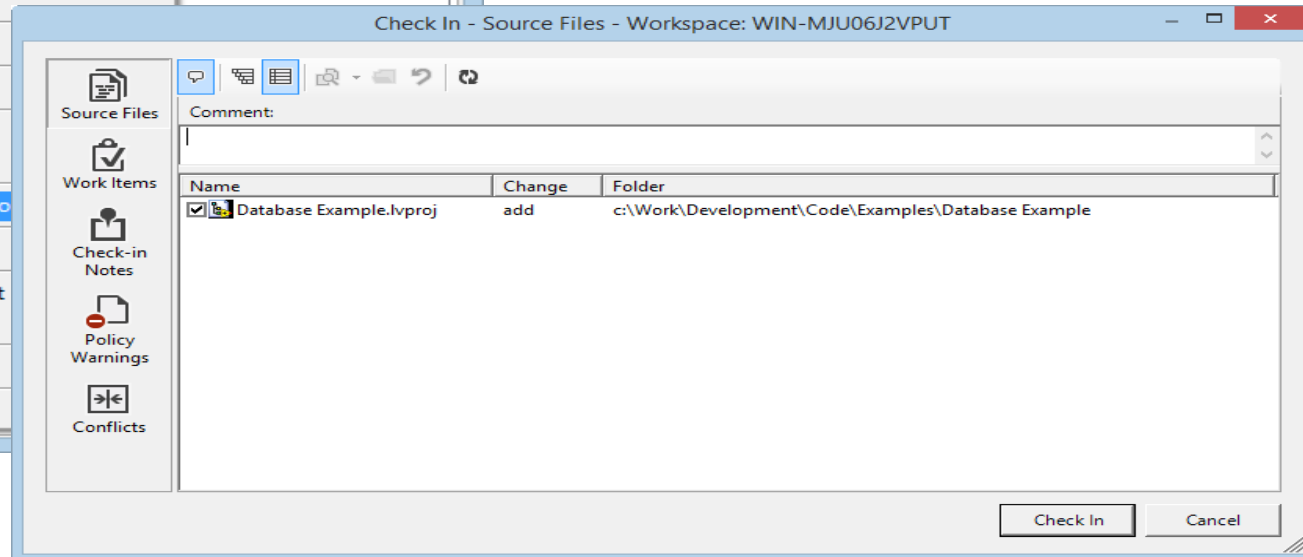
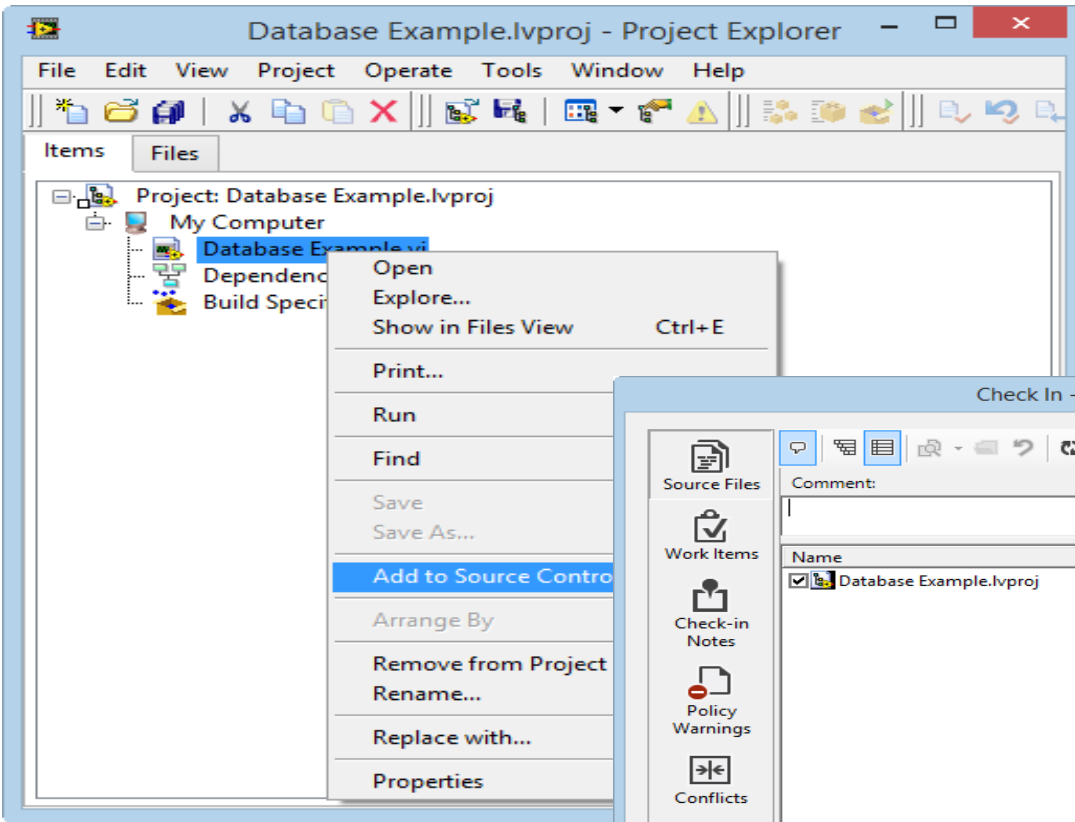


# TFS in LabVIEW - Configuration



# TFS in LabVIEW – Check In/Out

You need to use the  
**Project Explorer**







# Team Foundation Server

## Visual Studio

How-To use TFS/Visual Studio Online Features inside Visual Studio

Hans-Petter Halvorsen, M.Sc.

# Using TFS with Visual Studio

- Connect to TFS
- Add Source Code to TFS
- Check In/Check Out
- Team Explorer
- Source Control Explorer

# TFS in Visual Studio

The screenshot displays the Visual Studio interface with the 'TEAM' menu highlighted in the top menu bar. The 'Source Control Explorer' window is open on the left, showing a tree view of the project structure. The 'Team Explorer' window is open on the right, showing the 'Pending Changes' section. The 'Source Control Explorer' window has a title bar that says 'Source Control Explorer - Microsoft Visual Studio' and a toolbar with icons for source control operations. The 'Team Explorer' window has a title bar that says 'Team Explorer - Pending Changes' and a search bar for work items.

Source Control Explorer - Microsoft Visual Studio

FILE EDIT VIEW DEBUG TEAM TOOLS TEST ANALYZE WINDOW HELP

Source location: Development

Workspace: WIN-MJU06J2VPUT

Local Path: C:\Work\Development

Name	Pending Change	User	Latest	Last Check-in
BuildProcessTemplates			Yes	2013-04-26 12:...
Code			Yes	2013-05-31 12:...
Documents			Yes	2013-06-20 10:...
Project Management			Yes	2014-01-15 9:2...

Pending Changes | Development

Changeset 149 successfully checked in.

Check In Shelve Actions

Comment

Enter a check-in comment

Related Work Items

Queries Add Work Item by ID

Drag work items here to link them to the check-in.

Included Changes

Exclude All View Options

There are no pending changes.

Excluded Changes

Include All View Options

There are no pending changes.

Solution Explorer Team Explorer

```
-top: 60px;  
-bottom: 40px;
```

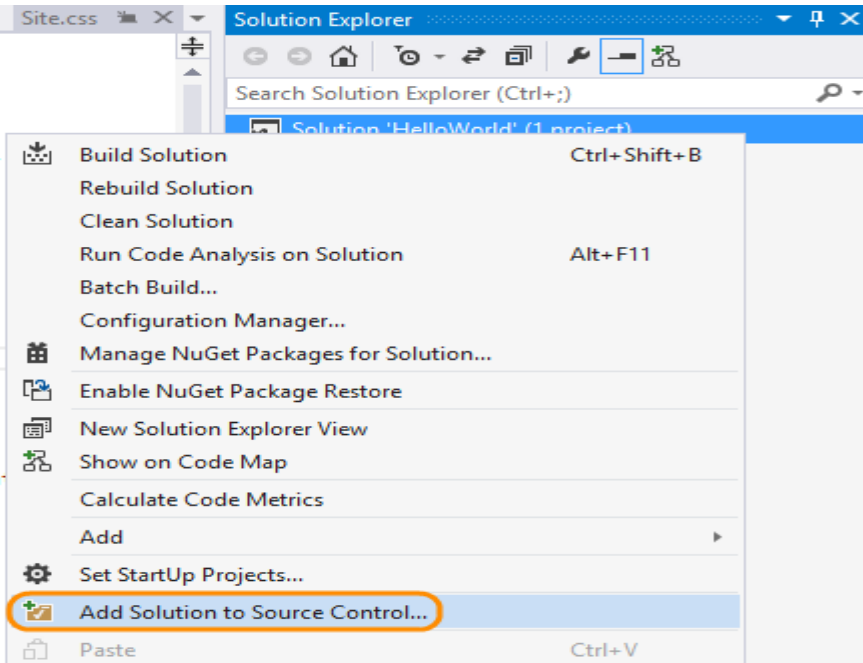
```
or validation helpers */  
.validation-error {  
    #b94a48;
```

```
.validation-valid {  
    : none;
```

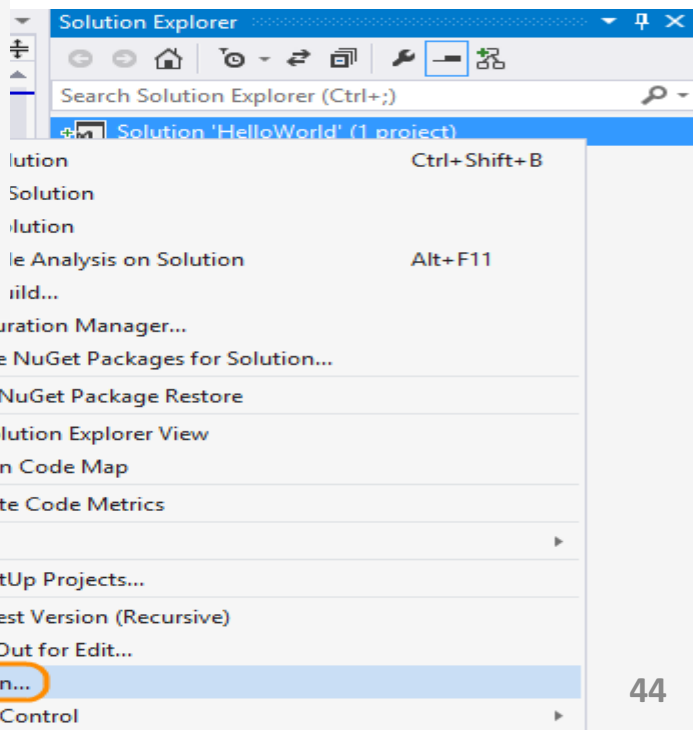
```
.validation-error {  
    1px solid #b94a48;
```

```
['checkbox'].input-validation {  
    : none;
```

```
.summary-errors {  
    #b94a48;
```



# Add Source Code



## Check In

```
.validation-error {  
    1px solid #b94a48;
```

```
['checkbox'].input-validation {  
    : none;
```

```
.summary-errors {  
    #b94a48;
```

```
.summary-valid {  
    : none;
```

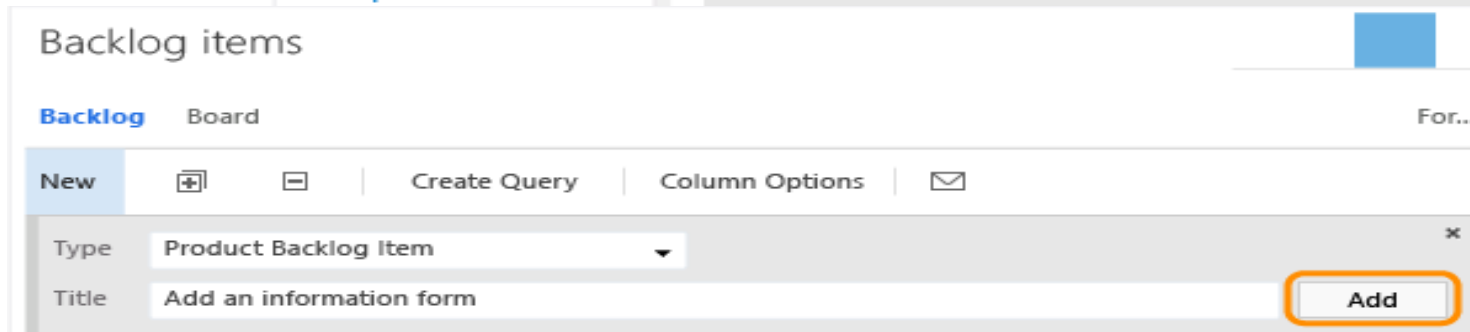
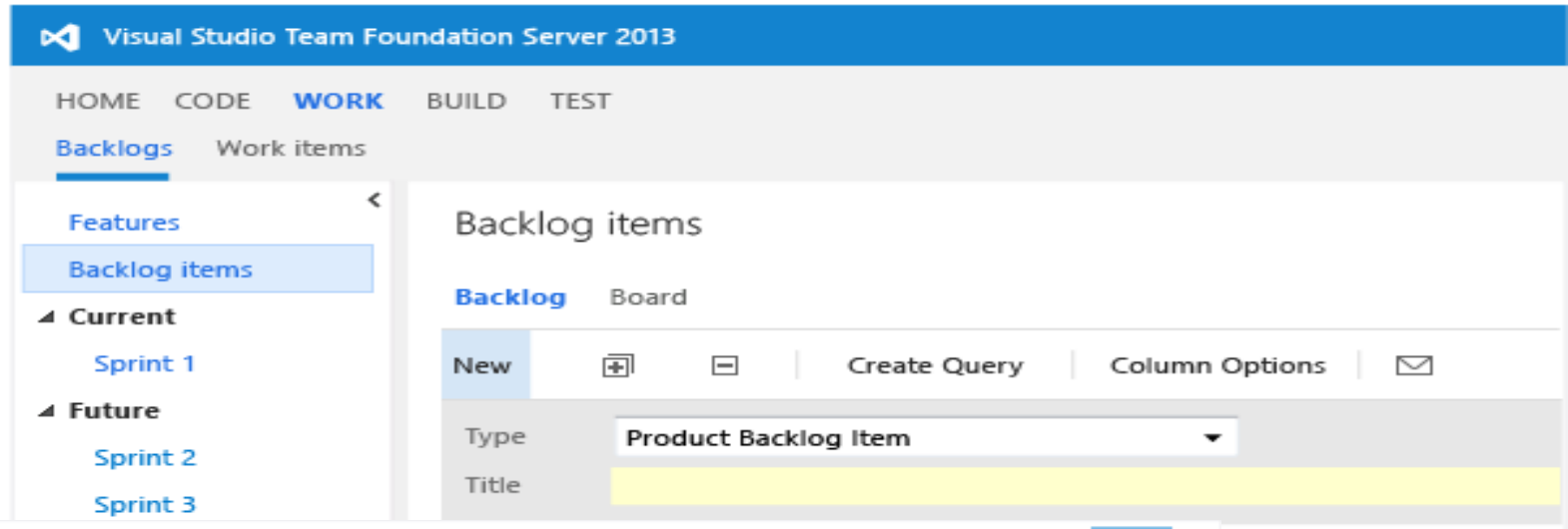


# Scrum in TFS

Product Backlog, Sprint Backlog, Task board, Burndown Chart

Hans-Petter Halvorsen, M.Sc.

# Using TFS to create the Backlog



<http://msdn.microsoft.com/en-us/library/ee518933.aspx>

# Sprint Backlog in TFS

HOME CODE **WORK** BUILD TEST

**Backlog items** Work items

Product Backlog

▲ **Current**

Sprint 1

▲ **Future**

Sprint 2

Sprint 3

Sprint 4

Sprint 5

Sprint 6

Backlog items

**Backlog** Board

New [Add] [Remove] Create Query Column Options [Email]

Type Product Backlog Item

Title

Order	Work Item Type	Title	State
1	Product Backlo...	Hello World Web Site	New
2	Bug	Slow response on welcome...	New
3	Product Backlo...	Change initial view	New
4	Product Backlo...	Add an information form	New
5	Product Backlo...	Welcome back	New
6	Product Backlo...	Resume	New
7	Product Backlo...	Interim save on long forms	New

# Break items down into Tasks

In the sprint backlog, add a task:

Sprint 1

April 29 - May ...  
12 work days remain...

Backlog

Board

Capacity

Create Query

Column Options

	Title	State	Assigned
+	Hello World Web Site	New	
+	Add an information form		
+	Change initial view		
+	Welcome back		

New Task 1\*: Welcome Screen

Copy template URL

Welcome Screen

IterationFabrikamFiber\Release 1\Sprint 1

STATUS	DETAILS
Assigned To	Remaining Work2
StateTo Do	Backlog Priority
ReasonNew task	Activity
Blocked	AreaFabrikamFiber

DESCRIPTION

HISTORYLINKSATTACHMENTS



# Final Results:

The screenshot displays a web application interface for project management. At the top, there is a navigation bar with tabs: HOME, CODE, WORK (highlighted), BUILD, and TEST. Below this, a sub-navigation bar shows 'Backlog item' and 'Work items'. The left sidebar contains a tree view with 'Features' and 'Product Backlog' as main categories. Under 'Product Backlog', there is a section for 'Current' sprints, with 'Sprint 1' selected and highlighted. Below this, under 'Future', are 'Sprint 2' through 'Sprint 6'. The main content area is titled 'Sprint 1' and has three tabs: 'Backlog' (selected), 'Board', and 'Capacity'. Below the tabs, there are icons for adding items, a minus sign, and buttons for 'Create Query', 'Column Options', and an email icon. The backlog list shows a hierarchy of tasks. The first item is 'Hello World Web Site' (blue icon), which is expanded to show sub-items: 'Welcome Screen', 'Change background color', and 'About Screen' (all with yellow icons). Below this are three more items, each with a green plus icon and a blue icon: 'Slow response on welcome page' (with a red icon), 'Change initial view', and 'Add an information form'. The 'Add an information form' item is expanded to show 'Auto-complete user's name in form if logged in' and 'Auto-save' (both with yellow icons). The final item is 'Welcome back' (with a blue icon).

HOME CODE **WORK** BUILD TEST

Backlog item Work items

Features  
Product Backlog

Current

Sprint 1

Future

Sprint 2  
Sprint 3  
Sprint 4  
Sprint 5  
Sprint 6

Sprint 1

Backlog Board Capacity

Create Query Column Options

Title

- + Hello World Web Site
  - Welcome Screen
  - Change background color
  - About Screen
- + Slow response on welcome page
  - Rework opening animation
- + Change initial view
- + Add an information form
  - Auto-complete user's name in form if logged in
  - Auto-save
- + Welcome back

# Use the Taskbord to update Tasks

The task board is at the heart of daily standups. Move tasks on the task board to reflect their current state.



# Use the Taskbord to update Tasks

You can assign a task to a specific person:

The screenshot displays the Jira Taskbord for 'Sprint 1' (April 29 - May 17, 11 work days remaining). The interface includes tabs for 'Backlog', 'Board' (selected), and 'Capacity'. A 'Group by' dropdown is set to 'Backlog items'. The tasks are organized into columns: 'TO DO 10 h' and 'IN PROGRESS 5 h'. A dropdown menu is open over the 'Unassigned' task in the 'TO DO' column, listing team members for assignment: Johnnie McLeod, Christie Church (highlighted), Nicole Zamora, Raisa Pokrovskaya, Francis Totten, and Jamal Hartnett. The tasks are as follows:

Task	Hours	Assignee	Status
Add an information form	7 h	Unassigned	TO DO
Welcome Back	2 h	Unassigned	TO DO
Hello World Web Site	6 h	Johnnie McLeod	TO DO
Auto-save	Unassigned	Unassigned	TO DO
Auto-complete user's name in form if logged in	3 h	Jamal Hartnett	IN PROGRESS
Welcome Screen	2 h	Johnnie McLeod	IN PROGRESS
Change Background Color	2 h	Christie Church	IN PROGRESS
About Screen	2 h	Nicole Zamora	IN PROGRESS

# Use the Taskbord to update Tasks

Update the remaining work by either using the drop-down list or typing a specific value:

Sprint 1

April 29 - May 17  
11 work days remaining

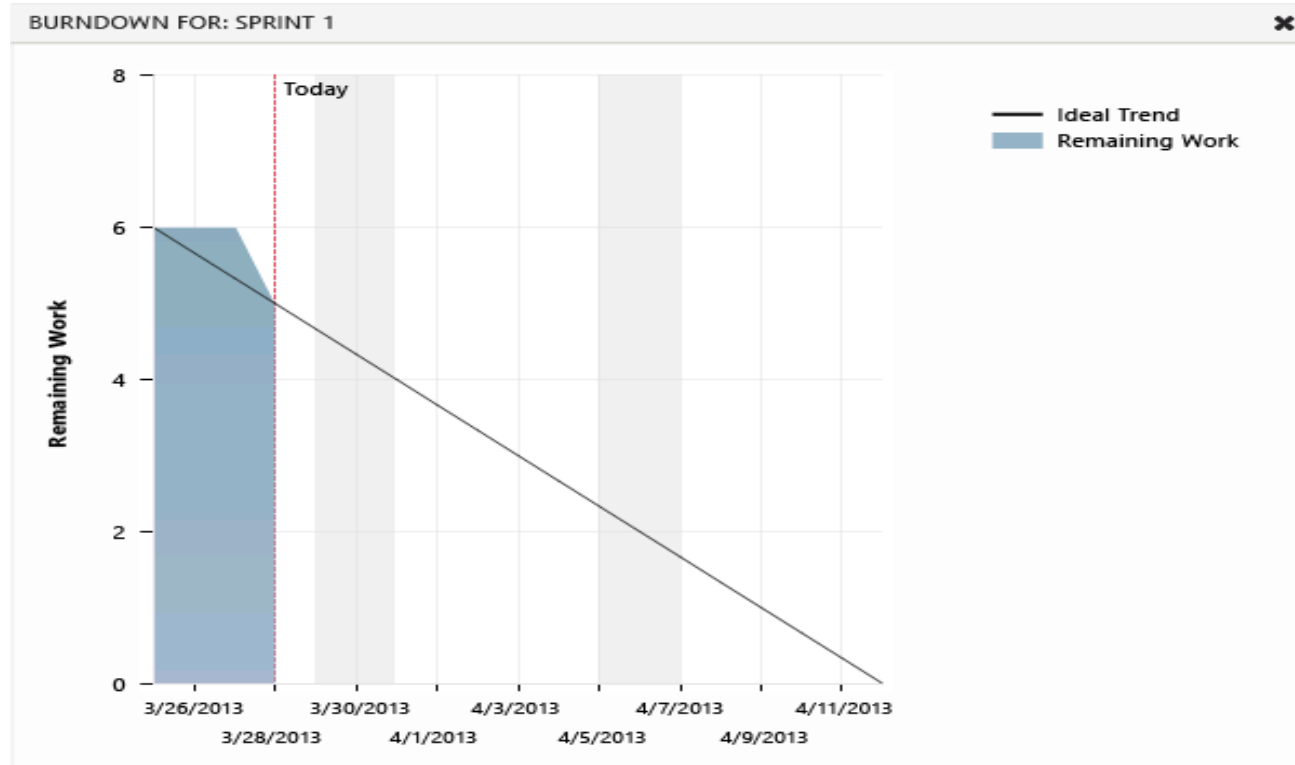
BacklogBoardCapacity

Group by Backlog itemsPerson...

	TO DO 10 h	IN PROGRESS 5 h
Add an information form 7 h	<div>Auto-save</div> <div>4istie Church...</div>	<div>Auto-complete user's name in form if logged in</div> <div>3Jamal Hartnett (...)</div>
Welcome Back 2 h	<div>...active text me Back</div> <div>1</div> <div>0.5kaya (Fabrik...</div>	
Hello World Web Site 6 h	<div>Welcome Screen</div> <div>2Johnnie McLe...</div>	<div>Change Background Color</div> <div>2Christie Chur...</div>
		<div>About Screen</div> <div>2Nicole Zamo...</div>

# Burndown Chart

Review overall progress by opening the burndown chart for the sprint:



# Summary



- We have lots of different **SCC** systems
- We have lots of different **ALM** systems
- If you are using Visual Studio, Team Foundation Server is a good choice
- Team Foundation Server is tightly integrated into Visual Studio
- TFS can be used with other IDEs as well, e.g. Eclipse (Team Explorer add-in), LabVIEW (MSSCCI Provider), etc.

SCC – Source Code Control

ALM – Application Lifecycle Management

# References



- E. Blankenship, M. Woodward, G. Holliday, and B. Keller, *Professional Team Foundation Server 2012*: Wiley, 2013.
- M. Gousset, B. Keller, and M. Woodward, *Professional Application Lifecycle Management with Visual Studio 2012*: Wiley, 2013.
- I. Sommerville, *Software Engineering*: Pearson, 2010.
- S. Adams. *Dilbert*. Available: <http://dilbert.com>
- O. Widder. (2013). *geek&poke*. Available: <http://geek-and-poke.com>
- B. Lund. (2013). *Lunch*. Available: <http://www.lunchstriper.no>,  
<http://www.dagbladet.no/tegneserie/lunch/>

Hans-Petter Halvorsen, M.Sc.



University College of Southeast Norway

[www.usn.no](http://www.usn.no)

E-mail: [hans.p.halvorsen@hit.no](mailto:hans.p.halvorsen@hit.no)

Blog: <http://home.hit.no/~hansha/>

