Agile Software Development

Tobias Almén
Training & Simulation

Agile Manifesto

- **Individuals and interactions** over processes and tools
- **Working software** over comprehensive documentation
- **Customer collaboration** over contract negotiation
- **Responding to change** over following a plan

How to become agile?

If you have Control you *might* be Agile!

- Take Control of your
  - Requirements
  - Design
  - Source Code
  - Builds
  - Quality

- Automate everything that can be automated
- Don’t use an agile method, make your method agile
- Visualize project status
- Keep the team close together – communication-wise
- There is always a better and faster way to do things

Take Control of your Requirements

- Understand what to deliver
- Do not do things that are not required
- Be prepared to change your requirements at all times
- Refine your requirements iteratively as a part of the development process

Take Control of your Design

- Design for change!
- With monoliths it is hard to be agile
- Create modules with well defined interfaces
  - Interfaces are abstract
  - Module contains logically related code

Take Control of Your Source Code

- Use Source Code Control!
- A well defined branching policy is crucial
  - Handle releases and maintenance of releases
  - Development branches for larger changes
- Check in often
- You must be able to recreate released versions

Branching policy – Basic

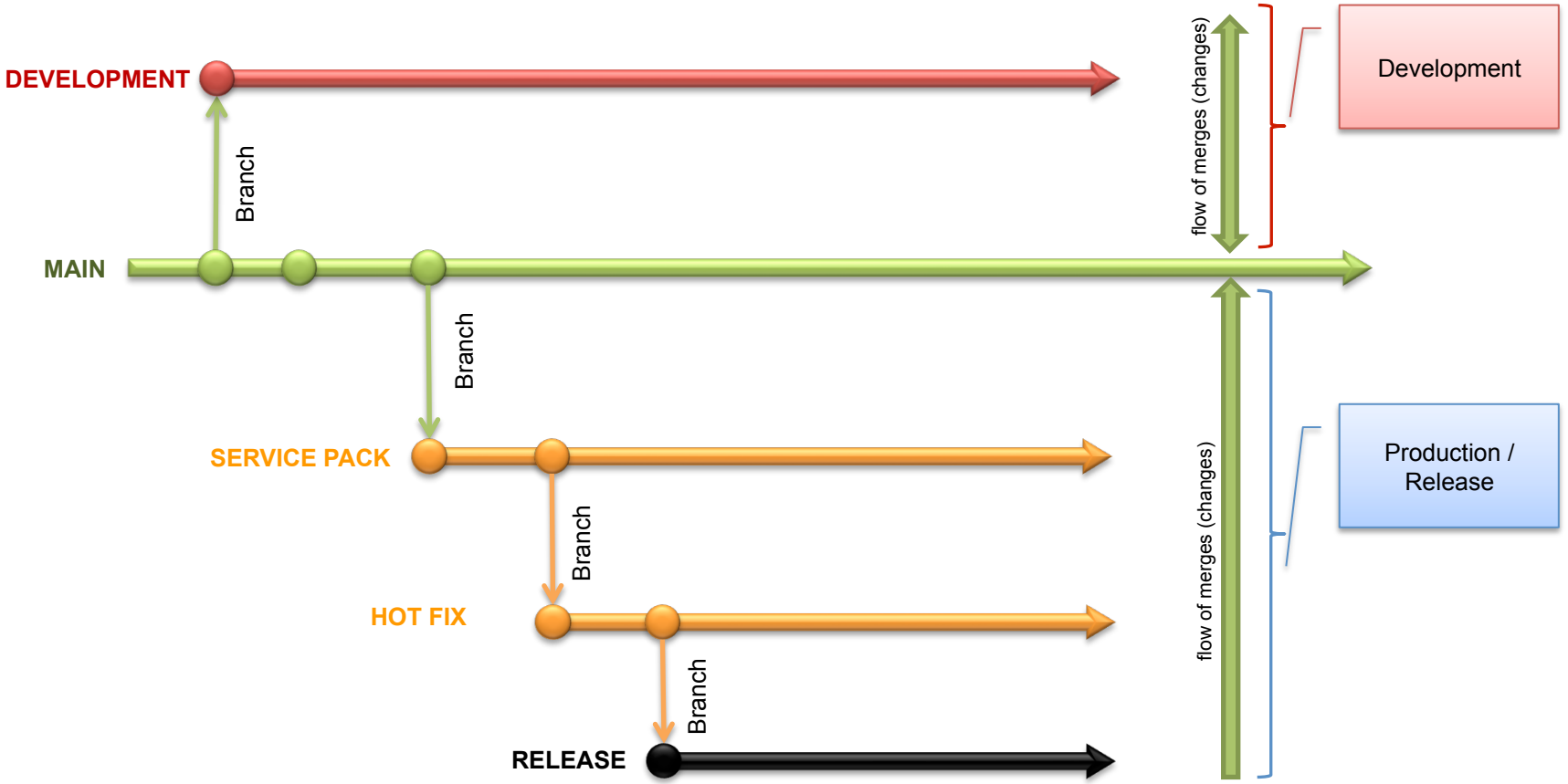
- Single development team, continuous releases

Branching policy – Standard

- Single team, continuous releases and service packs

Branching policy – Advanced

- Single team, continuous releases, service packs and hot fixes



Take Control of your Builds

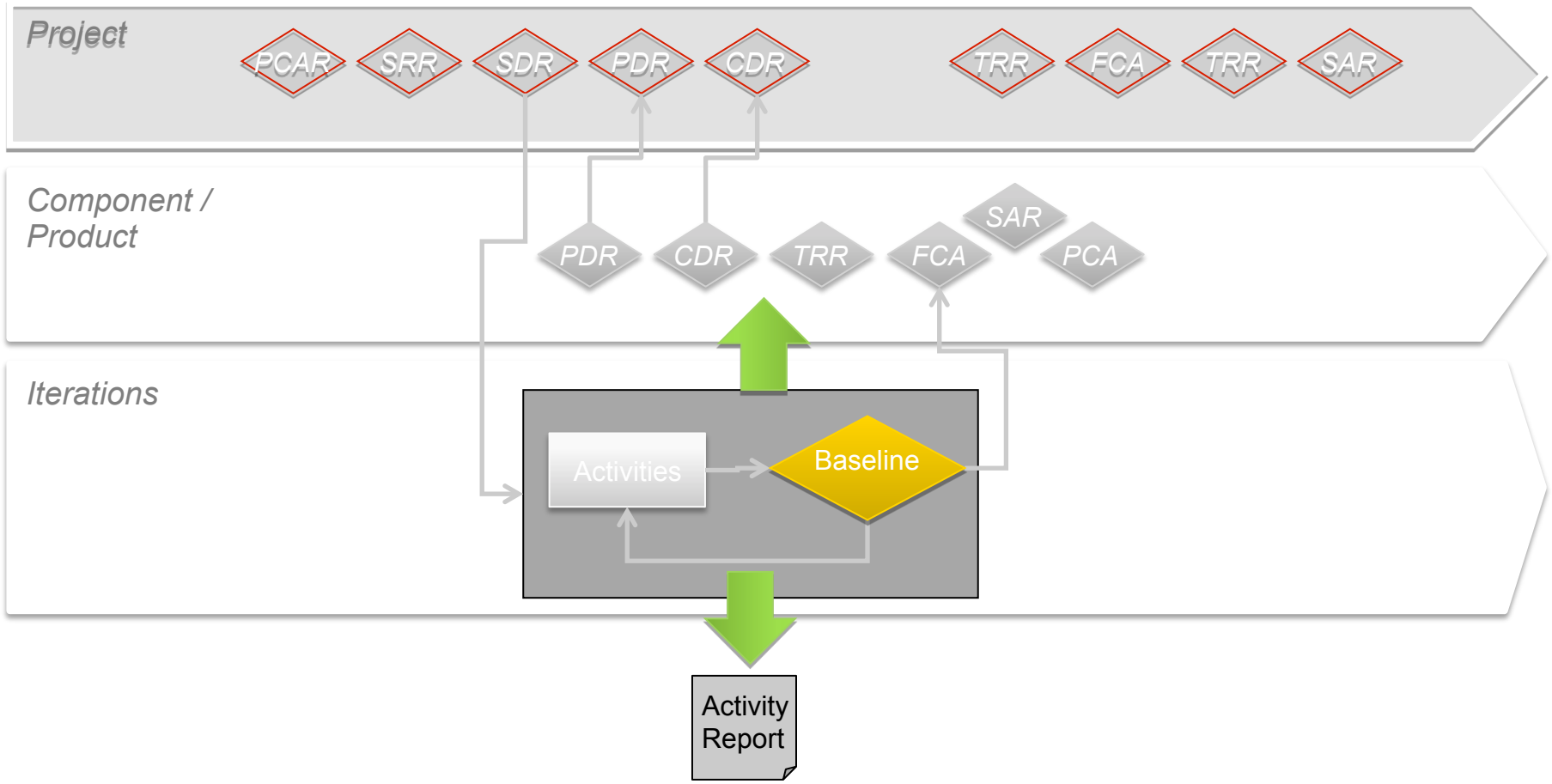
- Automate, automate, automate!
- Everyone should be able to build and run the code
- Visualize build results
  - Wallboard
  - Email
  - Easy-to-read build logs
- Use frequent builds (daily, weekly)

Take Control of your Quality

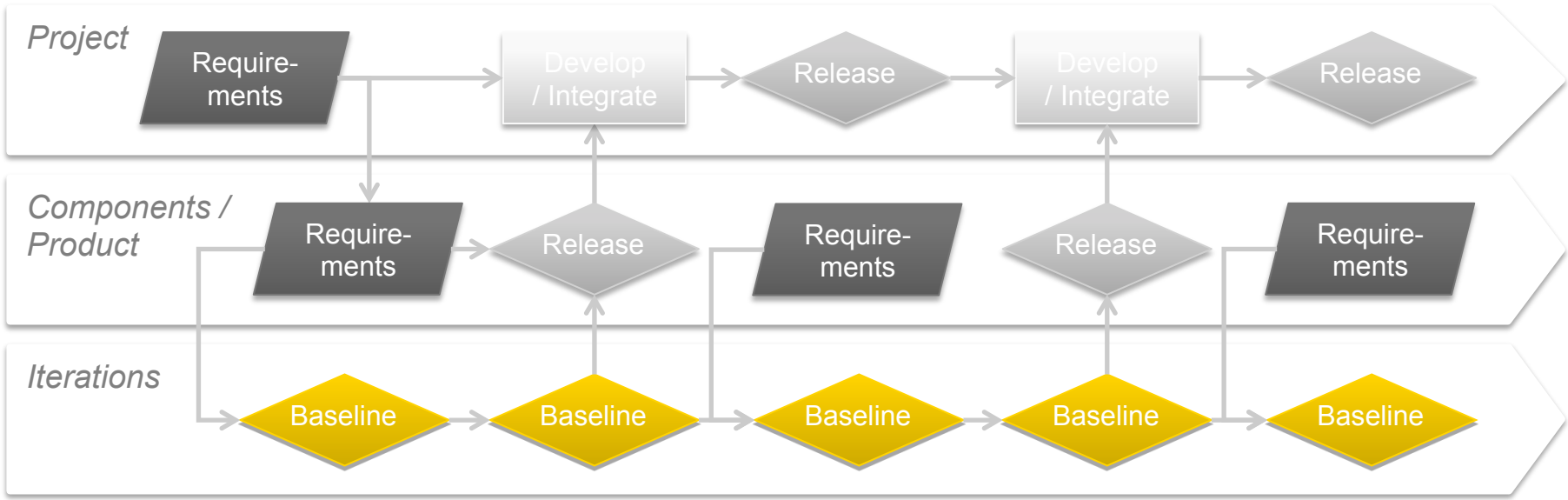
- Automate, automate, automate!
- Build errors have top priority!
- Testing is more important than coding
- Do automated test as a part of the build process
- Ensure code coverage
- Use installation packages or merge modules for every piece of software.
- No program is too small to have an installation package

How we do it!

SW DEVELOPMENT PROCESS

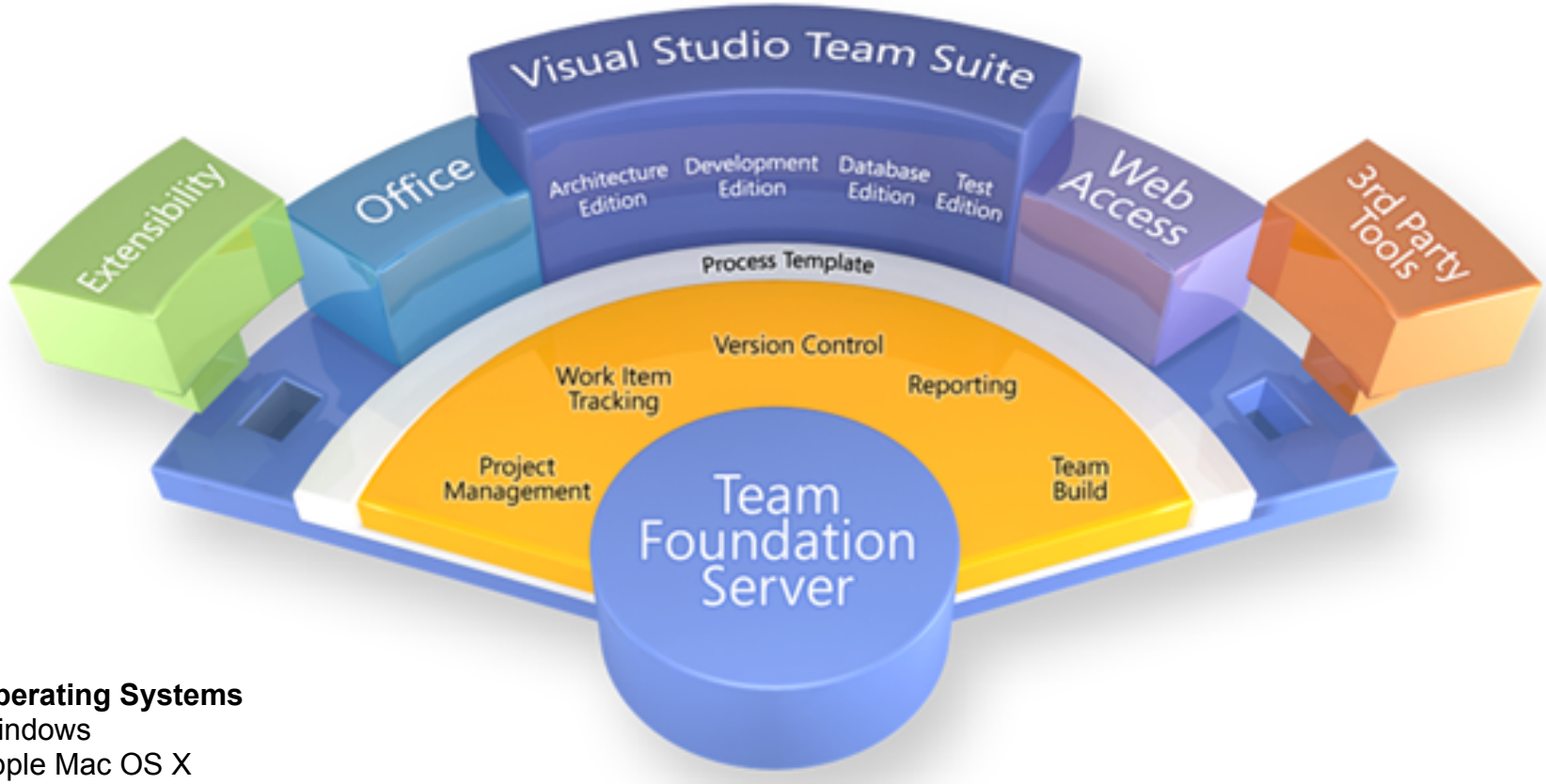


RELEASES PROCESS



Each baseline is about 2-4 weeks

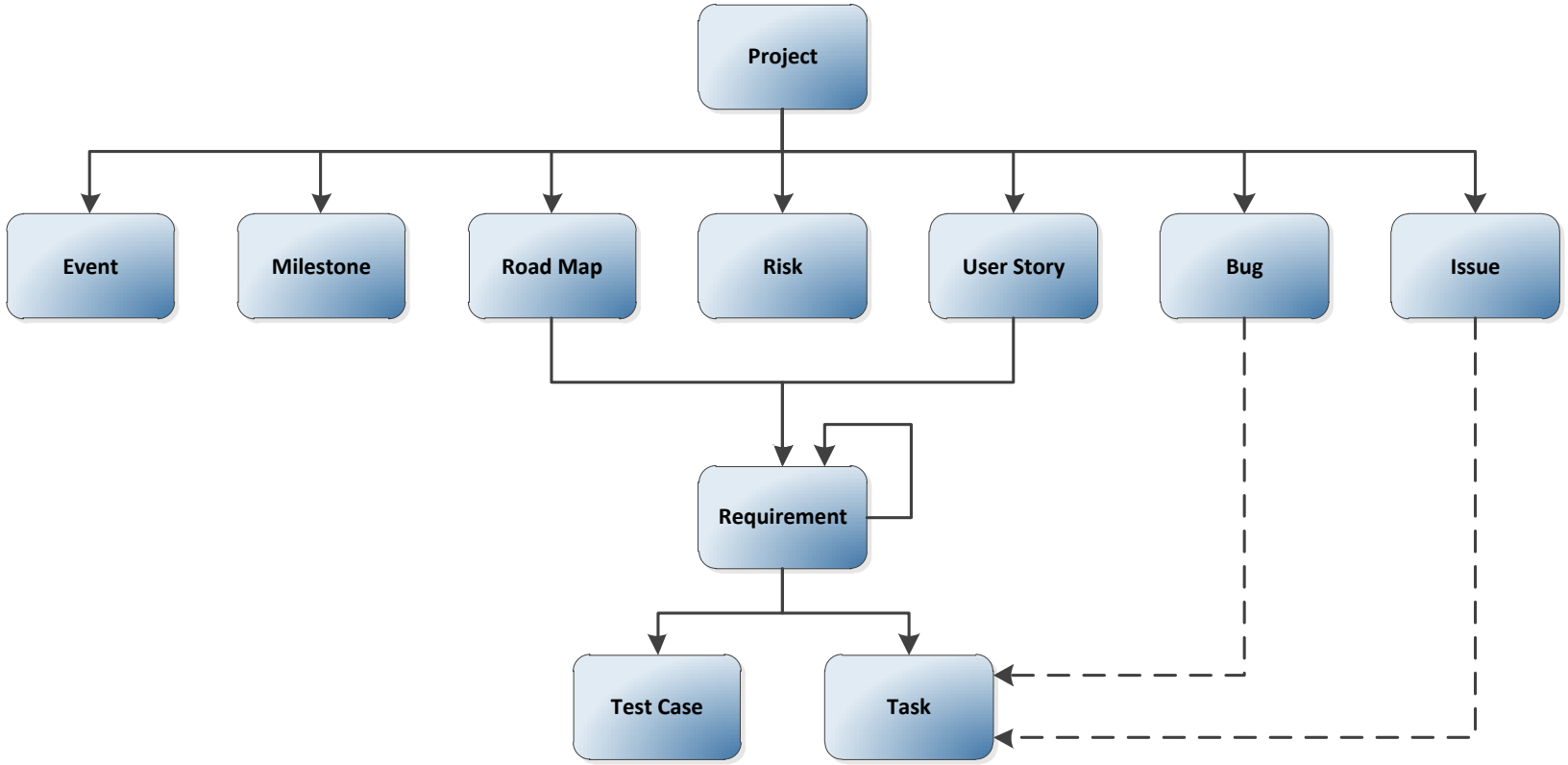
Team Foundation Server 2010



Operating Systems
Windows
Apple Mac OS X
HP-UX
IBM AIX
Linux
Solaris
Unix

IDE:s
Visual Studio
Eclipse
IBM Rational Application Developer

Team Foundation Work Items



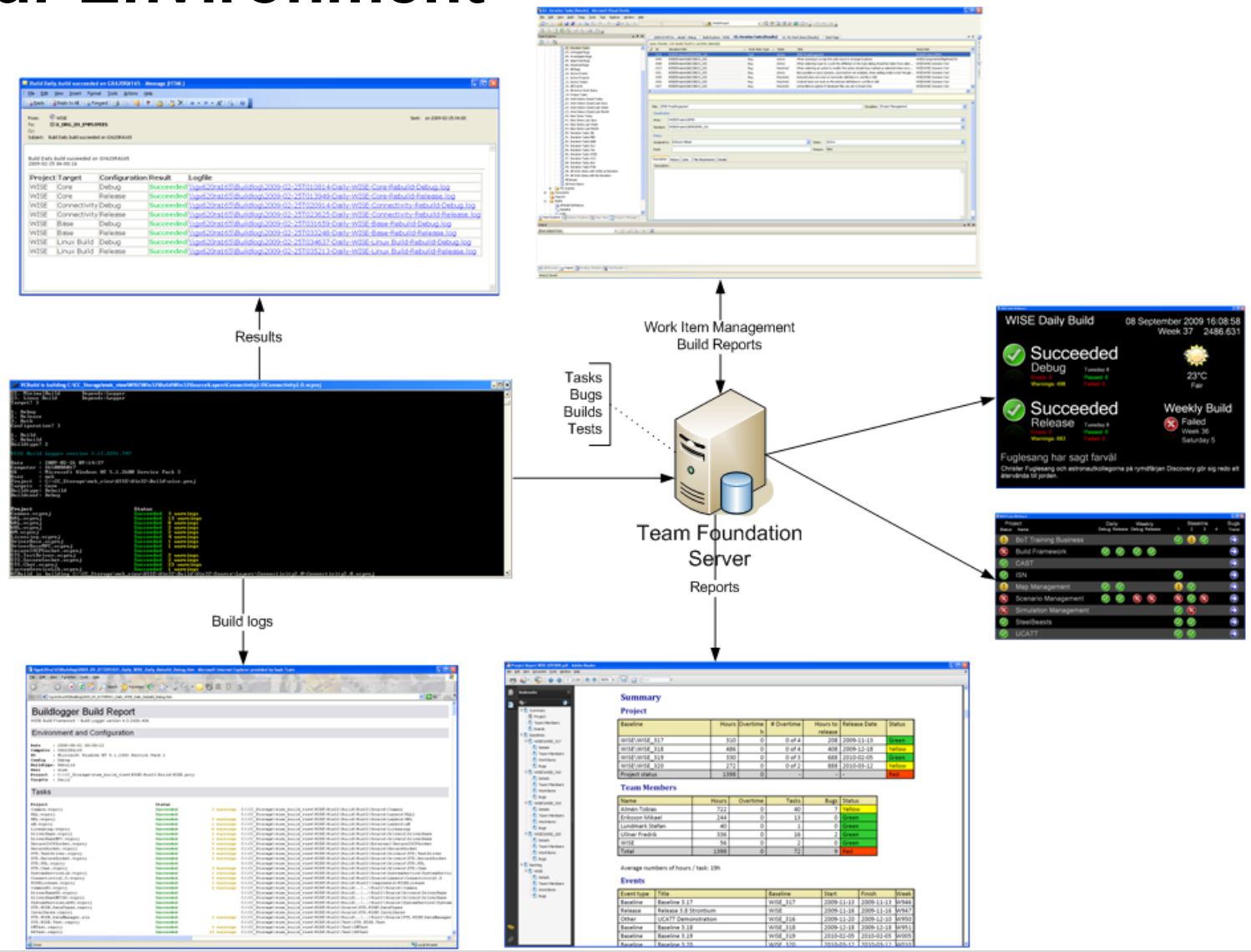
- These work items are adapted to our needs

BaTS – Build and Test Solution

- A set of tools that adapts and extends TFS

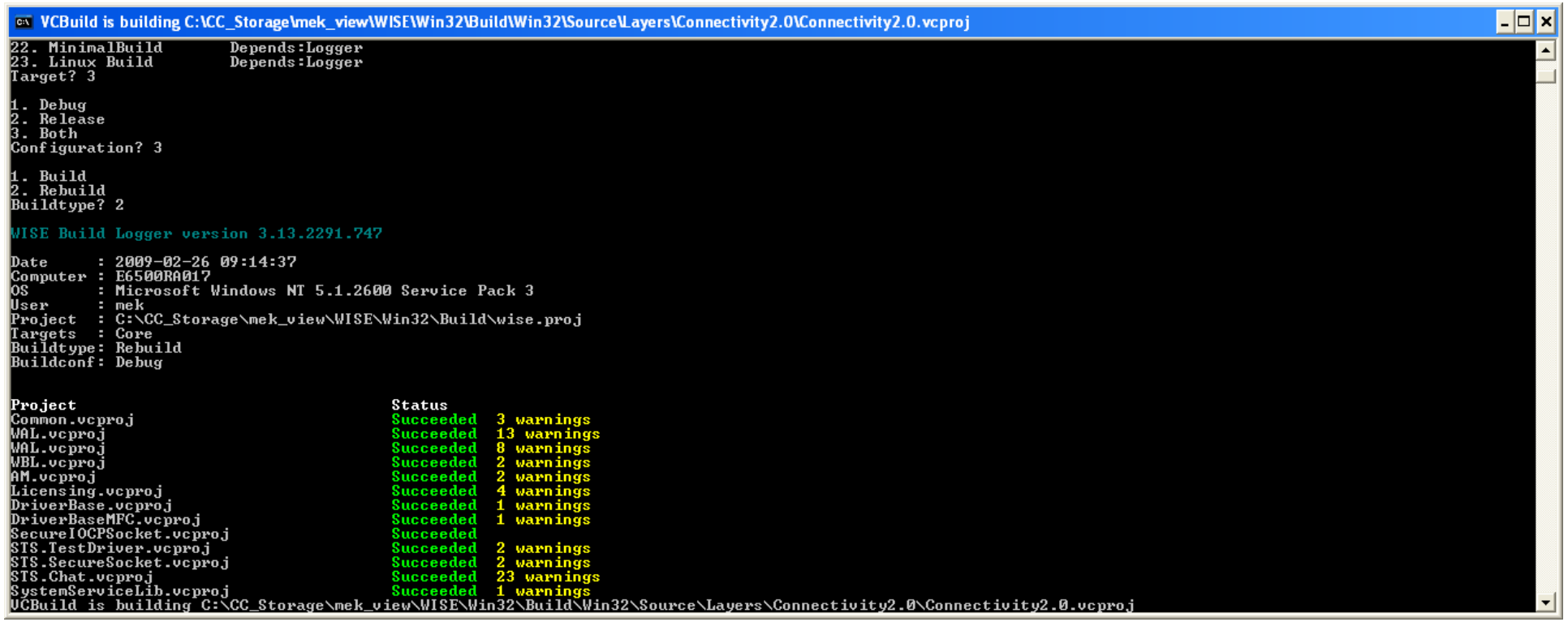
- Uses MsBuild as build engine
- A set of tasks that extends MsBuild
- All automated tests use MsTest as test engine
- Custom reports are created from the TFS repository

Our Environment



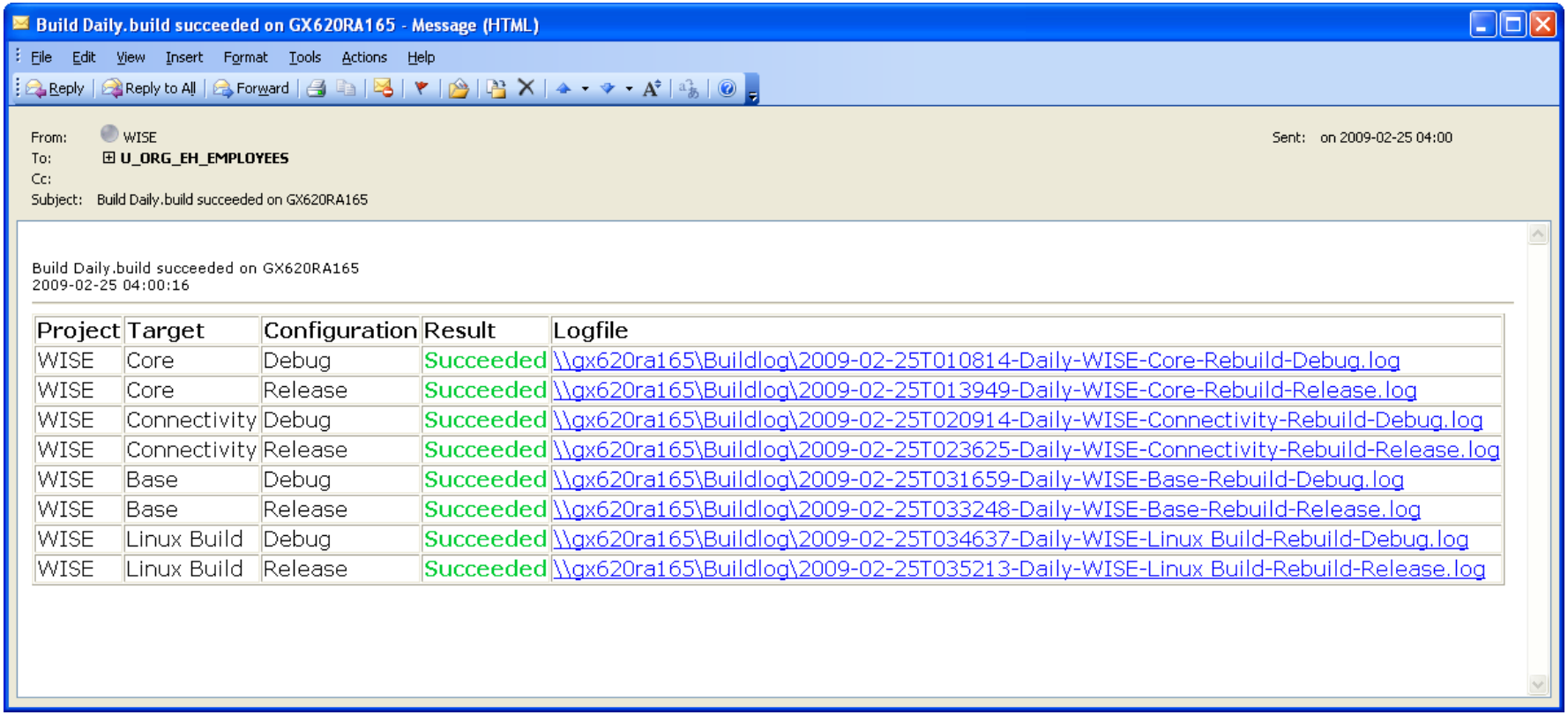
DevBuilder

- Easy to use
- Controlled by project files and build files



Mail Notifications

- Users are notified of the build status



Build Daily.build succeeded on GX620RA165 - Message (HTML)

File Edit View Insert Format Tools Actions Help

Reply | Reply to All | Forward

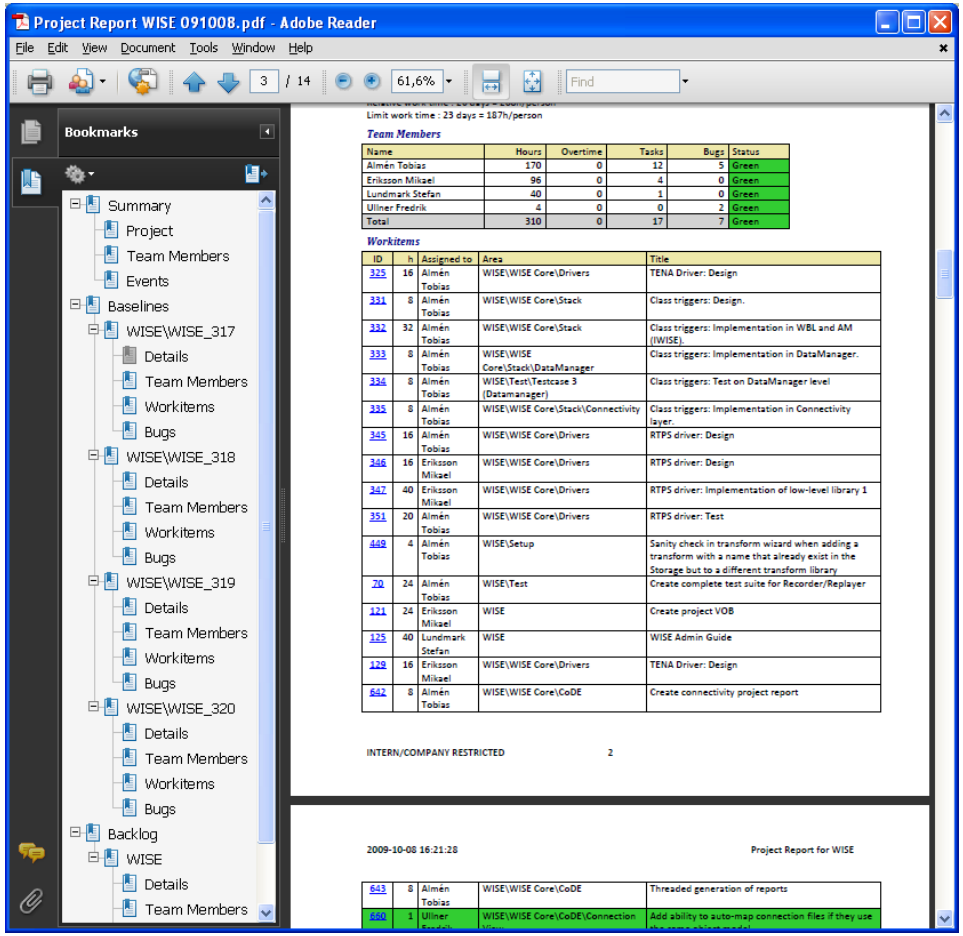
From: WISE
To: U_ORG_EH_EMPLOYEES
Cc:
Subject: Build Daily.build succeeded on GX620RA165
Sent: on 2009-02-25 04:00

Build Daily.build succeeded on GX620RA165
2009-02-25 04:00:16

Project	Target	Configuration	Result	Logfile
WISE	Core	Debug	Succeeded	\\gx620ra165\Buildlog\2009-02-25T010814-Daily-WISE-Core-Rebuild-Debug.log
WISE	Core	Release	Succeeded	\\gx620ra165\Buildlog\2009-02-25T013949-Daily-WISE-Core-Rebuild-Release.log
WISE	Connectivity	Debug	Succeeded	\\gx620ra165\Buildlog\2009-02-25T020914-Daily-WISE-Connectivity-Rebuild-Debug.log
WISE	Connectivity	Release	Succeeded	\\gx620ra165\Buildlog\2009-02-25T023625-Daily-WISE-Connectivity-Rebuild-Release.log
WISE	Base	Debug	Succeeded	\\gx620ra165\Buildlog\2009-02-25T031659-Daily-WISE-Base-Rebuild-Debug.log
WISE	Base	Release	Succeeded	\\gx620ra165\Buildlog\2009-02-25T033248-Daily-WISE-Base-Rebuild-Release.log
WISE	Linux Build	Debug	Succeeded	\\gx620ra165\Buildlog\2009-02-25T034637-Daily-WISE-Linux Build-Rebuild-Debug.log
WISE	Linux Build	Release	Succeeded	\\gx620ra165\Buildlog\2009-02-25T035213-Daily-WISE-Linux Build-Rebuild-Release.log

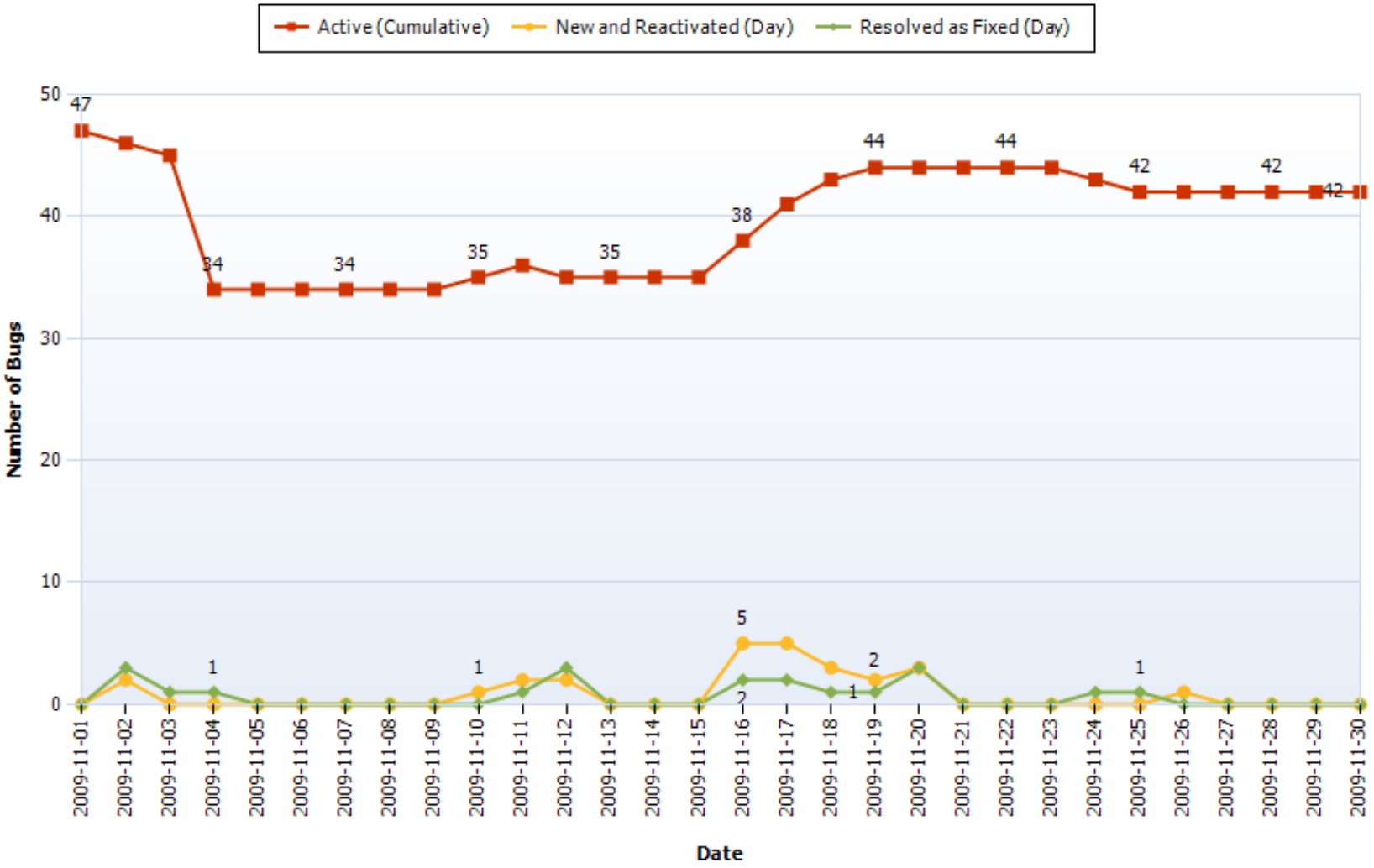
Reports

- Activity report
  - Baseline activities. Tasks & Bugs
- Project Area Report
  - Activities sorted by area
- Project Report
  - Project status
- Collective Project Report
  - Aggregated status for several projects
- Department Workload Report
  - Employee status for several projects
- Requirement Report
  - List of project requirements and their status.
- User Report
  - List of all items associated with a person.



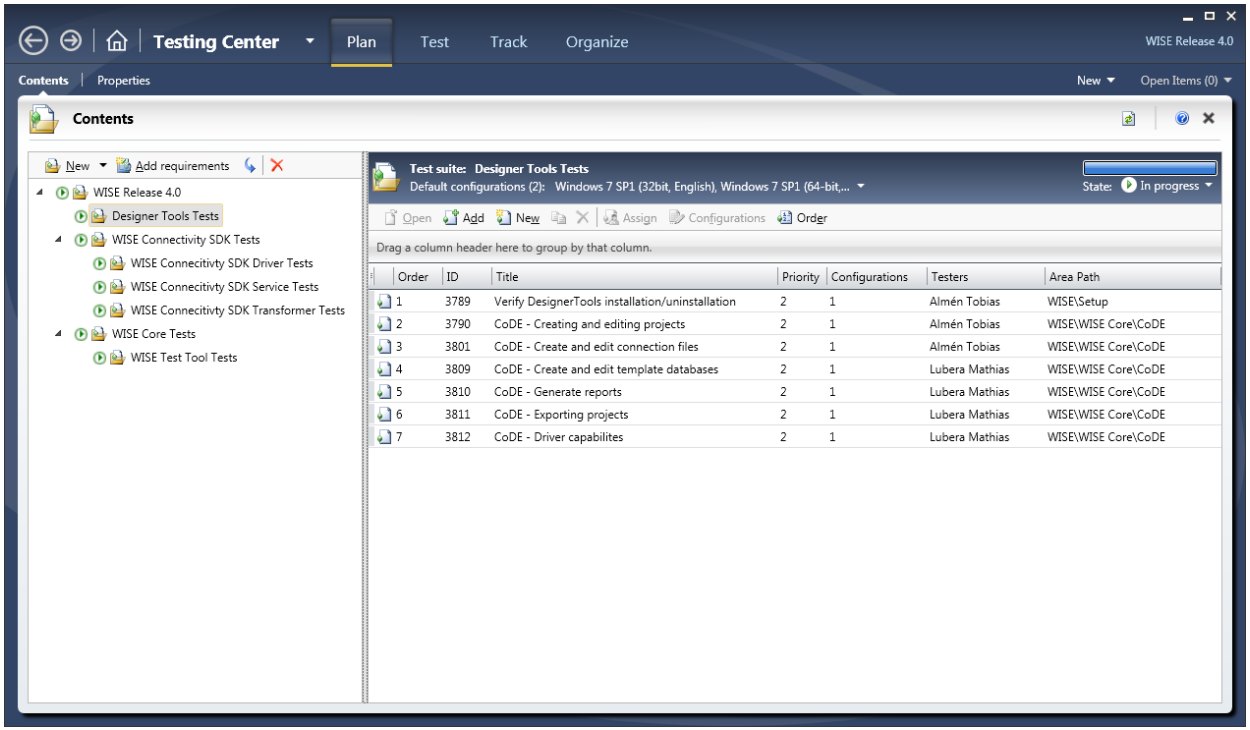
All reports can be automatically generated and distributed.

Trend - Open bugs



Test managment

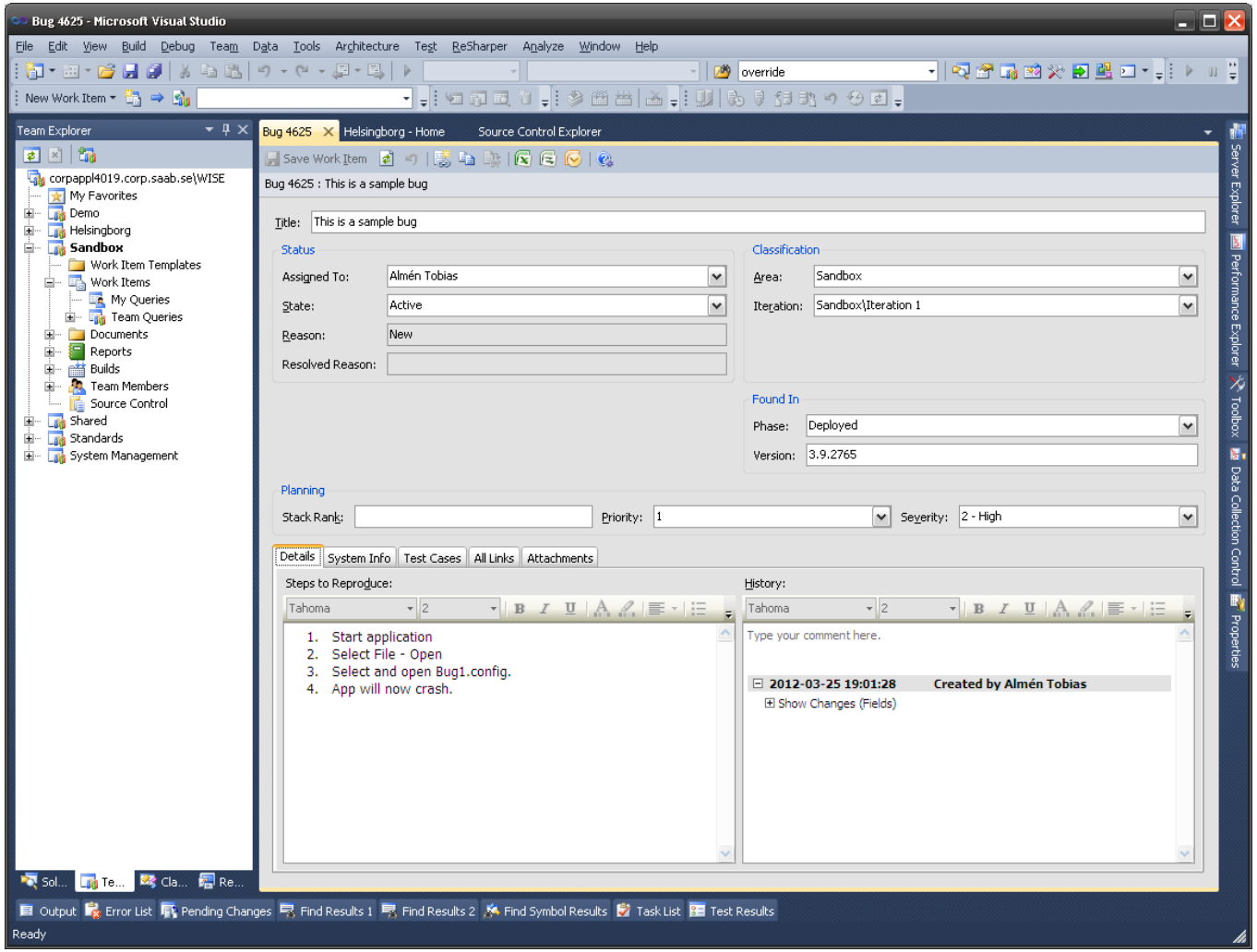
- TFS Test Manager
  - Test cases mapped to requirements.
  - Results logged in TFS.
  - Supports both manual and automated tests



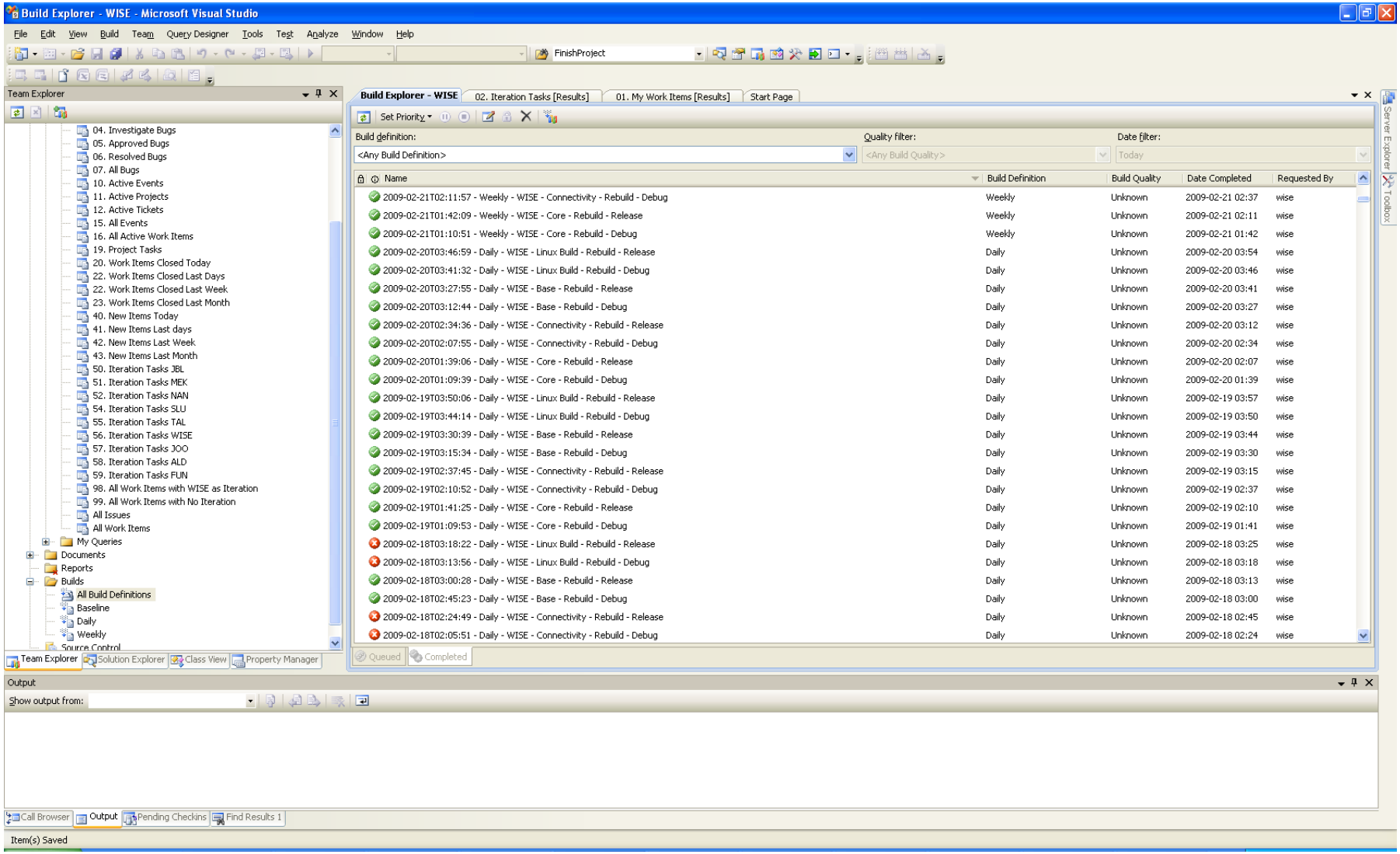
Visual Studio / Eclipse Integration

- TFS directly integrated in development environment.
- Quick access for developers.

Visual Studio Integration – Bug view

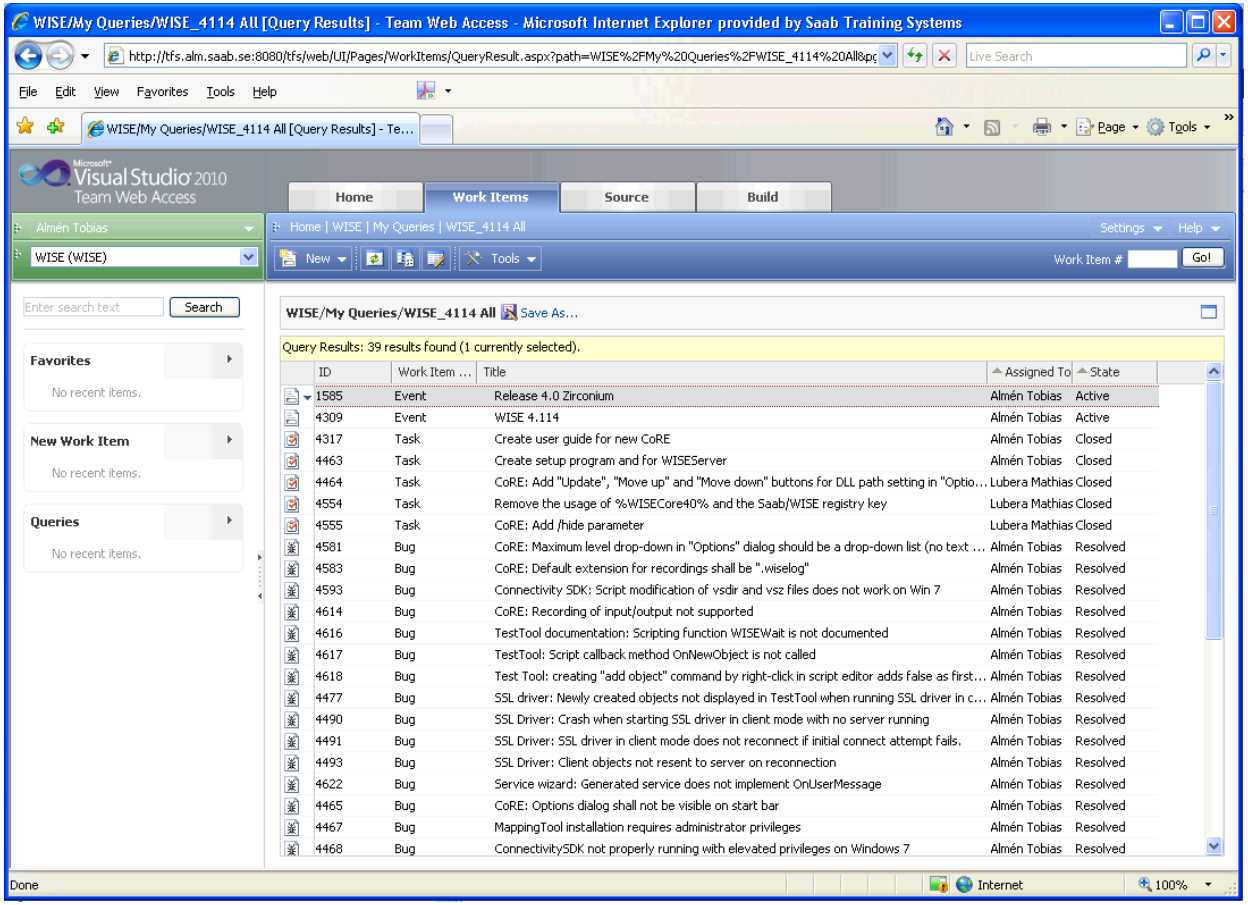


Visual Studio Integration – Builds view



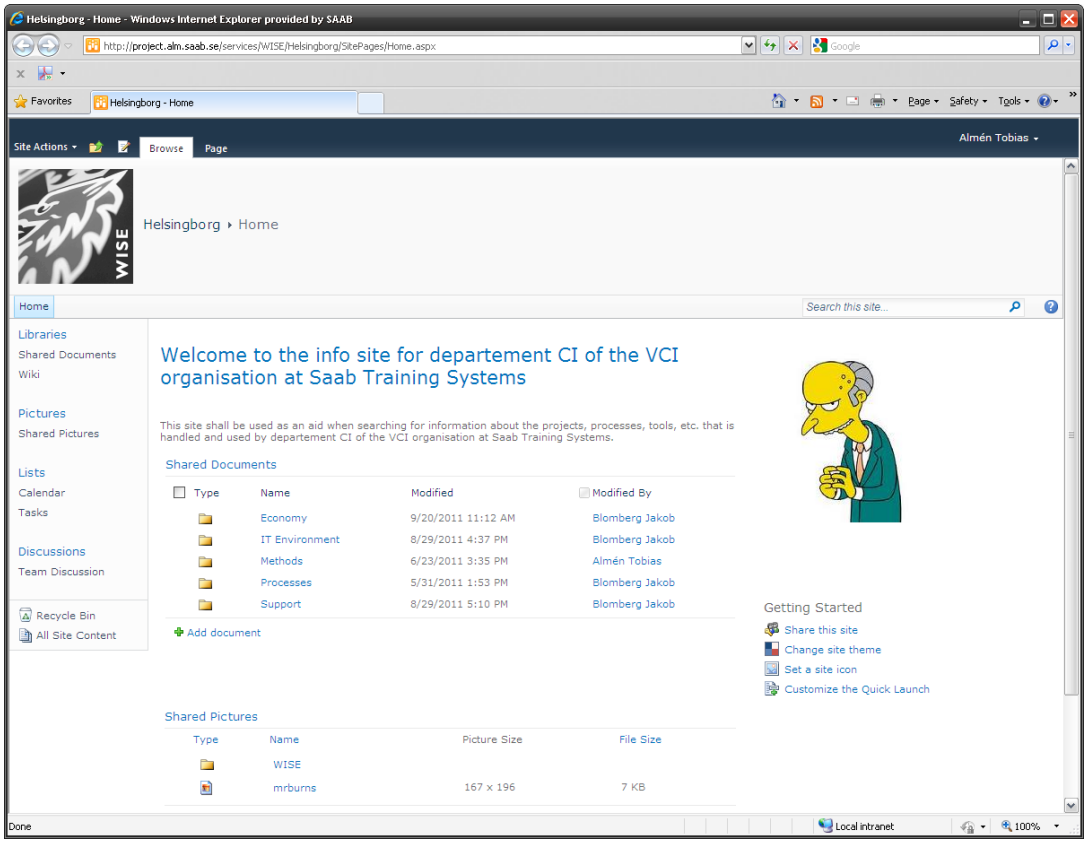
TFS Project Web Access

- Access to TFS data for non-developers.



Optional Project Sharepoint Site

- For extended project information such as documents, wikis, shared calendars etc.



Build Status Wallboard

SAAB

Project Status Wallboard

Questions?

SAAB
SAABGROUP.COM