

Enough of Process ?

Mark Carroll

Architect for Developer & Platform Strategy

Microsoft (NZ)

Key elements & types of process

- Process for efficient planning;
- Process to ensure that process guidance is both highlighted and completed;
- Process to monitor and track changes;
- Process to monitor and track progress;
- Audit process – sufficient to maintain the all important PIR stage which is often challenged by a lack of history ‘after the event’;

The problems with Process

- The Problem of Completeness
- The Problem of Out-of-Sync Processes
- The Problem of Acquiring Knowledge
- The Problem of Stupid Processes
- What Do These Problems Mean?

The Problem of Completeness

- “In terms of completeness, each process definition—large or small—wants to describe a complete process. Where the authors set out to describe a single area of the process, they inevitably end up trying to describe them all, typically in a tightly coupled and homogeneous fashion.”
- “When processes are published, their creators often find it hard to resist describing every aspect of software development. Even when they do, the methodologists who follow in their footsteps generally try to fill in the gaps. The end result is that, as time passes, every process strives to be complete. “

The problem of 'out of sync' Processes

- “What a team actually does never matches the adopted process. Teams improvise and invent more effective ways of doing things; they find they need to work out solutions to problems nobody thought of when the process was selected, and they never keep the process descriptions up to date.”
- “If you have ever been involved in a process assessment, you know that what the team does rarely matches the official process. This is true of all processes. People use a mixture of ideas and practices from various sources, and although they might claim to be advocates of specific processes, few of them are actually following them to the letter.”

The Problem of Acquiring Knowledge

- “To learn about a project, people don't read process manuals or language specifications. This is because they want to apply processes, not read about them. Moreover, they want different levels of detail at different times.”
- “A consequence of this is that it doesn't make sense to write a lot of long and lengthy process descriptions. Process documentation needs to be presented in a new way, one that provides the practitioners with the information they need when they need it; leaving the books and knowledgebases to publish the academic foundations that the processes draw upon. Let the processes facilitate the active development of the software and the application of the practices the team selects.”

The Problem of Stupid Processes

- “Most of the time, the process doesn't actively help people do their jobs because:
 - You have to know what you're looking for if you want to find anything useful.
 - When you do find something useful, you have to apply it yourself.
 - They remind you of all the boring things you have to do without making them any easier or less boring. “

Practices – the meat in the sandwich

In this talk it is not the ‘target platform’ for software developments which is our key focus - it is the underlying flexibility of process to permit ‘practices’ and governance over those ‘practices’ which are key focuses.

Process models

Simple models
Guidance mechanisms
Flexible and often automated

Practices

Some ‘codified’ – some not
Some rely on ‘skill and judgement’
Some leverage patterns / SDKS – eg : MS Enterprise framework 3.1, Acropolis

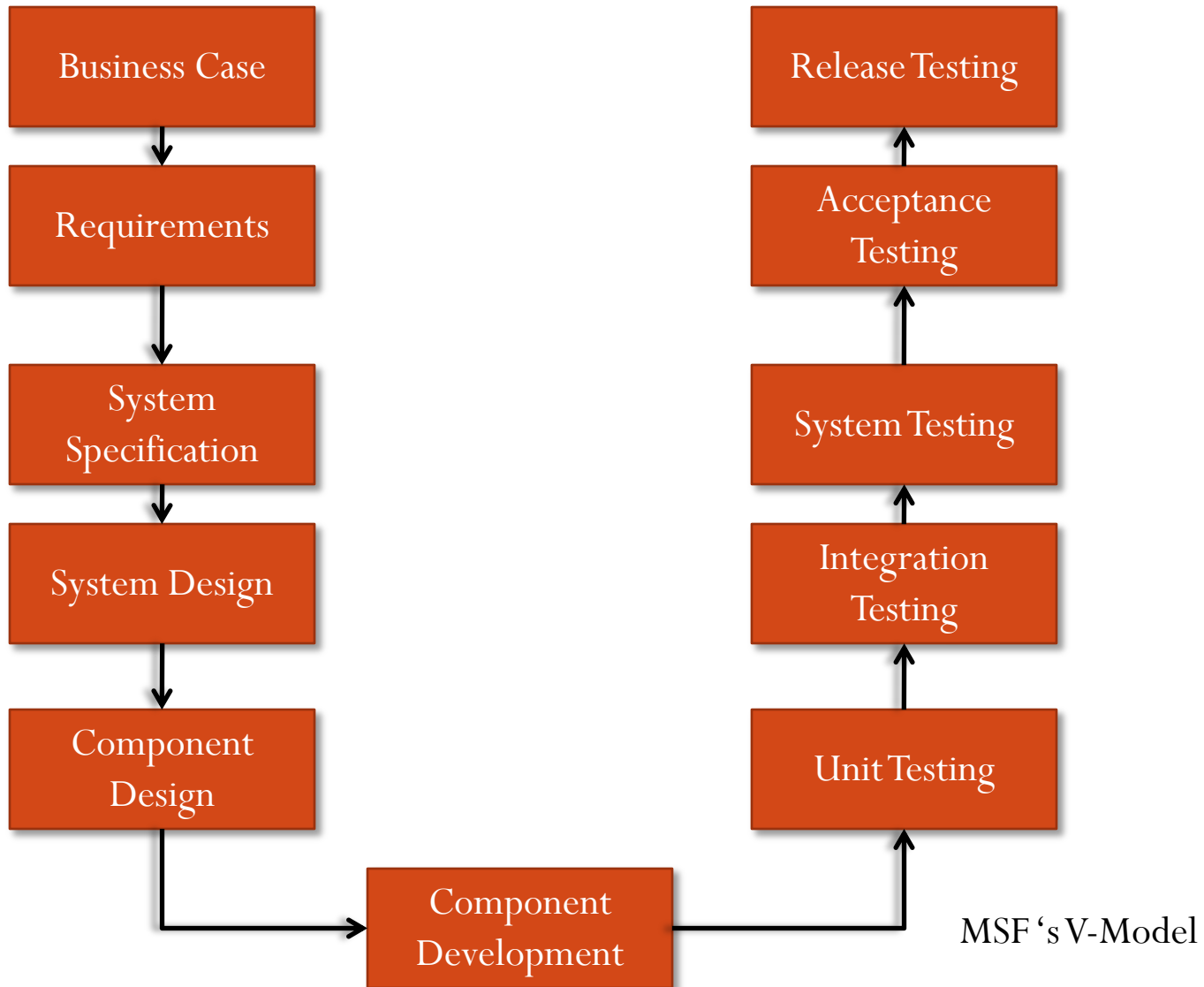
Governance facilitation - Check-points

Progress measurement
Problem identification
PIR with feedback up to Practices and Process models

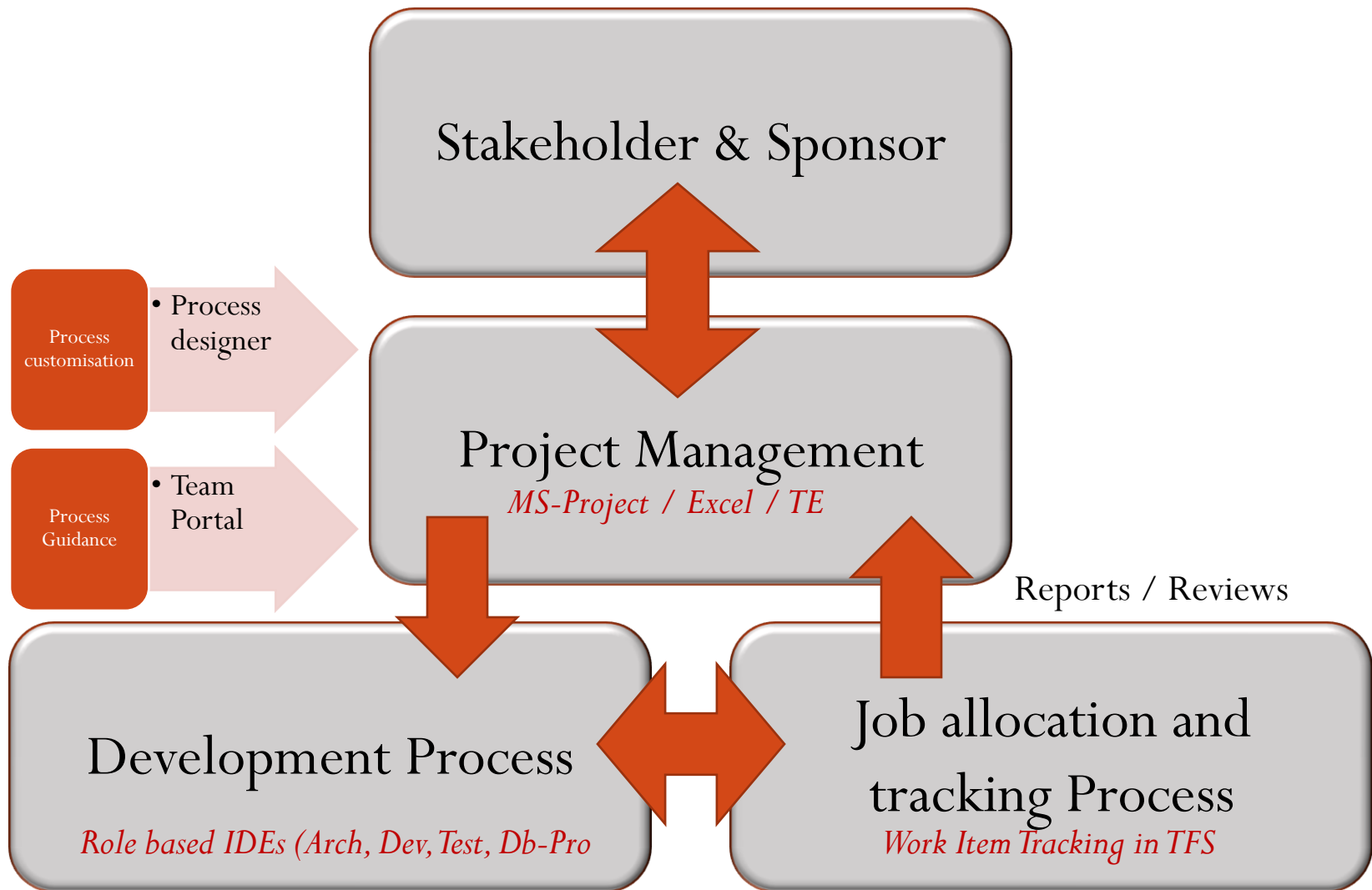
Platform example

- Modern ALM Tools along with the development of practices and patterns are helping to make this happen;
- Here we are using those Tools from the Microsoft platform – this is also a very grey world for example you can acquire Eclipse plug-ins to use with MS Visual Studio Team System for Java / .NET developments;

Simple process model for Development



Simple process model for Collaboration



Example: Linking MS-Project with Work-items

The screenshot shows a software interface for configuring project mappings. On the left is a tree view with the following structure:

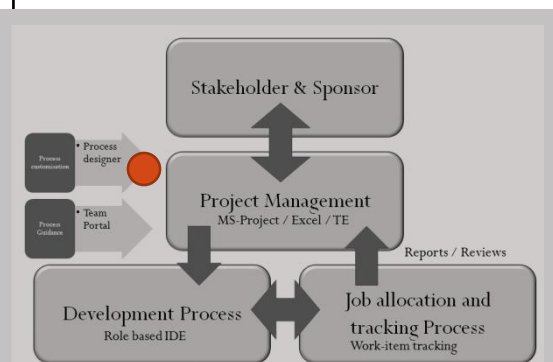
- Areas & Iterations
 - Areas
 - Iterations
 - MS Project Mapping**
- Groups & Permissions
- Source Control
 - Checkout Settings
 - Checkin Notes
 - Permissions
- Portal
- Reports

The main area displays a table with the following data:

System.AssignedTo	pjTaskResourceNames	
Microsoft.VSTS.Scheduling.Co...	pjTaskActualWork	
Microsoft.VSTS.Scheduling.Re...	pjTaskRemainingWork	
Microsoft.VSTS.Scheduling.Ba...	pjTaskBaselineWork	
Microsoft.VSTS.Scheduling.St...	pjTaskStart	
Microsoft.VSTS.Scheduling.Fin...	pjTaskFinish	

In the foreground, a 'Project Mapping' dialog box is open, showing the following configuration:

- Field Reference Name: Microsoft.VSTS.Scheduling.StartDate
- Project Field: pjTaskStart
- Project Name: (empty)
- Project Unit: (empty)
- Publish Only:



Creating Standard 'Tasks'

Task [X]

Title: Discipline:

Classification

Area:

Iteration:

Status

Assigned to: State:

Rank: Reason:

Description | History | Links | File Attachments | Details

Description:

Execute Baseline units and ensure these are in Build, if no baseline unit tests exist then they must be created.

OK Cancel

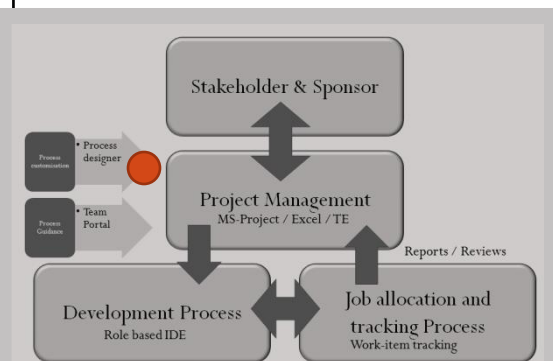
Schedule

Remaining work (hours):

Completed work (hours):

Start Date:

Finish Date:



Team Portal

Home - AdventureWorks - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address http://vstsb2/sites/AdventureWorks/default.aspx

Home Documents and Lists Create Site Settings Help

AdventureWorks Home Modify Shared Page

This is our team project for AdventureWorks

Documents

- Project Management
- Development
- Test
- Templates

Process Guidance

Reports

- Exit Criteria Status
- My Bugs
- Issues

Announcements

AW Reports are available 6/7/2005 1:13 PM
by Darren Parker
If you require custom reports, or further reports that are available, please contact Michael Leworthy

[Add new announcement](#)

Quality Indicators

Description:
Are the tests effective at determining completeness and correctness?

Date	Active Bugs	% Code Coverage	Tests Passed	Tests Inconclusive	Tests Failed
5/16/2005	24	10	3	3	6
5/17/2005	33	15	4	4	7
5/18/2005	24	10	2	5	1
5/15/2005	27	10	4	6	3

Bug Rates

Description:
Do the bug trends point toward closure?

Date	Active Backlog	Activated	Resolved
2005-05-16	22	25	5
2005-05-17	33	18	13
2005-05-18	24	5	10
2005-05-19	28	15	10
2005-05-31	0	0	28

Remaining Work

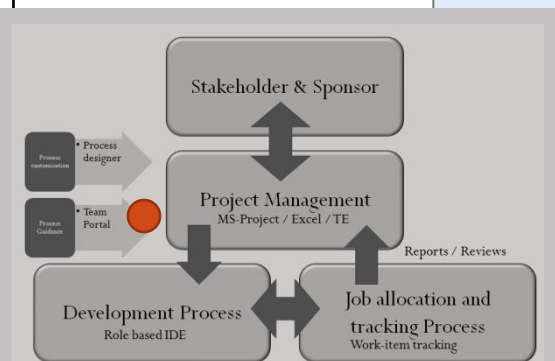
Description:
Is the team getting the work done?

Date	Active	Resolved	Closed
2005-05-02	28	0	0
2005-05-16	25	25	0
2005-05-17	25	25	0
2005-05-20	5	30	35

Builds

Description:
What builds are available?

Build	State	Lines Churned
AutomatedBuild_050503.00001	Successfully Completed	14374
AutomatedBuild_050518.00002	Successfully Completed	0



Process Guidance

Overview

Roles

Work Items

Views

Index

Glossary

Project Portal >

Concepts

Concepts

- Cycles and Iterations
- Governance
- Team Model
- Principles
- Mindsets
- Innovations
- Getting Started

See Also

How to: Get started with MSF

To obtain the latest process templates and whitepapers, visit the MSF workbench

To see a list of tasks to get started using MSF with VSTS

Visual Studio 2005 Team System

Process Guidance

Microsoft Solutions Framework (MSF) for Agile Software Development is the process guidance for the current team project. This page introduces key concepts.

Roles

Team members assume one or more roles.



Workstreams and Activities



1

Roles do activities, which are grouped in workstreams. Activities may produce certain work products and may require work products in certain states before they can be performed.

Visual Studio Team System

Visual Studio Team System provides tools that you can use to enact process guidance.

Users and Groups

Team members log in as users. Users belong to groups, which enforce security privileges, and can be used to implement roles.

Work Item Database and Metric Warehouse

Everything planned or tracked for the team project is managed in a database. Those records are called work items. They can track the state of activities or tasks within them.



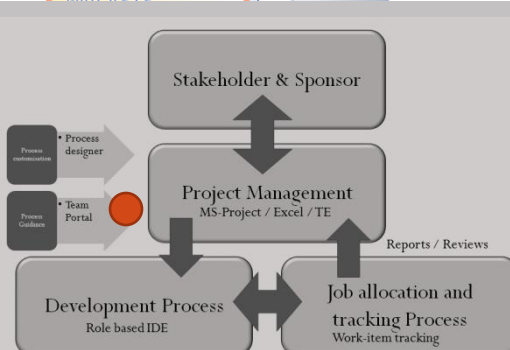
Queries and reports let you track your project status health in real time.

Source Control and Project Portal

Team Foundation source control stores some of the work products, such as source code and text. The rest, such as documents, spreadsheets, and project plans are stored on the project portal. Spreadsheets and project plans may be linked to the work item database.



ts
are the documents, spreadsheets, project
ode, and other tangible output from the



Embedding Practices & Processes

Home Documents and Lists Create Site Settings Help

AdventureWorks Home

Microsoft patterns & practices Home > Microsoft patterns & practices Home

Microsoft patterns & practices Developer Center

Microsoft patterns & practices are Microsoft's recommendations for how to design, develop, deploy, and operate architecturally sound applications for the Microsoft application platform.

Software Factory is a structured collection of related software assets. When a software factory is installed in a development environment, it helps architects and developers predictably and efficiently create high-quality instances of specific types of applications.

Application Blocks are reusable source-code components that provide proven solutions to common development challenges. They can be integrated as is into applications, or they can be extended or customized. Enterprise Library is a collection of general purpose application blocks addressing scenarios such as logging, data access and validation.

Guides consist of written guidance, either online or printed, which you can read to get a detailed understanding of technical problem domains and engineering practices. patterns & practices Guides cover topics that include patterns, application architecture, integration, performance, and security.

Announcements

Just Released: Team Development with Visual Studio Team Foundation Server - Beta 1
This guide shows you how to get the most out of Visual Studio Team Foundation Server to help improve the effectiveness of your team-based software development. Whether you are already using Team foundation Server or adopting from scratch, you'll find guidance and insights you can tailor for your specific scenarios.

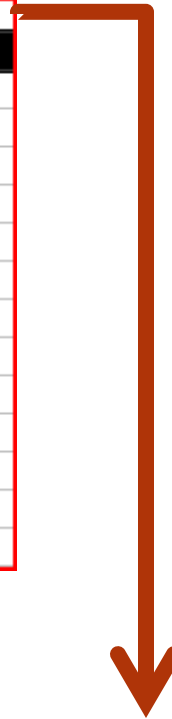
Just Released: Enterprise Library 3.1 - May 2007
The latest release of Enterprise Library features two new application blocks, the Validation Application Block and the Policy Injection Application Block. The Enterprise Library 3.1 release includes all improvements from the earlier Enterprise Library 3.0 - April 2007 release in addition to minor changes and fixes to the Data Access Application Block, the Policy Injection Application Block, and the Validation Application Block. Most notably, the default Remoting Policy Injector can now be replaced with alternative interception mechanisms modifying the application block code.

Just Released: Smart Client Software Factory - May 2007
The 2007 release of the Smart Client Software Factory includes support for occasionally connected clients, support for integrating Windows Presentation Foundation, source code composite UI Application Block, new QuickStarts, the latest version of the Enterprise Library assemblies, and new guidance package enhancements.



Project to Work-item and back

16	Set up: Set	1 day	Tue 5/1/07	Tue 5/1/07		Administrator
18	Set up: Migr	1 day	Wed 5/2/07	Wed 5/2/07	1	Administrator
19	Set up: Set	1 day	Thu 5/3/07	Thu 5/3/07	2	Administrator
20	Set up: Con	1 day	Fri 5/4/07	Fri 5/4/07	3	Administrator
21	Set up: Sen	1 day	Mon 5/7/07	Mon 5/7/07	4	Administrator
22	Create Visi	1 day	Tue 5/8/07	Tue 5/8/07	5	Administrator
23	Set up: Cre	1 day	Wed 5/9/07	Wed 5/9/07	6	Administrator
24	Create Pers	1 day	Thu 5/10/07	Thu 5/10/07	7	Administrator
25	Define Itera	1 day	Fri 5/11/07	Fri 5/11/07	8	Administrator
26	Create Test	1 day	Mon 5/14/07	Mon 5/14/07	9	Administrator
27	Brainstorm	1 day	Tue 5/15/07	Tue 5/15/07	10	Administrator
28	Brainstorm	1 day	Wed 5/16/07	Wed 5/16/07	11	Administrator
29	Set up: Cre	1 day	Thu 5/17/07	Thu 5/17/07	12	Administrator
30	Create Itera	1 day	Fri 5/18/07	Fri 5/18/07	13	Administrator
31	Execute Ba	1 day?	Mon 5/21/07	Mon 5/21/07	14	Administrator



Schedule

Remaining work (hours):

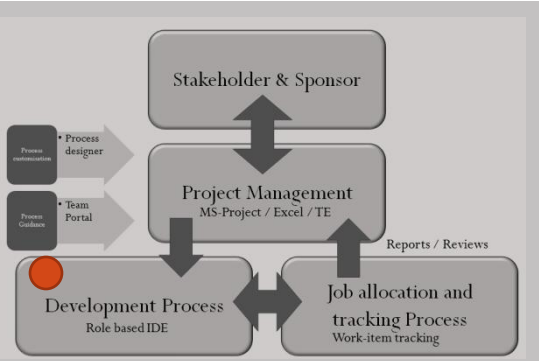
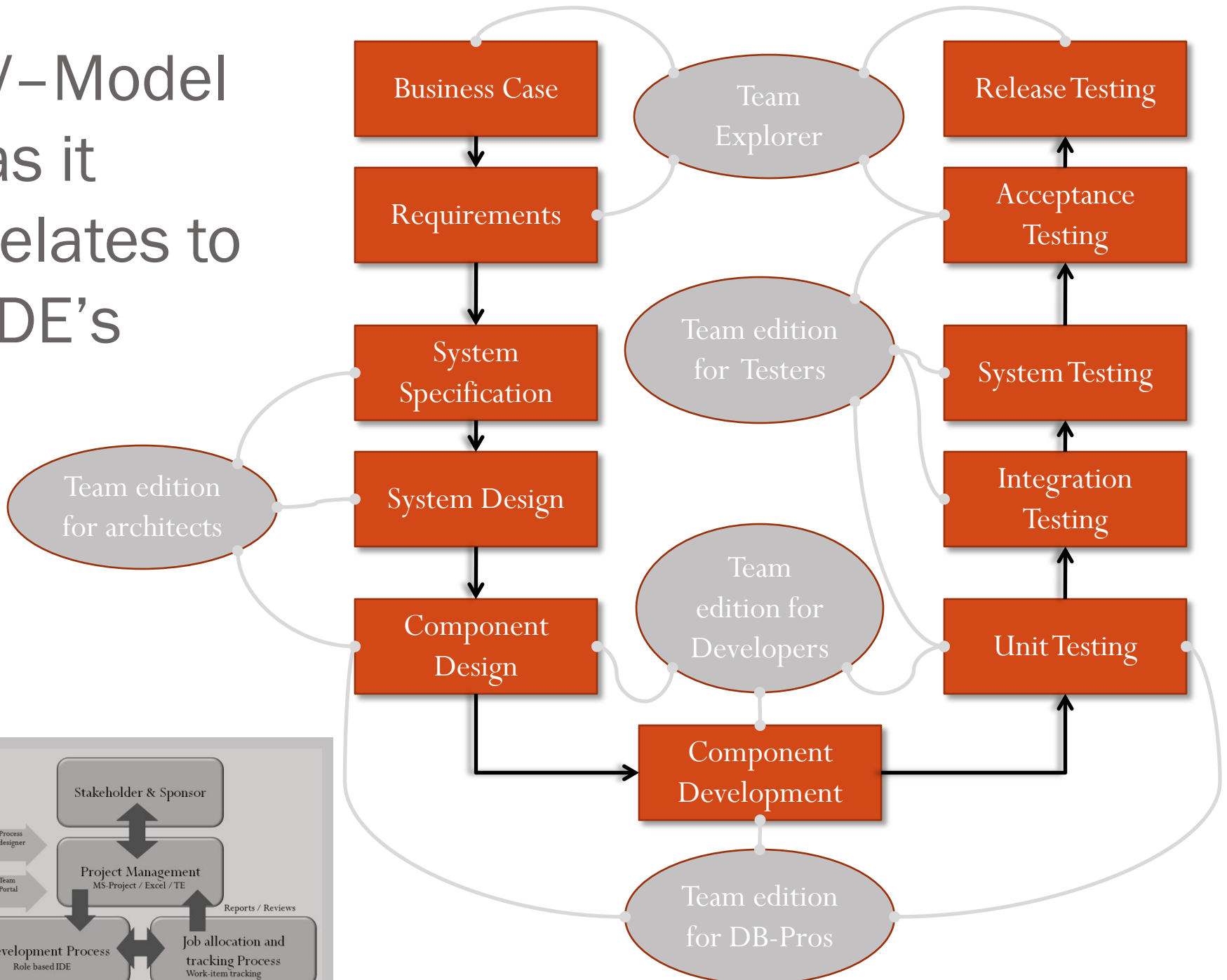
Completed work (hours):

Start Date:

Finish Date:

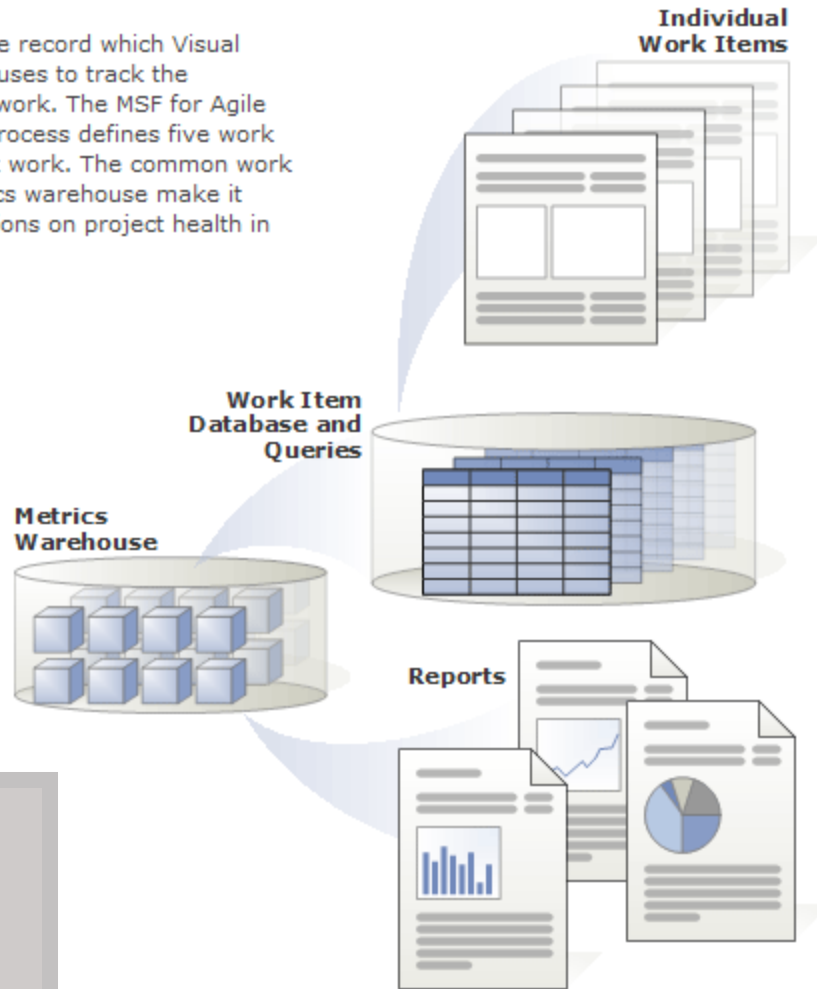


V-Model as it relates to IDE's



Work-Items

A work item is a database record which Visual Studio Team Foundation uses to track the assignment and state of work. The MSF for Agile Software Development process defines five work items to assign and track work. The common work item database and metrics warehouse make it possible to answer questions on project health in real time.



- Stateful with their own workflow by type;
- Can be defined;
- Accessible to roles determined by security policy;
- Typical types – Task, Risk, Bug etc

- SQL Server
- SQL Queries available for all roles

- SQL Reporting services
- Can be embedded in Portal using WSS

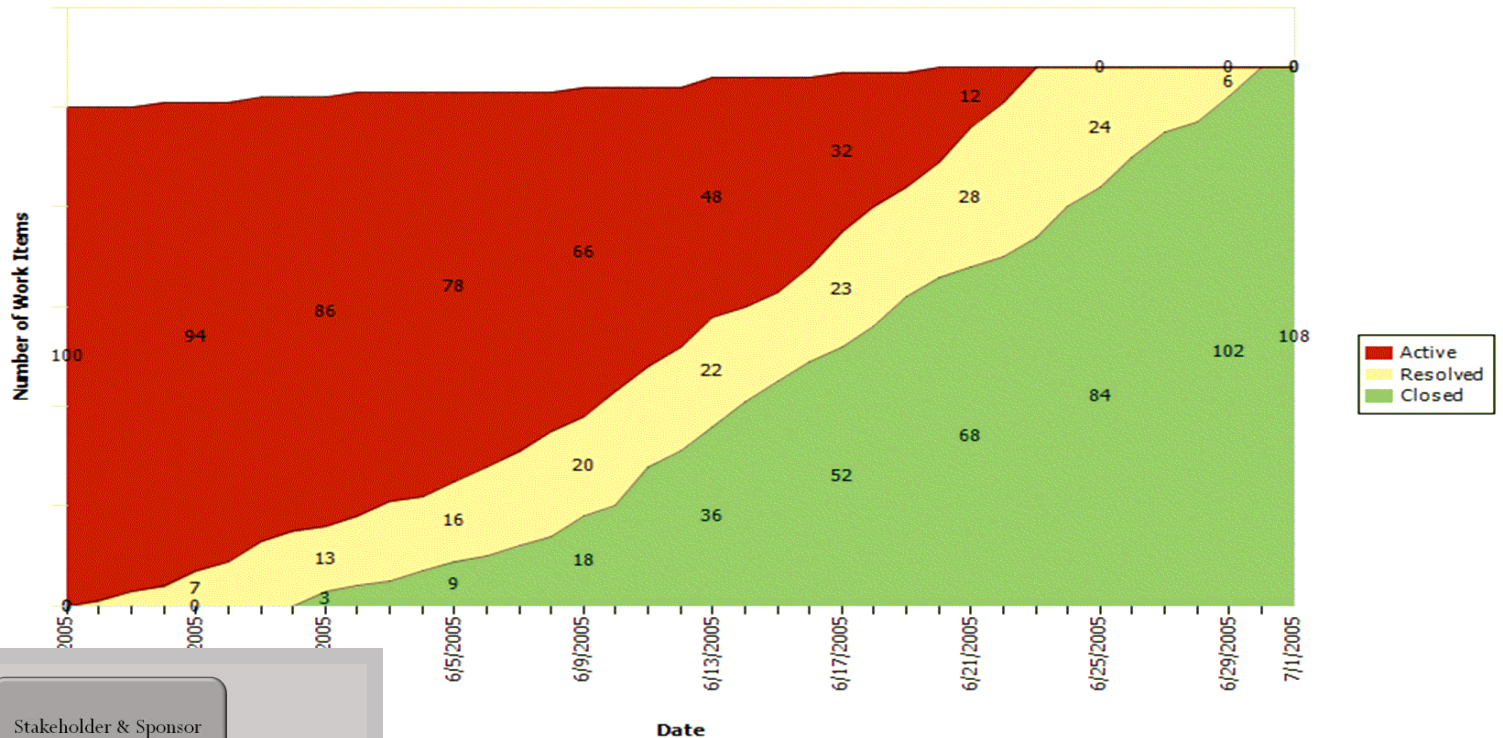


How Far Can We Get In The Available Time?

Remaining Work

Report Generated: 12/19/2005 10:45:37 AM by REDMOND\y-jperez; Last Warehouse Update: 12/19/2005 10:43:27 AM

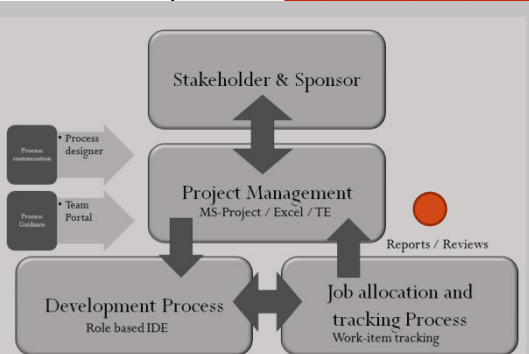
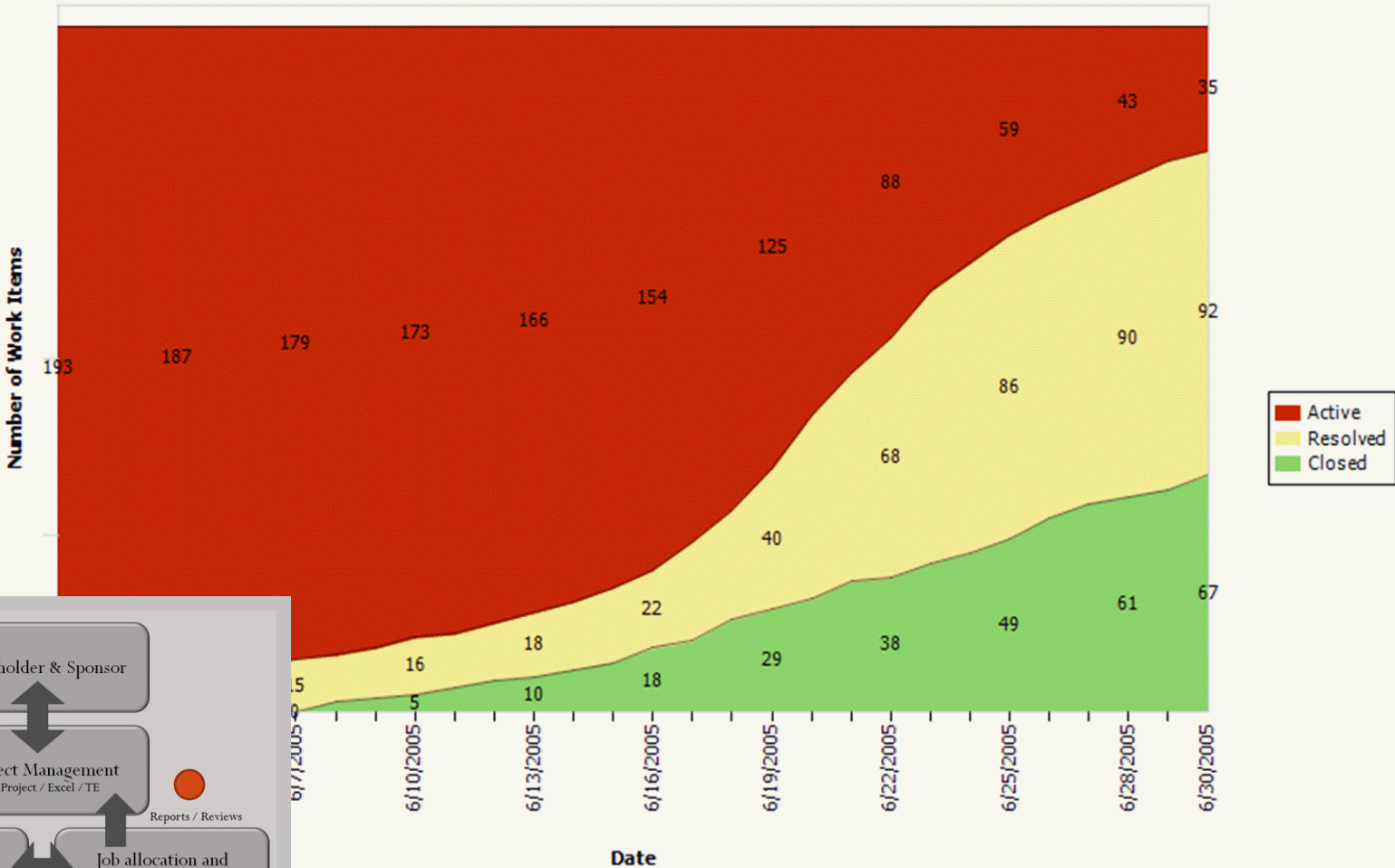
How much work is left and when will it be done?



Where do we need to shift resources ?

Remaining Work

How much work is left and when will it be done?



Build Failures

Report Viewer - Windows Internet Explorer

http://reportchow/ReportServer/Pages/ReportViewer.aspx?%2Fturnaround_3%2FReportPak%2FReport+Gallery&rs9

Report Viewer

1 of 1 100% Find | Next Select a format Export

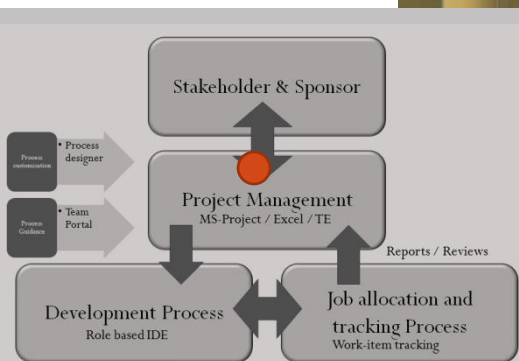
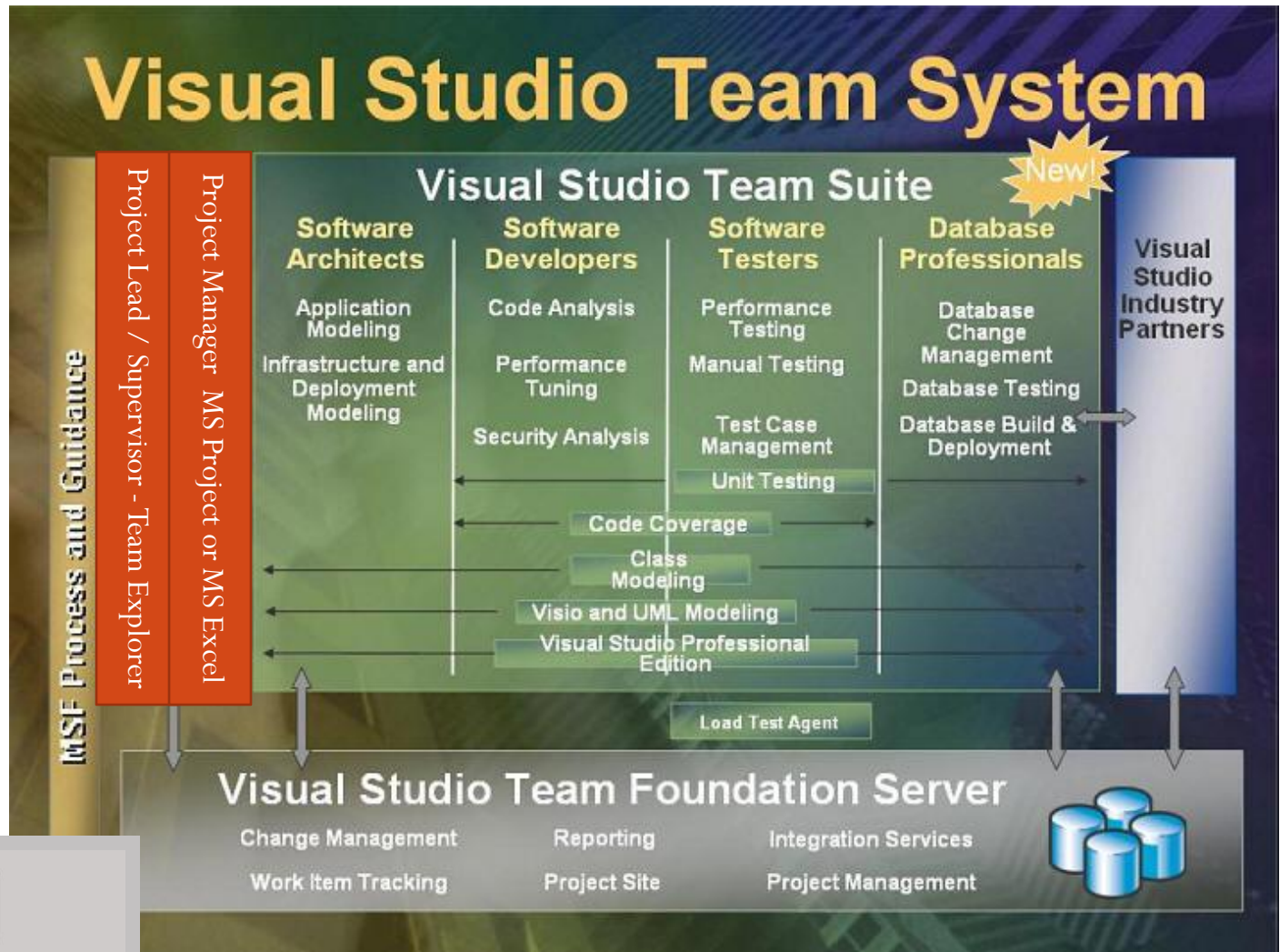
Builds
Report Generated: 5/25/2006 3:05:26 PM by REDMOND\samgu; Last Warehouse Update: 5/1/2006 9:34:32 PM

What are the details for the builds?

Build Name	Flavor	Platform	Build Quality	% Tests Passed	% Code Coverage	Code Churn
Debuqx86_20060222.5	Debug	x86	Unexamined	88.889 %	0.699 %	573
Debuqx86_20060222.4	Debug	x86	Unexamined	94.444 %	0.5 %	No Data
Debuqx86_20060222.3	Debug	x86	Unexamined	94.737 %	0.599 %	148
Debuqx86_20060222.2	Debug	x86	Unexamined	100 %	0.3 %	260
Debuqx86_20060222.1	Debug	x86	Unexamined	100 %	0.1 %	398
Debuqx86_20060221.21	Debug	x86	Unexamined	100 %	0.2 %	123
Debuqx86_20060221.20	Debug	x86	Unexamined	100 %	No Data	120
Debuqx86_20060221.19	Debug	x86	Unexamined	98.667 %	6.893 %	318
Debuqx86_20060221.18	Debug	x86	Unexamined	100 %	4.196 %	401
21.17	Debug	x86	Unexamined	98.182 %	4.396 %	1245
21.16	Debug	x86	Unexamined	96.364 %	3.896 %	3484
21.15	Debug	x86	Unexamined	100 %	2.597 %	1264
21.14	Debug	x86	Unexamined	97.619 %	2.198 %	2194



Visual Studio Team System



Summary

- IMHO : This is not the death of process as Ivar Jacobsen implied – process is still required but it is now aided which means the detailed prescriptive approach is no longer as appropriate as it once was;
- If you have tools like Team system ask yourself how effectively it is being used – assessments and training are available;
- If you are not using tools like Team system ask yourself if your current approach is attaining the levels of quality and efficiency you require from your Team based Software development;