

Implementing MSF to drive Competitive Advantage

David J. Anderson
Process Architect, MSF
Microsoft Corporation
<http://www.microsoft.com/msf>
david.anderson@microsoft.com

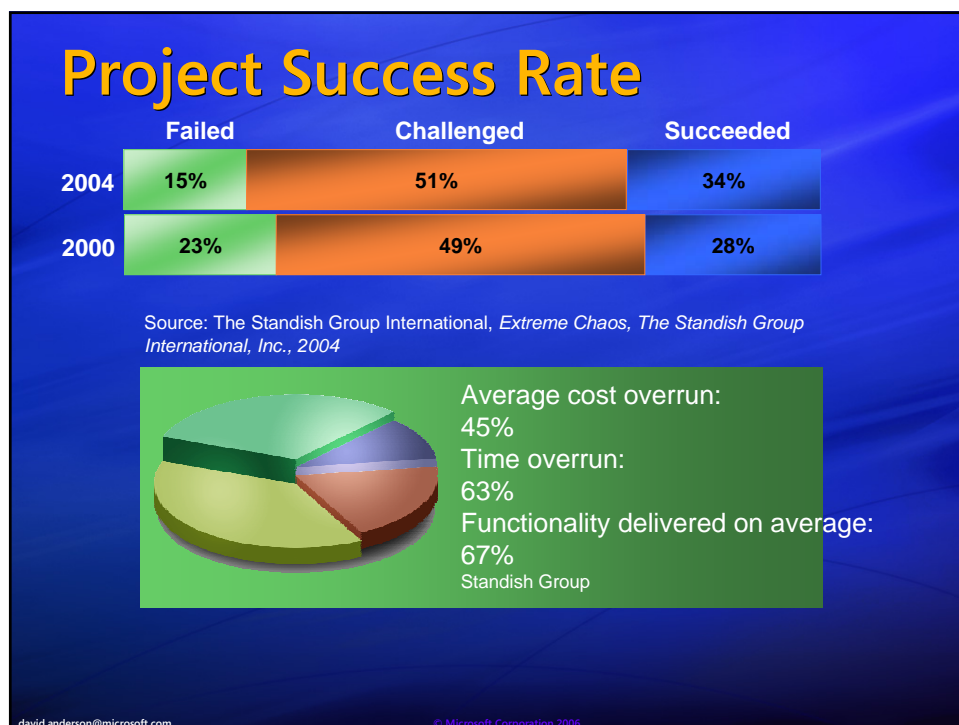
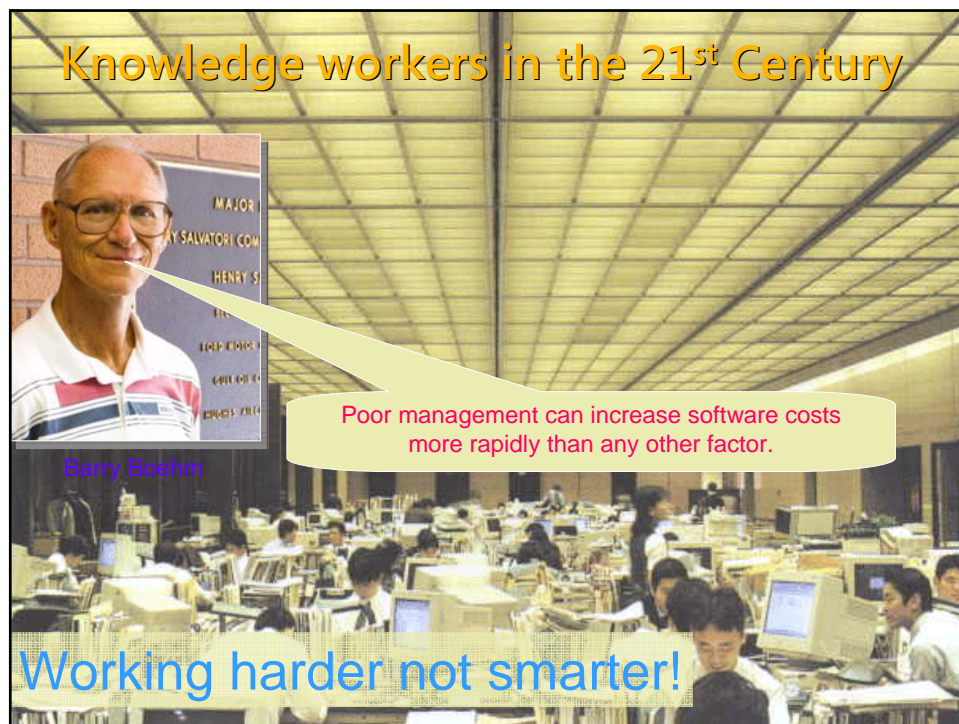
© Microsoft Corporation 2006

Agenda

- Why Process?
- MSF for Agile Software Development
- MSF for CMMI Process Improvement
- Trustworthy Transparency
- Continuous Improvement
- Implementing MSF

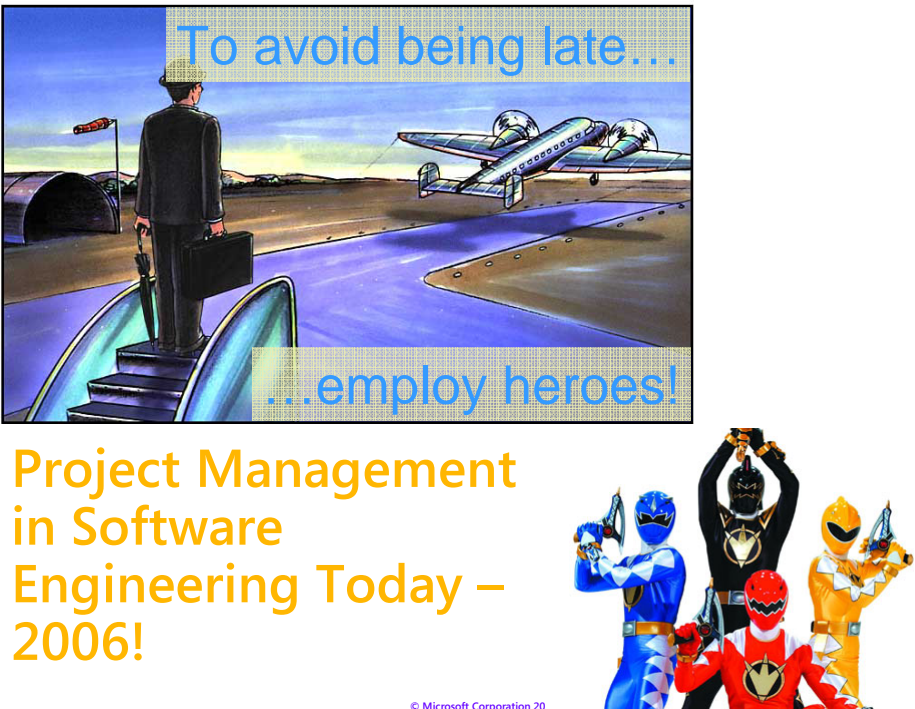
david.anderson@microsoft.com

© Microsoft Corporation 2006



To avoid being late...
...employ heroes!

**Project Management
in Software
Engineering Today –
2006!**



© Microsoft Corporation 2006

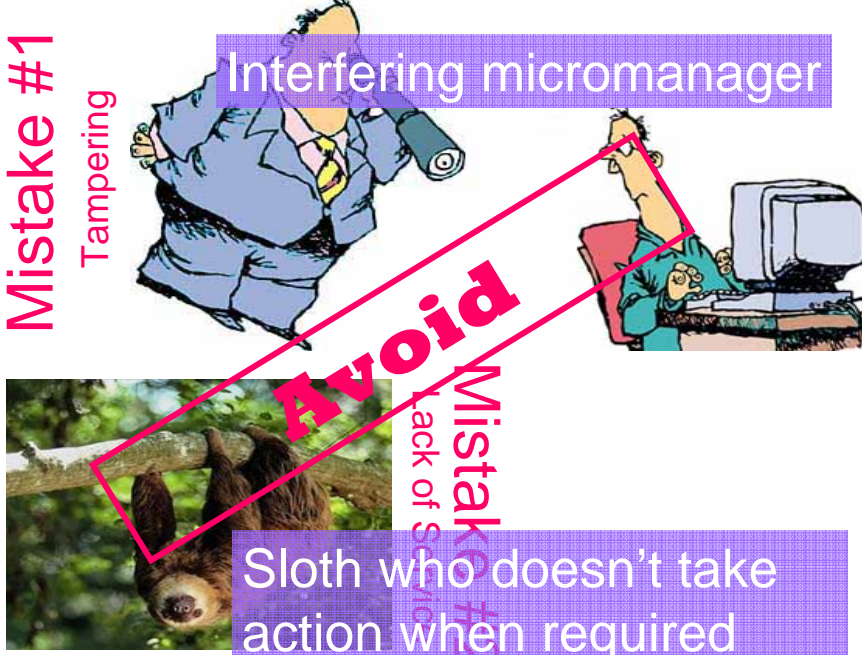
Mistake #1
Tampering

Interfering micromanager

Avoid

Mistake #2
Lack of action

Sloth who doesn't take action when required



© Microsoft Corporation 2006

Our Approach

Focus on People, Process and Tools

People

- Visibility at every level
- Real-time flow of information
- Quality mindset
- Team of peers

Process

- Customizable
- Lightweight
- Patterns and practices
- Agile and CMMI built-in

Tools

- Highly integrated
- Productive
- Quality tools for everyone
- Extensible
- Low admin

Successful IT organizations build application development capabilities through continuous improvement (Kaizen).

david.anderson@microsoft.com

© Microsoft Corporation 2006

Agenda

- Why Process?
- MSF for Agile Software Development
- MSF for CMMI Process Improvement
- Trustworthy Transparency
- Continuous Improvement
- Implementing MSF

david.anderson@microsoft.com

© Microsoft Corporation 2006

The Problem with Process

Productive

Predictable
Repeatable



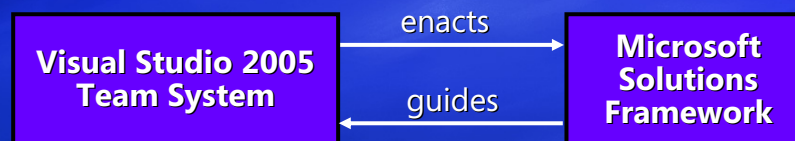
Challenges

- Getting people to know about it
- Getting people to use it

david.anderson@microsoft.com

© Microsoft Corporation 2006

Process & Tools Integration



- How VSTS enables MSF?
 - Productivity tools & processes
 - Integrated tools & processes
 - Extensible tools & processes

david.anderson@microsoft.com

© Microsoft Corporation 2006

The Origin of MSF

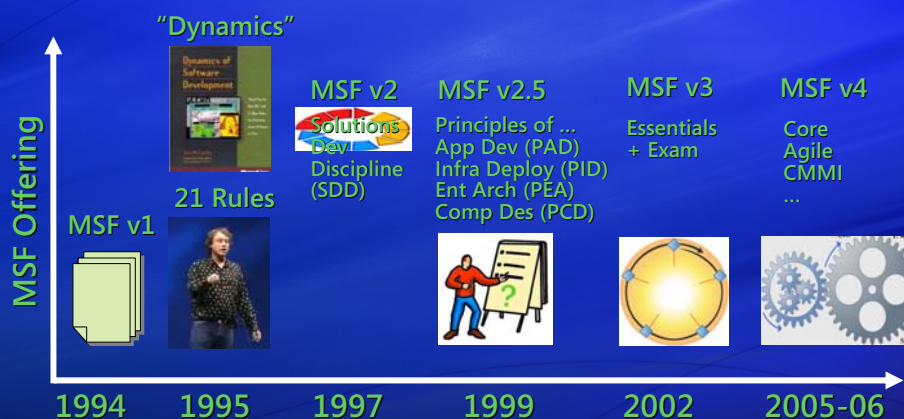


- Results from project teams and product groups are analyzed
- Analyzed results are contrasted with industry practices and methods
- Combined results are then organized and consolidated into "people and process"

david.anderson@microsoft.com

© Microsoft Corporation 2006

A Brief History



david.anderson@microsoft.com

© Microsoft Corporation 2006

MSF for Agile Software Development

- MSF for Agile Software Development is an iterative, scenario-driven, context-based software development process for building .NET, Web, Web Service, and other object-oriented applications.

david.anderson@microsoft.com

© Microsoft Corporation 2006

Essence of Agile

- Embraces Agile Manifesto, and...
- Enacts process in VSTS
- Full-lifecycle Agile Process
 - Business Analyst and Tester
 - Embraces Architecture
 - Designed to fit typical mid-size IT department
 - Builds on MSF v3
 - Team Model

david.anderson@microsoft.com

© Microsoft Corporation 2006

MSF for Agile Software Development - Microsoft Internet Explorer

Address: C:\Documents and Settings\jander\My Documents\MSF\MSF Agile\January_100_2\Build 100.2\Process Guidance\Supporting Files\Scenario_StatesandTransitions.htm

MSF for Agile Software Development

Overview | Rules | Work Items | Views | Index

Project Portal > Scenario

About Work Items

Scenario

Work Item Database

Overview

States and Transitions

Fields

Process Guidance

Activities

Workstreams

Scenario States and Transitions

A scenario is a type of work item, recording a single path of user interaction through the system. As the persona attempts to reach a goal, the scenario records the specific steps that they will take in attempting to reach that goal. Some scenarios will record a successful path; others will record an unsuccessful one. When writing scenarios, be specific as there are many possible paths.

New

Scenarios can be created in the quality of scenarios list found in the requirements folder in the document library or by using the Team Explorer.

New to Active

New A scenario is activated as a new scenario when it is first created.

Active

Scenarios begin in the Active state. The business analyst creates the scenario, provides a descriptive title, and fills in the Description field with as much detail as possible about the scenario. When the scenario is fully written, the business analyst assigns it to a lead developer. The Specified field is set to Yes, and the scenario remains in the active state while it is being implemented. The lead developer coordinates efforts with other developers to implement the scenario.

Active to Resolved

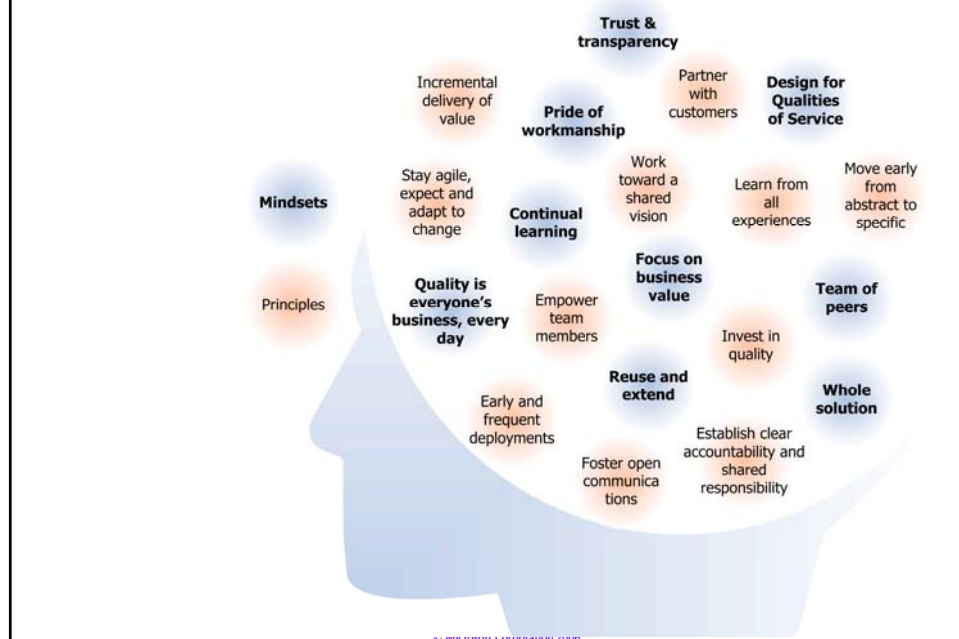
Completed	A scenario is resolved as Completed when the development team completes writing code for the scenario. The lead developer assigns the scenario to a tester.
Split	A scenario is resolved as Split when further review indicates that the scenario is too large, or that it needs more granular definition. When splitting a scenario, create the new scenarios and link them from the original scenario.
Deferred	A scenario is resolved as Deferred if it cannot be implemented in the current iteration. A scenario could be deferred because the team does not have enough time, or because blocking issues were discovered. Update the Iteration field to the correct iteration in which the scenario will be implemented. If the scenario is deferred to the next software product release version, leave the Iteration field blank. Be sure to include a detailed description of why the scenario was deferred, and when it is planned to be implemented.
Removed	A scenario is resolved as Removed if it is no longer deemed necessary to implement. When removing a scenario, check the Issue, and Exit Criteria fields. Typically these fields should be set to No for a removed quality of service scenario.

The states of a scenario work item.

Error on page.
david.santel@msn.com

© Microsoft Corporation 2006

Uses Best Innovations of Last Decade



Principles and Mindsets



Mindsets (individual approach)	Principles (govern team decisions)
1. Focus on Business Value	1. Partner with Customers
2. Advocate for Appropriate Constituencies of Risk	2. Work Toward a Shared Vision
3. Take Pride in Workmanship	3. Deliver Incremental Value
4. Deliver on Commitments	4. Invest in Quality
5. Look at the Big Picture	5. Empower Team Members
6. Foster a Team of Peers	6. Establish Clear Accountability
7. Practice Good Project Citizenship	7. Learn from All Experiences
8. Learn Continually	8. Foster Open Communications
9. Internalize Qualities of Service	9. Stay Agile, Adapt to Change

david.anderson@microsoft.com

© Microsoft Corporation 2006

Agenda

- Why Process?
- MSF for Agile Software Development
- MSF for CMMI Process Improvement
- Trustworthy Transparency
- Continuous Improvement
- Implementing MSF

david.anderson@microsoft.com

© Microsoft Corporation 2006

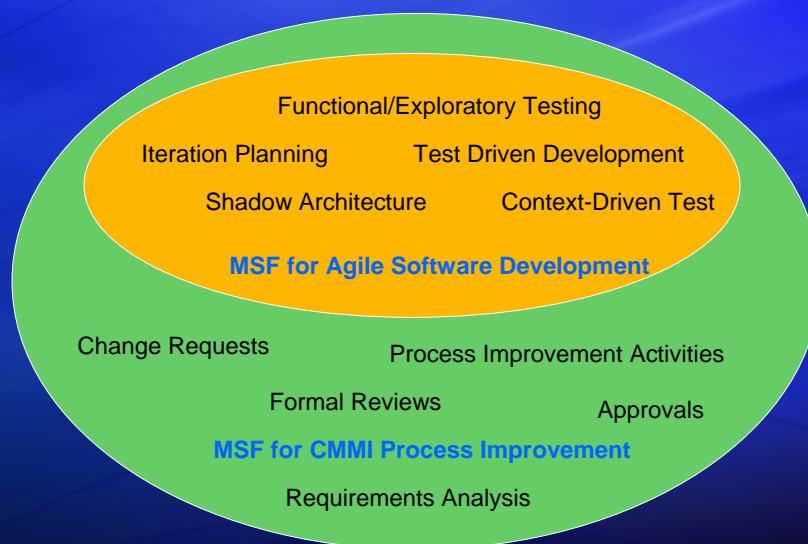
MSF for CMMI Process Improvement

- The first truly agile CMMI process implementation. Based on MSF for Agile Software Development
- Delivers on the original vision for CMM (1987) to implement the teachings of W. Edwards Deming on Quality Assurance for software engineering
- Process activities and artifacts mapped directly to CMMI Appraisal evidence
- Accelerates achievement of CMMI Level 3

david.anderson@microsoft.com

© Microsoft Corporation 2006

Agile or CMMI?



david.anderson@microsoft.com

© Microsoft Corporation 2006



OTOH, CMMI without MSF...

- Often overly bureaucratic and heavyweight
- Lots of manual data collection
- Appraisals are time consuming and expensive
- Requires forming SEPG or PMO process function
 - Lots of training from SEI
 - Often poor interpretation of CMMI specification
 - SCAMPI appraiser in consulting role

Why do this when MS has done it for you?

MSF for CMMI Process Improvement

Overview Roles Work Items Views CMMI Index Glossary

Project Portal >

About Roles

Program Management

Architecture

Development

Test

Release/Ops

User Experience

Product Management

Next Steps
Plan Project
Resources

Estimate Project (2 : PP : 1 : 1.1) (2 : PP : 1 : 1.2)

Project Manager

Activity:
Estimate Project

User Experience Architect
Business Analyst

Overview

To estimate the project, it is necessary to develop a rough guide to the scope of the project at this early stage. Detailed requirements have not yet been solicited. Detailed planning will be deferred to the start of each iteration. At this stage, conduct a focus group and brainstorming session to identify a minimum set of critical to quality (CTQ) end-to-end scenarios. This minimum set should include all the personas and cover the intent of the product vision. Develop a storyboard for each end-to-end scenario. Gain consensus that this set of scenarios represents the critical to quality success factor for the project. Publish the end-to-end scenarios as the high level scope of the project.

Entry Criteria

- Vision Statement**
The vision for the product.
- Business Case**
The case justifying the project.
- Quality of Service Requirements**
A list of non-functional requirements or constraints on the functionality of the system is available.

Sub-Activities

1 Identify End-to-End Scenarios

Hold a focus group and identify a list of end-to-end product scenarios which deliver the essence of the product vision and enable the business case when brought to market.

Responsible:
User Experience Architect
Business Analyst

Accountable:
Project Manager

Consult:
Any

Informed:
All

CMMI cross reference
Level 2 Project Planning
Goal 1
Specific Practice 1 & 2

Process Improvement

Overview Roles Work Items Views CMMI Index Glossary

CMMI Process Area:
Project Planning

collapse all

Workstreams referenced by CMMI Goal 'Establish Estimates'

Key

- Activities used
- Activities not used

Supporting Practices:

Estimate the Scope of the Project

Establish Estimates of Work Product and Task Attributes

Define Project Life Cycle

Determine Estimates of Effort and Cost

Develop a Plan for Data Management

Analysis

Architect

Design and Development Analysis

Test Planning

User Experience Work Breakdown Analysis

User Education Work Breakdown Analysis

Establish Project Process

Release Manager

Select Project Process Template

Tailor Project Process

Review Project Process

Establish Project Measures

Implement a Development Task

Developer

Cost a Development Task

Design

Create or Update System or Verification Test

Prepare for Design Review

Design Review

Write or Update a Unit Test

Write Code

Refactor Code

Refactor Code

Refactor Unit Test

Refactor Code

Prepare for Code Review

Plan an Iteration

Project Manager

Select Iteration Backlog

Iteration Analysis

Plan Knowledge and Skills

Plan Iteration Resources

Form Iteration Team(s)

Define Iteration Roles and Responsibilities

Identify Iteration Stakeholders

Plan Iteration Stakeholder Involvement

Estimate Iteration

Define Iteration Budget and Schedule

Plan Project

Project Manager

Determine Risk Sources and Categories

Define Risk Parameters

Determine Risk Management Strategy

Plan Project Resources

Plan Project Knowledge and Skills

Form Project Team

Establish Project Team Charter

Define Project Roles and Responsibilities

Define Project Life Cycle

Highlighted activities provide SCAMPI evidence for CMMI appraisal Process Area: Project Planning; Goal: Establish Estimates

MSF CMMI Reference (2).xls [Read-Only]								
	A	B	C	D	E	F	G	H
1			CMMI					
2	Workstream	Activity	Level	PA	Goal	Prac		
78			3	IT	2	2.1		
79		Develop Lifestyle Snapshot	3	RD	1	1.1	Elicit Needs, Establish Operational Scenarios	
80			3	RD	3	3.1		
81		Review Vision Statement	3	IPM	2	2.1		
82			3	VER	2	2.2		
83								
84	Close a Bug	Verify a Fix	3	VER	3	3.1	Perform Test	
85		Close a Bug						
86								
87								
88	Create a Quality of Service Requirement							
89		Brainstorm Quality of Service Reqs	2	REQM	1	1.1	Understanding Requirements. Elicit Needs	
90			3	RD	1	1.1		
91			3	RD	1	1.2		
92		Prioritize Quality of Service Req	3	RD	3	3.4		
93		Write Quality of Service Req	3	RD	3	3.1		
94		Validate Quality of Service Req	3	RD	3	3.5	Develop Requirements	
95		Write UAT	3	VAL	1	1.3	Develop Requirements	
96								
97	Create a Scenario							
98		Brainstorm Scenarios	2	REQM			Understanding Requirements, Elicit Needs, Establish Operational Scenarios	
99			3	RD				
100			3	RD				
101								
102		Prioritize Scenarios						
103		Write Scenarios						
104								
105		Storyboard Scenarios						
106								
107		Validate Scenarios						
108		Write User Acceptance Tests						
109								
110	Create Product Requirements							
111		Allocate Product Component Requirements	3	RD	2	2.2		
112		Define Interface Requirements	3	RD	2	2.2		
113								
114								
115								
116								
117								
118								
119								
120								
121								
122								
123								
124								
125								
126								
127								
128								
129								
130								
131								
132								
133								
134								
135								
136								
137								
138								
139								
140								
141								
142								
143								
144								
145								
146								
147								
148								
149								
150								
151								
152								
153								
154								
155								
156								
157								
158								
159								
160								
161								
162								
163								
164								
165								
166								
167								
168								
169								
170								
171								
172								
173								
174								
175								
176								
177								
178								
179								
180								
181								
182								
183								
184								
185								
186								
187								
188								
189								
190								
191								
192								
193								
194								
195								
196								
197								
198								
199								
200								
201								
202								
203								
204								
205								
206								
207								
208								
209								
210								
211								
212								
213								
214								
215								
216								
217								
218								
219								
220								
221								
222								
223								
224								
225								
226								
227								
228								
229								
230								
231								
232								
233								
234								
235								
236								
237								
238								
239								
240								
241								
242								
243								
244								
245								
246								
247								
248								
249								
250								
251								
252								
253								
254								
255								
256								
257								
258								
259								
260								
261								
262								
263								
264								
265								
266								
267								
268								
269								
270								
271								
272								
273								
274								
275								
276								
277								
278								
279								
280								
281								
282								
283								
284								
285								
286								
287								
288								
289								
290								
291								
292								
293								
294								
295								
296								
297								
298								
299								
300								
301								
302								
303								
304								
305								
306								
307								
308								
309								
310								
311								
312								
313								
314								
315								
316								
317								
318								
319								
320								
321								
322								
323								
324								
325								
326								
327								
328								
329								
330								
331								
332								
333								

Agenda

- Why Process?
- MSF for Agile Software Development
- MSF for CMMI Process Improvement
- Trustworthy Transparency
- Continuous Improvement
- Implementing MSF

david.anderson@microsoft.com

© Microsoft Corporation 2006

Trustworthy Transparency

- *VSTS lets everyone see exactly what is happening to all the customer valued work on a project. That transparency is trustworthy because it comes from the same tool that is used to do the work. It necessarily has to reflect the working reality of the project.*

david.anderson@microsoft.com

© Microsoft Corporation 2006

Friction-Free Data Collection

Check In - Source Files - Workspace: PERSONID-BHITS3_Test1_v-jpercz

Changed the Web Service client call from Synchronous to Asynchronous.
Adding new Build Verification Tests

Name	Change	Folder
Windows.sp.sp2.gm.testrunco...	add	c:\T\
WindowsServer2003.testrunco...	add	c:\T\
Build Verification Tests.csproj...	edit	C:\T\Build Verification Tests
Build Verification Tests.csproj...	edit	C:\T\Build Verification Tests
Message.xml	add	C:\T\Build Verification Tests
RegisterLoadTest	add	C:\T\Build Verification Tests
Teambook Unit Test.k		
TeambookWPTest.k		
ThisApplicationTest.c		
TLAboutTest.cs		
TLProjectFolderTest.cs		
TLProjectFolderTest.c		
TLReconnectingTest.c		
Teambook.csproj		
Teambook.csproj.vsp		
AuthenticatingTests.txt		
RegisterForAccount		

Check In - Work Items (Shareware Starter Kit/Team Queries/My Work Items)

Query: My Work Items

Work Item ...	ID	Title	State	Check-in Action
Bug	6886	Web service client code needs to be async	Active	Resolve

Check In

Cancel

Check-in dialog captures the work items resolved with the code (and tests) delivered

Policy makes sure that the association is made (and unit tests and code analysis have been run)

david.anderson@microsoft.com

© Microsoft Corporation 2006

MSF for CMMI Process Improvement

Visual Studio Team System

Overview Roles Work Items Views CMMI Index

Glossary

Project Portal >

Workstreams

Activities

Work Items

Work Products

Reports

Queries

How To's

Index Reports

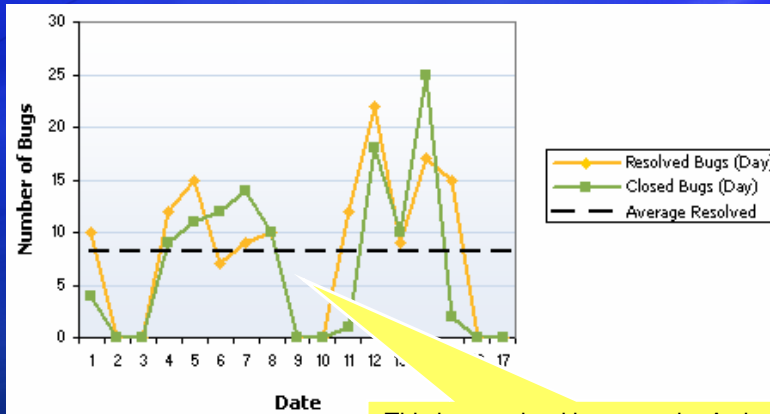
Project health charts aggregate metrics from work items, source control, test results, and builds. They answer questions about the actual state of your project at many scales: for the days within an iteration, iterations within a project, or projects within a program. The questions are also relevant for many kinds of work items such as scenarios, quality of service requirements, tasks, and bugs.

- Actual Quality versus Velocity
- Bug Rates
- Bugs by Priority
- Builds
- Open Issues and Blocked Work Items Trend
- Quality Indicators
- Reactivations
- Regressions
- Related Work Items
- Remaining Work
- Requirements Test History
- Scenario Details
- Test Failure Without Active Bug
- Test Passing With Active Bug
- Triage
- Unplanned Work
- Velocity
- Work Items
- Work Items by Owner
- Work Items by State

david.anderson@microsoft.com

© Microsoft Corporation 2006

Velocity



This is an unhealthy example. Action should be taken to resolve issues blocking work items and to change the working environment to reduce randomization of team members.

david.anderson@microsoft.com

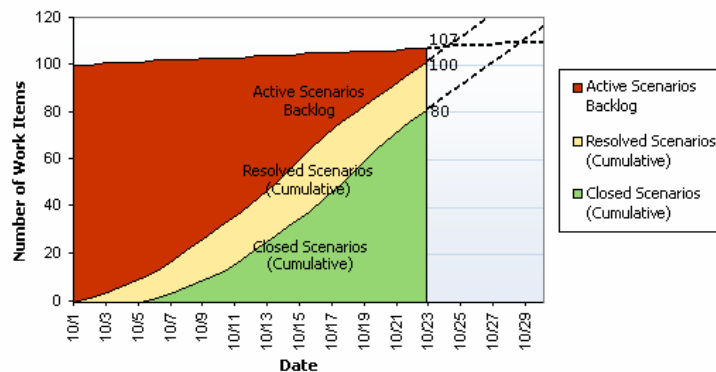
© Microsoft Corporation 2006

Work Remaining

Remaining Work: Scenarios

Report generated: 11/04/2004 11:25 AM by someone@example.com

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



david.anderson@microsoft.com

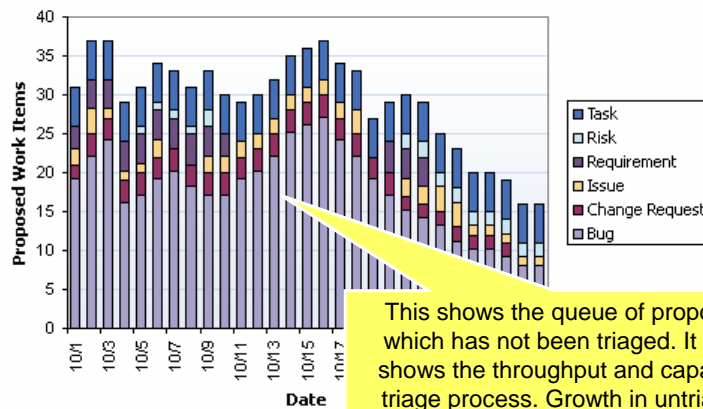
© Microsoft Corporation 2006

Triage

Triage

Report generated: 11/04/2004 11:25 AM by someone@example.com

Which proposed work items have not yet been triaged?



This shows the queue of proposed work which has not been triaged. It effectively shows the throughput and capacity of the triage process. Growth in untriated work items may indicate more time and resources need to be spent on triage activities.

david.anderson@microsoft.com

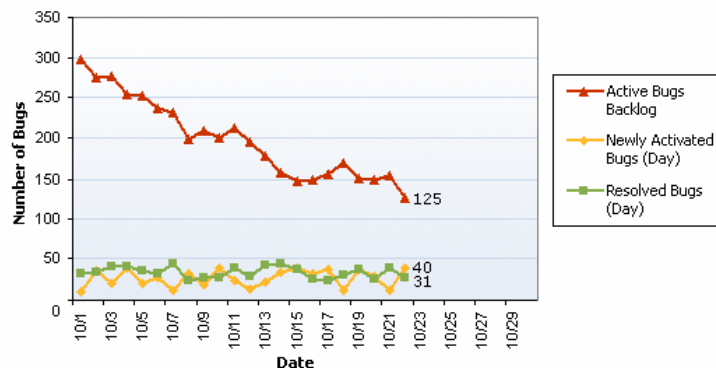
© Microsoft Corporation 2006

Bug Rates

Bug Rates

Report generated: 11/04/2004 11:25 AM by someone@example.com

How effectively are we finding, fixing and closing bugs?



david.anderson@microsoft.com

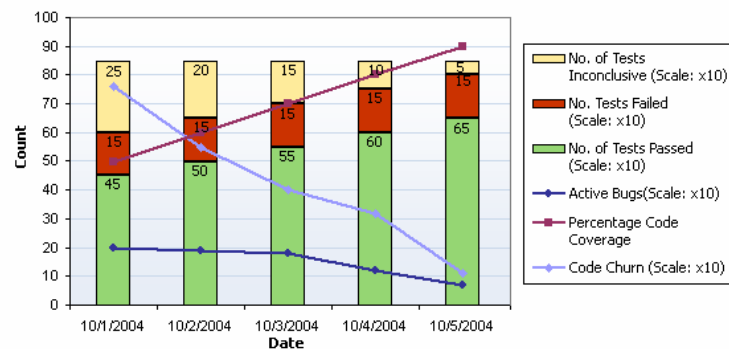
© Microsoft Corporation 2006

Quality Indicators

Quality Indicators

Report generated: 11/04/2004 11:25 AM by someone@example.com

What is the quality of the software?



david.anderson@microsoft.com

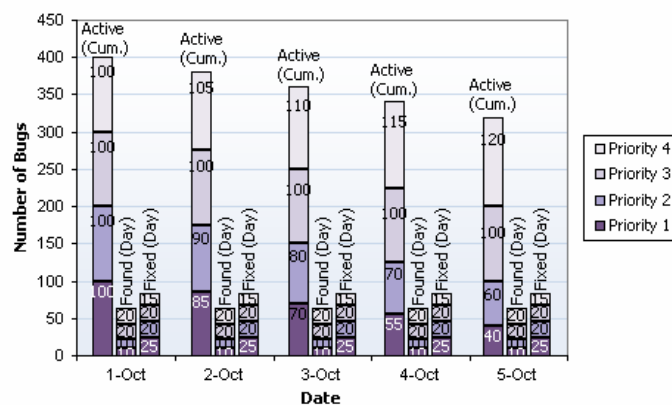
© Microsoft Corporation 2006

Bugs by Priority

Bugs by Priority

Report generated: 11/04/2004 11:25 AM by someone@example.com

Are we finding and triaging the right bugs?



david.anderson@microsoft.com

© Microsoft Corporation 2006

Builds

Builds
Report generated: 11/04/2004 11:25 AM by someone@example.com

Which builds are available and what are the build details for each?

Build ID	Build Quality	% Tests Passed	% Code Coverage	% Code Churn
41001.00	Failed	75%	42%	1.7%
Test Results: 4/4 test runs completed, 16/20 tests passed				
Test Run				
		Total Tests	Passed	Failed
BVT_TPSv21_04110600		5	3	1
Test Name				
TestMethod01	Passed	Nightly	DevMachine-00	This is a description of the test.
TestMethod02	Passed	Nightly	DevMachine-00	This is a description of the test.
TestMethod03	Passed	Nightly	DevMachine-00	This is a description of the test.
TestMethod04	Failed	Nightly	DevMachine-00	This is a description of the test.
TestMethod05	Inconclusive	Nightly	DevMachine-00	This is a description of the test.
BVT_TPSv21_04110601		5	4	1
BVT_TPSv21_04110602		5	4	1
BVT_TPSv21_04110603		5	4	1
Code Coverage Results: 42% lines covered, 49% lines not covered, 9% lines covered				
Assembly				
CurrencyConverter.dll	Covered (Lines)	Not Covered (Lines)	Partially Covered (Lines)	Date
	330	600	70	04/15/2005
StoreInventory.dll	600	480	120	04/15/2005
Code Churn Details: 25990 Total LOC, 450 Churned LOC, 89 Deleted LOC				
Relative Code Churn Measure				
	Value			
Churned LOC/Total LOC	.017			
Deleted LOC/Total LOC	.003			
Churned LOC/Deleted LOC	5.056			
41001.01	Failed	72%	50%	1.7%
41001.02	Failed	70%	50%	1.7%

david.anderson@microsoft.com

© Microsoft Corporation 2006

Tests Failing Without Bugs

Tests Failing Without Active Bugs
Report generated: 11/04/2004 11:25 AM by someone@example.com

Are there failing tests that don't have associated active bugs?

Test Name	Category	Machine	No. of Bugs	Description
TestMethod01	Nightly	DevMachine-001	5	This is a description of the test.
Bug ID				
1176	Resolved	Credit card validation delays the checkout		
1190	Resolved	Security settings notification when cookies are disabled		
1203	Resolved	Tab order incorrect		
1204	Resolved	Required fields need correct background color		
1205	Resolved	Do not cache credit card information on checkout page		
TestMethod02	Nightly	DevMachine-001	2	This is a description of the test.
TestMethod03	Nightly	DevMachine-001	1	This is a description of the test.
TestMethod04	Nightly	DevMachine-001	1	This is a description of the test.
TestMethod05	Nightly	DevMachine-001	4	This is a description of the test.

david.anderson@microsoft.com

© Microsoft Corporation 2006

Tests Passing with Active Bugs

Tests Passing With Active Bugs
Report generated: 11/04/2004 11:25 AM by someone@example.com

Are there passing tests that do have active bugs?

Assigned To	Total Tests
<input checked="" type="checkbox"/> Brian Cox	5

Test Name	Category	Machine	Total Bugs	Description
<input checked="" type="checkbox"/> TestMethod11	Nightly	DevMachine-00	4	This is a description of the test.
<input checked="" type="checkbox"/> TestMethod12	Nightly	DevMachine-00	6	This is a description of the test.
<input checked="" type="checkbox"/> TestMethod13	Nightly	DevMachine-00	5	This is a description of the test.
<input checked="" type="checkbox"/> TestMethod14	Nightly	DevMachine-00	5	This is a description of the test.
<input checked="" type="checkbox"/> TestMethod15	Nightly	DevMachine-00	3	This is a description of the test.

Bug ID	State	Title
1176	Active	Client timeout when checking product availability
1190	Active	Welcome page not picking up user ID
1203	Active	Adding a new column to the catalog table sheet triggers exception

<input checked="" type="checkbox"/> Doris Krieger	8
<input checked="" type="checkbox"/> Fernando Sousa	11
<input checked="" type="checkbox"/> Judy Lew	10
<input checked="" type="checkbox"/> Martin Weber	7

david.anderson@microsoft.com

© Microsoft Corporation 2006

Requirements Details

Scenario Details
Report generated: 11/04/2004 11:25 AM (GMT -08:00) by someone@example.com

Are we making progress with our scenarios?

Rank	Scenario	Specified	Overall Status
1	Browse products by category	Yes	
	<input checked="" type="checkbox"/> Architecture Tasks	45 Closed	■ Completed
	<input checked="" type="checkbox"/> Development Tasks	12 Closed	■ Completed
	<input checked="" type="checkbox"/> Test Tasks	37 Closed	■ Completed
	<input checked="" type="checkbox"/> Test Cases	82 Passed, 82 Run	
2	Order for in-store pickup	Yes	
	<input checked="" type="checkbox"/> Architecture Tasks	22 Closed	■ Completed
	<input checked="" type="checkbox"/> Development Tasks	3 Active, 2 Resolved, 1 Closed	■ In Progress
	<input checked="" type="checkbox"/> Test Tasks	10 Active, 12 Resolved, 14 Closed	■ In Progress
	<input checked="" type="checkbox"/> Test Cases	45 Passed, 65 Run	
3	Search local inventory	Yes	
	<input checked="" type="checkbox"/> Architecture Tasks	4 Active, 5 Resolved, 45 Closed	■ In Progress
	<input checked="" type="checkbox"/> Development Tasks	4 Active, 2 Resolved, 1 Closed	■ In Progress
	<input checked="" type="checkbox"/> Test Tasks	23 Active, 5 Resolved, 10 Closed	■ In Progress
	<input checked="" type="checkbox"/> Test Cases	3 Passed, 4 Run	

david.anderson@microsoft.com

© Microsoft Corporation 2006

Agenda

- Why Process?
- MSF for Agile Software Development
- MSF for CMMI Process Improvement
- Trustworthy Transparency
- Continuous Improvement
- Implementing MSF

david.anderson@microsoft.com

© Microsoft Corporation 2006

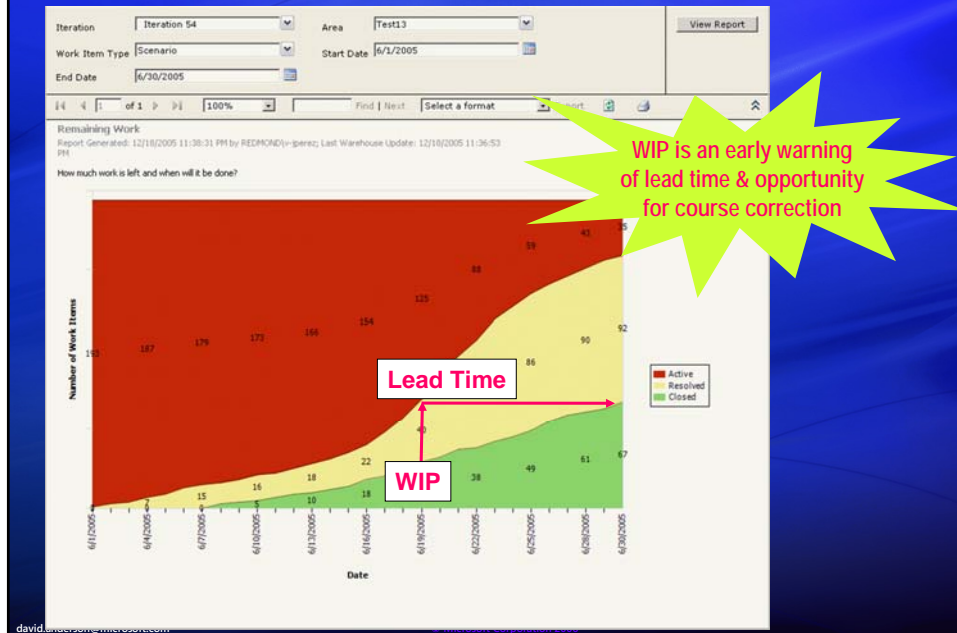
Continuous Improvement

- MSF enables Lean Thinking, Constraints Management, and Six Sigma for IT

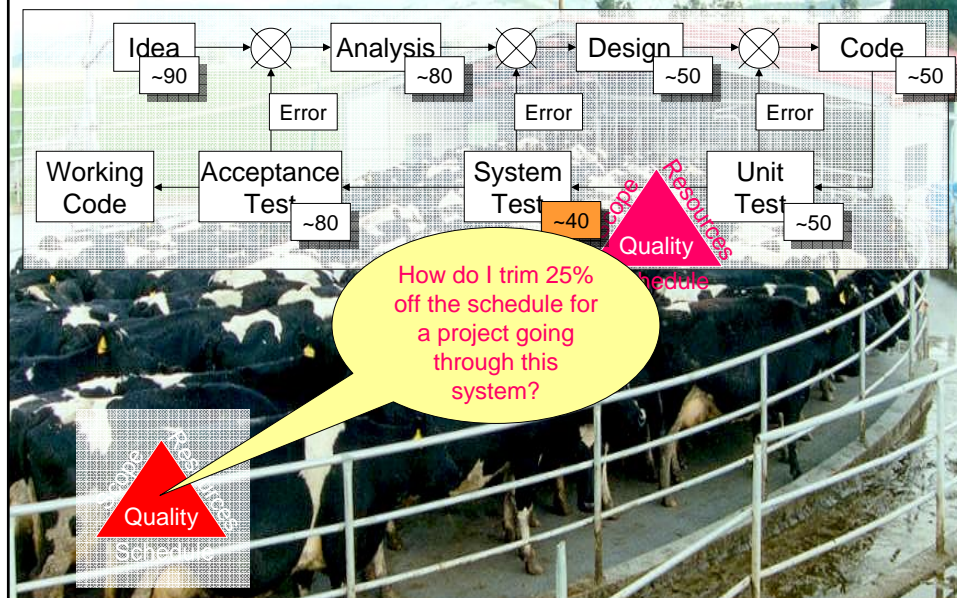
david.anderson@microsoft.com

© Microsoft Corporation 2006

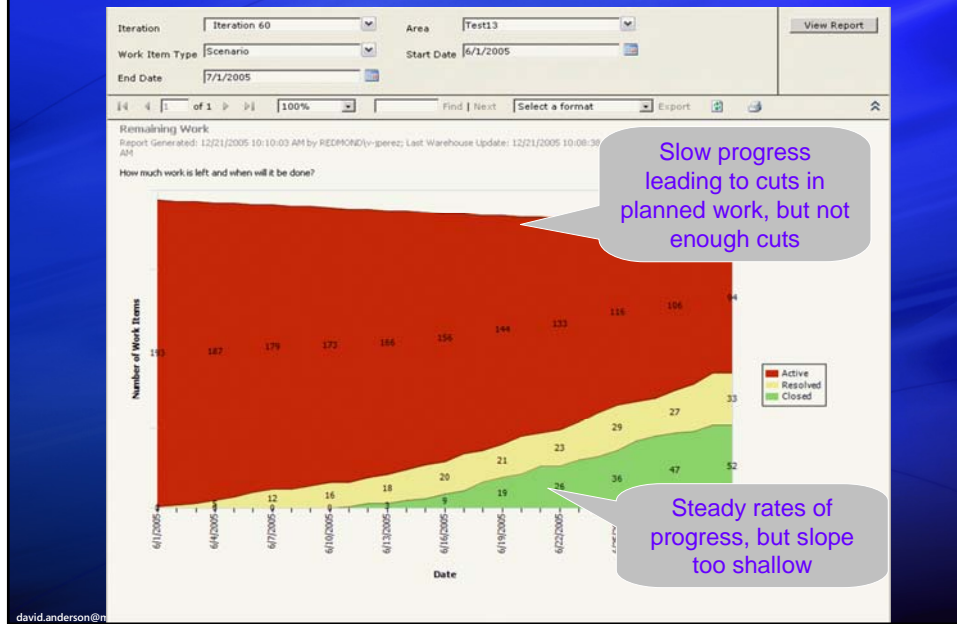
Lean: Managing with Queues



What is Constraints Management?



Constraints Become Visible

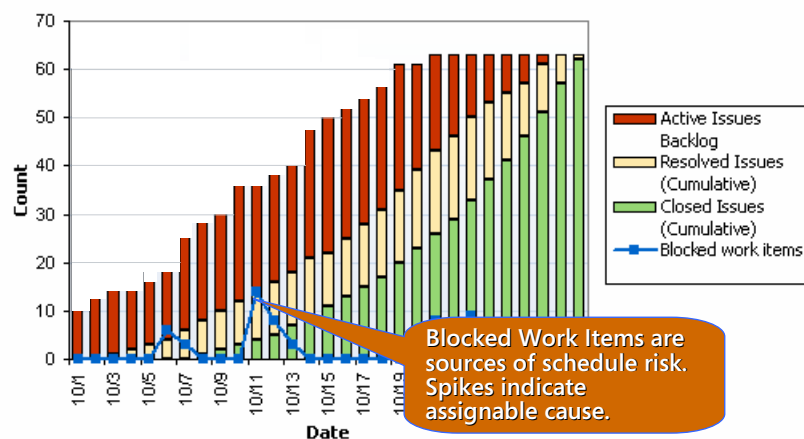


Removing Unforeseen Causes of Risk

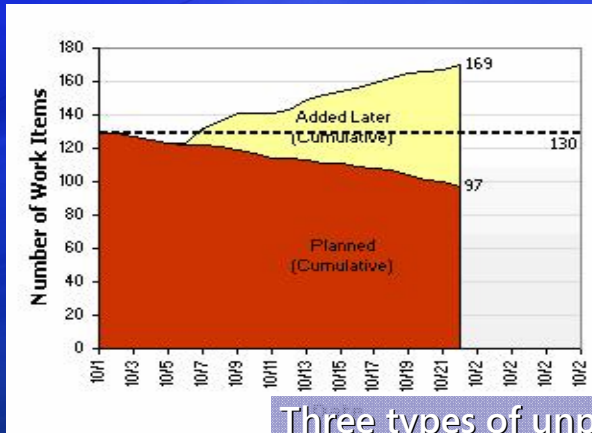
Issues and Blocked Work Items

Report generated: 11/04/2004 11:25 AM by someone@example.com

Are issues causing work items to block?



Unplanned Work Report



Agenda

- Why Process?
- MSF for Agile Software Development
- MSF for CMMI Process Improvement
- Trustworthy Transparency
- Continuous Improvement
- Implementing MSF

david.anderson@microsoft.com

© Microsoft Corporation 2006

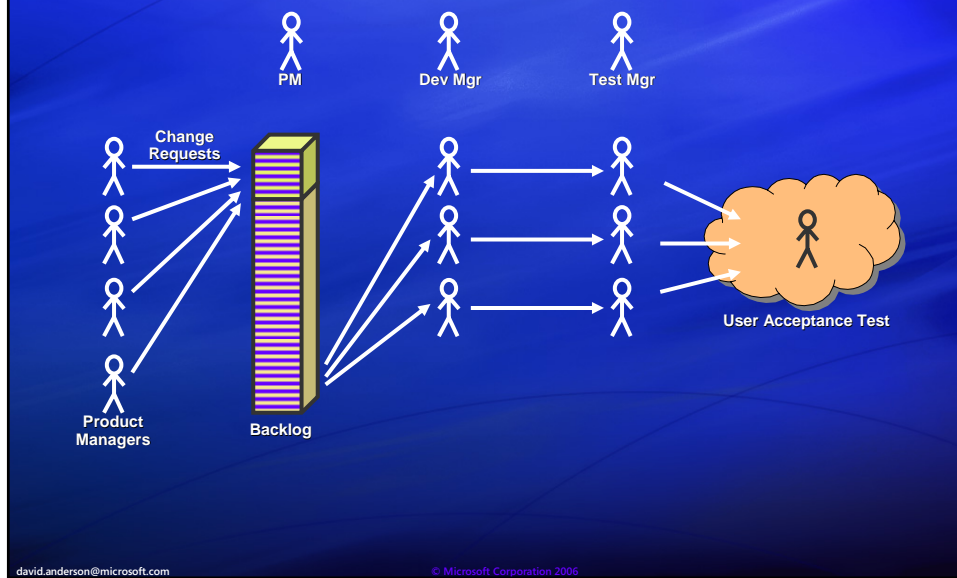
Implementing MSF

- Start with Workflow Modeling - Design Work Item Types for Team Foundation Server to enable reporting in context

david.anderson@microsoft.com

© Microsoft Corporation 2006

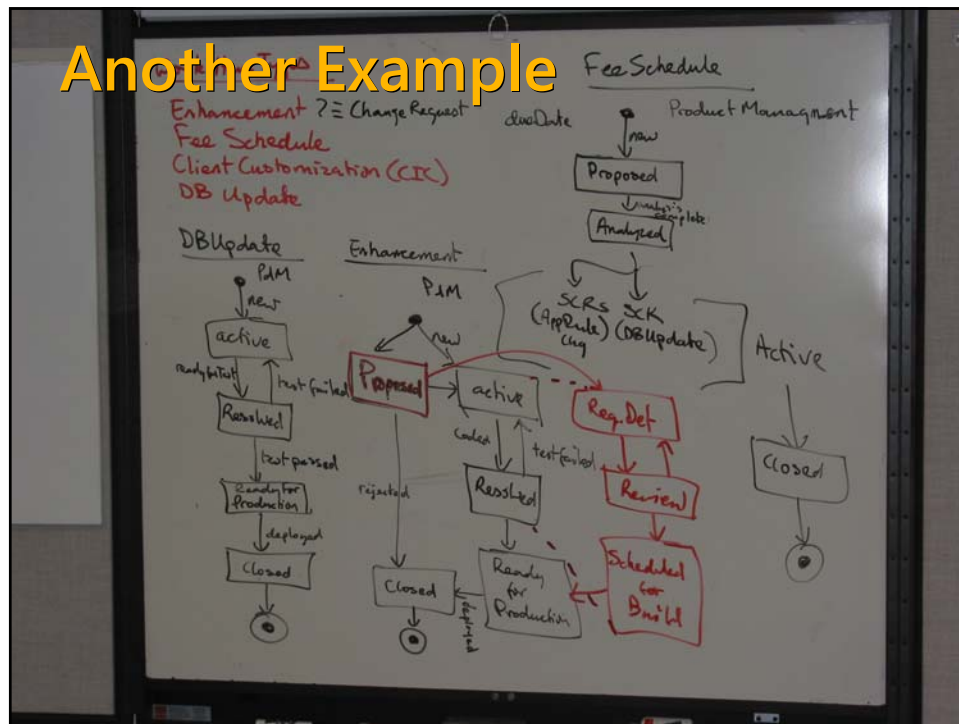
Workflow Modeling



Doing It For a Real Customer



Another Example



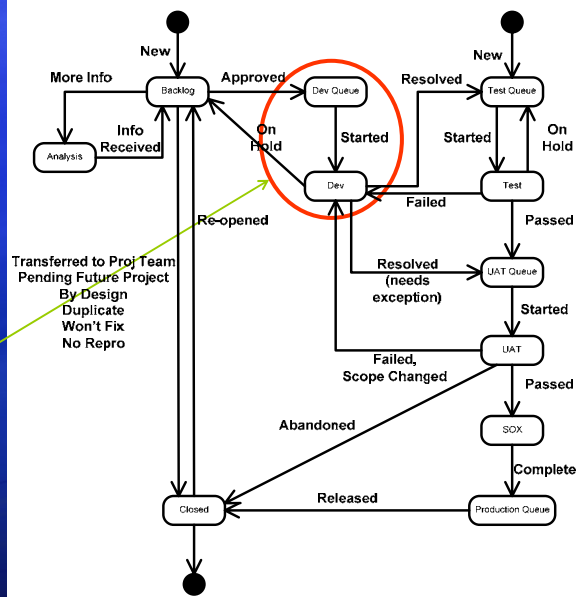
State Model

Modeling the flow enables and informs

- reduction of waste,
- management of constraints, and
- reduction of variance with VSTS reporting

Example:
Discovery of capacity allowed Virtual Kanban limit as solution, initially 8 = WIP + 7 days buffer

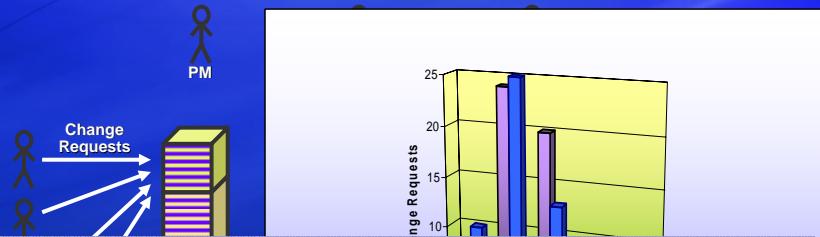
XIT Change Request



david.anderson@microsoft.com

© Microsoft Corporation 2006

How much waste is there in the process?

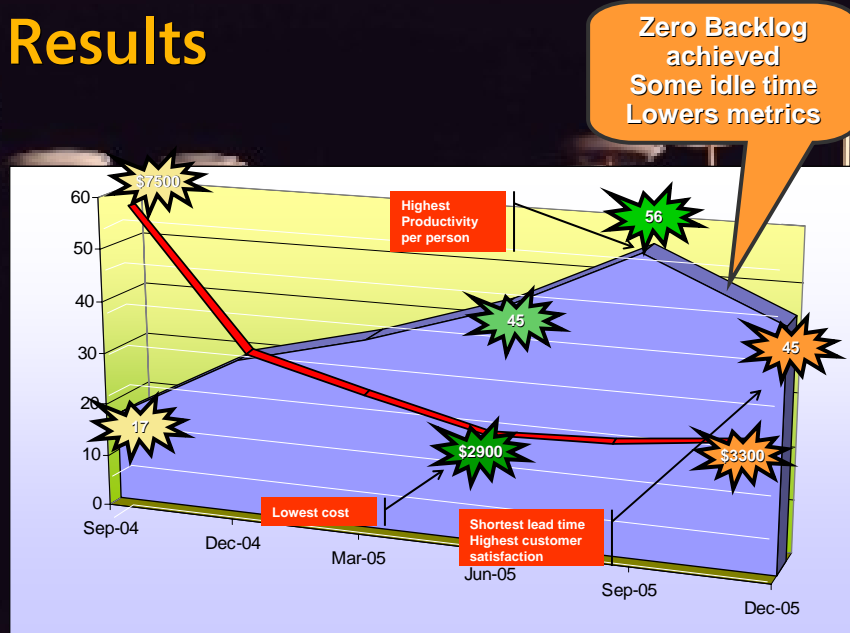


- Touch time mean was 11 days
- Distribution (spread) is tight
- But, Lead Time was 155 days
- **Touch Time : Lead Time Ratio < 1 : 10**
- At least 90% of this process is waste

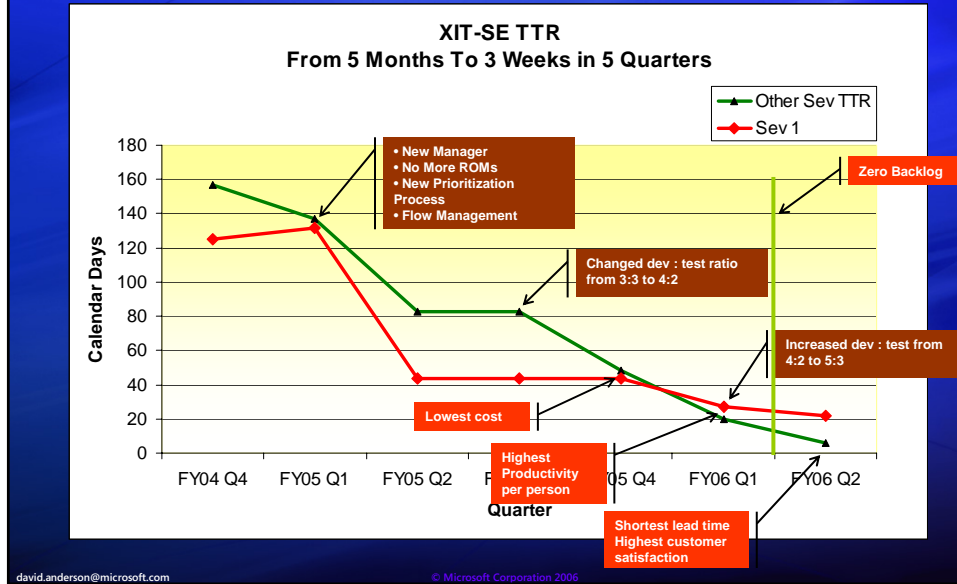
david.anderson@microsoft.com

© Microsoft Corporation 2006

Results



Change Request Lead Time



Bottom Line

- Realize Your Potential
 - ▣ Model workflow
 - ▣ Enable reporting
 - ▣ Achieve Trustworthy Transparency
 - ▣ Drive positive cultural change
- Deliver Continuous Improvement
 - ▣ Agile
 - ▣ CMMI, Lean, Six Sigma

Agenda

- ✓ Why Process?
- ✓ MSF for Agile Software Development
- ✓ MSF for CMMI Process Improvement
- ✓ Trustworthy Transparency
- ✓ Continuous Improvement
- ✓ Implementing MSF

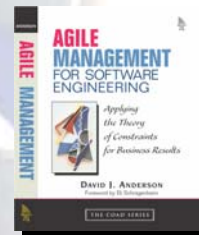
david.anderson@microsoft.com

© Microsoft Corporation 2006



<http://www.agilemanagement.net/>

- Book published September 2003
- Winner - **Best Project Management Blog 2005** by readers of Inside Blogging
- Agile Project Leadership Network Founder <http://www.apln.org/>



For More Information...



david.anderson@microsoft.com

© Microsoft Corporation 2006

References

- **MSF Site**
 - [//msdn.microsoft.com/msf/](http://msdn.microsoft.com/msf/)
 - [//www.agilemanagement.net/](http://www.agilemanagement.net/)
 - CMMI Appraisers' workshop
 - [//msdn.microsoft.com/vstudio/teamsystem/msf/appraisers/default.aspx](http://msdn.microsoft.com/vstudio/teamsystem/msf/appraisers/default.aspx)
- **VSTS Content**
 - [//vsts/](http://vsts/)
- **Competitive Content**
 - [//evangelism/vs/compete/](http://evangelism/vs/compete/)
- **MSF Webcasts**
 - [Introducing Microsoft Solutions Framework for CMMI Process Improvement \(Level 100\)](#)
 - [Returning to the Roots of CMMI \(Level 200\)](#)
 - [Microsoft Solutions Framework for CMMI Process Improvement and the Standard CMMI Assessment Method for Process Improvement \(SCAMPI\) \(Level 200\)](#)

david.anderson@microsoft.com

© Microsoft Corporation 2006

