



New Synergy Group

Systems Thinking... Enterprise Performance

Turbo-Charging Agile Software Development **with** **Lean Methods and Systems Thinking**

Technical Seminar
Philly Spin Meeting
Vertex, Inc.

1041 Old Cassatt Road, Berwyn, PA 19312
27 September 2011

Dr. Satish Thatte
CEO, New Synergy Group
Kendall Park, NJ
732.410.9990

www.NewSynergyGroup.com



Agenda

- Challenges facing software industry
- Waterfall vs. Agile/Scrum
- Overview of Agile/Scrum Framework
- Agile/Scrum framework in Action
- Overview of Lean Framework
- Lean framework in action
- Journey from Waterfall/Ad hoc to Agile to Agile-Lean
- Overview of Systems thinking
- Synergies among Agile/Scrum, Lean and Systems Thinking
- **Strategy and Operations** must be well-aligned



Challenges facing Software-Intensive Industry

- Failure to meet the real requirements of users
- Delays in time to market
- Cost overruns
- Low quality, productivity, or “Productivity x Quality”
- Failure to meet important goals:
 - Performance
 - Usability
 - Scalability
 - Security
 - Extensibility, etc.
- Failure to understand and resolve *organizational systems* issues
- These challenges *will* continue to be with us, as the bar is rising:
 - Ever more ambitious and complex software systems
 - Rising expectations of users
 - Fierce global competition to deliver innovative solutions and services in shorter cycle times
 - Increasingly volatile business environment
 - Fast changing technology platforms and interoperability with legacy applications
 - Little change in human nature!



Example: Smart Phone Product

- **Basic phone operations:** Make call, Receive call
- **Set-Up:** Sound, Display, Ring tones, Blue Tooth
- **Contact Management:** New, List, Group, Search
- **Call Log Management:** Dialed, Received, Missed, All
- **Messaging:** NewText, InBox, Voice Mail, MobileIM, Chat
- **Tools:** Calendar, WorldClock, AlarmClock, Notes
- **Music:** Play Lists, Artists, Genre, Album, Settings & controls
- **Camera:** Picture, Album, Settings



Waterfall vs. Scrum

		Waterfall (Release 1.0): 6 months .. delay .. 9 months				
Sequential Engineering	Analysis	Design	Code		Test	
	Basic phone Set-Up Tools	Basic phone Set-Up Tools	Basic phone Set-Up Tools		Basic phone Set-Up Tools	
		Scrum (Release 1.0)				
Concurrent Engineering	Sprint 0 Architecture Framework	Sprint 1 Basic phone	Sprint 2 Set-up	Sprint 3 Tool Set	Hardening Sprint 4	
	Analysis	Analysis	Analysis	Analysis	Analysis	
	Design	Design	Design	Design	Design	
	Code	Code	Code	Code	Code	
	Test	Test	Test	Test	Test	

Waterfall development depends on a perfect understanding of requirements at the outset and minimal errors in each phase.

Scrum blends all development activities into each sprint, adapting to discovered realities.

Consider feedback from each sprint to the next sprint



Scrum Framework

- Develop complex products, while productively and creatively delivering highest possible value.
- Employ appropriate processes, practices and techniques within the framework that suits your organization.
- The framework: Scrum Team, and **3 roles**, **4 artifacts**, **5 events**, and many rules that bind them.
- Based on empirical process control: Transparency, Inspection, Adaptation
 - **Lightweight process**
 - **Simple to understand:** 3 roles, 4 artifacts, 5 events, and rules
 - **Extremely difficult to master:** How to use customized processes, practices, and techniques consistently throughout an organization; How not to be a *Scrumbut*; How to do continuous improvements; Scalability

3 Roles

- Product Owner (*What*)
- ScrumMaster (*How*)
- Development Team

4 Scrum Artifacts

- Product Backlog
- Sprint Backlog
- Burn-down Chart
- Sprint Increment

5 Scrum Events

- Sprint Planning Meeting
- Sprint
- Daily Scrum
- Sprint Review
- Sprint Retrospective

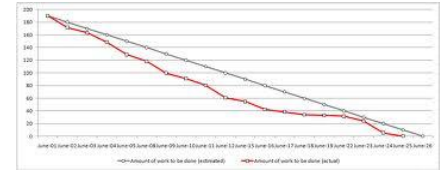


Scrum Framework: In Action

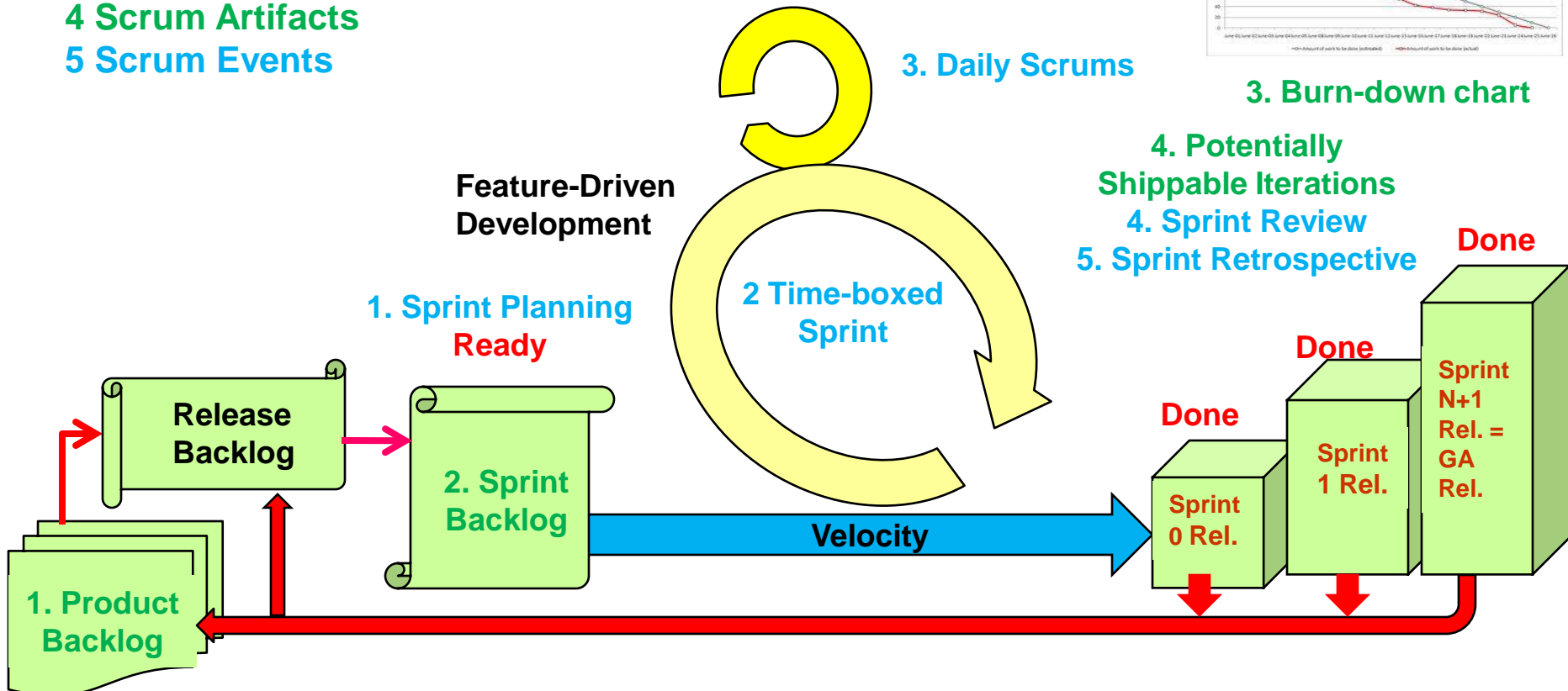
3 Roles

4 Scrum Artifacts

5 Scrum Events



3. Burn-down chart





Lean Framework

- **Lean** is the English term given by MIT researchers to describe the system of the *Toyota Way*
- Refers to *Lean production*, as opposed to *Mass production*
- Viewed as a set of operating principles and policies, not as a prescriptive step-by-step methodology.
- Does not mean “lean and mean” – or fire and downsize

Lean Principles & Policies

- Minimize waste
- Obliterate/reduce queues
- Increase the value flow
- Pull management
- Level the work
- Kanban – Visual signaling
- Kaizen – Continuous improvement
- Find and eliminate the root cause



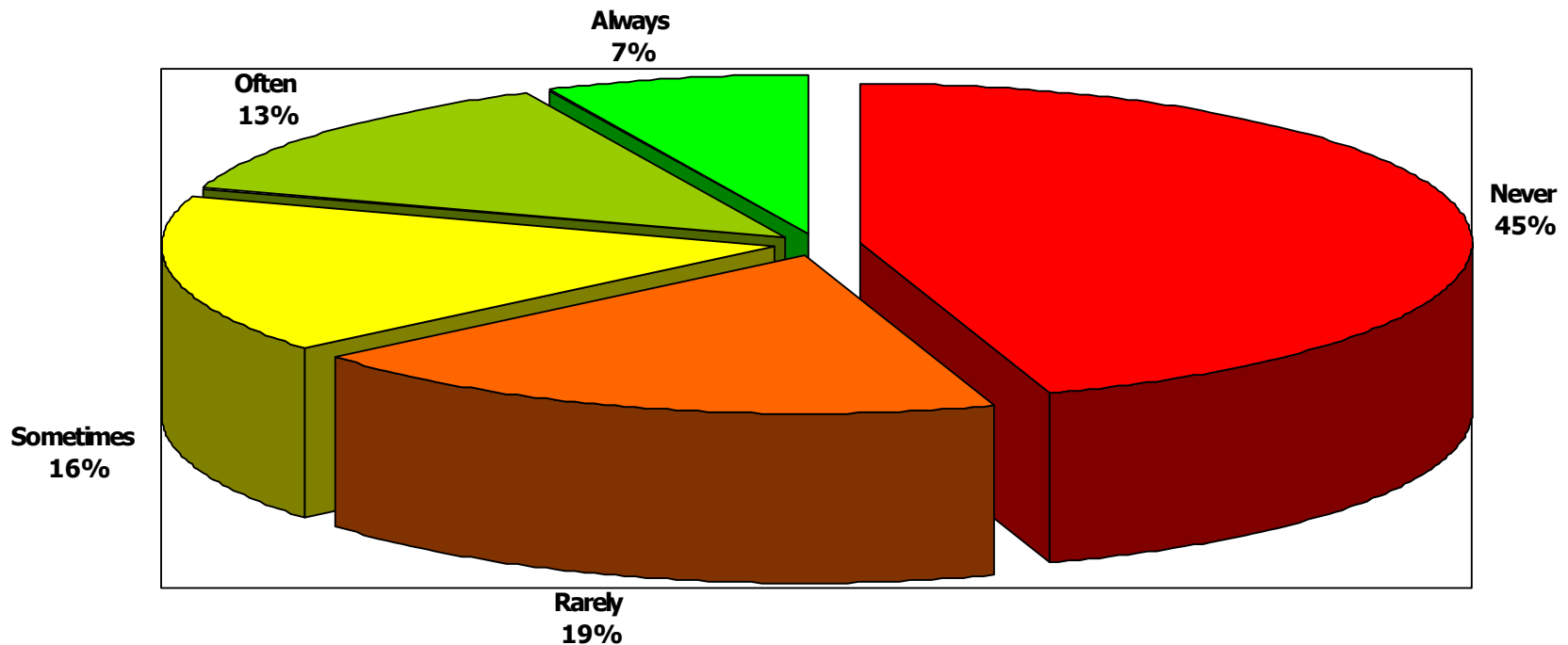
Value Ratio, and its Shocking Truth!

- **Value:** The time periods while developing a product that the customer is willing to pay for. Value is in the eyes of (external) customer.
- **Waste:** All other time periods that do not add value, but take resources
 - Multi-tasking, context-switching workers
 - Waiting in queues
 - Bottlenecks: single experts pulled for every crisis; narrow specializations
 - Hand-offs (in sequential waterfall process)
 - Undo-Redo cycles (due to poor communication, poor understanding, etc.)
- **Value Ratio** = Value time / Lead time (concept to consumption time)
 - Value ratio is less than 7% in many product development organizations!
[Source: Larman & Vodde, 2008; Poppendiek, 2006]
- Value-adding moments (7%), and huge amounts of waste (93%)
 - Improvement strategy needs to be subtractive and not additive
- Requires mindset change and need to develop “Eyes for Waste”

“There is nothing so useless as doing efficiently that which should not be done at all.” -- Peter Drucker



Most Products are Wastefully Over-Engineered



Source: Jim Johnson of the Standish Group, Keynote Speech XP 2002



Kanban Visual Signaling; Pull Management; WIP & Queue Limits

Sprint Backlog	Design (WIP Limit: 2)		Development, Defect Fixing (WIP Limit: 2)		Test Case Design, Testing (WIP Limit: 2)		Acceptance Testing by Product Owner
	WIP	Done Queue	WIP	Done Queue	WIP	Done Queue	
3 Features, 2 Defects							
1. DS,DV,TC, TS, DF, AT							
2. DS,DV,TC, TS, DF, AT							
3. DS,DV,TC, TS, DF, AT							
A. DF, TS							
B. DF, TS							

DS: Design DV: Development TC: Test Case Design TS: Testing DF: Defect Fix AT: Acceptance Test



Kanban Visual Signaling; Pull Management; WIP, Queue Limits

Sprint Backlog	Design (WIP Limit: 2)		Development, Defect Fixing (WIP Limit: 2)		Test Case Design, Testing (WIP Limit: 2)		Acceptance Testing by Product Owner
	WIP	Done Queue	WIP	Done Queue	WIP	Done Queue	
1 Feature, 1 Defect							
	1. Scott DS, DV, TC, TS, DF, AT				1. Susan DS, DV, TC, TS, DF, AT		
	2. John DS, DV, TC, TS, DF, AT				2. Clyde DS, DV, TC, TS, DF, AT		
3. DS, DV, TC, TS, DF, AT							
			A. Jake DF, TS				
B. DF, TS							

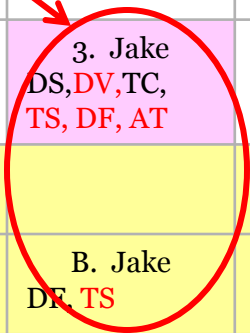
DS: Design DV: Development TC: Test Case Design TS: Testing DF: Defect Fix AT: Acceptance Test



Kanban Visual Signaling; Pull Management; WIP, Queue Limits

Sprint Backlog	Design (WIP Limit: 2)		Development, Defect Fixing (WIP Limit: 2)		Test Case Design, Testing (WIP Limit: 2)		Acceptance Testing by Product Owner
	WIP	Done Queue	WIP	Done Queue	WIP	Done Queue	
0 Feature, 0 Defect			1. Scott DS, DV, TC, TS, DF, AT		1. Susan DS, DV, TC, TS, DF, AT		
			2. John DS, DV, TC, TS, DF, AT				
	3. Jake DS, DV, TC, TS, DF, AT				3. Clyde DS, DV, TC, TS, DF, AT		
				A. DF, TS			
	B. Jake DF, TS						

Jake working on 2 concurrent tasks



DS: Design DV: Development TC: Test Case Design TS: Testing DF: Defect Fix AT: Acceptance Test



Kanban Visual Signaling; Pull Management; WIP, Queue Limits

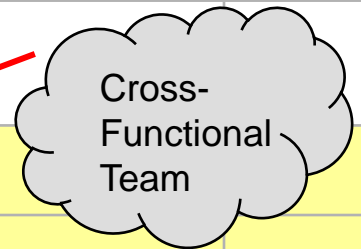
Sprint Backlog	Design (WIP Limit: 2)		Development, Defect Fixing (WIP Limit: 2)		Test Case Design, Testing (WIP Limit: 2)		Acceptance Testing by Product Owner
	WIP	Done Queue	WIP	Done Queue	WIP	Done Queue	
o Feature, o Defect			1. Scott DS, DV, TC, TS, DF, AT				
			2. John DS, DV, TC, TS, DF, AT				
	3. Jake DS, DV, TC, TS, DF, AT						
					A. Susan DF, TS		
					B. Clyde DF, TS		

DS: Design DV: Development TC: Test Case Design TS: Testing DF: Defect Fix AT: Acceptance Test



Kanban Visual Signaling; Pull Management; WIP, Queue Limits

Sprint Backlog	Design (WIP Limit: 2)		Development, Defect Fixing (WIP Limit: 2)		Test Case Design, Testing (WIP Limit: 2)		Acceptance Testing by Product Owner
	WIP	Done Queue	WIP	Done Queue	WIP	Done Queue	
0 Feature, 0 Defect			1. Scott DS, DV, TC, TS, DF, AT				
			2. John DS, DV, TC, TS, DF, AT				
		3. DS, DV, TC, TS, DF, AT					
					A. Susan, Jake DF, TS		
					B. Clyde DF, TS		

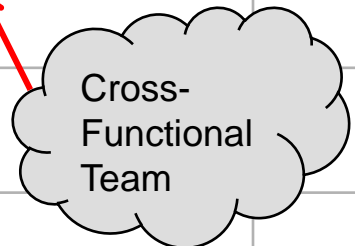
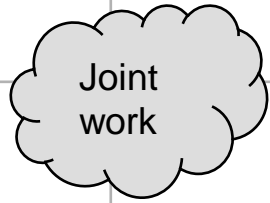


DS: Design DV: Development TC: Test Case Design TS: Testing DF: Defect Fix AT: Acceptance Test



Kanban Visual Signaling; Pull Management; WIP, Queue Limits

Sprint Backlog 0 Feature, 0 Defect	Design (WIP Limit: 2)		Development, Defect Fixing (WIP Limit: 2)		Test Case Design, Testing (WIP Limit: 2)		Acceptance Testing by Product Owner
	WIP	Done Queue	WIP	Done Queue	WIP	Done Queue	
					1. Susan, Scott DS, DV, TC, TS, DF, AT		
				2. DS, DV, TC, TS, DF, AT			
			3. John, Jake DS, DV, TC, TS, DF, AT				
						A. DF, TS	
					B. Clyde DF, TS		



DS: Design DV: Development TC: Test Case Design TS: Testing DF: Defect Fix AT: Acceptance Test



Kanban Visual Signaling; Pull Management; WIP, Queue Limits

Sprint Backlog	Design (WIP Limit: 2)		Development, Defect Fixing (WIP Limit: 2)		Test Case Design, Testing (WIP Limit: 2)		Acceptance Testing by Product Owner
	WIP	Done Queue	WIP	Done Queue	WIP	Done Queue	
o Feature, o Defect							
			1. Scott DS, DV, TC, TS, DF, AT				
					2. Clyde DS, DV, TC, TS, DF, AT		
					3. Susan DS, DV, TC, TS, DF, AT		
						B. DF, TS	

DS: Design DV: Development TC: Test Case Design TS: Testing DF: Defect Fix AT: Acceptance Test



Kanban Visual Signaling; Pull Management; WIP, Queue Limits

Sprint Backlog	Design (WIP Limit: 2)		Development, Defect Fixing (WIP Limit: 2)		Test Case Design, Testing (WIP Limit: 2)		Acceptance Testing by Product Owner
	WIP	Done Queue	WIP	Done Queue	WIP	Done Queue	
o Feature, o Defect					1. Susan DS, DV, TC, TS, DF, AT		
						2. DS, DV, TC, TS, AT	
							3. PO DS, DV, TC, TS, AT

DS: Design DV: Development TC: Test Case Design TS: Testing DF: Defect Fix AT: Acceptance Test



Kanban Visual Signaling; Pull Management; WIP, Queue Limits

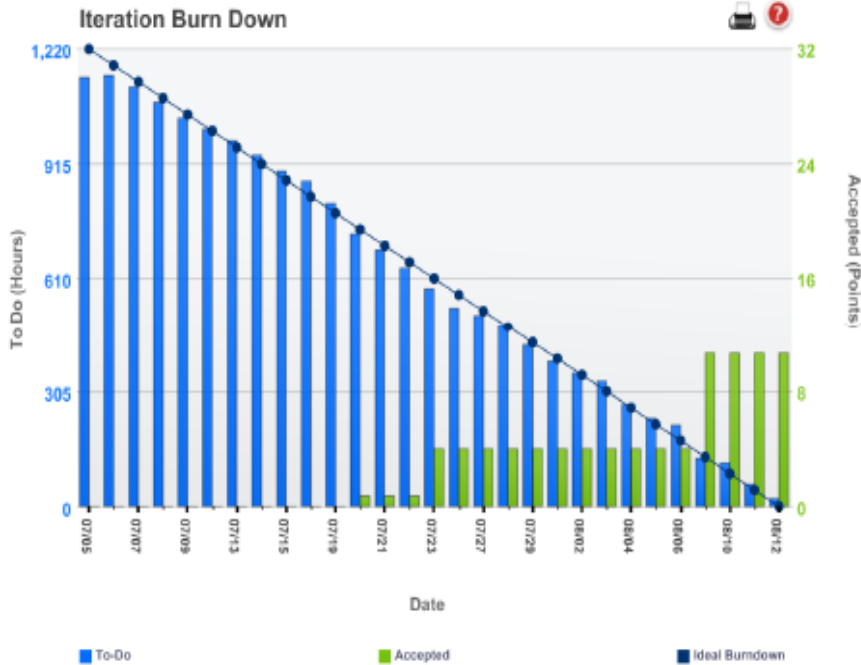
Sprint Backlog	Design (WIP Limit: 2)		Development, Defect Fixing (WIP Limit: 2)		Test Case Design, Testing (WIP Limit: 2)		Acceptance Testing by Product Owner
	WIP	Done Queue	WIP	Done Queue	WIP	Done Queue	
o Feature, o Defect							1. PO DS, DV, TC, TS, DF, AT
							2. PO DS, DV, TC, TS, AT

DS: Design DV: Development TC: Test Case Design TS: Testing DF: Defect Fix AT: Acceptance Test

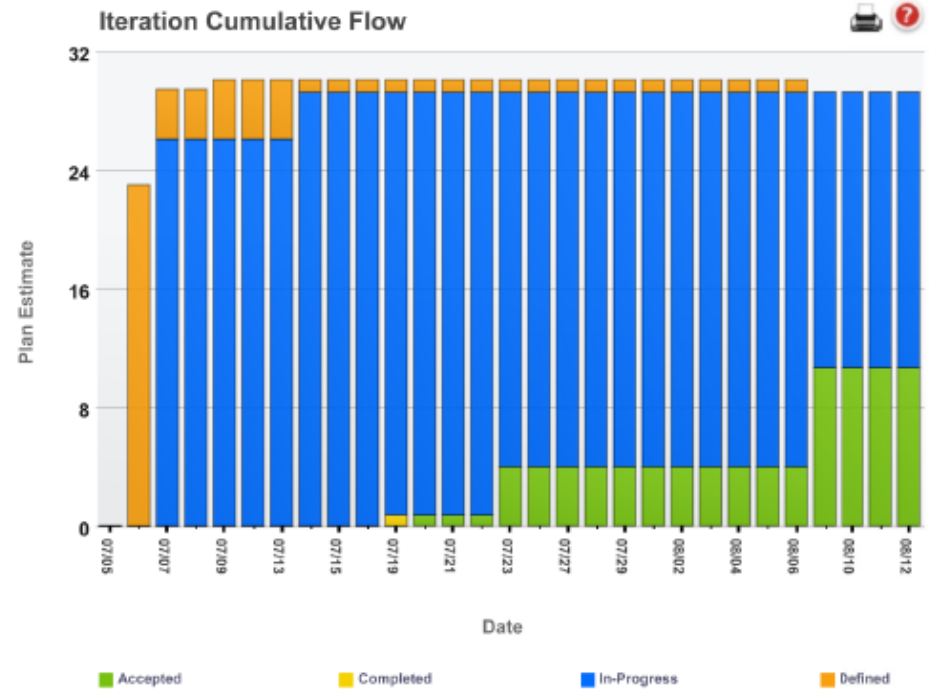


Scrum Team 1: No Emphasis on Lean

Burn-Down, Burn-Up Charts



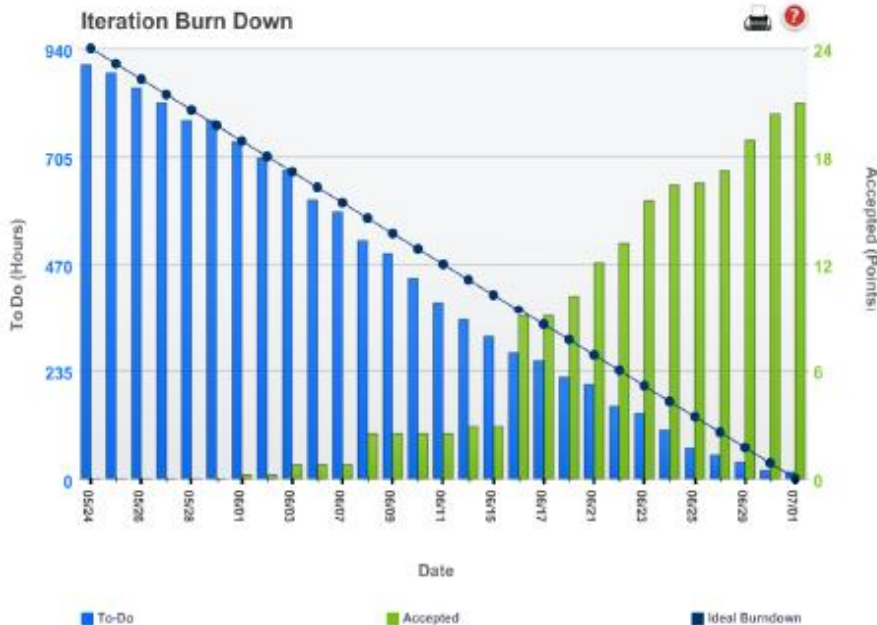
Sprint Cumulative Flow



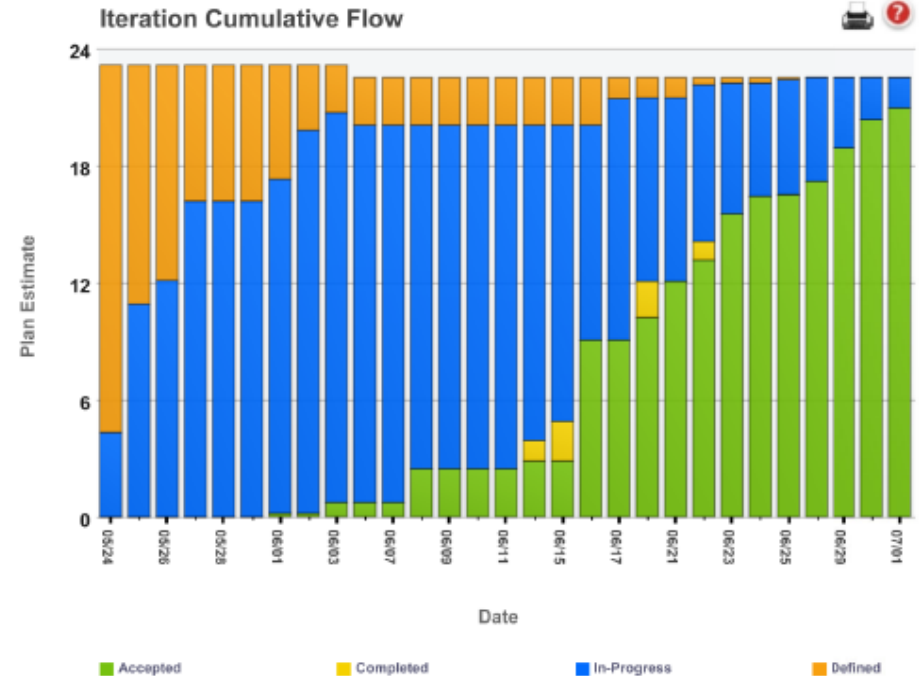


Scrum Team 2: Some Emphasis on Lean Methods

Burn-Down, Burn-Up Charts



Sprint Cumulative Flow

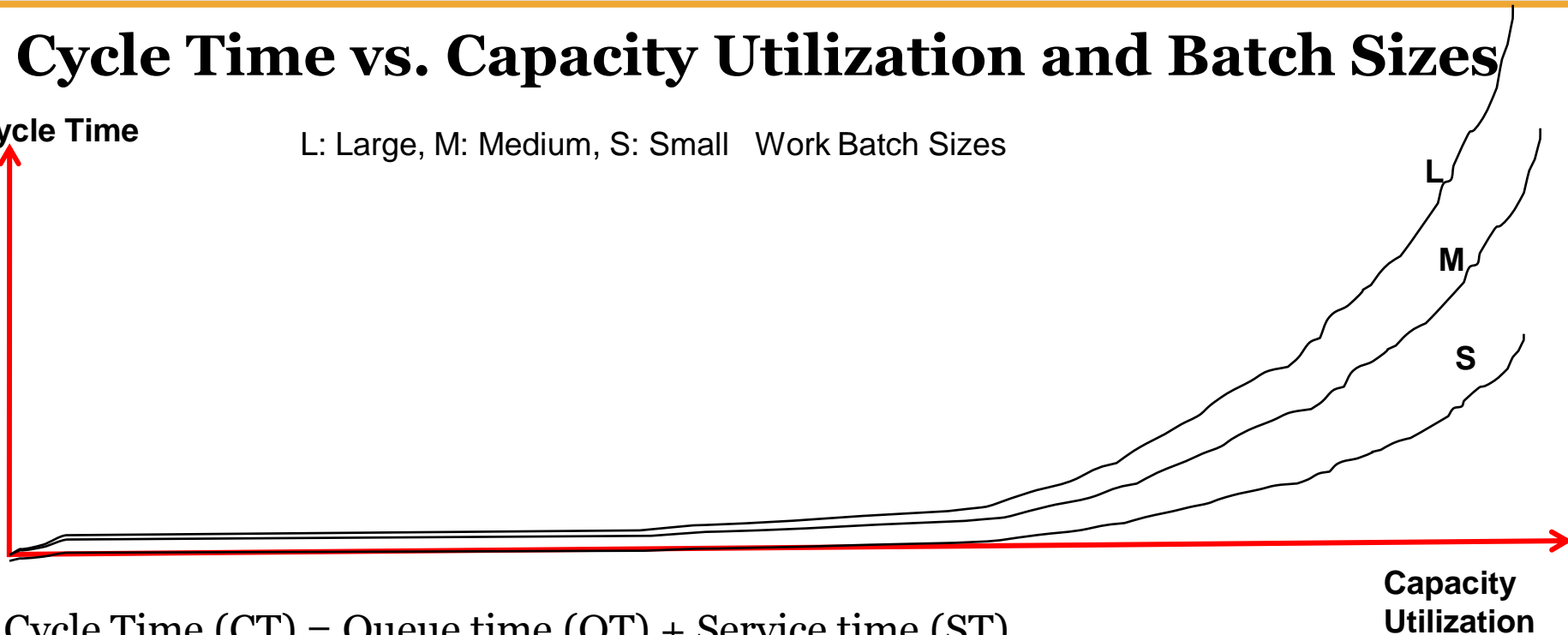




Cycle Time vs. Capacity Utilization and Batch Sizes

Cycle Time

L: Large, M: Medium, S: Small Work Batch Sizes



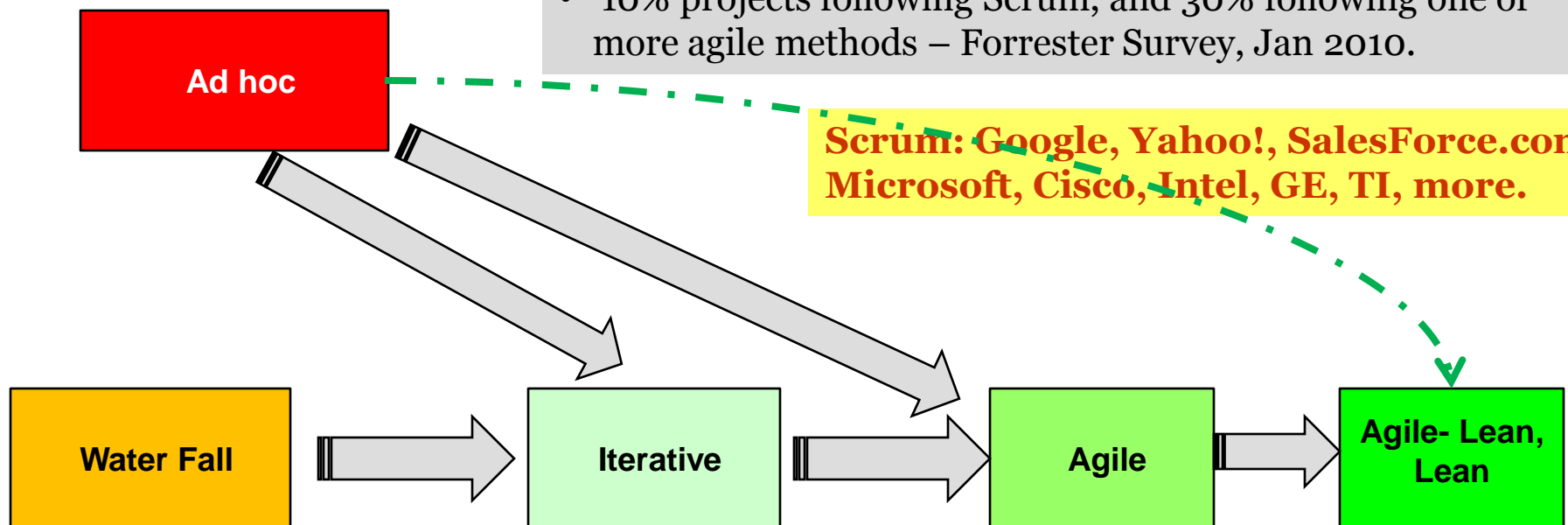
- Cycle Time (CT) = Queue time (QT) + Service time (ST)
- As capacity utilization or batch size increase, queue time increases non-linearly
- Decompose large features into smaller sub-features hierarchy; at the leaf level, no feature should be taking more than $N/4$ staff-weeks of effort for N-week sprint



Journey to Agile to Agile-Lean

Scrum is the market and mindshare leader

- Simple framework is extensible and customizable.
- 10% projects following Scrum, and 30% following one or more agile methods – Forrester Survey, Jan 2010.



Lean, Agile-Lean: Amdocs, BBC Worldwide, CitySearch, Microsoft, Corbis, Facebook, NBC Universal, Premier Healthcare



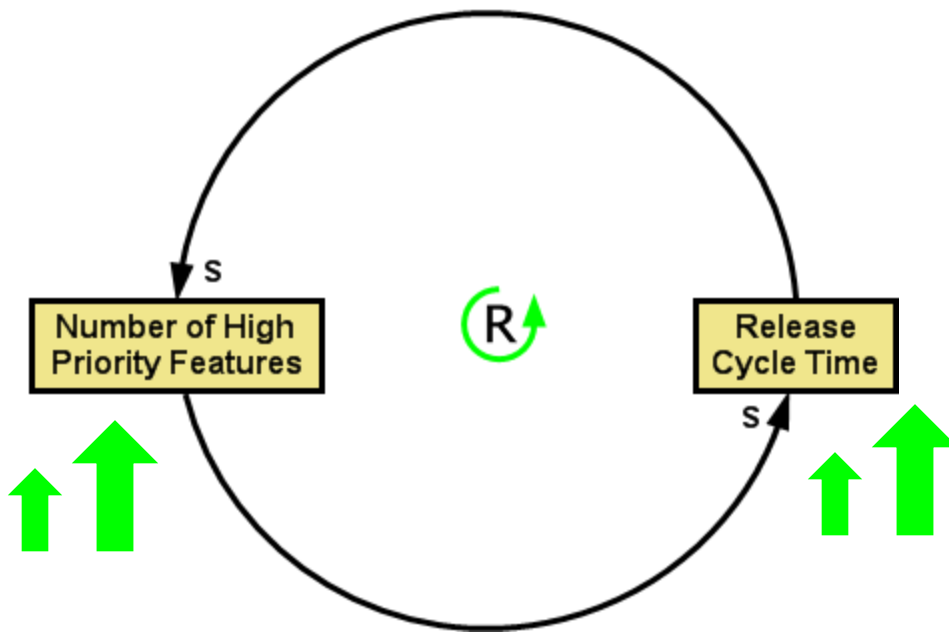
Introduction to Systems Thinking

- **System:** interacting components or subsystems forming an integrated whole.
- **Purpose or function:** Usually deduced by observing system behavior, involving inputs (information, material or energy), processing, and outputs (information or material or energy).
- **Components:** People, principles and values, processes and workflows
- **Interactions:** Functional and structural relationships among components allowing them to interact
- **Structure:** Defined by components and their interconnections
- A **model** is used to represent a system, capturing only the key aspects
 - **Causal Loop Diagrams (CLDs):** Great for qualitative understanding and reaching consensus; they cannot be simulated
 - **Stock-and-Flow Networks (SFNs):** Can be simulated for quantitative analysis and *what-if* experiments, but require more effort to build and validate

Note: We are *not* talking about only software system to be developed, but systems in general...specifically organizational systems.



CLD for Relationship between Release Cycle Time and Number of High Priority Features



R: Reinforcing (positive) feedback loop
B: Balancing (negative) feedback loop

Notation

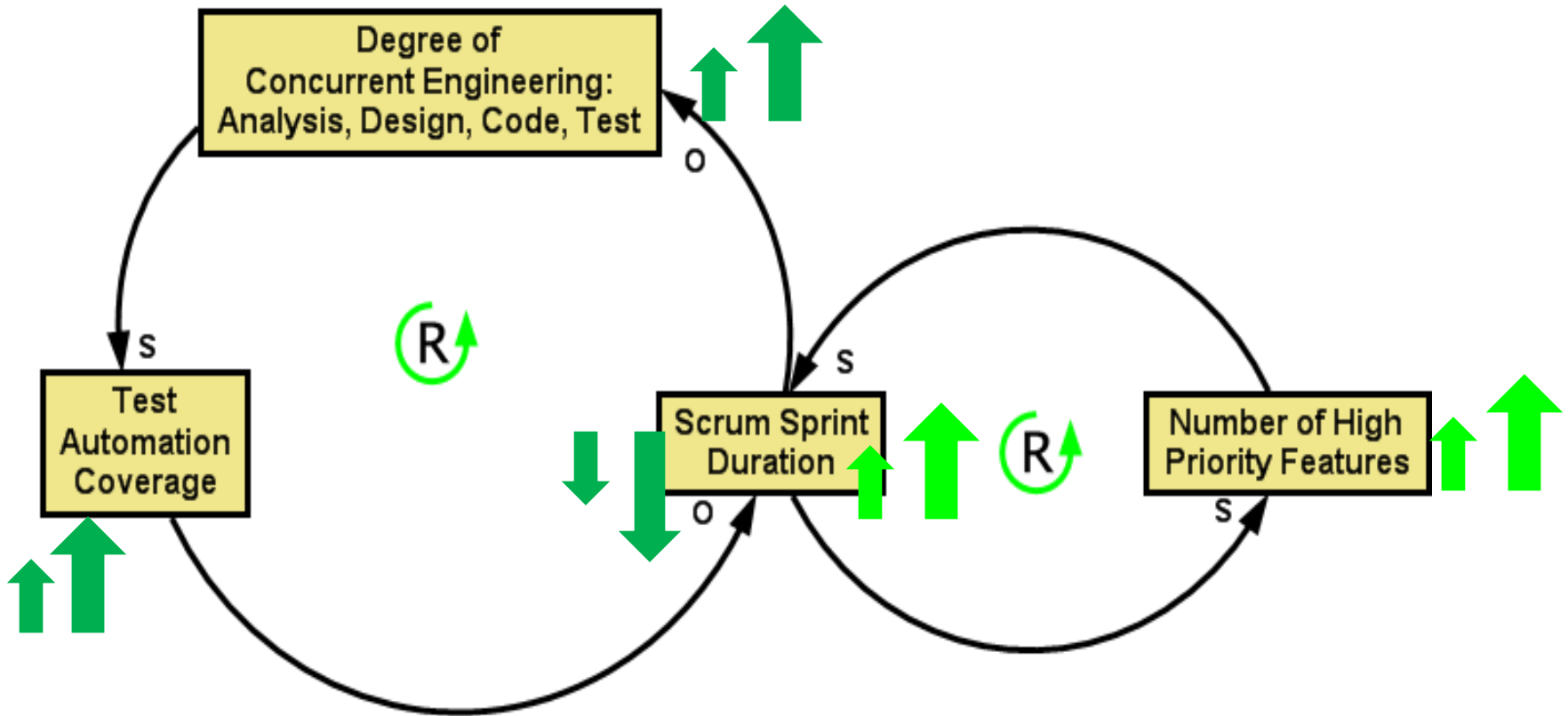
Cause and **Effect** variables are shown in rectangular boxes, with links showing causal relationships

- **S** (Supports): If the cause **increases**, the effect **increases** *above what it would otherwise have been*, and if the cause **decreases**, the effect **decreases** *below what it would otherwise have been*.

- **O** (Opposes): If the cause **increases**, the effect **decreases** *below what it would otherwise have been*, and if the cause **decreases**, the effect **increases** *above what it would otherwise have been*.

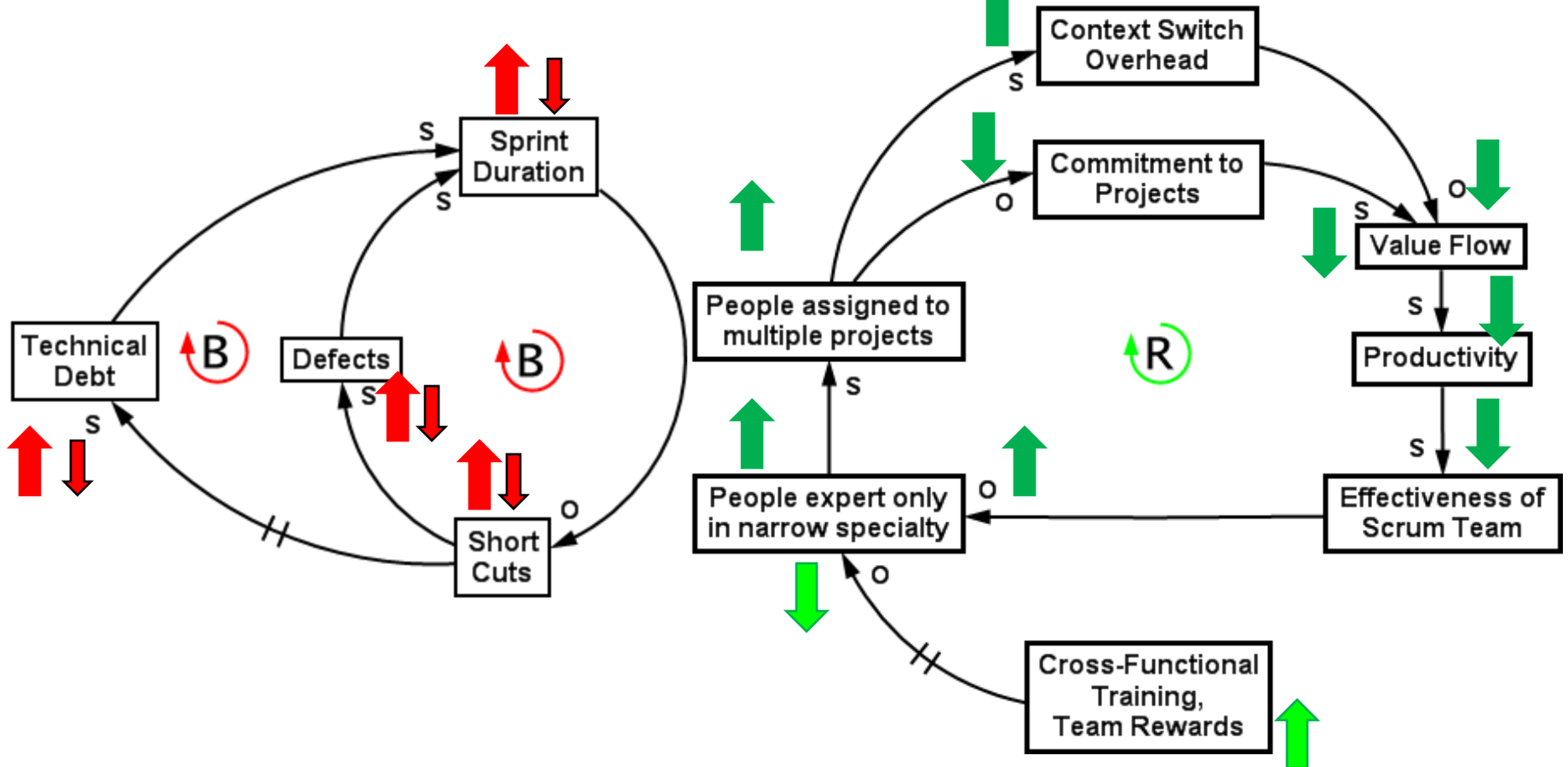


CLD for Relationship among Sprint Duration, Test Automation, Number of High Priority Features





CLDs: Short-cuts and Sprint Duration, Scrum Team Effectiveness





New Synergy Group

Systems Thinking... Enterprise Performance

Systems Thinking Perspective on Two-Week Sprints

Sprint Planning and Analysis Workshop	RR	2-week Sprint (End-to-End)	DD	Sprint Retrospective and Review
1-day		8-days of Sprint work		1-day
Actual Engineering Work	48%	Next Sprint Backlog Grooming (12.5%), Daily Scrum (5%), Contingency (7.5%), Context-Switching and Non-Availability (15%) = Total (40%)	52%	"Overhead"

Likely lower quality and effectiveness of Sprint retrospectives and reviews, if done every two weeks!

Sprint Planning and Analysis Workshop	RR	5-week Sprint (End-to-End)	DD	Sprint Retrospective and Review; and Implement Action Plan
3-day		20-days of Sprint work		2-day
Actual Engineering Work	56%	Next Sprint Backlog Grooming (5%), Daily Scrum and Weekly Plan (7.5%), Contingency (7.5%), Context-Switching and Non-Availability (10%) = Total (30%)	44%	"Overhead"

Systems thinking provides economic basis and larger context.

Sprint Planning and Analysis Workshop	RR	7-week Sprint (End-to-End)	DD	Sprint Retrospective and Review; and Implement Action Plan
3-day		30-days of Sprint work		2-day
Actual Engineering Work	62%	Next Sprint Backlog Grooming (5%), Daily Scrum and Weekly Plan (7.5%), Contingency (7.5%), Context-Switching and Non-Availability (7.5%) = Total (27.5%)	38%	"Overhead"



Synergies among Agile/Scrum, Lean & Systems Thinking

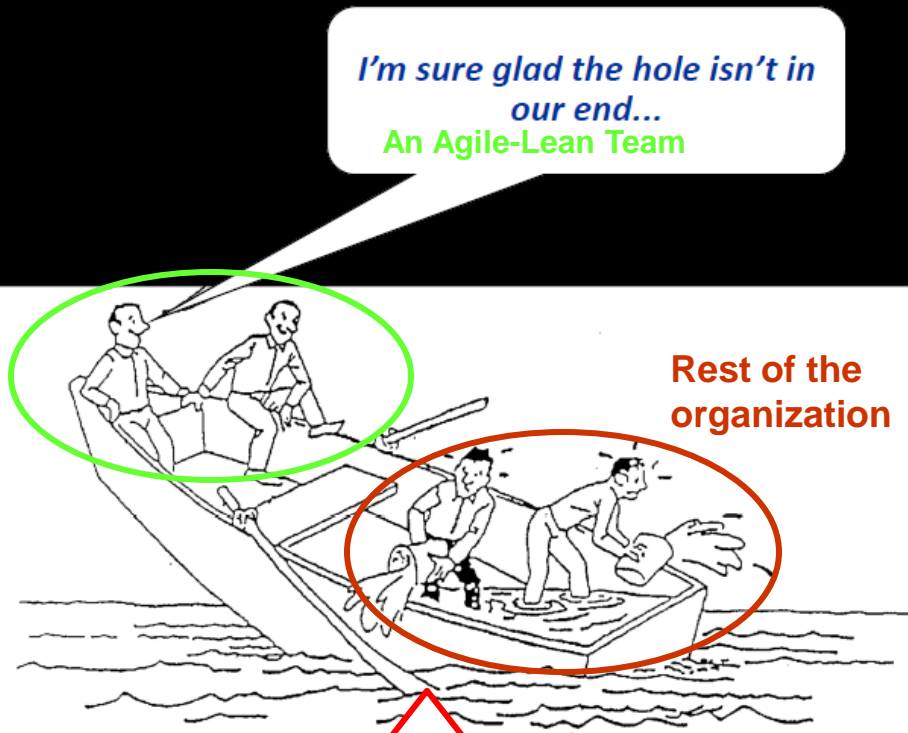
	Agile/Scrum	Lean, Systems Thinking
Time-boxing	Yes	No → Great for maintenance work, IT operations
Reduce end-to-end feature cycle time	No	Yes; Level features to reduce avg. cycle time; <i>Stop Starting, Start Finishing</i> mantra
Value Flow	No	Yes
Visibility	Burn charts, Impediment log	Cumulative flow, Visible queues, Kanban board
Optimization and scalability	Automation, Refactoring, Scrum of Scrum, Meta Scrum	System of systems (Systems thinking); Optimize the <i>whole system</i> (not parts)
Control mechanism	Daily Scrums, Sprint Review, Sprint Retrospective	WIP limits, Queue reduction
Validation and Verification	Acceptance test as part of feature specification, Test-driven development	Reduce the total cost of (Defect prevention + Defect detection and correction); Find the root cause and fix it (<i>5-Why's</i> may not always work)
Overall	Somewhat revolutionary framework for process change	More evolutionary framework with which to extend your current processes



New Synergy Group

Systems Thinking... Enterprise Performance

Other Major Application Areas of Systems Thinking to Agile-Lean Development and Beyond



- Use standardized practices and instrumented templates across agile teams
- Make agile-lean methods work well with all organizational processes
- Change the *organizational system to change culture*
 - *A bad system, will defeat a good person, every time. – Edward Deming*
- Strategy and operational processes must be well-aligned

9/28/2011

© 2011 New Synergy Group

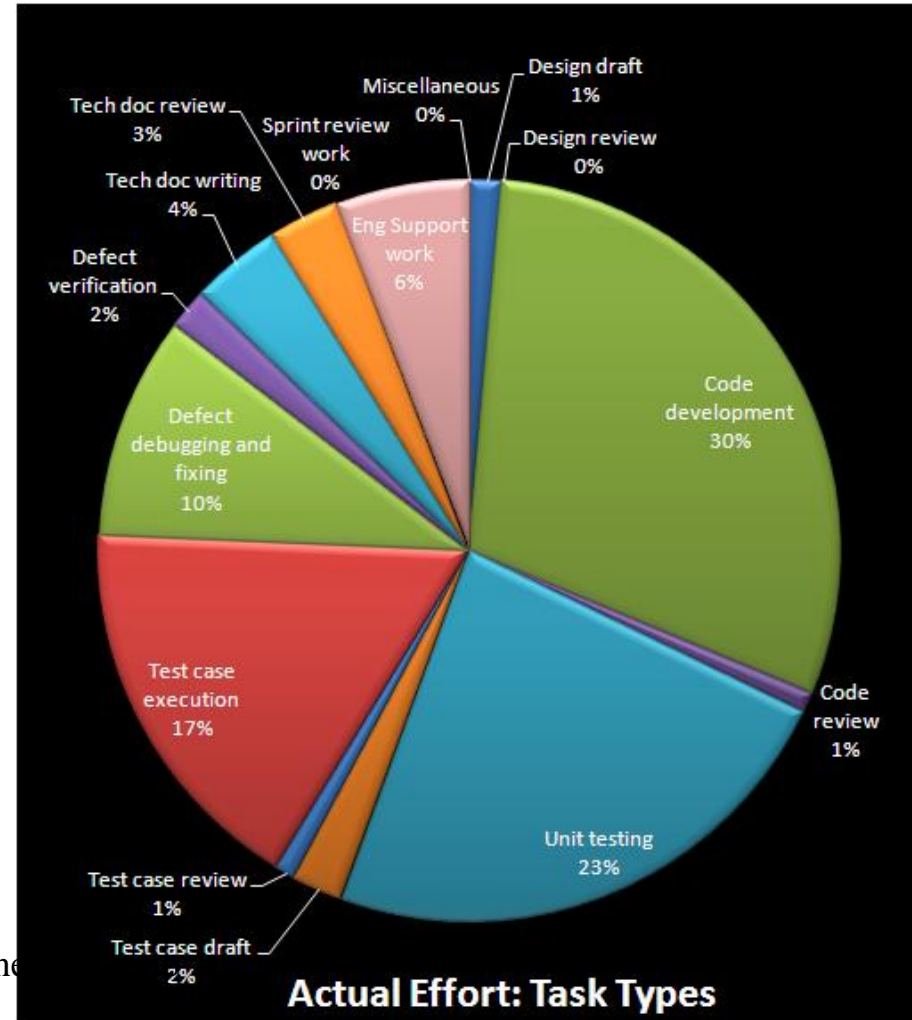
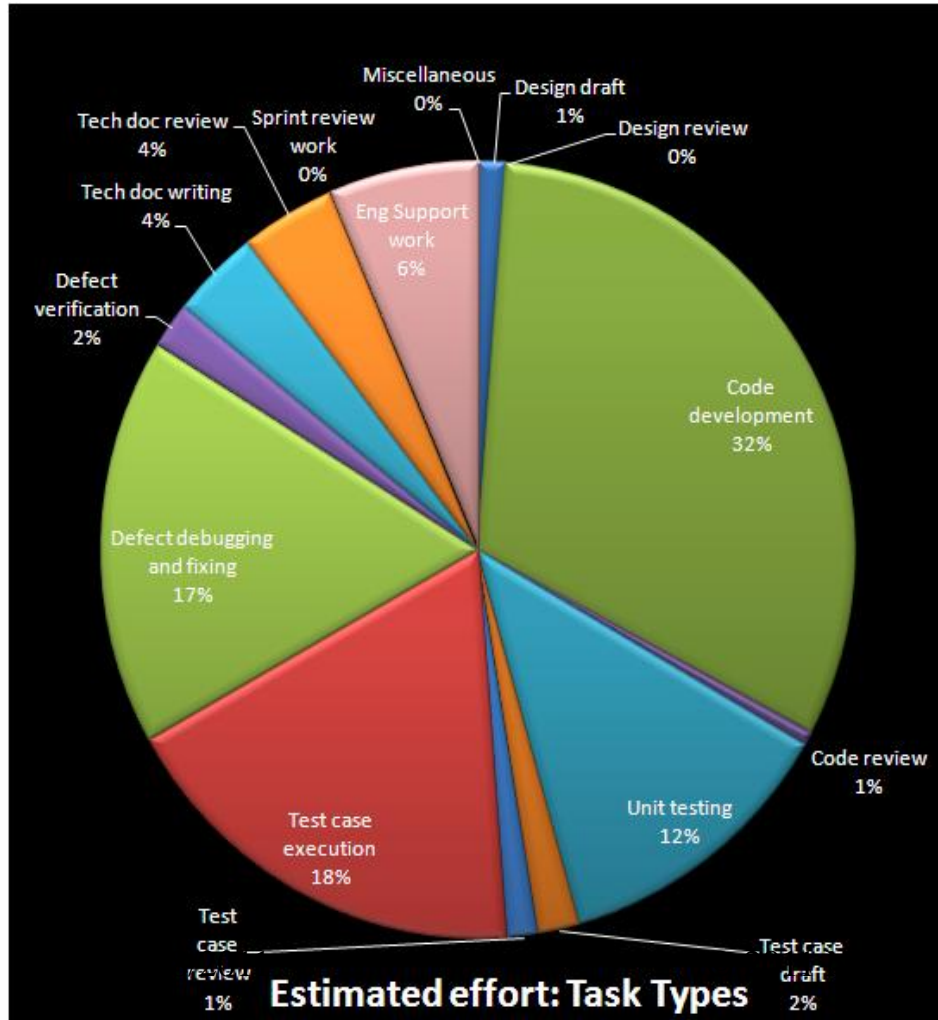
31



New Synergy Group

Systems Thinking... Enterprise Performance

Instrumented Template for Sprint Effort





Wisdom from Systems Thinking

- **Faster is slower; slower is faster: Rush job → defects, delays**
 - Work in sustainable steady sprints...avoid death march
- **There are cause → effect → cause feedback loops with possible delays...avoid causation fallacy**
 - Every effect has a cause... but don't assume you can tell which is which!
 - *5-Why's* method for root cause analysis may not always work
- **Cause and effect are often far removed in space and time**
- **Avoid short-termism**
 - Short-term pain vs. long-term gain | Short-term gain vs. long-term pain
 - Today's problems often come from yesterday's "solutions"
- **Optimize the whole system, not sub-optimize its parts**
 - Optimize the total cost of (Anticipatory design + Adaptive design)
 - Optimize the total cost of (Defect prevention + defect correction)
 - Take a broader view of all organizational systems and their interactions

Reference: "Thinking in Systems: A Primer" by Donella Meadows



Business Strategy and Operations Must be Aligned

Business strategy: What is the right thing to do?

Operations: How to do things right?

Mission, Values

BHAGs

Strategic Themes

Strategic Objectives

Strategic Initiatives

Product Lifecycle Operations

All Other Operations

Big Hairy Audacious Goals

A common pattern that appears across many elements of strategy

Strategic maps show cause-effect links among strategic objectives and/or initiatives

Agile-Lean methods

Lean methods, with Scrum-style process improvements





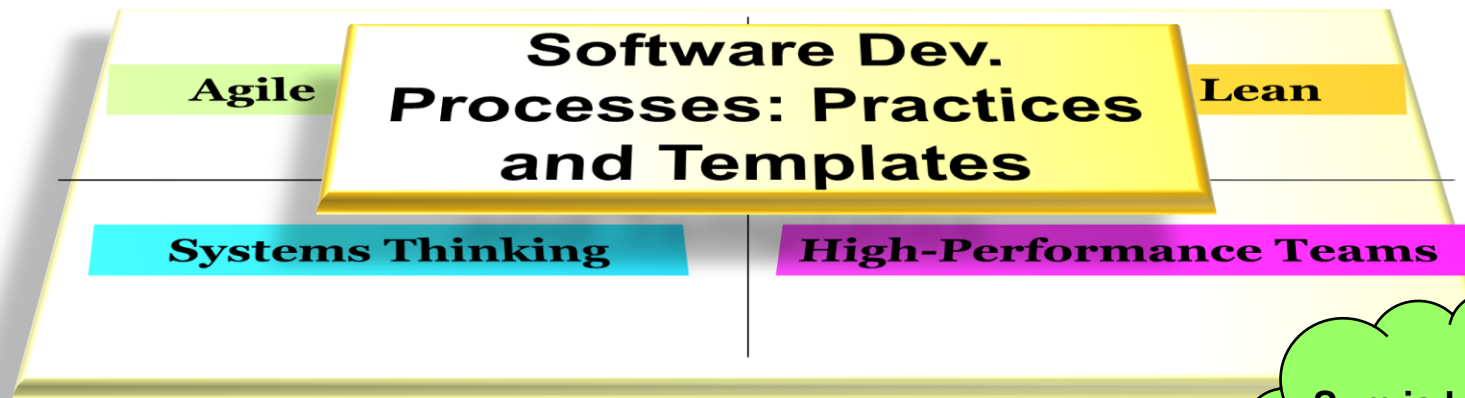
New Synergy Group

Systems Thinking... Enterprise Performance

Software Development Processes Turbo-Charged with: Agile and Lean Frameworks, Systems Thinking, High-Performance Teams

Focus on Synergistic Patterns, Not on Wars of Dogmas and Commandments
Methodologies is *not* the end goal; Pleasing Customers and Growing Business is!

Align with upstream strategy and downstream operations



Sum is bigger
than its parts

smthatte@NewSynergyGroup.com

www.NewSynergyGroup.com



New Synergy Group

Systems Thinking... Enterprise Performance

Agile, Scrum and Lean Methods and Practices

A two-Day Course offered by Agile University

Dr. Satish Thatte, CEO New Synergy Group

October 13-14, 2011, Washington, DC

- **Course fee:** covers course material hard copy binder, agile templates (Microsoft Excel and Google Docs)
- **16 PDU Credits:** All attendees will receive a certificate of completion which can be used to claim 16 PDUs.
- **For details and course registration by October 5th 2011**
http://www.agileu.org/course_details.jsp?courseid=606&schid=1564
- **\$100 fee discount for Philly Spin members or attendees of this seminar given on 9/27/11 at Philly Spin meeting.**