



How Semat Can Change The Future of Software Engineering

Paul E. McMahon

Report Documentation Page				Form Approved OMB No. 0704-0188	
Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.					
1. REPORT DATE MAY 2011		2. REPORT TYPE		3. DATES COVERED 00-00-2011 to 00-00-2011	
4. TITLE AND SUBTITLE How Semat Can Change The Future of Software Engineering				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) PEM Systems,118 Matthews Street,Binghamton,NY,13905				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited					
13. SUPPLEMENTARY NOTES Presented at the 23rd Systems and Software Technology Conference (SSTC), 16-19 May 2011, Salt Lake City, UT. Sponsored in part by the USAF. U.S. Government or Federal Rights License					
14. ABSTRACT					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT Same as Report (SAR)	18. NUMBER OF PAGES 25	19a. NAME OF RESPONSIBLE PERSON
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified			

Introduction



Semat founders
Bertrand Meyer, Richard Soley,
Ivar Jacobson

- ♦ **SEMAT stands for Software Engineering Method and Theory**
- ✓ **Includes industry, academia, and research**
- ✓ ***“refound software engineering based on a solid theory, proven principles and best practices”***

Topics

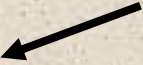
- ✓ **Background and Goals**
- ✓ **The Challenge & Big Picture Solution**
- ✓ **Semat Architecture & Way Work Accomplished**
- ✓ **What We Have Achieved So Far**
- ✓ **A Little More Detail**
 - **A Few Terms**
 - **The Universals**
- ✓ **Why Semat is Important to Future and Potential Significant Savings it Can Bring**

From the Semat Web Site (www.semat.org)

- ♦ *Software engineering is gravely hampered today by immature practices. Specific problems include:*
 - ✓ *The prevalence of fads more typical of fashion industry than of an engineering discipline.*
 - ✓ *The lack of a sound, widely accepted theoretical basis.*
 - ✓ *The huge number of methods and method variants, with differences little understood and artificially magnified.*
 - ✓ *The lack of credible experimental evaluation and validation.*
 - ✓ *The split between industry practice and academic research.*

Also curriculum ↗ 4

Background

- ✓ July, 2009 call for action
 - ✓ February, 2010 Semat Vision Statement published
 - ✓ February, 2011
 - 9 Corporate Sponsors
 - Over 1400 Supporters
 - 35 individual signatories
 - Over 20 volunteers have contributed significant time
 - Multiple tracks met periodically in 2010
 - Three workshops held in 2010
- Note: Watts Humphrey was an initial signatory* 

Goal

Note: Quite small!



- ✓ A “kernel of widely agreed elements”

Note: Referred to as “Universals”



- ✓ *This common ground will allow people to easily describe the essentials of their current and future practices and methods so that they can be*

*composed, simulated, compared,
evaluated, used, measured, taught and
researched, adapted*

The Challenge: Today's Process Aids Landscape



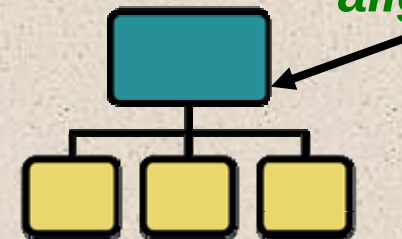
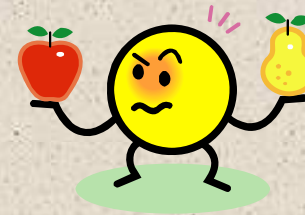
Just how different are all these process aids when it comes to helping people get their job done?

Large have Process Engineers, What about small & medium size organizations? 7

Practitioners

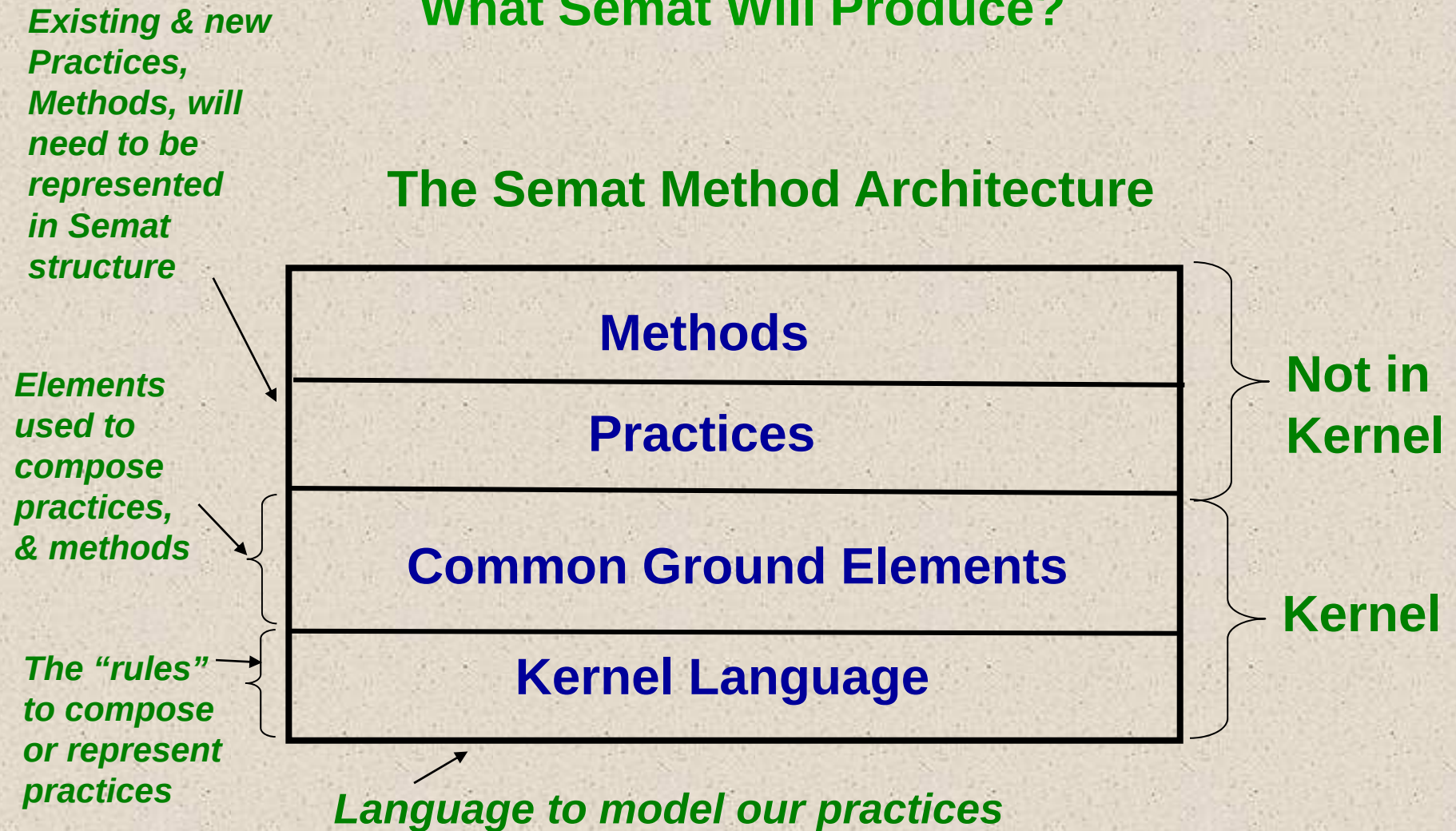
Objectives of Semat

- ◆ Help users compare practices and make better decisions
- ◆ But more...also about using... practices what do, not just what say we do
 - ◆ As team's understanding evolves it's practices must adapt – but must occur with appropriate degree of monitoring
- ◆ Can think of Semat as reference framework– but not kind you might think
 - ◆ Essentials underlie best practices, not best practices



Big Picture Solution: What Semat Will Produce?

The Semat Method Architecture



The Primary Product of Semat is The Kernel

The Way The Work Accomplished So Far: Multiple Tracks

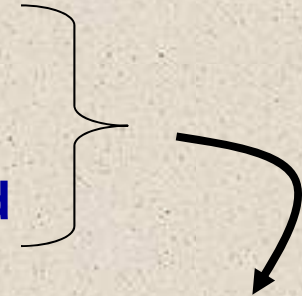
- ✓ Requirements (active early 2010) ← *Develop Use Cases (e.g. Define Practice)*
- ✓ Universals (currently active)
 - Develop the Universals
- ✓ Kernel Language (currently active) ← *“Well-formed” rules to form practices, methods*
- ✓ Assessment (active throughout 2010) ← *“Certify” practices*
 - Help users self-assess use of Semat
- ✓ Theory (planned future activity) ← *Theories supporting Universals & states*
- ✓ Architecture (new track October 2010)
 - Coordinates activities of other tracks through Architecture “Spikes”

What We Have Achieved So Far: 2010 Workshops Held

◆ Zurich, Switzerland, March 17-18



- ✓ Position papers presented by 28 attendees
- ✓ Track leaders identified & track work initiated



*Watts Humphrey
planned to attend*

◆ Washington, D.C. . July 13-14

- ✓ Track progress reports provided by track leaders to 21 attendees



◆ Milan, Italy, Sept 30-Oct 1



- ✓ Track progress reports provided by track leaders
- ✓ 1st Architecture spike report provided

What We Have Achieved So Far: Products

- ✓ A set of use cases (e.g. Define Practice) driving the work on the kernel language
- ✓ An initial set of Universals (8 kernel elements)
- ✓ A draft kernel language specification
- ✓ A draft assessment framework
- ✓ An initial architecture spike

Example From 1st Architecture Spike

- ♦ Applied “Define Practice” Use Case to Scrum
- ♦ Results reported at 3rd Semat Workshop in Milan

- ✓ Demonstrated

- ✓ One way to “represent” Scrum in Semat Structure

- ✓ Potential to aid in comparing & aiding decisions

- ✓ “Holes” and “clashes” visibility can help

Note: Results might be obvious to expert Process Engineer, but intent to help Practitioner

Example how it can help later- actual case study

A Little More Detail: A Few Terms (still evolving)

- ♦ Practice

- ✓ A repeatable approach to doing something with a specific purpose in mind

Note: We spend a great deal of time seeking widely acceptable terms

- ♦ Way of working

- ✓ The tailored set of *practices* and tools used by a *team* to guide and support their *work*



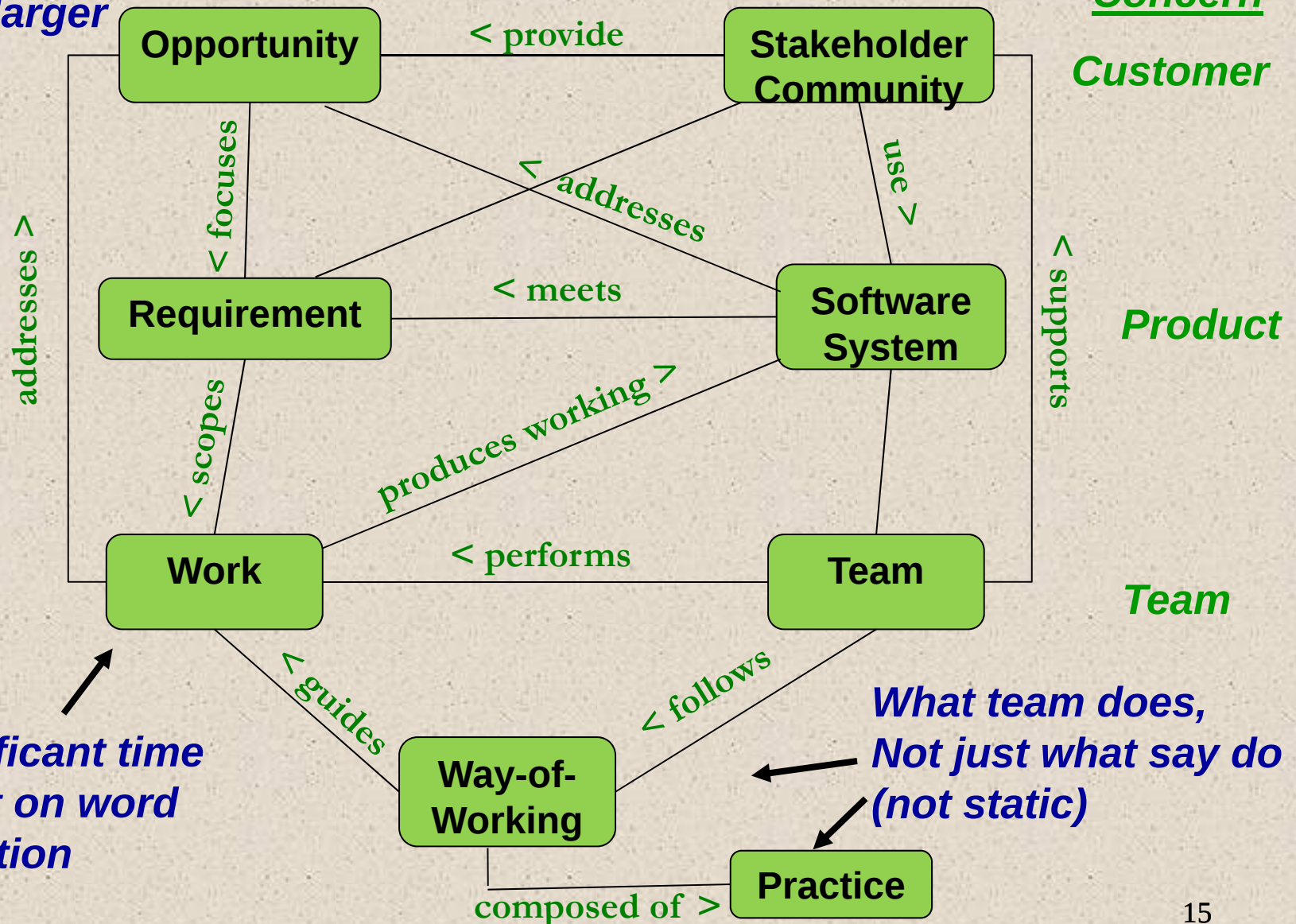
Universals (or Kernel elements)

- ✓ The essential elements in all software engineering endeavors

*Don't expect
model to get
much larger
(risk?)*

Universals (Kernel elements): So far...

Area of
Concern
Customer



Current Universals Challenge: Measuring Health & Progress

♦ Example: Work

✓ Can we agree on a set of Universal states and definitions?

*Potential
great
value to
Industry
(more
later)*

- Initiated
 - Work has been requested
- Prepared
 - All pre-conditions for starting the work have been met
- Underway...
- Under control...
- Concluded...
- Closed...

*Checklists support
definition using small
consistent vocabulary*



*Critical to
potential value*



Keep in mind essential states
can be extended, but not
deleted or redefined

Why Important to Academia?

- ♦ Today different universities have different requirements related to how software engineering should be taught and what should be taught
- ♦ To academia a kernel based on a common ground means a consistent foundation to teach software engineering and to demonstrate to students the pros and cons of different ways of working



Engaging the Universities is a priority of Semat

Why Important to Practitioner?



Learning curve expensive for all of us & frustrating to practitioner

Why Important to Industry?

- ◆ Consider the significant expenditures that occur today on each new project start up

- ◆ Terminology reinvention
- ◆ How progress measured and reported
- ◆ Process tailoring



Not saying some of this isn't needed, but...

- ◆ Are the differences across different projects and different companies large enough to warrant the degree of continual reinvention?
- ◆ We believe what we need to agree on is very small, but the potential savings is significant in project start up costs, improved communication & training!

Next: Measurement Example

Why Important to Industry: Measurement Example

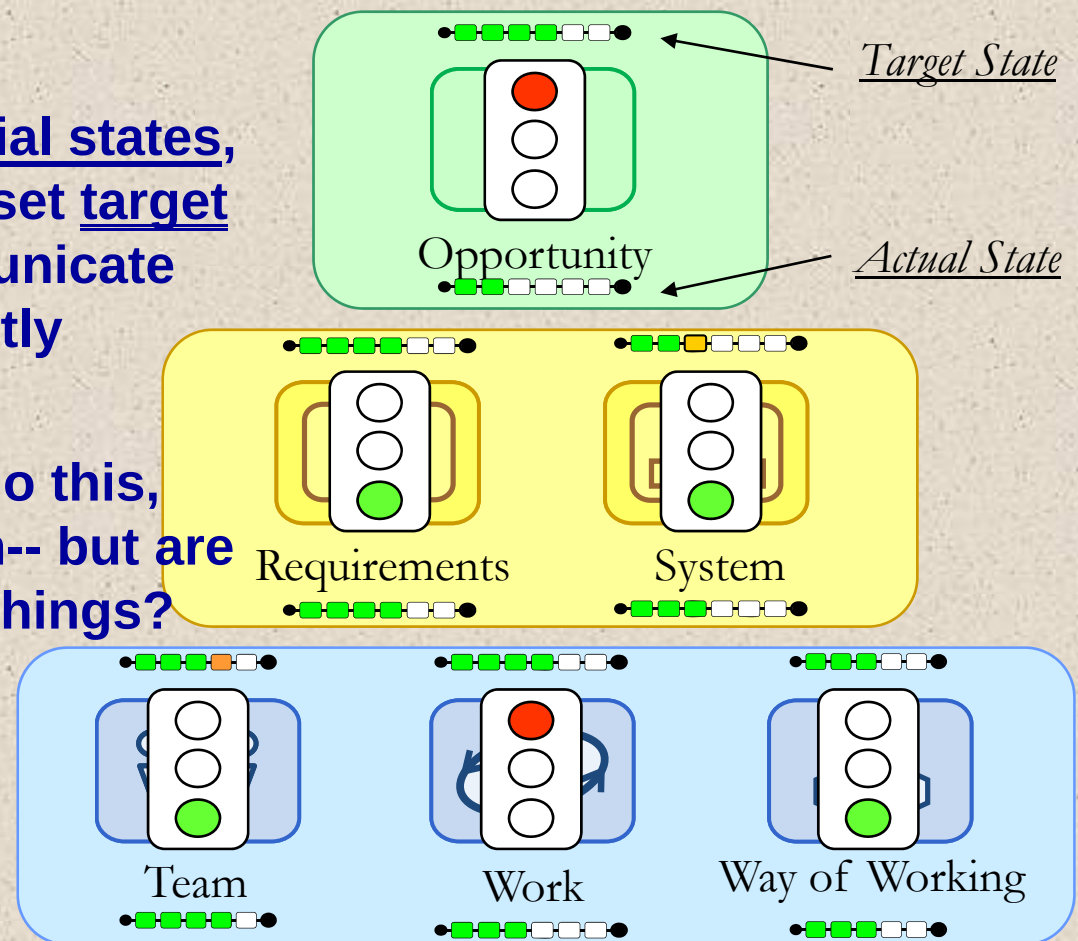
- ◆ Think of the potential value of a “common ground” starting point for measurement

Improving progress measurement has huge cost savings potential

- ◆ If we can agree to essential states, & their meaning, we can set target states & assess & communicate progress more consistently

- ◆ Today great companies do this, but each grows their own-- but are they looking at the right things?

- ◆ Example later



Why Semat Goals Are Achievable: Common Ground Same as Tailor Up From “Must dos”

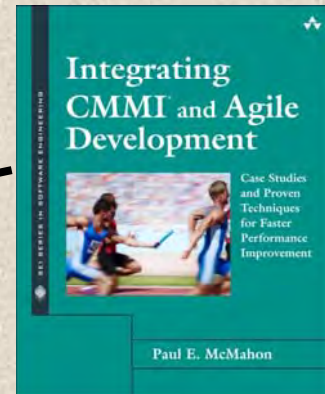
- ♦ Tailor up from “must dos”

- ✓ Proven approach

*What all agree “must do”
regardless of factors*

- ✓ Refer to BOND and GEAR case studies in [1]

Saves significant project start up and training costs because don't need to discuss & reinvent on each project!



*Small project may
need little more, but*

Keep aware:

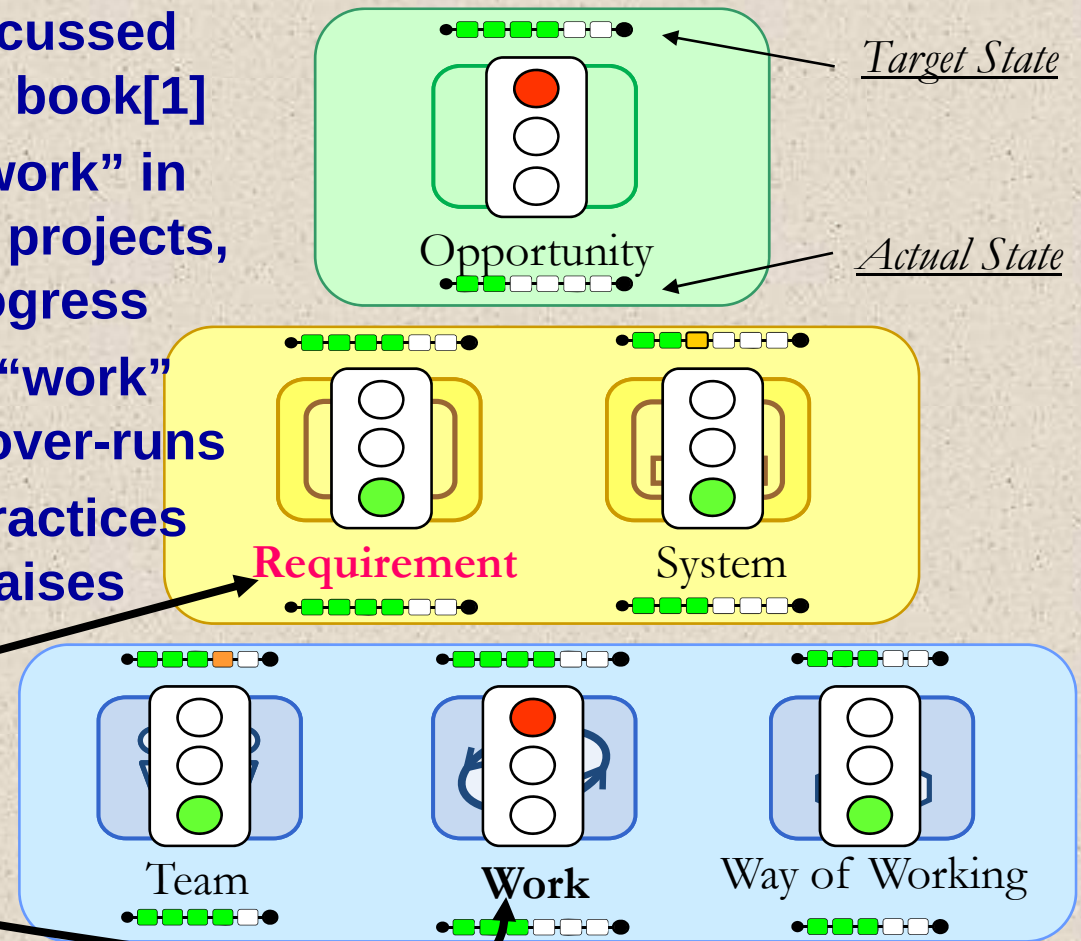
*Depending on project specific factors,
you extend, or “tailor up” focusing your
start up effort on value-added areas*

*[1] Integrating CMMI and Agile Development: Case Studies
And Proven Techniques for Faster Performance Improvement*

Actual Case Study Example: Measurement

- ♦ LACM is Case Study discussed in previously referenced book[1]
 - ♦ Defined states for “work” in common way across projects, and tracked work progress
 - ♦ Frequent unplanned “work” leading to schedule over-runs
 - ♦ Representing their practices using Semat kernel raises visibility of a “hole”

Tracked work, **not** requirements



How Semat kernel can raise visibility of why organizations have trouble meeting schedules

Summary:

What Is Really Different About Semat?

- ♦ Semat will not produce a methodology, nor will it compete with any previous movements
 - ✓ We don't believe past movements, or existing aids (e.g. CMMI) were failures (value in each)
- ♦ In fact we believe underneath the constant change across last 4 decades lies “universals” that do not change
 - ✓ Essentials that remain when one movement loses steam, and another rises – “common ground”
- ♦ If can agree on terminology, language, how to describe practices & methods, then we will have made a significant contribution that can be counted on for a long time

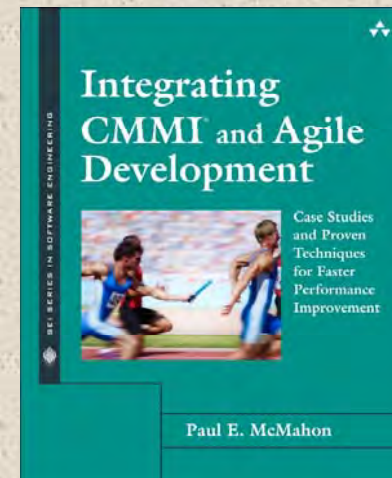


Contact Information & Questions



- ◆ Email: pemcmahon@acm.org
- ◆ WEB: www.pemsystems.com
- ◆ Blog: www.paulemcmahon.wordpress.com

Questions???



- ◆ Do you have a “common ground” in your organization? Is it working for you?
- ◆ Do you believe a “common ground” can be found for all software engineering endeavors?

Acronyms & References



Acronyms

CMMI – Capability Maturity Model Integration

Semat – Software Engineering Method and Theory

SPEM – Software & Systems Process Engineering Meta-model

TSP – Team Software Process

Scrum– Not an acronym, mechanisms in game of rugby for getting an out-of-play ball back into play

References & Notes

Parts of this presentation have been developed based on previous Semat presentations given at:

- Rochester Institute of Technology
“How Semat Can Change the Future of Software Engineering”, Paul McMahon, October 2010**
- SEPG Conference Portland, Oregon 2011
“The Quest for the Holy Grail of Software Engineering”
Winifred Menezes, Paul McMahon**