

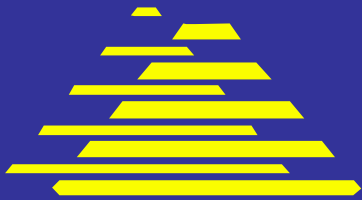
Agile, TSPSM, CMMI® pick one, pick two, pick all three!

Daniel M. Roy

Cape Town SPIN

20 February, 2008

PSP, TSP, Personal Software Process and Team Software Process are service marks of CMU
CMM is and Capability Maturity Model are registered in the U.S. patent and trademark office



Agenda

When I was doing real work

What's an agile "method"

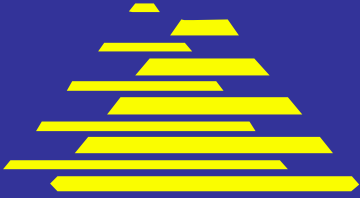
Agile and CMMI®

PSP/TSPSM : the agile CMMI®

The experience factory

PSP/TSPSM results in practice

Synergy PSP/TSPSM/CMMI®



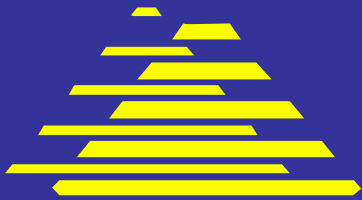
What's agile?

A super-class, featuring sets of:

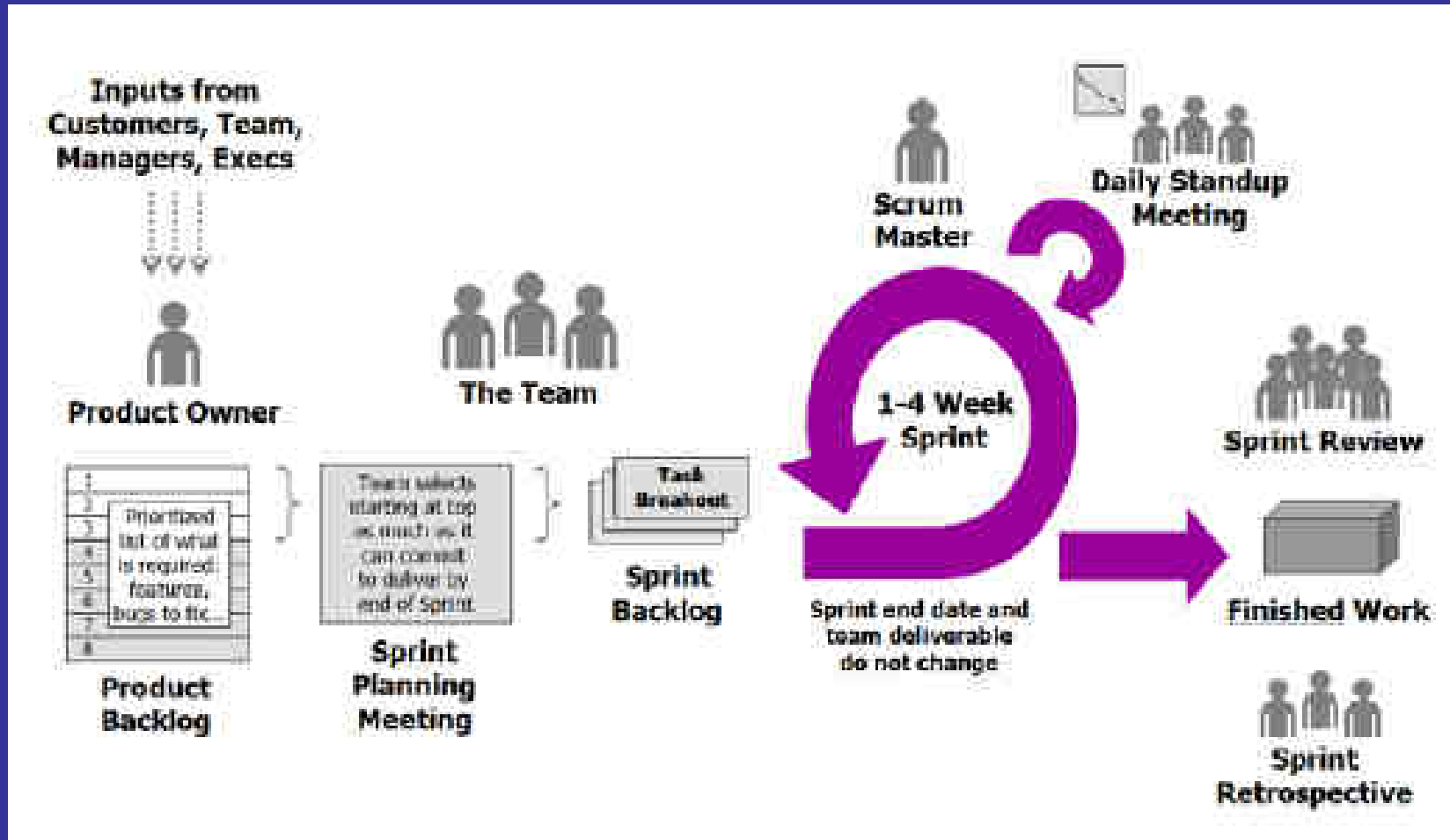
- Methods and “methodologies”
- Iterative life cycle models
- Programming practices
- Software development philosophies

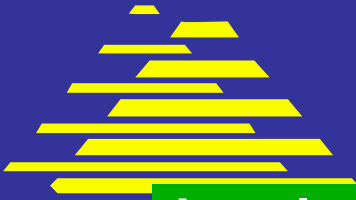
This super-class is called “Agile software development”

The underlying principles behind agile development are listed in “The manifesto”



Scrum

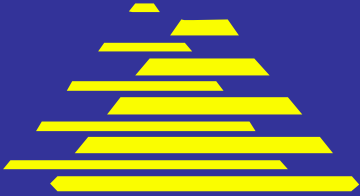




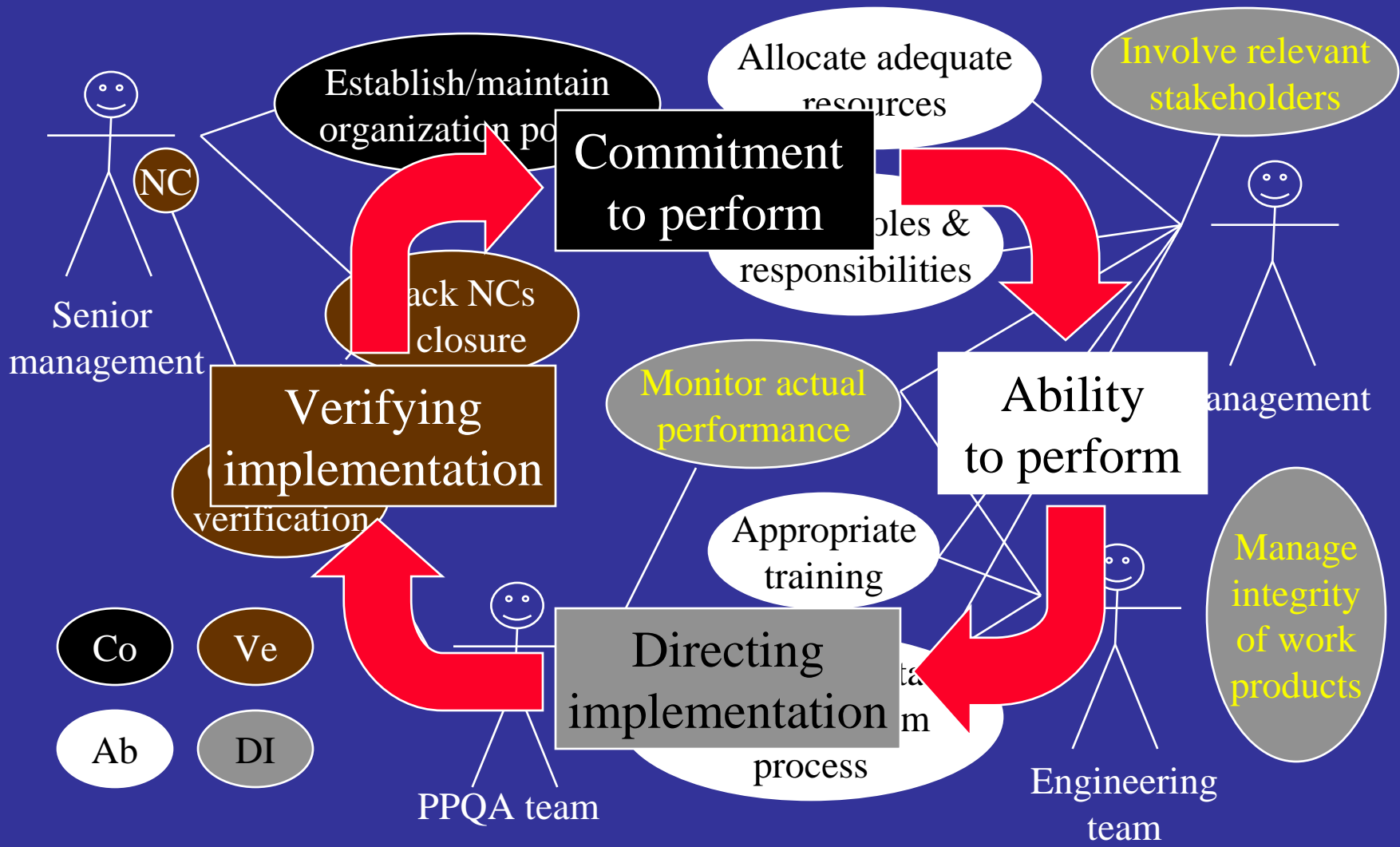
Scrum and CMMI-Dev

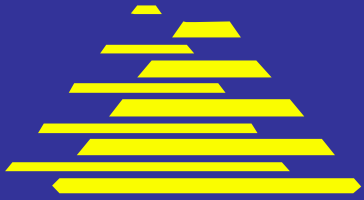
5

| Level | Focus | Process Areas |
|----------------------|--|---|
| 5 Optimizing | Continuous process improvement | Organizational Innovation and Deployment Causal Analysis and Resolution |
| 4 Quantitat. Managed | Quantitative management | Organizational Process Performance Quantitative Project Management |
| 3 Defined | Organization engineering process standardization | Requirements Development Technical Solution Verification Validation Organization Process Focus Organization Process Definition Organizational Training Integrated Project Management Risk management Decision Analysis and Resolution Product Integration |
| 2 Managed | Basic Project management | Requirements management Project Planning Project Monitoring and Control Supplier Agreement Management Measurement and Analysis Process and Product Quality Assurance Configuration Management |



Scrum and I18N





Is CMMI the agile nemesis?

"Optimizing processes that are agile and innovative depends on the participation of an empowered workforce aligned with the business values and objectives of the organization."

CMMI Dev. V1.2 August 2006
GG5 in Generic goals and practices



Building High-Performance Teams

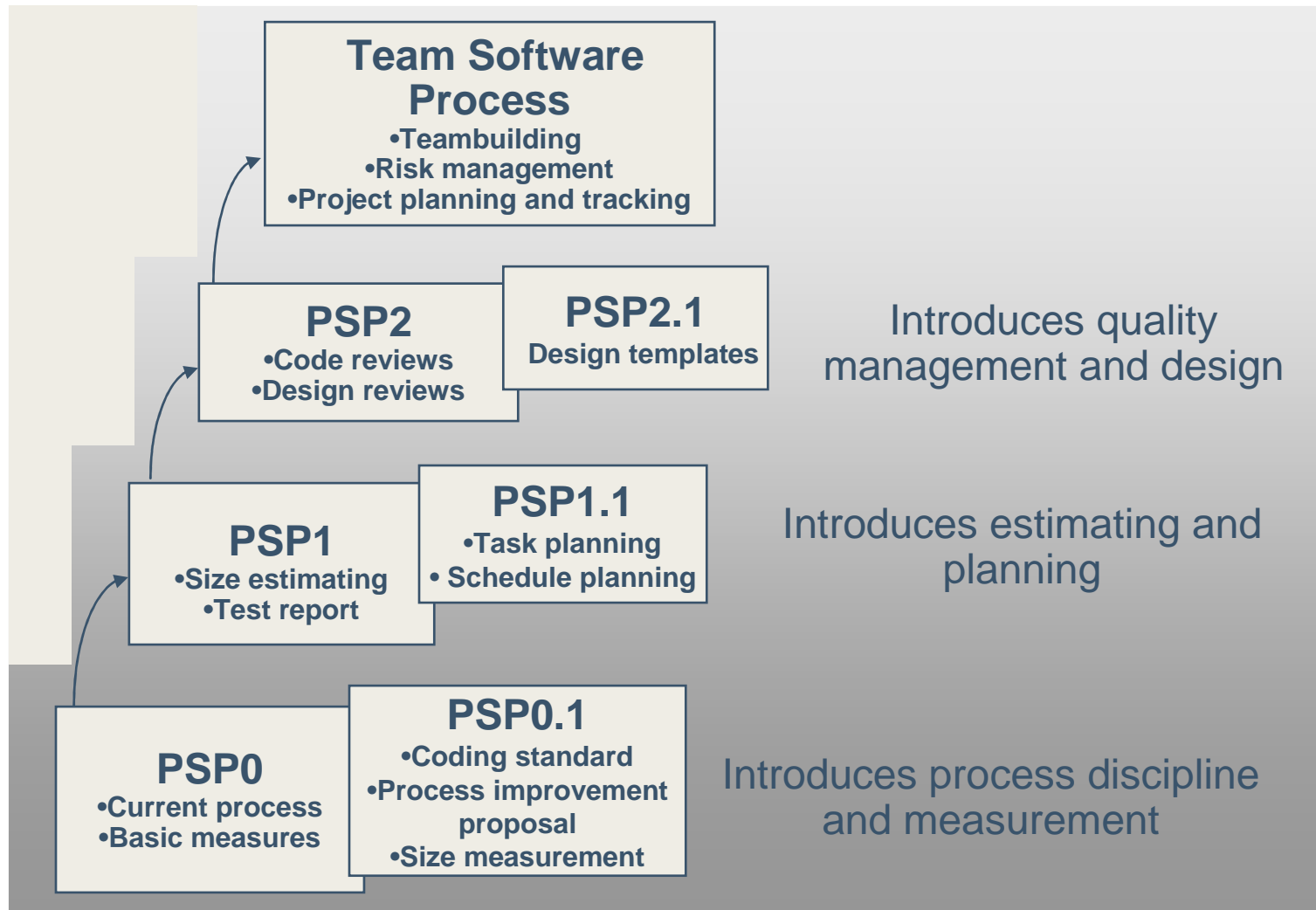
The TSP strategy is to improve performance from the bottom up.

This strategy starts with PSP training.





The PSP Course



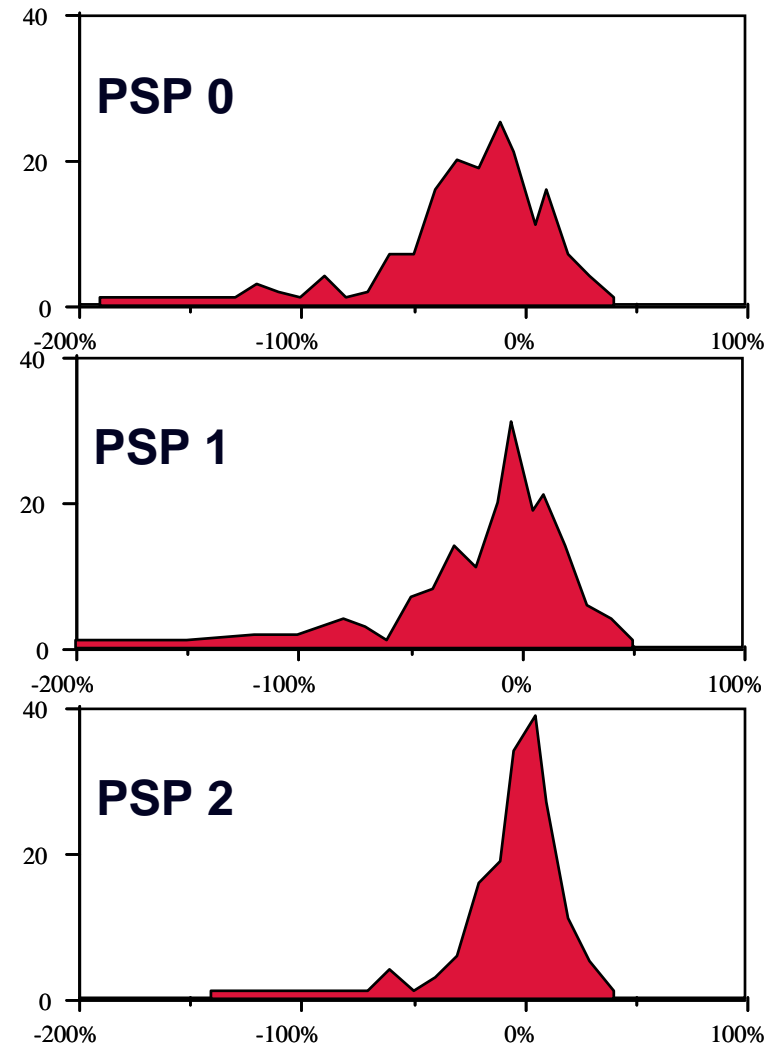


PSP Estimating Accuracy

Majority are under-estimating

Balance of over- and underestimates

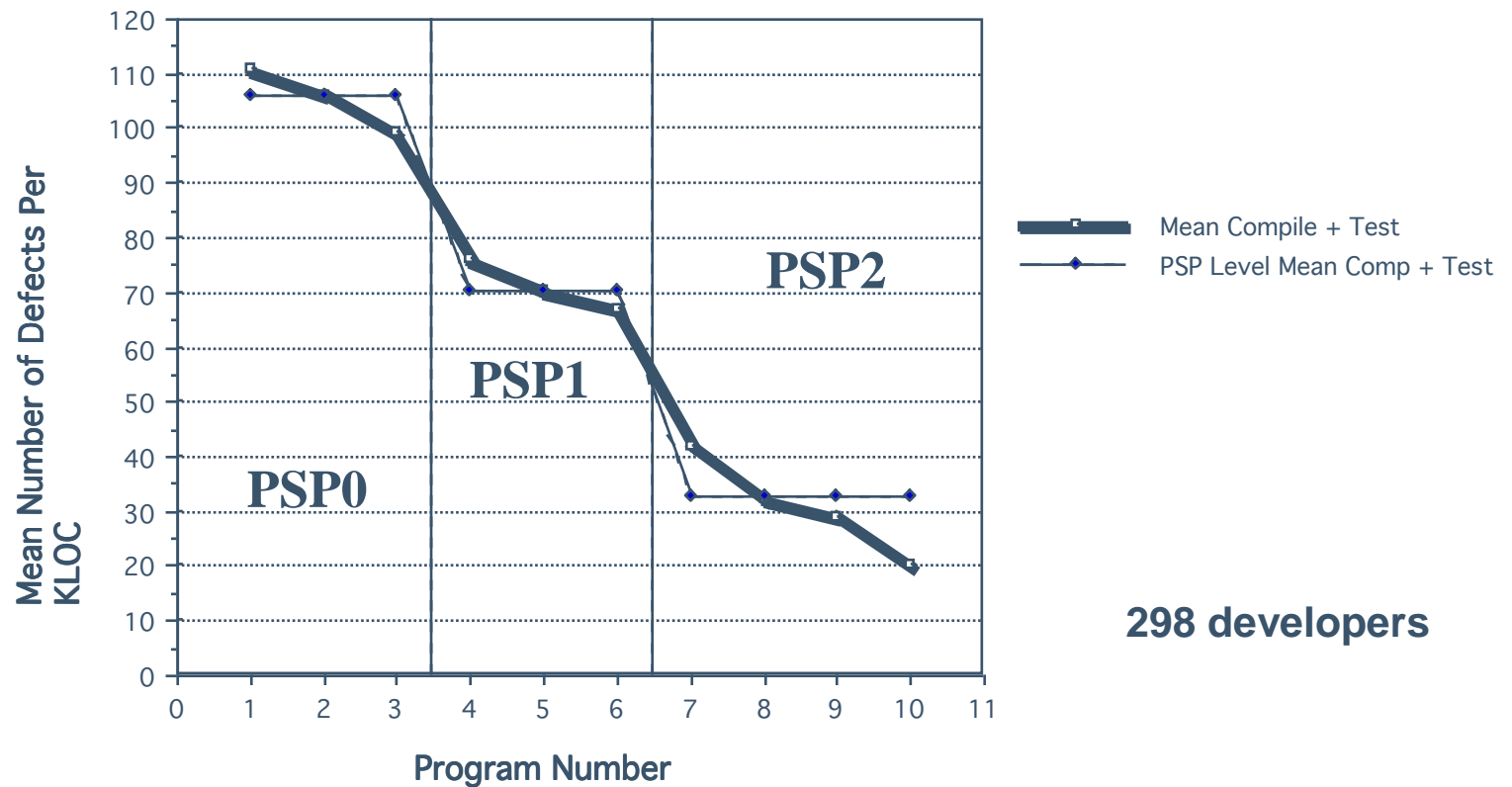
Much tighter balance around zero





PSP Quality Results

Defects Per KLOC Removed in Compile and Test



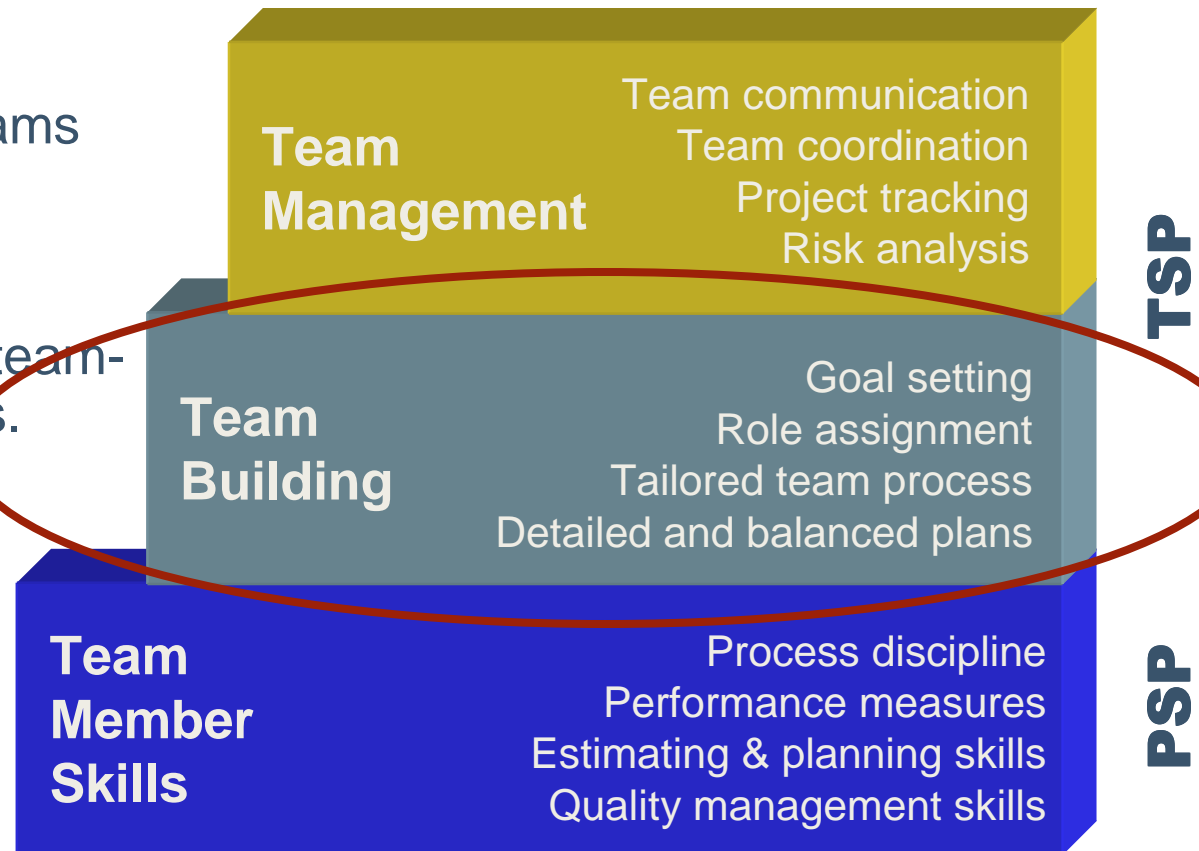
298 developers

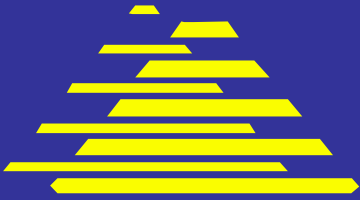


Building Self-directed Teams

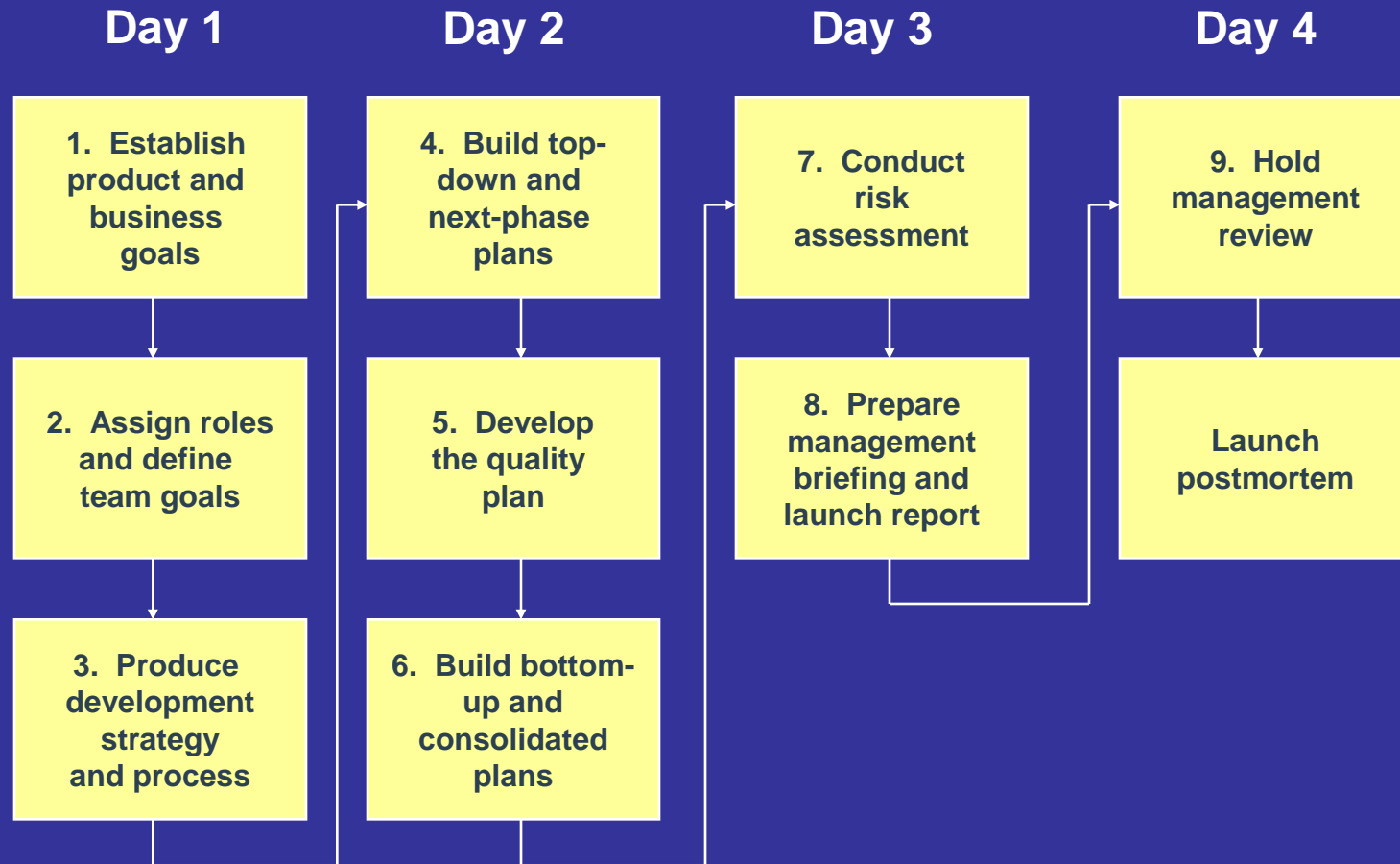
Self-directed teams
must be built.

This requires a team-
building process.





TSP launch process

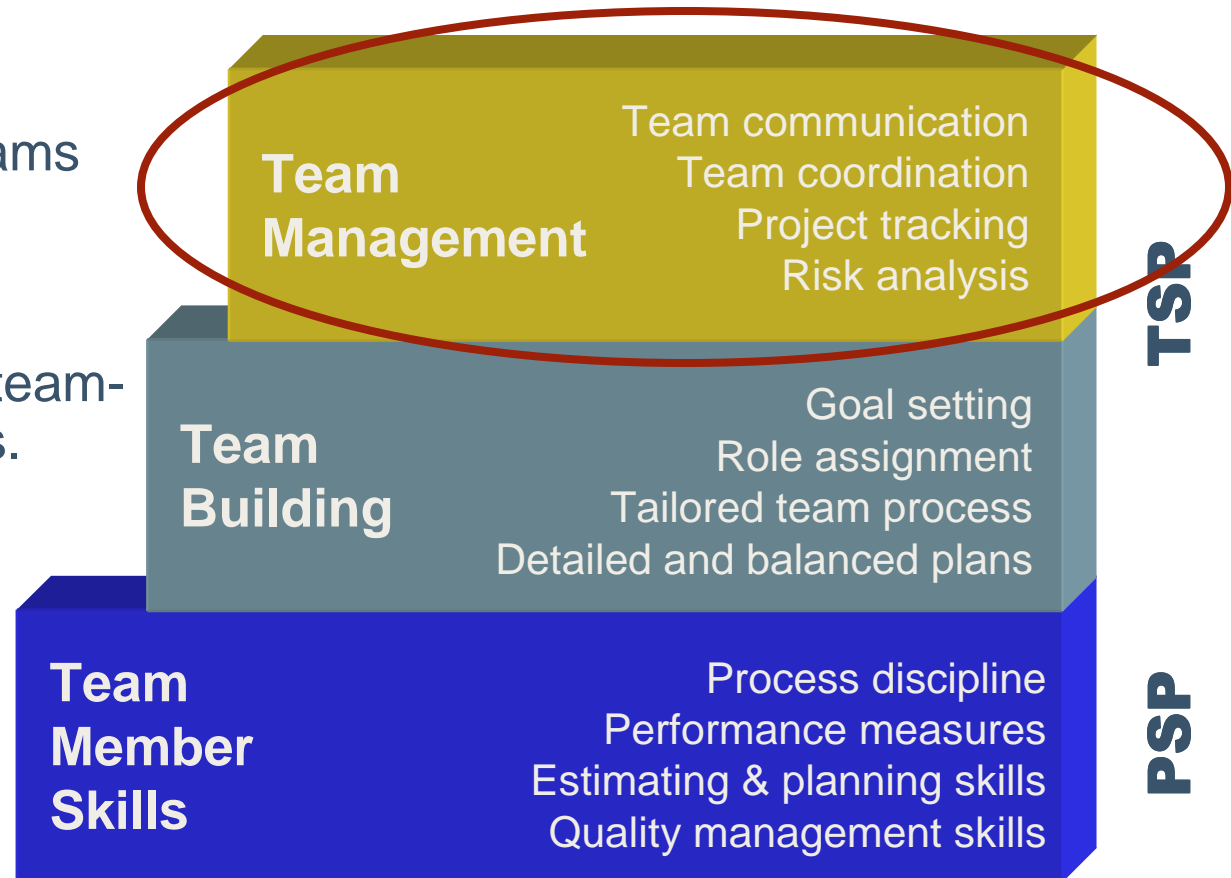


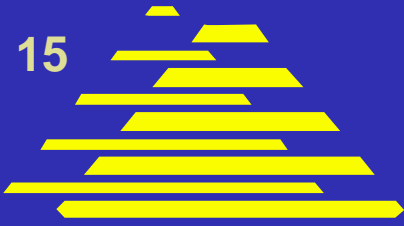


Managing Self-directed Teams

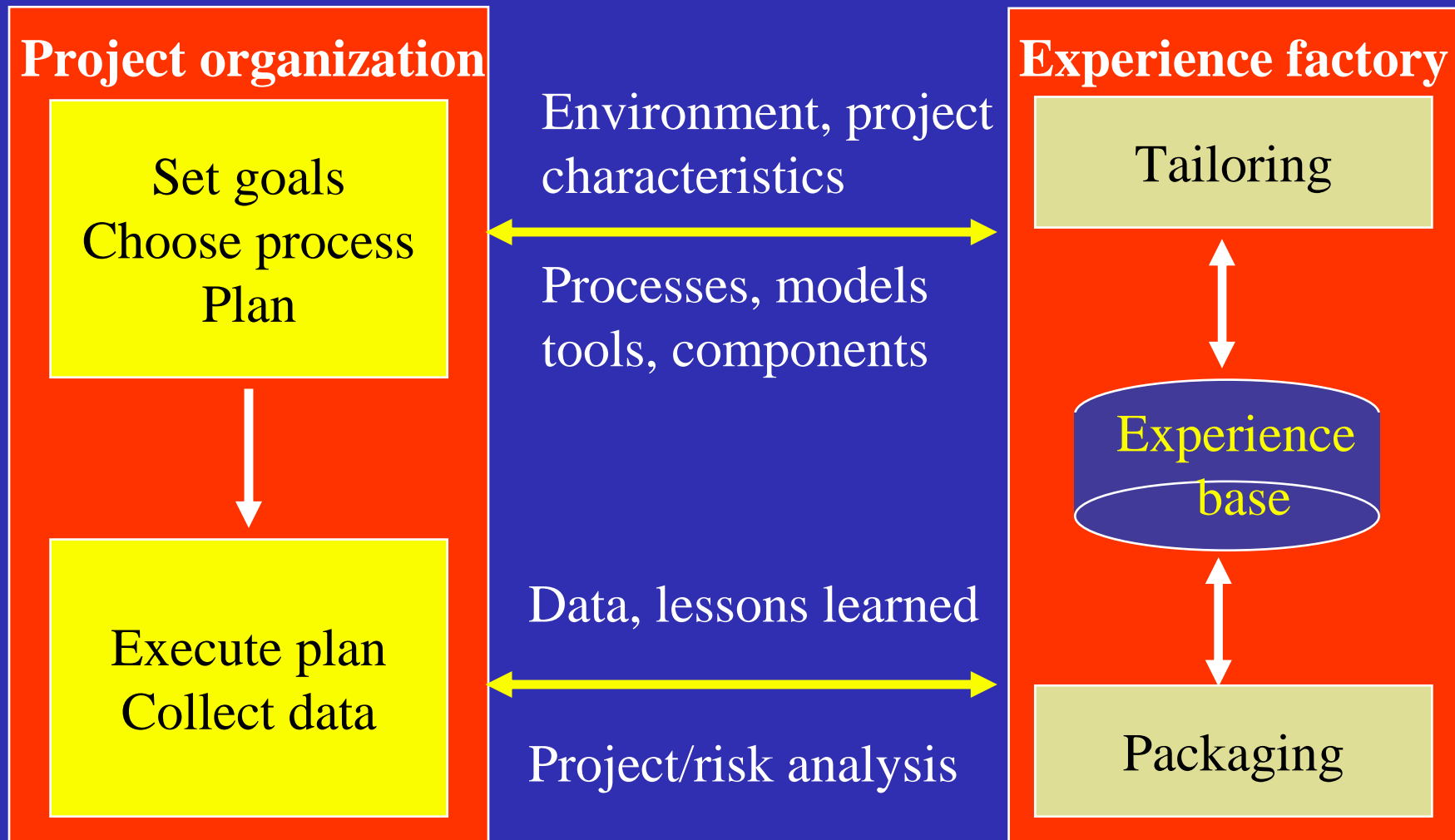
Self-directed teams must be built.

This requires a team-building process.

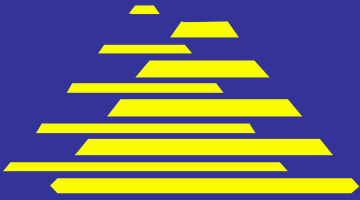




The experience factory

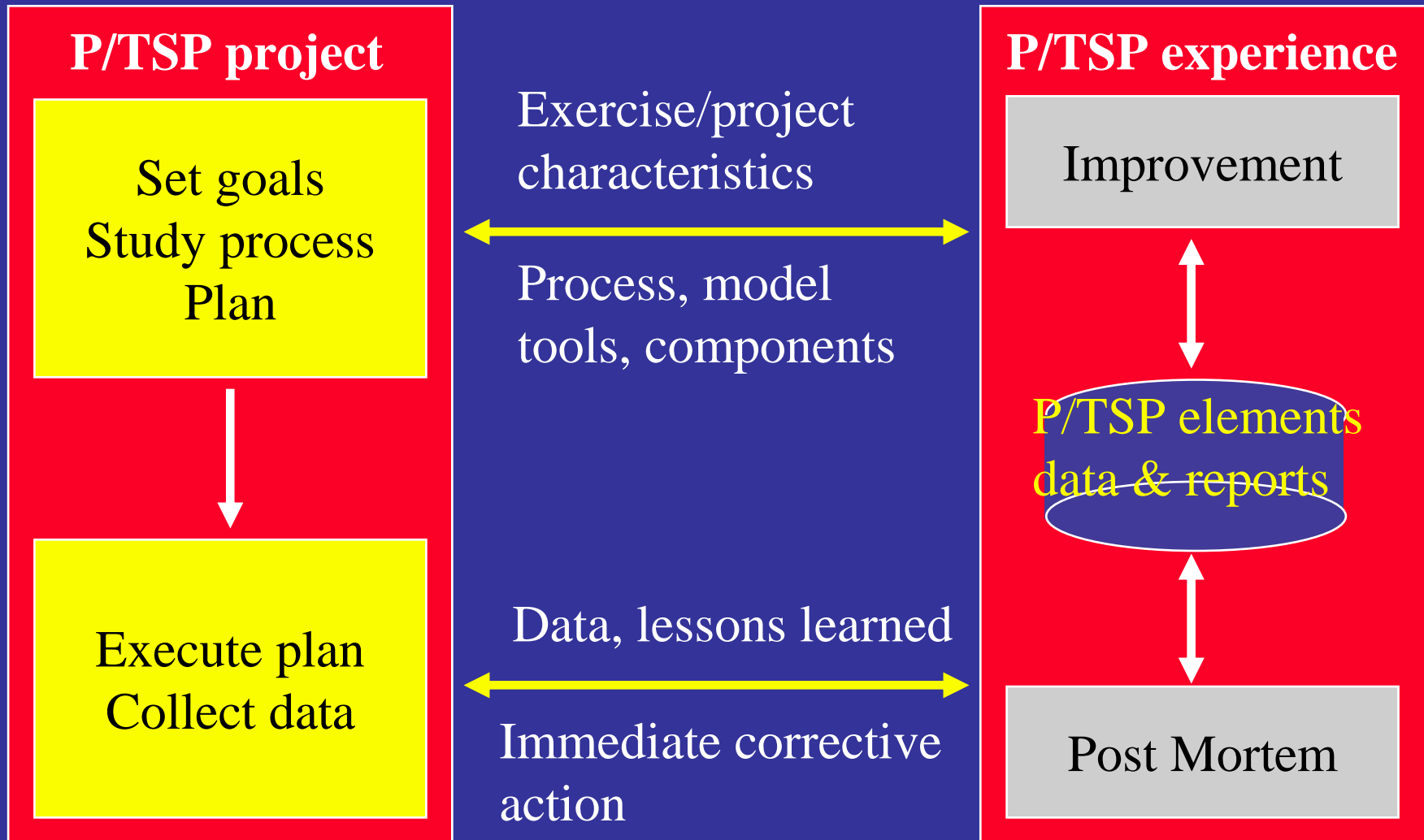


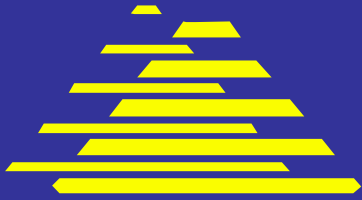
From 'The experimental paradigm in software eng.',
Rombach, Basili, Selby, Springer-Verlag, 1994



PSP/TSP: Agile Exp. Fact.

16

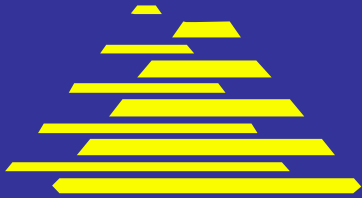




TSP and Scrum Principles

TSP projects are team-directed. Team members

- plan their work
- regularly track and report their performance
- own their own processes
- are all involved in the planning and decision making
- have defined and explicit roles



Tracking the TSP project

TSP Week Summary - Form WEEK

| | |
|-----------------|-------------------------|
| Name | Consolidated Team Plan |
| Team | Security System Upgrade |
| Status for Week | 5 |
| Week Date | 1/31/2000 |

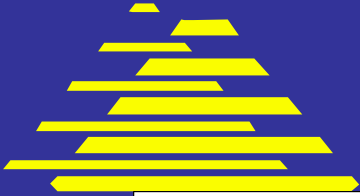
| | |
|-------|----------|
| Date | 2/7/2000 |
| Cycle | |

Weekly Data

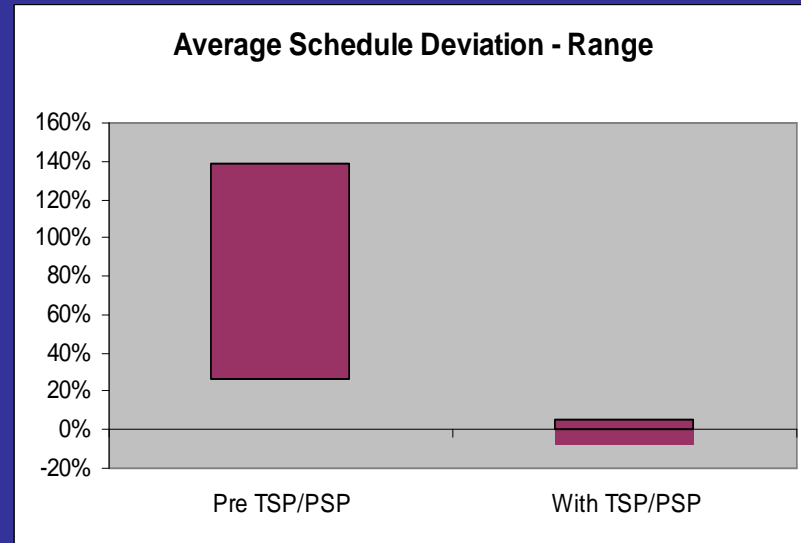
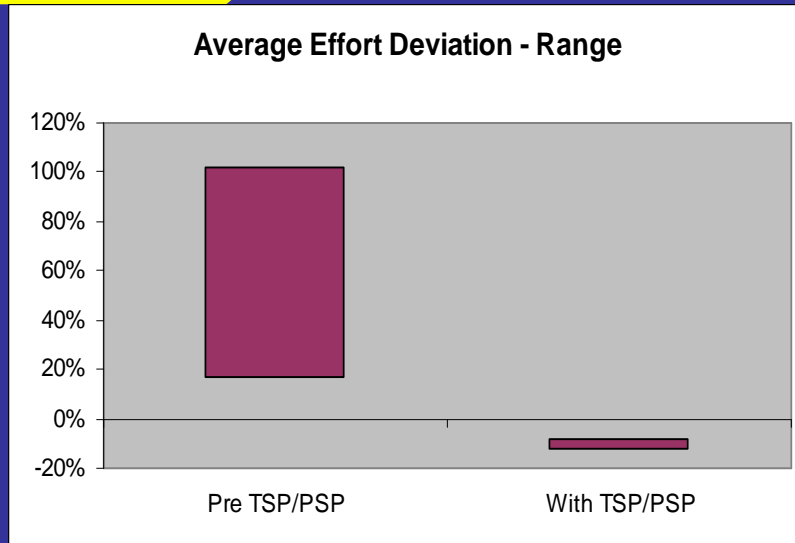
| | Plan | Actual | Plan/ Actual |
|-----------------------------------|-------|--------|-----------------|
| Project hours for this week | 80.0 | 69.0 | 1.16 |
| Project hours this cycle to date | 400.0 | 344.8 | 1.16 |
| Earned value for this week | 10.3 | 3.1 | 3.37 |
| Earned value this cycle to date | 40.2 | 30.0 | 1.34 |
| To-date hours for tasks completed | 293.0 | 303.8 | 0.96 |

| Assembly | Phase | Tasks Completed | Resource | Plan Hours | Actual Hours | Earned Value | Planned Week | Plan Hrs./ Actual Hrs. |
|----------|-------|--|----------|------------|--------------|--------------|--------------|---------------------------|
| SYSTEM | REQ | Write SRS general sections | tmc | 14.0 | 12.0 | 1.4 | 4 | 1.17 |
| SYSTEM | REQ | Weekly requirements analysis meeting 5 | tma | 4.0 | 4.0 | 0.4 | 5 | 1.00 |
| SYSTEM | REQ | Weekly requirements analysis meeting 5 | tmb | 4.0 | 4.0 | 0.4 | 5 | 1.00 |
| SYSTEM | REQ | Weekly requirements analysis meeting 5 | tmc | 4.0 | 4.0 | 0.4 | 5 | 1.00 |
| SYSTEM | REQ | Weekly requirements analysis meeting 5 | tmd | 4.0 | 4.0 | 0.4 | 5 | 1.00 |
| | | TASKS DUE THROUGH WEEK 7 | | | | | | |
| SYSTEM | REQ | Review SRS general sections | tmc | 5.0 | | 0.0 | 4 | |
| SYSTEM | STP | Complete Validation Test Plan | tmd | 8.0 | 8.5 | 0.0 | 4 | 0.94 |

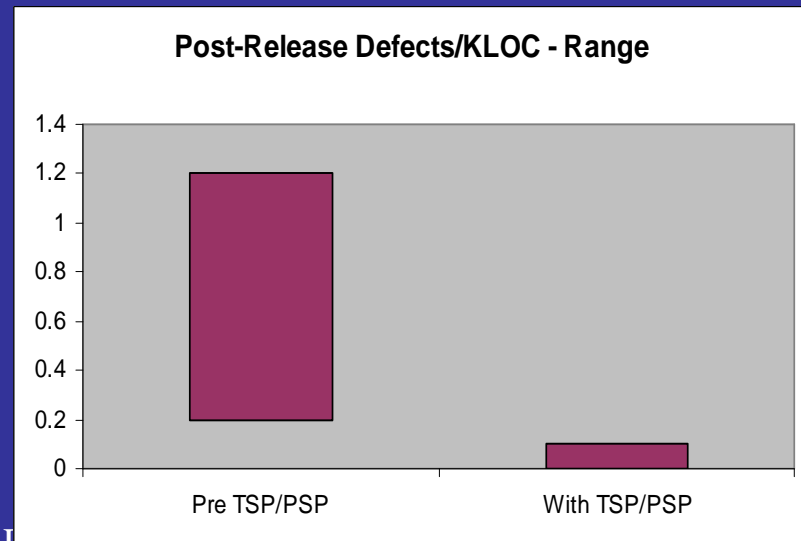
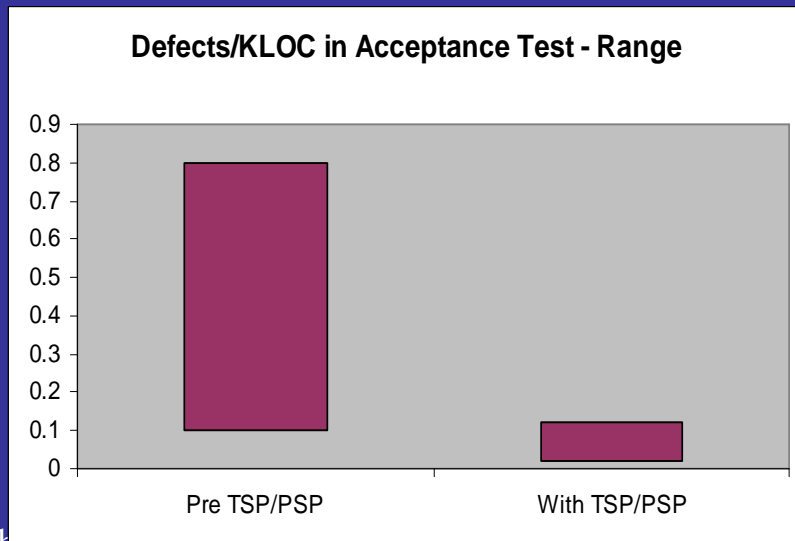
From the SEI class "Managing TSP teams" (Module 15)



PSP/TSP Results



N. Davis, J. Mullaney *The Team Software Process in Practice: A Summary of Recent Results* Sept. 2003.

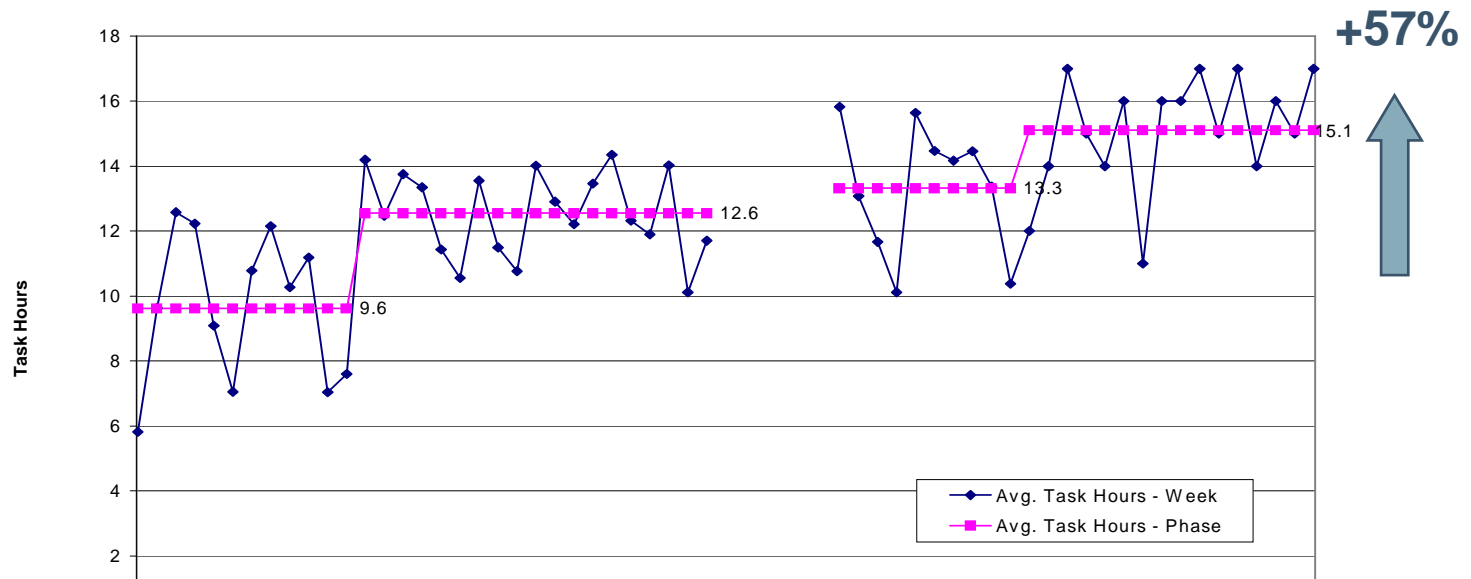




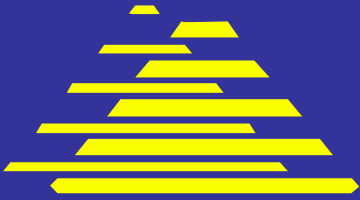
Improving Task Hours

Average task hours per developer per week were improved from 9.6 hours to 15.1 hours through quiet time, process documentation, more efficient meetings, etc.

Average Task Hours Per Week



Source: Allied Signal



Agile CMM vs. Agile

Similarities

Self directed teams
Well defined roles
Working sw and biz value
Incremental development
Manager as coach
Quick reaction/incremental
Reviews/inspections
Meaningful meetings
People interaction
Continuous self improvement

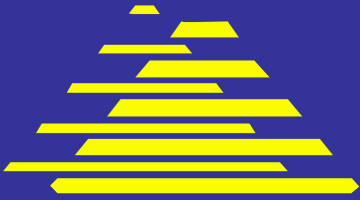
Differences

True blue agile quite a bit more:

- collegial
- code driven
- project centered
- test focused
- time boxed

And quite a bit **less worried** about

- detailed process documentation
- early phases planning/tracking
- product doc. (except code)
- detailed data (but there is some)
- CMM stuff!!!

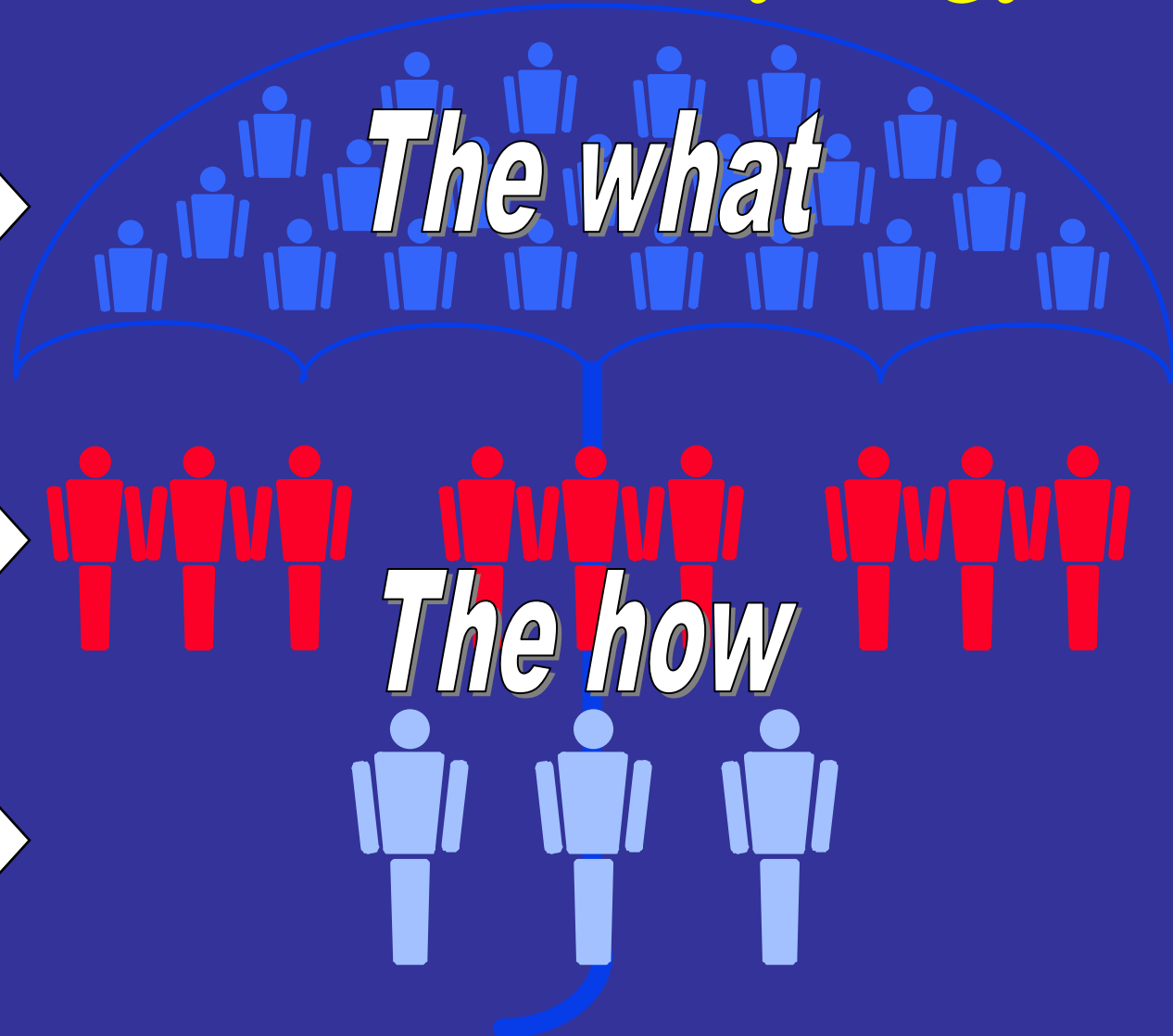


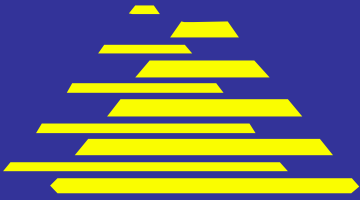
PSP/TSP-CMM synergy

**CMM - Builds
organizational
capability**

**TSP - Builds
quality products
on cost and
schedule**

**PSP - Builds
individual skill
and discipline**





To probe further-1

<http://www.agilemanifesto.org>

<http://www.agilealliance.org>

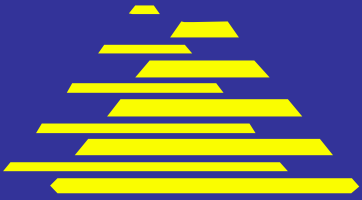
<http://www.xprogramming.com/>

<http://groups.yahoo.com/group/extremeprogramming>

Cockburn, Alistair *Agile Software Development* Boston, MA: Addison-Wesley, 2002.

Highsmith, Jim *Adaptive Software Development: A Collaborative Approach to Managing Complex Systems*, Dorset House, 2000.

Beck, K. *Extreme Programming Explained: Embrace Change*. Boston: Addison-Wesley, 2000.

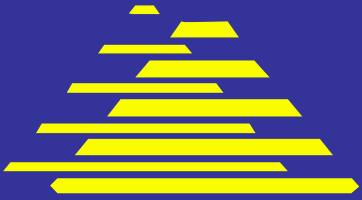


To probe further-2

David Kane, Steve Ornburn *Agile Development: Weed or Wildflower?*
Crosstalk October 2002.

Barry Boehm, Richard Turner *PSP/TSP vs. XP*. Oct. 24, 2003. Great on-line paper.
Read it at:

http://www.informit.com/isapi/product_id~%7B156D4D3C-6DA4-4E46-9591-34A341EB7DB2%7D/content/index.asp



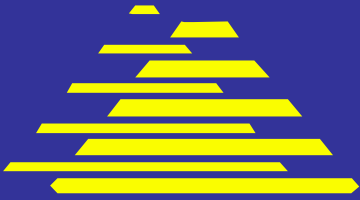
To probe further-3

<http://www.stsc.crosstalk.org>

Mark C. Paulk *Extreme Programming from a CMM Perspective*. IEEE Software Nov/Dec. 2001.

Mark C. Paulk *Agile Methodologies and Process Discipline* Crosstalk October 2002.

Donald J. Reifer *XP and the CMM*. IEEE software May/June 2003.

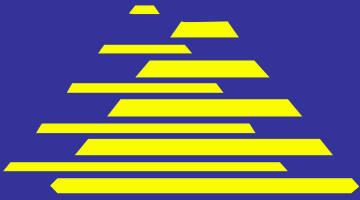


To probe further-4

<http://www.sei.cmu.edu/tsp/>

Noopur Davis, Jim McHale *Relating the Team Software ProcessSM (TSPSM) to the Capability Maturity Model for Software (SW-CMM)* CMU/SEI-2002-TR-008, June 2002.

Noopur Davis, Julia Mullaney *The Team Software ProcessSM (TSPSM) in Practice: A Summary of Recent Results* Sept. 2003.



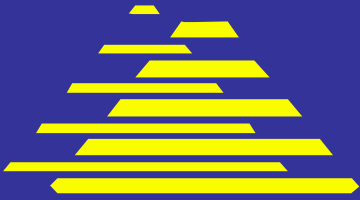
To probe further-5

“Stretching Agile to fit CMMI Level 3,” David J. Anderson,
http://www.agilemanagement.net/Articles/Papers/Agile_2005_Paper_DJA_v1_5.pdf

“Extreme Programming Explained: Embrace Change” (2nd Edition), Ken Beck, ISBN 0-321-27865-8 • Balancing Agility and Discipline – A Guide for the Perplexed, Barry Boehm and Richard Turner, Addison Wesley

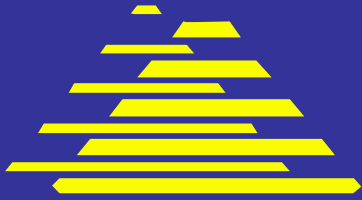
“CMMI –Guidelines for Process Integration and Product Improvement”, Mary Beth Chrissis, et al, Addison Wesley
• Crystal Clear: A Human-Powered Methodology for Small Teams, Alistair Cockburn, ISBN 0- 201-69947-8

“An Agile Approach to Achieving CMMI” Christine Davis, et al www.agiletek.com/thoughtleadership/whitepapers



To probe further-6

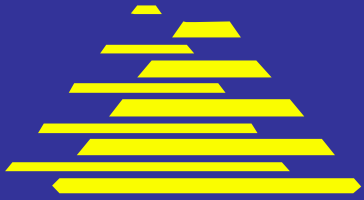
- “Real Time Embedded Software Development Using Agile Technology,” Vincent Rivas and Joseph N Frisina,
http://www.omg.org/news/meetings/workshops/RT_2005/01-4_Rivas-Frisina.pdf
- “A Practical Guide to Feature-Driven Development”,
Stephen Palmer and John Felsing, ISBN 0-130-67615-2
- “Lean Software Development” –An Agile Toolkit, Mary and
Tom Poppendieck, Addison Wesley
- “Agile Project Management with Scrum”, Ken Schwaber,
ISBN 0-130-67634-9
- “Lean Thinking –Banish Waste and Create Wealth in Your
Corporation”, James P. Womack and Daniel T. Jones,
Free Press



To probe further-7

Clement, Paul et al. *Documenting Software Architectures in an Agile World*,
CMU/SEI-2003-TN-023.

Smiley, Karen *Comparing Agile and TSP*,
First TSP Users Group Conference,
Pittsburgh PA, September 29-30, 2003.



To discuss some more...

Daniel M. Roy

Software Technology, Process & People, Inc.

20 Forest Rd

BradfordWoods PA 15015 (USA)

Land: +001 (724) 934 0943

Cell: +001 (412) 310 2263

E-mail: danroy@stpp.com

Web site: <http://www.stpp.com>