Doing CMMI the Agile way

Q:PIT Ltd

Reducing the Cost of Quality
through
Process Improvement,
Information Management
and Teamwork

Copyright Statement

SM Architecture Tradeoff Analysis Method, ATAM, CMM Integration, COTS Usage Risk Evaluation, CURE, EPIC, Evolutionary Process for Integrating COTS-based systems, Framework for Software Product Line Practice, IDEAL, Interim Profile, OAR, OCTAVE, Operationally Critical Threat, Asset, and Vulnerability Evaluation, Options Analysis for Reengineering, Personal Software Process, PLTP, Product Line Technical Probe, PSP, SCAMPI, SCAMPI Lead Appraiser, SCE, SEPG, SoS Navigator, T-Checks, Team Software Process, TSP, are service marks of Carnegie Mellon University

TM Carnegie Mellon Software Engineering Institute (stylized), Carnegie Mellon Software Engineering Institute (and design), Simplex, Stylized hexagon are trademarks of Carnegie Mellon University
Q:PIT is a trademark of Q:PIT Ltd, Milton Keynes, UK

® Capability Maturity Model, Capability Maturity Modeling, Carnegie Mellon, CERT, CERT Coordination Center, CMM, CMMI, FloCon, OCTAVE are registered in the U.S. Patent and Trademark Office by Carnegie Mellon University
The Q:PIT swirl is a registered trademark of Q:PIT Ltd, Milton Keynes, UK

Peter Leeson

- Director of Q:PIT Ltd
- SCAMPI Lead Appraiser
- CMMI Instructor
- SEI Visiting Scientist
- Over 35 years of software engineering
- 14 years as process improvement consultant
- Process Improvement experience in Australia, Belgium, China, Denmark, England, France, Germany, India, Ireland, Japan, Mexico, Netherlands, Poland, Romania, Spain…

Q:PIT Ltd

International association of independent process improvement professionals

Q:PIT means…

- Reducing the cost of **Q**uality through **P**rocess improvement, **I**nformation management and **T**eamwork

What This is About

"Doing CMMI"

- Implementing a process improvement programme that corresponds to the expectations of a model such as CMMI

"the Agile Way"

- Working according to the principles and concepts of Agile software development

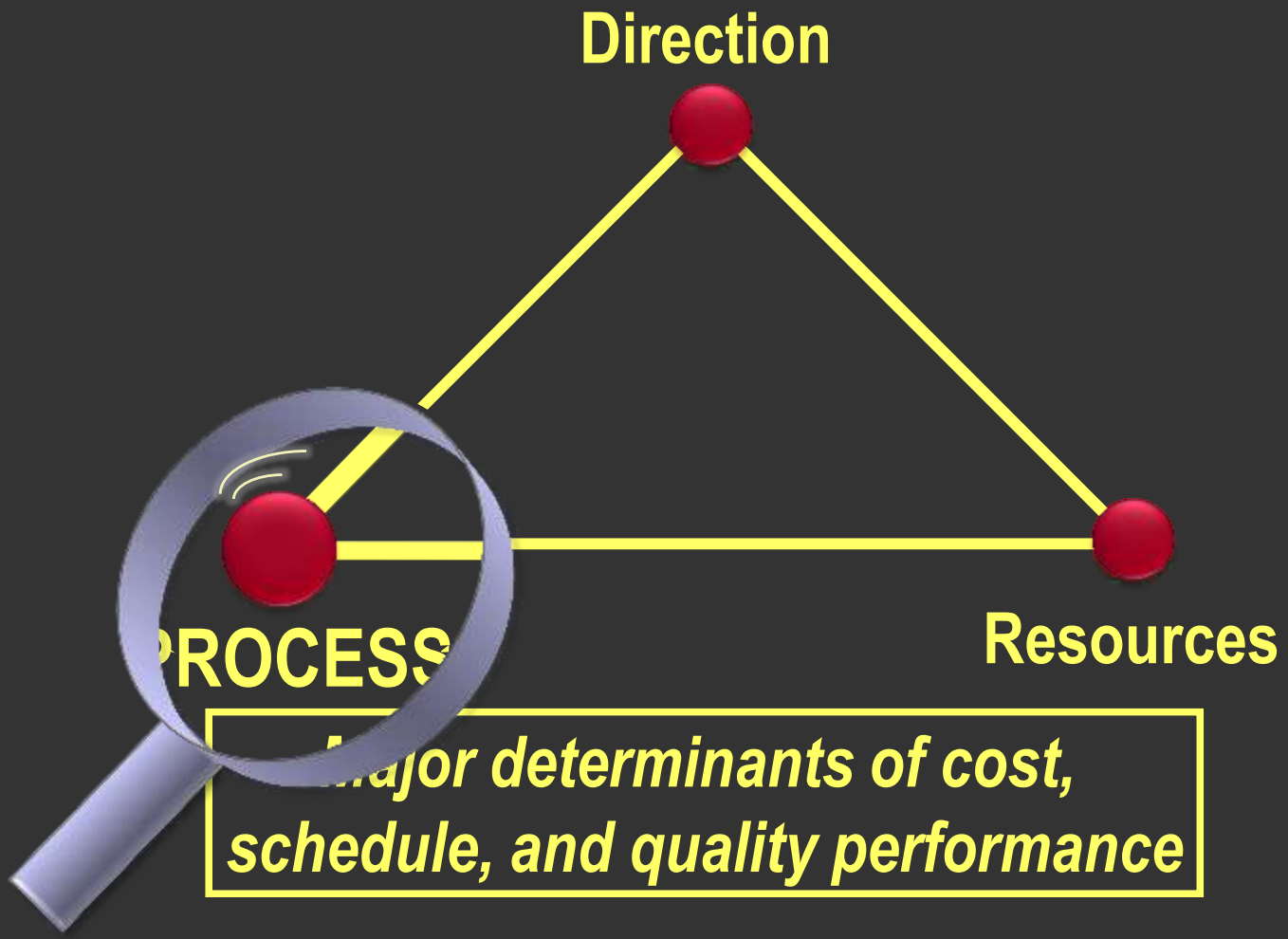
What this is not about

- Using CMMI and Agile in parallel as software development techniques

Process Improvement

What PI is all about

Ingredients for Success



To Engineer...

"Engineers today, like Galileo three and a half centuries ago, are not superhuman. They make mistakes in their assumptions, in their calculations, in their conclusions. That they make mistakes is forgivable; that they catch them is imperative.

Thus it is the essence of modern engineering not only to be able to check one's own work, but also to have one's work checked and to be able to check the work of others.

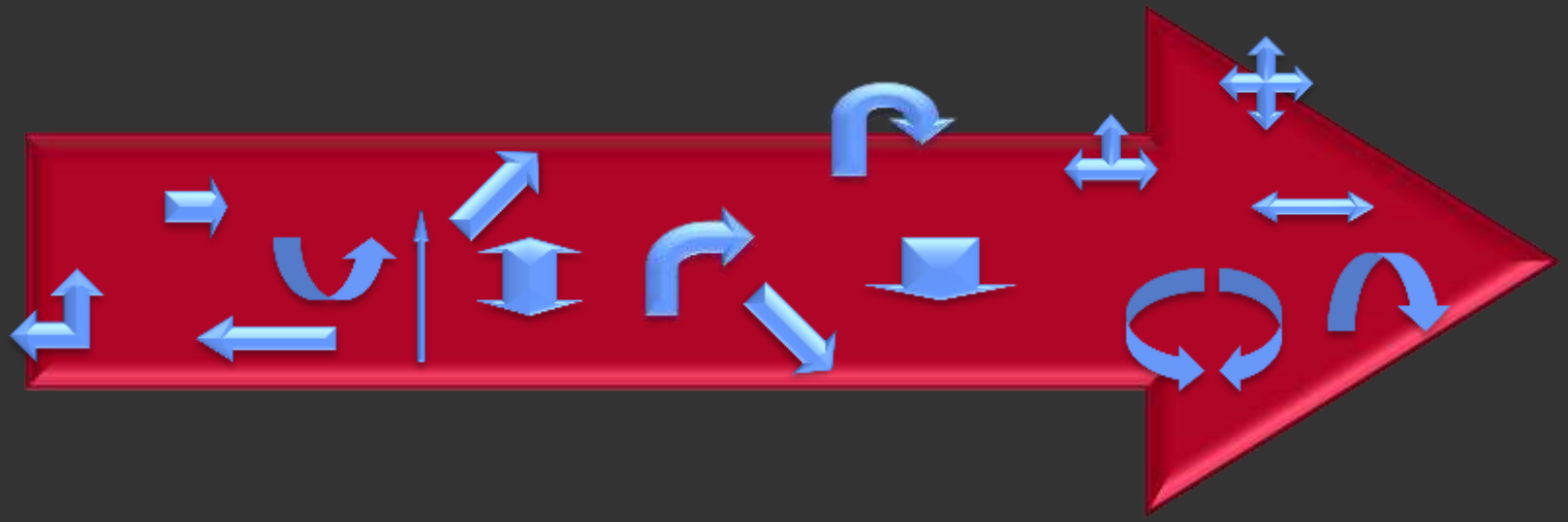
In order for this to be done, the work must follow certain conventions, conform to certain standards, and be an understandable piece of technical communication."

- Source: Petroski, Henry. "To Engineer is Human". NY: Vintage Books, 1992, p. 52

Life in an Immature Organization

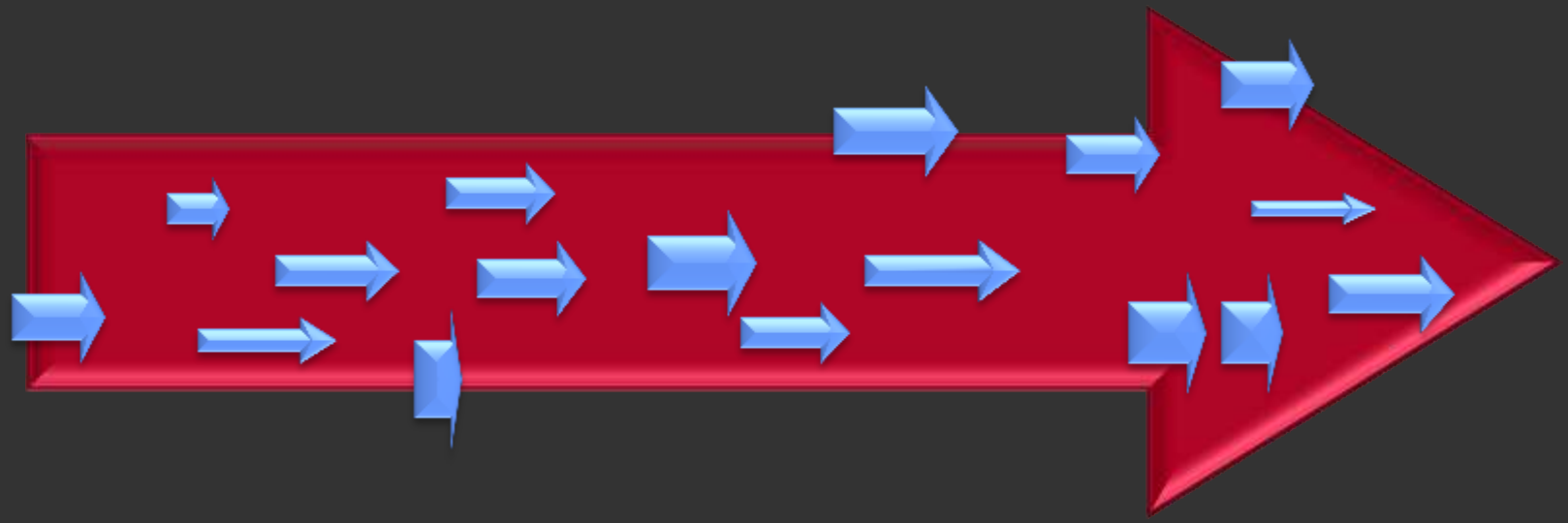
No consensus on what is the objective of goals of

- The organisation
- The project
- The product
- The process



Life in a Mature Organization

Working together towards a common goal



Quality

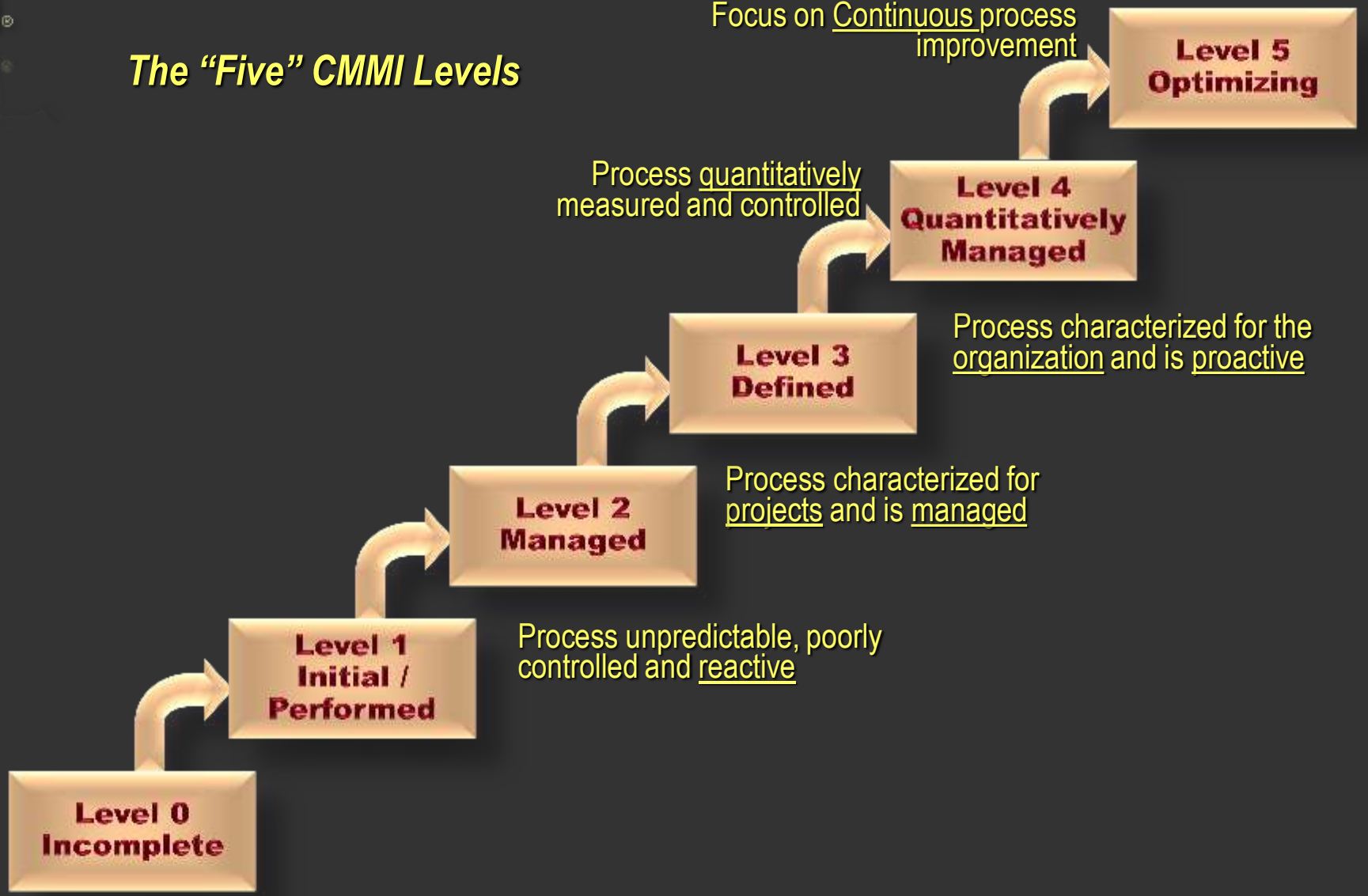
"Good quality is a stupid idea. The only thing that counts is your quality getting better at a more rapid rate than your principal competitors. It's real simple: if we are not getting more, better, faster than they are getting more, better, faster, then we are getting less better or more worse."

- Tom Peters

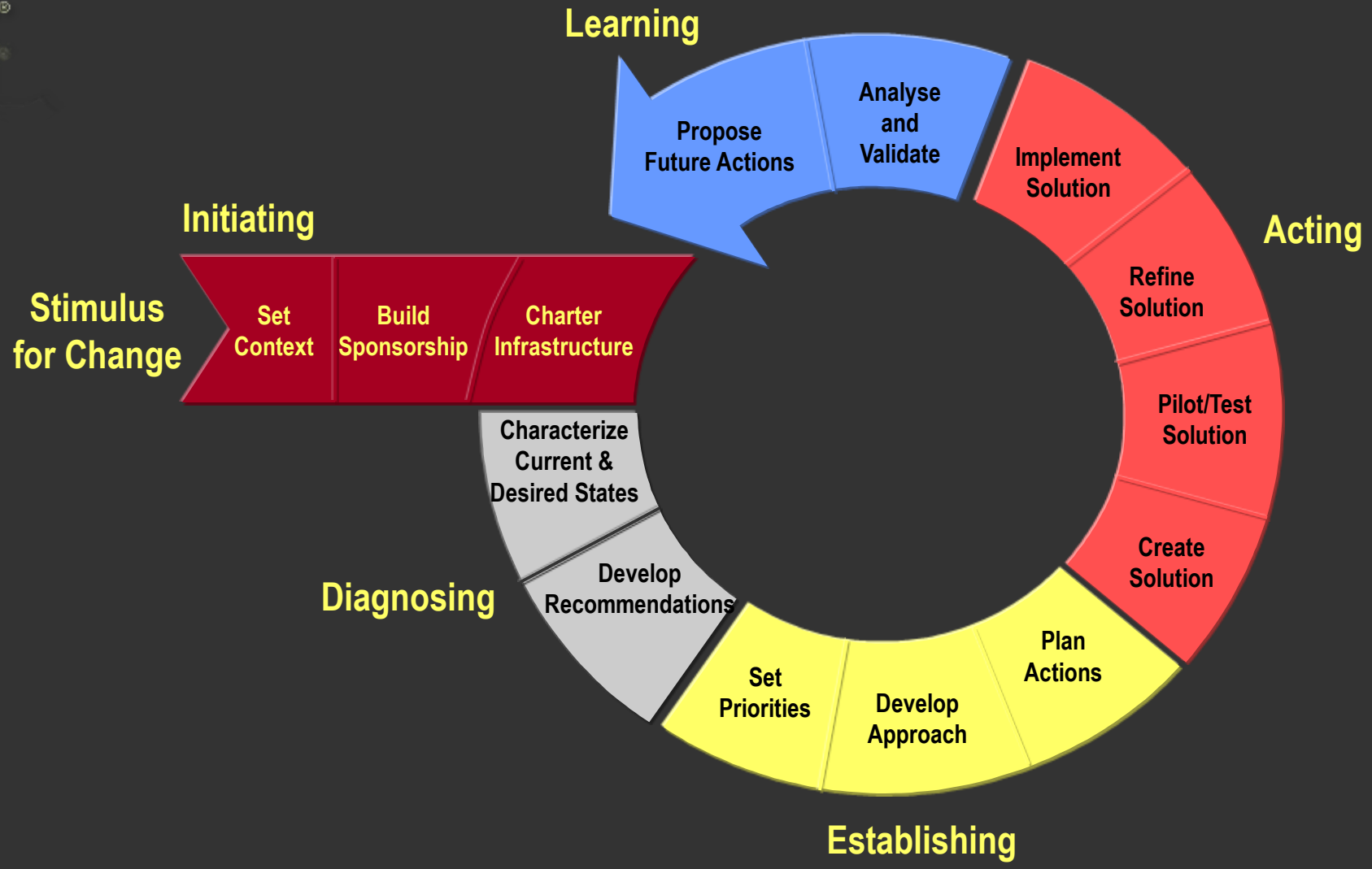
The Hierarchy of Quality

CMMI is a Model of Improvement Based on Processes

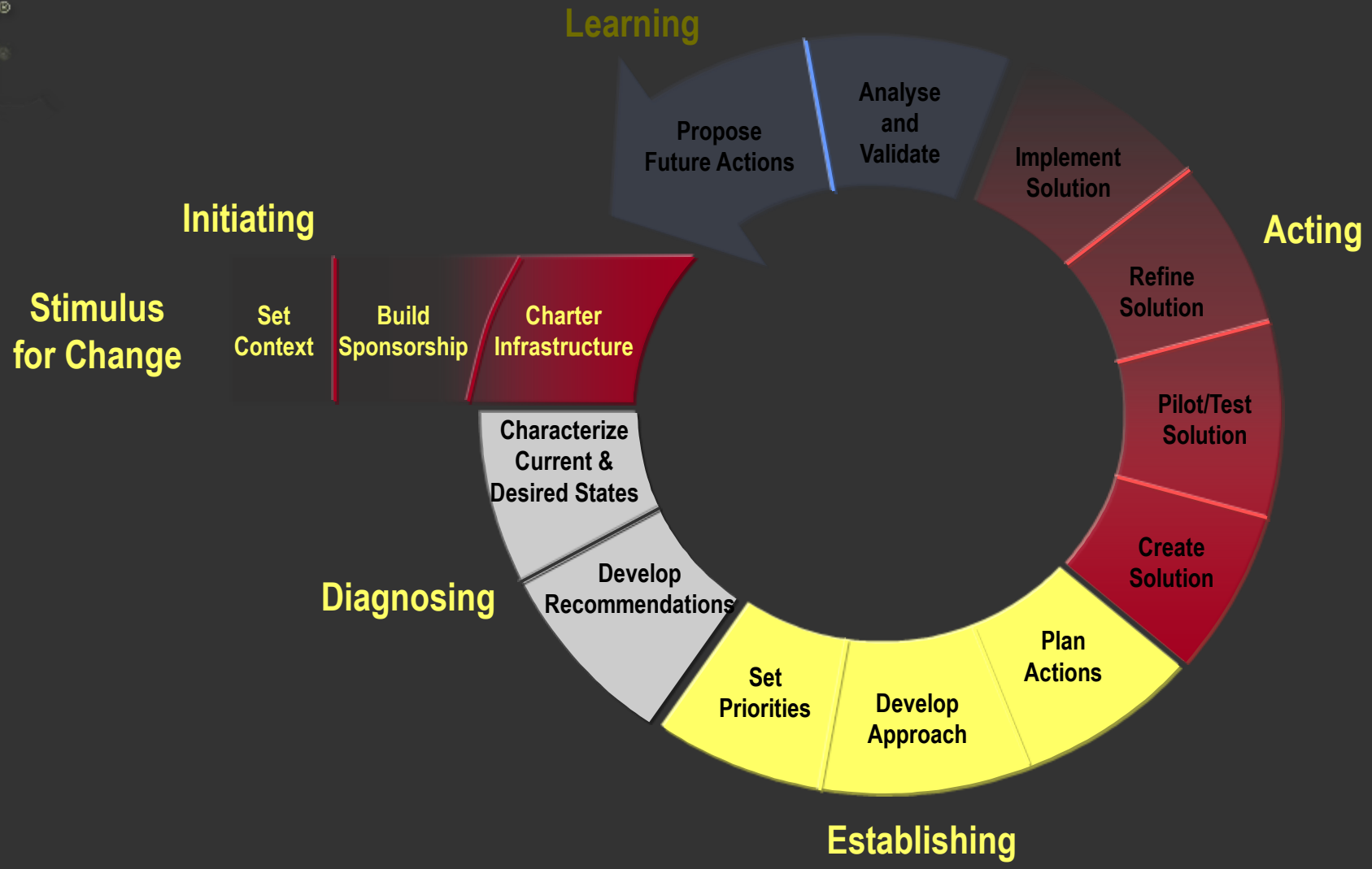
The "Five" CMMI Levels



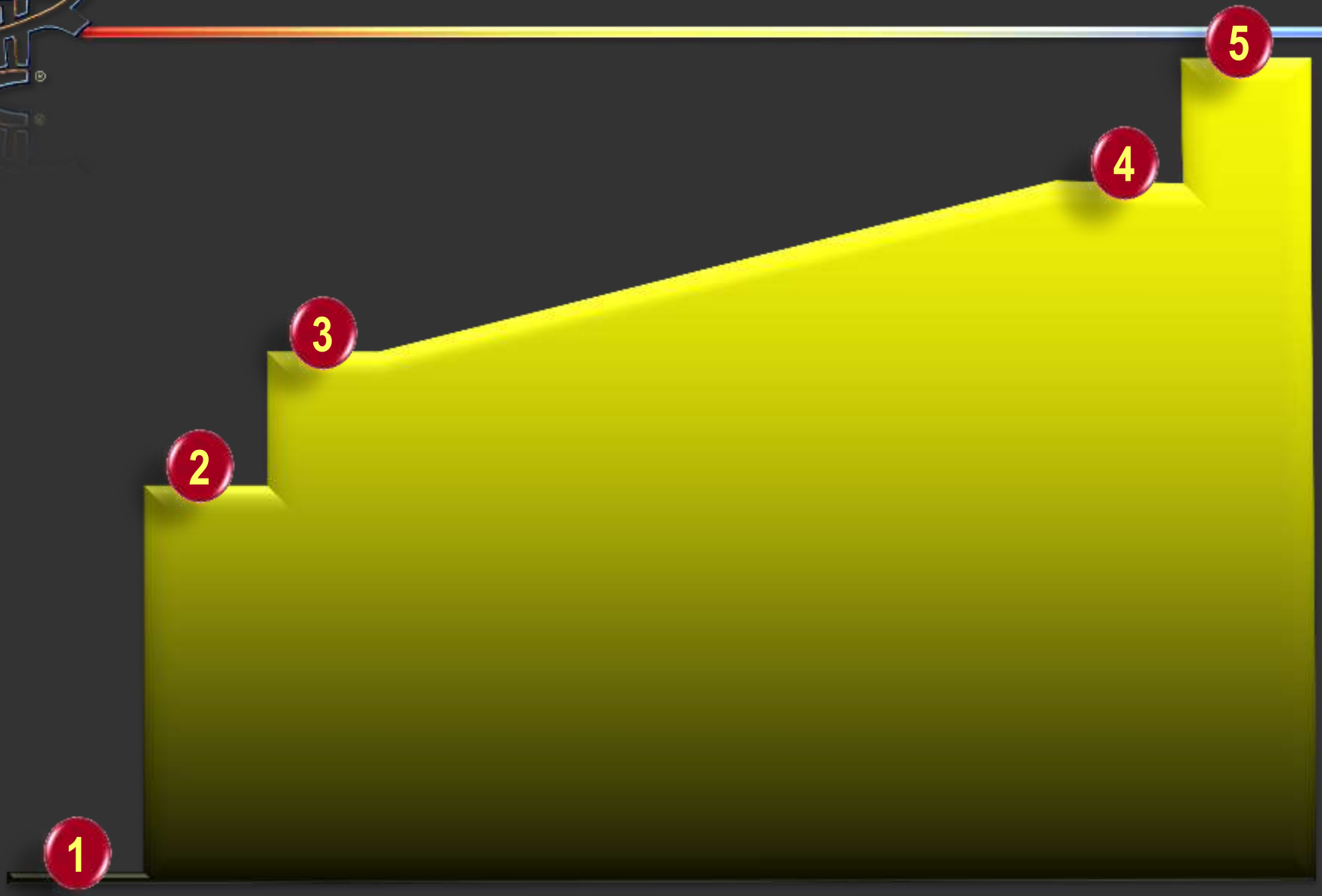
The IDEAL℠ Approach



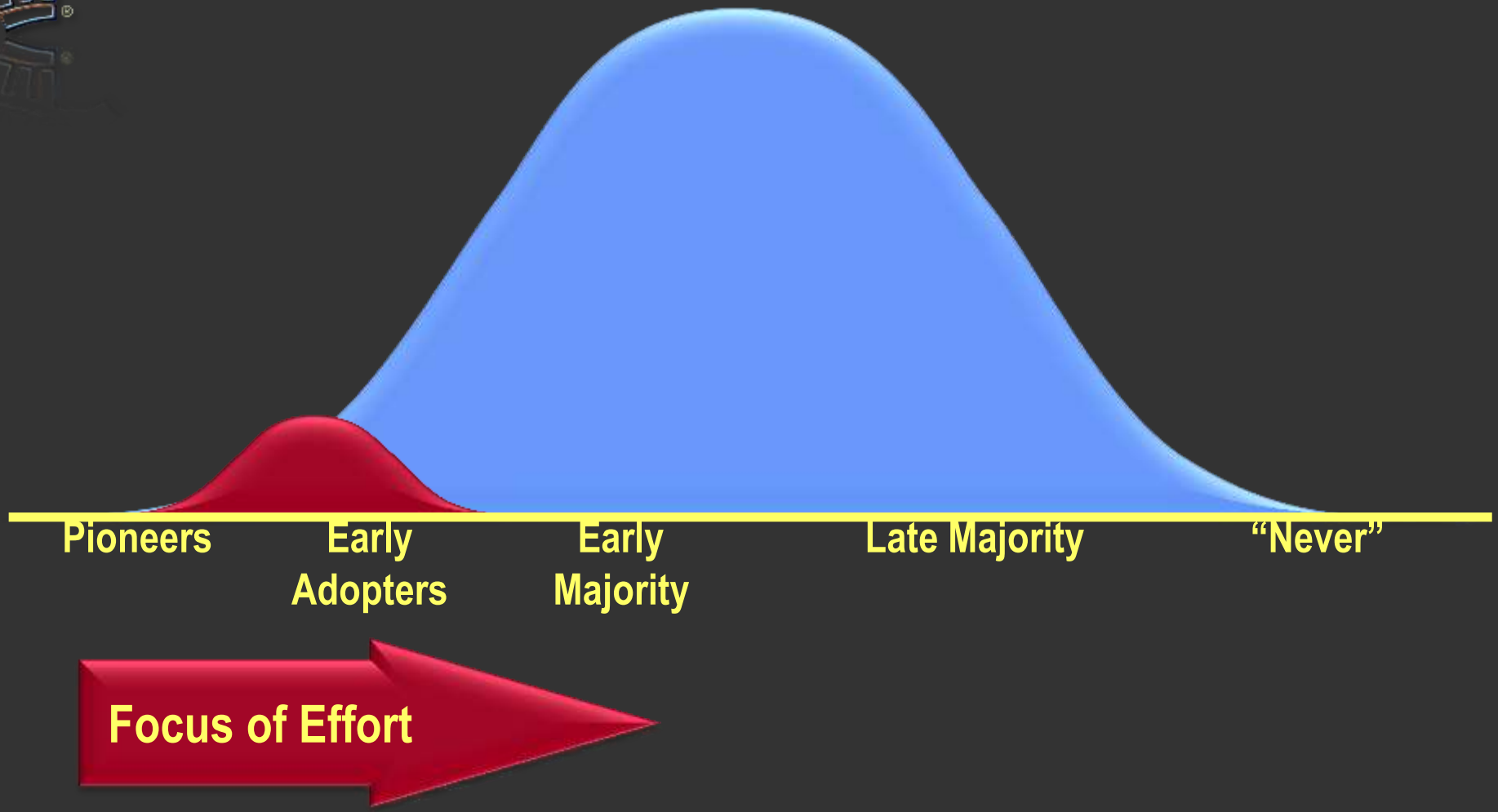
The IDEAL℠ Approach



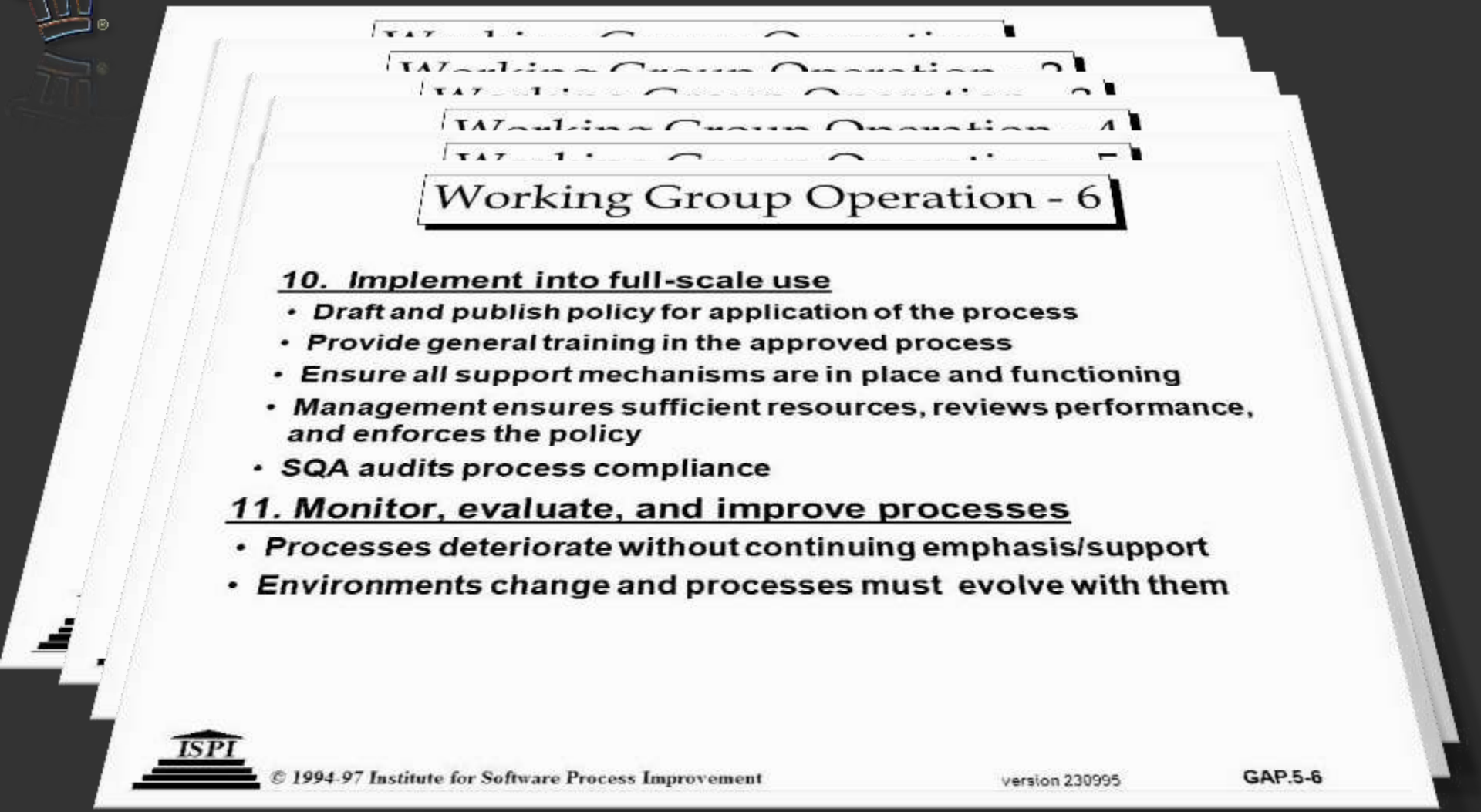
Effort



Process Adoption Curve



GWGM


Working Group Operation - 6
10. Implement into full-scale use
• Draft and publish policy for application of the process
• Provide general training in the approved process
• Ensure all support mechanisms are in place and functioning
• Management ensures sufficient resources, reviews performance, and enforces the policy
• SQA audits process compliance
11. Monitor, evaluate, and improve processes
• Processes deteriorate without continuing emphasis/support
• Environments change and processes must evolve with them
ISPI
© 1994-97 Institute for Software Process Improvement
version 230995
GAP.5-6

Additional Recommendations

- Regular small improvements
- Start with “quick-wins”, “low-hanging fruit”
- Plan the long term – start early

Causes of Failure 1/2

- Process Improvement is an engineering topic, not a management issue
- Level 2 is a project management issue, does not concern engineering
- Let's go straight to Level 3
- Let's buy our processes

Causes of Failure 2/2

SEPG is “different”

- No requirements
- No change management
- No planning
- No monitoring or control
- No clear objectives
- No measurement
- No progress
- No results

Agile

Some of the Basics

Agile Principles

Preference is given to:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

Agile Concepts 1/3

- Embracing Change
  - Seeing change as an ally rather than an enemy – providing quicker value to the customer
- Fast Cycle / Frequent Delivery
  - Scheduling many releases with short time spans between them
    - Forces implementation of only the highest priority functions
    - Delivers value to the customer quickly
    - Speeds requirements emergence
- Simple Design
  - Strip down design to cover just what is currently being developed.
    - Since change is inevitable, planning for future functions is a waste of effort

Agile Concepts 2/3

- Re-factoring
  - The restructuring of software to remove duplication, improve communications, simplify or add flexibility without changing its behaviour
- Pair Programming
  - A style of programming in which two programmers work side-by-side at one computer, continually collaborating on the same design, algorithm, code, or test
- Retrospective
  - A post-iteration review of the effectiveness of the work performed, methods used, and estimates
  - The review supports team learning and estimation for future iterations

Agile Concepts 3/3

- Tacit Knowledge
  - Agility is achieved by establishing and updating project knowledge in the participant's heads rather than in documents (explicit knowledge)
- Test-driven Development
  - Module or method tests are incrementally written by the developers and customers before and during code

Applying Agile to Process Improvement

If it works for them, why wouldn't it work for us?

Implementation

Applying the philosophy of Agile development to process improvement means:

- Accepting that the process needs and structure will have to change even before it is finalized
- Seeking to deliver real benefits rapidly and regularly
- Focusing on the simple common stuff rather than trying to solve everyone's problems before they occur
- Focus on communications and flexibility; make sure things are done once
- Team work, allowing the process users and process specialists to work together
- Continuous reviews of what is being done
- Learning lessons – actually learning them
- Ensure that the process responds to the need

Embracing Change - Why

Change or die!

The first reason for process improvement is to be ready for change and to be able to implement change:

- Market change
- Technological change
- Requirements change

Embracing Change - How

Continuous process improvement means working very closely with your customers:

- The people using the process will identify the weaknesses
- They may be willing to try the process, they will probably be willing to tell you why it does not work

Work with your customers (process users), listen to them, take their changes into account

Fast Cycle / Frequent Delivery – Why?

- Large changes are not easy to accept or assimilate
- Big “how-to” manuals are never read
- Infrequent news means people forget you, therefore do not see why you are telling them to change
- “I told them this did not work and they just ignored me and said ‘later’”

Fast Cycle / Frequent Delivery – How?

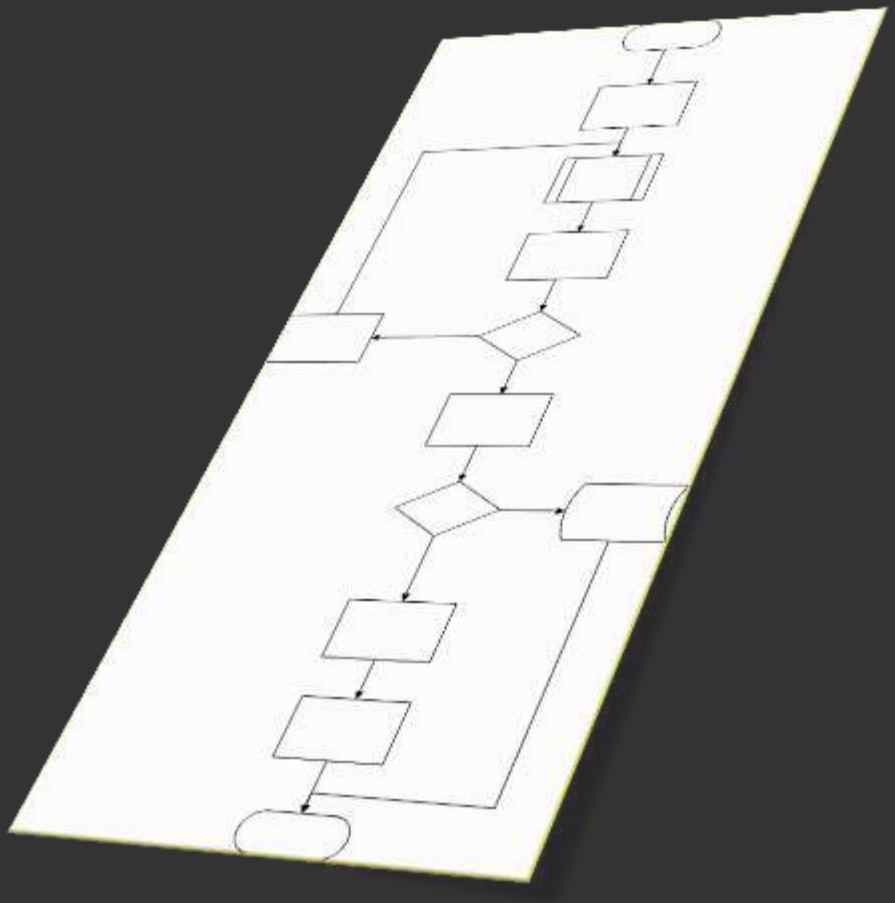
- Make sure you respond every request within 24 hours
- Plan on making a real, measurable improvement available every month ("sprint")

Simple Design – Why?

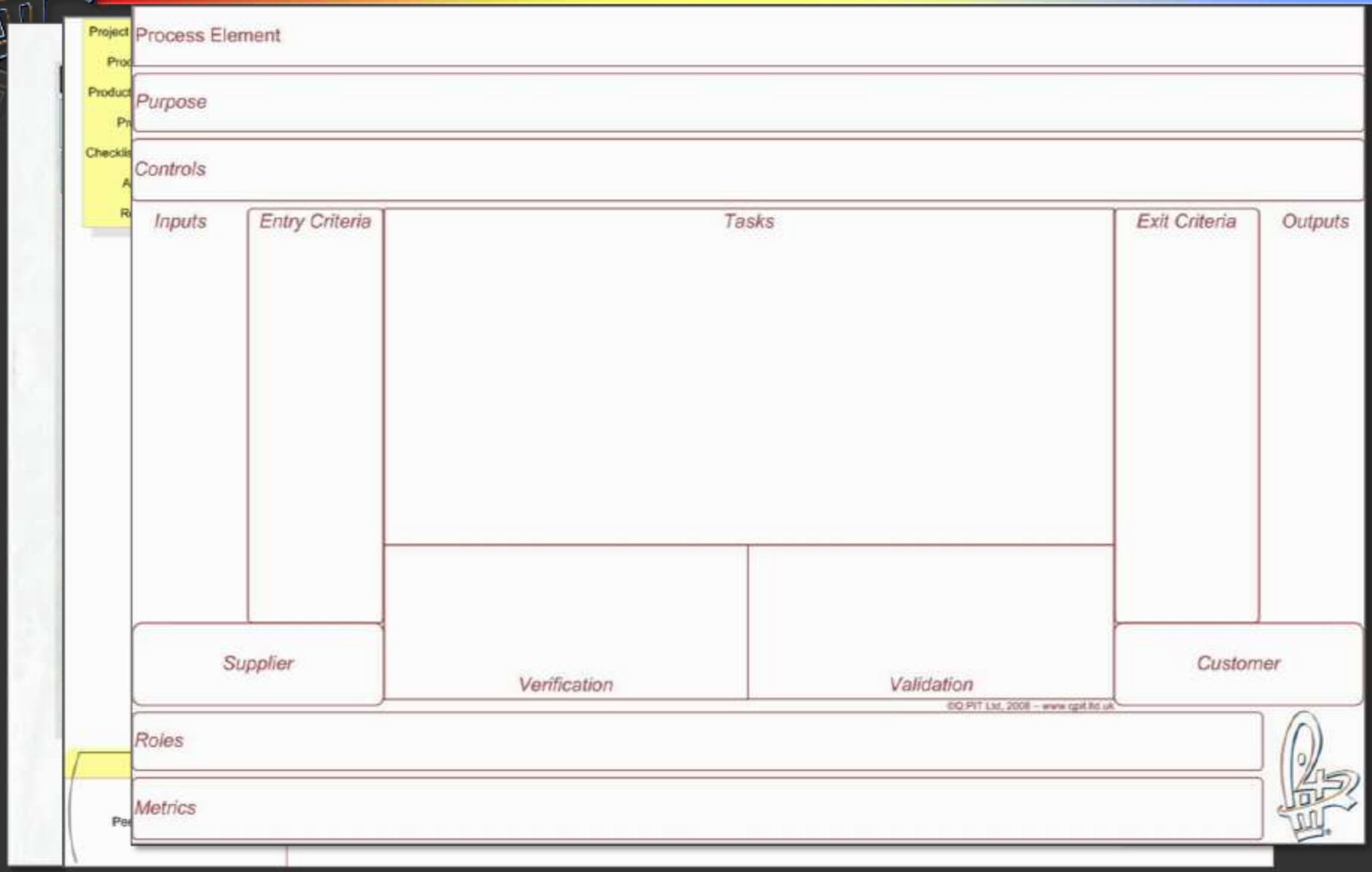
- Complex is difficult to change
- Complex is difficult to understand
- Don't try to solve all your problems in one go

Simple Design – How? 1/2

- Clear focus on what needs to be done and not how
- Present process as simple graphics with links to relevant documents and tools



Simple Design – How? 2/2



Refactoring

- Wikipedia definition:
  - “Code refactoring is the process of changing a computer programme's source code without modifying its external functional behaviour in order to improve some of the non-functional attributes of the software. Advantages include improved code readability and reduced complexity to improve the maintainability of the source code, as well as a more expressive internal architecture or object model to improve extensibility.”

Refactoring – Why?

- Update the process without modifying its usage and behaviour to
  - Improve non-functional attributes of the process
  - Improve process usability
  - Reduce complexity
  - Improve the maintainability
  - Improve process architecture
  - Improve extensibility

Refactoring – How?

Structure your processes in little components: a review is a review is a review, whether it is a requirements inspection or code review

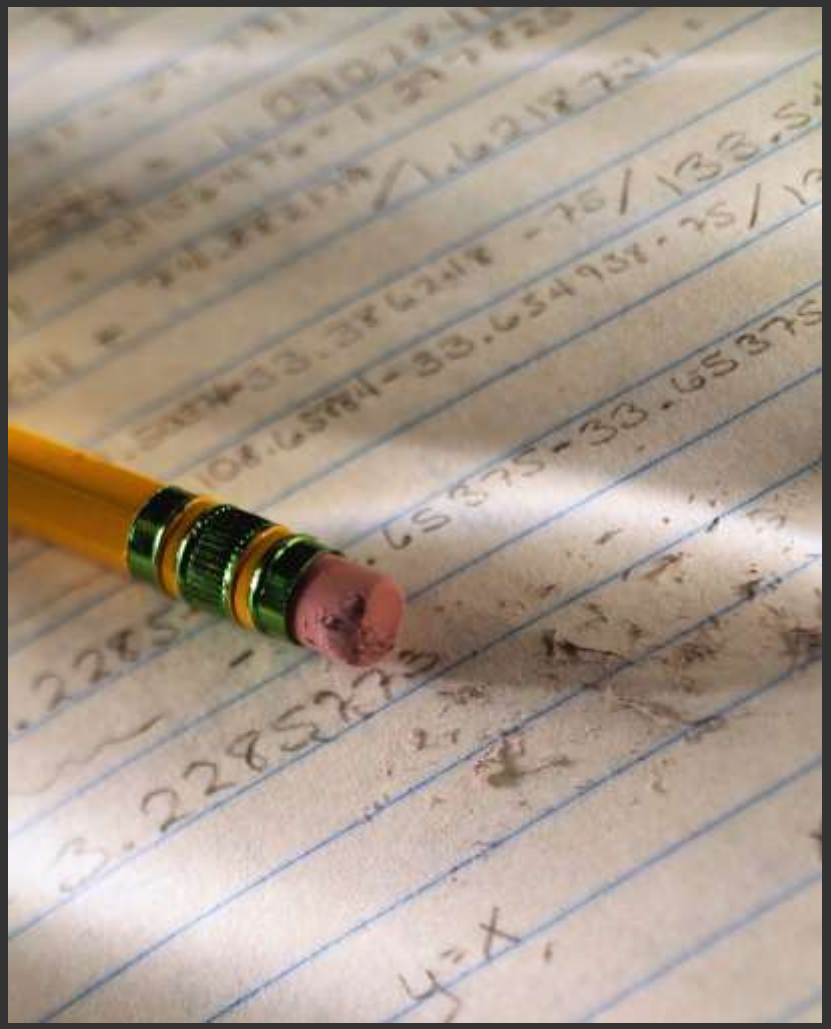
Build the processes with re-usable blocks rather than trying to recreate everything every time

Pair Programming – Why?

- Reviews are the cheapest and most efficient quality check methodology that exist
- Collaboration between the functional / technical specialist and the process specialist are key to success
- Integrated teams are even better…

Retrospective

If you don't understand the need and benefit of learning and applying lessons after each sprint, you need to go back to the beginning of the process improvement principles!



Tacit Knowledge

- Use the people who know, the people who understand what you are trying to do
- Make sure the SEPG people focus on the principles of process rather than on the functional content

Test-driven Development

If the resulting process does not work for the process improvement people, don't expect it to work for anyone else!

Conclusion

Conclusion

The reason process improvement fails is because the process people believe that they are different and should not follow the simple rules that they preach!

Agile principles are a very simple way to manage your improvements and deliver results that will measurably support your business objectives

Contact Information

Q:PIT Ltd

- PO Box 6066
  Milton Keynes
  MK1 9BH
  United Kingdom
- Tel: +44 (0)1 908 506 908
- Fax: +44 (0)7006 010 575
- Internet: http://www.qpit.net

www.cmmi.info

Peter Leeson

- Direct Line: +44 (0)20 8433 4120
- Mobile/Cell: +44 (0)773 998 98 67
- Skype & Hotmail: qpitpjl
- E Mail: Peter@qpit.ltd.uk