МУК 2022/2023:
Модели за управление на качеството на софтуера и ИТ услуги
(Увод в подобряване на процесите - PI, CMMI)

Software Quality Management Models:
Intro to Process Improvement (PI)

[SEMP Program course, 2009-2023 in FMI as МУК]

Dr. George Sharkov, Christina Todorova, Krassimir Baylov
ESI Center Eastern Europe/Bulgaria
gesha@esicenter.bg

www.esicenter.bg

Оганизация

Лекции:упражнения = 2:1 (15 седмици, 45 часа)

Понеделник, 18-21ч. – зала 200, ФМИ
Online: moodle + presentations
Координация - moodle
Students quarter – 15 minutes

November – Test 1 (parts 1,2,3)

> teams for presentations ("+1" bonus to the result of Test-2)
> or presentation on a case-study (one man show)

January – Test 2 (final > instead of exam)

Session – optional exam

ESI European Software Institute Center Eastern Europe

USAID FROM THE AMERICAN PEOPLE

AMERICA FOR BULGARIA FOUNDATION
Фондация Америка за България

SEMP SOFTWARE ENGINEERING MANAGEMENT PROGRAM

The course is developed (and compiled) jointly by ESI Center (Eastern Europe) and CMU from the main lines and materials for SEMP, in partnership with SEI/CMU.

It introduces students to process improvement as a main factor for the quality of products and services.

Based on process-oriented models - CMMI, the "industrial" standard developed by SEI/CMU, project management (PMI/PM BOK), personal/team management (PSP/TSP BOK), strategic planning (Balanced ScoreCards), information security.

Augmented by modern methods and techniques – Agile CMMI, Six Sigma, etc. Mapping between main industrial models and standards. Implementation. Models for quality improvement in small settings and SMEs. Business aspects – cost of quality, what is "the right model for my company", why invest in PI, what is the return, who can help.

http://semp.esicenter.bg/

Информация, източници:

ESI Center Eastern Europe - Resources:
https://esicenter.bg/resources
Education > Resources > (Software) Quality Management - CMMI
(+ the links: - model in pdf ver 1.3)

Links to CMMI models (from the new source – CMMI Institute, spin-off of Carnegie Mellon/SEI):
https://cmmiinstitute.com/resource-files/public/cmmi-v2-0-development-model (paid!!!)
[free] ver 2.0 Practices mapping (to ver 1.3)
https://cmmiinstitute.com/resource-files/public/v2-0-materials/cmmi-v2-0-to-v1-3-practice-mapping

> Access V 1.3 to download CMMI –DEV v 1.3 model *(free, upon registration)*

old SEI repository – VALID for FREE DOWNLOAD:
https://resources.sei.cmu.edu/asset_files/TechnicalReport/2010_005_001_15287.pdf

https://en.wikipedia.org/wiki/Capability_Maturity_Model_Integration

General sources (Software Engineering, Quality)

www.sei.cmu.edu
http://resources.sei.cmu.edu/library/
www.cmmiinstitute.com

Съдържание (модули)

1	Увод в управление на качеството. Компоненти и цена на качеството. Процеси. Преглед на моделите за управление на качеството и подобряване на процесите. Методи за оценка на зрелостта на ИТ-интензивни и софтуерни организации. Стратегически карти/Балансирана система от показатели (balanced ScoreCards).
2	Модел CMMI (ver 1.3). История, внедряващи организации. Обща структура. Процесни области. Генерични и специфични цели и практики. Презентации – Maturity/Capability нива на Continuous и Staged representations. Категории процесни области: Process Management, Project Management, Engineering, Support.
3	Процесни области от ниво 2 на CMMI. Детайлно представяне на: REQM – Requirements Management PP – Project Planning MA – Measurement and Analysis PPQA – Process and Product Quality Assurance CM – Configuration Management PMC – Project Monitoring and Control Преглед на:SAM-Supplier Agreement Management
4	Процесни области от ниво 3 на CMMI. Детайлно представяне на: RD – Requirements Development VAL - Validation VER - Verification RSKM - Risk Management TS - Technical Solution Преглед на: DAR - Decision Analysis and Resolution , IPM - Integrated Project Management , OPD - Organizational Process Definition , OPF - Organizational Process Focus, OT - Organizational Training , PI - Product Integration. Преглед на Maturity Level 4 и 5. **Обобщение на връзките между процесните области: Tying all together** **Update for ver. 2.0 (CMMI Institute)**
5	Внедряване на програма за подобряване на процесите на база CMMI. Адаптирани подходи – Agile CMMI, CMMI/ISO. Нови модели CMMI – CMMI for Services, CMMI for Acquisition. Оценка (SCAMPI), роли. DevOps, DevSecOps – Security Requirements (for SW), Security by Design, Resilience by Design (CERT RMM) TMM (Testing Maturity Model)
6	Подобряване на процесите в малки фирми – IT Mark. Компненти на зрелостта – бизнес, организация/процеси, информационна сигурност. Оценка на нивото и план за подобрения.

Practical Exercises

Objective 1: remember-understand-apply

1. SW project/product lifecycle
2. Cost of Quality
3. Process policy and definitions (samples – REQM, PP, CM)
4. Project Planning, Estimates, PMC
5. (optional) VER/VAL – Peer Review

Objective 2:analyze-evaluate

1. Presentations (team work exercise)
2. Case study (+ presentation)
3. Real project (team work) – RD, TS, PP (?)

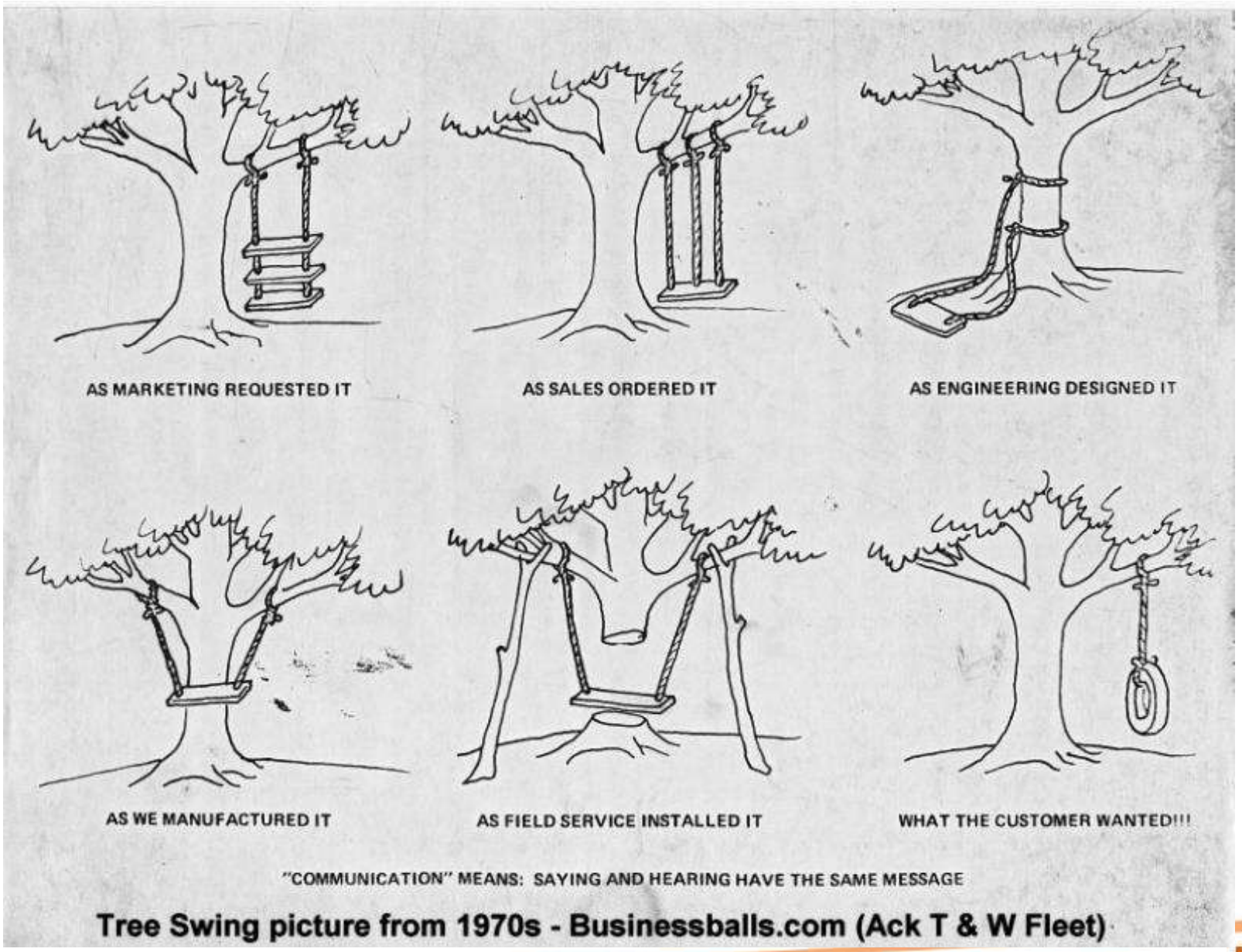
Objective 3: create

1. Elevator Pitch (perform)
2. Students quarter (15 minutes "free mind")

Why are we here?

What is Software Quality and how we assure it?

Do we want this?



Part 1: Introduction

Увод в управление на качеството. Компоненти и цена на качеството. Процеси. Преглед на моделите за управление на качеството и подобряване на процесите. Методи за оценка на зрелостта на ИТ-интензивни и софтуерни организации. Стратегически карти/Балансирана система от показатели (balanced ScoreCards).

Who are we?

ESI European Software Institute tecnalia

Since 1993

partner of:

Software Engineering Institute | Carnegie Mellon

ESI European Software Institute Center Eastern Europe

Since 2003

ESI Center Eastern Europe (ESI CEE)

PPP: SW Industry(BASSCOM), ESI & State ICT agency, supported by: USAID, UNDP

SEI Partner | Carnegie Mellon

CERT RMM

CMMI Institute Partner

CMMI

it mark

Affordable "BIG" standards for "small" companies

ESICenters • ESI@net Partners

Since 1993

European Software Institute

ESI European Software Institute tecnalia

SEI Partner CMMI

ZAINTEK servicio de vigilancia tecnológica e inteligencia competitiva

sare TEK

FEDIT FEDERACIÓN ESPAÑOLA DE ENTIDADES DE INNOVACIÓN Y TECNOLOGÍA

SEDISI Asociación Española de Empresas de Tecnologías de la Información

- Non-profit member-based Foundation
- Founded in 1993 by the European Commission and the Basque Government
- Established in Zamudio, near Bilbao, Spain

Software Engineering Institute | CERT | Carnegie Mellon

Software Engineering Institute | Carnegie Mellon

CMMI Institute

ISACA | CMMI

Carnegie Mellon University

Software Engineering Institute (SEI)

- Federally funded research and development center based at Carnegie Mellon University
- Basic and applied research in partnership with government and private organizations
- Helps organizations improve development, operation, and management of software-intensive and networked systems

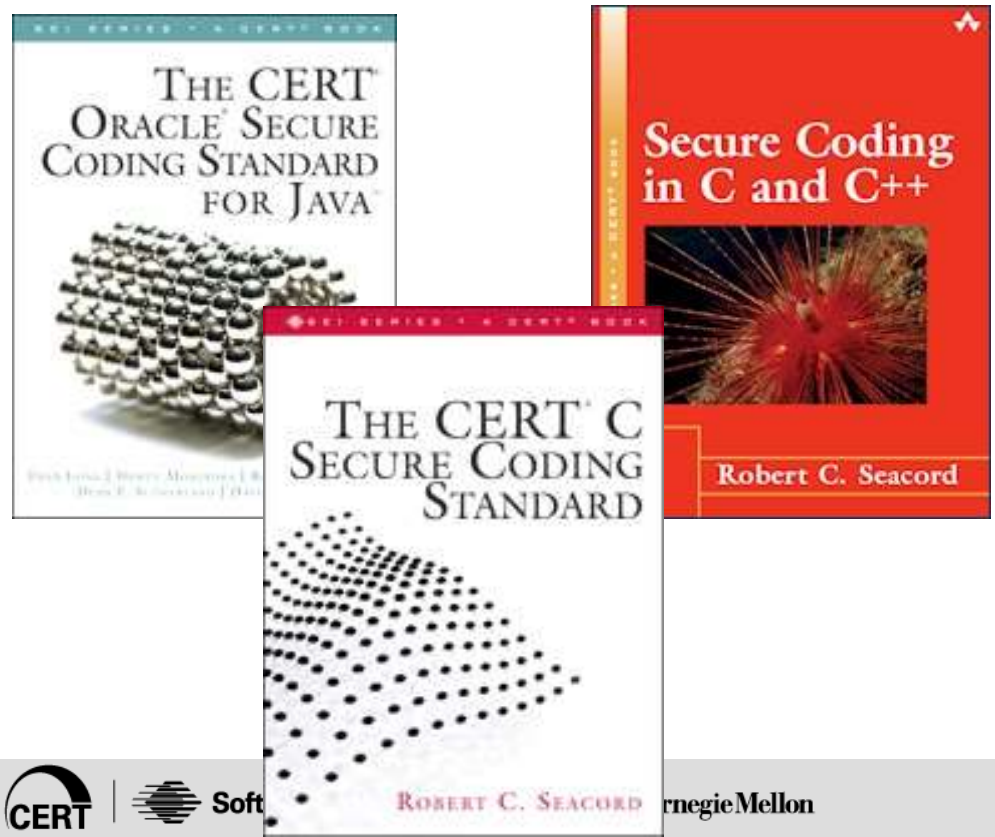
CERT – *Anticipating and solving our nation's cybersecurity challenges*

- Largest technical program at SEI
- Focused on internet security, digital investigation, secure systems, insider threat, operational resilience, vulnerability analysis, network situational awareness, and coordinated response

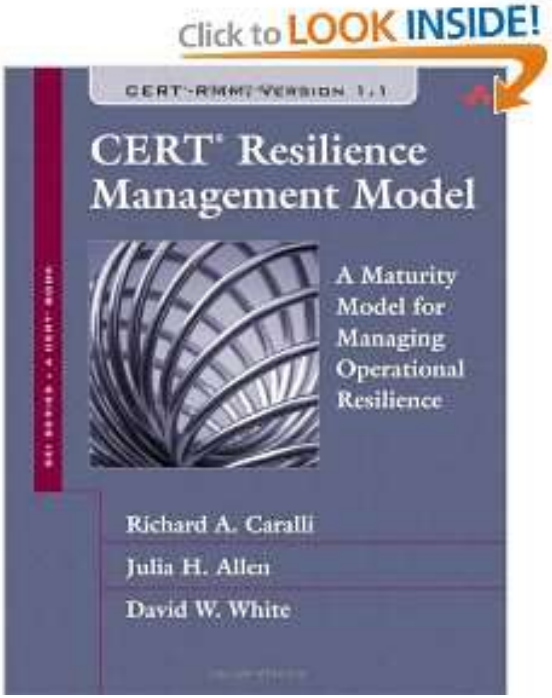
Also from SEI: Computer Emergency Response Team

Software Engineering Institute | Carnegie Mellon

Closing gaps & develop good code:
Secure Coding Standards
[languages + compilers]



Generic Model to Manage and Assess the Operational Resilience
[Information Security, Security Business Continuity]



small or BIG

business depends on excellence

What is excellence?

Corporate Excellence is *a* feature of an organizational entity that manifests how incomparably excellent it is when assessed adhering to success criteria (ISO, CMMI, 6 Sigma etc.); excellence refers always to excellent performance concerning the best methodologies in the world and it manifests in official certification according to them.

Corporate excellence perspectives

Corporate excellence is a balanced model

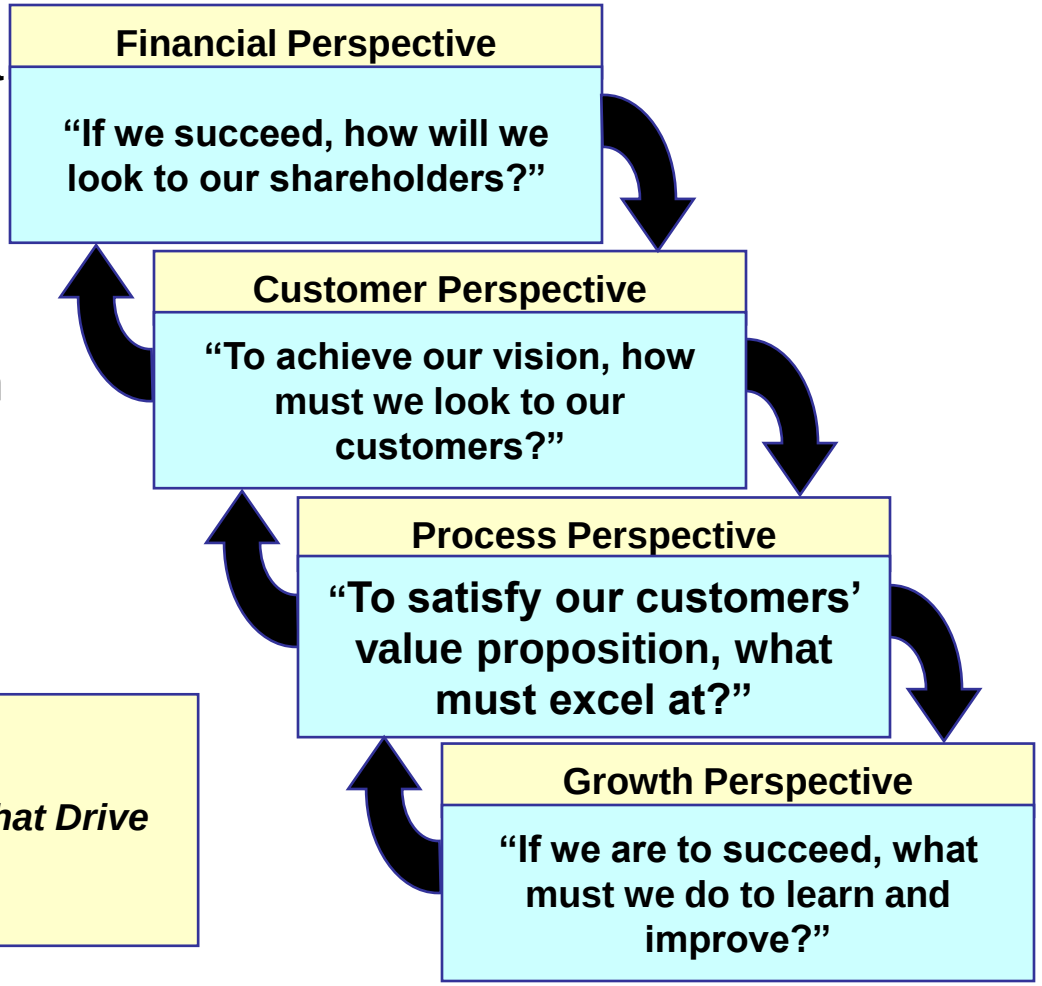
Kaplan and Norton structured it in four perspectives:

- Financial perspective
- Customers perspective
- Internal Processes perspective
- Learning & Growth perspective (Organizational Capacity)

https://balancedscorecard.org/bsc-basics-overview/

So what is the Balanced Scorecard?

The Balanced Scorecard is a framework for translating a vision into a strategy by focusing on shareholder, customer, internal and learning requirements which collectively describe the strategy of an organisation and how that strategy can be achieved.



Kaplan & Norton
Harvard Business Review ,1992
"The Balanced Scorecard - Measures that Drive Performance"

Excellence is in:

repeating the success

turn it to sustainable growth

make the best with your people

for higher profit

Financial Perspective

Results-oriented perspective that covers goals and performance measures related to the financial performance of the company.

Typical indicators: Return on Investment (ROI), Shareholder Value, Increase of Revenue, Increase of Turnover, Cash Flow, etc.

Customer Perspective

Related to the market and customer segments and it directly supports the implementation of financial objective.

Typical indicators are: market segments, customer satisfaction, percentage of new customers, life cycle, quality, service, price - quality, delivery times, reputation, commitment to delivery times

Process Perspective

Defines and measures the processes, in which the company should invest and improve so that it can attain the goals in the customer and finance related perspectives.

Typical indicators: Processing time, % millstones met , process frequency, process costs, process quality, time to market, innovation cycle etc.

Learning and Growth Perspective

Structuring goals and performance measures related to the knowledge necessary for maintenance and further development of all perspectives.

Typical indicators: market innovation, intellectual competences, staff satisfaction, fluctuation, staff productivity, number of improvement proposals, quality of improvement proposals, training days, etc.

Corporate excellence – FINANCIAL

The RESULT produced by the corporate excellence is high profitability

- The major goal of the companies is to produce profit for their shareholders rather than have the "ideal company"
- Corporate excellence is a tool for sustainable financial results
- The key social impacts of corporate excellence are higher employment and increased fiscal stability

Corporate excellence – CUSTOMERS

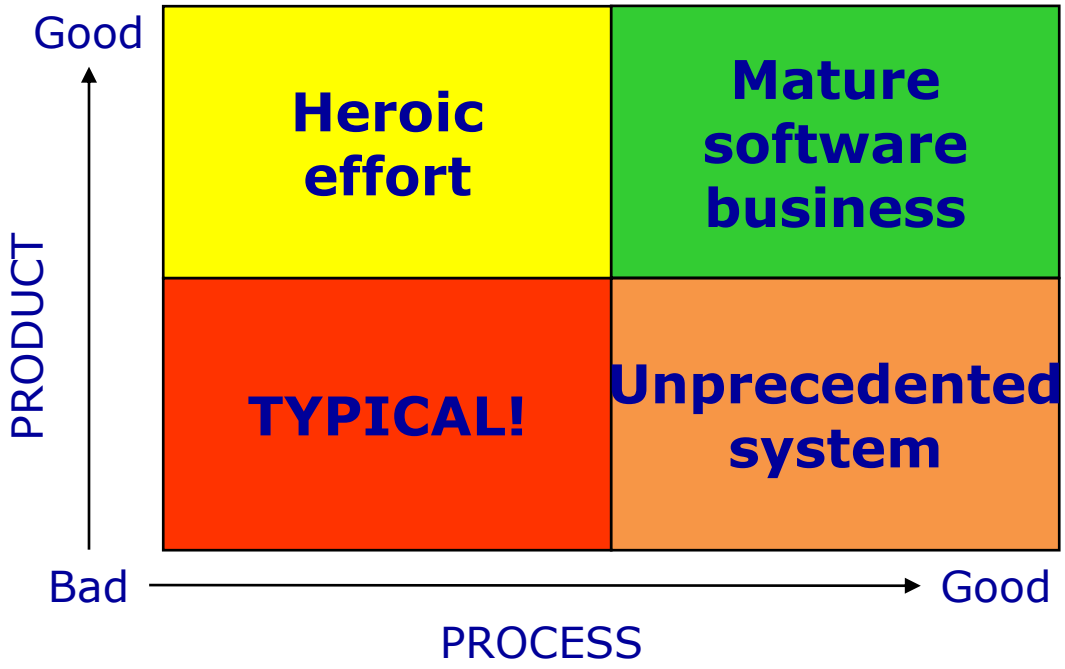
The corporate excellence is CERTIFIED by the customers

Understanding, predicting and managing **the customers expectations are critical:**

low cost	<->	creativity and efficiency
coding	<->	complex solution
outsourcing	<->	partnership with the clients
competition	<->	"coopetition"

Corporate excellence – INTERNAL

The corporate excellence is BASED on good internal processes



"The quality of a product is largely determined by the quality of the process that is used to develop and maintain it."

Based on TQM principles as taught by Shewhart, Juran, Deming and Humphrey.

Corporate excellence – LEARNING and GROWTH

The corporate excellence is EMPOWERED by learning and innovations

- Motivated and qualified human resources
- Knowledge management
- Organizational learning

Why focus on the processes?

Quality Is More Than Making a Good Product

The company inside: Why should a manager care about the software process?

"It's very difficult to consistently deliver quality products to your customers, while also making a profit, if your development process is poor."

The sad truth

25% of all software projects are killed.

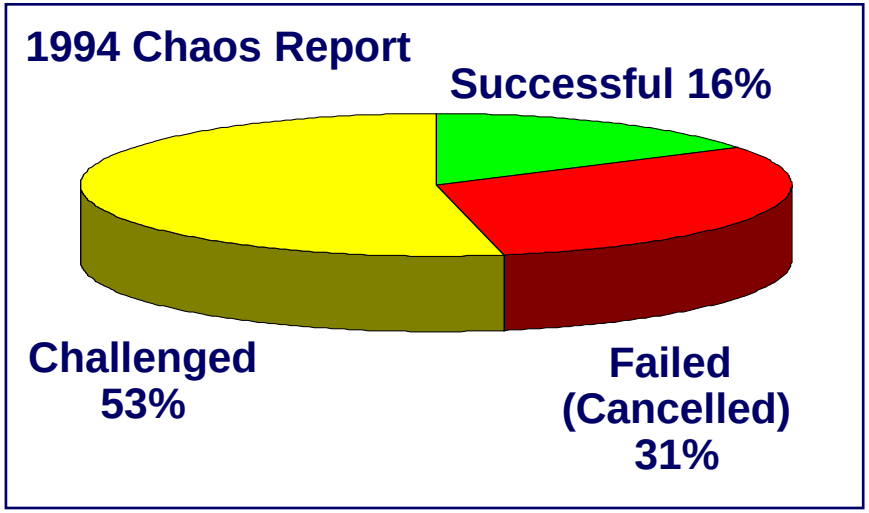
Companies are releasing products to their customers with 15% of the defects remaining in the product.

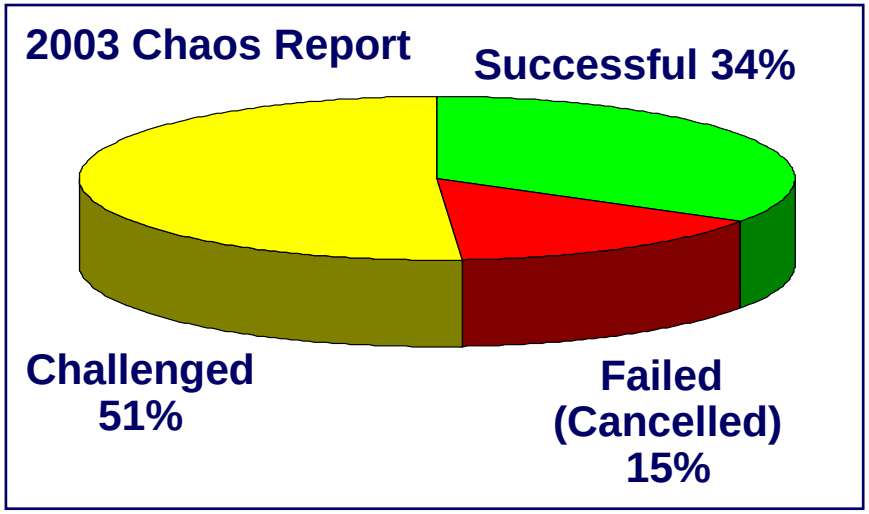
Many companies are spending 30-44% of their time and money on reworking software they have already written.

Companies meet their schedules only 50% of the time.

Sources: Capers Jones and Bill Curtis

We're getting better, but …





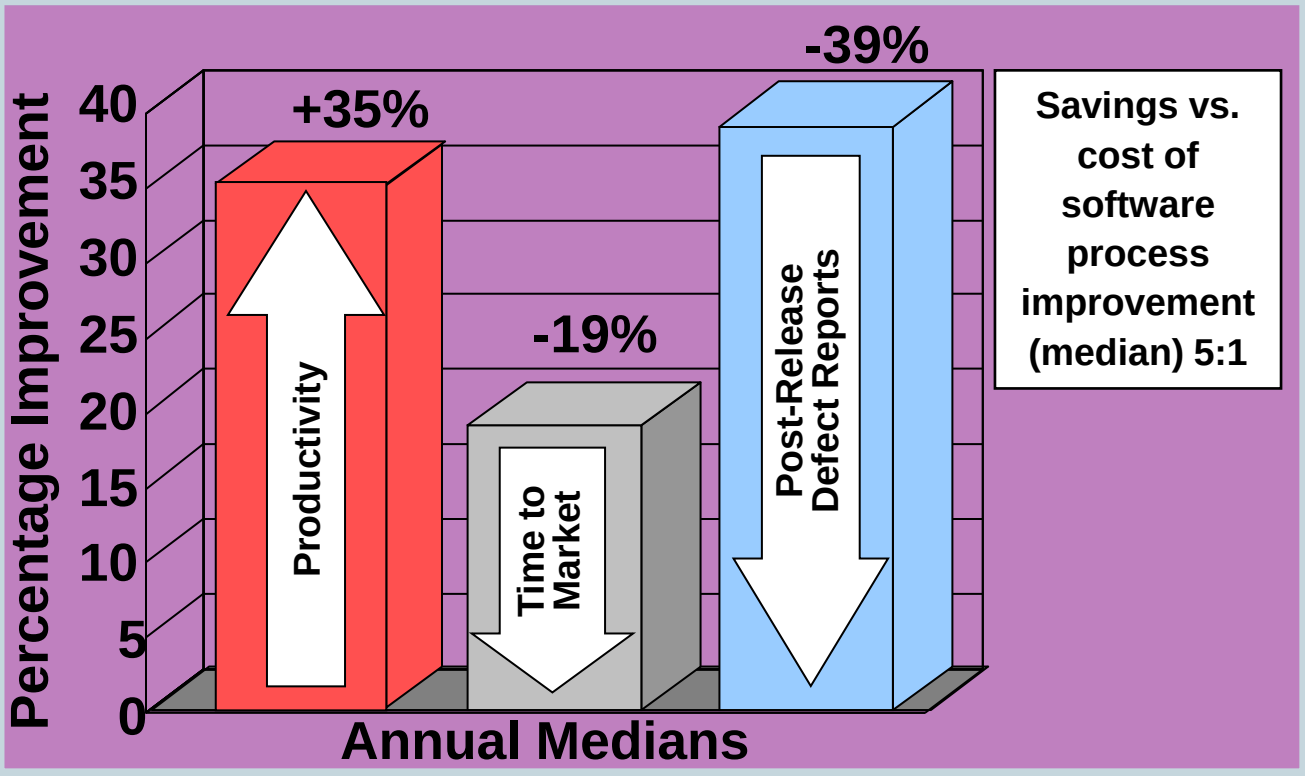
Source: Standish Group Chaos Report - 2003

- Project waste has dropped from 32% to 21.5% of project spending
- Cost overruns have dropped from 180% to 43%
- Project waste of $55 billion against $255 billion in project spending
- For every 100 project starts, there are 94 restarts
- 52% of required features and functions make it to the released product
- Projects cost, on average, 143% of the original estimate and 82% have schedule overruns

Definitions	
Successful	on time, on budget, promised functionality
Challenged	late, over budget and / or missing functionality
Failed	Severely impaired projects; cancelled projects

Things are Looking Brighter

Improvements From Adopting SW-CMM (SEI, 1994)

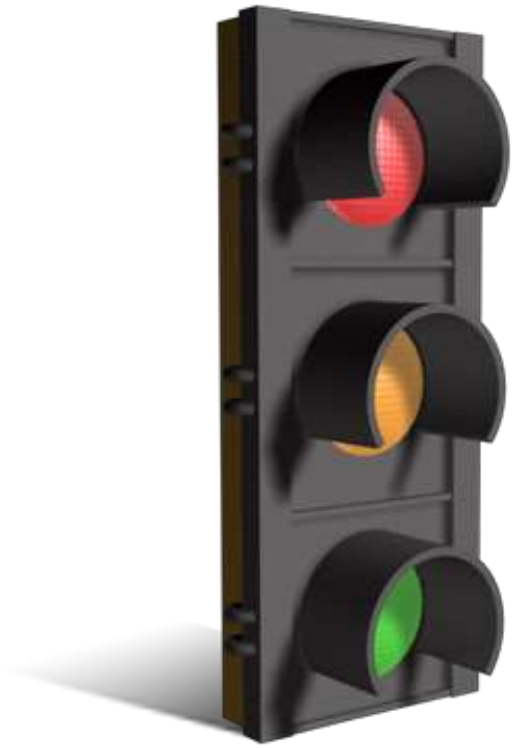


Current ROI Value to Programs (DACS, 1999)

Development Costs	Reduced 73%
Rework Costs	Reduced 96%
Average Schedule Length	Reduced 37%
Post-Release Defects	Reduced 80%
Weighted Risk Likelihood	Reduced 92%
Return On Investment	21:1

Expect Even Higher ROI For CMMI

You can only do 3 things



Work harder

Hire better people

Invest in improving the processes that you use to do your job

Cost of Quality (CoQ)

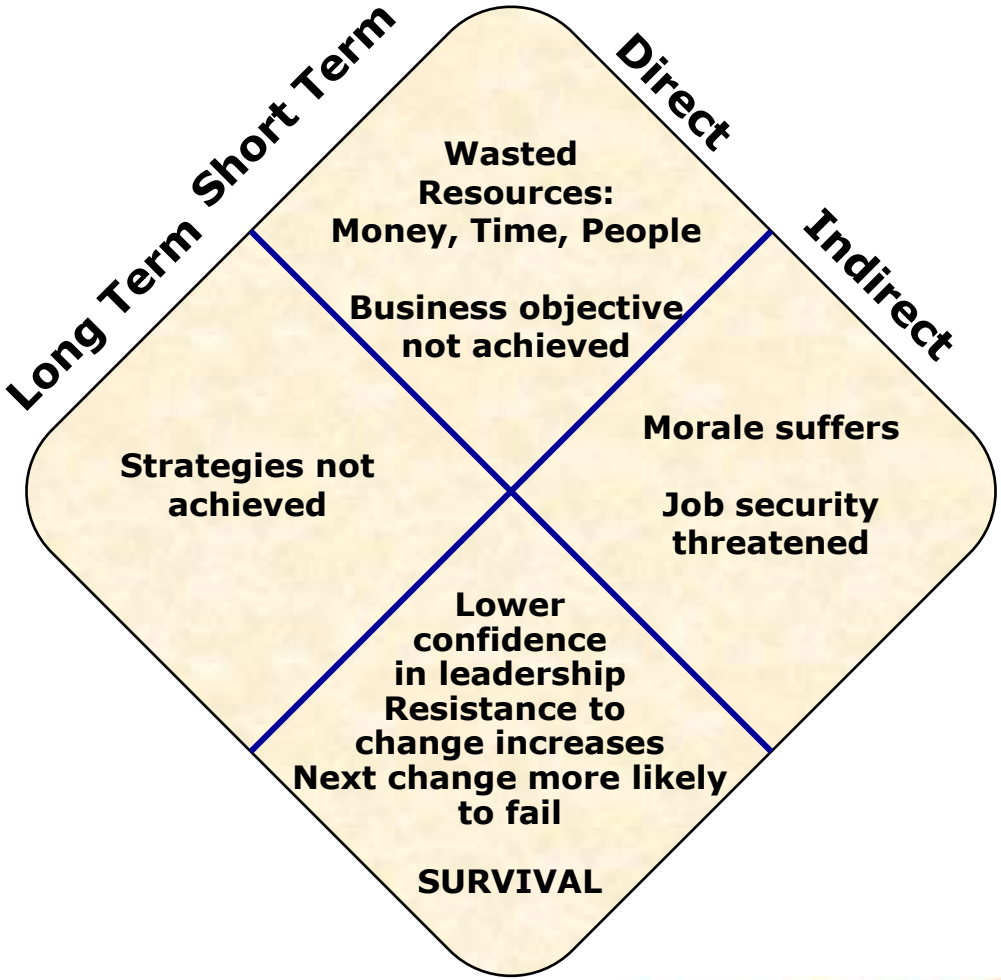
Cost of implementation failure

Quality is NOT Free…

- Cost of conformance

…but quality is cheaper than the alternatives

- Cost of nonconformance



Cost of Quality (CoQ)

Crosby describes Cost of Nonconformance as the extra cost incurred because a product or service wasn't done right the first time.

Cost Categories

Cost of Nonconformance
+ Cost of Conformance
= Cost of Quality

- Internal Failures + External Failures
- Prevention + Appraisal

CoQ Cost Categories (exercise)

Prevention	Appraisal	Internal Failure	External Failure
Costs associated with preventing defects	***Costs associated with "looking" for defects***	***Costs associated with defects found prior to implementation / release***	***Costs associated with defects found after the product is implemented / released***
Planning	Reviews	Rework	Warranties
Documentation	• System	• Requirements	Complaint adjustments
Training	• Requirements	• Design	Lost projects
Tools	• Design	• Code	Tech support
Policies and procedures	• Test Plan	• Documentation	Subsequent releases, patches, "Service Packs" (MS terminology)
Quality improvement projects	• Test Script	Defect re-testing	
Data gathering and analysis	Walkthroughs and code inspections	Process losses (testing downtime, changing deliverables, schedule slips, cost overruns, etc.)	
Fault and root cause analysis	• Testing (First-time)		
Quality reporting	Audits		
	CMM Assessments		
	• Class A,, B, C		

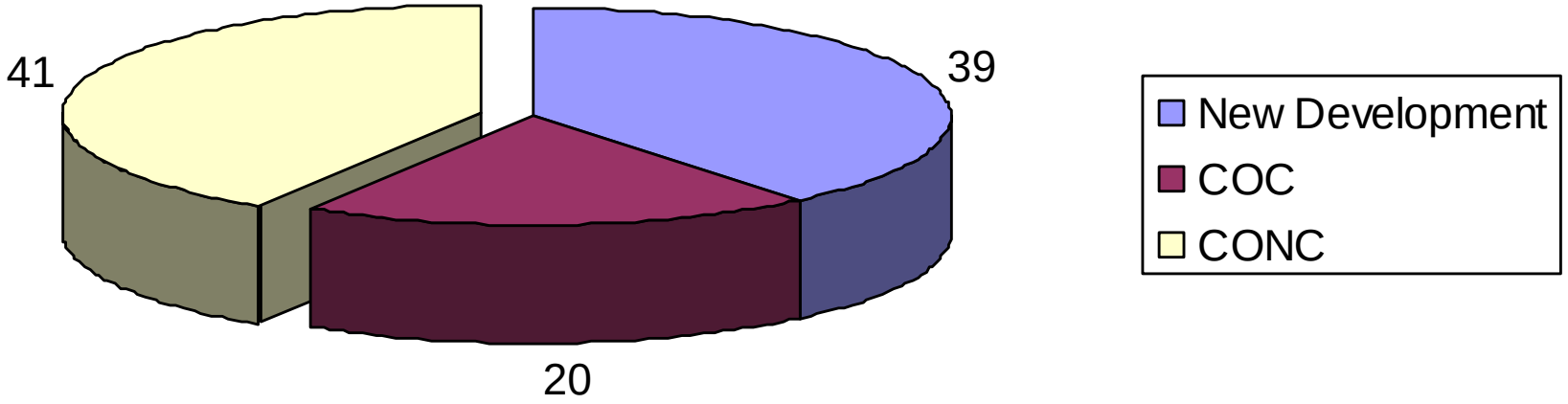
An Early CoSQ Experience

Raytheon

Where are software engineers spending their time?

OR

Where are we spending our software engineering budget?



Source: Raytheon Electronic Systems Experience in Software Process Improvement, CMU/SEI-95-TR-017, November 1995

Successful software process improvement programs can

reduce the number of defects delivered to customers by 95%

reduce software development schedules by 71%

increase productivity (measured in lines-of-code or function points per day) by 222%

realized an average ROI of 5:1

Sources: Capers Jones and Software Engineering Institute

Why Focus on Process?

Process provides a constructive, high-leverage focus...

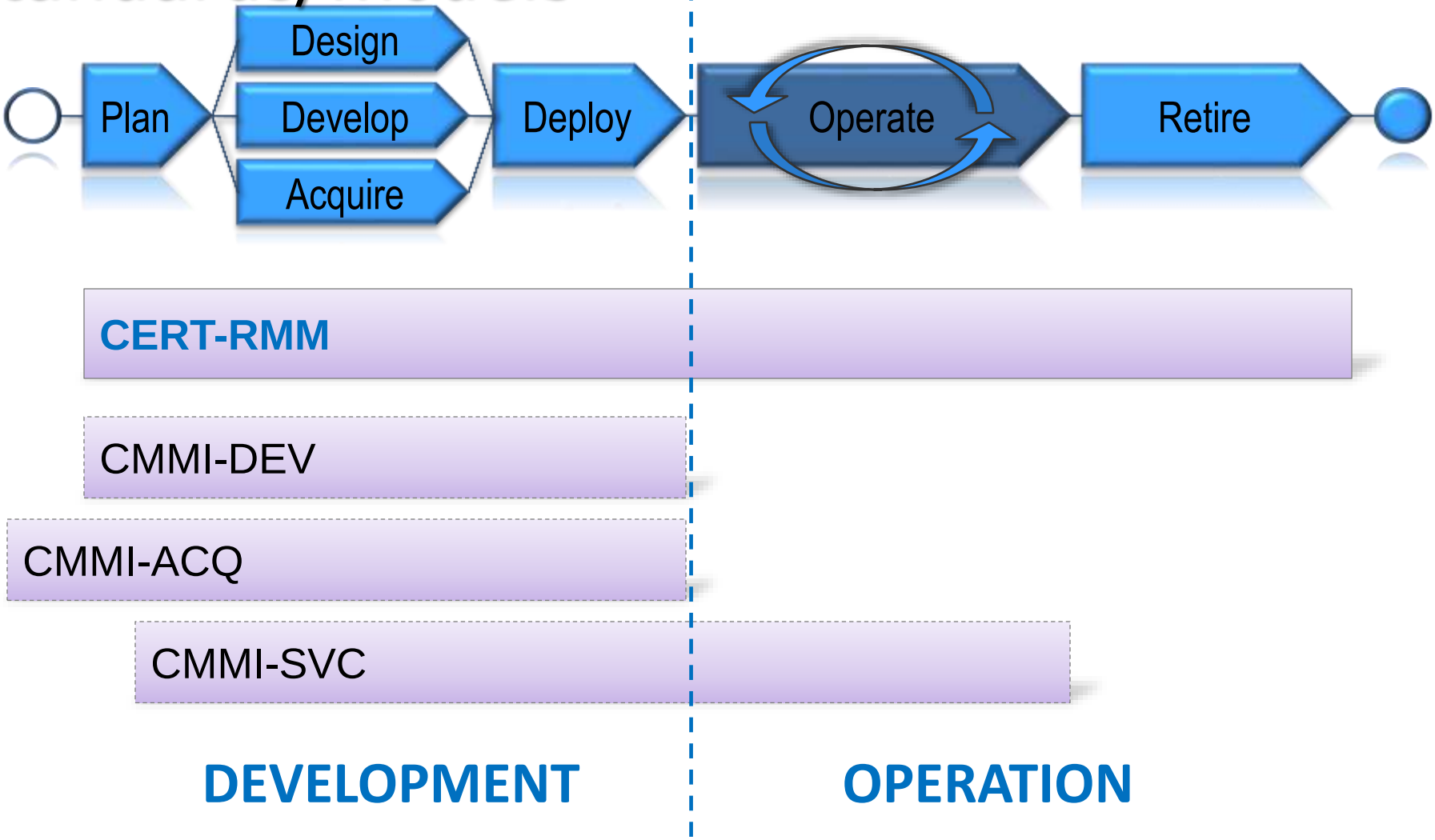
... as opposed to a focus on people

- Your work force, on the average, is as "good" as it is *trained* to be.
- Working harder is not the answer.
- Working smarter, through process, is the answer.

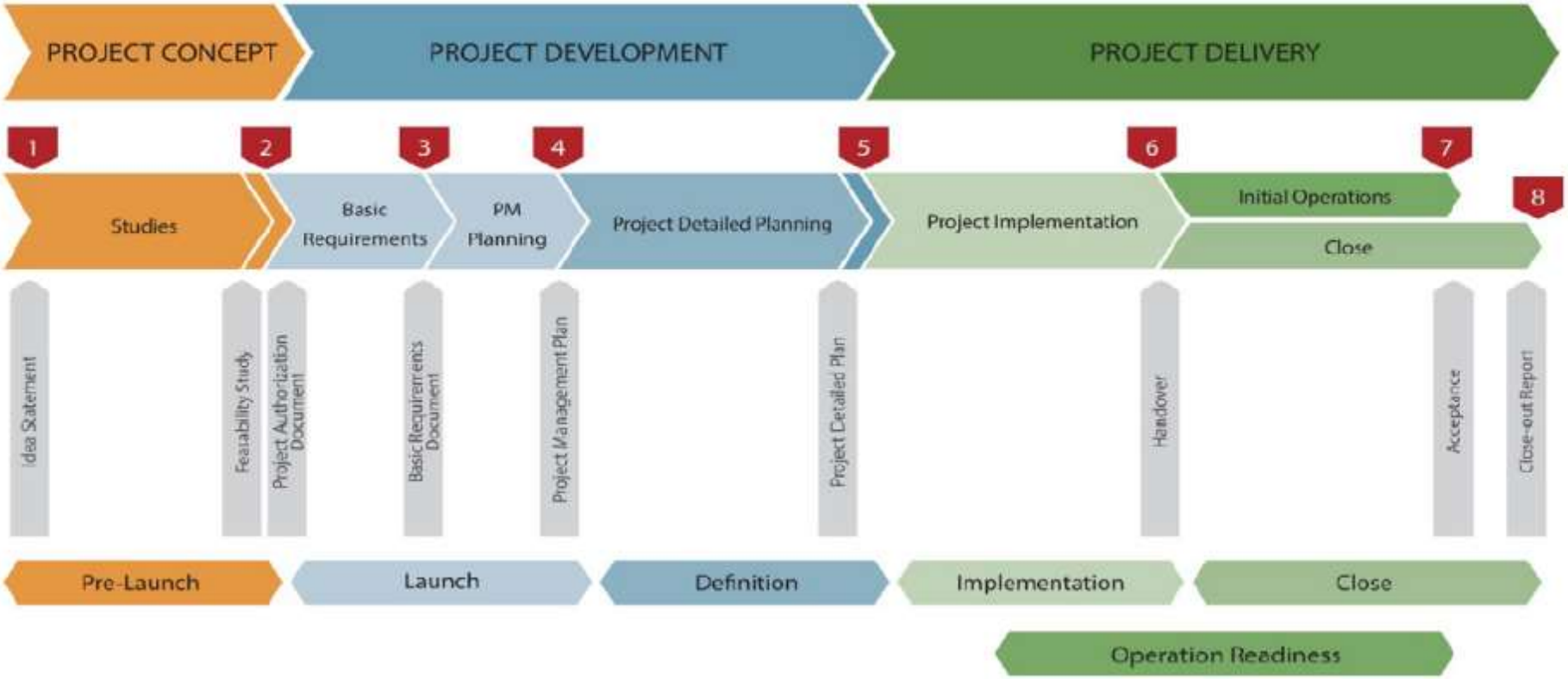
... as opposed to a focus on technology

- Technology applied without a suitable roadmap will not result in significant payoff.
- Technology provides the most benefit in the context of an appropriate process roadmap.

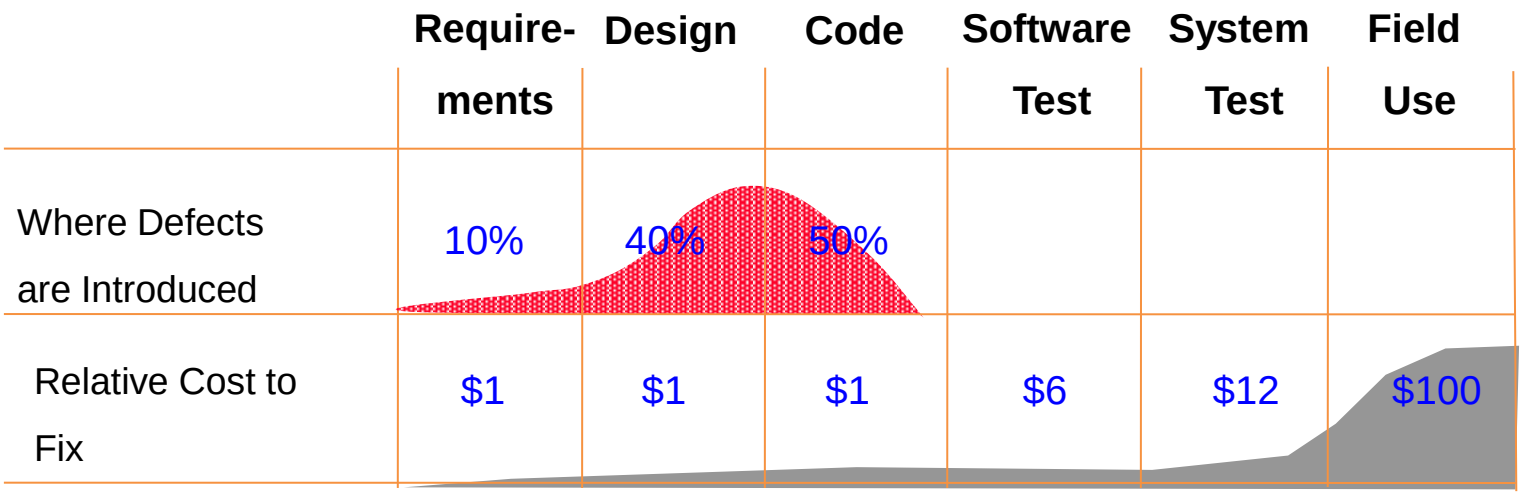
SW life cycle, software (quality) assurance standards/models



SW Project life cycle (detailed)

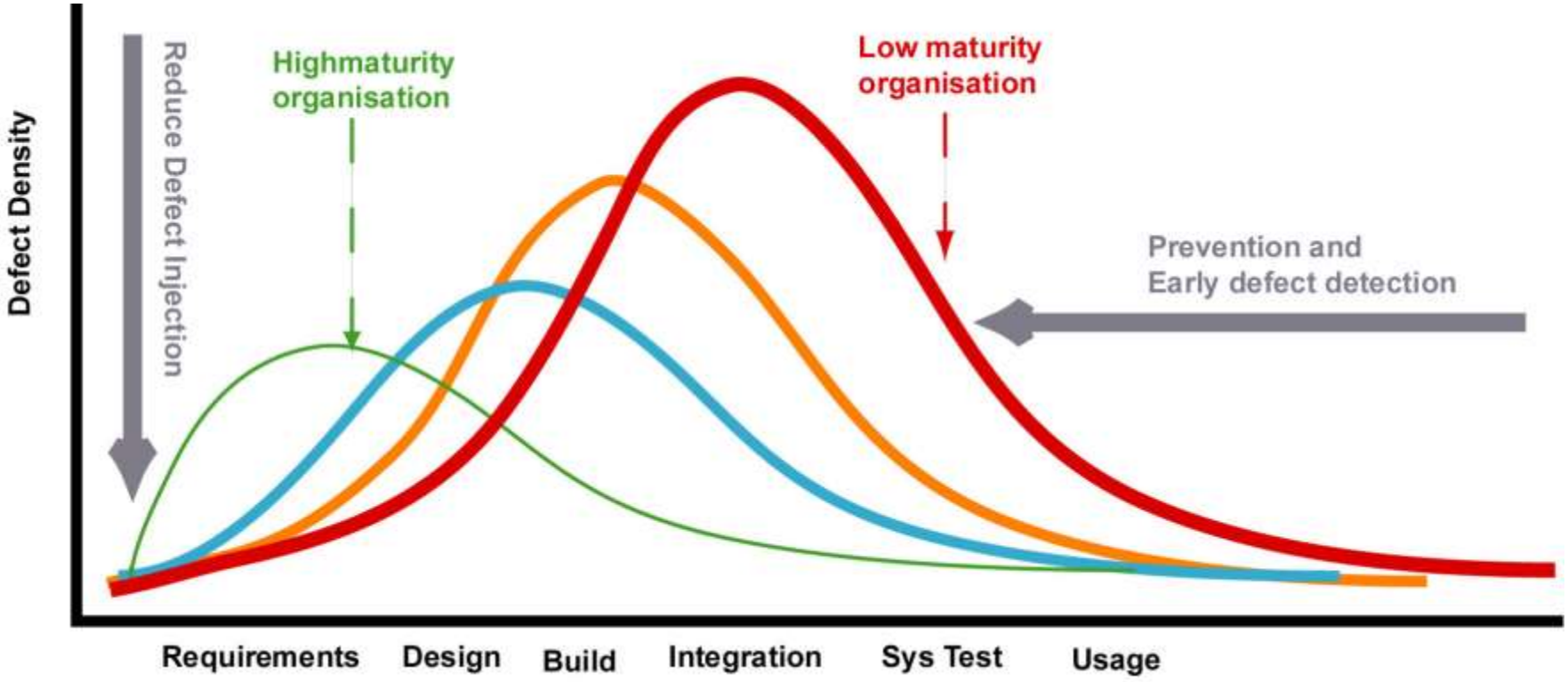


Defects : Insertion Pattern & Cost of Removal



Source: SEPG Asia Pacific 2009
presented by Ravindra Nath, KUGLER MAAG CIE GmbH

Defects-2: Injection & Prevention

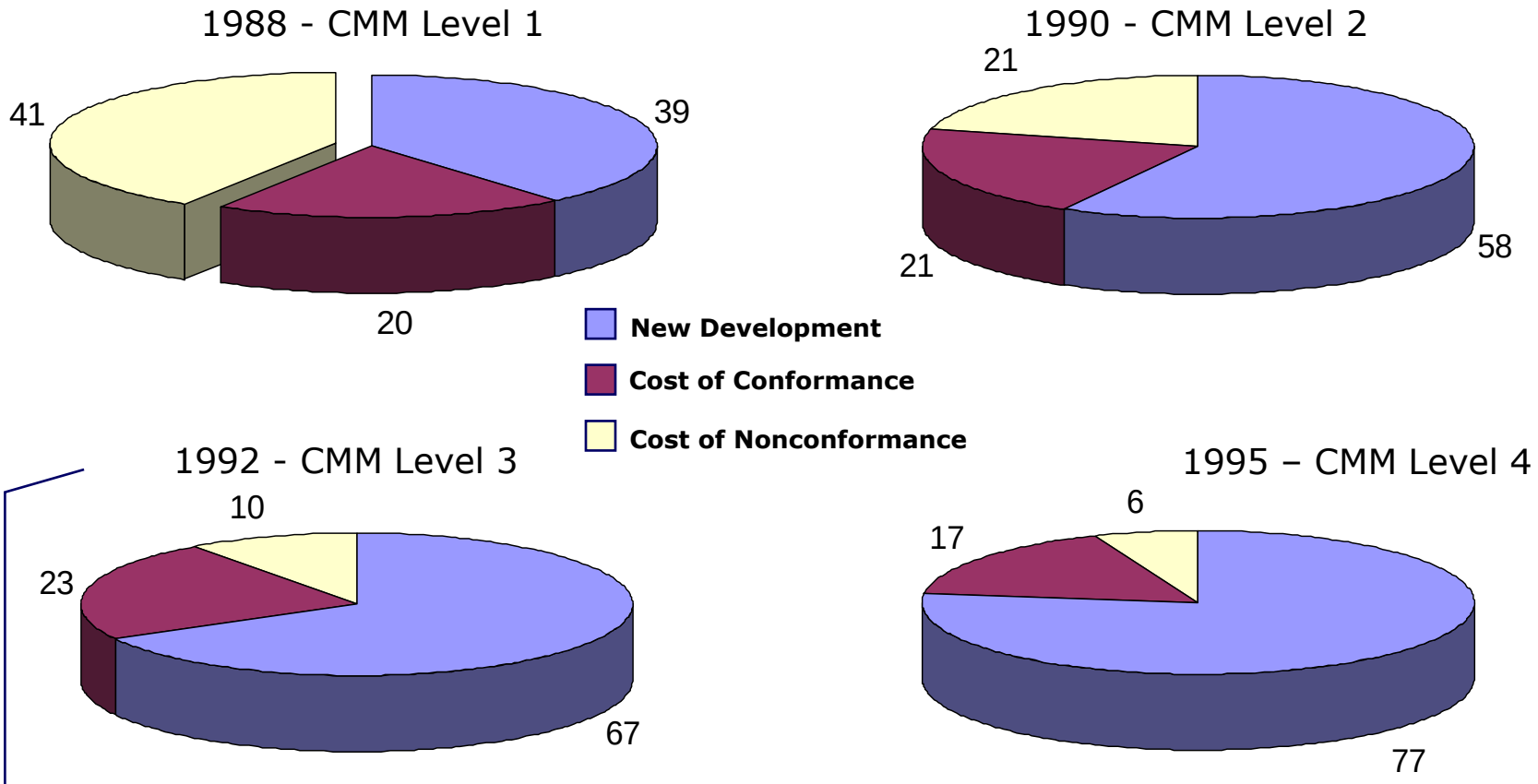


Source: Six Sigma and DFSS for IT and Software Engineering Position Paper
Radouane Oudrhiri, CTO, Systonomy Limited

This is also about SW Quality?

SELECT name FROM users WHERE name='' OR ''='' AND passwd= '' OR ''=''

The shift to increased profitability



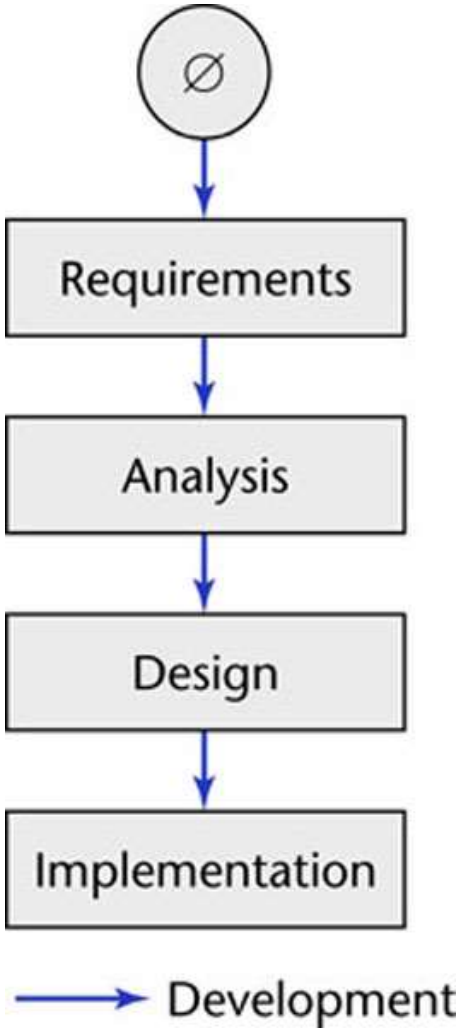
ROI 7.7:1, Productivity ⇧140%, $4.48M savings over 6 projects in 1 year

Source: Raytheon Electronic Systems Experience in Software Process Improvement, CMU/SEI-95-TR-017, November 1995

Software Development in Theory

Ideally, software is developed:

- Linear
- Starting from scratch



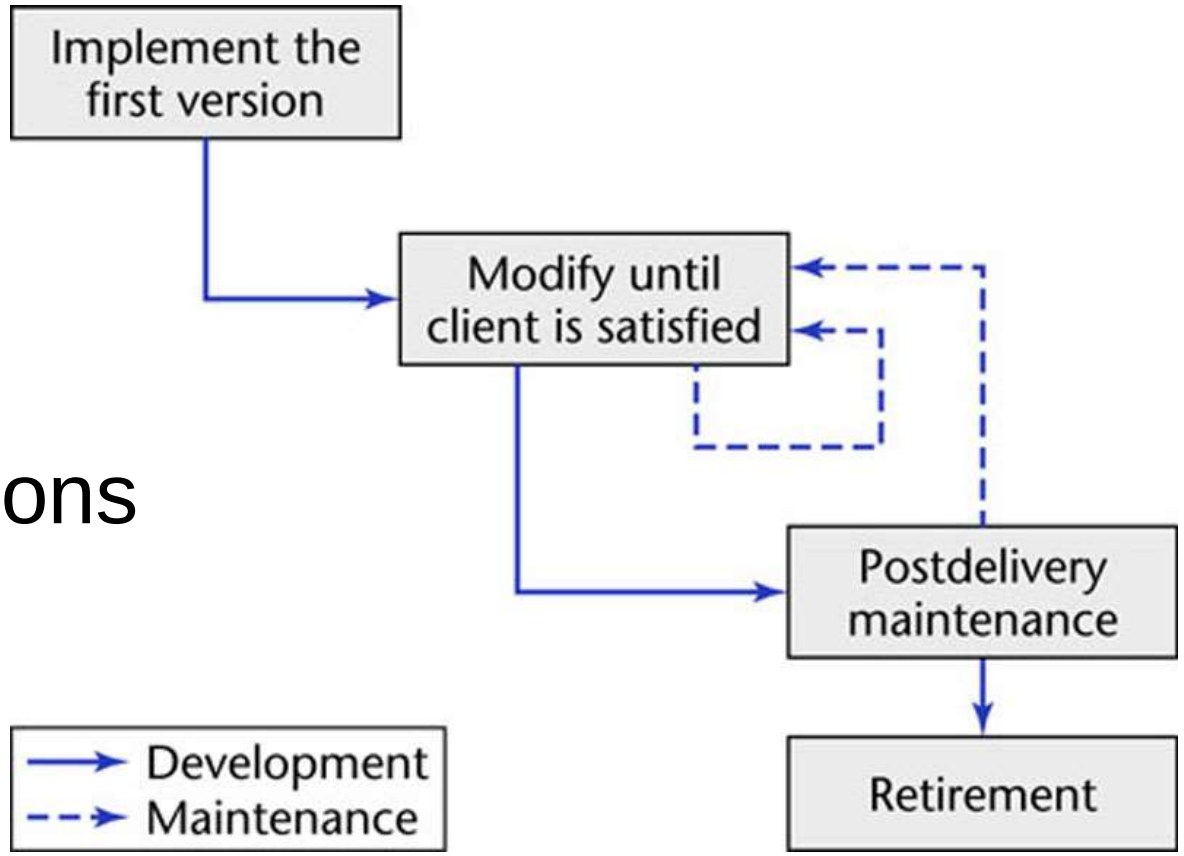
Software Development in Practice

In the real world, software development is totally different and is more chaotic

- Software professionals make mistakes
- The client's requirements change while the software product is being developed
- A software product is a model of the real world, and the real world is continually changing.

Code-and-Fix Life-Cycle Model

- No design
- No specifications



The easiest way to develop software
The most expensive way for maintenance (i.e., maintenance nightmare)

Code-and-Fix Life-Cycle Model (Cont.)

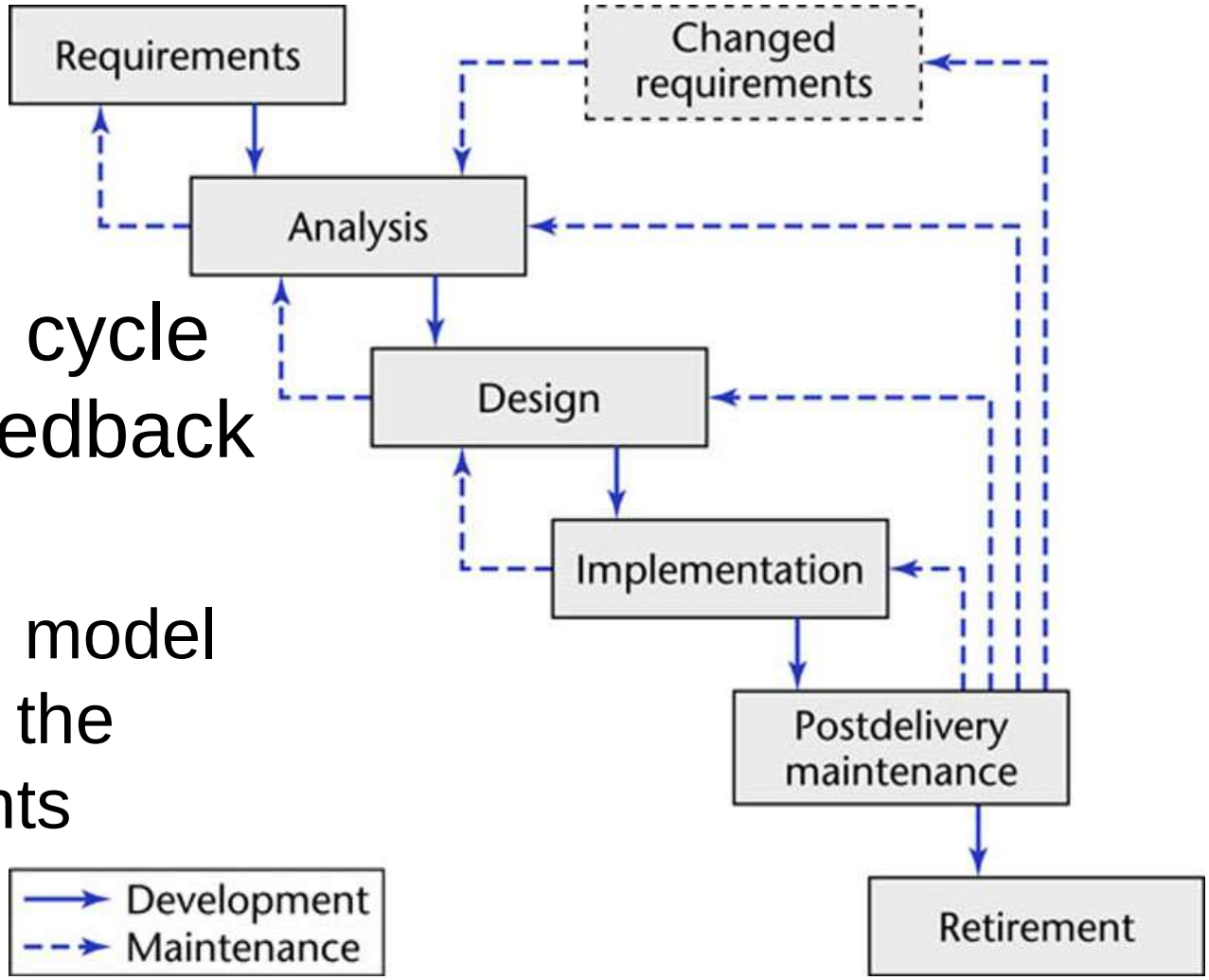
The product is implemented without requirements or specifications, or any attempt at design.

The developers simply throw code together and rework it as many times as necessary to satisfy the client.

It is used in small project and is totally unsatisfactory for products of any reasonable size.

Waterfall Life-Cycle Model

- The linear life cycle model with feedback loops
  - The waterfall model cannot show the order of events



Waterfall Life-Cycle Model (Cont.)

No phase is complete until the **documentation** for that phase has been completed and the products of that phase have been approved by the **software quality assurance** (SQA) group.

If the products of an earlier phase have to be changed as a consequence of following a **feedback loop**, that earlier phase is deemed to be complete only when the documentation for the phase has been modified and the modifications have been checked by the SQA group.

Waterfall Life-Cycle Model (Cont.)

Advantages:

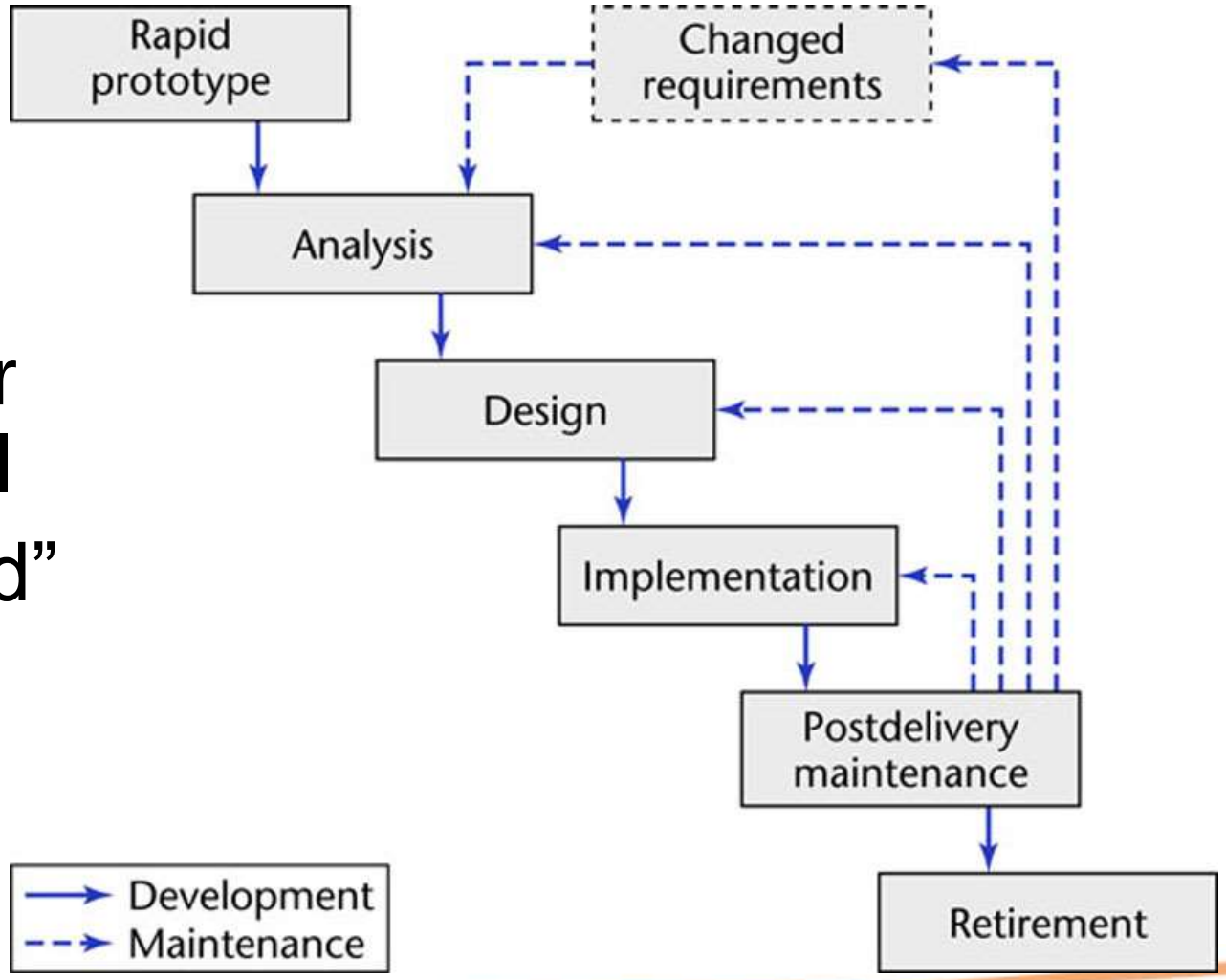
- Documentation is provided at each phase
- All the products of each phase (including the documentation) are meticulously checked by SQA. ➔ Maintenance is easier

Disadvantages:

- Specification documents are long, detailed, and boring to read.

Rapid-Prototyping Life-Cycle Model

- Linear model
- “Rapid”



Rapid-Prototyping Life-Cycle Model (Cont.)

A rapid prototype is a working model that is functionally equivalent to a subset of the product.

The first step is to build a rapid prototype and let the client and future users interact and experiment with the rapid prototype.

Strength:

- The development of the product is essentially **linear**, proceeding from the rapid prototype to the delivered product.
- The feedback loops of the waterfall model are less likely to be needed in the rapid prototyping model.
- It is built rapidly and modified rapidly to reflect the client's needs. → Speed is of the essence.

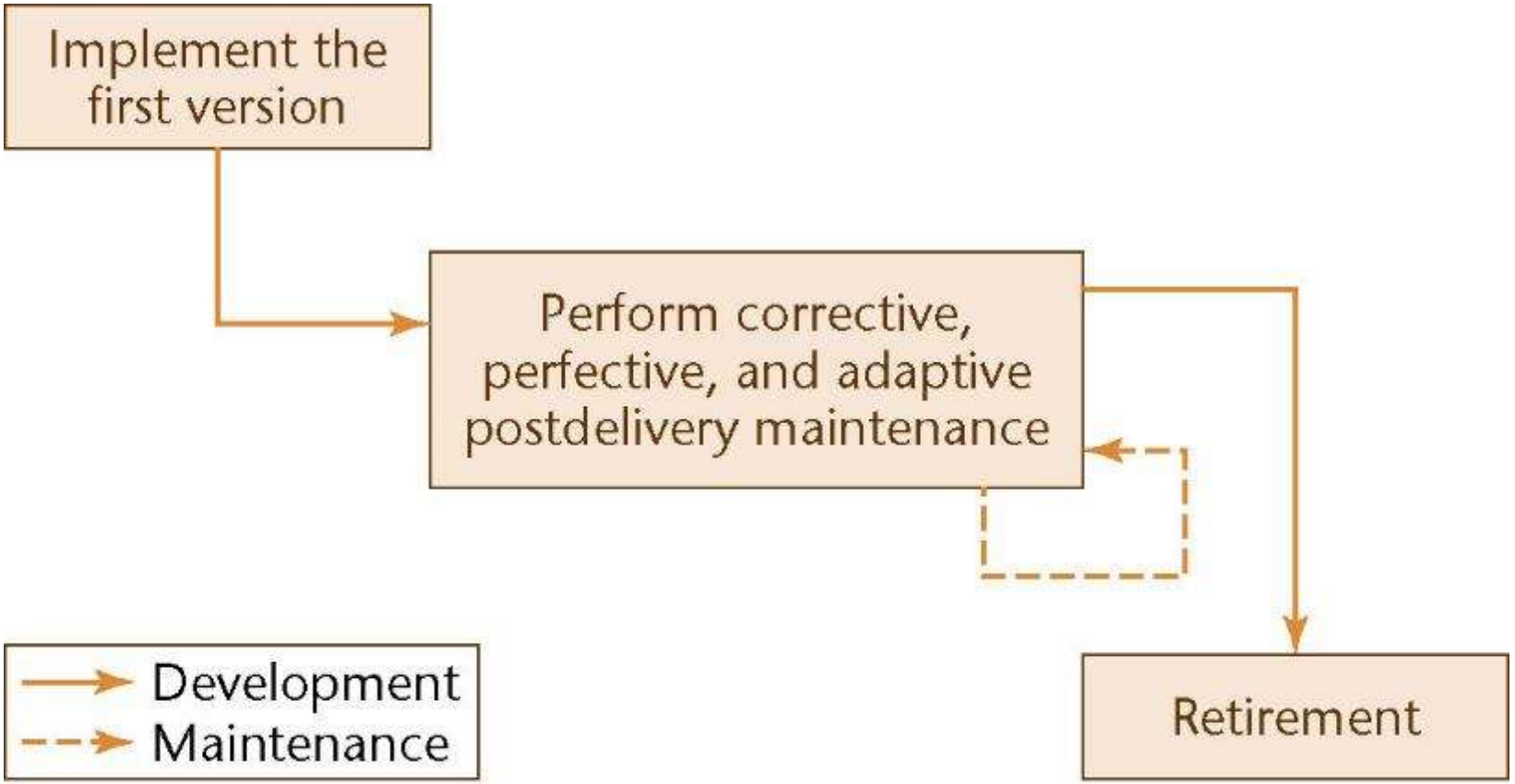
Rapid-Prototyping Life-Cycle Model (Cont.)

Weakness:

- One the client's real needs have been determined, the rapid prototype implementation is discarded.

The lessons learned from the rapid prototype implementation are retained and used in subsequent development phases.

4. Open-Source Life-Cycle Model



Postdelivery maintenance life-cycle model

Open-Source Life-Cycle Model (Cont.)

An initial working version is produced using the rapid-prototyping model, the code-and-fix model, and the open-source life-cycle model.

The initial version of the rapid-prototyping model is then discarded. The initial versions of Code-and-fix model and open-source life-cycle model become the target product

There are generally no specifications and no design. However, open-source software production has attracted some of the world's finest software experts. They can function effectively without specifications or designs

Open-Source Life-Cycle Model (Cont.)

A point will be reached when the open-source product is no longer maintainable

The open-source life-cycle model is restricted in its applicability

- It can be extremely successful for infrastructure projects, such as : Operating systems (Linux, OpenBSD, Mach, Darwin), Web browsers (Firefox, Netscape), Compilers (gcc), Web servers (Apache), and Database management systems (MySQL)
- There cannot be open-source development of a software product to be used in just one commercial organization
- The open-source life-cycle model is inapplicable unless the target product is viewed by a wide range of users as useful

Open-Source vs. Closed-Source

- Closed-source software is maintained and tested by employees
  - Users can submit failure reports but never fault reports
- Open-source software is generally maintained by unpaid volunteers
  - Users are strongly encouraged to submit defect reports, both failure reports and fault reports
    - Core group: Small number of dedicated maintainers with the inclination, the time, and the necessary skills to submit fault reports ("fixes"); They take responsibility for managing the project; They have the authority to install fixes
    - Peripheral group: Users who choose to submit defect reports from time to time

Open-Source vs. Closed-Source (Cont.)

New versions of closed-source software are typically released roughly once a year

- After careful testing by the SQA group

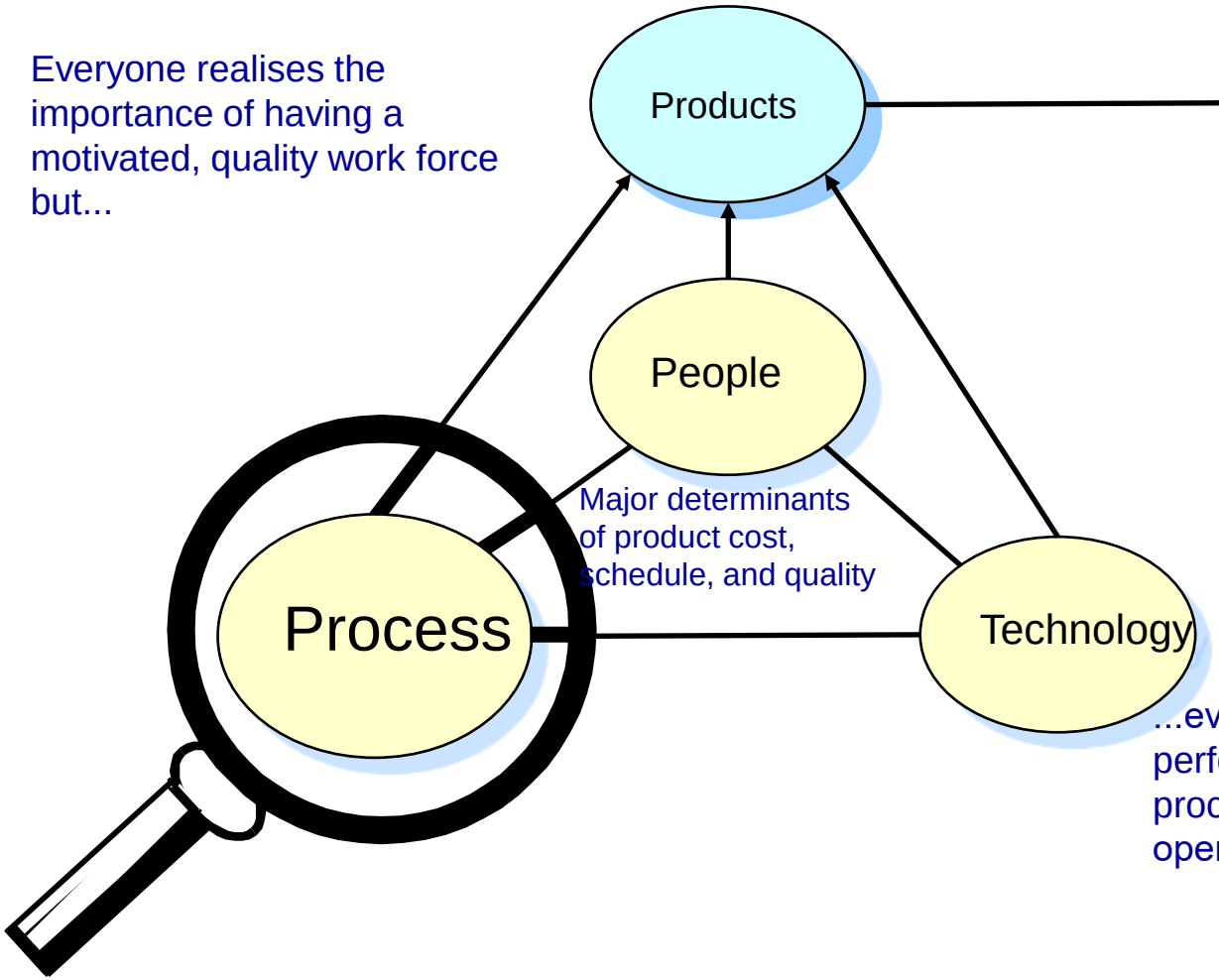
The core group releases a new version of an open-source product as soon as it is ready

- Perhaps a month or even a day after the previous version was released
- The core group performs minimal testing
- Extensive testing is performed by the members of the peripheral group in the course of utilizing the software
- "Release early and often"

Focus on the processes (2)

Quality Leverage Points

Everyone realises the importance of having a motivated, quality work force but...



CUSTOMER SATISFACTION

...even our finest people can't perform at their best when the process is not understood or operating "at its best."

General Definition of Process

- How do you define process?

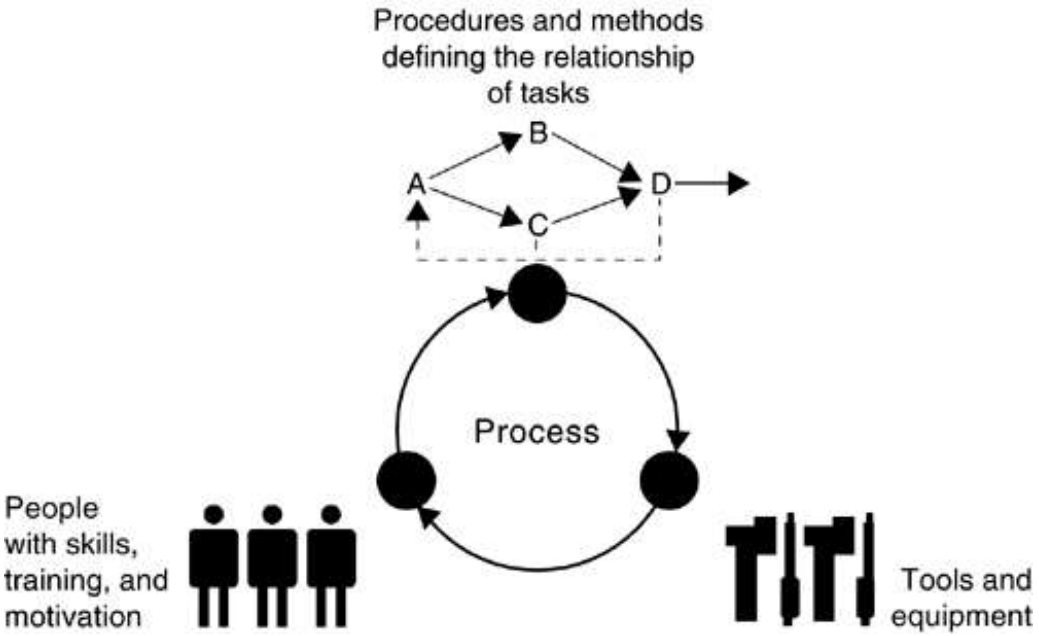
A process is a set of practices performed to achieve a given purpose; it may include tools, methods, materials, and/or people.

While process is often described as a leg of the process-people-technology triad, it may also be considered the "glue" that unifies the other aspects.

Why using models?

"All models are wrong,
but some are useful."

George Box



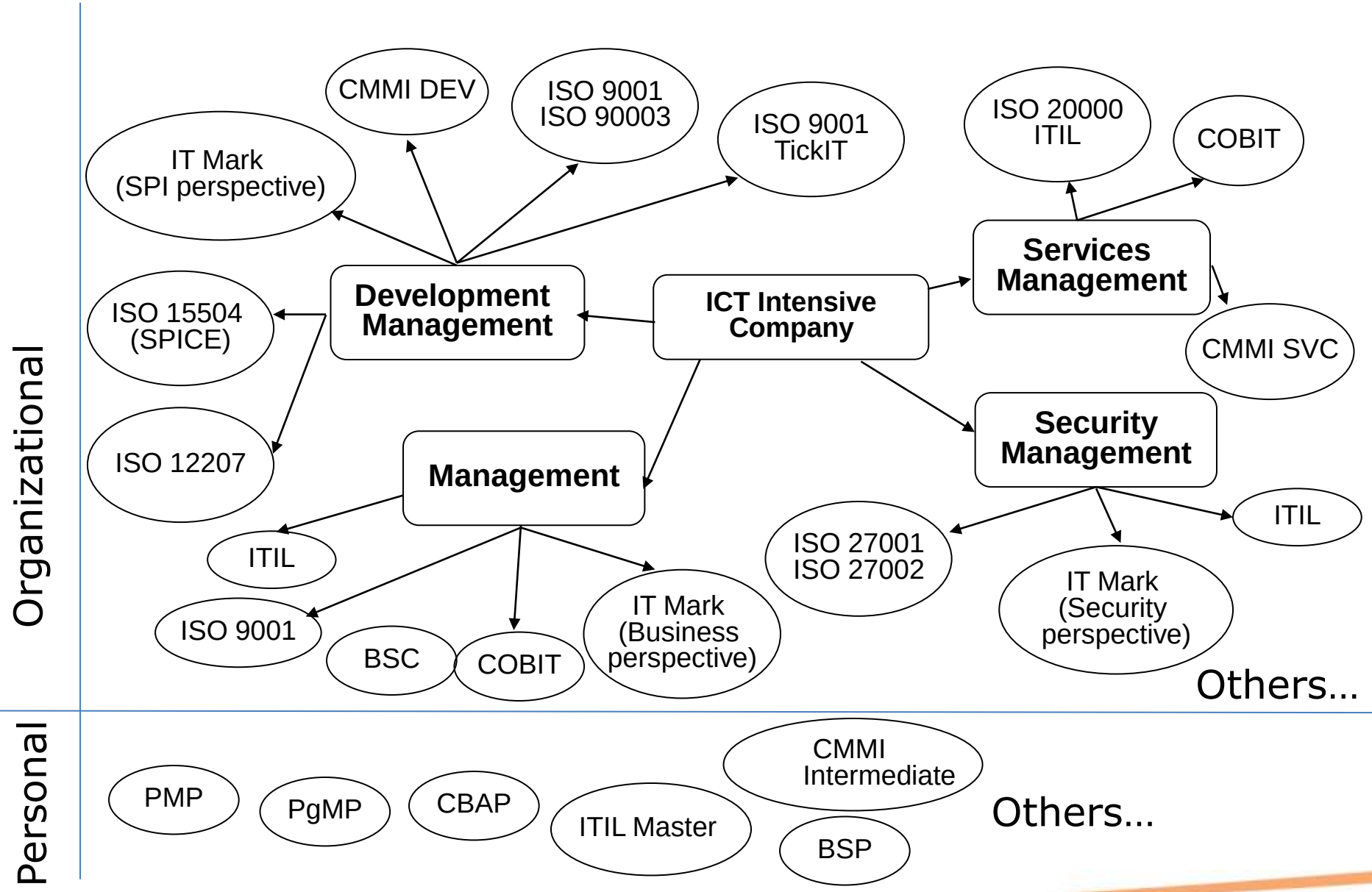
Process ≠ Bureaucracy

Process = Work

Part 2: CMMI model

Модел CMMI (ver 1.3). История, внедряващи организации. Обща структура. Процесни области. Цели и практики. Презентации – Maturity/Capability нива на Continuous и Staged representations. Категории процесни области: Process Management, Project Management, Engineering, Support.

So many models and standards…



Информация, източници:

ESI Center Eastern Europe - Resources:
www.esicenter.bg >> general info and files and links at:
Education > Resources > (Software) Quality Management - CMMI
(+ the links: - model in pdf ver 1.3)

Links to CMMI models (from the new source – CMMI Institute, spin-off of Carnegie Mellon/SEI):
https://cmmiinstitute.com/cmmi/dev - new version 2.0 (paid)
> Access V 1.3 to download CMMI –DEV v 1.3 model *(free, upon registration)*

old SEI repository – VALID for FREE DOWNLOAD:
https://resources.sei.cmu.edu/asset_files/TechnicalReport/2010_005_001_15287.pdf

https://en.wikipedia.org/wiki/Capability_Maturity_Model_Integration

General sources (Software Engineering, Quality)

www.sei.cmu.edu
http://resources.sei.cmu.edu/library/
www.cmmiinstitute.com

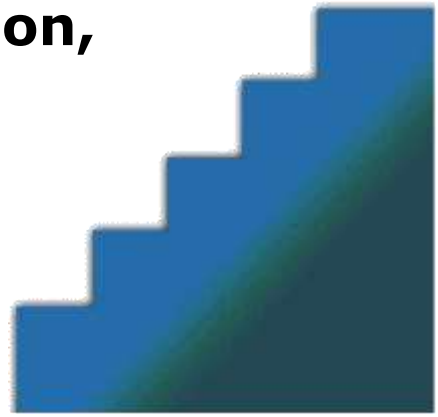
What is a Capability Maturity Model?

Capability Maturity Model:

A reference model of mature practices in a specified discipline, used to assess a group's capability to perform that discipline

CMMs differ by

- **Discipline (software, systems, acquisition, etc.)**
- **Structure (staged versus continuous)**
- **How Maturity is Defined (process improvement path)**
- **How Capability is Defined (institutionalisation)**



"Capability Maturity Model®" and CMM® are used by the Software Engineering Institute (SEI) to denote a particular class of maturity models

History of CMMs

CMM for Software V1.1 (1993)

INCOSE SECAM (1996)

Systems Engineering CMM V1.1 (1995)

Software CMM V2, draft C (1997)

EIA 731 SECM (1998)

Integrated Product Development CMM (1997)

Software Acquisition CMM V1.03 (2002)

V1.02 (2000)

V1.1 (2002)

CMMI for Development V1.2 (2006)

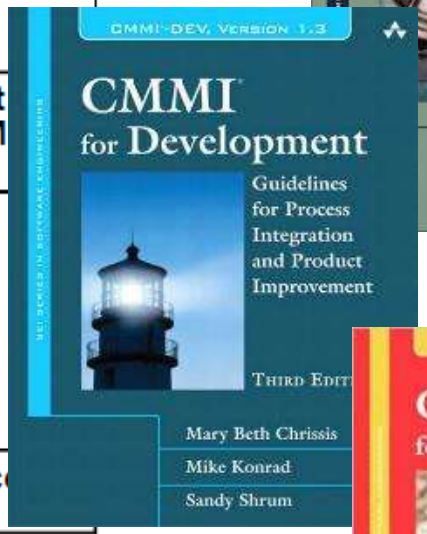
CMMI for Acquisition V1.2 (2007)

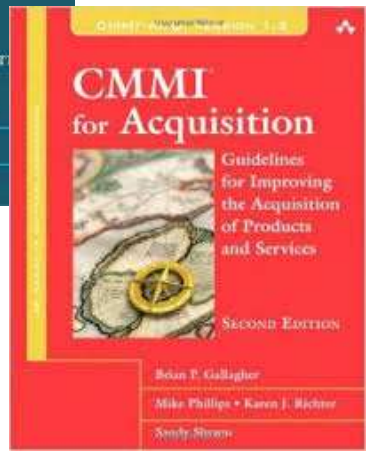
CMMI for Servic V1.2 (2009)

CMMI for Acquisition V1.3 (2010)

CMMI for Development V1.3 (2010)

CMMI for Services V1.3 (2010)





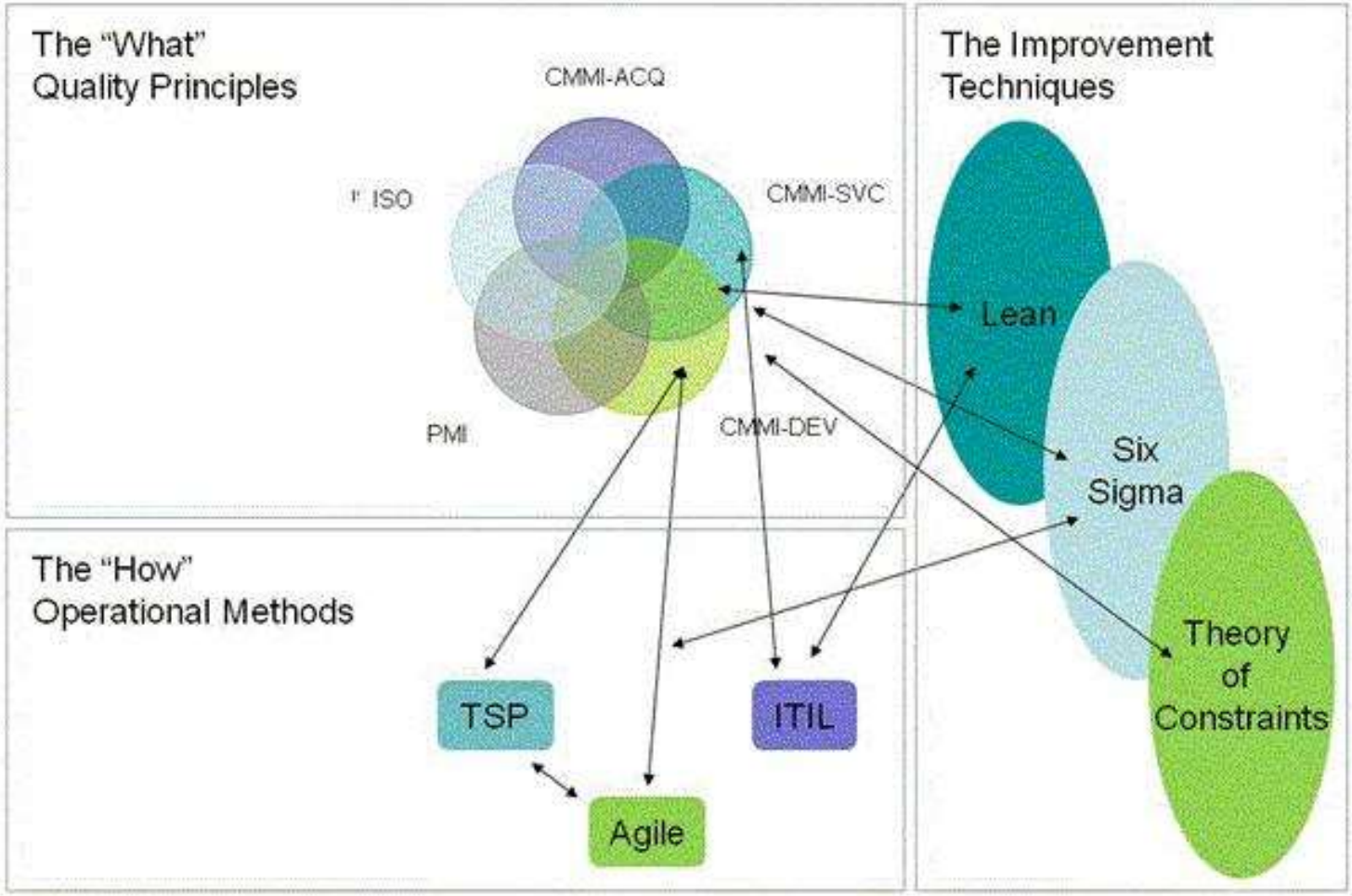
Software Engineering Institute | Carnegie Mellon

CMMI Institute

Founded 2012

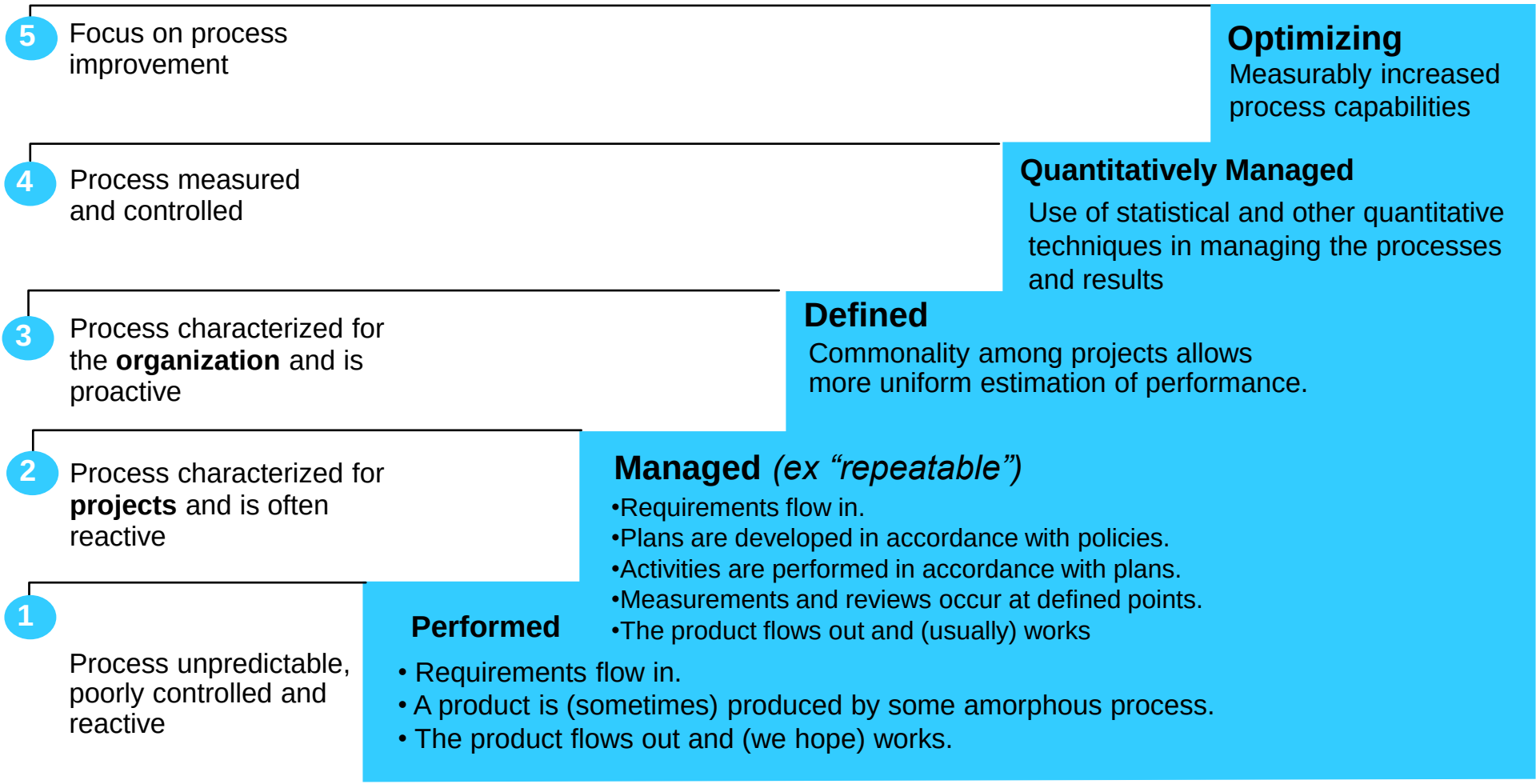
Acquired by ISACA (March 2016)

CMMI and other models



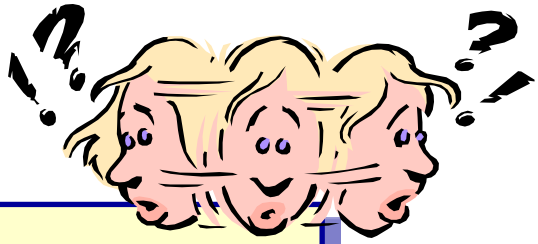
CMMI – reference model & de facto industrial standard

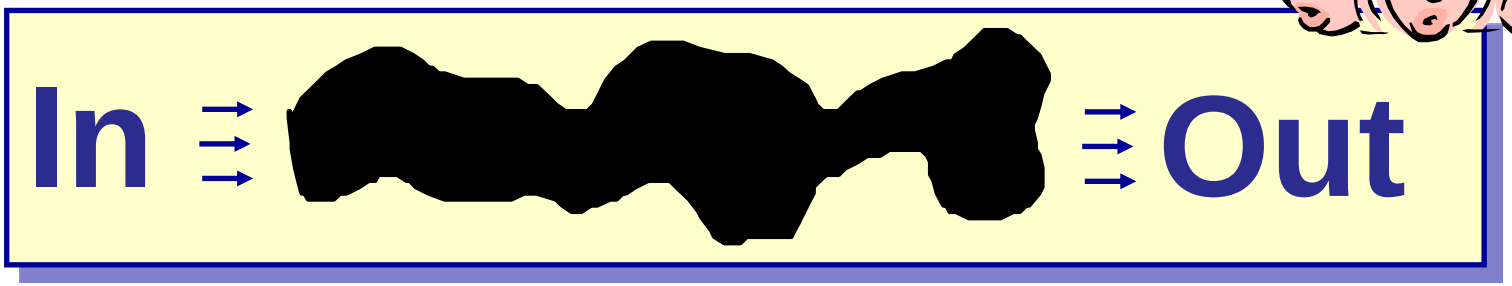
Maturity Levels (ML 1-5) - Staged Representation



CMMI DEV, CMMI ACQ, CMMI SVC

ML1: Performance Is Unpredictable





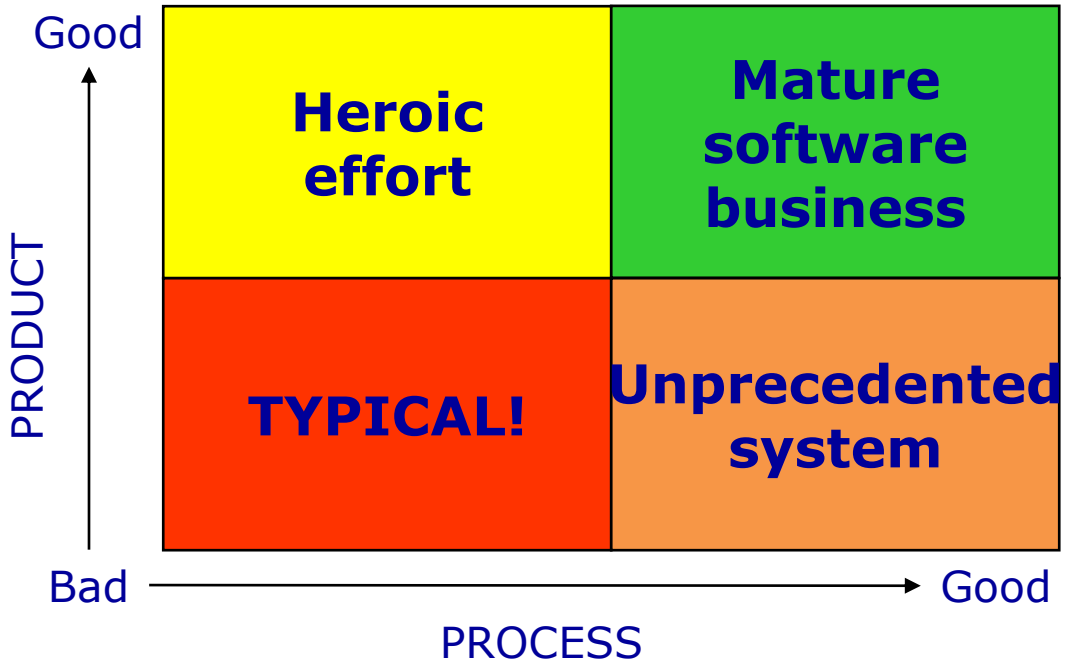
Requirements flow in.

A product is (sometimes) produced by some amorphous process.

The product flows out and (we hope) works.

REMEMBER? Corporate excellence – INTERNAL

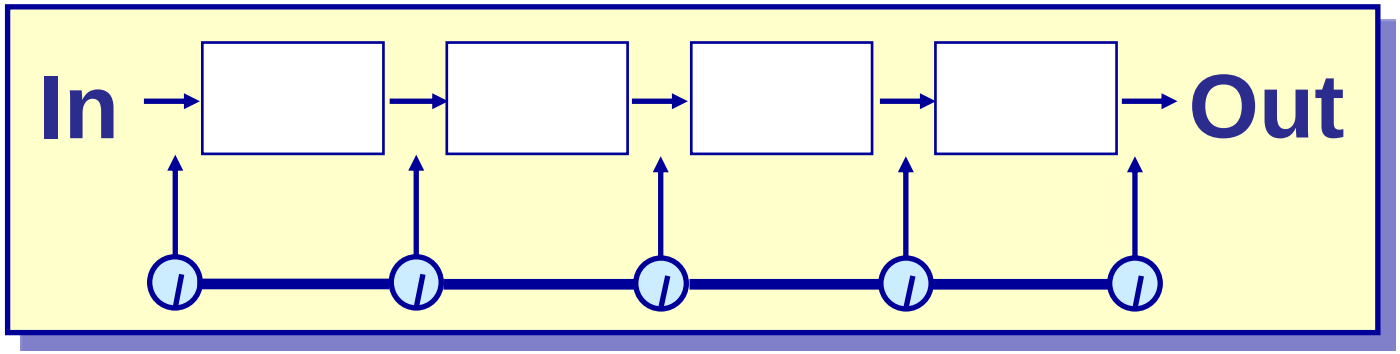
The corporate excellence is BASED on good internal processes



"The quality of a product is largely determined by the quality of the process that is used to develop and maintain it."

Based on TQM principles as taught by Shewhart, Juran, Deming and Humphrey.

ML2: Process Is "Managed"



Requirements flow in.

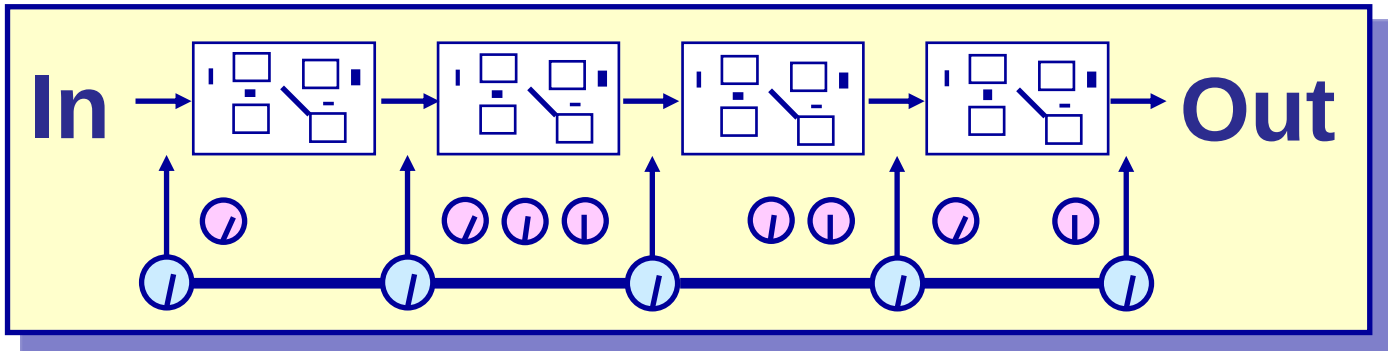
Plans are developed in accordance with policies.

Activities are performed in accordance with plans.

Measurements and reviews occur at defined points.

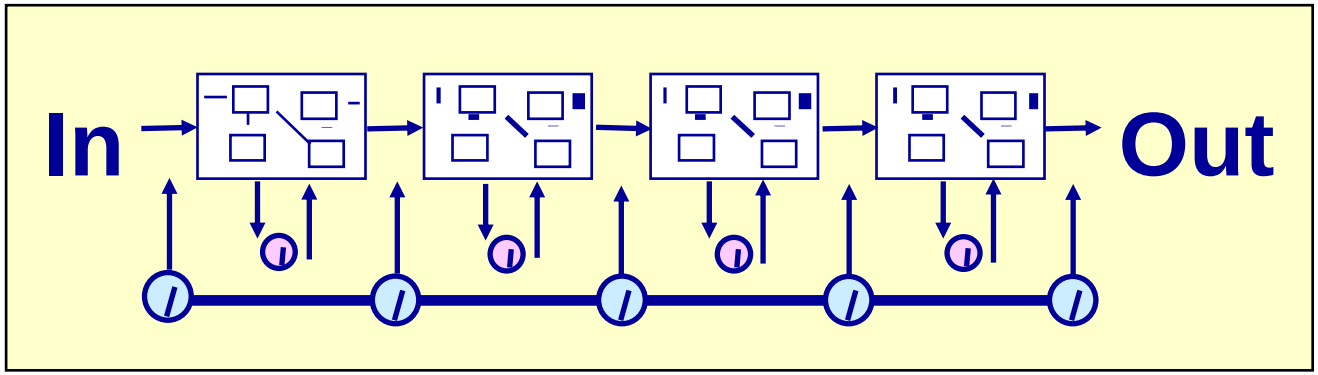
The product flows out and (usually) works.

ML3: Managed According to a Defined Process



Commonality among projects allows more uniform estimation of performance.

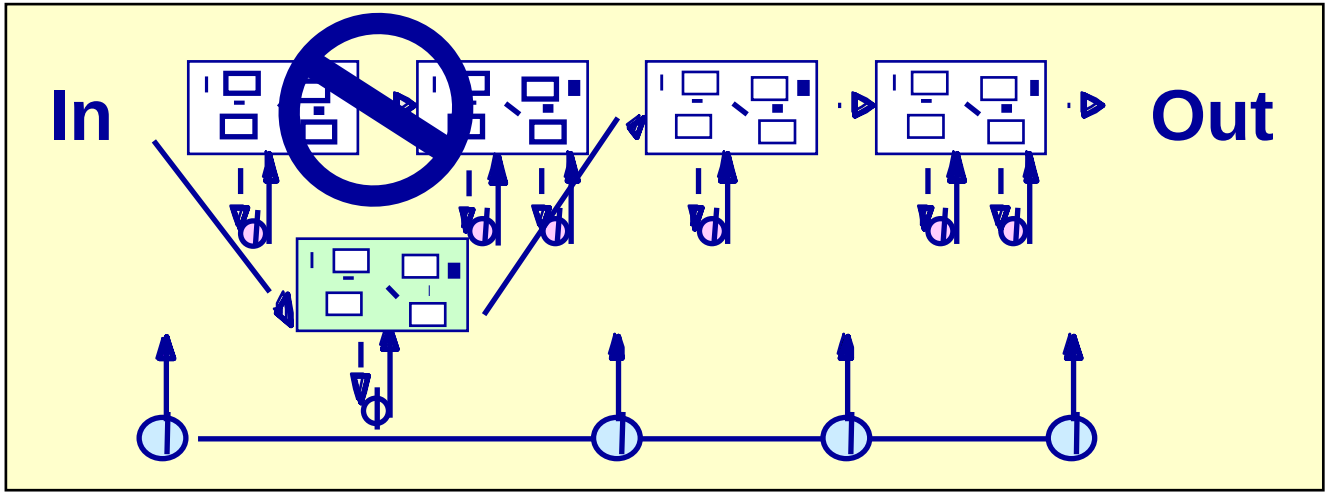
ML4: Quantitatively Managed Process



The process performance is predictable and quantitatively understood

There is a quantitative-based decision making that permits to achieve the established processes objectives, the quality of the product and the quality of the service.

ML5: Optimizing Processes



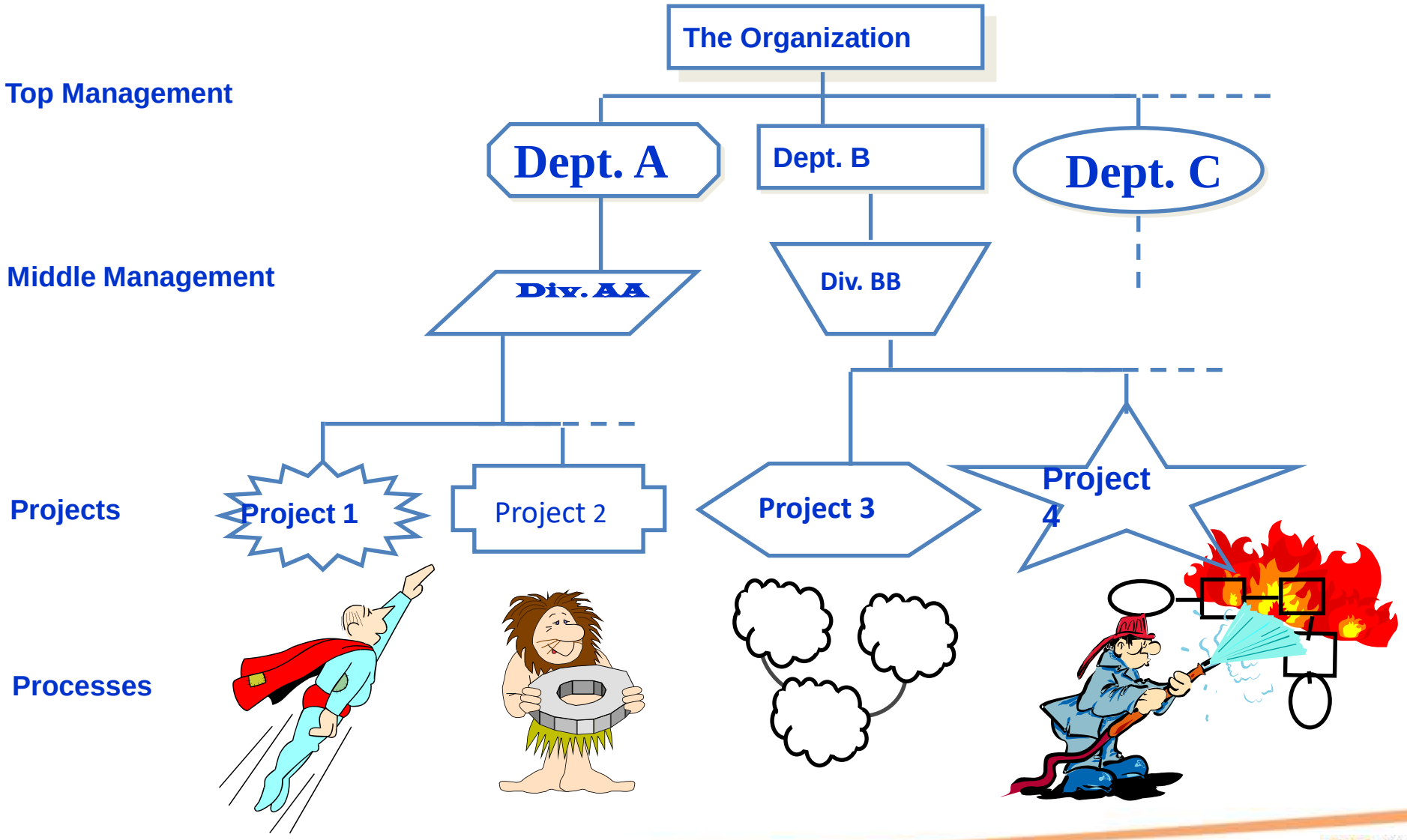
Measurable and continuous process improvement (while the process stability is managed) is integrated in the daily work

Measures are used to:

- Select improvements and innovations
- estimate the costs and benefits of the improvements and innovations
- Measure the current costs and benefits of the improvements and innovations.

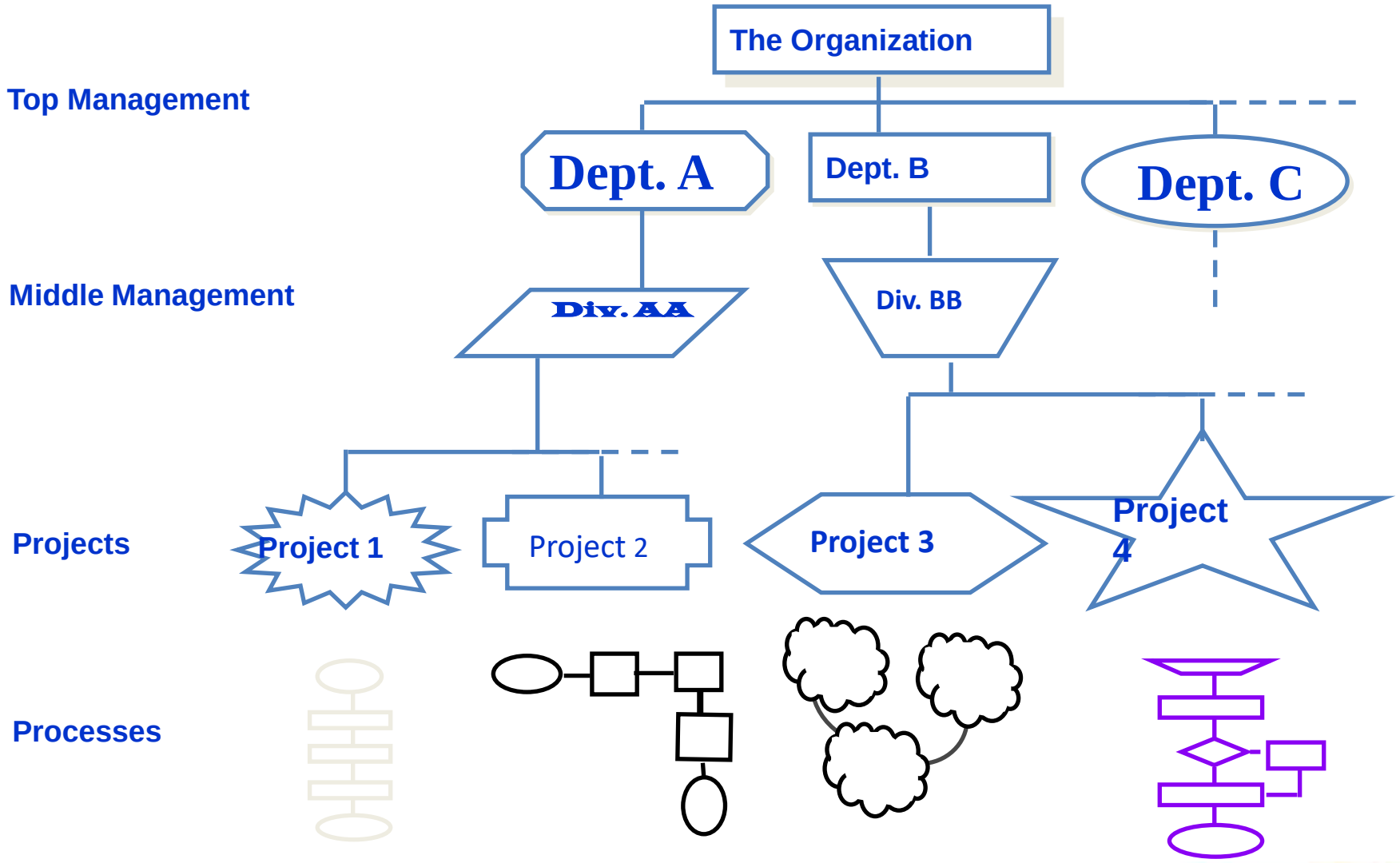
Sample Level 1 Organization

few processes in place



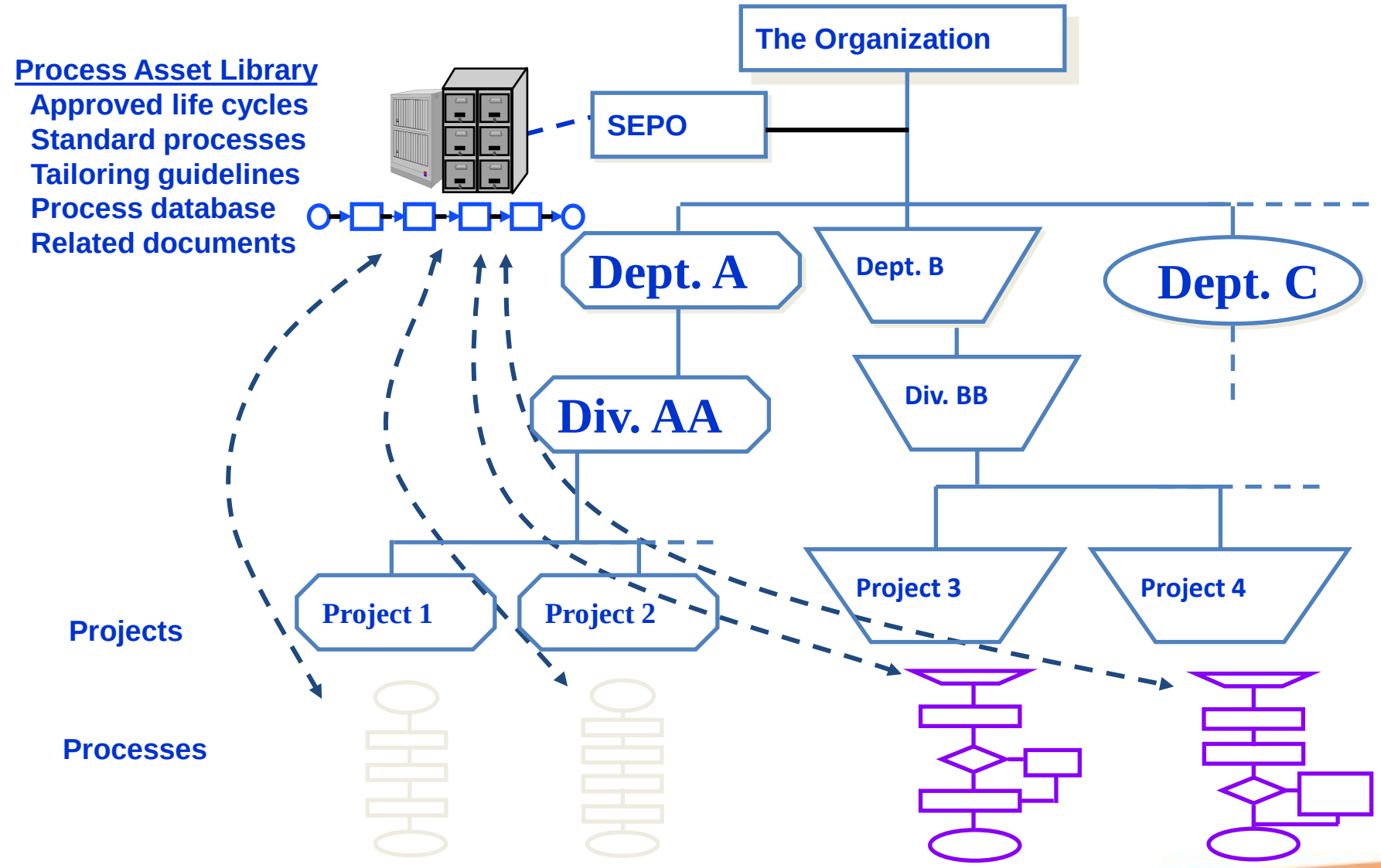
Sample Level 2 Organization

many processes in place; but they are project-specific



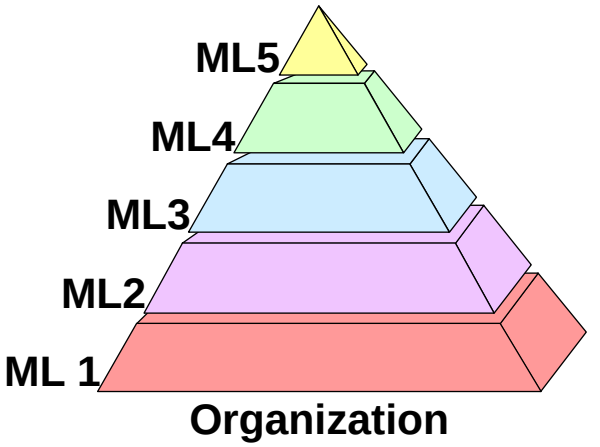
Sample Level 3 Organization

processes based on organization's Process Asset Library (PAL)



CMMI Representations

Staged



Maturity Level 5 OID, CAR

Maturity Level 4 OPP, QPM

Maturity Level 3 RM, TS, PI, VER, VAL, OPF, OPD, OT, IPM, RSKM, DAR

Maturity Level 2 REQM, PP, PMC, MA, PPQA, CM, SAM

Process Areas

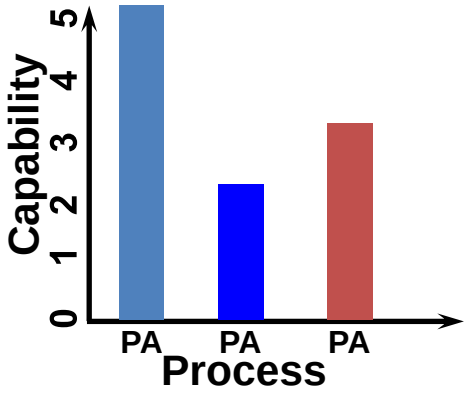
Organizational Innovation & Deployment (OID)
Causal Analysis and Resolution (CAR)

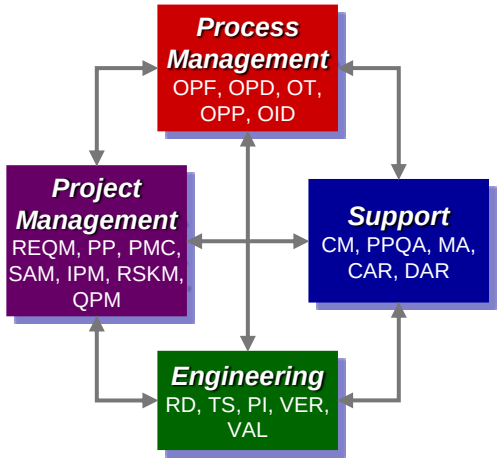
Organizational Process Performance (OPP)
Quantitative Project Management (QPM)

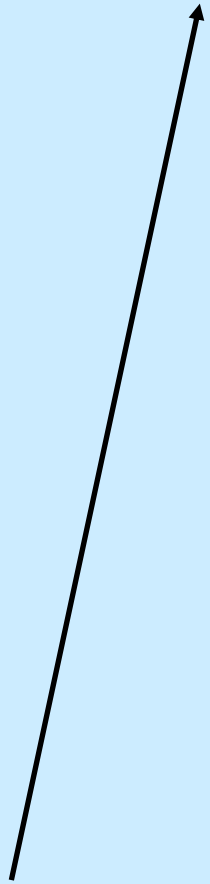
Requirements Development (RD)
Technical Solution (TS)
Product Integration (PI)
Verification (VER)
Validation (VAL)
Organizational Process Focus (OPF)
Organizational Process Definition (OPD) + IPPD
Organizational Training (OT)
Integrated Project Management (IPM) + IPPD
Risk Management (RSKM)
Decision Analysis and Resolution (DAR)

Requirements Management (REQM)
Project Planning (PP)
Project Monitoring and Control (PMC)
Supplier Agreement Management (SAM)
Measurement and Analysis (MA)
Process and Product Quality Assurance (PPQA)
Configuration Management (CM)

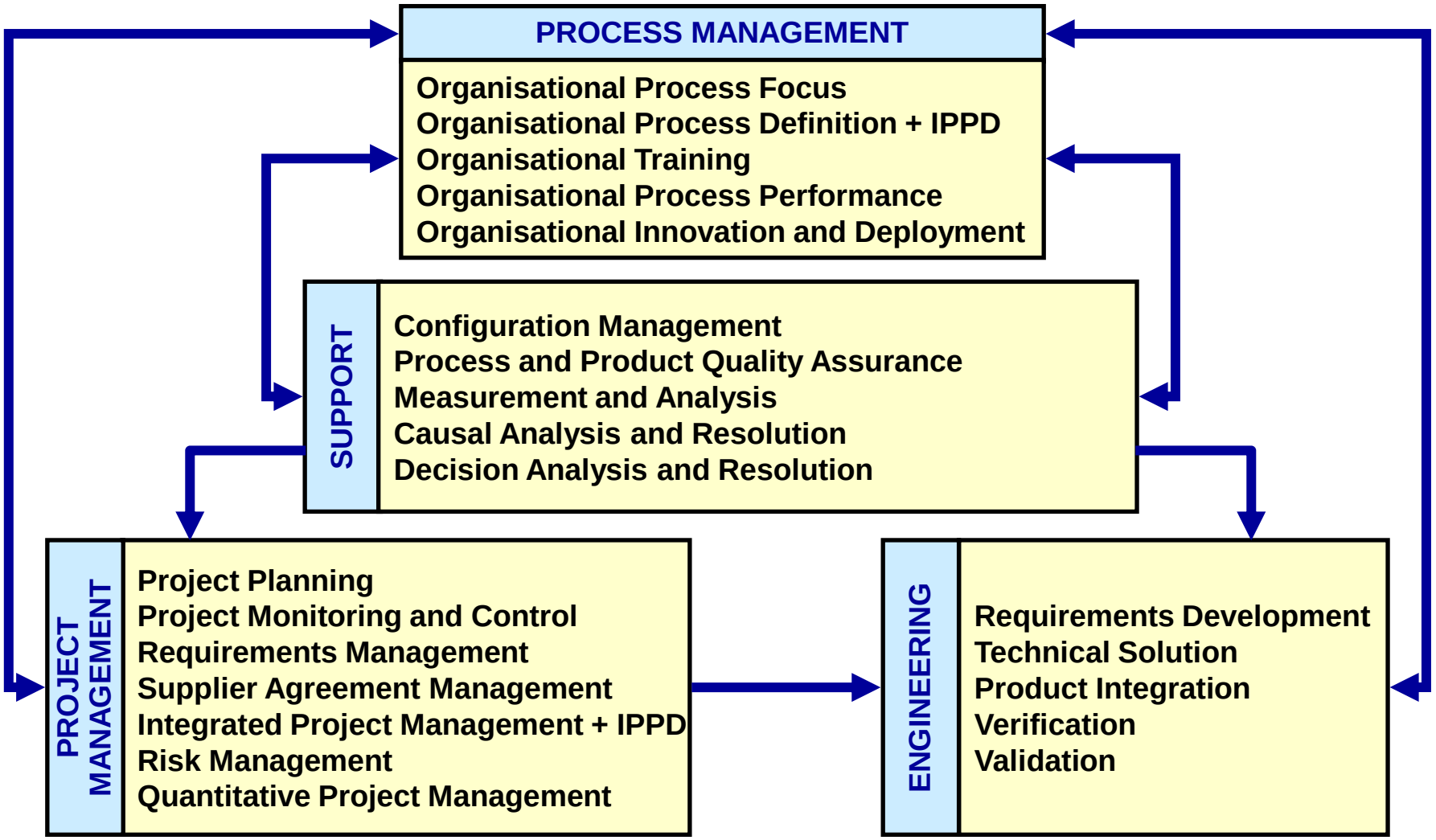
Continuous



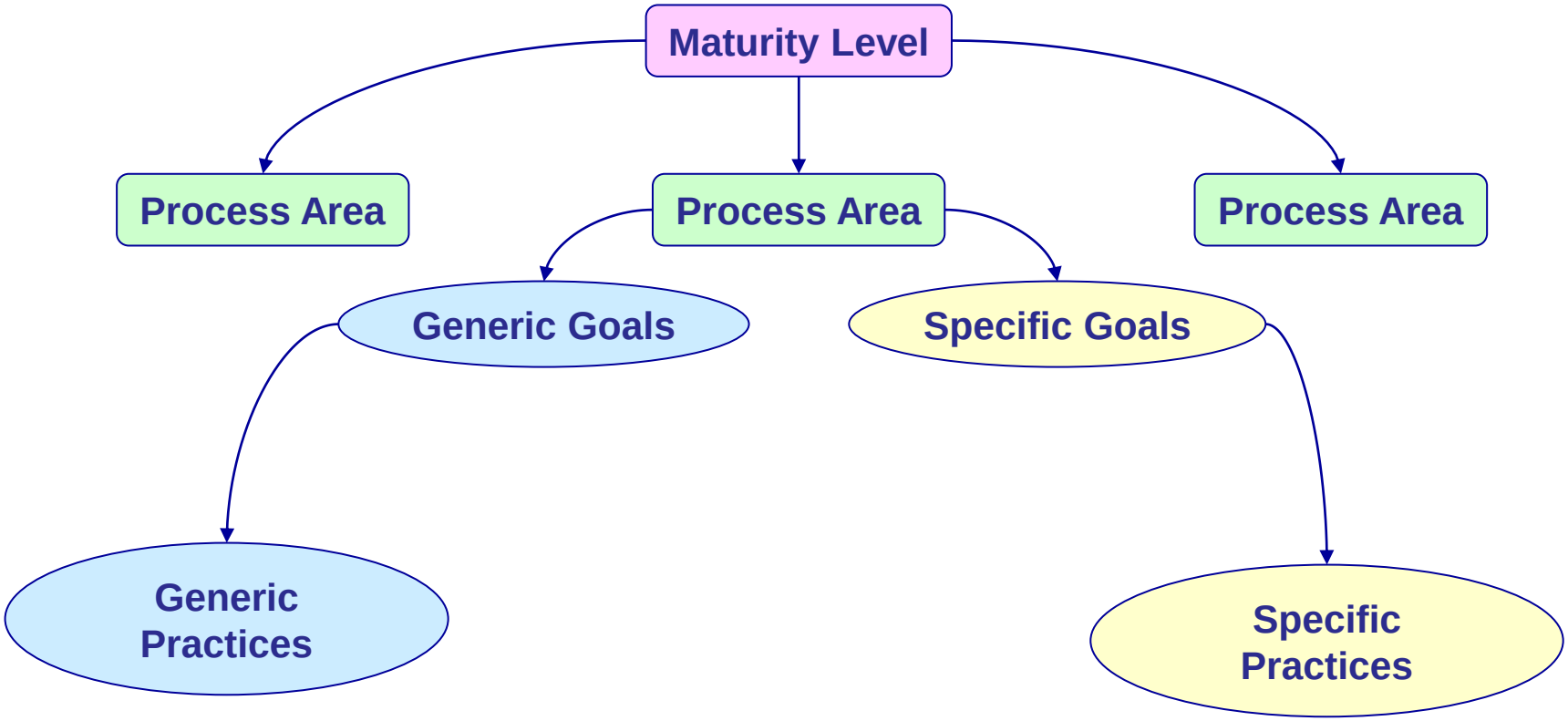


LEVEL	FOCUS	PROCESS AREAS	Quality Productivity  Risk Rework
5 Optimising	Continuous Process Improvement	Organisational Innovation and Deployment Causal Analysis and Resolution	
4 Quantitatively Managed	Quantitative Management	Organisational Process Performance Quantitative Project Management	
3 Defined	Process Standardisation	Requirements Development Technical Solution Product Integration Verification Validation Organisational Process Focus Organisational Process Definition Organisational Training Integrated Project Management Risk Management Decision Analysis and Resolution	
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	
1 Initial	No process areas – the work just gets done somehow!		

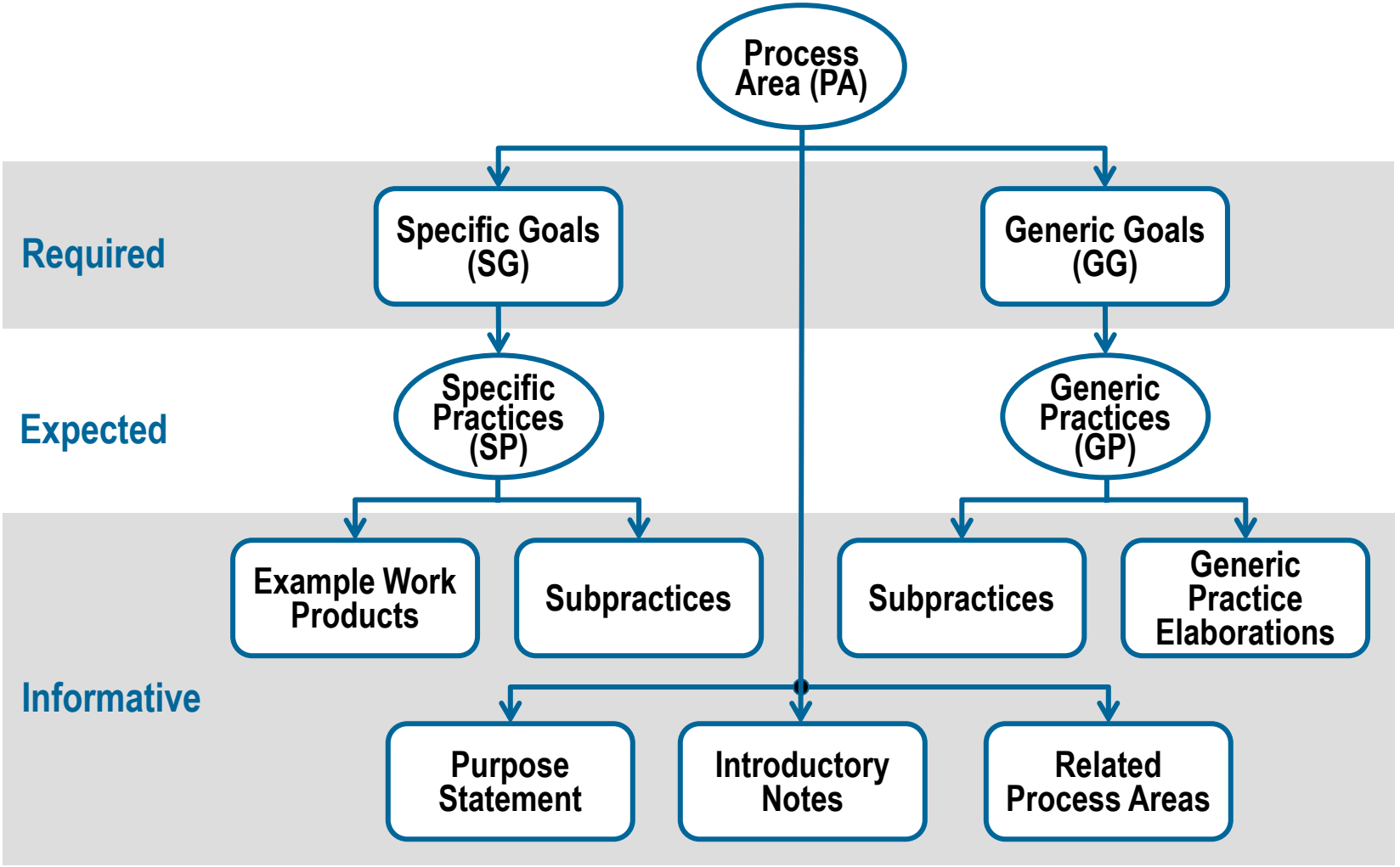
Process areas categories (v 1.3)



Structure of the CMMI Staged Representation



Process Area Components (or how to read the book)



Example
Requirements Management (REQM)
Specific Practices

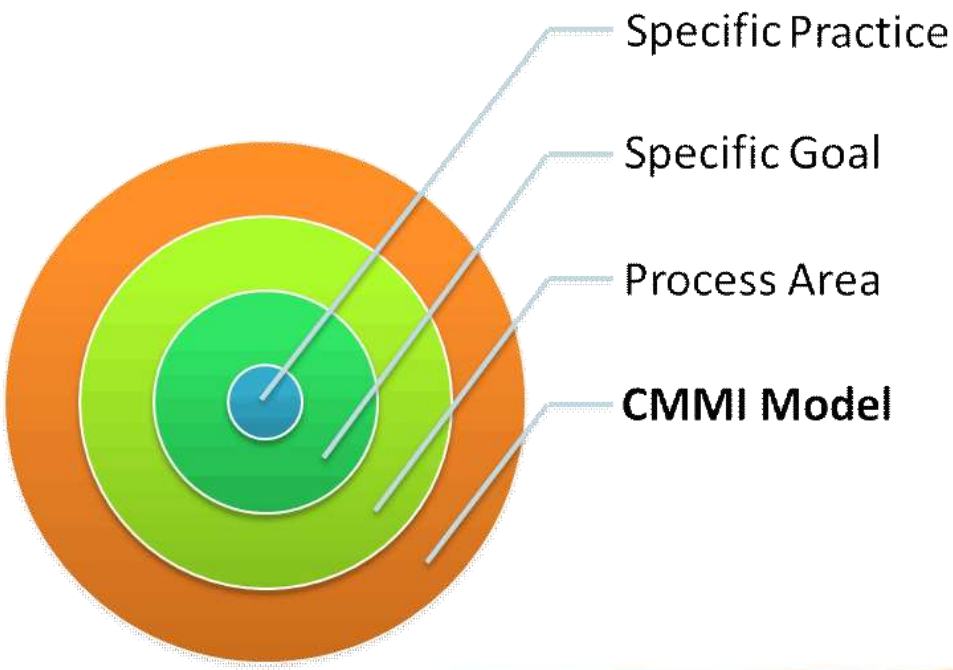
SP 1.1 Obtain an Understanding of Requirements

SP 1.2 Obtain Commitment to Requirements

SP 1.3 Manage Requirements Changes

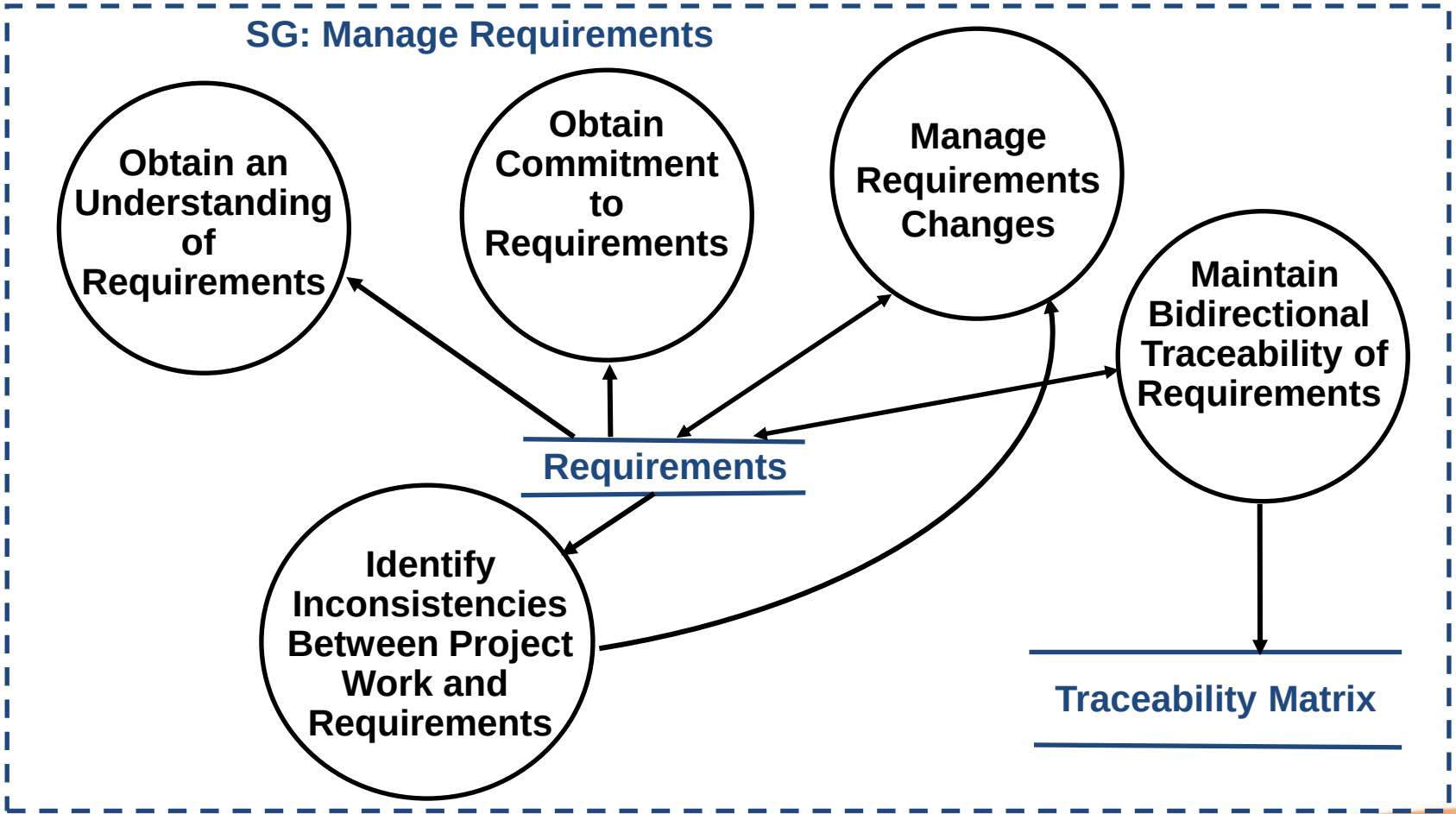
SP 1.4 Maintain Bidirectional Traceability of Requirements

SP 1.5 Identify Inconsistencies between project work and requirements



Example: Requirements Management (REQM) Context

Specific Goal, Specific Practices



Example: Requirements Development (RD, ML3) Specific Practices

SG 1 Develop Customer Requirements
SP 1.1 Elicit Needs
SP 1.2 Develop the Customer Requirements

SG 2 Develop Product Requirements
SP 2.1 Establish Product and Product-Component Requirements
SP 2.2 Allocate Product-Component Requirements
SP 2.3 Identify Interface Requirements

SG 3 Analyze and Validate Requirements
SP 3.1 Establish Operational Concepts and Scenarios
SP 3.2 Establish a Definition of Required Functionality
SP 3.3 Analyze Requirements
SP 3.4 Analyze Requirements to Achieve Balance
SP 3.5 Validate Requirements with Comprehensive Methods

Maturity Levels Cannot Be Skipped

A level provides a necessary foundation for effective implementation of processes at the next level.

- Higher level processes are easily sacrificed without the discipline provided by lower levels.
- The effect of innovation is obscured in a noisy process.

Higher maturity level processes may be performed by organisations at lower maturity levels, with risk of not being consistently applied in a crisis.

GG (Generic goals) = Institutionalization

GG2 (ML2): Institutionalize a Managed Process

The process is institutionalized as a managed process.

- A managed process is a performed process that is planned and executed in accordance with policy; employs skilled people having adequate resources to produce controlled outputs; involves relevant stakeholders; is monitored, controlled, and reviewed; and is evaluated for adherence to its process description.
- Management of the process is concerned with institutionalization and the achievement of specific objectives established for the process, such as cost, schedule, and quality objectives.

ML2 (maturity level)> GG2 (generic goal) > GPs (generic practices)

Applied to ALL Process Areas (ML2 and higher!!!)

GP2.1: Establish an Organizational Policy

GP2.2: Plan the Process

GP2.3: Provide Resources

GP2.4: Assign Responsibility

GP2.5: Train People

GP2.6: Control Work Products

GP2.7: Identify and Involve Relevant Stakeholders

GP2.8: Monitor and Control the Process

GP2.9: Objectively Evaluate Adherence

GP2.10: Review Status with Higher Level Management

Maturity levels: generic and specific practices

Maturity Level 2

- Requirements management
- Project planning
- Project monitoring and control
- Supplier agreement management
- Measurement and analysis
- Process and product quality assurance
- Configuration management

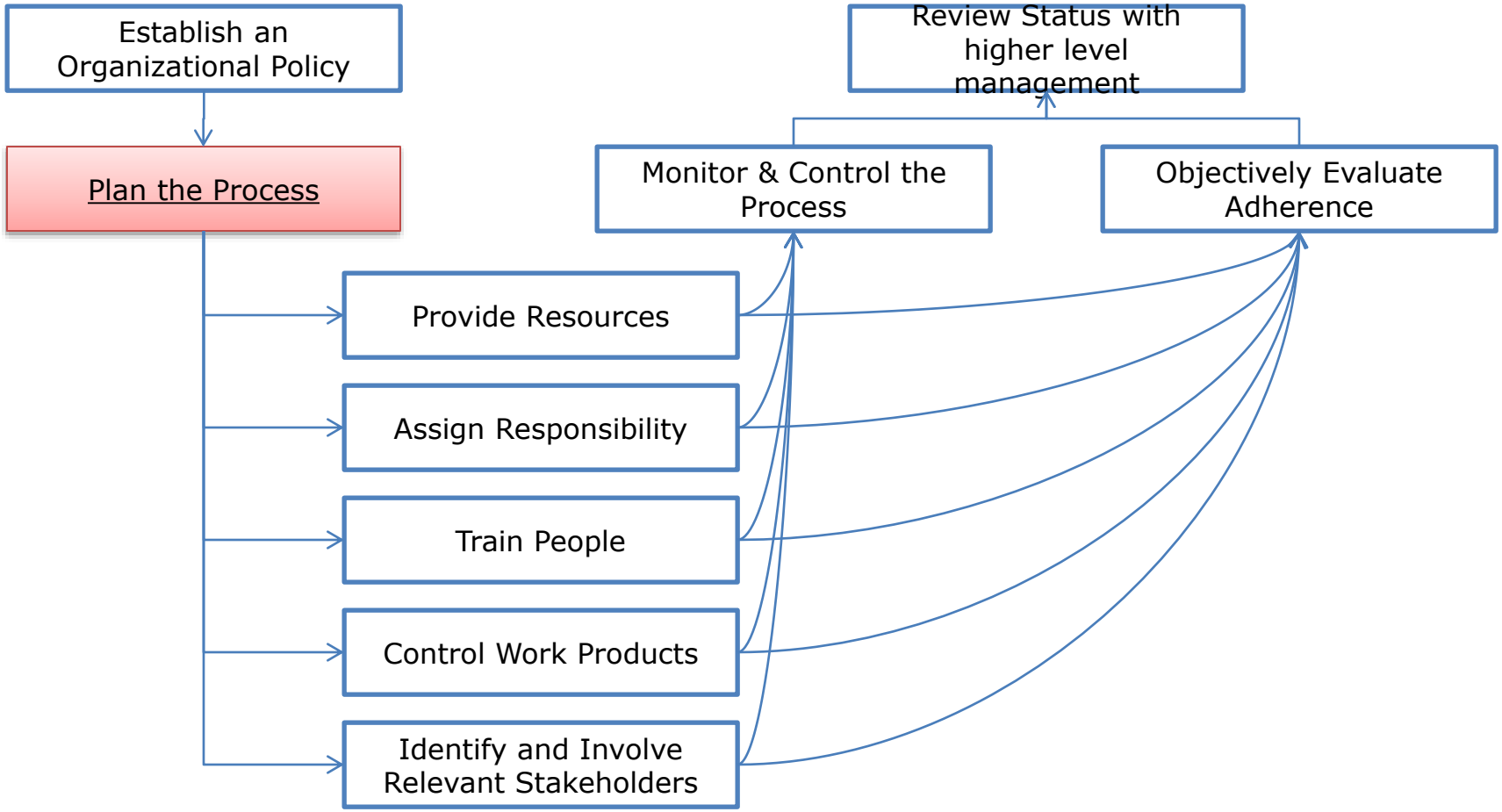
Maturity Level 3

- Requirements development
- Technical solution
- Product integration
- Verification
- Validation
- Organizational process focus
- Organizational process definition + IPPD
- Organizational training
- Integrated project management + IPPD
- Risk management
- Decision analysis and resolution

GP 2.1 Establish organizational policy
GP 2.2 Plan the process
GP 2.3 Provide resources
GP 2.4 Assign responsibility
GP 2.5 Train people
GP 2.6 Control Work Products (Manage configuration)
GP 2.7 Identify and involve relevant stakeholders
GP 2.8 Monitor and control the process
GP 2.9 Objectively evaluate adherence
GP 2.10 Review status with higher level management

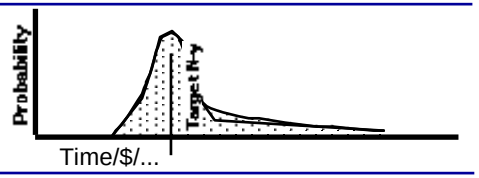
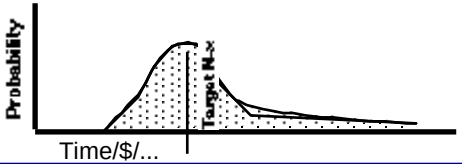
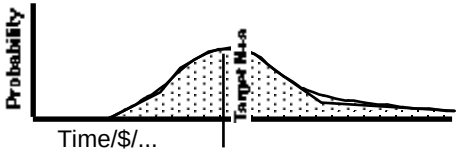
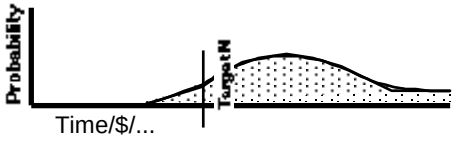
GP 3.1 Establish a defined process
GP 3.2 Collect improvement information

How PAs relate to Generic Practices?

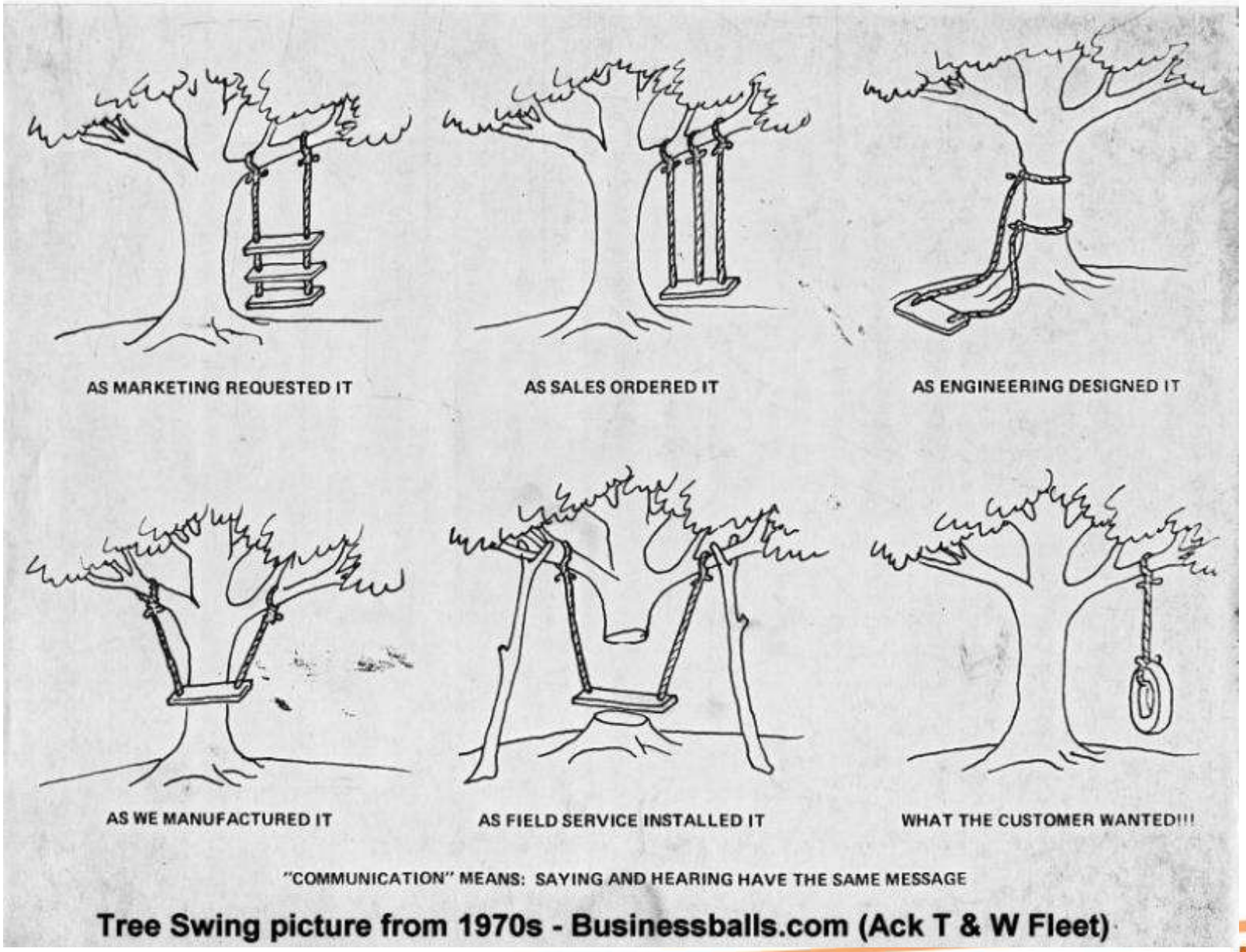


Source: Kiril Karaatanasov, ESI Center Bulgaria

Evolution of Process Capability

Level	Process Characteristics	Predicted Performance
5 Optimising	Process improvement is institutionalised	
4 Quantitatively Managed	Product and process are quantitatively controlled	
3 Defined	Software engineering and management processes are defined and integrated	
2 Managed	Project management system is in place; performance is repeatable	
1 Initial	Process is informal and unpredictable	

Remember: We want to avoid this!



DO NOT FORGET!!!

Process ≠ Bureaucracy

Process = Work

Analysis & Conclusions ...