



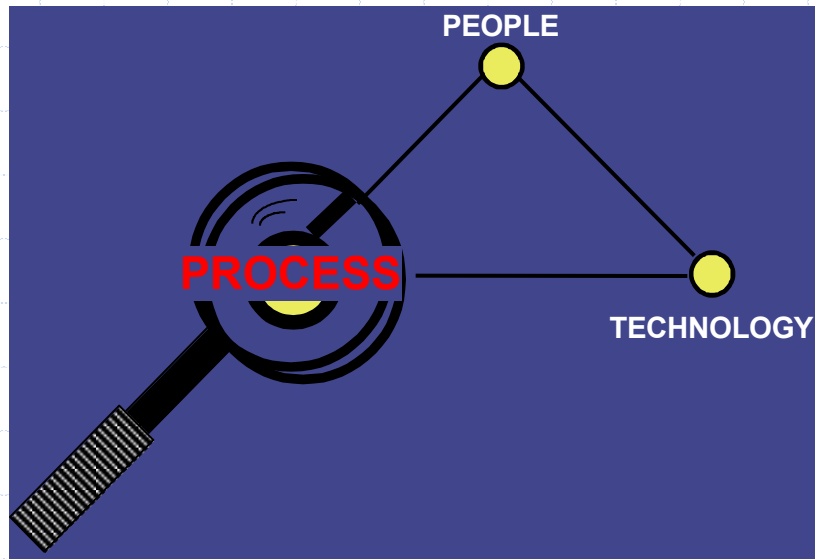
# CMM Integration (CMMI) Fundamental

# Outline

- ◆ Software Process Characteristics
- ◆ Evolution of CMMI
- ◆ Structure and Representations of Model
- ◆ CMMI Staged Representation
- ◆ CMMI Continuous Representation

# Why Focus on Process? -1

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



◆积极、优质的劳动力

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

◆即使是我们最优秀的员工也无法在流程不被理解或无法“最佳”运行的情况下发挥最佳水平

# Why Focus on Process? -2

## ◆ 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**
  - ◆ 回答的是：工作地更聪明，整个过程中
- As opposed to a focus on technology
  - ◆ technology applied without a suitable roadmap will not result in significant payoff
  - ◆ **technology provides most benefit in context of appropriate process roadmap**
  - ◆ 技术在适当的过程路线图中才能提供最大的收益

# Why Focus on Process? -3

## ◆ The process management premise: (前提)

- The quality of a system is highly influenced by the quality of the process used to acquire, develop, and maintain it.
- 系统的质量受获取、开发和维护过程的质量影响很大。

## ◆ This premise implies focus on processes as well as on product

- This is a long-established practice in manufacturing.
- Belief in this premise is visible worldwide in quality movements in manufacturing and service industries, e.g., ISO standards.

# Process Characteristics

◆ A process can be characterized by:

Its maturity. 成熟度

Its capability. 能力

Its performance. 表现性能

# Software Process Maturity

- ◆ Software process maturity is:
  - The extent to which a specific process is explicitly defined, managed, measured, controlled, and effective [Paulk].
  - 明确定义、管理、测量、控制和有效的特定过程的程度
- ◆ The maturity of an organization's process will help to determine how well it is able to create quality products **on time and on budget**.
- ◆ An organization's software process can be either **mature** or **immature**.
- ◆ A mature software process is **effective** in building an organization process capability.

# Software Process Maturity (cont.)

## ◆ A mature software process is:

- Defined, 已定义
- Managed, 可管理
- Measured, 可度量
- Controlled, 可控
- Effective, and 有效
- Able to improve. 可以改进



# Immature Software Organization

◆ An organization with an immature software process:

- Has **ad hoc** software processes which are usually improvised by practitioners and their management during the course of the project.
- May have a process that has been specified, but which is usually **not rigorously followed or enforced**.
- Is **reactionary** and is focused on **fire-fighting** (solving immediate crisis).
- Is highly dependent on capable and talented practitioners to perform **heroic efforts** to get the job done.

# Immature Software Organization (cont.)

- Is characterized by **cost and schedule overruns**.
- May **compromise** product functionality and quality in order to meet the schedule because of commitments made to hard deadlines.
- Finds it **difficult** to predict product **quality**.
- Sometimes just **doesn't know** what is going on until its over.

# Mature Software Organization

- ◆ An organization with a mature software process:
  - Knows **how to manage** software development and maintenance processes.
  - Is able to **plan and perform the plan** during normal and crisis situations.
  - Has **documented, effective and usable processes** that are consistent with the way work is actually done.
  - Has **buy-in** at **all levels** of the organization, everyone “talks the walk and walks the talk”.
  - Has **continuous improvement** as a way of life and an integral part of the organizational culture.

# Mature Software Process (cont.)

- Is able to objectively and quantitatively judge the **quality** of their products.
- Has schedules and cost **estimates** which are supported by historical data and are realistic.
- Is where **everyone follows a disciplined process** not because they are mandated to do so, but because there is value in doing so.

# Organizational Foundation for Software Process Maturity

- ◆ In order for an organization to mature, it needs to have an infrastructure and a supporting culture to support its software process.
- ◆ **Culture** is defined as, “That’s the way we do things around here”.
- ◆ **Infrastructure** is the underlying framework which supports the organization’s process:
  - **policies**, 方针
  - **standards**, 标准
  - **training**, 培训
  - **facilities**, and 设施
  - **tools**. 工具

# Institutionalized Process

- ◆ Having a supporting culture and infrastructure will allow a organization to institutionalize the software process as just the way the organization does business.

# Software Process Capability

- ◆ Software process capability is:
  - The range of expected results that can be achieved by following a software process [Paulk].
  - 通过遵循软件过程可以达到的预期结果的范围
- ◆ Knowing the software process capability allows an organization to predict the most likely **outcome** of future projects.

Upper control limit



Lower control limit

# Software Process Performance

- ◆ Software process performance is:
  - The actual results achieved by following a software process [Paulk].
  - 通过遵循软件过程所获得的实际结果
- ◆ A mature process has **better performance** than a immature process.
- ◆ A mature process can have **lower costs, lower development times, higher productivity and better quality** than an immature process.
- ◆ Caution: Performance depends on many factors.
- ◆ Not all factors are controlled by the process, such as people and technology.



# Capability vs. Performance

## ❖ Capability

- A sprinter who trains for the 100 yard dash is able to run the 100 yards in an average time of 10 seconds.
- Her range is really between 9 and 11 seconds. This is her capability.

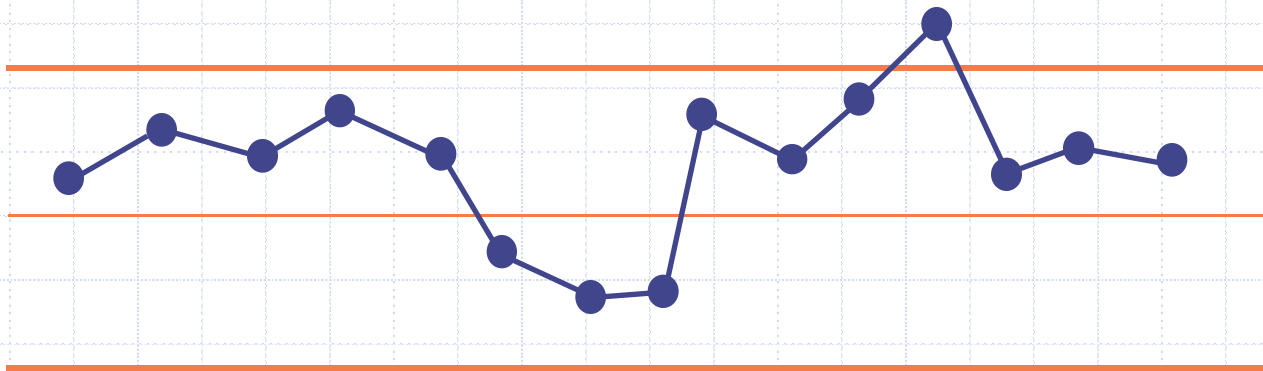
## ❖ Performance

- Her performance is the actual times she puts in when competing.
- It usually falls between 9 and 11 seconds.
- How good her performance is dependent on she is feeling, track conditions, what she had for breakfast.

# Process Capability Vs. Process Performance

## Process Capability

- ◆ Your review process on average finds 10 - 12 defects per 1000 lines of code.
- ◆ This is review process capability.



# Process Capability Vs. Process Performance (cont.)

## Process Performance

- ◆ The performance of the review process is the actual actual number of defects per 1000 lines of code found in reviews
- ◆ It usually falls between 10 and 12 defects per 1000 lines of code.
- ◆ How good the performance is dependent on a variety of factors, the readiness of the reviewers, the quality of the code under review.
- ◆ In one review, they might find 10, another 12, another 6 and another 18.

# Outline



- ◆ Software Process Characteristics
- ◆ Evolution of CMMI
- ◆ Structure and Representations of Model
- ◆ CMMI Staged Representation
- ◆ CMMI Continuous Representation

# Brief History

- ◆ In 1982, the U.S. Department of Defense (DoD) set up a joint task force to review the software problems in the DoD.
- ◆ Software Engineering Institute (SEI) was established at Carnegie Mellon University in December, 1984.
- ◆ 1989, Watts Humphrey, “Managing the Software Process”
- ◆ In 1991, the SEI created the Capability Maturity Model (CMM v1.0).
- ◆ 1993, CMM V1.1

# The CMM Explosion

**Software  
CMM**

**EIA 731**

**Systems  
Engr  
CMM**

**People  
CMM**

**IPD  
CMM**

**Software  
Acq  
CMM**

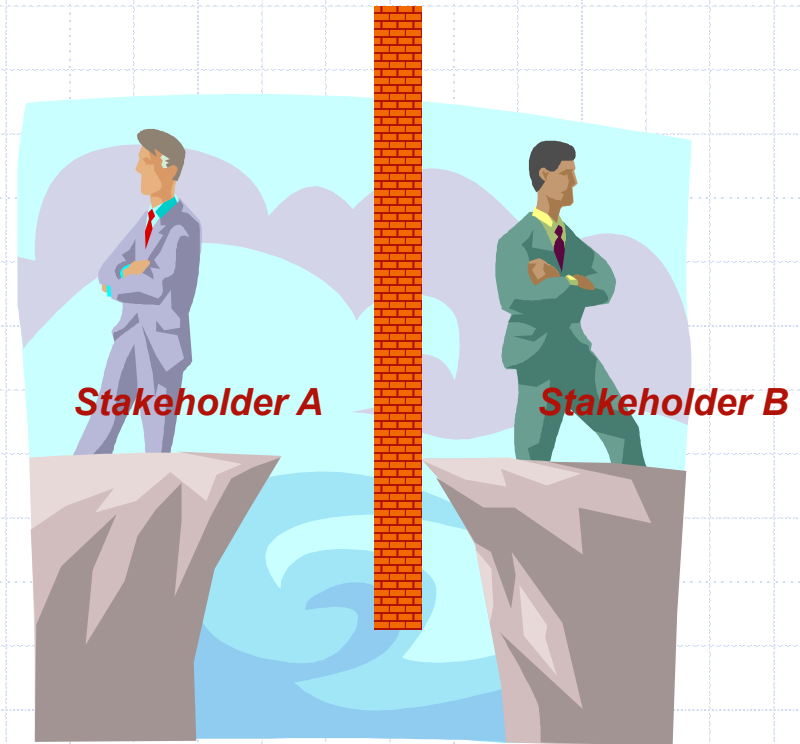
**Systems  
Security  
Engr CMM**

Success of the Software CMM® caused development of other CMMs, but they

- Have different structures, formats, terms, ways of measuring maturity
- Cause confusion, especially when more than one are used
- Are difficult to integrate into a combined improvement program
- Are difficult to use in supplier selection

# Challenge Unraveled...

- ◆ **Communication** between stakeholders within a software disciplines have traditionally **not been well integrated**
- ◆ The importance of communication software in systems has increased dramatically



\* Source: Standish Group Chaos Report

# CMMI - Bridging the Divide

- ◆ Software engineering processes across teams are **integrated**.
- ◆ Integrates stakeholders into **one** process improvement framework.
- ◆ Provides a **framework** for introducing new disciplines as needs arise.





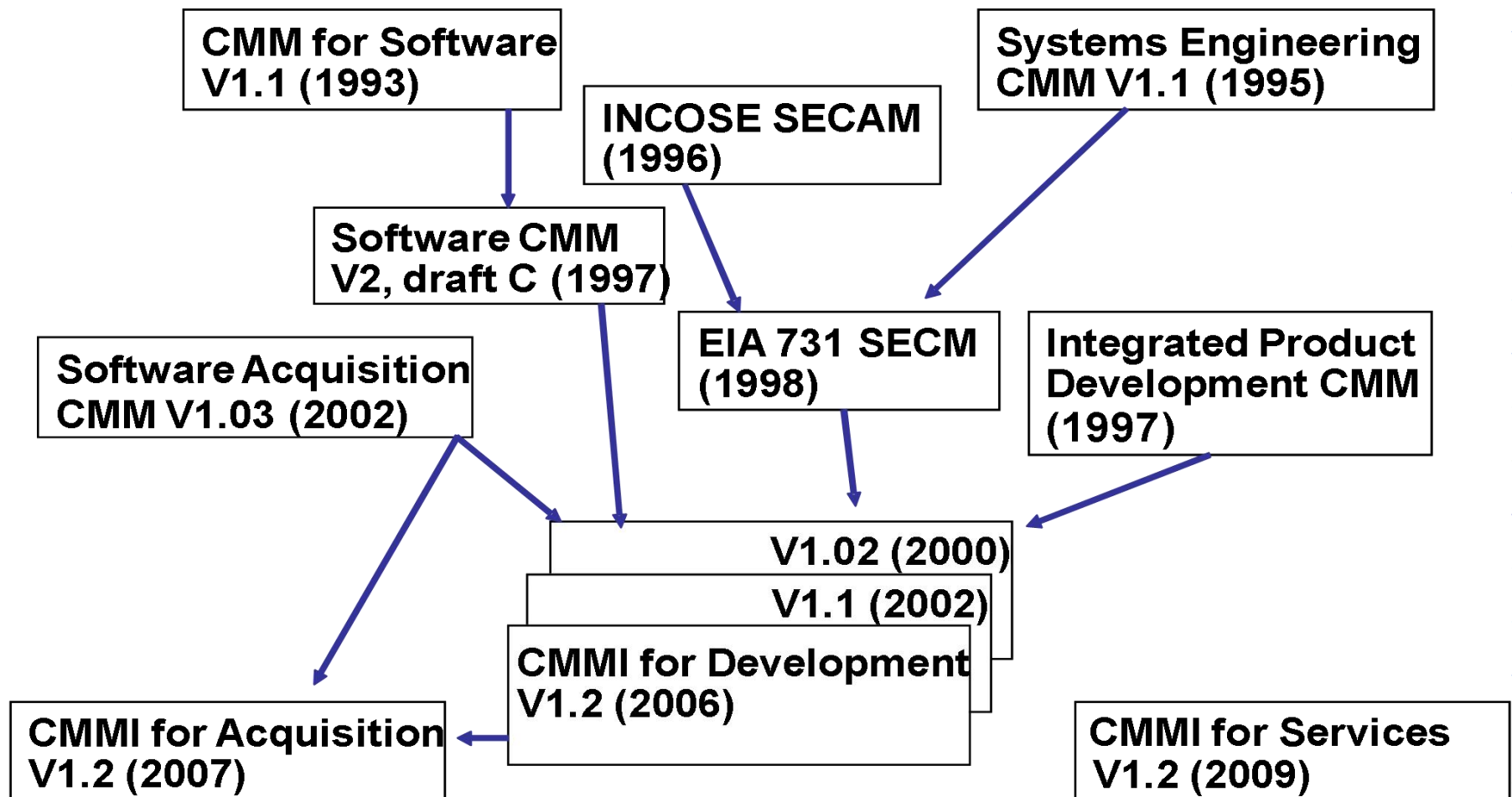
# The Three Source Models in CMMI

<b><i>Model Discipline</i></b>	<b><i>Source Model</i></b>
Software	SW-CMM, draft version 2c
Systems engineering	Systems Engineering Capability Model (SECM)
Integrated product and process development	IPD-CMM, version 0.98

# Numbers of Process Areas

Model	Areas	Term for Major Model Component
SW-CMM version 2(c)	19	Key process area
EIA/IS 731	19	Focus area
IPD-CMM version 0.98	23	Process area
CMMI-SW/SE version 1.1	22	Process area
CMMI-SW/SE/IPPD version 1.1	24	Process area
CMMI-SW/SE/IPPD/SS version 1.1	25	Process area
CMMI for Development –version 1.2	22	Process area

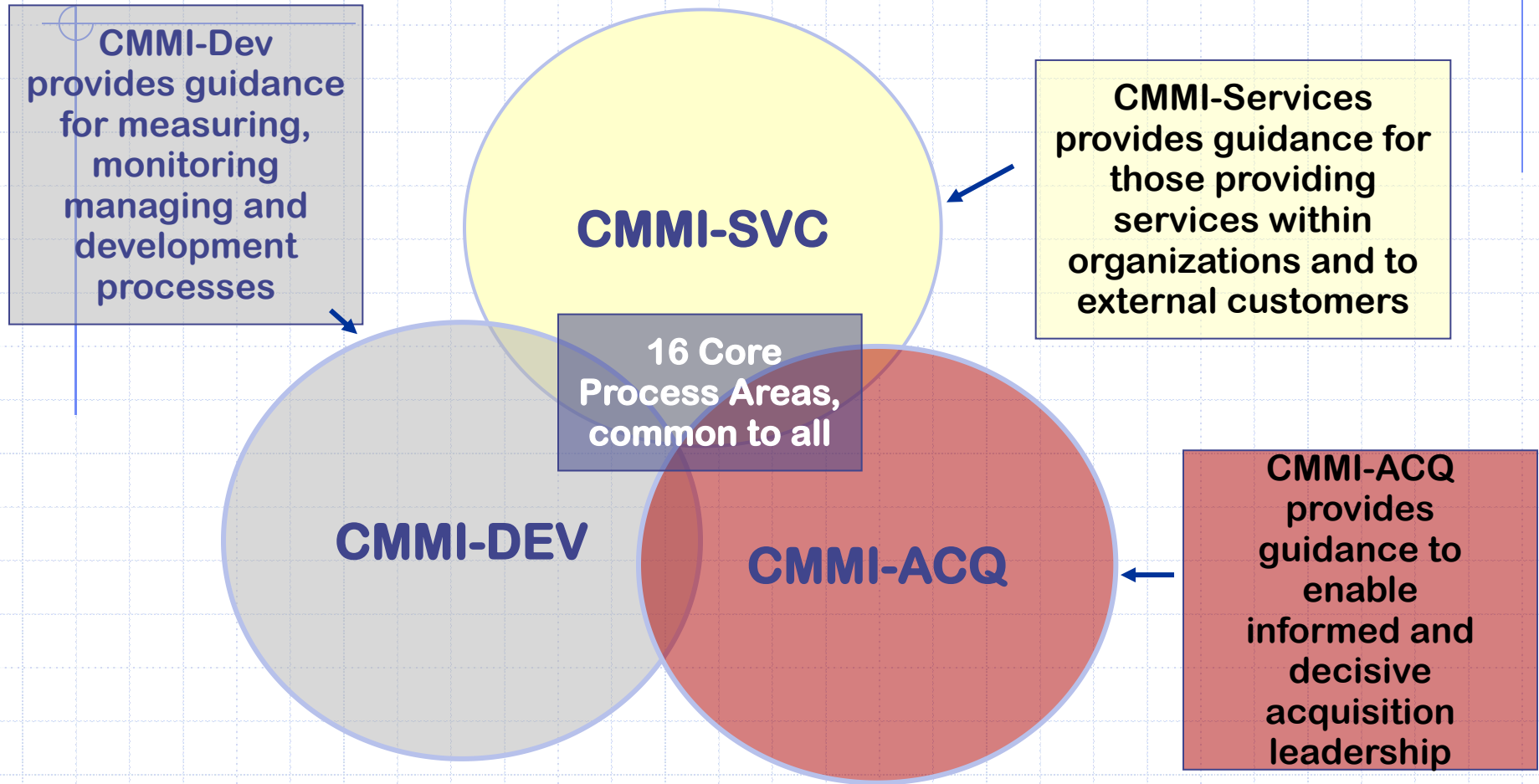
# Evolution of CMM/CMMI



# CMF (CMMI Model Foundation)

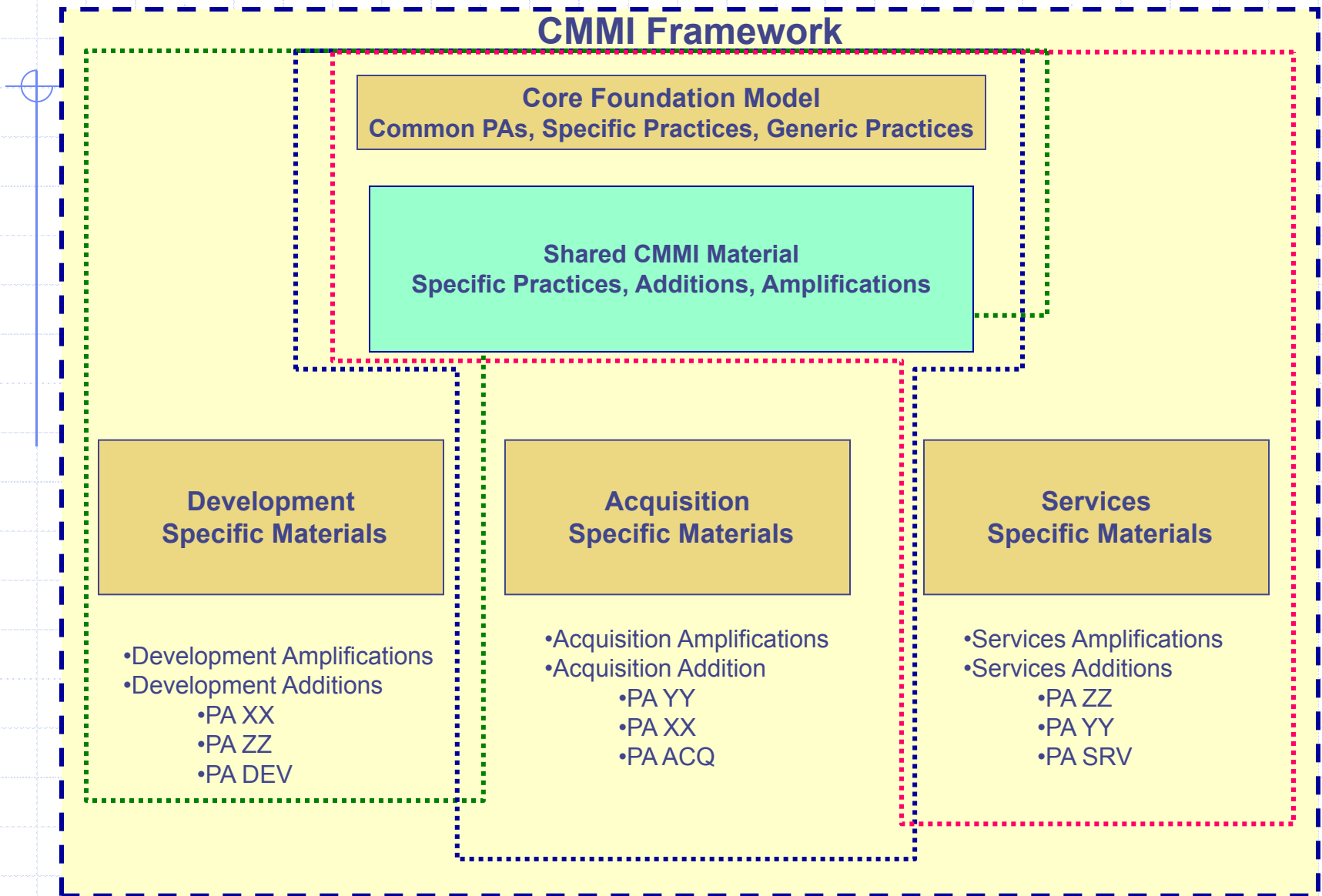
- ◆ “To allow the use of multiple models within the CMMI Framework, model components are classified as either common to all CMMI models or applicable to a specific model. The common material is called the “CMMI Model Foundation” or “CMF.”
- ◆ There are 16 Process Areas that constitute the CMF, these are also known as “Core” Process Areas.

# 3 Complementary “Constellations”



A “constellation” is defined as a collection of components that are used to construct models, training materials, and appraisal materials in an area of interest.

# Architecture & Constellations



# CMMI for Development

- ◆ Covers the development and maintenance activities applied to both products and services
- ◆ Models in CMMI-DEV contain practices that cover
  - project management
  - process management
  - systems engineering
  - hardware engineering
  - software engineering
  - supporting processes used in development and maintenance
- ◆ The CMMI for Development +IPPD model also covers the use of integrated teams for development and maintenance activities.

# CMMI for Acquisition

- ◆ CMMI-ACQ integrates bodies of knowledge that are essential for an acquirer
  - focus on activities for initiating and managing the acquisition of products and services that meet the needs of the customer
- ◆ CMMI for Development (CMMI-DEV) may be treated as a reference for supplier-executed activities for systems engineering, software development, and hardware design work in an acquisition initiative



# CMMI for Services

- ◆ CMMI-SVC draws on concepts and practices from CMMI and other service-focused standards and models, including
  - Information Technology Infrastructure Library (ITIL)
  - ISO/IEC 20000: Information Technology—Service Management
  - Control Objects for Information and related Technology (CobiT)
  - Information Technology Services Capability Maturity Model (ITSCMM)
- ◆ Covers the activities required to establish, deliver, and manage services
  - project management, process management, service establishment, service delivery and support, and supporting processes
- ◆ CMMI for Development (CMMI-DEV) may be treated as a reference for the development of the service system, which supports delivery of the service

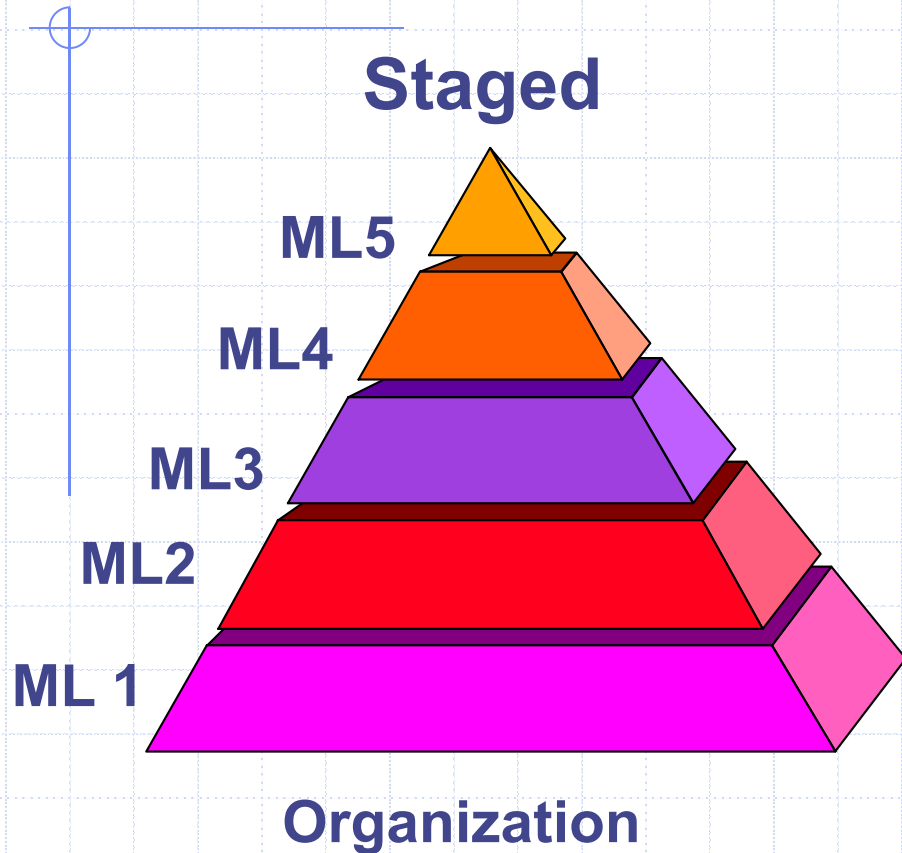
# Outline

- ◆ Software Process Characteristics
- ◆ Evolution of CMMI
- ◆ Structure and Representations of Model
- ◆ CMMI Staged Representation
- ◆ CMMI Continuous Representation

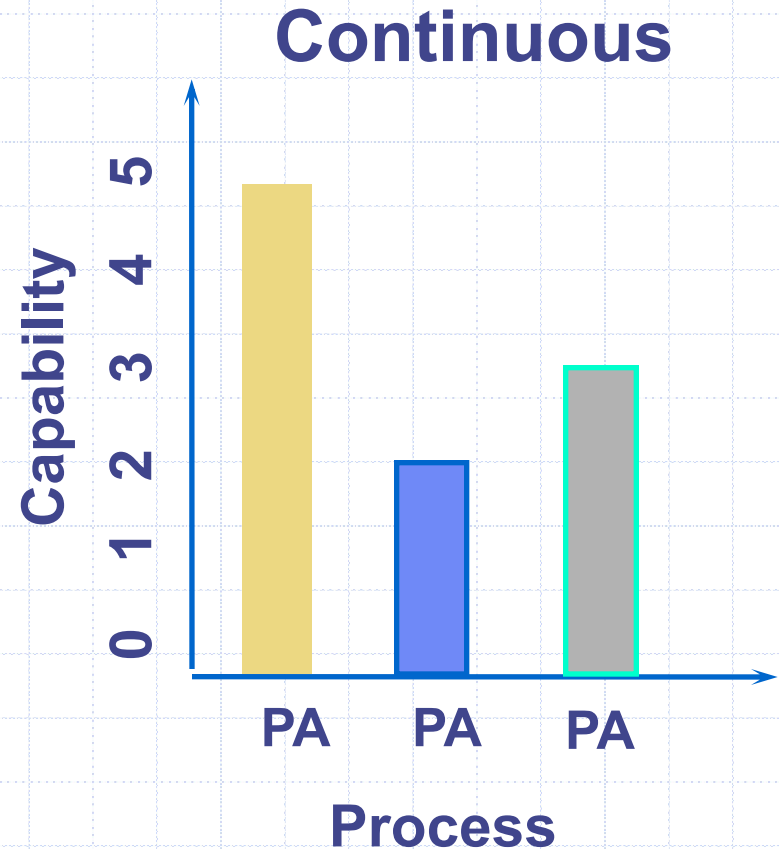
# CMMI Model Representations

- ◆ Historically, the community has used **two approaches** to process improvement.
- ◆ The CMMI provides **two representations** to facilitate users familiar with either approach:
  - process capability approach – **continuous representation**  
■ 连续式表述
  - organizational maturity approach – **staged representation**  
■ 阶段式表述
- ◆ Two Important Concepts
  - Organizational maturity      组织成熟度
    - ◆ pertains to **a set of PAs**
  - Process area capability      过程域能力
    - ◆ pertains to an **individual PA**

# CMMI Model Representations



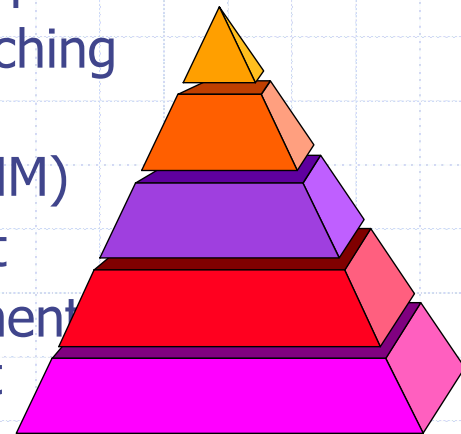
...for an established  
**set of process areas** across an  
organization



...for a **single process area**  
or a set of process areas

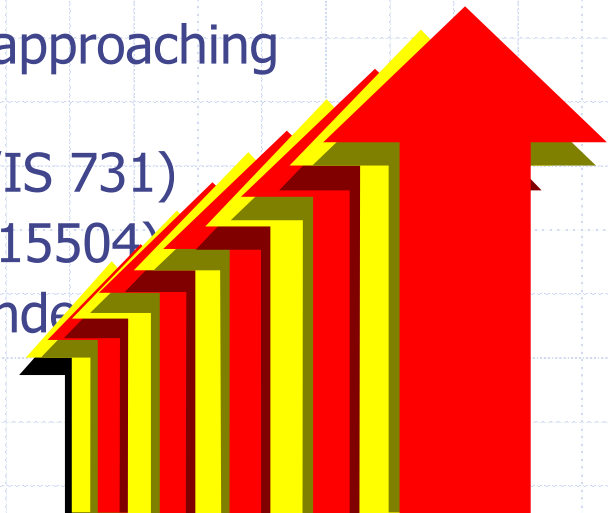
# Staged Representation

- ◆ Provides a pre-defined roadmap for organizational improvement based on proven grouping and ordering of processes and associated organizational relationships.
- ◆ 根据经过验证的分组和排序的过程和相关组织关系，为组织改进提供预定义的路线图。
- ◆ Why Use Staged?
  - Indicates maturity of an organization's standard process
    - ◆ to answer, “What is a good order for approaching improvement across the organization?”
  - Legacy influence (e.g., maturity levels in SW-CMM)
  - Proven roadmap for organizational improvement
  - Specifies an ordering for process area improvement however, in practice, the order is tailored to suit business needs.



# Continuous Representation

- ◆ Provides flexibility for organizations to choose which processes to emphasis for improvement, as well as how much to improve each process
- ◆ 为组织提供灵活性，以选择哪些流程需要重点改进，以及每个流程需要改进多少
- ◆ Why Use Continuous?
  - Indicates improvement within a single process area
    - ◆ to answer, “What is a good order for approaching improvement of this process area?”
  - Legacy influence (capability levels in EIA/IS 731)
  - Influence of international work (ISO/IEC 15504)
  - Treats process areas more or less independent
    - ◆ however, in practice, there are dependencies that must be recognized and dealt with.



# Comparison of Representations

<i>Continuous Representation</i>	<i>Staged Representation</i>
The organization selects process areas and capability levels based on its <b>process</b> improvement objectives	The organization selects process areas based on the <b>maturity levels</b>
Improvement is measured using <b>capability levels</b> . Capability levels <ul style="list-style-type: none"><li>• Measure maturity of a particular process across an organization.</li><li>• <b>Range from 0 through 5.</b></li></ul>	Improvement is measured using <b>maturity levels</b> . Maturity levels <ul style="list-style-type: none"><li>• Measure maturity of a set of processes across an organization.</li><li>• Range from <b>1 through 5.</b></li></ul>
Equivalent staging allows an organization using the continuous approach to process improvement to derive a maturity level as part of an appraisal.	There is no need for an equivalence mechanism back to the continuous approach.

# Factors to Consider when Selecting a Representation

◆ When making the decision to choose a representation, consider the following factors.

- Business
- Culture
- Legacy



## 小测验1: Example of Choosing Representation

- ◆ Two companies: Foo Toys and Widget Toys
- ◆ Both companies manufacture software-intensive toys and currently might be at CMM level 1
- ◆ Foo Toys
  - Wants to improve how the company handles **risks and integrates product components**.
  - Is happy with how the company's other processes are operating and so decides to focus on only those two process areas .  
(**continuous representation**)
  - Aim for Level 3
- ◆ Widget Toys
  - wants to improve the company's **overall development capability** and sees many process areas requiring attention. Recognizing the many interdependencies across process areas,  
(**staged representation**)
  - Aim to level 2

# Related Process Areas 相关的过程域

- ◆ Identify other process areas that could interact with the process area of interest.
- ◆ Process areas are grouped into categories (basic and advanced) of
  - Process Management 过程管理
  - Project Management 项目管理
  - Engineering 工程
  - Support 支持

# ◆ Process Management PAs

## ◆ The Process Management PAs of CMMI are as follows:

- Organizational Process Focus (OPF)
- Organizational Process Definition (OPD)
- Organizational Training (OT)
- Organizational Process Performance (OPP)
- Organizational Innovation and Deployment (OID)

# Project Management PAs

## ◆ The Project Management PAs of CMMI are as follows:

- Project Planning (PP)
- Project Monitoring and Control (PMC)
- Supplier Agreement Management (SAM)
- Integrated Project Management (IPM)
- Risk Management (RSKM)
- Quantitative Project Management (QPM)

# Engineering PAs

## ◆ The Engineering PAs of CMMI are as follows:

- Requirements Management (REQM)
- Requirements Development (RD)
- Technical Solution (TS)
- Product Integration (PI)
- Verification (VER)
- Validation (VAL)

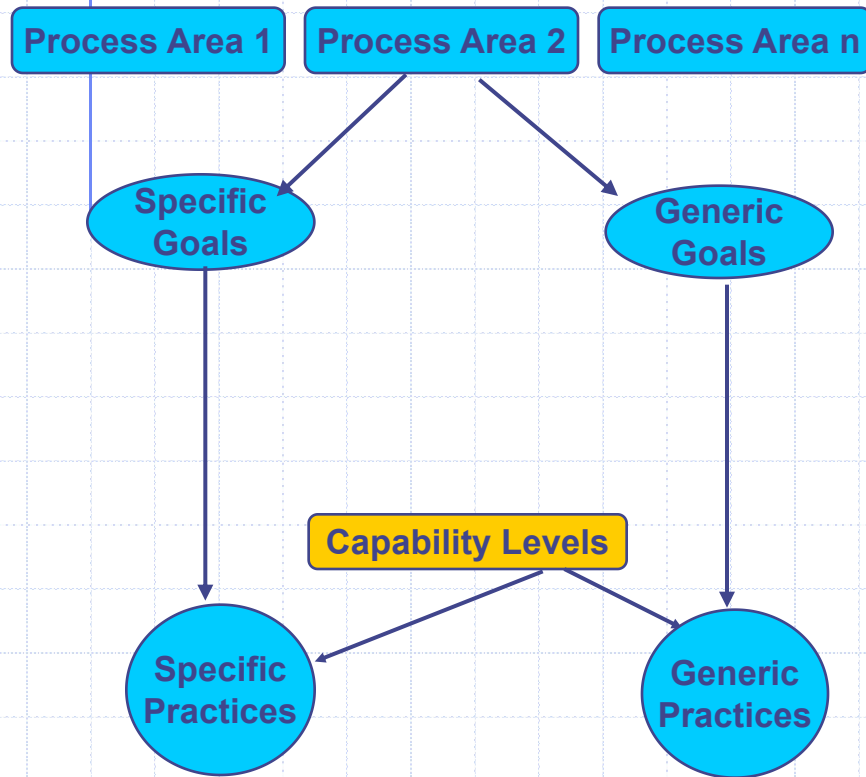
# Support PAs

## ◆ The Support PAs of CMMI are as follows:

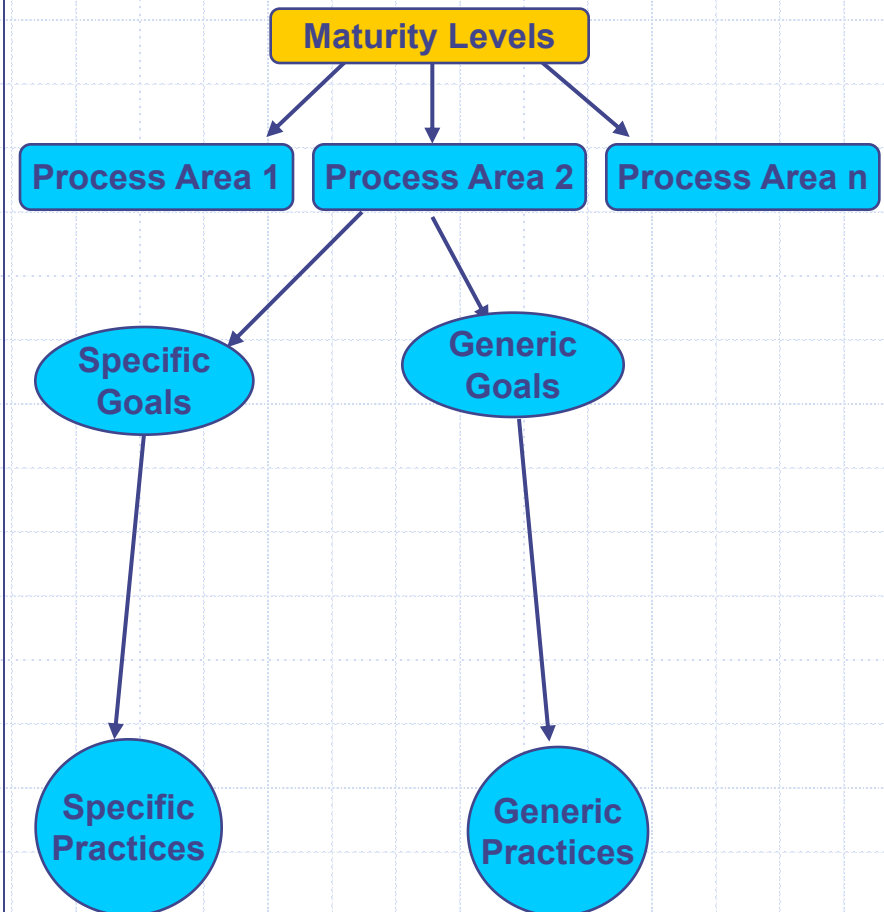
- Configuration Management (CM)
- Process and Product Quality Assurance (PPQA)
- Measurement and Analysis (MA)
- Causal Analysis and Resolution (CAR)
- Decision Analysis and Resolution (DAR)

# CMMI Model Structure (V1.2)

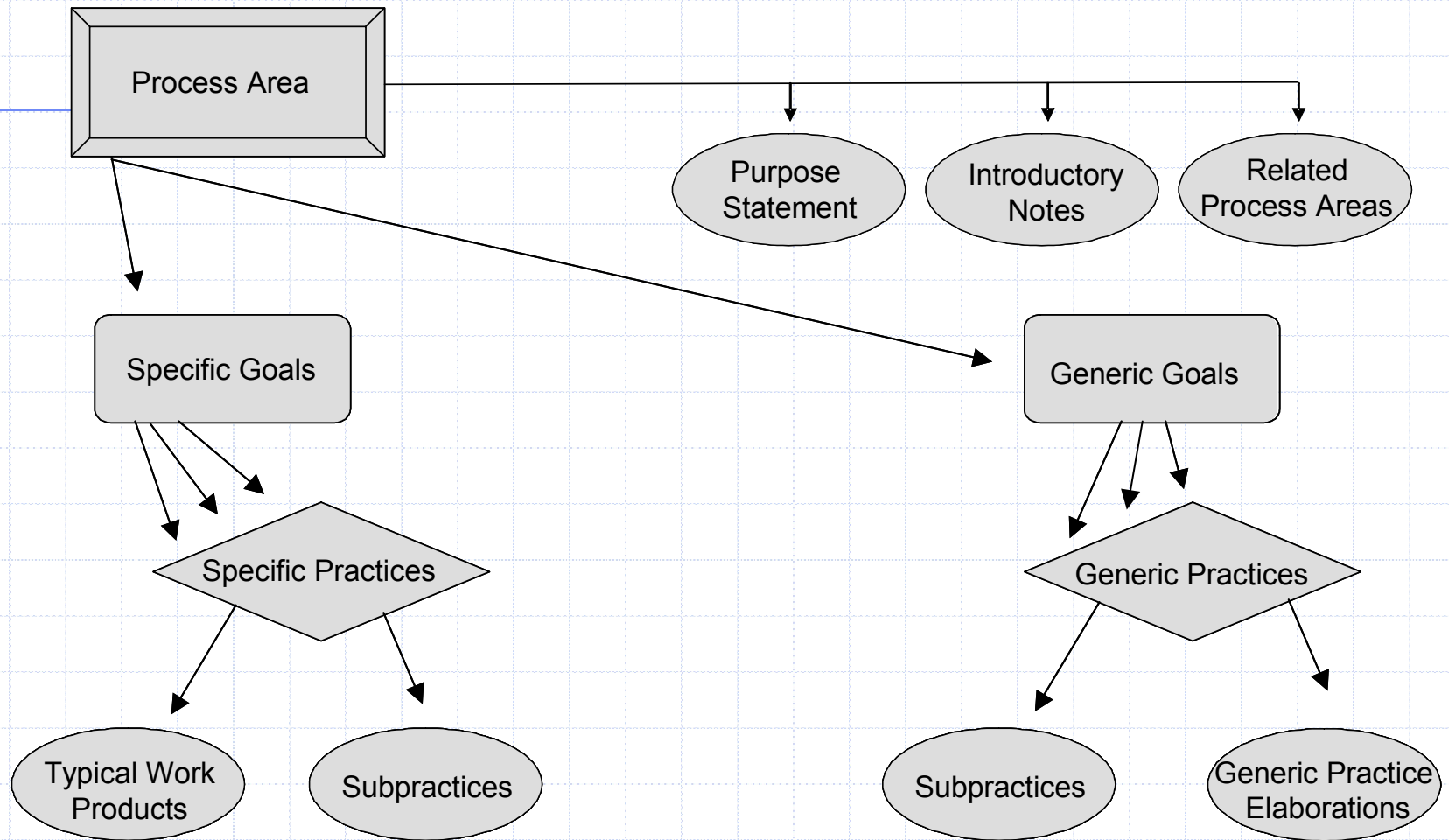
## Continuous



## Staged



# CMMI Model Components



KEY:

**Required**

**Expected**

**Informative**



# Process Areas 过程域

- ◆ A Process area is a cluster of related **practices** in an area that, when performed collectively, satisfy a set of **goals** considered important for making **significant improvement** in that area
- ◆ 一个过程域是一个领域中相关实践的集群。当集体执行时，它满足一组被认为对该领域的重大改进很重要的目标
  - All CMMI process areas are **common**相同 to **both continuous and staged representations**.
- ◆ **Practices** are actions to be performed to achieve the goals of a process area.
  - A major contribution of CMMI models is that they **structure and organize** these practices.

# Goals and Practices 目标和实践

- ◆ *Goals and Practices* are the model elements used to realize the values on both the capability and process dimensions.
  - Goal
    - ◆ A high level statement of the outcome to be achieved by effective implementation of a group of practices. (These are “required.”)
    - ◆ 一个总体的成果描述，说明通过有效实施一组实践将取得的成果
  - Practice
    - ◆ A description of an action that is necessary to enact a key element of a process area. (These are “expected,” and “alternate practices” are acceptable.)
    - ◆ 对一个行动的描述，这是制定一个过程领域的关键要素所必需的。

# Specific Goals 特定目标

- ◆ A specific goal applies to a process area and addresses the unique characteristics that describe what must be implemented to satisfy the process area.
- ◆ 一个特定的目标应用于一个过程域，描述为满足过程域必须要做的独特特性。
- ◆ **Example** from the Requirements Management
  - SG 1: Requirements are managed and inconsistencies with project plans and work products are identified.
  - SG 1: 需求要被管理，项目计划和工作产品的不一致要被识别。

# Specific Practices 特定实践

- ◆ A specific practice is an activity that is considered important in achieving the associated specific goal.
- ◆ 特定实践是指在实现相关特定目标时被认为是重要的活动。
- ◆ Example from Requirements Management
  - SP 1.3: Manage Requirements Changes
    - ◆ Manage changes to the requirements as they evolve during the project.
  - SP 1.3: 管理需求变更
    - ◆ 管理需求在项目期间的变化。

# Process Institutionalization 过程制度化

- ◆ Institutionalization: the ingrained根深蒂固的 way of doing business that an organization follows routinely as part of its corporate culture.
  - Institutionalization implies that the process is ingrained in the way the work is performed and there is commitment and consistency to performing the process.
  - An institutionalized process is more likely to be retained during times of stress.
- ◆ Generic goals and generic practices directly address process institutionalization.

# Generic Goals通用目标

- ◆ Generic goals are called “generic” because the same goal statement applies to **multiple process areas**. A generic goal describes the characteristics that must be present to **institutionalize the processes** that implement a process area.
- ◆ 一个通用目标描述了实施一个过程域的**过程制度化**所必须具备的特征。
- ◆ An example of a generic goal
  - The process is institutionalized as a managed process.
  - 该过程被制度化为一个可管理的过程。

# Generic Practices 通用实践

- ◆ Generic practices are called “generic” because the same practice applies to **multiple process areas**. A generic practice is the description of an activity that is considered important in achieving the associated generic goal.
- ◆ 一个**通用实践**是对一项**活动**的描述，该活动被认为对实现相关的**通用目标**很重要。
- ◆ An example of a generic practice for the generic goal “The process is institutionalized as a managed process”
  - Provide adequate resources for performing the process, developing the work products, and providing the services of the process.
  - **为执行过程、开发工作产品和提供过程服务提供足够的资源。**

# Informative Model Components 知识性模型组件

## ◆ Informative

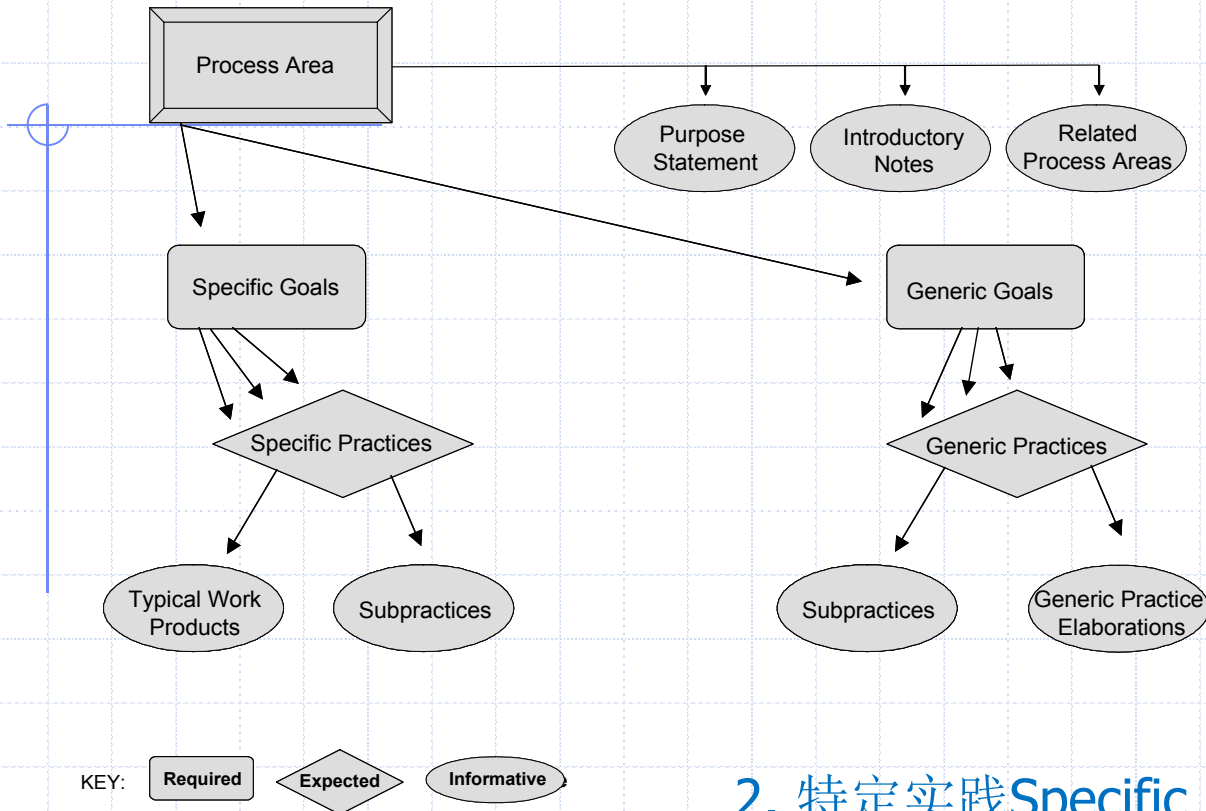
- Everything else is informative.
- Notes, typical work products, subpractices, discipline amplifications, and generic practice elaborations are informative.
- 注释、典型的工作产品、子实践、原理扩展和通用实践阐述都是信息性的
- These model components provide **details** about the model.



# Purpose, Notes 目的、注释

- ◆ The purpose is a brief statement of what is to be accomplished by the implementation of the practices of a particular process area.
- ◆ 目的是一个简要介绍说明，说明通过实施特定过程域的实践将要完成的任务。
  - E.g. “The purpose of Requirements Management is to manage the requirements of the **project's products** and **product components** and to identify inconsistencies between those requirements and the project's plans and **work products**.”
  - 比如：“需求管理的目的是管理项目的产品和产品组件的需求，并识别这些需求与项目计划和工作产品之间的不一致。”
- ◆ **Notes** provide details that help you understand the core information of the model.

# 小测验2



1. 如果需要把“项目策划”这个过程域定义成公司的一项制度，完成制度化的工作活动在下一部分描述的。

- A. Generic practices
- B. Generic goals
- C. Specific practices
- D. Specific goals

2. 特定实践Specific practices是一组重要的活动，。

- A. 用于实现相关通用目标
- B. 用于实现相关特定目标
- C. 由子实践和典型工作产品具体详细说明
- D. 由子实践和通用实践精化具体详细说明

# Subpractices and Typical Work Products

- ◆ Subpractices are suggested courses of action that correspond to specific practices.
- ◆ 子实践是与特定实践相对应的建议行动方案。
  - Example from SP 1.4:
    - ◆ Maintain requirements traceability to ensure that the source of lower level (derived) requirements is documented.
    - ◆ 维护需求的可追溯性，以确保低层次（派生）需求的来源被记录在文档中。
- ◆ Typical work products provide example outputs from a practice.
- ◆ 典型的工作产品就是提供了一个实践的示例输出。

# Discipline Amplifications 原理扩展

- ◆ Discipline amplifications contain information relevant to a particular discipline.
  - Example from Requirements Management
    - ◆ SG 1: Manage Requirements
    - ◆ *For Software Engineering*
      - The requirements may be a subset of the overall product requirements, or they may constitute the entire product requirements.
    - ◆ *For Systems Engineering*
      - Each level of product component design (e.g., segment, subsystem) received the requirements from the higher level.

# Sample Page from CAR (Causal Analysis and Resolution)

## Specific Goal and Practice Summary

- SG 1 Determine Causes of Defects
- SP 1.1 Select Defect Data for Analysis
  - SP 1.2 Analyze Causes
- SG 2 Address Causes of Defects
- SP 2.1 Implement the Action Proposals
  - SP 2.2 Evaluate the Effect of Changes
  - SP 2.3 Record Data

specific goal and practice summary

## Specific Practices by Goal

SG 1 Determine Causes of Defects

**Root causes of defects and other problems are systematically determined.**

specific goal

A root cause is a source of a defect such that if it is removed, the defect is decreased or removed.

SP 1.1 Select Defect Data for Analysis

**Select the defects and other problems for analysis.**

specific practice

typical work product

Typical Work Products

1. Defect and problem data selected for further analysis

Subpractices

1. Gather relevant defect or problem data

subpractice

examples

Examples of relevant defect data may include the following:

- Defects reported by the customer
- Defects reported by end user
- Defects found in peer reviews
- Defects found in testing

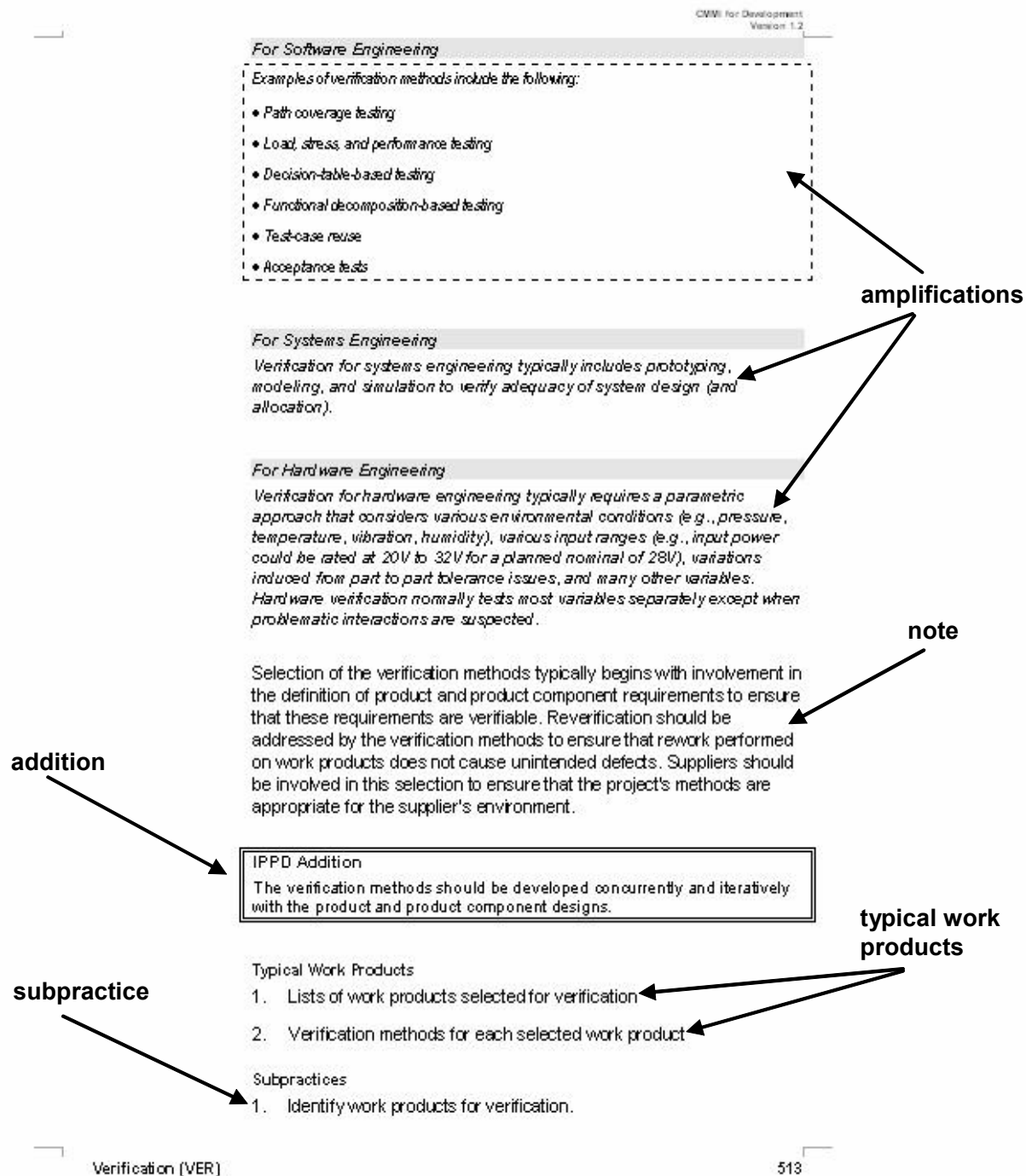
Examples of relevant problem data may include the following:

- Project management problem reports requiring corrective action
- Process capability problems
- Process duration measurements
- Earned value measurements by process (e.g., cost performance index)
- Resource throughput, utilization, or response time measurements

reference

Refer to the Verification process area for more information about work product verification.

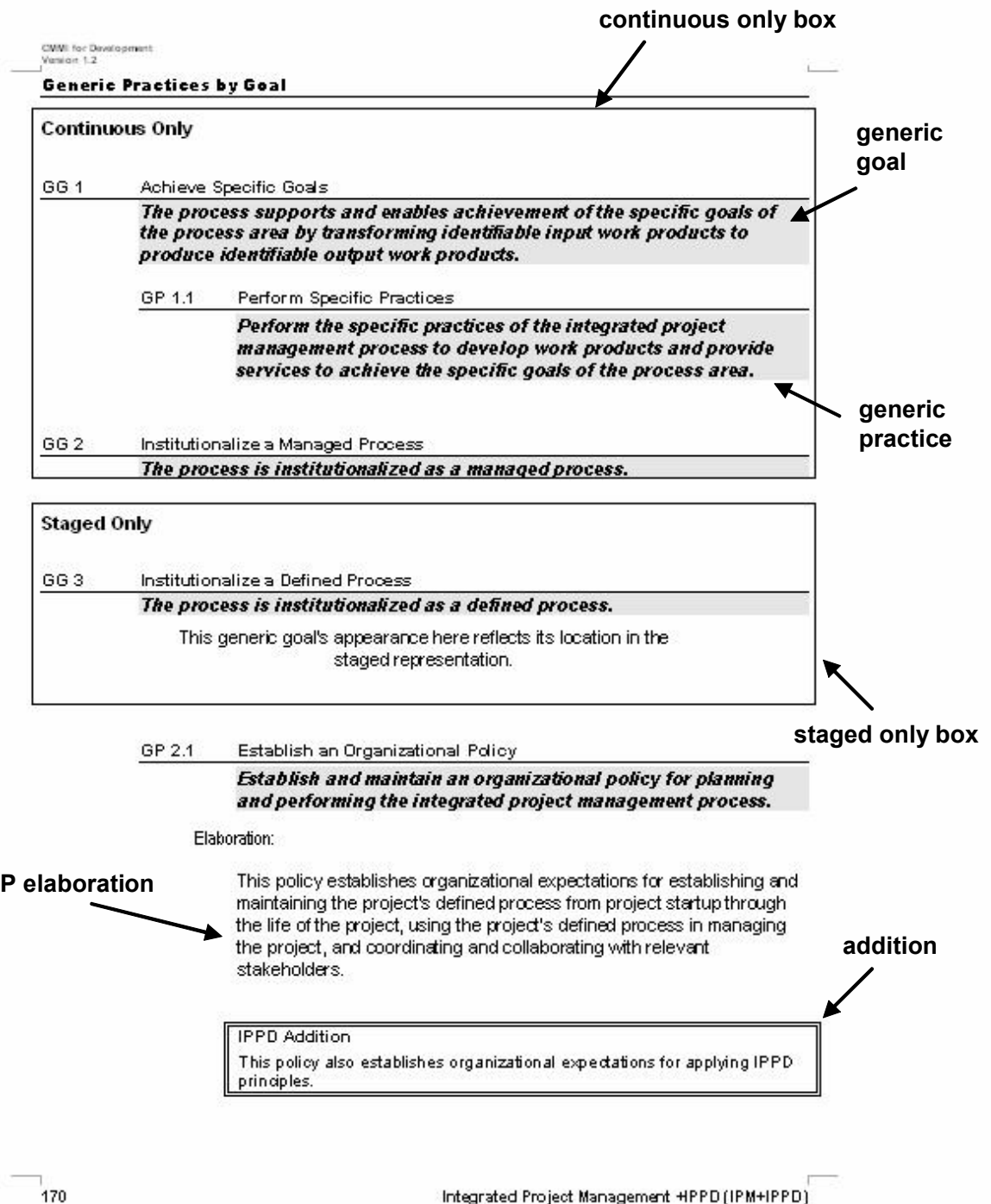
# Sample Page from VER (Verification )





# Sample Page from IPM (Integrated Project Management)

2020-3-5



# Outline



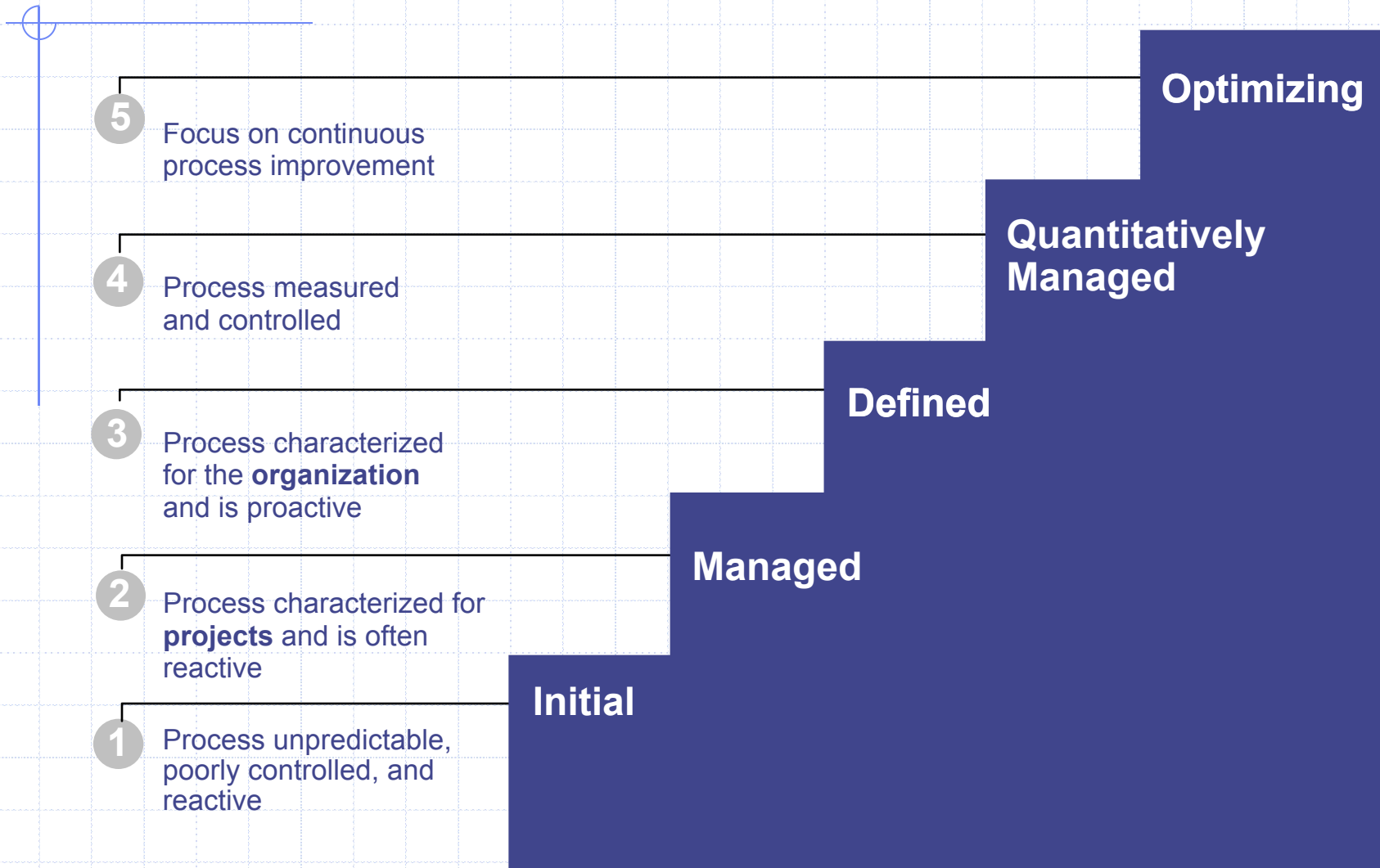
- ◆ Software Process Characteristics
- ◆ Evolution of CMMI
- ◆ Structure and Representations of Model
- ◆ CMMI Staged Representation
- ◆ CMMI Continuous Representation



# Maturity Levels

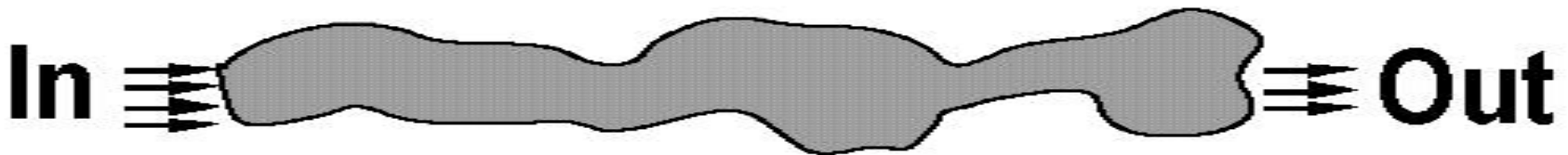
- ◆ A maturity level is a well-defined evolutionary plateau on the path to becoming a mature organization.
- ◆ 成熟度等级是一个定义明确的进化平台，平台是在成为成熟组织的路径。
- ◆ There are **five** maturity levels.
- ◆ Each level is a layer in the foundation for staged process improvement.

# The Maturity Levels



# Maturity Level 1: The Initial Level

- ◆ The software process is a black box, where visibility into the project's process is limited.
- ◆ It is difficult to establish the project's progress and activity status since the activities are poorly defined.
- ◆ Problems result in crisis situations and fire-fighting.
- ◆ Success depends on the competence and heroics of the people doing the work.
- ◆ Is characterized by the **absence of sound managerial practice.**



# Level 1: the “Initial” Level

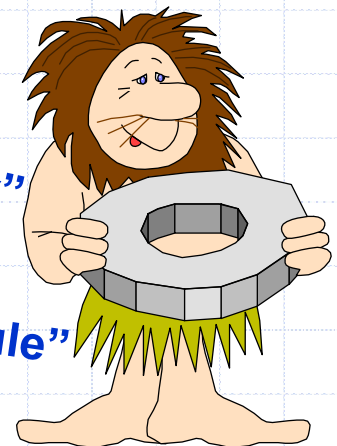
## Success depends on heroes

### ***Good performance is possible - but***

- ◆ Requirements often misunderstood, uncontrolled
- ◆ Schedules and budgets frequently missed
- ◆ Progress not measured
- ◆ Product content not tracked or controlled
- ◆ Engineering activities nonstandard, inconsistent
- ◆ Teams not coordinated, not trained
- ◆ Defects proliferate

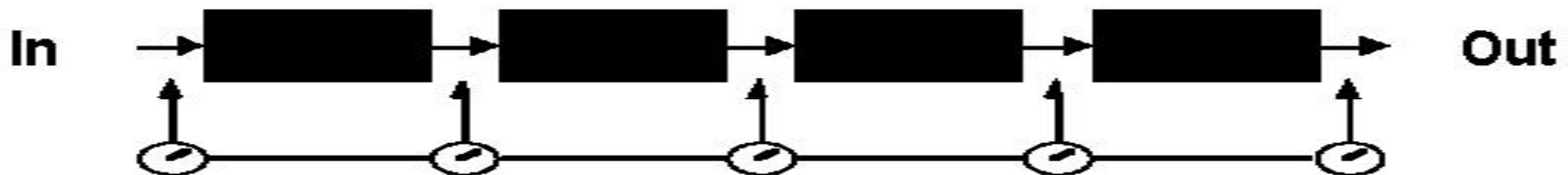
“Schedules  
run everything”  
“Just send in  
the Tiger Team”

“Processes limit my creativity”  
“Processes don’t help my delivery schedule”



# Maturity Level 2: The Managed Level

- ◆ Uses **experience** from previous successful projects in planning and managing new projects.
- ◆ Has established **organizational policies** for managing a software project and procedures for implementing these policies.
- ◆ The **process of building the software** is a series of black boxes where visibility is provided at predefined checkpoints (ex., **milestones**).
- ◆ **Management** reacts in a controlled manner to problems as they occur.



# CMMI Level 2: the “Managed” Level - Establishing basic project management controls

## 7 Process Areas

### CLARIFY REQUIREMENTS

- Baseline the product requirements

### DOCUMENT PLANS

- Estimate project parameters,
- Develop plans and processes

### TRACK PROGRESS

- Measure actual progress to enable timely corrective action
- Measure for mgmt. info needs
- Verify adherence of processes and products to requirements

### CONTROL PRODUCTS

- Identify and control products, changes, problem reports
- Select qualified suppliers / vendors; manage their activities

– Requirements Management (REQM)

} Project Planning (PP)

– Project Monitoring and Control (PMC)

– Measurement & Analysis (M&A)

– Process & Product Quality Assurance (PPQA)

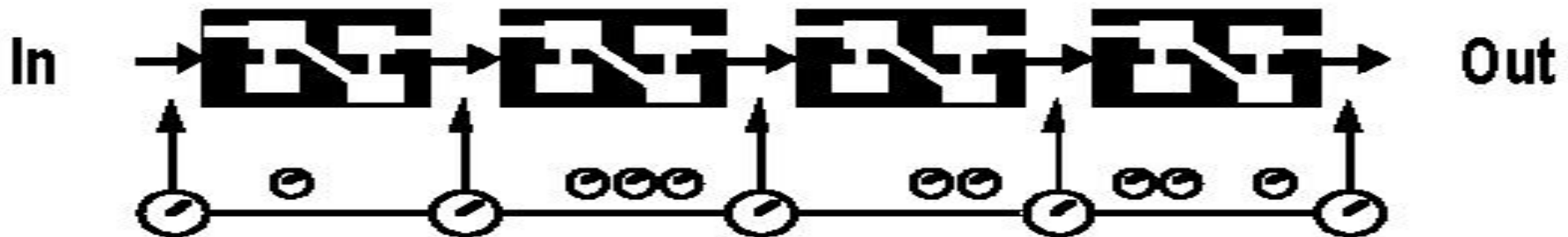
– Configuration Management (CM)

– Supplier Agreement Management (SAM)



# Maturity Level 3: The Defined Level

- ◆ Builds a **standard process or set of processes** for developing and/or maintaining software which is then used across the organization.
- ◆ Creates a **group** within the organization responsible for software process activities, sometimes called a Software Engineering Process Group (**SEPG**).
- ◆ The internal structure of each black box (i.e., the tasks) are **visible** to all.
- ◆ Management prepares **proactively for risks** that may arise.



# CMMI Level 3: the “Defined” Level - Standardizing the organization's process

## 11 Process Areas\*

### ENGINEER THE PRODUCT

- Clarify customer requirements
- Solve design requirements; develop implementation processes
- Assemble product components, deliver
- Ensure products meet requirements
- Ensure products fulfill intended use
- Analyze decisions systematically

- *Requirements Definition* (RD)
- *Technical Solution* (TS)
- *Product Integration* (PI)
- *Verification* (Ver)
- *Validation* (Val)
- *Decision Analysis & Resolution* (DAR)

### MANAGE THE PROCESSES

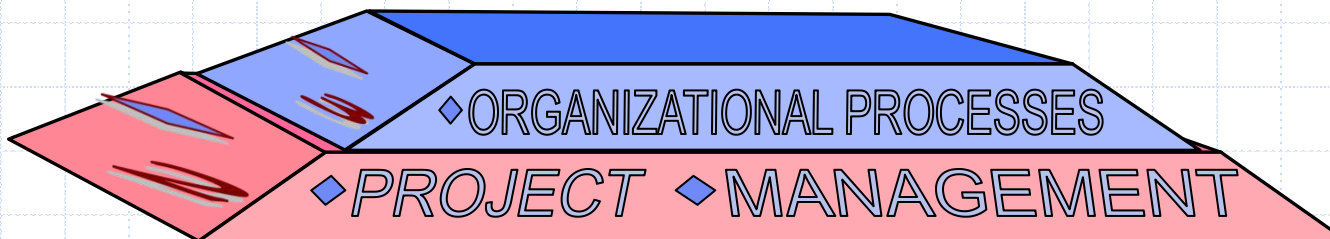
- Follow integrated, defined processes
- Identify and control potential problems

- *Integrated Project Mgmt* (IPM)
- *Risk Management* (RSKM)

### PROVIDE ORG. INFRASTRUCTURE

- Establish org. responsibility for PI
- Define the org's best practices
- Develop skills and knowledge

- *Org. Process Focus* (OPF)
- *Org. Process Definition* (OPD)
- *Org. Training* (OT)



\*SE/SW model



# 小测验4

1.过程域Process & Product Quality Assurance (PPQA过程域产品质量保证)属于哪一个成熟度等级

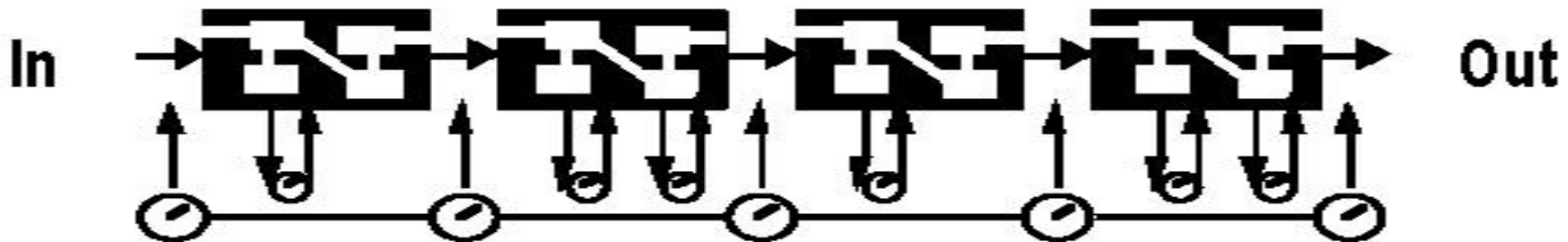
- A. Level 1
- B. Level 2
- C. Level 3
- D. Level 4

2. 成熟度等级2是项目级的过程管理，主要关注的方面有哪些

- A. Requirements Management
- B. Project Planning , Project Monitoring and Control
- C. Measurement & Analysis, Process & Product Quality Assurance
- D. Configuration Management, Supplier Agreement Management

# Maturity Level 4: The Quantitatively Managed Level

- ◆ Sets **quantitative quality goals** for both the software process and products.
- ◆ Establishes a organization-wide software process **database** to collect, analyse and store data from the projects.
- ◆ Uses **measurements** to establish a quantitative foundation in order to **evaluate** software processes and products.
- ◆ The software processes within the black boxes are now **instrumented and controlled quantitatively**.



# CMMI Level 4: the “Quantitatively Managed” Level - Quantitative analysis of processes and products for monitoring and control

## MANAGE PROJECTS QUANTITATIVELY

- ◆ Statistically manage the project's processes and sub-processes

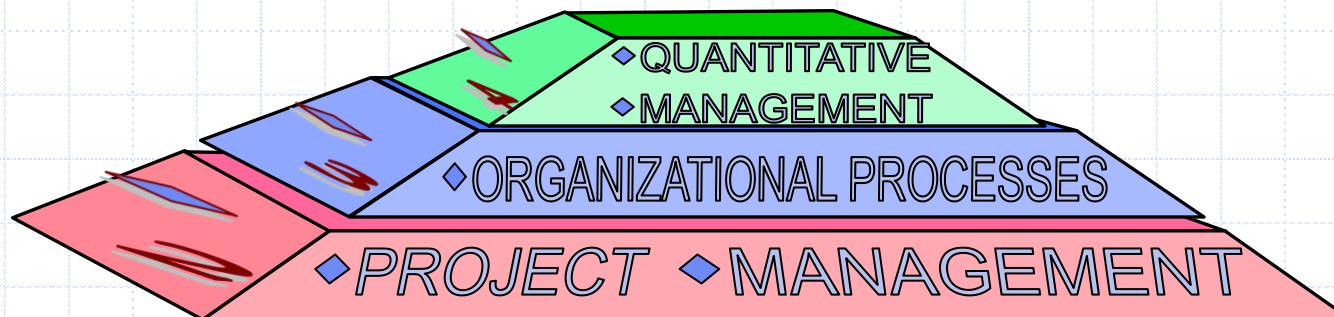
### 2 Process Areas

- Quantitative Project Management (QPM)

## MANAGE THE ORGANIZATION QUANTITATIVELY

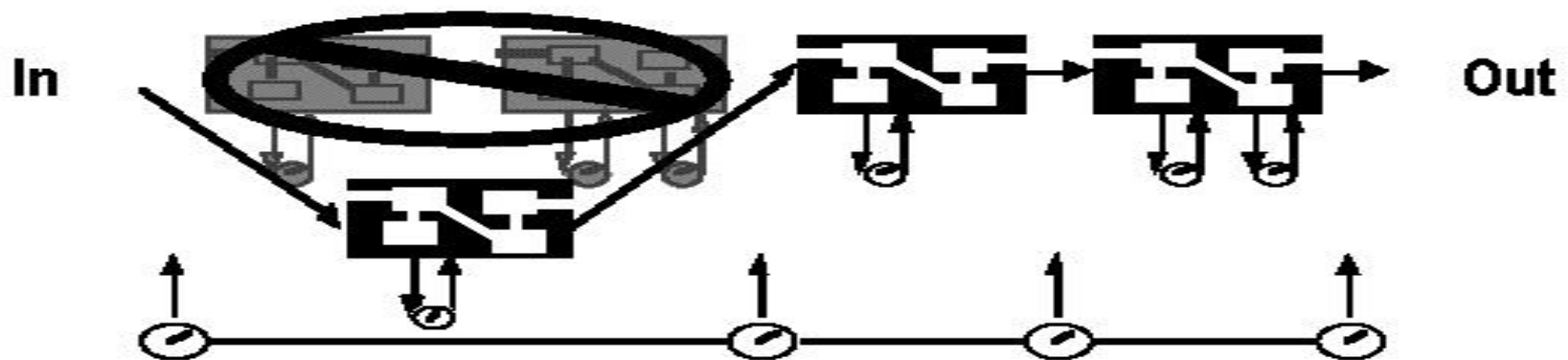
- ◆ Understand process performance; quantitatively manage the organization's projects

- Organizational Process Performance (OPP)



# Maturity Level 5: The Optimizing Level

- ◆ Can **identify weaknesses** and deal with them proactively.
- ◆ Has software teams that practice **defect prevention** as a way of life.
- ◆ The black boxes can now be **removed** and new and improved black boxes can be **introduced**.
- ◆ Managers are able to effectively **estimate** and then **track** the impact and effectiveness of change.



# CMMI Level 5: the “Optimizing” Level - Institutionalizing process improvement

## OPTIMIZE PERFORMANCE

- Identify and eliminate the cause of defects early

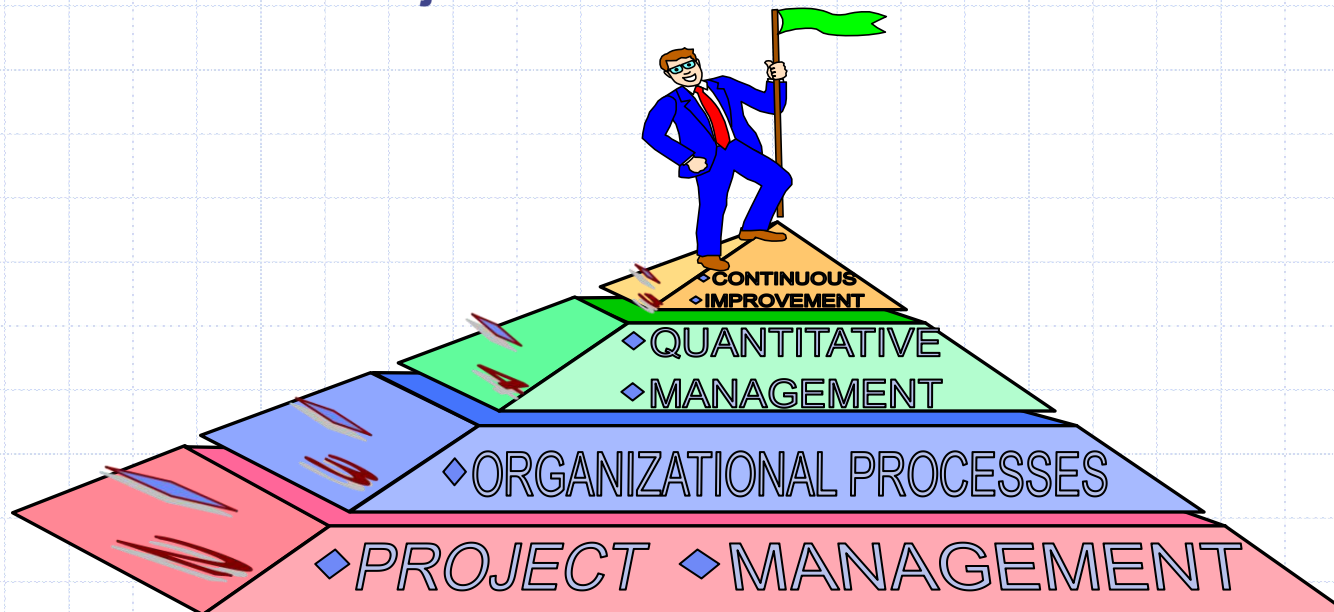
## 2 Process Areas

— *Causal Analysis and Resolution* (CAR)

## ADOPT IMPROVEMENTS

- Identify and deploy new tools and process improvements to meet needs and business objectives


— *Organizational Innovation and Deployment* (OID)



# Maturity Levels Should Not Be Skipped

- ◆ Each maturity level provides a necessary foundation for effective implementation of processes at the next level.
  - Higher level processes have less chance of success without the discipline provided by lower levels.
  - The effect of innovation can be obscured in a noisy process.
  
- ◆ Higher maturity level processes may be performed by organizations at lower maturity levels, with the risk of not being consistently applied in a crisis.

# Process Areas by Maturity Level

Level	Focus	Process Areas	
5 Optimizing	<i>Continuous process improvement</i>	Organizational Innovation and Deployment Causal Analysis and Resolution	<div>Quality Productivity</div>  <div>Risk Rework</div>
4 Quantitatively Managed	<i>Quantitative management</i>	Organizational Process Performance Quantitative Project Management	
3 Defined	<i>Process standardization</i>	Requirements Development Technical Solution Product Integration Verification Validation Organizational Process Focus Organizational Process Definition Organizational Training Integrated Project Management Risk Management Decision Analysis and Resolution	
2 Managed	<i>Basic project management</i>	Requirements Management Project Planning Project Monitoring and Control Supplier Agreement Management Measurement and Analysis Process and Product Quality Assurance Configuration Management	
1 Performed			

# Generic Goals and Practices for Staged Representation

Maternity Level:    Generic Goals (GG):    Generic Practices (GP):

5	(Optimizing)	(None)	(None)	
4	(Quantitatively Managed)	(None)	(None)	
3	(Defined)	Institutionalize a Defined Process.	Establish a defined process. Collect improvement information.	
2	(Managed)	Institutionalize a Managed Process.	Establish org. policy. Plan the process. Provide resources. Assign responsibility. Train people. Perform managed process.	Manage configurations. Identify & involve relevant stakeholders. Monitor and control the process. Objectively verify adherence. Review status with mgmt.
1	Initial	(None)	(None)	
0	NA	(None)	(None)	



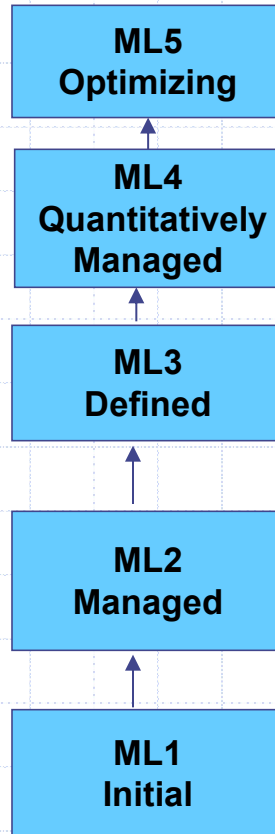
# Achieving Maturity Levels

GP2.1 through GP3.2  
All ML2, ML3, ML4,  
and ML5 PAs

GP2.1 through GP3.2  
All ML2, ML3, and ML4  
PAs

GP2.1 through GP3.2  
All ML2 and ML3 PAs

GP2.1 through GP2.10  
All ML2 PAs



Defect prevention, proactive improvement,  
innovative technology insertion and deployment

Measure process performance,  
stabilize process, control charts,  
deal with causes of special variations

Project's process is tailored from organization's  
standard processes, understand process qualitatively,  
process contributes to the organizations assets

Adhere to policy, follow documented plans and processes,  
apply adequate resources, assign responsibility and  
authority, train people, apply CM, monitor, control, and  
evaluate process, identify and involve stakeholders,  
review with management

Processes are ad hoc and chaotic

# The CMMI and China's Kung Fu

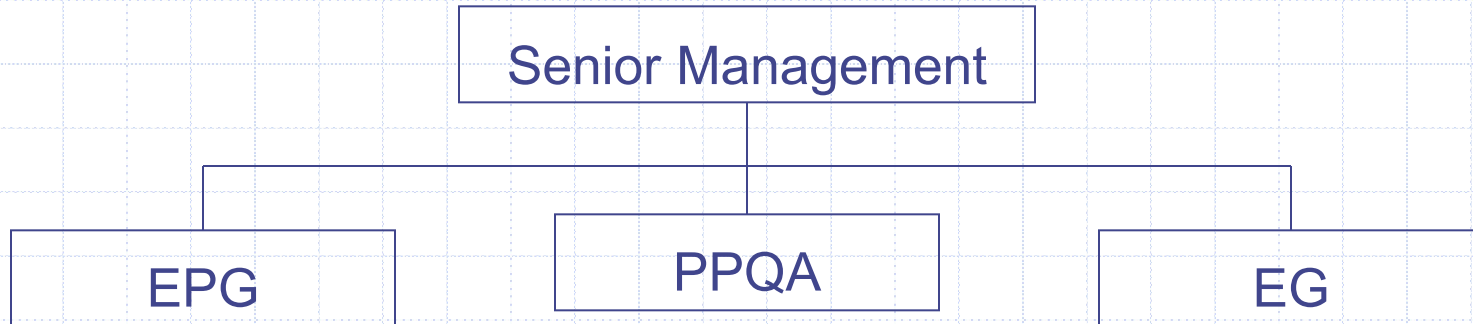
## ◆ CMMI

- ◆ CMMI Levels
- ◆ Goals on each level
- ◆ Key practices to reach each level's goals
- ◆ Management
- ◆ Technology & tools (non-CMMI)

## ◆ China's Kung Fu

- ◆ Kung Fu's accomplish
- ◆ States on each accomplish
- ◆ Mastery & practices to reach each accomplish
- ◆ Internal kung
- ◆ External kung

# West Legal System Concept in CMMI



- ◆ EPG – legislate rules for the organization
- ◆ PPQA – control quality for the organization
- ◆ EG – execute the process for the organization

# Questions

- ◆ In your groups, take an organization that you worked at or the case study background, and using the definitions of each of the CMMI levels, identify the level and state why you believe it was at that level.
- ◆ Present your results

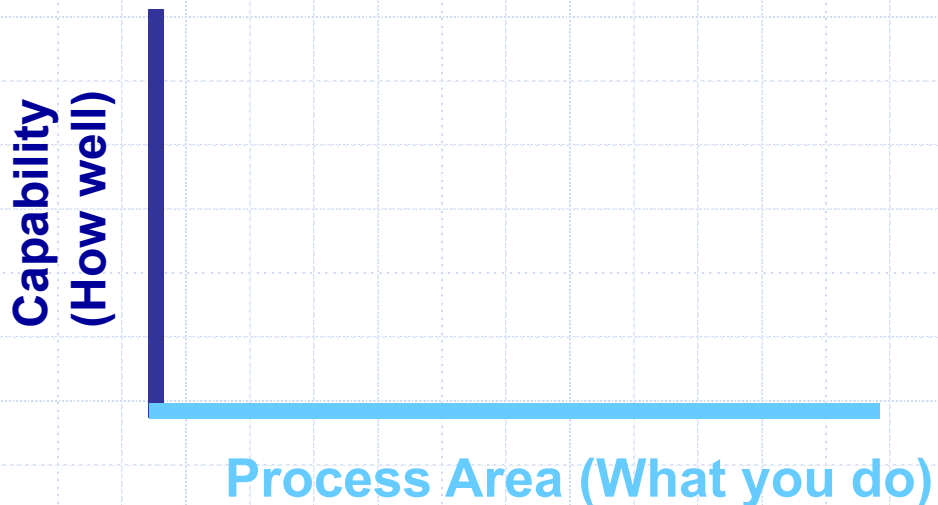
# Outline

- ◆ Software Process Characteristics
- ◆ Evolution of CMMI
- ◆ Structure and Representations of Model
- ◆ CMMI Staged Representation
- ◆ CMMI Continuous Representation

# Process Area Capability Profile

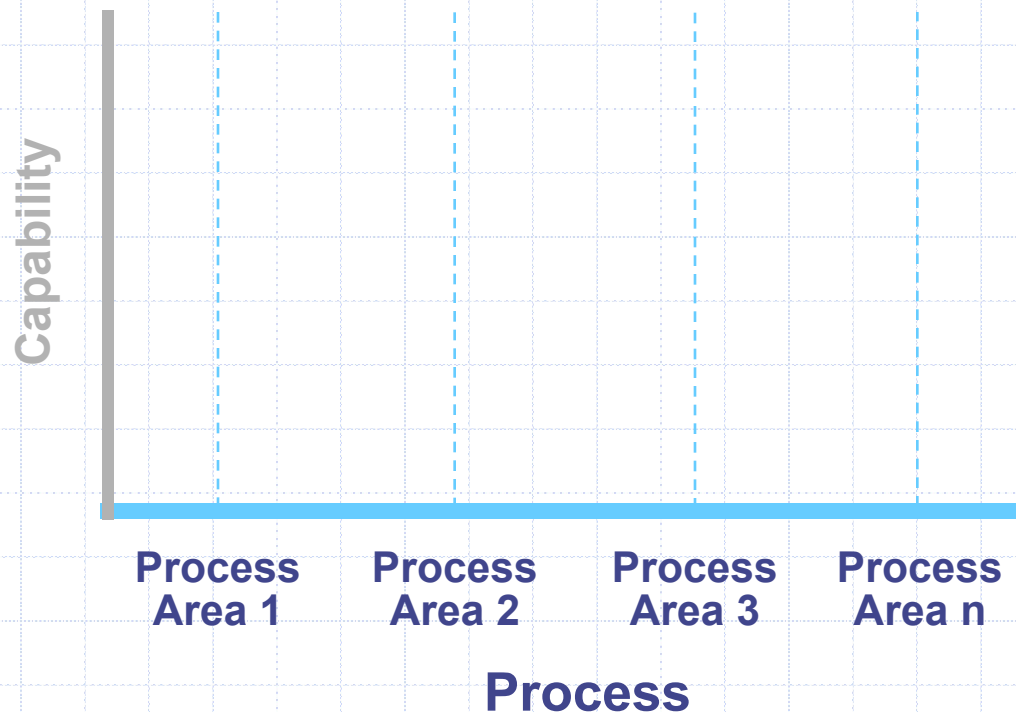
A process area capability profile may be represented by a set of points in two dimensions.

- the *process dimension*
  - ◆ “What” you do
- the *capability dimension*
  - ◆ “How well” you do it



# The Process Dimension

- ◆ The values on this axis describe what processes (described within *Process Areas*) you perform.



# Continuous Organization of Process Areas

## Category

## Process Area

### Project Management

Project Planning  
Project Monitoring and Control  
Supplier Agreement Management  
Integrated Project Management  
Risk Management  
Quantitative Project Management

### Support

Configuration Management  
Process and Product Quality Assurance  
Measurement and Analysis  
Causal Analysis and Resolution  
Decision Analysis and Resolution

### Engineering

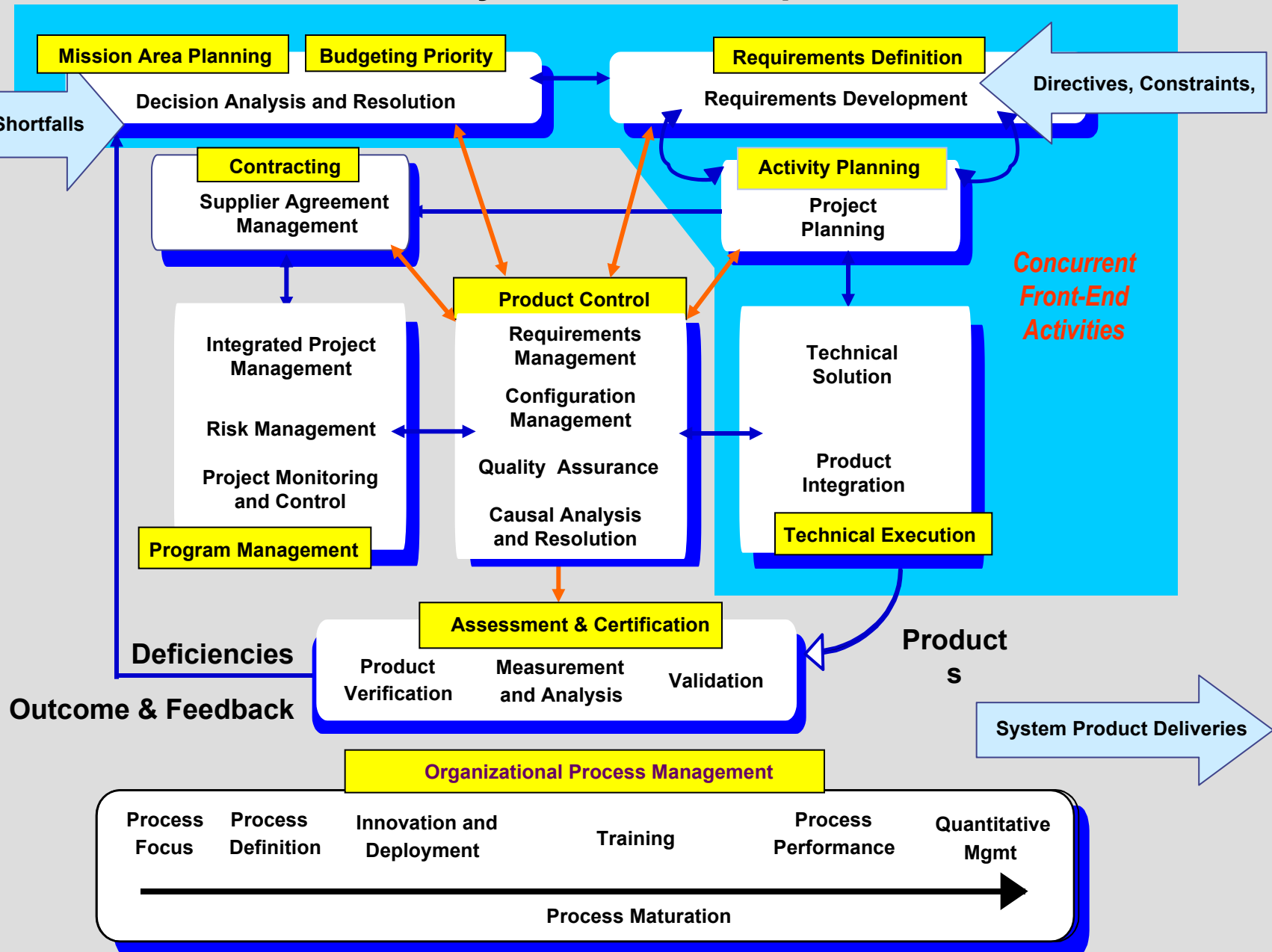
Requirements Management  
Requirements Development  
Technical Solution  
Product Integration  
Verification  
Validation

### Process Management

Organizational Process Focus  
Organizational Process Definition  
Organizational Training  
Organizational Process Performance  
Organizational Innovation and Deployment



# Life Cycle Relationships



# Capability Levels

- ◆ A capability level is a well-defined evolutionary plateau describing the capability of a *process area*.
- ◆ There are **six capability levels**.
- ◆ Each level is a layer in the foundation for continuous process improvement.
- ◆ Thus, capability levels are cumulative, i.e., a higher capability level includes the attributes of the lower levels.

# The Capability Levels

**5 Optimizing**

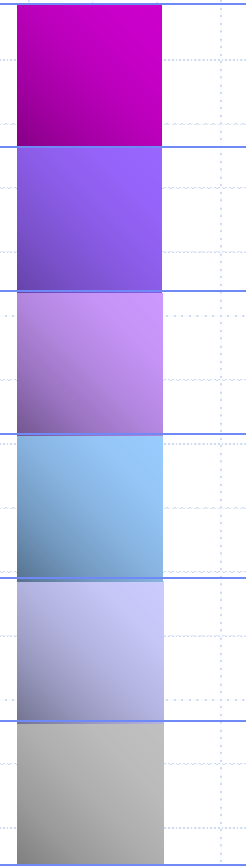
**4 Quantitatively Managed**

**3 Defined**

**2 Managed**

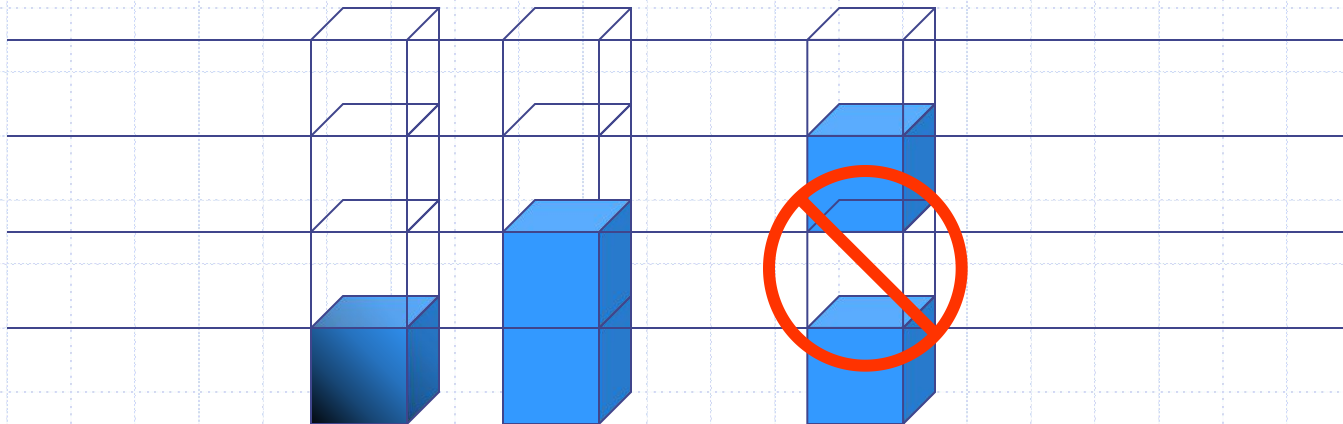
**1 Performed**

**0 Incomplete**



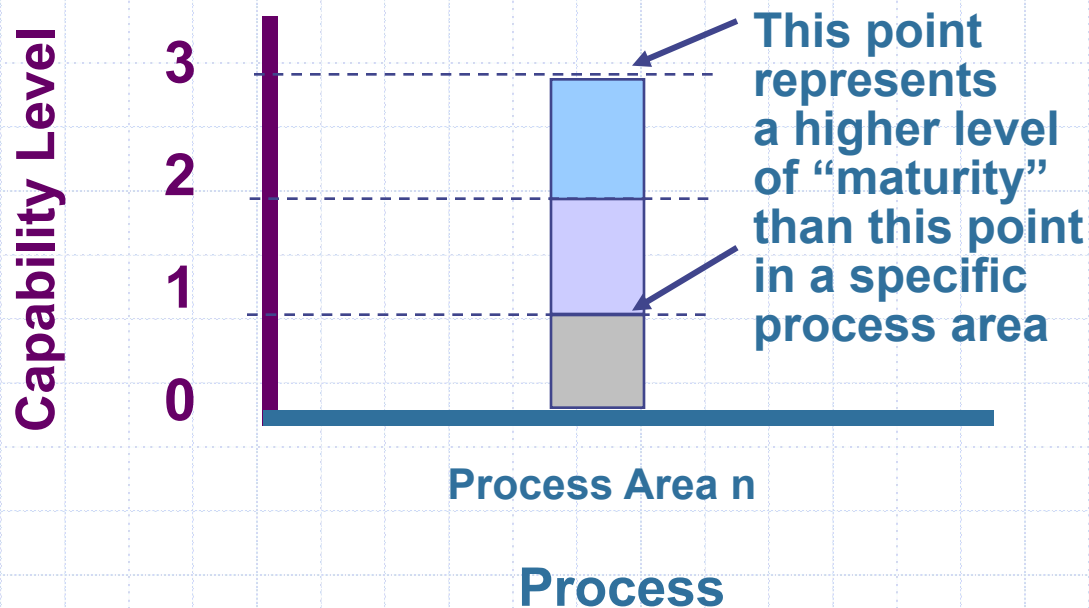
# Capability Levels are Cumulative

- ◆ Because capability levels build upon one another, there can be no gaps.

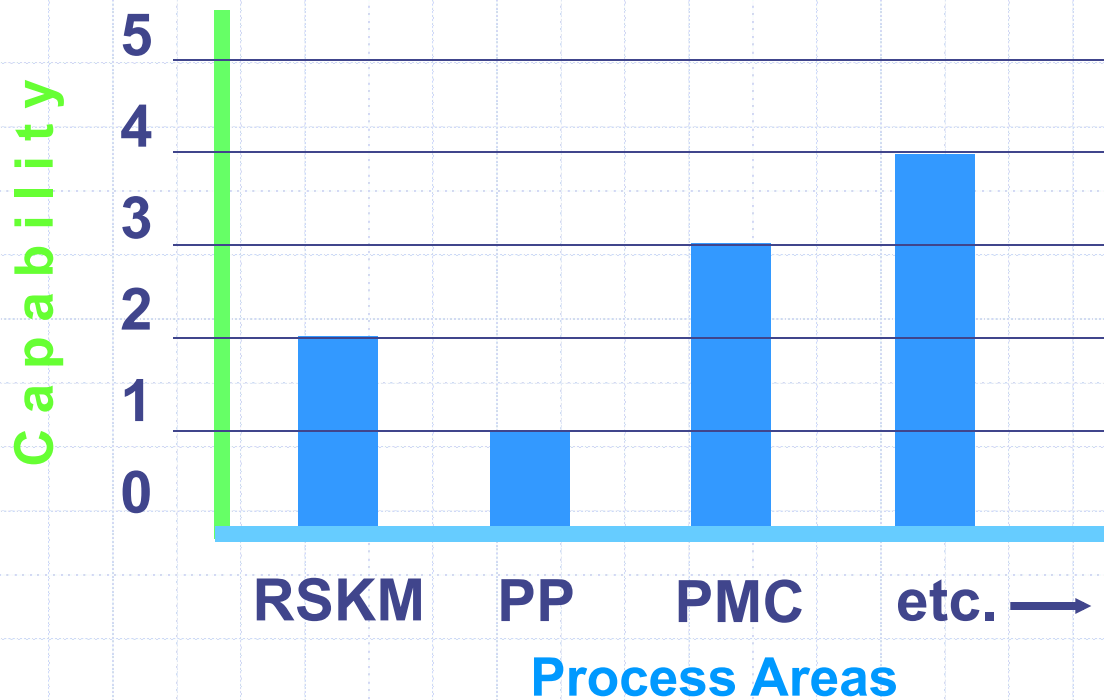


# Representing Capability Levels for a Single Process Area

- ◆ The process area capability of an implemented process can be represented by a bar.



# An Example Organizational Target Profile for Continuous



# Generic Goals and Practices -Continuous

Capability Level:		Generic Goals (GG):	Generic Practices (GP):	
5	(Optimizing)	Institutionalize an Optimizing Process.	Ensure continuous process improvement. Correct root causes of problems.	
4	(Quantitatively Managed)	Institutionalize a Quantitatively Managed Process.	Establish quality objectives for the process. Stabilize subprocess performance.	
3	(Defined)	Institutionalize a Defined Process.	Establish a defined process. Collect improvement information.	
2	(Managed)	Institutionalize a Managed Process.	Establish org. policy. Plan the process. Provide resources. Assign responsibility. Train people. Perform managed process.	Manage configurations. Identify & involve relevant stakeholders. Monitor and control the process. Objectively verify adherence. Review status with mgmt.
1	(Performed)	Achieve Specific Goals.	Perform specific practices.	
0	(Incomplete)	(None)	(None)	

# Critical Distinctions

**performed**

**vs.**

**managed**

--the extent to which the process is planned; performance is managed against the plan; corrective actions are taken when needed

**managed**

**vs.**

**defined**

--the scope of application of the process descriptions, standards, and procedures (i.e., project vs. organization)

**defined**

**vs.**

**quantitatively managed**

--the predictability of process performance

**quantitatively managed**

**vs.**

**optimizing**

--the process is continuously improved by addressing common causes of process variation



# Improving a Process Area

GP1.1 through GP5.2  
All SPs

CL5  
Optimizing

Defect prevention, proactive improvement,  
innovative technology insertion and deployment

GP1.1 through GP4.2  
All SPs

CL4  
Quantitatively  
Managed

Measure process performance,  
stabilize process, control charts,  
deal with causes of special variations

GP1.1 through GP3.2  
All SPs

CL3  
Defined

Project's process is tailored from organization's  
standard processes, understand process qualitatively,  
process contributes to the organizations assets

GP1.1 through GP2.10  
All SPs

CL2  
Managed

Adhere to policy, follow documented plans and processes,  
apply adequate resources, assign responsibility and  
authority, train people, apply CM, monitor, control, and  
evaluate process, identify and involve stakeholders,  
review with management

GP1.1  
All SPs

CL1  
Performed

Perform the work

No GPs or SPs exist

CL0

Not performed, incomplete

# REQM - Capability Levels 1 & 2

## *Requirements Management*

### 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

### Specific practices (CL2)

All SPs

### Generic practices (CL1)

**GP1.1: Perform Specific Practices**

### Generic practices (CL2)

- GP1.1: Perform Specific Practices**
- 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: Manage Configurations**
- 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**

# REQM - Capability Level 3 & 4 & 5

## *Requirements Management*

### Specific practices (CL3)

All SPs

### Generic practices (CL3)

All the CL1 & CL2 Generic Practices plus(+):

GP3.1: Establish a Defined Process  
GP3.2: Collect Improvement Information

---

### Specific practices (CL4)

All SPs

### Generic practices (CL4)

All the CL1 & CL2 & CL3 Generic Practices plus(+):

GP4.1: Establish Quantitative Objectives for the Process  
GP4.2: Stabilize Subprocess Performance

---

### Specific practices (CL5)

All SPs

### Generic practices (CL5)

All the CL1 & CL2 & CL3 & CL4 Generic Practices plus(+):

GP5.1: Ensure Continuous Process Improvement  
GP5.2: Correct Root Causes of Problems

# Summary

- ◆ CMMI models were developed with broad participation and review.
- ◆ The CMMI Model staged Representation
  - Structured for implementation based on proven grouping and ordering of processes.
- ◆ The CMMI Model continuous representation
  - Flexible in its application so the organization can choose which areas to emphasize.
  - Process Areas identify “what you do.”
  - Capability Levels identify “how well you do it.”
- ◆ The CMMI model should be applied using intelligence, common sense, and professional judgment.

# For More Information...

## ◆ For more information about CMMI

- <http://www.sei.cmu.edu/cmmi/> (main CMMI site)

## ◆ Other Web sites of interest include

- <http://seir.sei.cmu.edu/seir/> (Software Engineering Information Repository)
- <http://dtic.mil/ndia> (annual CMMI Technology Conferences)
- <http://seir.sei.cmu.edu/pars> (publicly released SCAMPI appraisal summaries)
- <https://bscw.sei.cmu.edu/pub/bscw.cgi/0/79783>