

# Harmonization of Systems and Software Engineering Processes

James W. Moore, F-IEEE, CSDP  
The MITRE Corporation

June 2007

*The author's affiliation with The MITRE Corporation is provided for identification purposes only, and is not intended to convey or imply MITRE's concurrence with, or support for, the positions, opinions, or viewpoints expressed by the author.*

# Introduction

- **ISO/IEC JTC 1/SC 7 (software and systems engineering) has a large collection of standards.**
  - **Some of the key process standards are difficult to use together.**
- **IEEE Software and Systems Engineering Standards Committee has a large collection of standards.**
  - **Some of the key process standards were adopted (and slightly modified) from SC 7.**
- **The goal of process harmonization is to:**
  - **Create a single definitive set of processes ...**
  - **... described in a set of standards that are easy to use together ...**
  - **... agreed and shared by both organizations.**
- **This presentation describes progress toward that goal.**

# Harmonization of 15288 and 12207

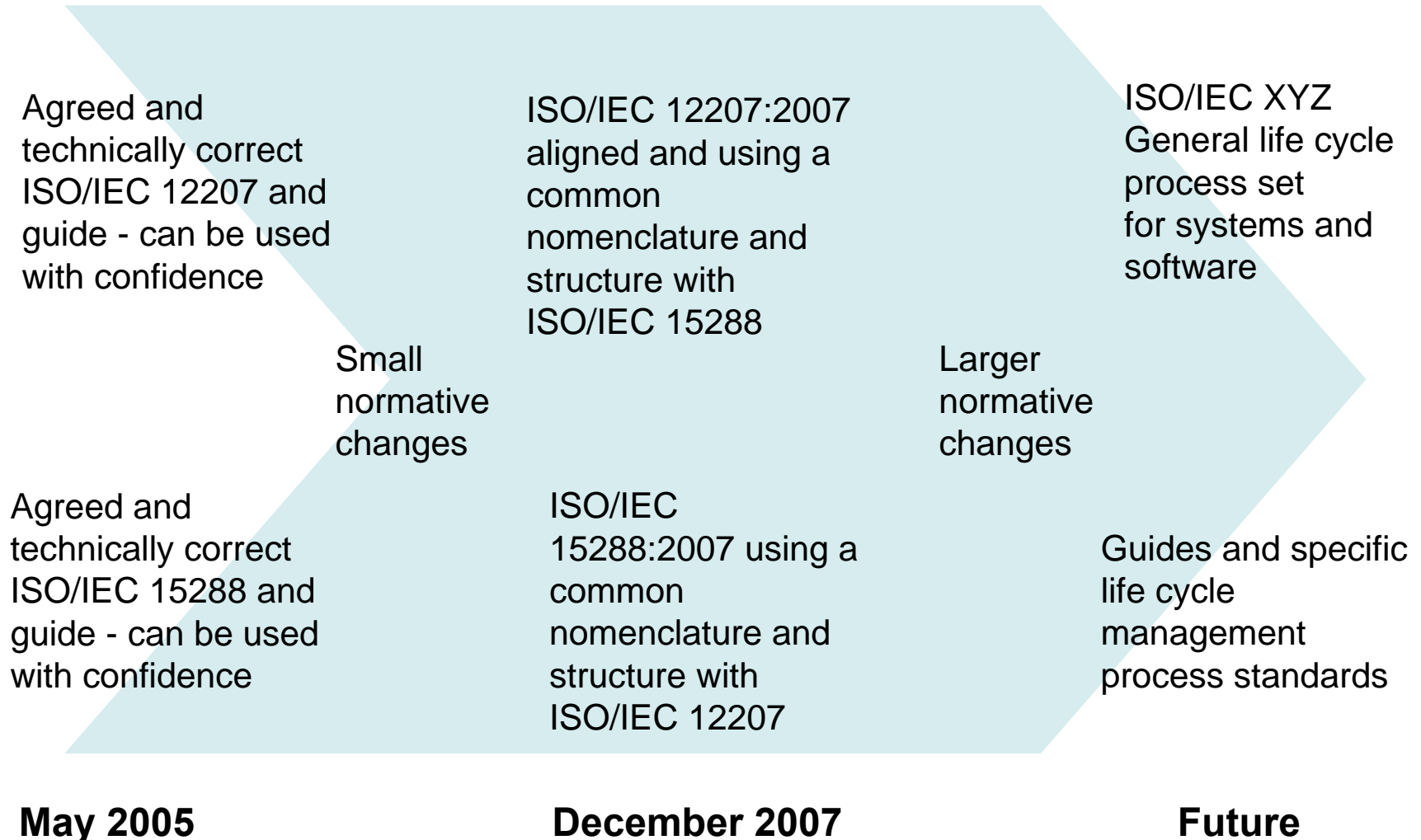
- **Two technically excellent standards are at the center of process harmonization:**
  - ISO/IEC/IEEE 15288:2002, System life cycle processes
  - ISO/IEC 12207:1995, Software life cycle processes, and the substantially identical IEEE/EIA 12207.0:1997
- **A large international user base wants standards that can be used together for development of systems with software content**
  - However, the standards were difficult to use together.
  - Furthermore, a set of amendments to 12207 (for the process assessment community) compounded the difficulty.
- **A two-step project is underway to “harmonize” the two standards.**
  - The first step – aligning the processes – will complete in late 2007.
  - The second step – integration – will seek a single set of shared software/system processes.
- **In addition, a companion project, 24748, is underway. It will provide a guide to life cycle management using the two standards.**

# Status of Harmonization

- A project to solve the problem in “one swell foop” made little progress in two years.
- In May 2005, IEEE Computer Society made a proposal for Harmonization “Lite”.
  - Perform enough harmonization to achieve most of the benefits.
  - Emphasize “alignment” of the standards, rather than complete “integration”.
  - Focus on interoperability of the standards.
  - Focus on backward compatibility with previous versions of the standards
- ISO/IEC JTC 1/SC 7/WG 7 accepted the proposal and formulated a two-phase plan:
  - Phase 1, *Alignment*: Structural refactoring, emphasizing backward compatibility
    - Coincides with normal 5-year maintenance cycle required of international standards.
  - Phase 2, *Integration*: Full integration of processes
    - An integration model will be developed and non-backward-compatible changes will be considered.
- Coordinated projects were initiated in ISO/IEC JTC 1/SC 7 and in IEEE.
- Draft revisions prepared for both standards are now in near-final ballot.
- Completion of alignment phase is expected by year-end 2007.

# Overview of approach for Harmonization

Source: ISO/IEC JTC1/SC7 WG7 briefing material

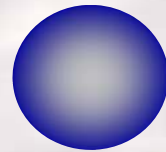


# Many Standards are Names

*A standard is a Name for an otherwise fuzzy concept*

In a complex, multidimensional trade space of solutions ...

... a standard gives a name to a bounded region.

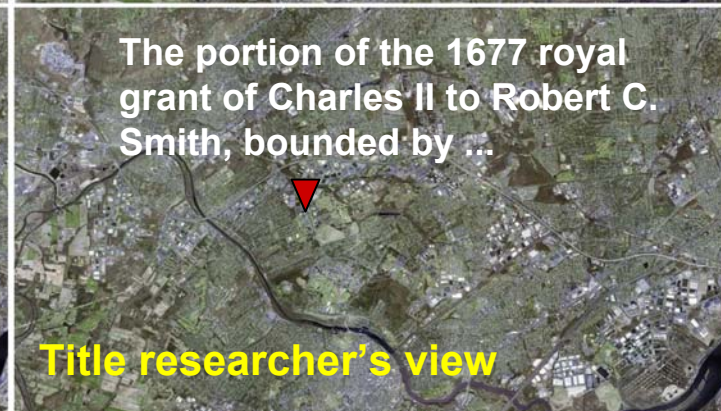
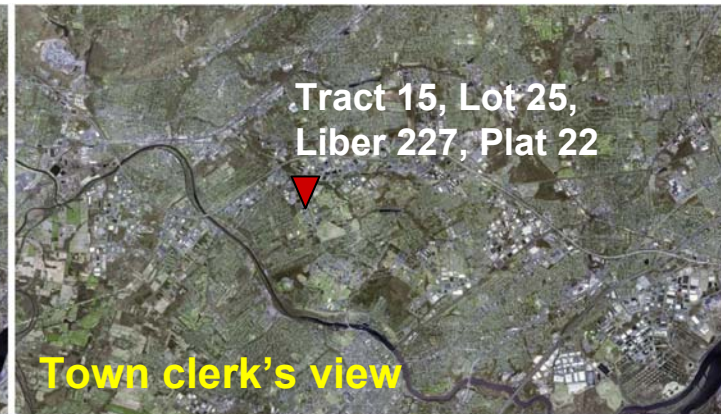


*It defines some characteristics that a buyer can count on.*

- Many software engineering standards assign names to practices or collections of practices.
- This enables communication between
  - Buyer and seller
  - Government and industry
  - Insurer and insured

# Names are Important

*We use names to localize the subject under discussion. But sometimes confusion results because we use different name spaces.*



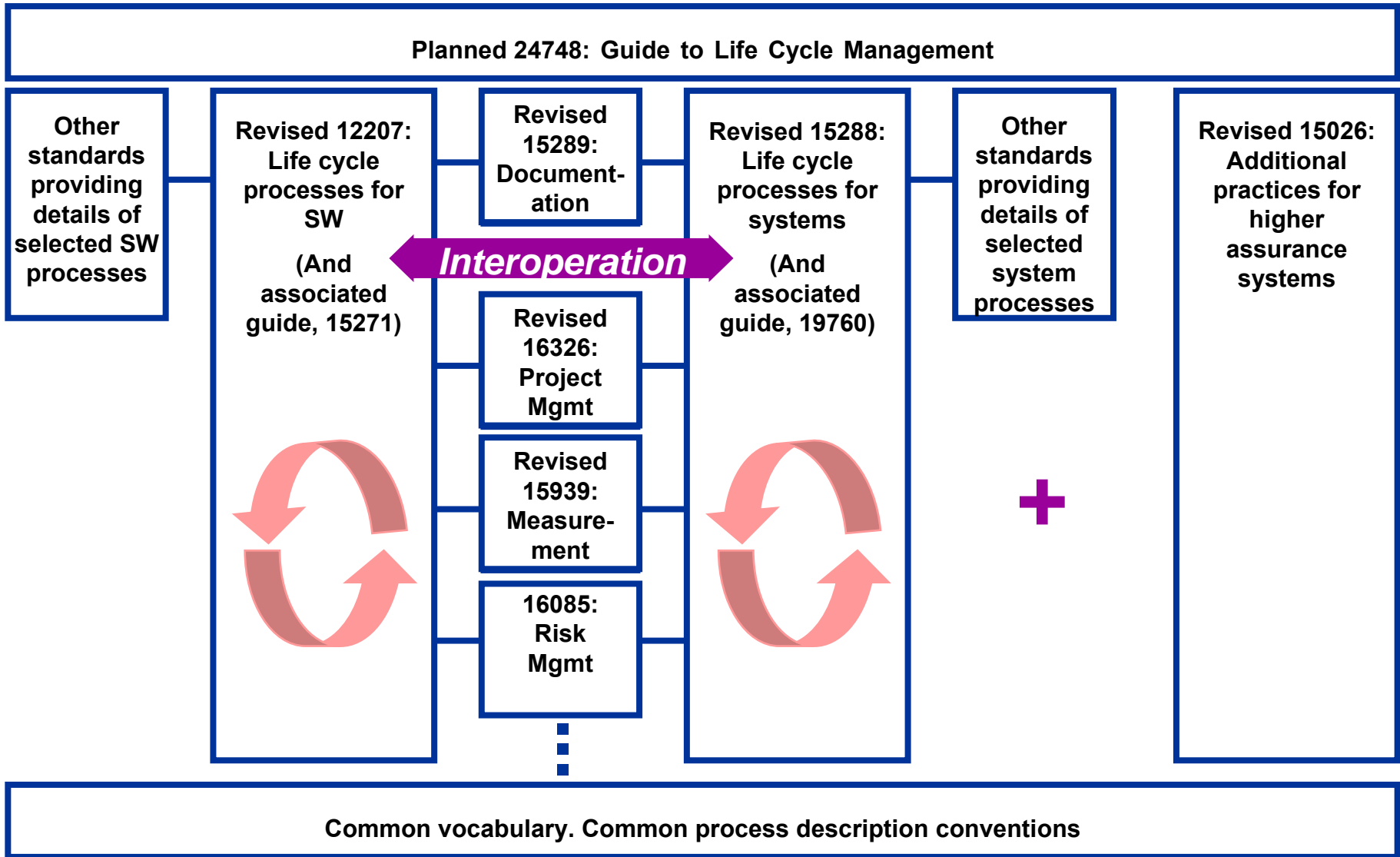
Would you know that these are different names for the same thing? Would you know without the map?

# 15288 and 12207 Give “Names” to Processes

- ISO/IEC 15288:2002 gives names to 25 processes in the life cycle of a system. It is more *descriptive* than 12207.
- ISO/IEC 12207:1995 gives names to 17 processes in the life cycle of a software product or service. It is more *prescriptive* than 15288.
  - (Two amendments re-described the processes for assessment purposes.)
- The names are important so that acquirers and suppliers can communicate regarding their practices.
  - “Oh, when you say ‘implementation’, you include ‘testing’? No, no, no, that’s a separate thing; our contract doesn’t include that!”
- The names are important as a basis for process evaluation and improvement.
- The names are important to provide a context for implementing *improved practices*. – **Our goal.**



# Intended Relationships of Key System and Software Engineering Life Cycle Process Standards



# The Harmonization Problem

**Starting from a diverse set of standards with ...**

- different terms
- different process sets
- different process architectures
- different levels of prescription, and
- different audiences\*



**... develop a revised set of standards with ...**

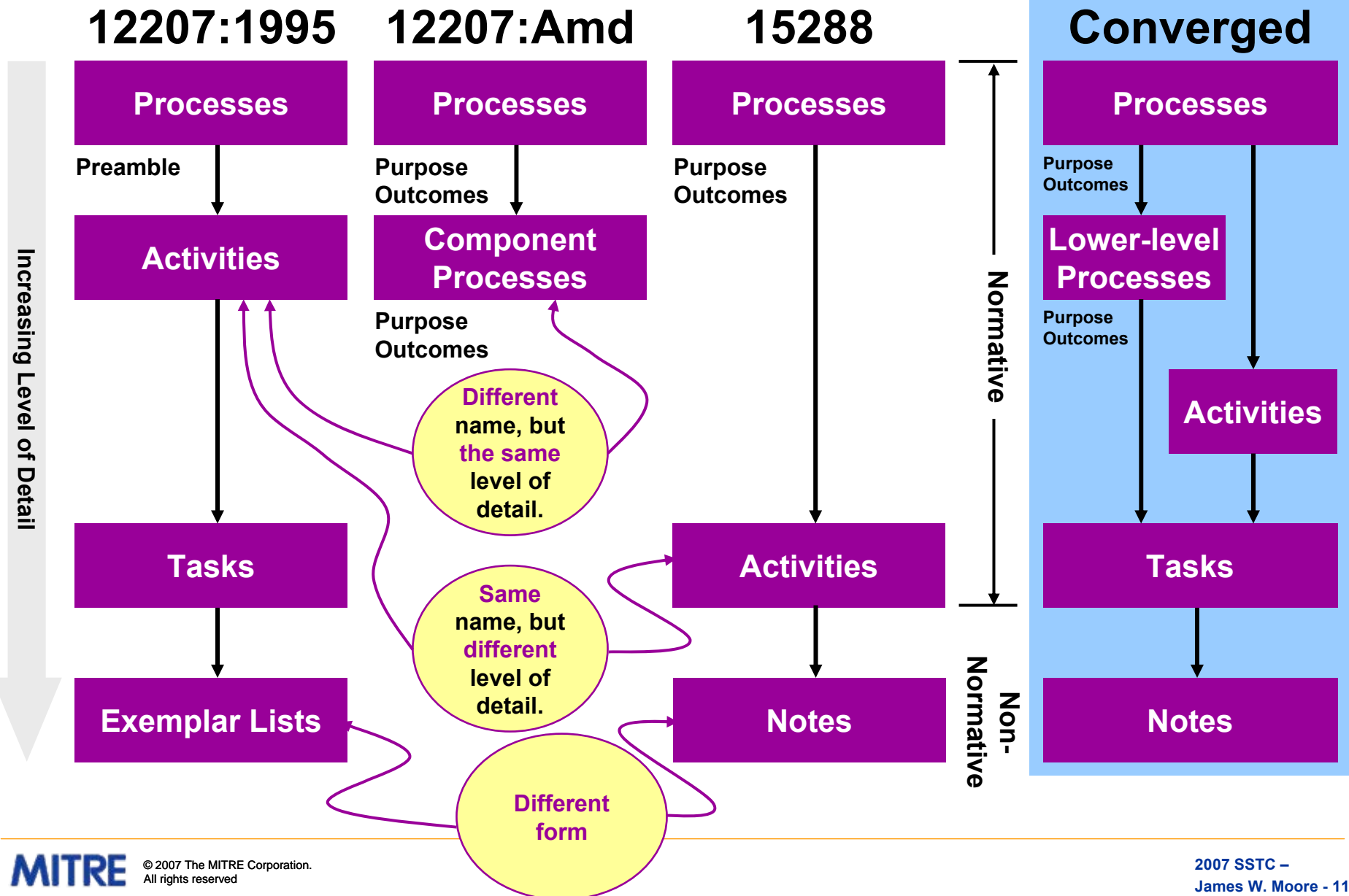
- a single vocabulary
- a single process set
- a single, uniform architecture
- a shared level of prescription, and
- suitability across the audiences

**... all without needlessly disrupting current organizational investment based on usage of the current set of standards.**

**\* Systems versus software and process definition versus process assessment**

# Architectural Constructs

*It is **not** required to use all architecture constructs in any particular document or for any particular process.*



This chart is an overview.  
There are exceptions to the  
general cases.

# Relating Old Content to New Content

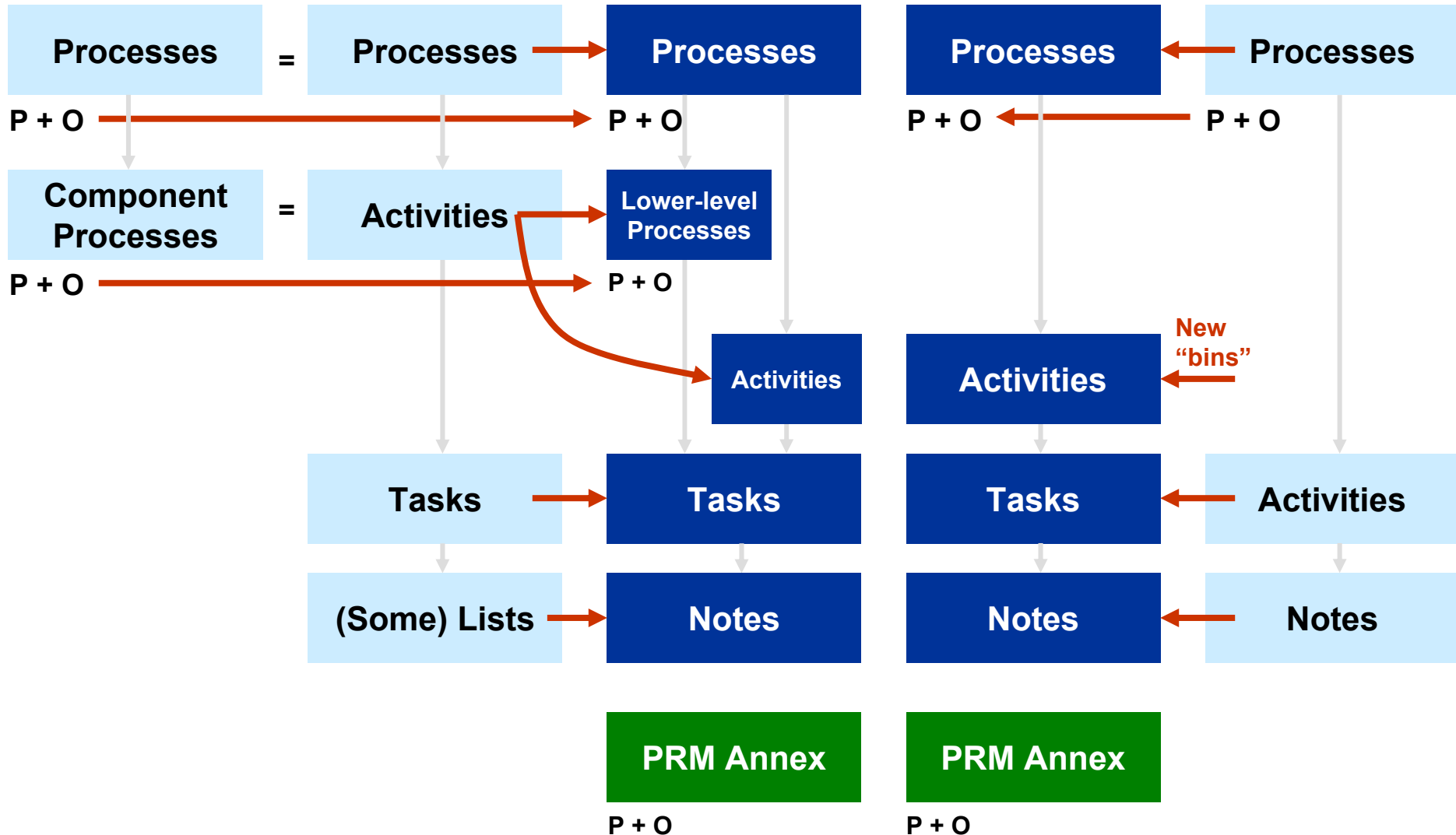
12207:Amd

12207:1995

12207R

15288R

15288:2003



# Example Process: 15288 Stakeholder Requirements Definition Process

## 6.4.1.1 Purpose

The purpose of the Stakeholder Requirements Definition Process is to define the requirements for a system that can provide the services needed by users and other stakeholders in a defined environment.

## 6.4.1.2 Outcomes

As a result of the successful implementation ... :

- a) The required characteristics and context of use of services and operational concepts are specified.
- b) The constraints on a system solution are defined.
- c) Traceability of stakeholder requirements to stakeholders and their needs is achieved.
- d) The stakeholder requirements are defined.
- e) Stakeholder requirements for validation are identified.

## 6.4.1.3 Activities and Tasks

The project shall implement the following activities and tasks ... :

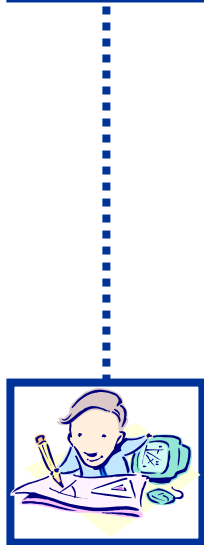
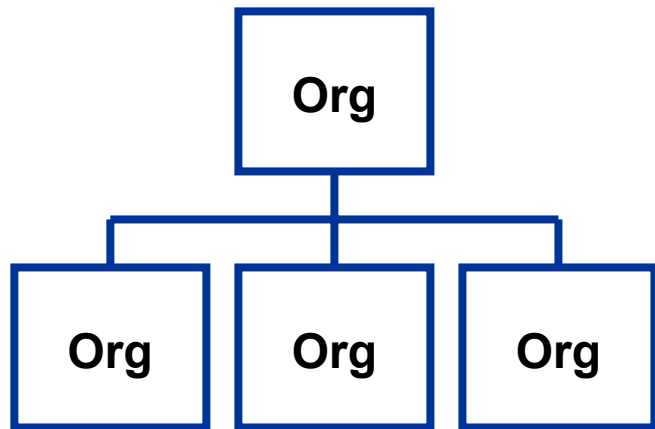
- a) Stakeholder Requirements Elicitation. This activity consists of the following tasks:
  - 1) Identify the individual stakeholders or stakeholder classes who have a legitimate interest in the system throughout its life cycle.  
**NOTE** This includes, but is not limited to, users, operators, ...
  - 2) Elicit stakeholder requirements.
- b) Stakeholder Requirements Definition ...
- c) Etc.

**Caution: This wording may change during balloting.**

# Key Terminology and Concepts

- **Organization**: a person or a group of people and facilities with an arrangement of responsibilities, authorities and relationships [ISO 9000]
  - A part of an organization is an organization if it meets the definition.
  - An *individual* can be an organization if s/he meets the definition.
- **Party**: an organization entering into an agreement
- **Project**: an endeavour with defined start and finish dates undertaken to create a product or service in accordance with specified resources and requirements [adapted from ISO 9000]

# Key Terminology and Concepts



An *individual* can be an *organization*



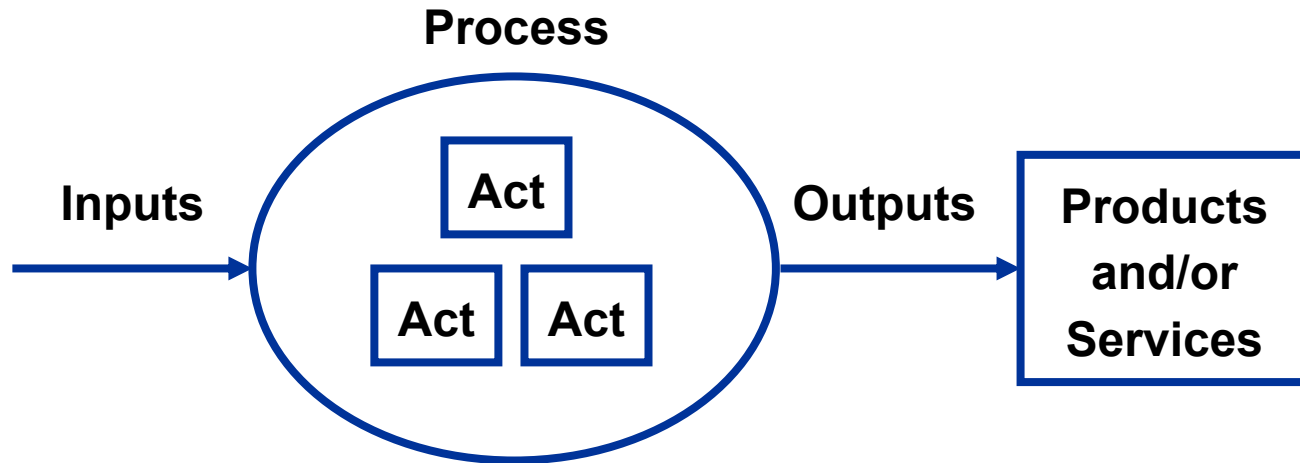
*Organizations* conduct *projects* to do things, notably to deal with *systems*.

*Organizations* make *agreements* to *acquire* and *supply products* and *services*.



Agreeing *organizations* are called *parties*.

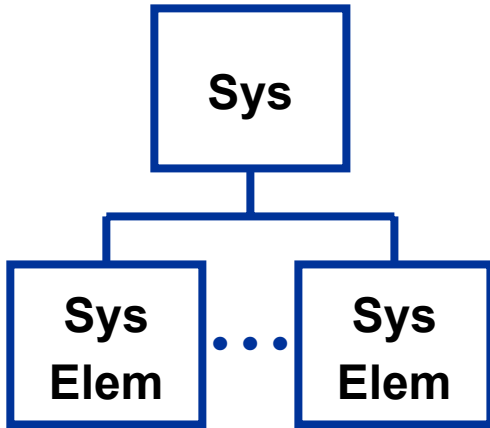
# Key Terminology and Concepts



- **Process:** set of interrelated or interacting *activities* which transforms inputs into outputs [ISO 9000]
- **Product:** the result of a process [ISO 9000]
- **Service:** performance of activities, work, or duties associated with a product

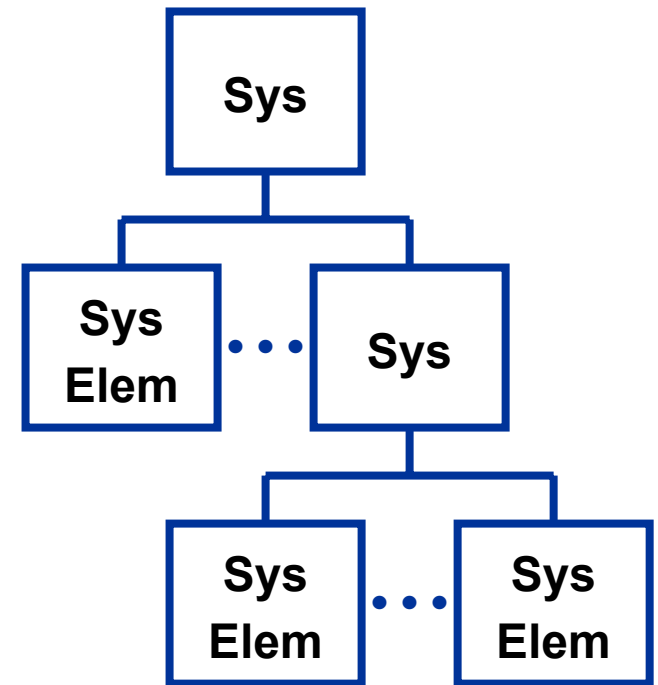


# Key Terminology and Concepts

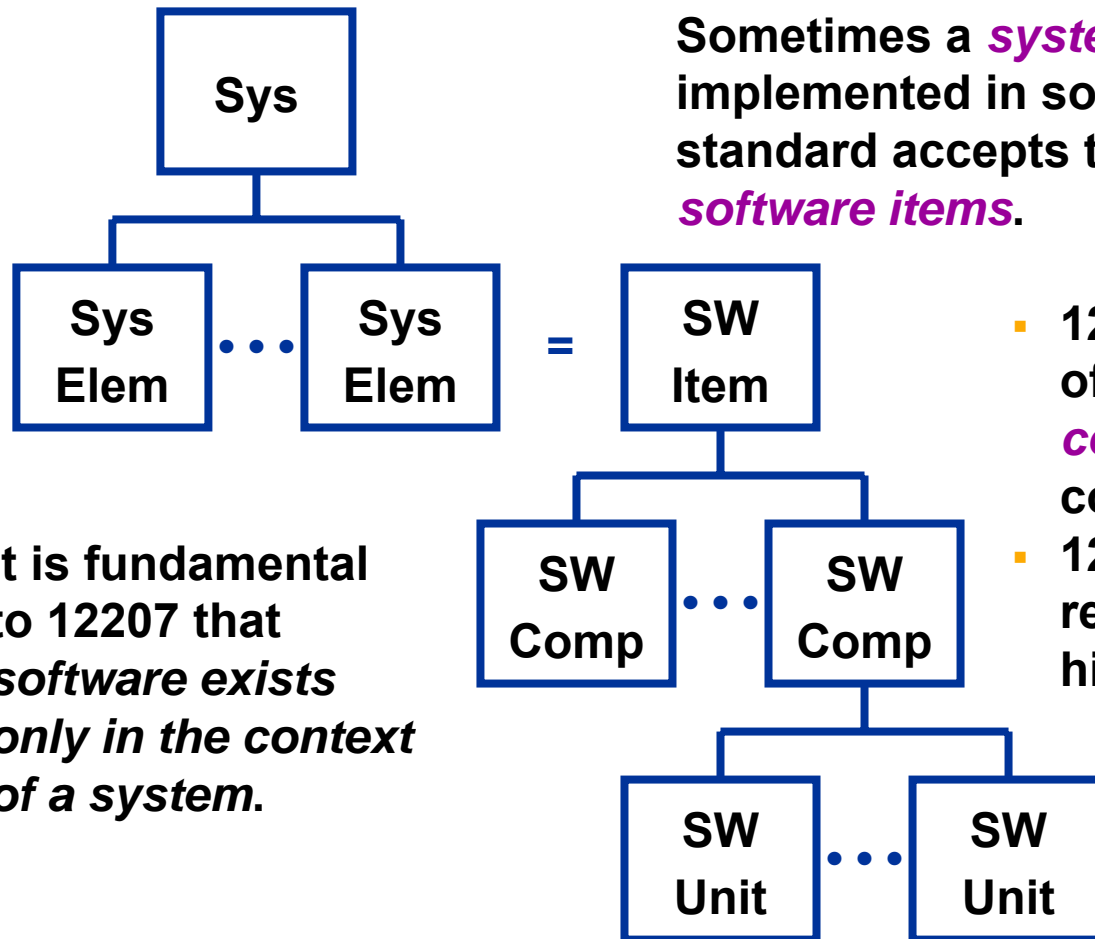


A *system* is composed of *system elements*. Each element is *implemented* and then *integrated* into the system. One invocation of 15288 suffices to create a system composed of a set of elements.

However, 15288 states that a system element can itself be regarded as a system. So, 15288 can be invoked recursively to create a hierarchy of systems and their elements. A hierarchy of systems often implies a hierarchy of projects.



# Key Terminology and Concepts



Sometimes a *system element* is to be implemented in software. The 12207 standard accepts this as one or more *software items*.

It is fundamental to 12207 that *software exists only in the context of a system*.

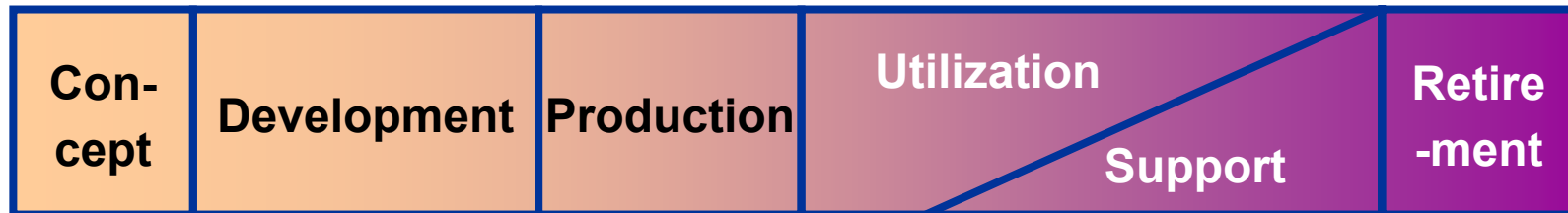
- 12207 uses a hierarchy of *items* – composed of *components* – composed of *units*.
- 12207 is *not* invoked recursively to create this hierarchy.

# Key Terminology

- Every *system* has a *life cycle* which is viewed as composed of *stages*. (The standards do not require a particular set of stages.)
  - Each stage has a purpose and makes a contribution to the life cycle.
- Stages are initiated and terminated by *decision gates*.
- Stages may overlap and may be concurrent.
- The purpose of each stage is accomplished by executing *processes*.
- Any process may be useful in any stage.

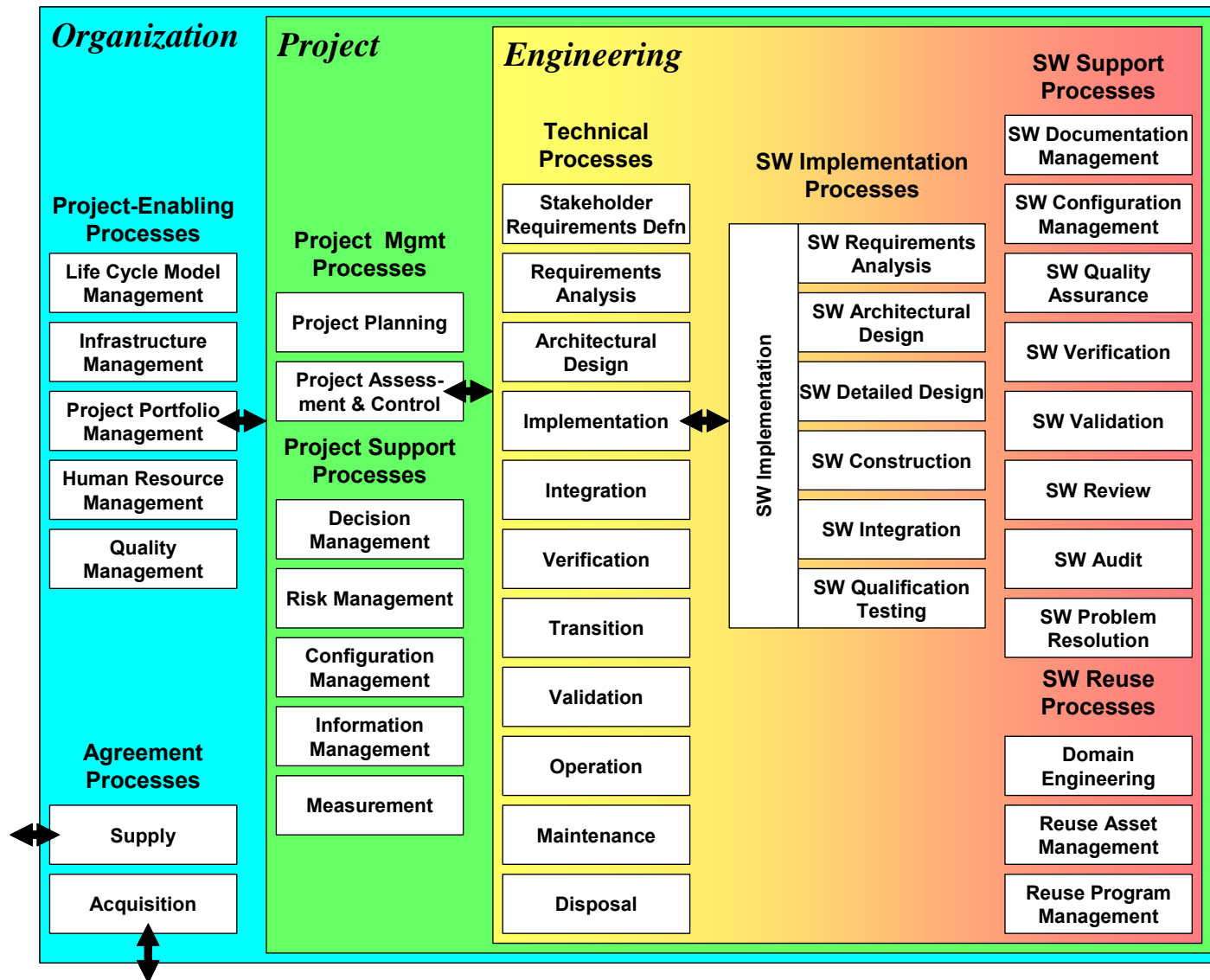
This is important.

*A typical set of life cycle stages*

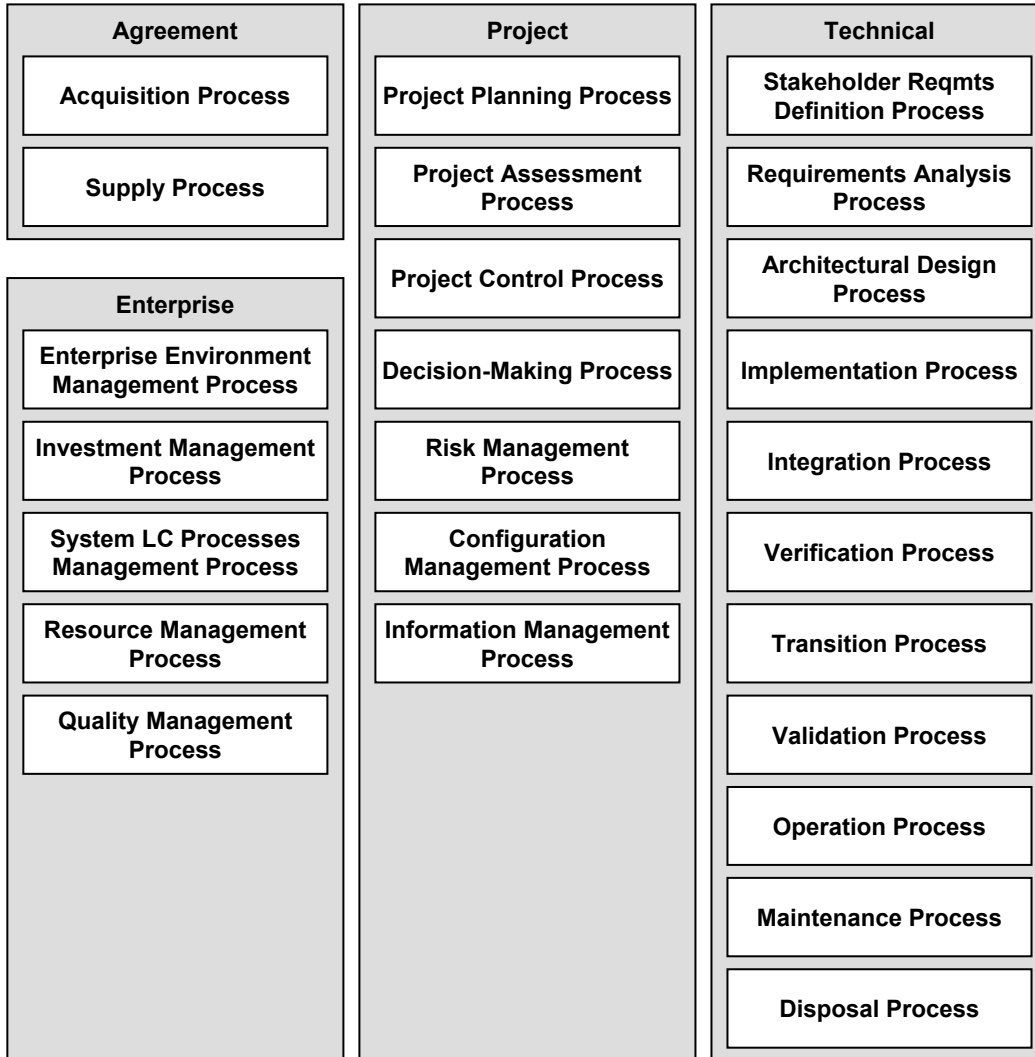


- It is a common error to talk about life cycle stages when one really means processes or vice-versa.
- Locating practices with respect to processes provides much greater precision.

# Anticipated Result of Process Alignment

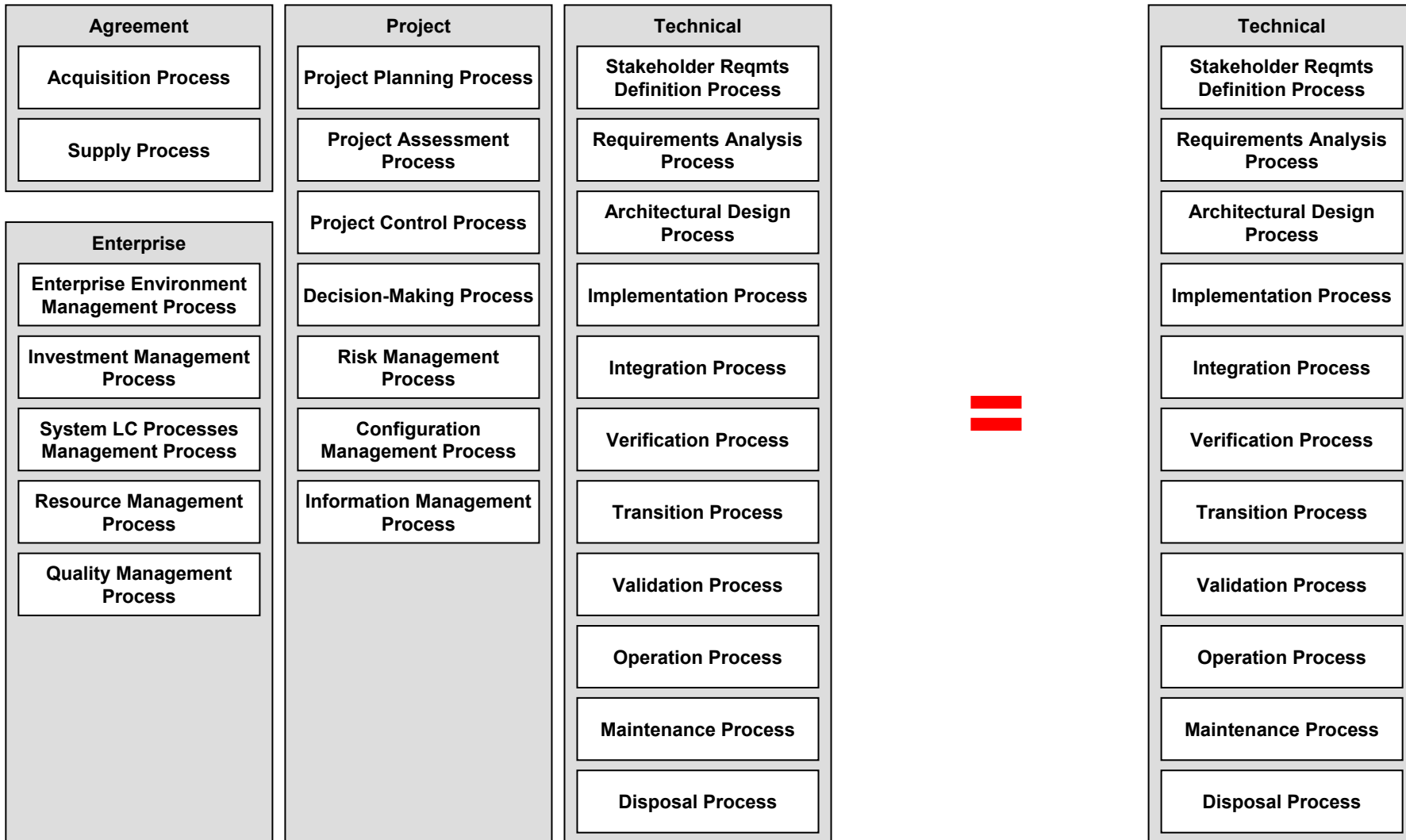


# The Life Cycle Processes of 15288:2002



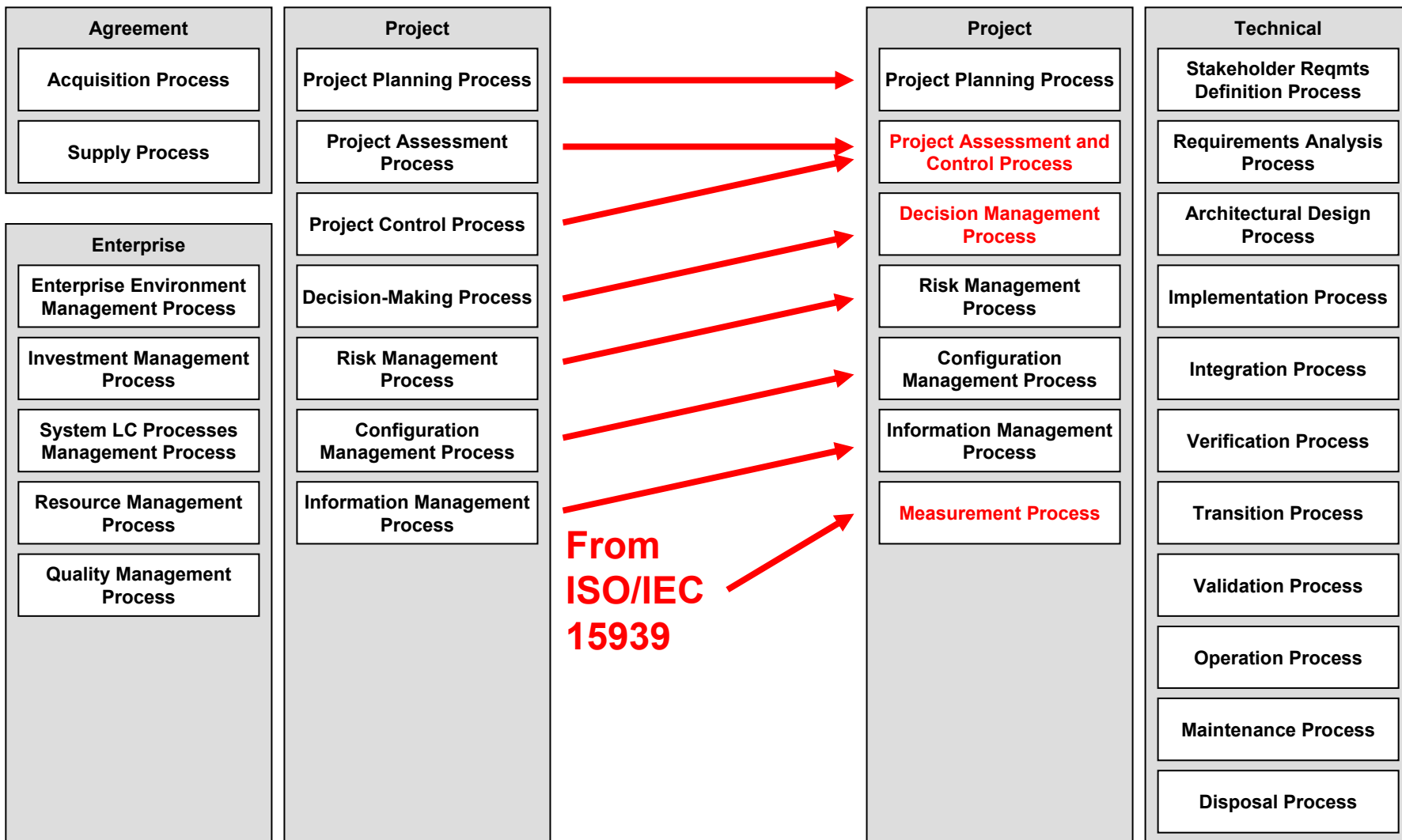
# Building 15288:2007 – Technical Processes

15288:2007 has the same set of technical processes as 15288:2002



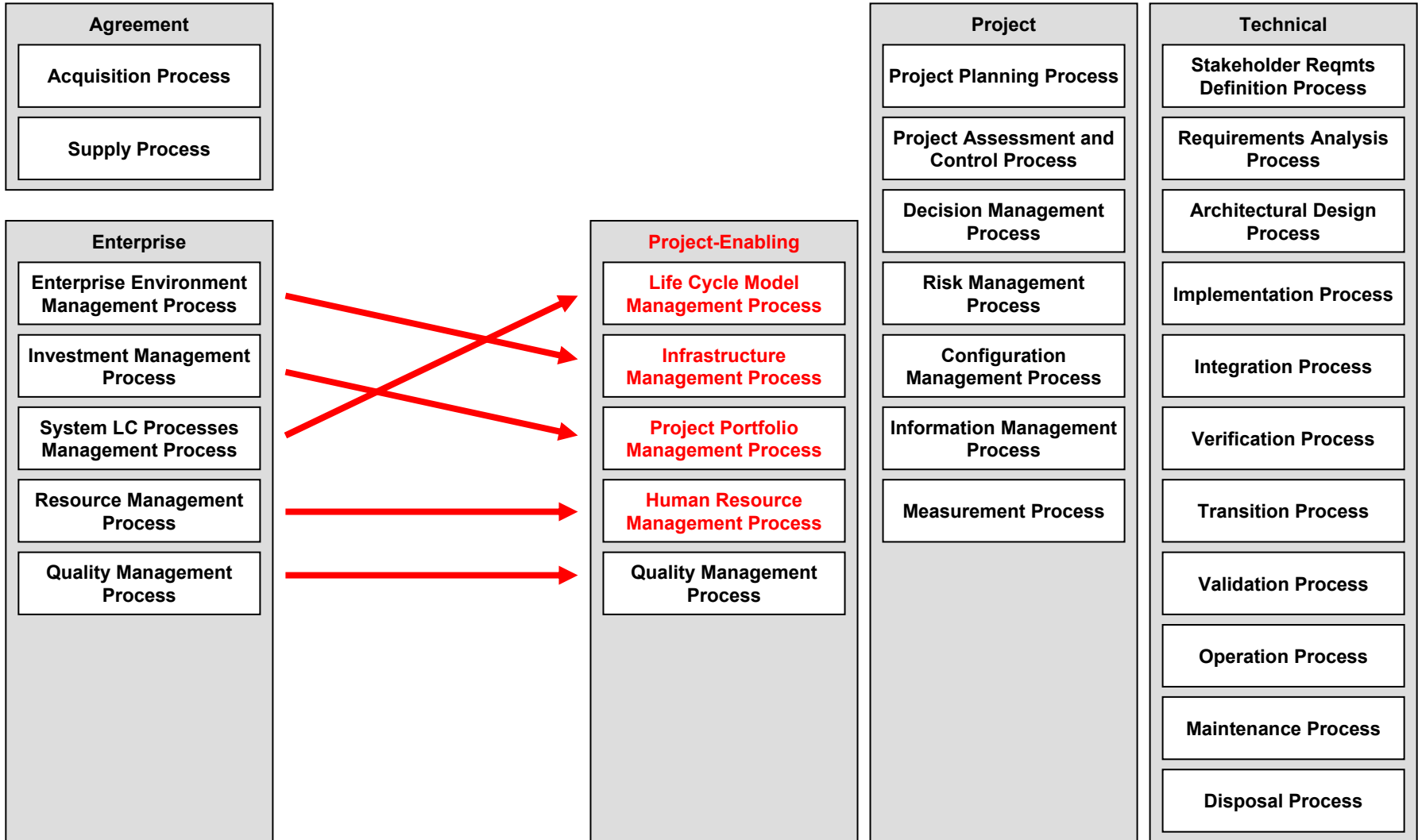
# Building 15288:2007 – Project Processes

15288:2007 has a similar set of project processes as 15288:2002



# Building 15288:2007 – Project-Enabling Processes

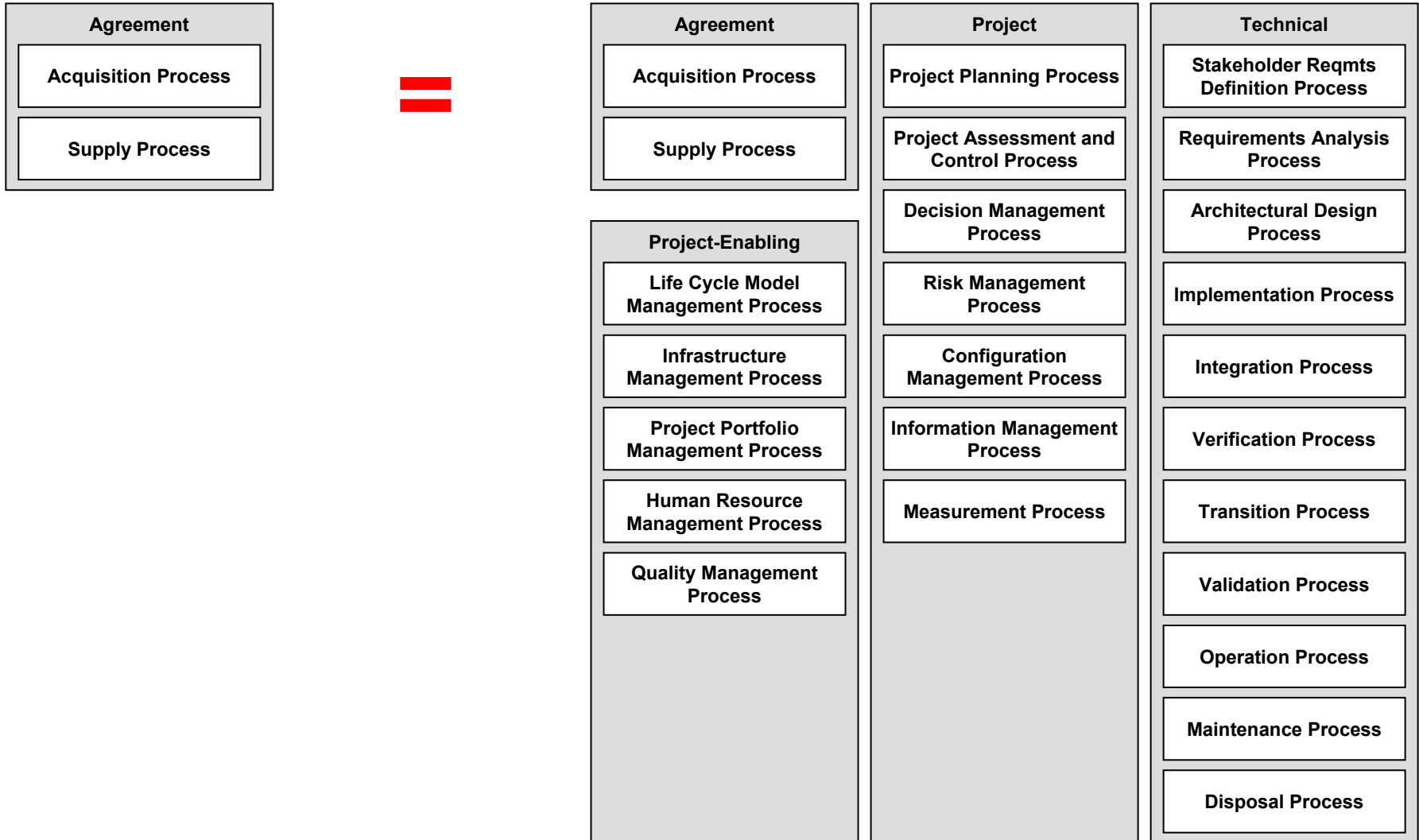
15288:2007 has a similar set of project-enabling processes as 15288:2002



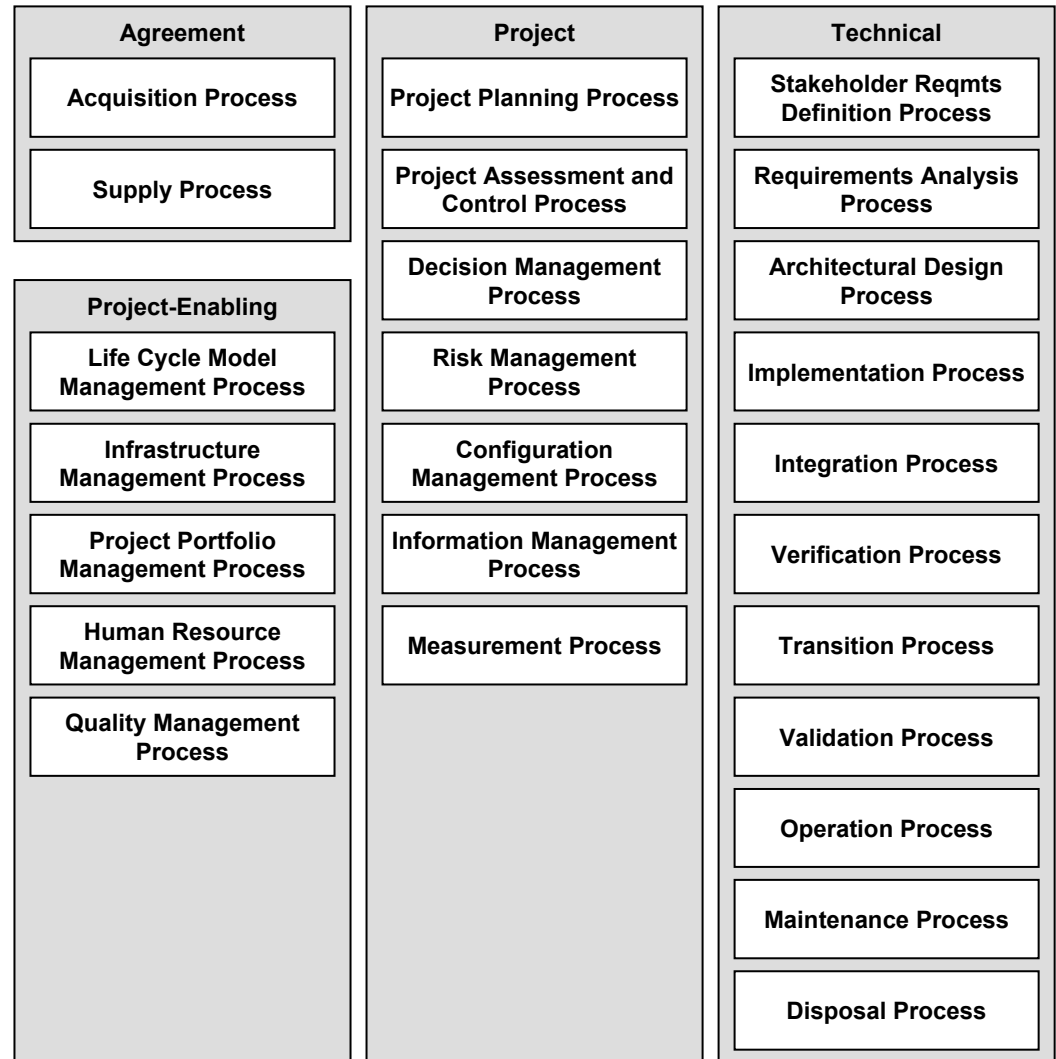


# Building 15288:2007 – Agreement Processes

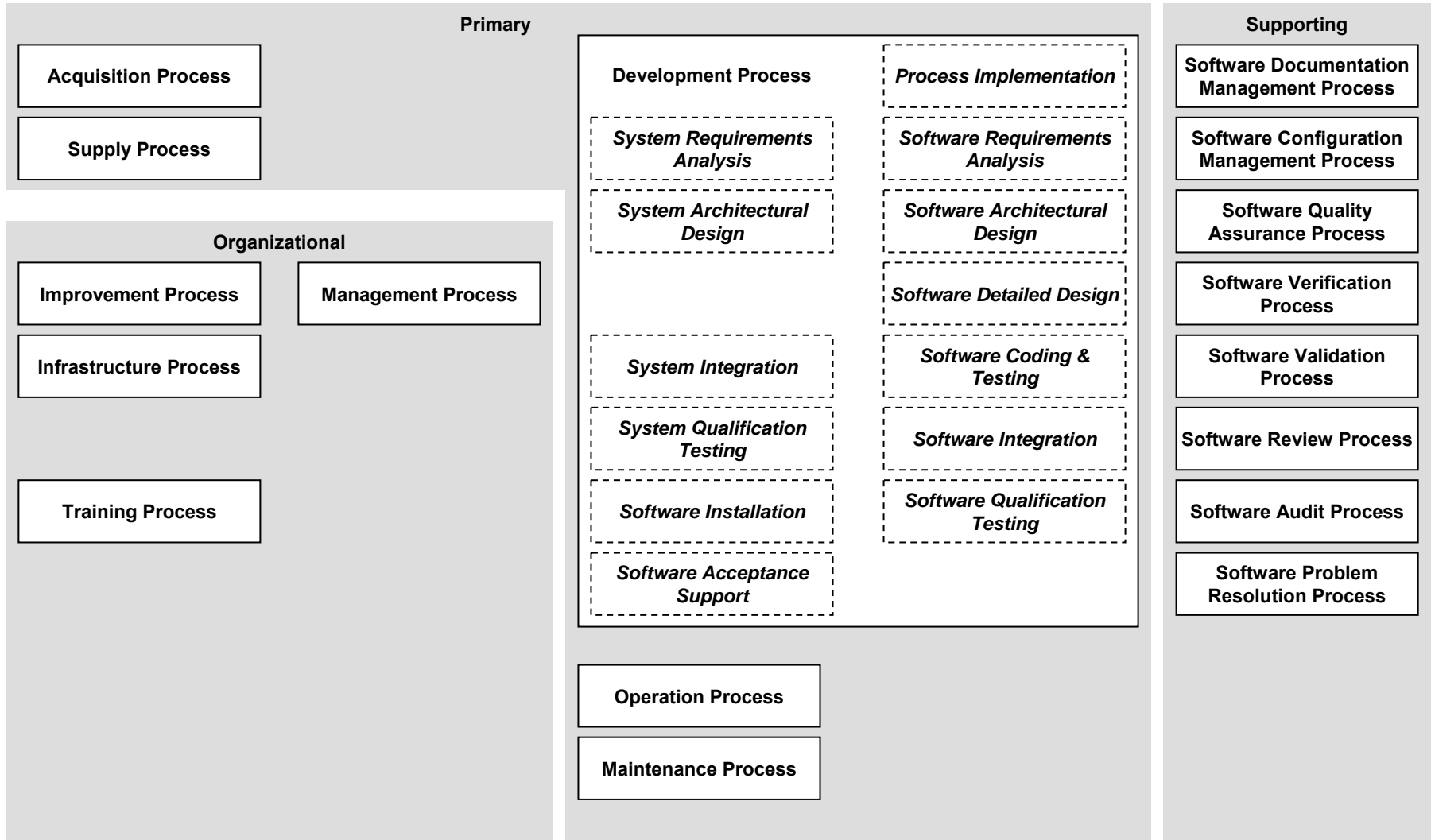
15288:2007 has the same set of agreement processes as 15288:2002



# The Life Cycle Processes of 15288:2007

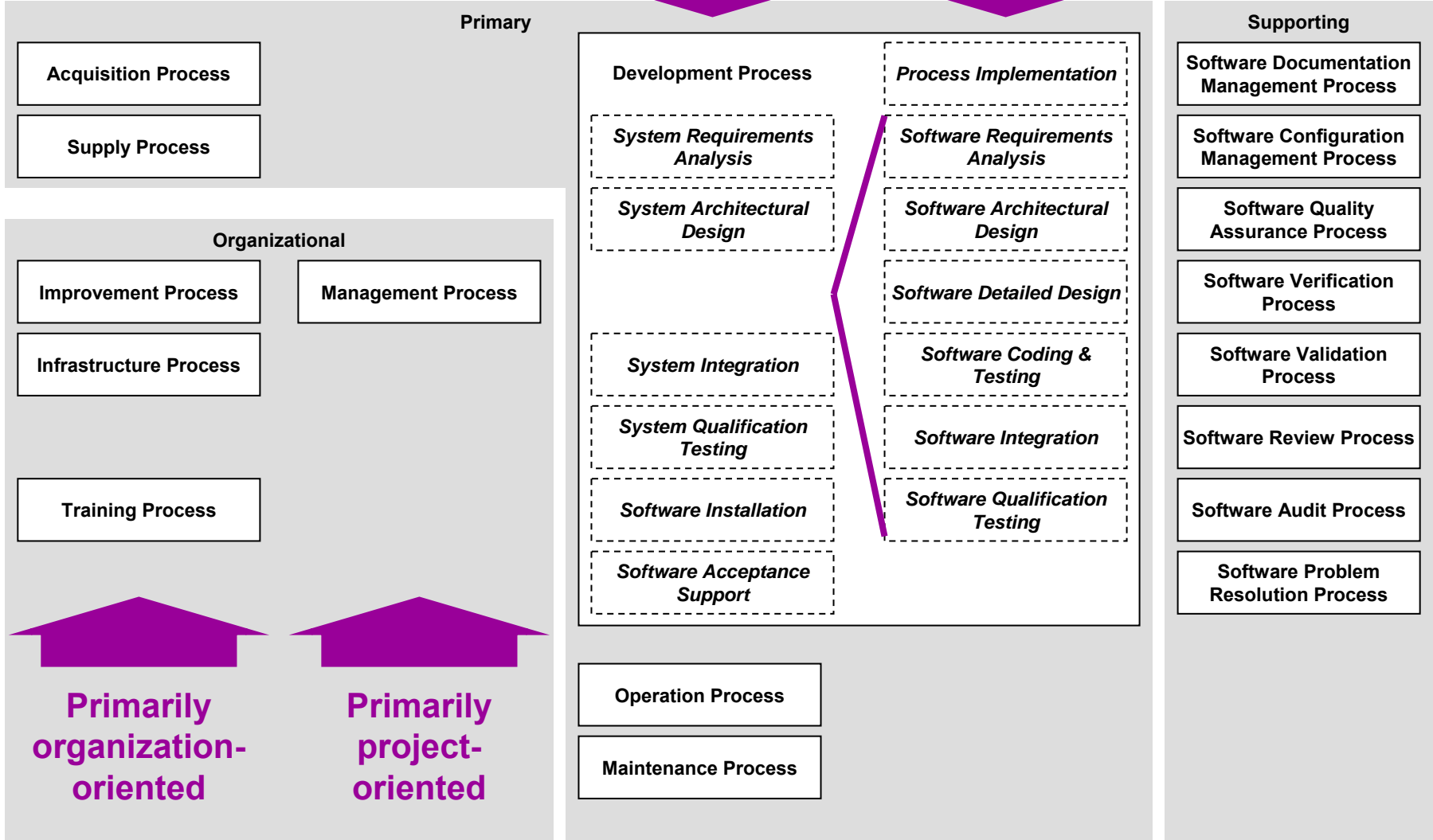


# The Life Cycle Processes of 12207:1995



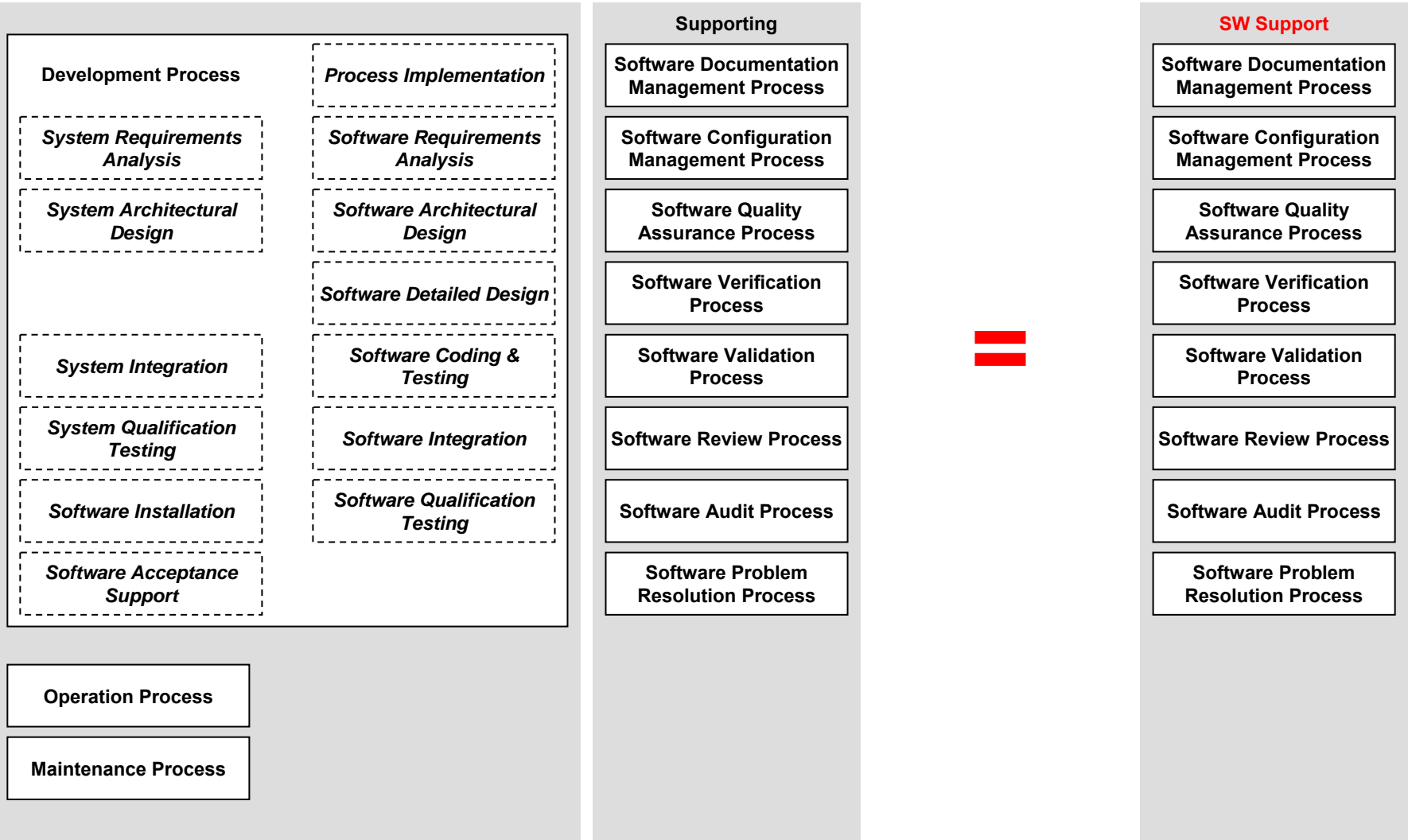
# System Context Activities

# Software Activities



# Building 12207:2007 – Software Support Processes

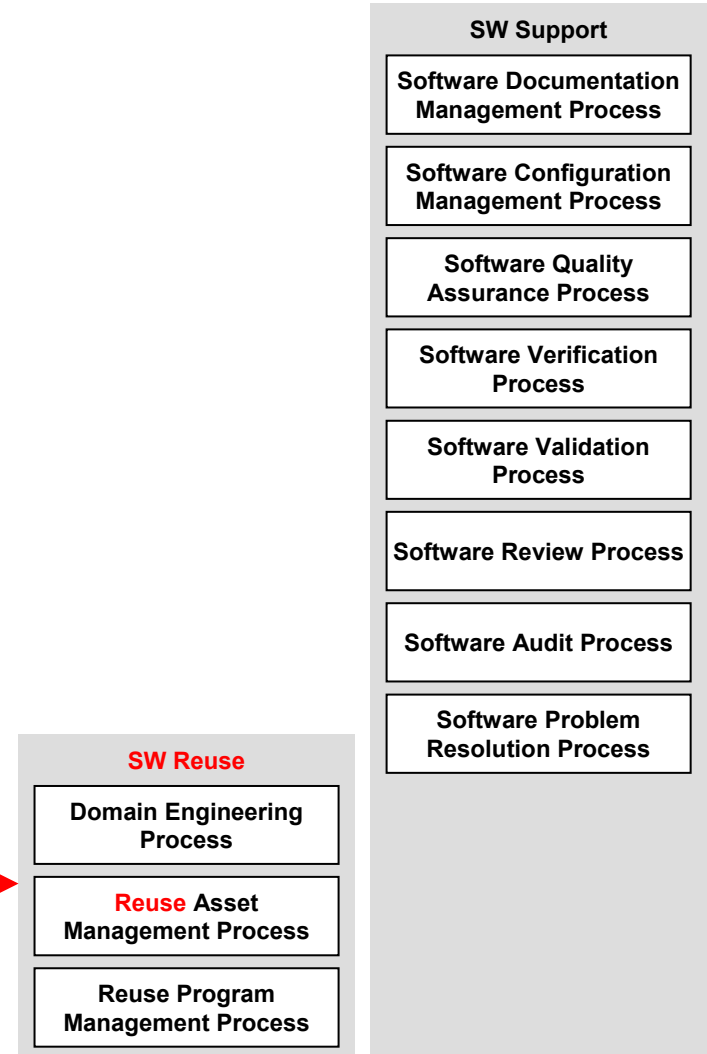
12207:2007 has the same set of SW support processes as 12207:1995



# Building 12207:2007 – Software Reuse Processes

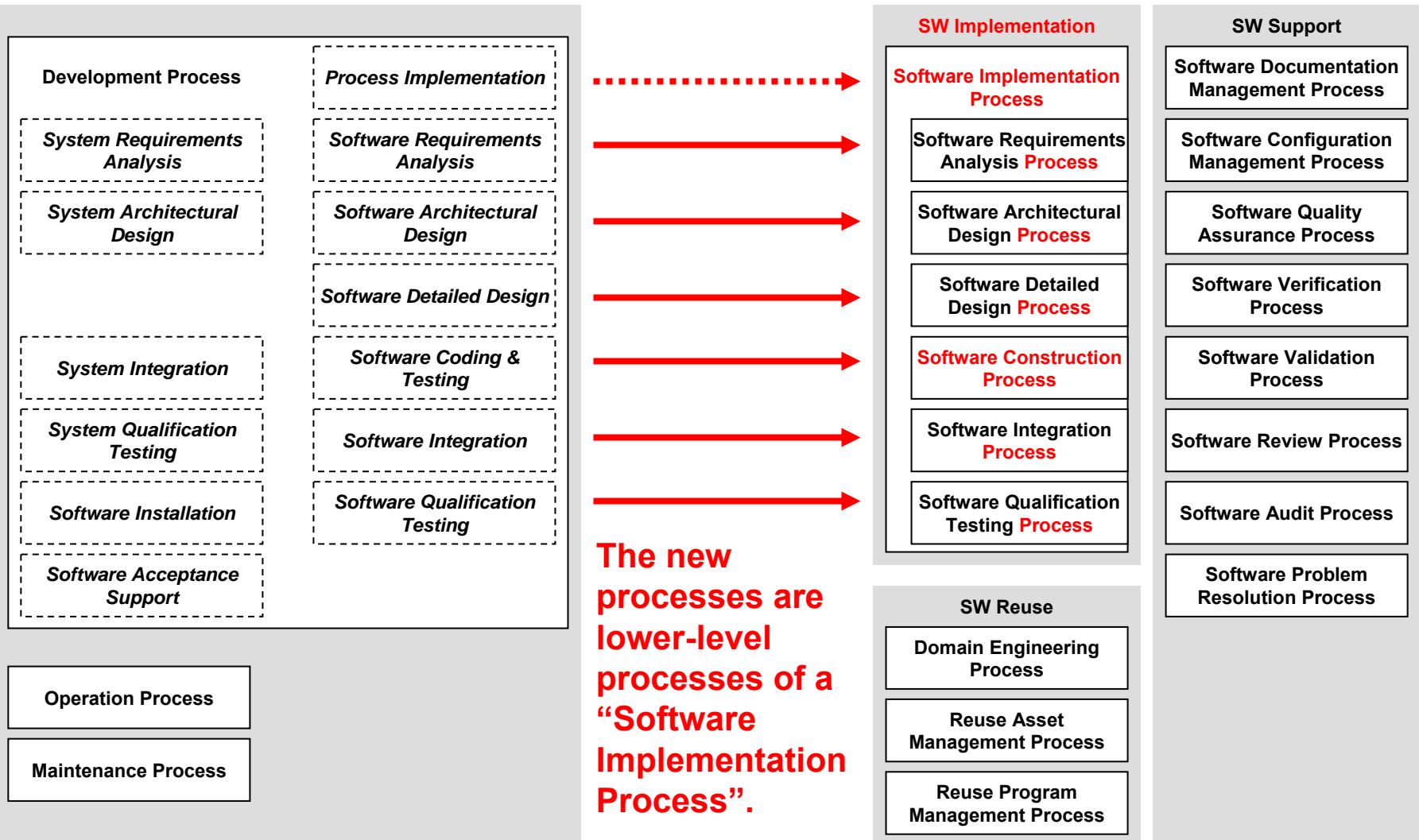
12207:2007 has a similar set of SW reuse processes as 12207 Amd.1 (which came from IEEE Std 1517).

From IEEE  
Std 1517 and  
from 12207  
Amd



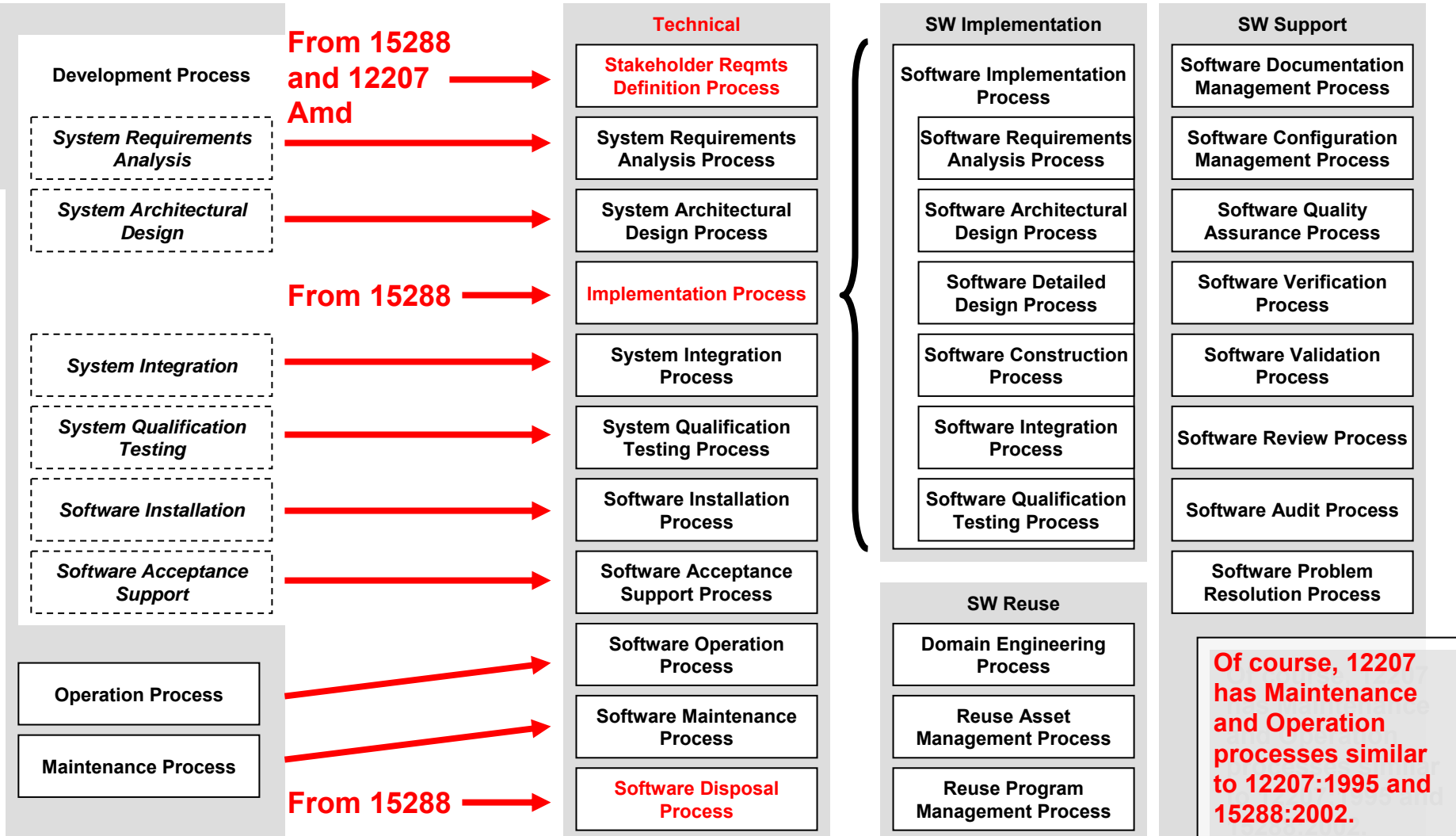
# Building 12207:2007 – Software Implementation Processes

12207:2007 has a set of SW implementation processes similar to activities of the development process of 12207:1995 and component processes of 12207 Amd.1.



# Building 12207:2007 – 12207 Technical Processes

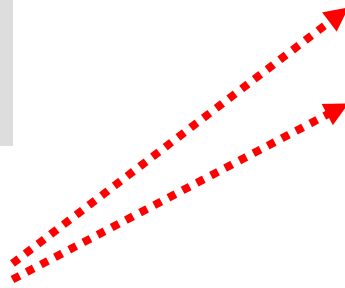
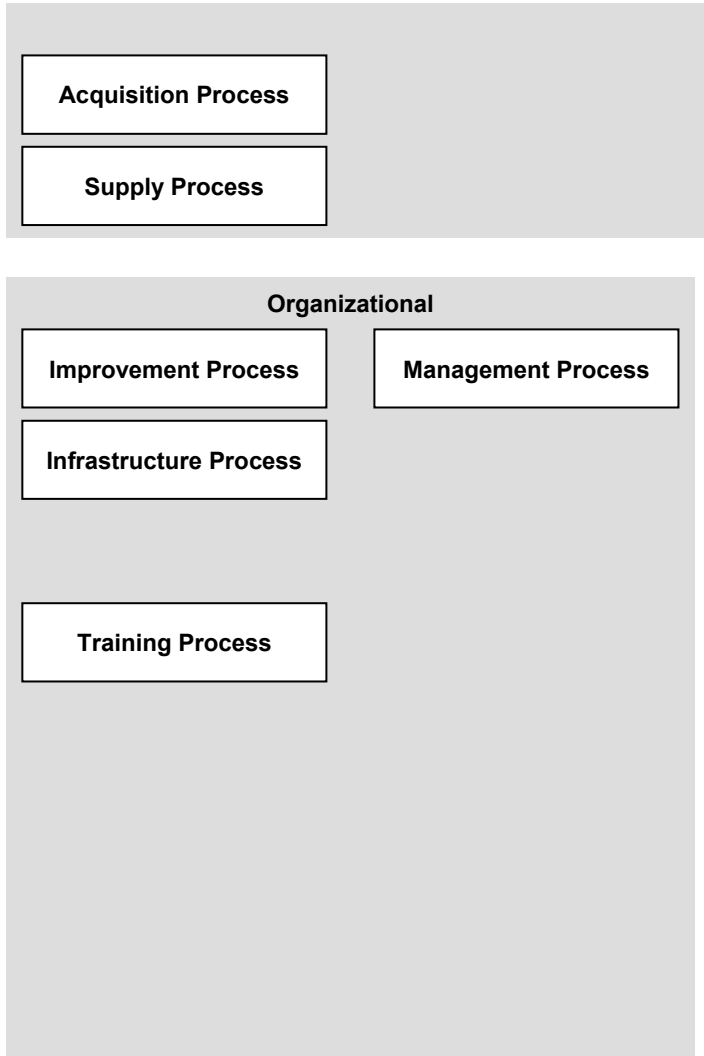
12207:2007 has a set of technical processes similar to system context activities of the development process of 12207:1995, the component processes of 12207 Amd.1, and the technical processes of 15288:2002.



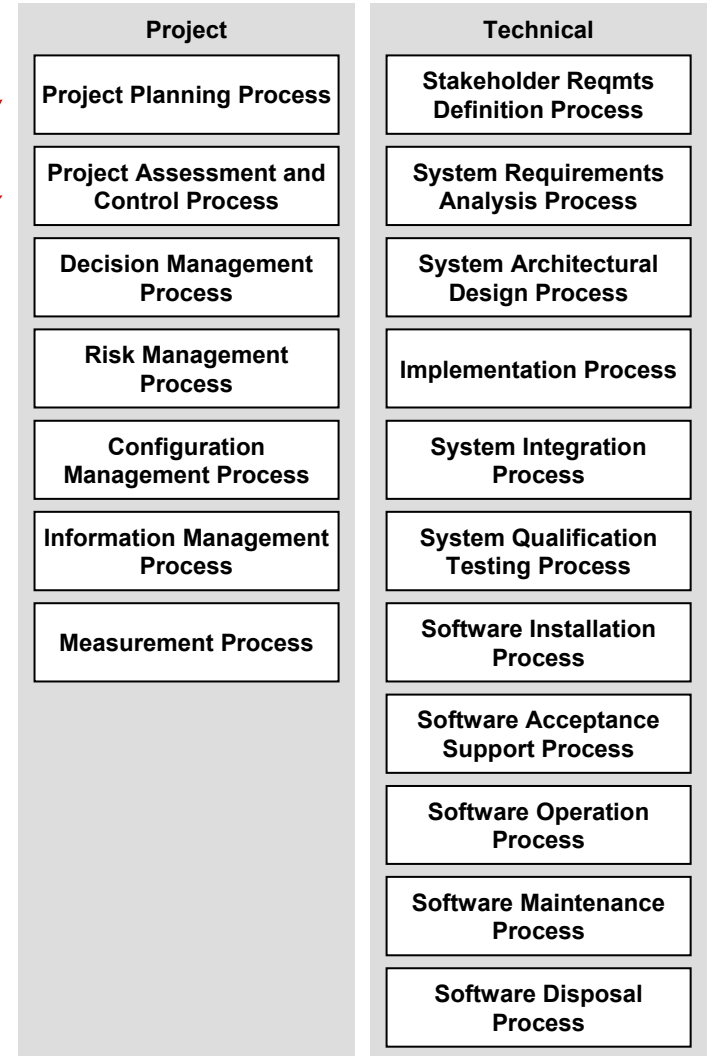


# Building 12207:2007 – Project Processes

12207:2007 has the same set of project processes as 15288:2007.

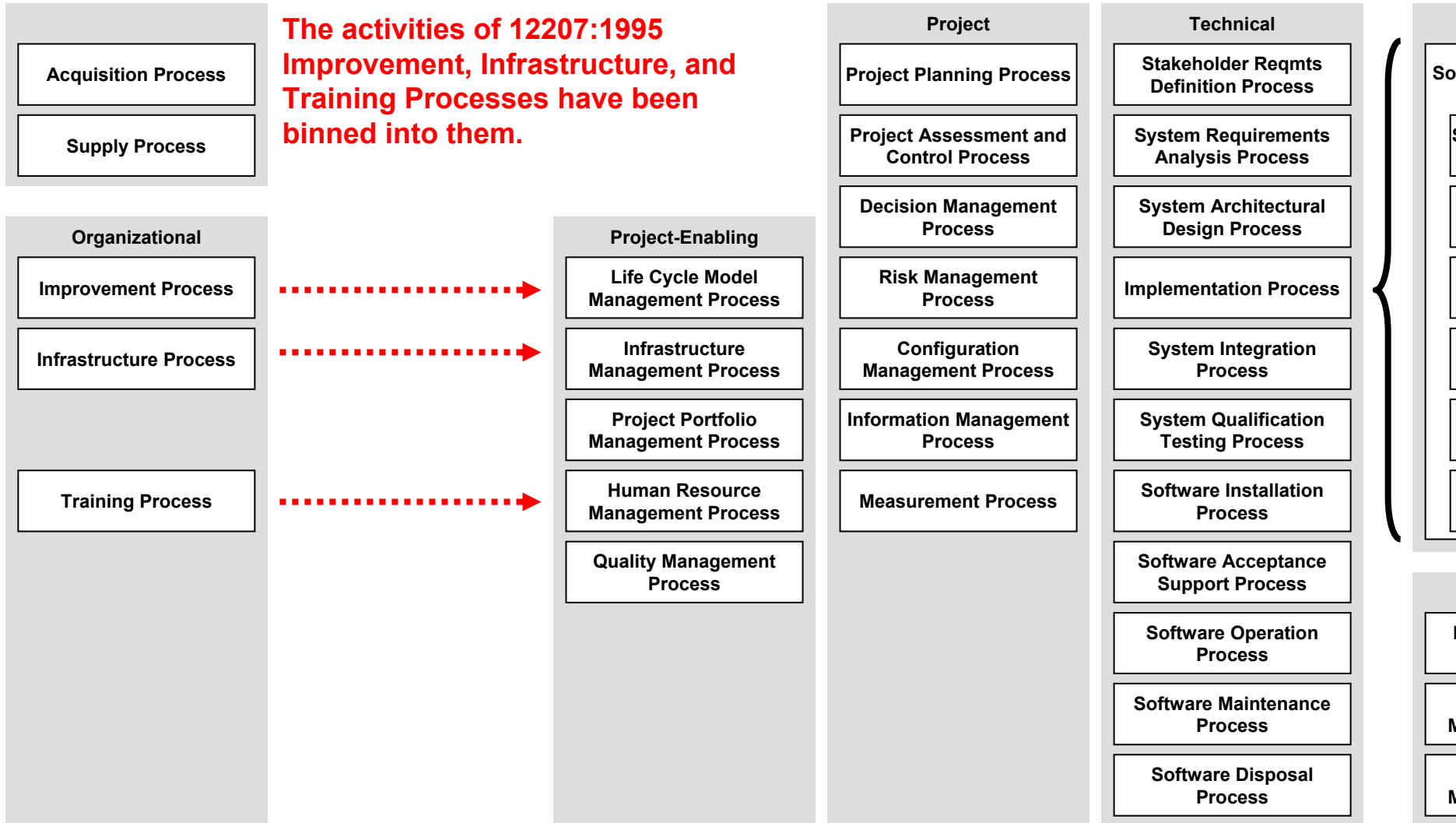


The activities of the 12207:1995 Management Process have been binned into them.



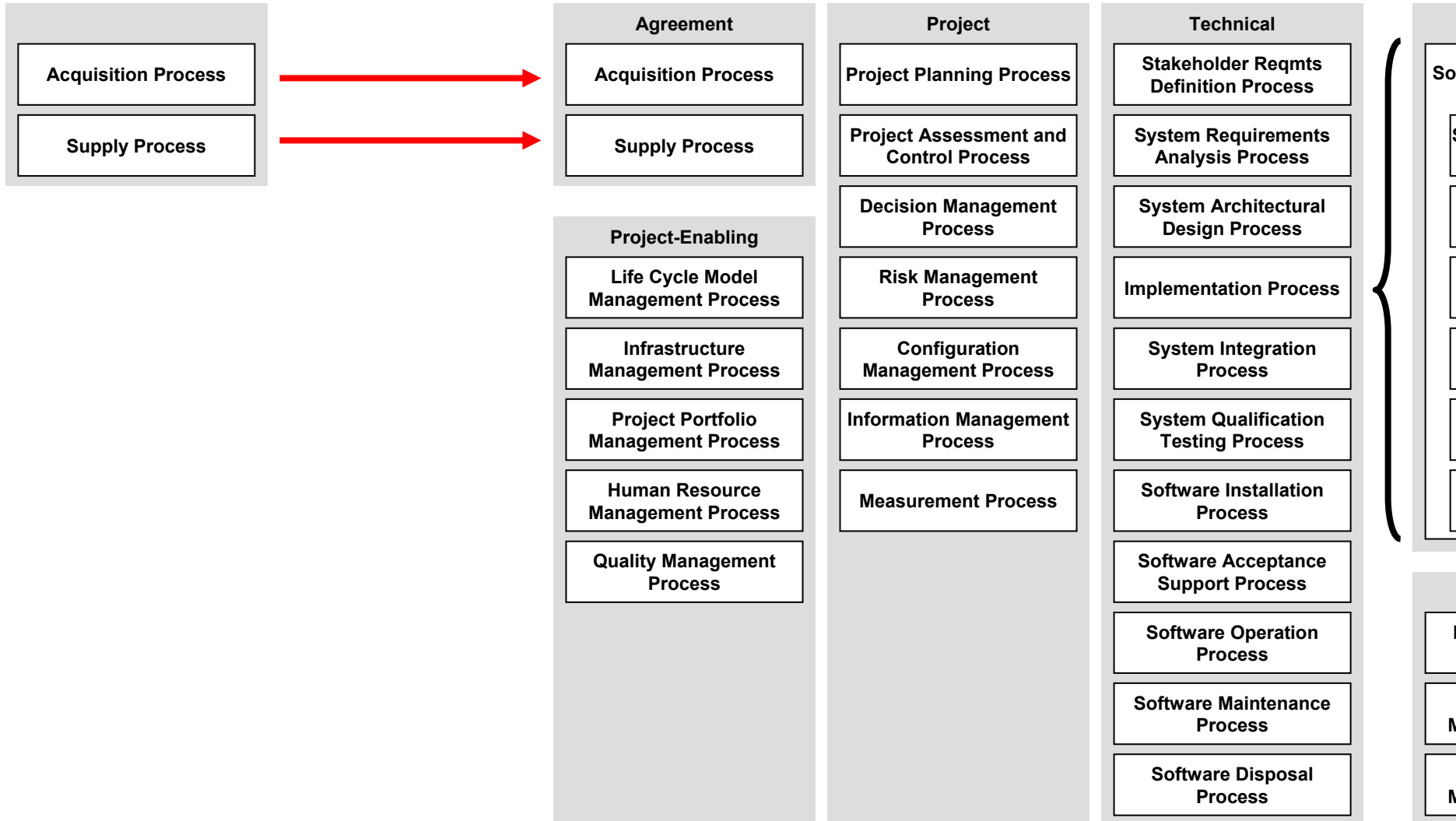
# Building 12207:2007 – Project-Enabling Processes

12207:2007 has the same set of project-enabling processes as 15288:2007.

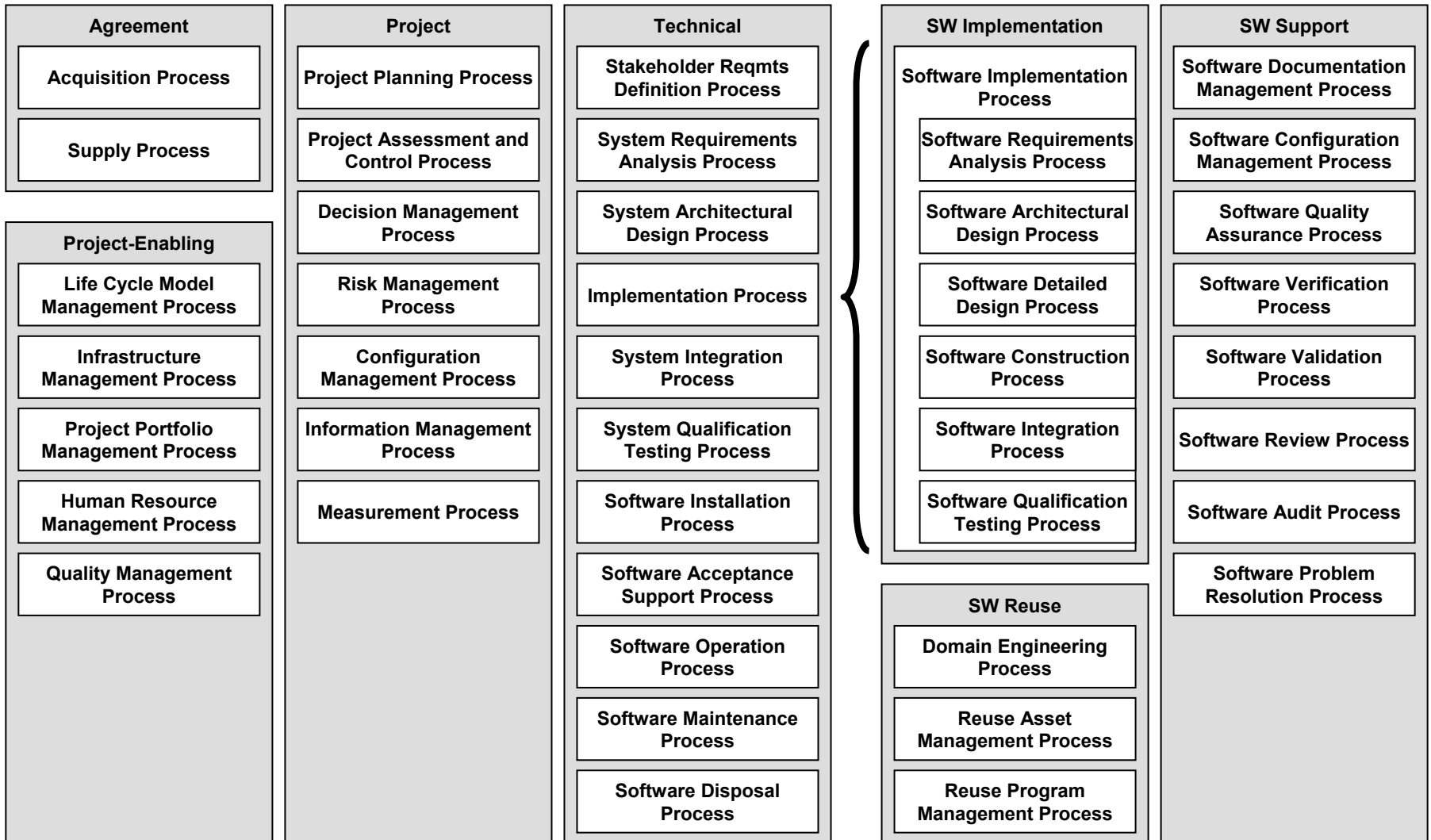


# Building 12207:2007 – Agreement Processes

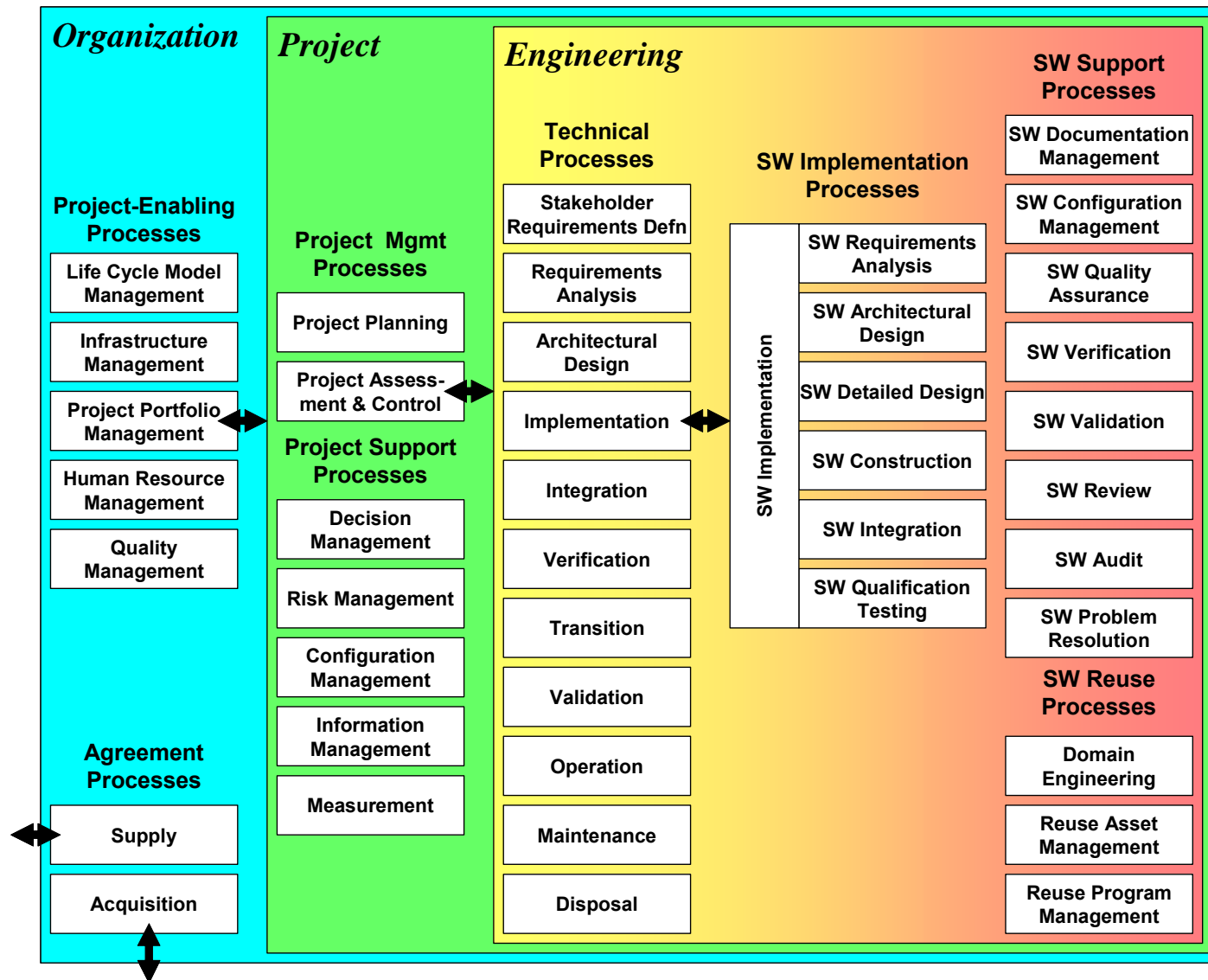
12207:2007 has the same set of agreement processes as 15288:2007.



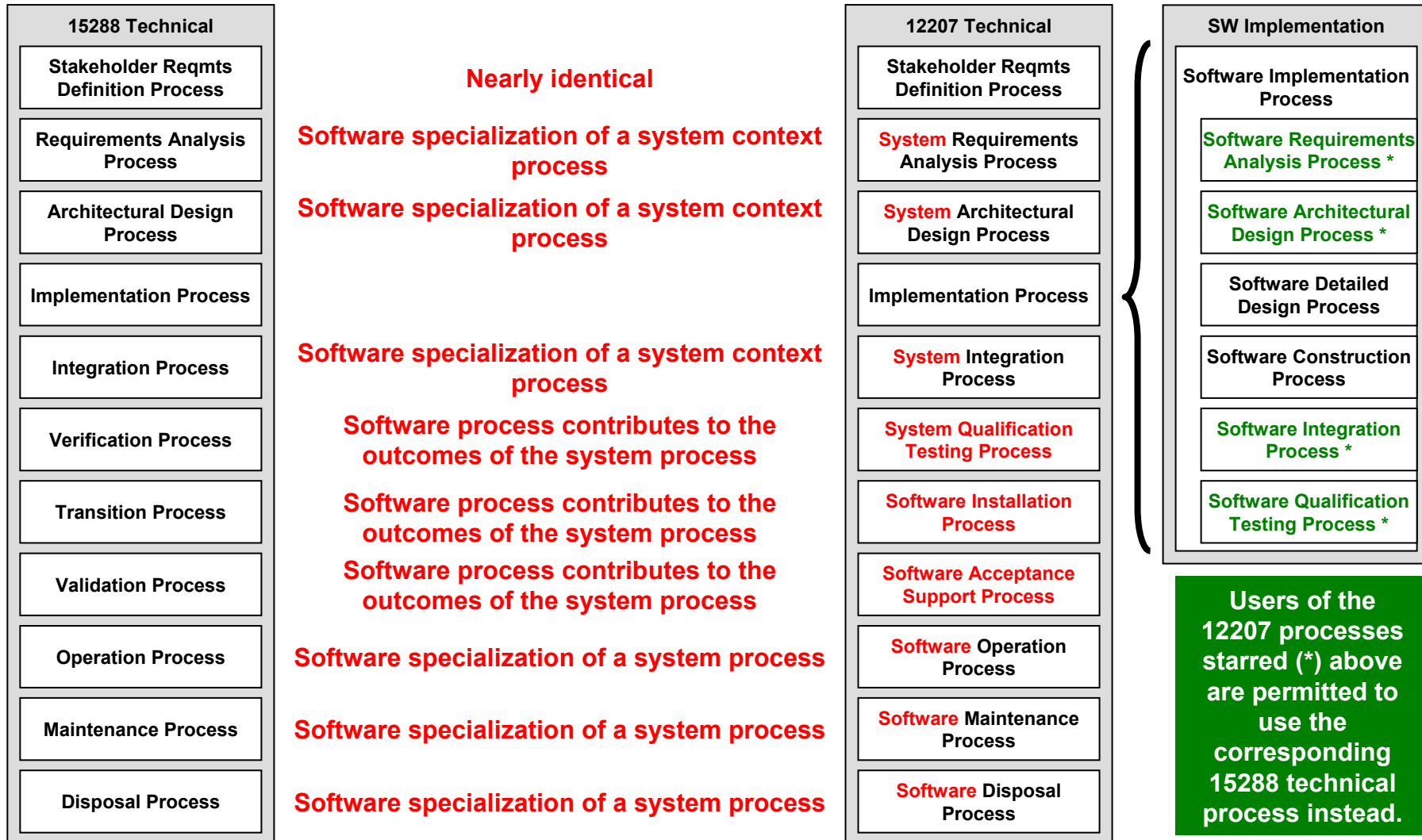
# The Life Cycle Processes of 12207:2007



# Anticipated Result of Process Alignment



# Comparison of Technical Processes of 12207:2007 and 15288:2007



Users of the 12207 processes starred (\*) above are permitted to use the corresponding 15288 technical process instead.

Users of the 12207 technical processes are permitted to use the corresponding 15288 technical process instead.

# Selecting which Standard to Use

- Both 12207 and 15288 contain process models that are nearly identical:
  - The differences are rational rather than accidental.
- 15288 describes the processes at the system level.
- 12207 specializes the same processes to software, and adds processes specific to software.

To deal with a system ...

... use 15288.

To deal with a software element of a system ...

... use 15288 and the software processes of 12207.

To deal with a software product or service  
(with minimal surrounding system) ...

... use 12207.

# The Harmonization Problem – Problems addressed in the Alignment Phase

Starting with a diverse set of standards with ...

- Different terms
- Different process sets
- Different process architectures
- Different levels of prescription
- Different audiences



... develop a revised set of standards with ...

- A single vocabulary
- A single process set
- A single, uniform architecture
- A shared level of prescription
- Suitable across the audiences



... all without needlessly disrupting current organizational investment based on usage of the current set of standards.

\* System versus software and process definition versus process assessment



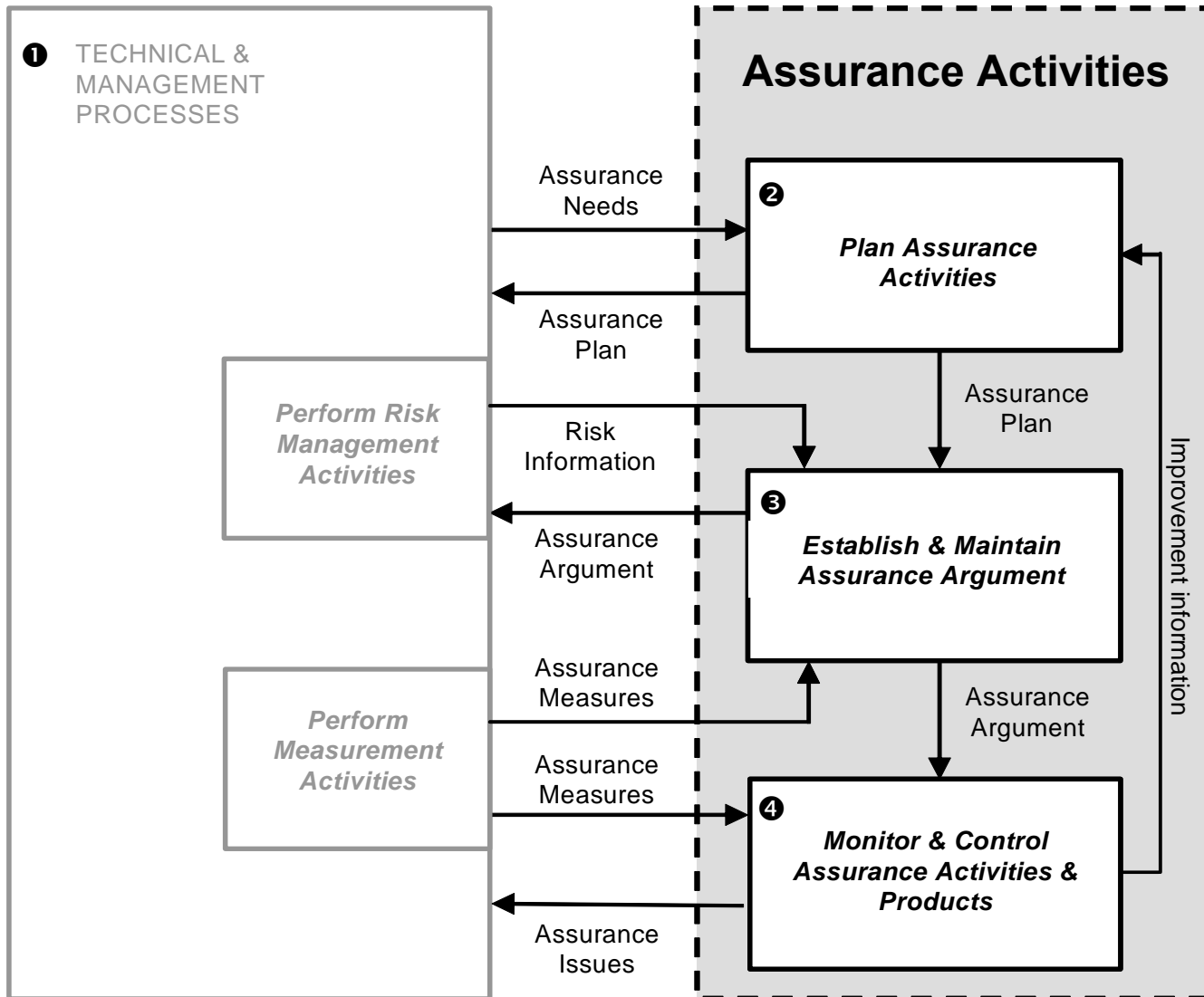
# Status and Plan

- **We expect to complete Alignment phase by end of 2007**
- **We are starting to plan for the full integration phase**
  - **A Study Group was created to determine scope and content of Integration Phase of Harmonization to achieve a fully harmonized view of the system and software life cycle processes**
  - **Integration will consider:**
    - **Common purpose and outcomes**
    - **Document architecture**
    - **Level of prescription of activities and tasks**
    - **Treatment of life cycle models and other issues**
    - **Treatment of services**
    - **Common verification and validation concepts**
    - **Common configuration management concepts**
    - **Deferred recommendations**
    - **Alignment with other applicable standards**
    - **Rationalization of application guides**

# Related Projects

- **24774** summarizes the agreed conventions for describing processes.
- **15289** summarizes the data products produced by the processes of 15288 and 12207. It exists, but must be revised to deal with the 2007 revisions.
- Three standards provide additional details on selected life cycle processes: **15939**, Measurement; **16085**, Risk management; **16326**, Project management.
- **15026** provides additional practices for the assurance of systems and software when particular critical properties are required.
- **24765** is a database of vocabulary that will occasionally be published as a conventional printed standard:
  - Database is publicly available at: <http://www.computer.org/sevocab>

# Relating Assurance to Life Cycle Processes



At the center of assurance is the concept of life cycle maintenance of an assurance case.

**Problem: How to relate the important tasks involved in assurance to the existing life cycle processes, without creating new processes.**

*Adapted from a chart by Paul Croll*

# A New Concept: The “Process View”

- Addresses a particular engineering interest
- Gathers together process activities that address those interests
- Cuts across all or part of the life cycle
  
- *Like* a process, a process view has:
  - Purpose
  - Outcomes
  
- *Unlike* a process, it does **not** have activities and tasks. Instead, it selects activities and tasks from existing processes and adds guidance on how to use them to achieve the purpose and outcomes.

# System and software assurance process view

- **Purpose:** ... develop, maintain and provide grounds for confidence and decision making, throughout its life cycle, that the system and its constituent software meets its safety, security and dependability goals or objectives, and possesses the related required safety, security and dependability properties.
- **Outcomes:** As a result of successful execution of the process view:
  - Goals and objectives for safety, security, dependability and any other designated critical properties are formulated.
  - Product assurance-related objectives, properties, or characteristics are explicitly selected for special attention and application of this standard to address the goals and objectives.
  - Requirements for the achievement of these objectives, properties, or characteristics are defined.
  - Measures for the requirements are selected and related to the desired characteristics.
  - Criteria for the achievement or degree of achievement of these objectives, properties, or characteristics are selected and traced to requirements.
  - Approaches for achieving the objectives, properties, or characteristics are planned, designed, and implemented, as well as demonstrating and documenting that achievement.
  - The extent of achievement is continuously monitored, documented, and communicated to stakeholders and managers.
  - An assurance case documenting and communicating the extent of achievement is specified, developed, and maintained as an element of the system.
  - The artefacts for documenting, analyzing, and communicating the required or claimed properties and characteristics and the extent of achievement are specified, developed, and maintained.
  - Requirements of the approval authority are satisfied and necessary licenses or certifications are received.

**Caution: This wording may change during balloting.**

# Status of Process Harmonization Projects

Project	Title	Stage	Target	IEEE plans
24748	Guide to life cycle management	Entering PDTR ballot	December 2008	Adoption
12207	SW LC processes	Entering FDIS ballot	December 2007	Joint project
15288	System LC processes	Entering FDIS ballot	December 2007	Joint project
15289	LC data	Revision planned	June 2009?	Adoption
15939	Measurement process	About to be published	Sep 2007	Adoption
16085	Risk mgmt process	Published in 2006		Published in 2006
16326	Project mgmt	Entering FCD ballot	Sep 2008	Joint project
15026	System & SW assurance	Entering CD ballot	Dec 2008	Joint project
24765	Vocabulary	Entering CD ballot	Dec 2008	Joint project
24774	Process definition guide	About to be published; will be revised.	Dec 2007	N/A

Normal stages:

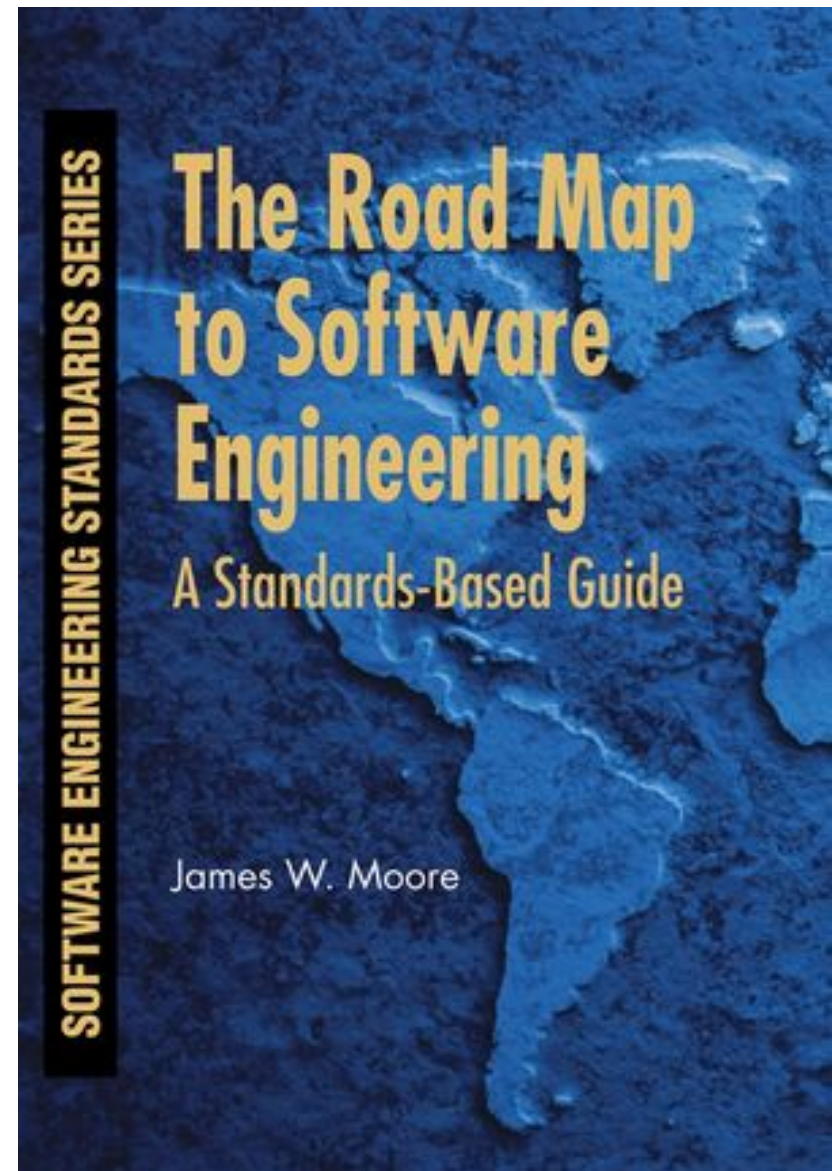


# Questions?

For more information,  
contact presenter:

**Jim Moore**

[moorej@mitre.org](mailto:moorej@mitre.org)



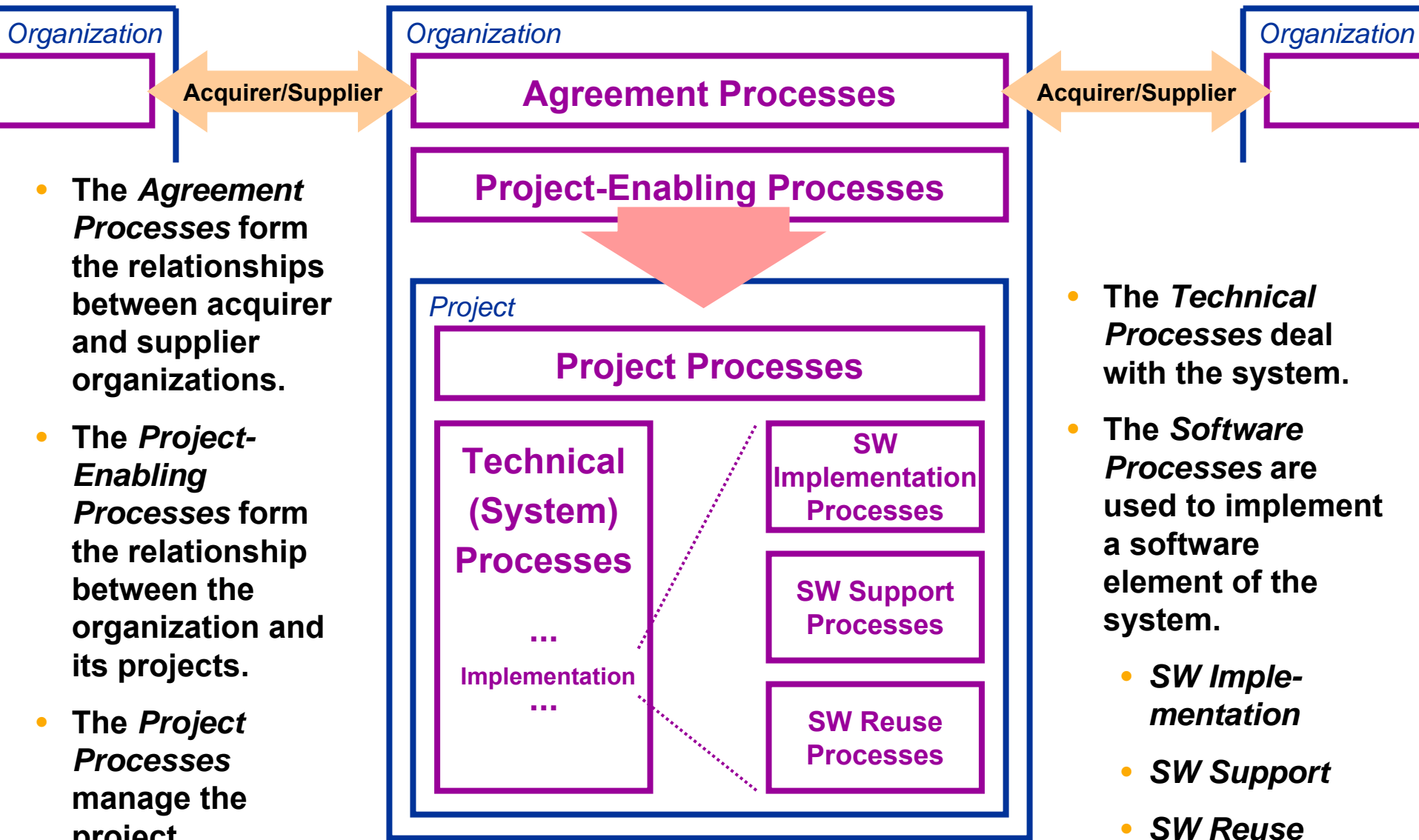
John Wiley / IEEE Computer Society Press, 2006,  
ISBN-10: 0471683620, ISBN-13: 978-0471683629





# Backup Slides

# Process Model of 15288 and 12207



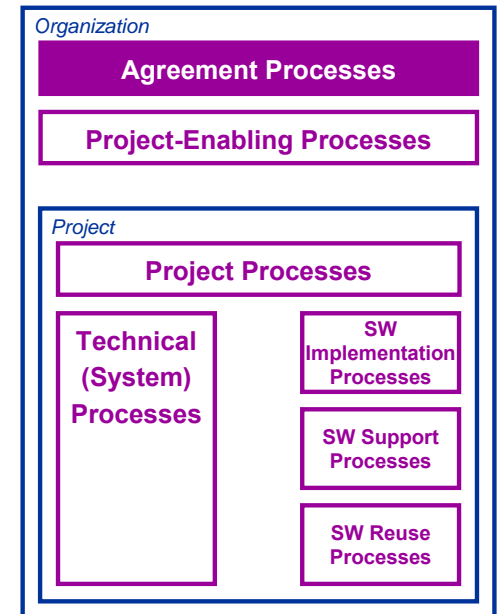
- The **Agreement Processes** form the relationships between acquirer and supplier organizations.
- The **Project-Enabling Processes** form the relationship between the organization and its projects.
- The **Project Processes** manage the project.

- The **Technical Processes** deal with the system.
- The **Software Processes** are used to implement a software element of the system.
  - **SW Implementation**
  - **SW Support**
  - **SW Reuse**

# Agreement Processes

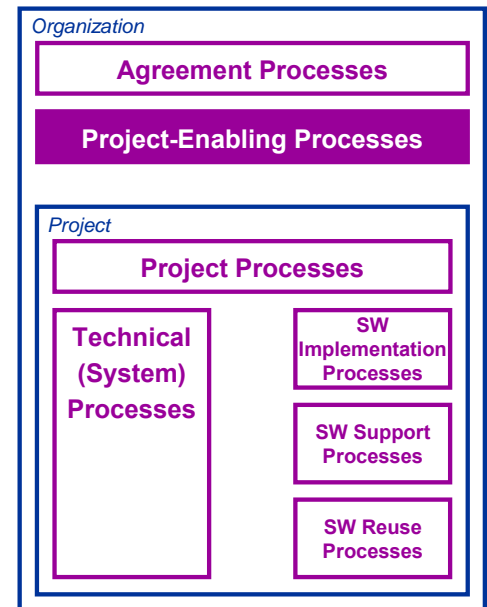
<b>Activities of the Acquisition Process</b>	<b>Activities of the Supply Process</b>
Acquisition preparation	Opportunity identification
Acquisition advertisement and supplier selection	Supplier tendering
Agreement initiation	Agreement initiation
Agreement monitoring	Agreement execution
Acquirer acceptance	Product/service delivery & support
	Closure

- **Two Agreement Processes:**
  - *Acquisition*
  - *Supply*
- The activities of the two processes “talk to” each other.
- 12207 is specialized to software and is more prescriptive.

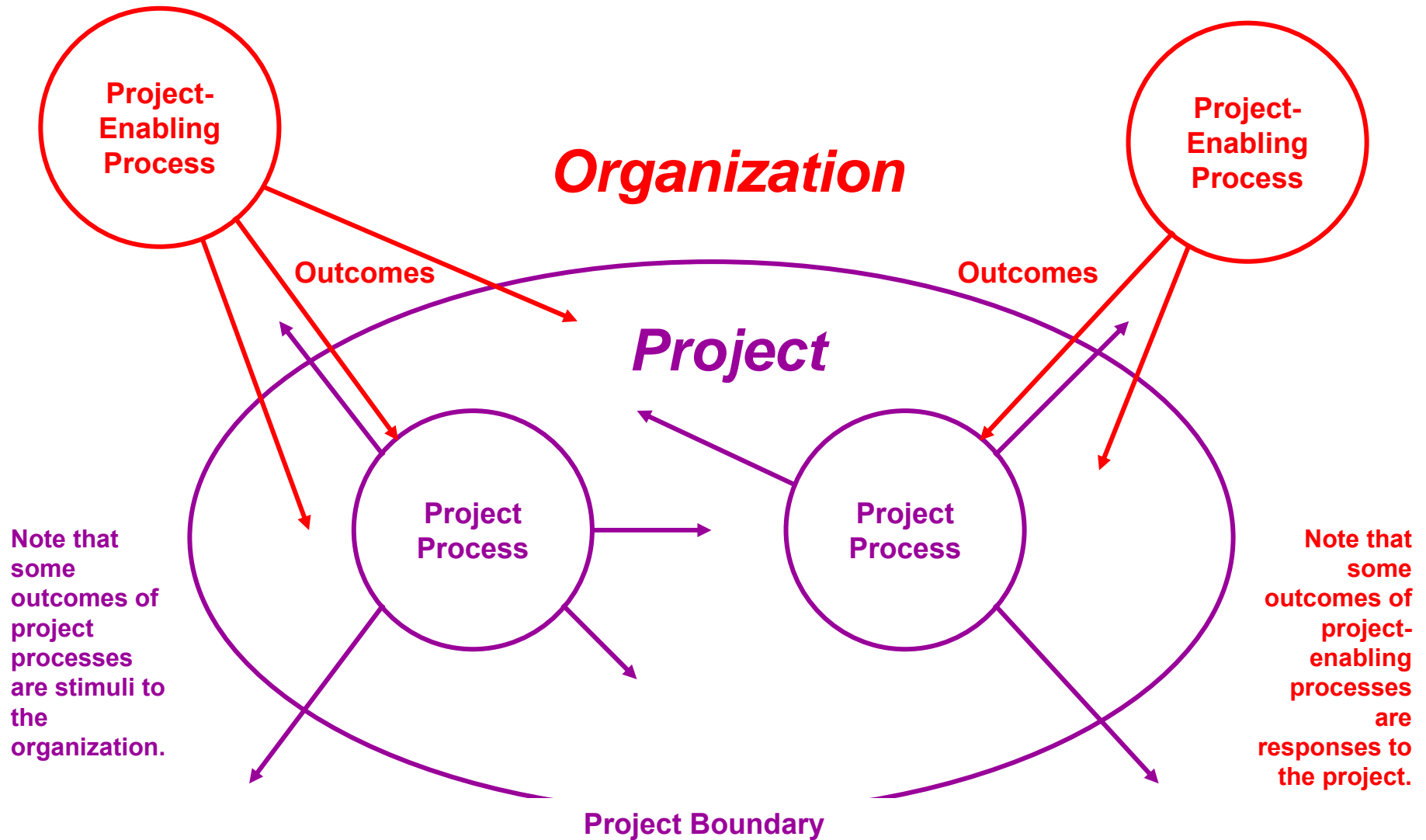


# Project-Enabling “Processes”

- There are issues of scope and competency when system and software engineers try to define organizational processes. Therefore, these are really resource dependencies rather than complete process definitions.
- The dependencies may be grouped by subject area.
- “Processes”:
  - Life Cycle Model Management
  - Infrastructure Management
  - Project Portfolio Management
  - Human Resource Management
  - Quality Management
- 12207 is more prescriptive and specializes the processes to software.



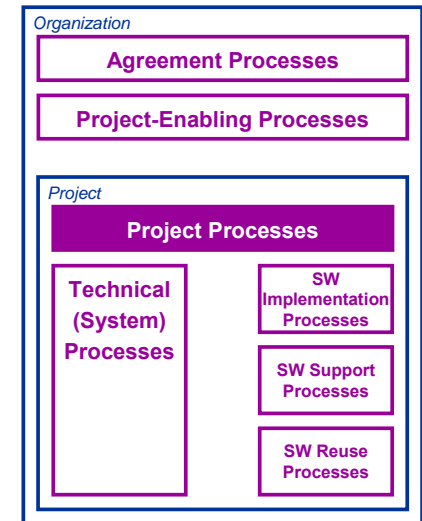
# Relationship of project-enabling processes to project



# Project Processes

## ■ Two categories:

- Project Management Processes (similar to PMBOK “process groups”):
  - *Project Planning, Project Assessment and Control*
- Project Support Processes (similar to selected PMBOK “knowledge areas”):
  - *Decision Management, Risk Management, Configuration Management, Information Management, Measurement*



- 15288 and 12207 have nearly identical processes in this area.
- 12207 is more prescriptive re software project plans.

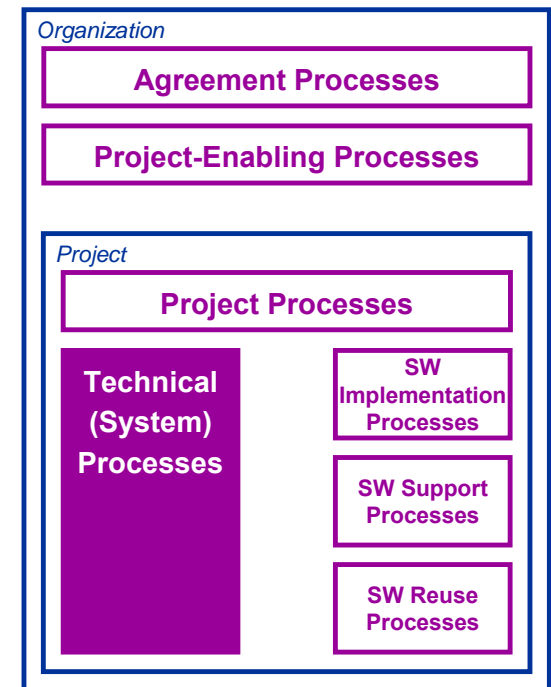
Process Groups Knowledge Area	Initiating	Planning	Executing	Controlling	Closing
4. Project Integration Management		4.1 Project Plan Development	4.2 Project Plan Execution	4.3 Integrated Change Control	
5. Project Scope Management	5.1 Initiation	5.2 Scope Planning 5.3 Scope Definition		5.4 Scope Verification 5.5 Scope Change Control	
6. Project Time Management		6.1 Activity Definition 6.2 Activity Sequencing 6.3 Activity Duration Estimating 6.4 Schedule Development		6.5 Schedule Control	
7. Project Cost Management		7.1 Resource Planning 7.2 Cost Estimating 7.3 Cost Budgeting		7.4 Cost Control	
8. Project Quality Management		8.1 Quality Planning	8.2 Quality Assurance	8.3 Quality Control	

# 15288 Technical Processes

- Stakeholder Requirements Definition
- Requirements Analysis
- Architectural Design
- Implementation
  - Example: Implementation of a Sub-system
  - Example: Implementation of a Software Element
  - Example: Implementation of a Hardware Element
- Integration
- Verification
  - Center of gravity is “after-the-fact” verification that the integrated system achieves its requirements.
- Transition
- Validation
  - Center of gravity is “after-the-fact” evaluation that the completed system meets user needs.
- Operation
- Maintenance
- Disposal

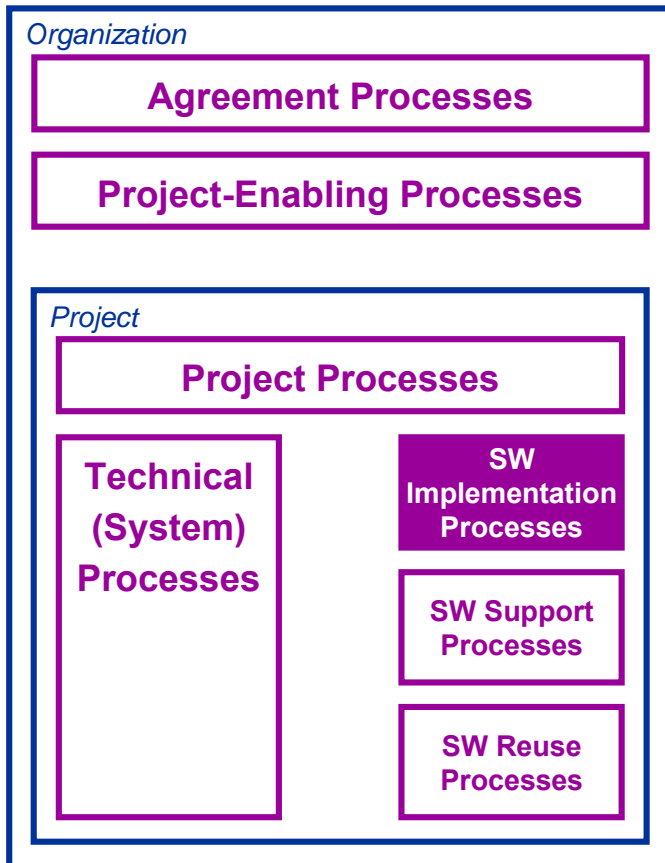
*Some see a “V Model” in this set of processes.*

**Connection to 12207**



# Software Implementation Processes

## Connection to System



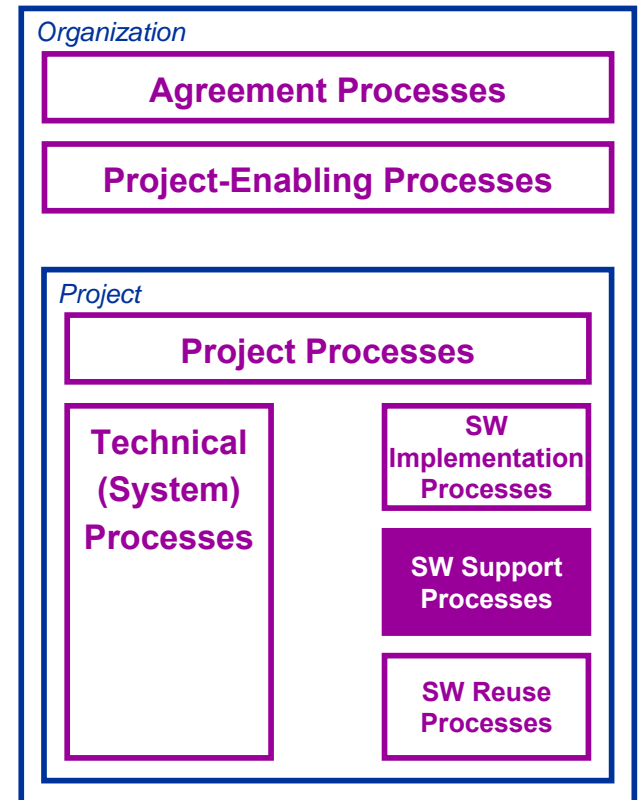
- **Software Implementation** is a specialization of System Implementation. It contains some lower-level processes distinct to software development.

- **Software Requirements Analysis**
  - (Option to use 15288 process instead.)
- **Software Architectural Design**
  - (Option to use 15288 process instead.)
- **Software Detailed Design**
- **Software Construction**
- **Software Integration**
- **Software Qualification Testing**
  - (Option to use 15288 process instead.)



# Software Supporting Processes

- **Software Documentation**
  - Focused on SW life cycle artifacts
- **Software Configuration Management Process**
  - Specialized to SW
- **Software Quality Assurance Process**
  - Specialized to SW
- **Software Verification Process**
  - Center of gravity is continuous verification
- **Software Validation Process**
  - Center of gravity is continuous validation
- **Software Review Process**
- **Software Audit Process**
- **Software Problem Resolution Process**



# Software Reuse Processes

- **Software Reuse Processes**
  - Domain Engineering
  - Reuse Asset Management
  - Reuse Program Management
- **These are grouped separately because**
  - They inherently cut across project boundaries.
  - But, they don't seem to be “enterprise” level processes.
  - They are sort of a “loose end”.
- **Ultimately, these may be treated as a special kind of project.**

