

ISO/IEC/IEEE 15288 and ISO/IEC/IEEE 12207: The *Entry-Level* Process Standards

Jim Moore

moorej@mitre.org

Rev 4, April 2010

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.

Purpose of Briefing

Claim: ISO/IEC/IEEE 15288, *System Life Cycle Processes*, and ISO/IEC/IEEE 12207, *Software Life Cycle Processes*, are the *entry-point* for life-cycle process implementation. Beginners should *start* with them.



Whoa! *That can't be true.* They are too big, too clumsy, too rigid, and too heavy-weight.

This presentation will explain why the claim is true!

Some Background

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.

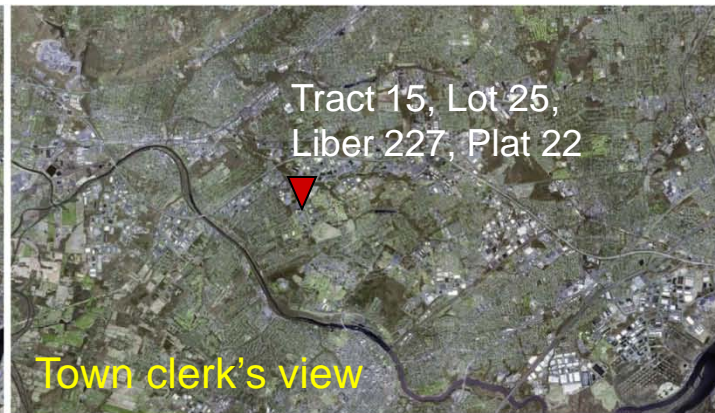


It defines some characteristics that a buyer can count on.

- 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.

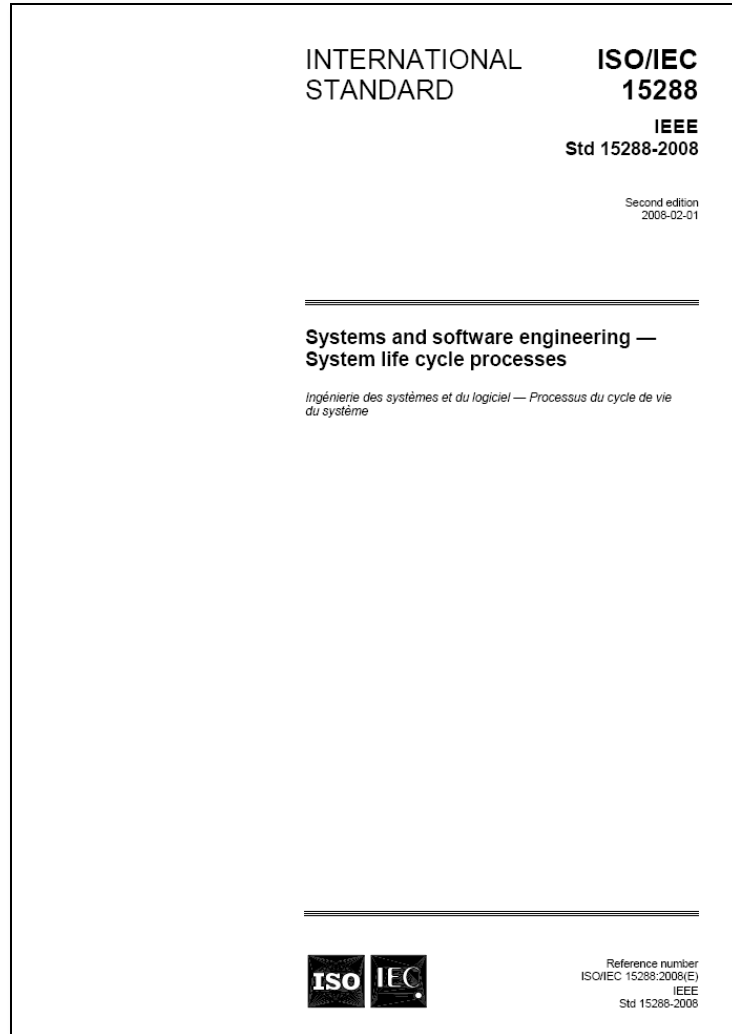


- Would you know that these are different names for the same thing?
- Would you know without the map?
- 12207 and 15288 are “maps” for the life cycle process space.

15288 and 12207 Give Names to Processes

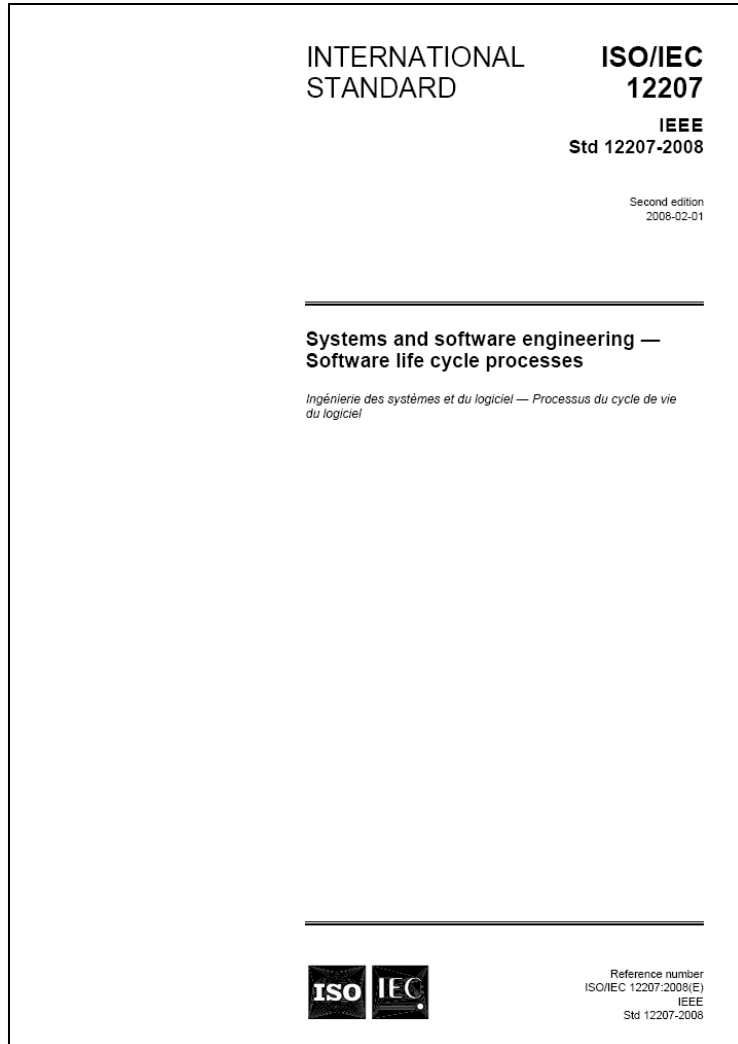
- ISO/IEC 15288:2008 gives names to processes in the life cycle of a system.
- ISO/IEC 12207:2008 gives names to processes in the life cycle of a software product or service.
- The two standards are designed to be used together for software-intensive systems.
- The names are important so that acquirers and suppliers can communicate regarding their practices.
 - “Oh, when you say ‘implementation’, you include ‘testing’? Oh, no, no, no-- in our corporate process, testing is a separate thing; so our contract doesn’t include that! You have to pay us more if you want testing.”
- 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.**

ISO/IEC/IEEE 15288, System Life Cycle Processes



- Provides 25 processes covering the life-cycle of any human-made system
- 84 pages
- First written by ISO/IEC JTC 1/SC 7 in 2002
- Adopted by IEEE in 2003
- Jointly revised in 2008

ISO/IEC/IEEE 12207, Software Life Cycle Processes



- Provides 43 processes covering the life-cycle of any software product or system element
- 138 pages
- First written by ISO/IEC JTC 1/SC 7 in 1995
- Adopted by IEEE in 1996
- Jointly revised in 2008

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.

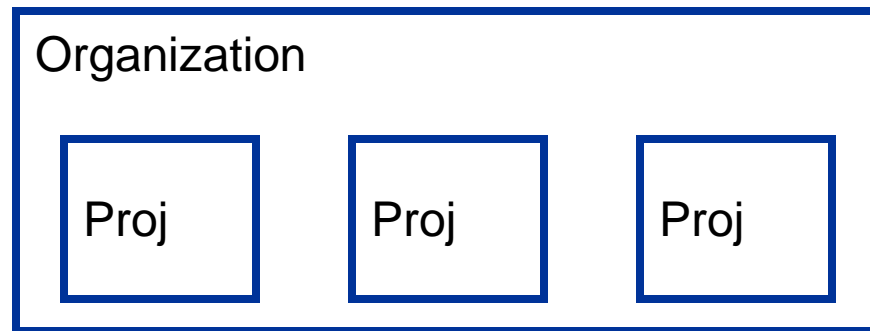
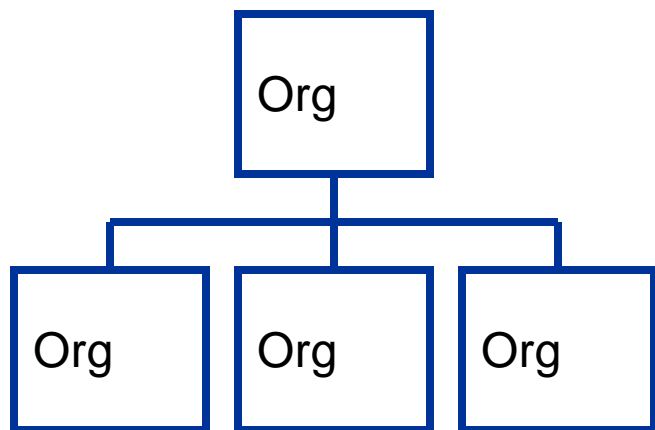
To deal with a software product or service
(with minimal surrounding system) ...

... use 12207.

Key Terminology and Concept

- **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



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

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

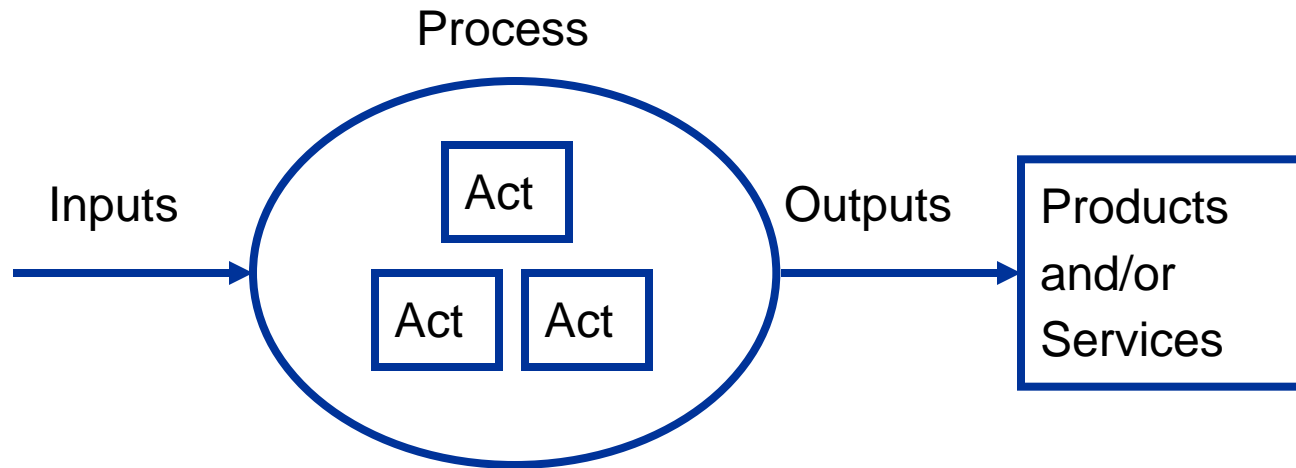


Agreeing *organizations* are called *parties*.

An *individual* can be an *organization*

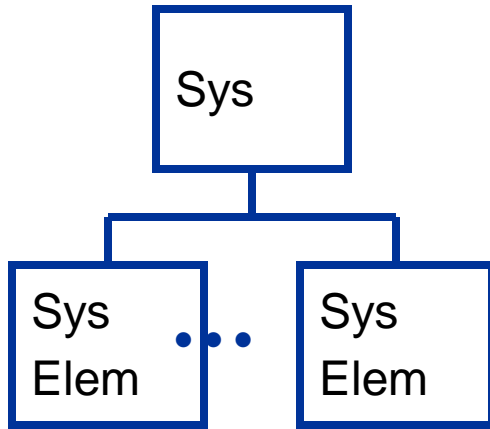


Key Terminology and Concepts



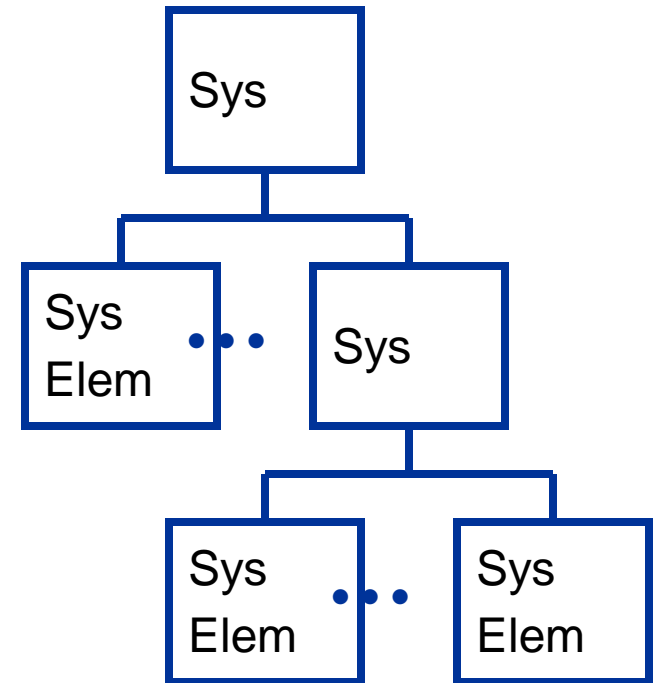
- **Process**: set of interrelated or interacting *activities* which transforms inputs into outputs [ISO 9000]
- **Product**: the result of a process [ISO 9000]
- **Service**: performance of activities, work, or duties associated with a product

Key Terminology and Concepts

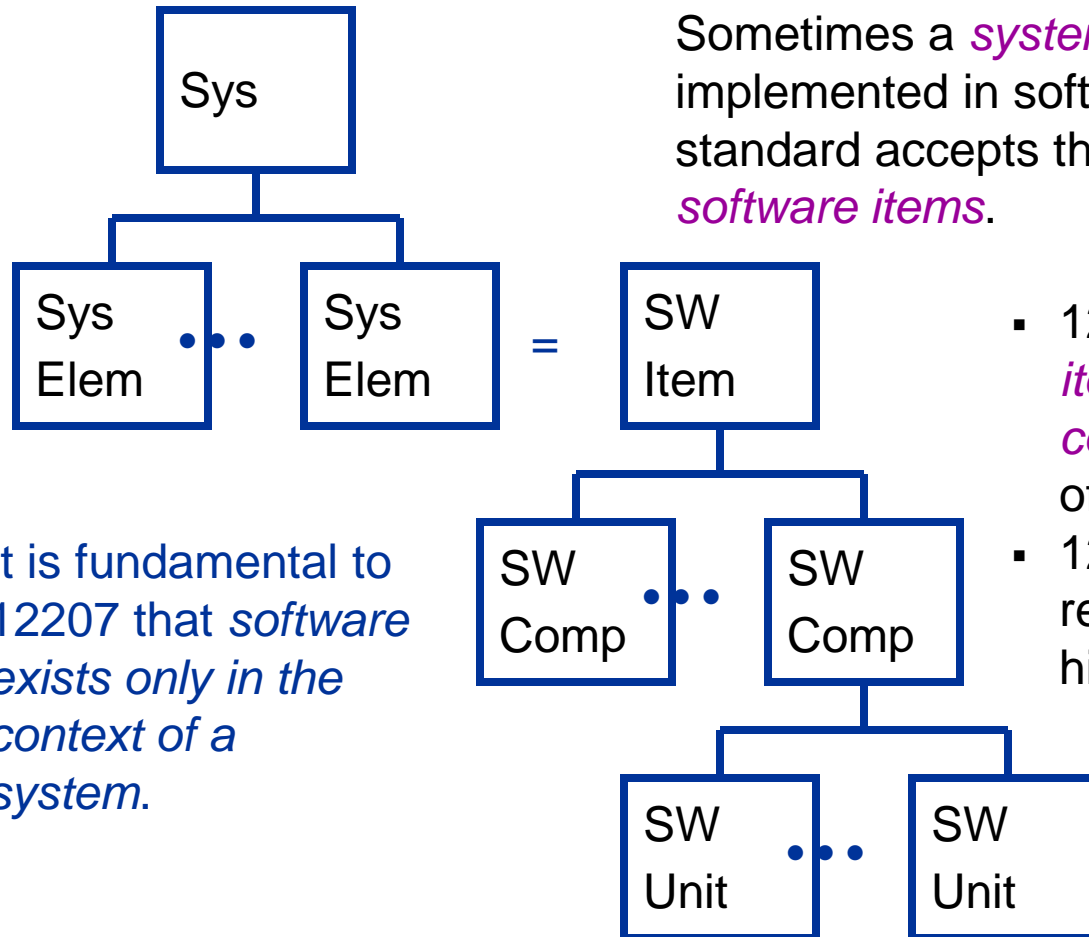


A *system* is composed of *system elements*. Each element is *implemented* and then *integrated* into the system. One invocation of 15288 suffices to create a system composed of a set of elements.

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*.

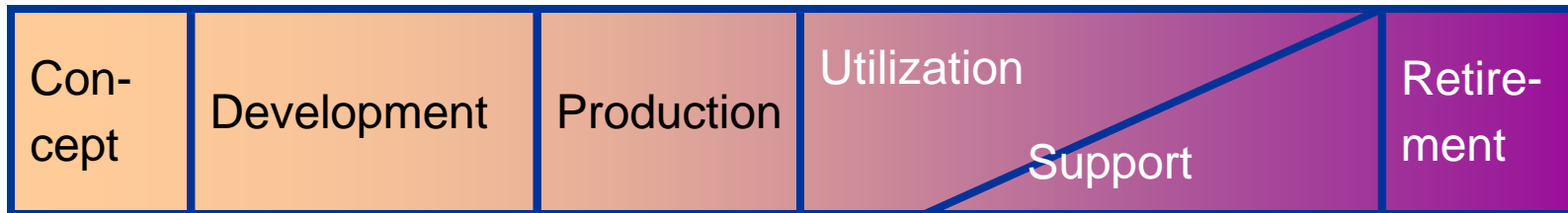
- 12207 uses a hierarchy of *items* – composed of *components* – composed of *units*.
- 12207 is *not* invoked recursively to create this hierarchy.

Key Terminology and Concepts

- 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 separated 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.



Here's what I want in an entry-level standard

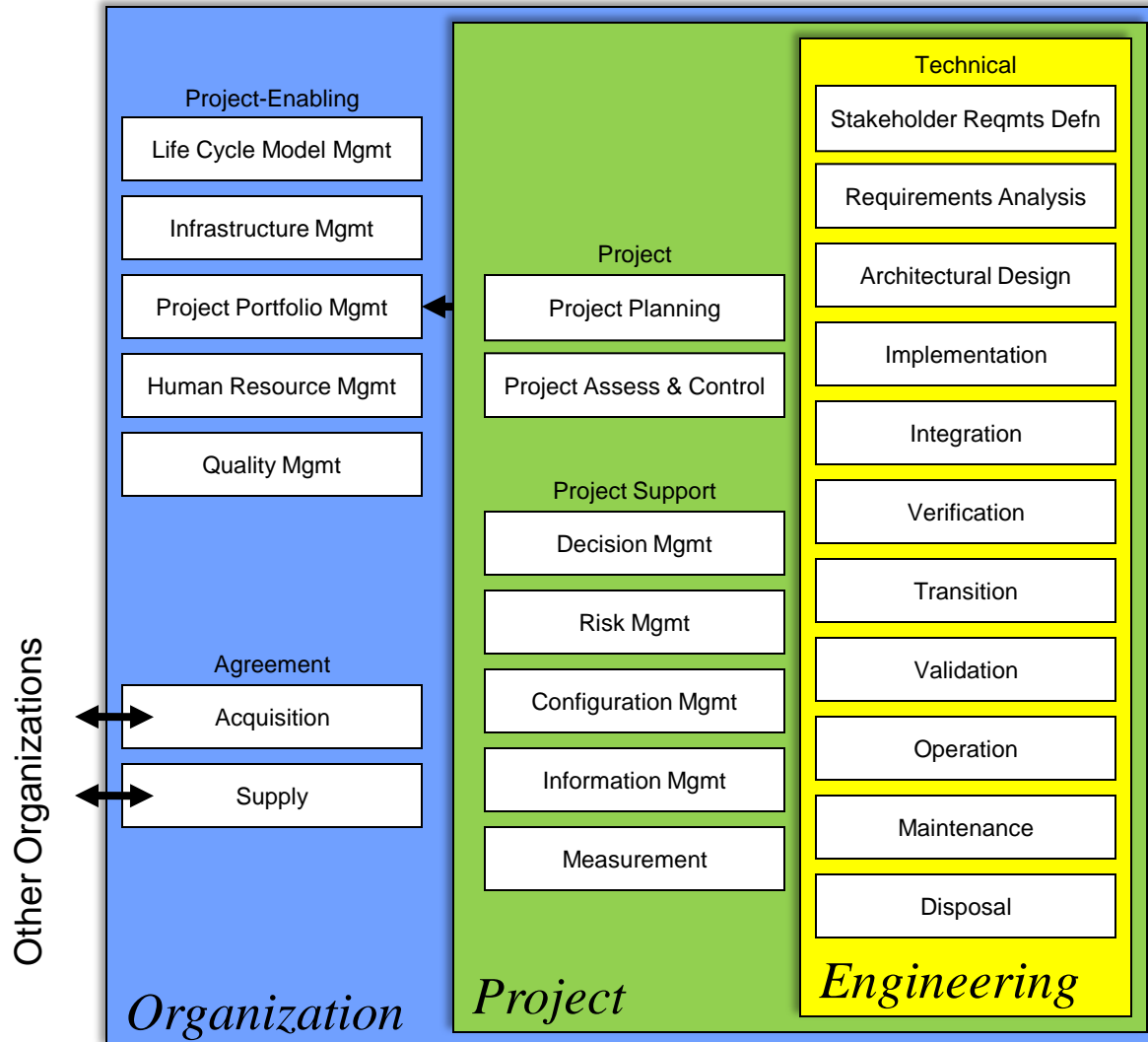


So, the standard should ...

Start small I should be able to select a few high-priority processes that are important to me.	... provide <i>a broad set of coherent, cohesive processes</i> that satisfy a variety of needs.
Start simple I should be able to select a level of detail that is appropriate to my situation.	... provide <i>varying levels of detail</i> and information where to get more.
Give me credibility I should be able to make a succinct, unhedged claim that explains what I do—a claim that my customers can understand and respect.	... provide <i>conformance criteria that customers can easily understand</i> and that separate responsible claimants from irresponsible ones.
Allow me to grow When I decide to add more processes, add more detail, or grow in capability, I should not have to throw away my existing processes.	... support <i>adding processes, adding detail, and improving capability without causing incompatibility.</i>
Adapt to my needs I should be able to implement processes that I recognize.	... provide processes that are <i>widely applicable, yet capable of adaptation.</i>

Broad Set of Coherent, Cohesive Processes

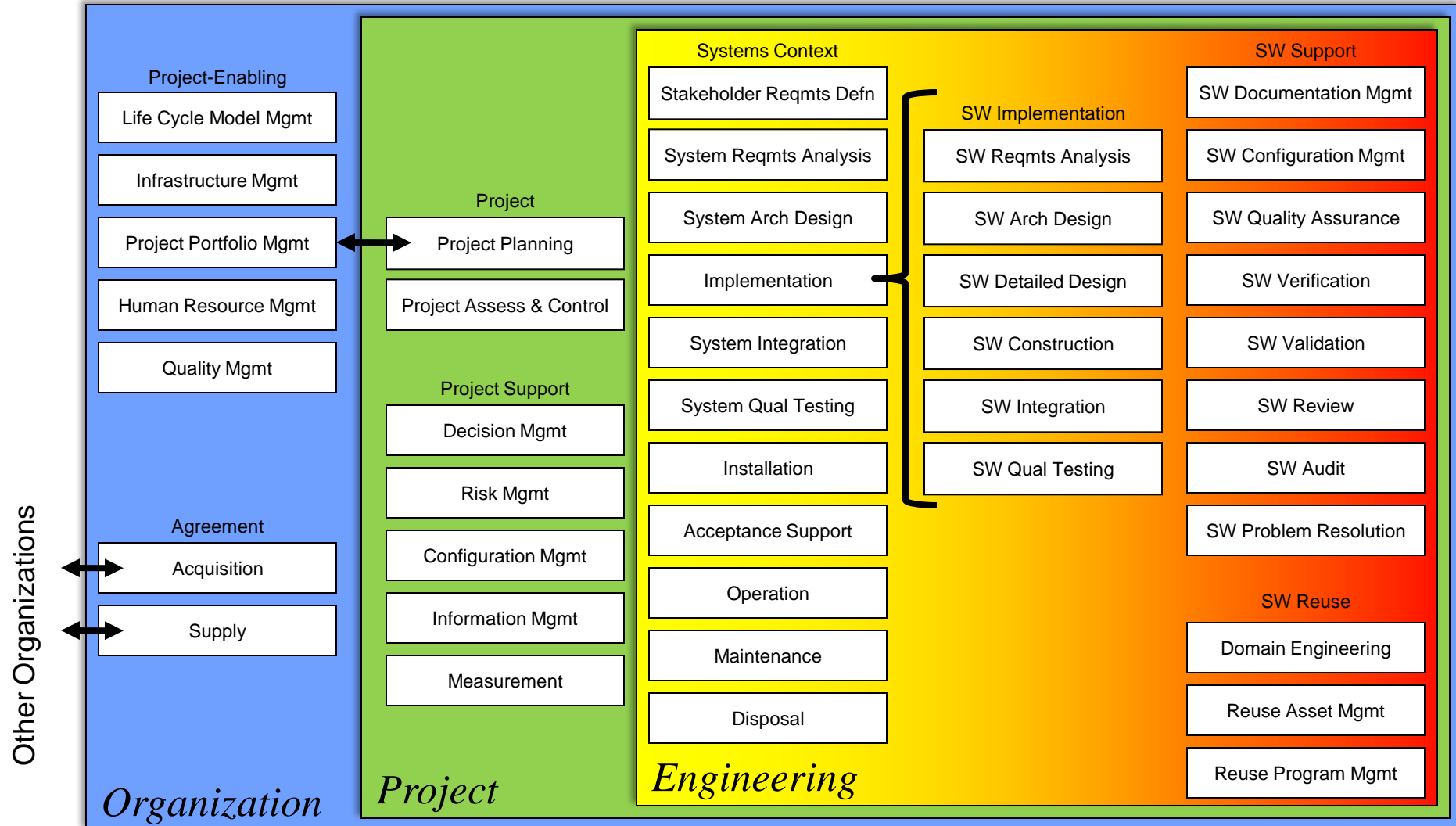
System Life Cycle Processes of 15288



- Provides all of the **Technical** processes for the entire life cycle of the system.
- Provides all of the **Project Management** processes for any stage in a system's life.
- Provides *just enough* organizational context to enable projects.

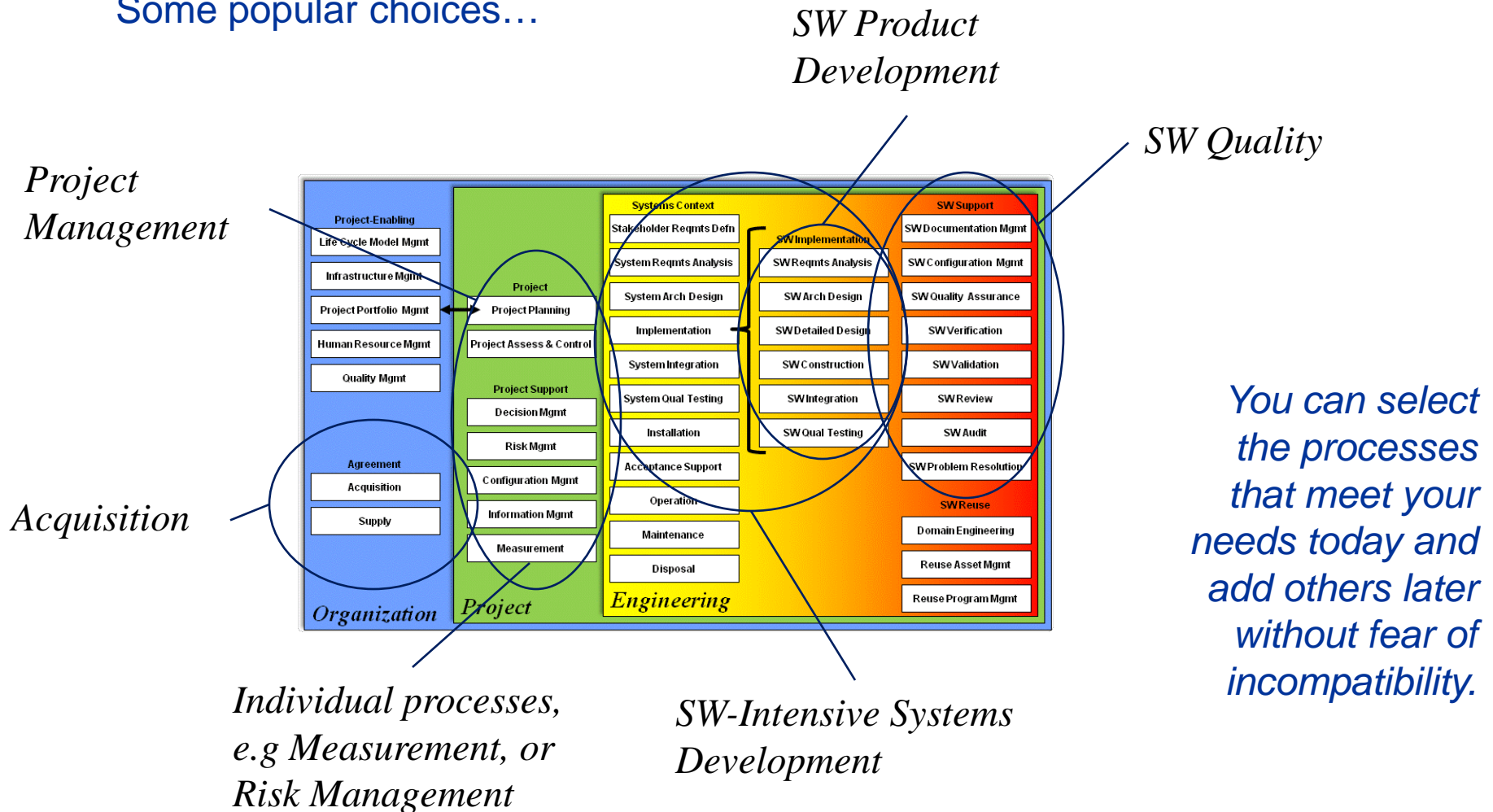
SW Life Cycle Processes of 12207

12207 specializes the processes of 15288 for software and adds software-unique processes.



Select the Part(s) of the Standard that Makes Sense for Your Needs

Some popular choices...



Varying Levels of Detail

Each Process is Described at Two Levels of Detail

1	Title	The title summarizes the scope of the process, and its principal concerns.	Software Configuration Management Process
	Purpose	The purpose is the high level, overall goal for performing the process.	The purpose of the Software Configuration Management Process is to establish and maintain the integrity of the software items of a process or project and make them available to concerned parties.
	List of Outcomes	An outcome is an observable result of the successful achievement of the process purpose.	a) a software configuration management strategy is developed; b) items generated by the process or project are identified, defined and baselined; c) modifications and releases of the items are controlled; d) modifications and releases are made available to affected parties; e) the status of the items and modifications are recorded and reported; f) the completeness and consistency of the items is ensured; and g) the storage, handling and delivery of the items are controlled.
2	Set of Activities and Tasks	Tasks define specific requirements and recommendations on the execution of the process. Activities group together related tasks.	[See next page]

Plus a third level of detail provided by supplementary standards

Activities and Tasks of Software Configuration Management Process

- **7.2.2.3.1 Process implementation.** This activity consists of the following task:
 - 7.2.2.3.1.1 A software configuration management plan shall be developed. The plan shall describe: the configuration management activities; procedures and schedule for performing these activities; the organization(s) responsible for performing these activities; and their relationship with other organizations, such as software development or maintenance. The plan shall be documented and implemented.
 - NOTE The plan may be a part of the system configuration management plan.
- **7.2.2.3.2 Configuration identification.** This activity consists of the following task:
 - 7.2.2.3.2.1 A scheme shall be established for identification of software items and their versions to be controlled for the project. For each software item and its versions, the following shall be identified: the documentation that establishes the baseline; the version references; and other identification details.
- **7.2.2.3.3 Configuration control.** This activity consists of the following task:
 - 7.2.2.3.3.1 The following shall be performed: identification and recording of change requests; analysis and evaluation of the changes; approval or disapproval of the request; and implementation, verification, and release of the modified software item. An audit trail shall exist, whereby each modification, the reason for the modification, and authorization of the modification can be traced. Control and audit of all accesses to the controlled software items that handle safety or security critical functions shall be performed.
 - NOTE The Software Problem Resolution Management Process could provide support for this activity.
- **7.2.2.3.4 Configuration status accounting.** This activity consists of the following task:
 - 7.2.2.3.4.1 Management records and status reports that show the status and history of controlled software items, including baselines shall be prepared. Status reports should include the number of changes for a project, latest software item versions, release identifiers, the number of releases, and comparisons of releases.
- **7.2.2.3.5 Configuration evaluation.** This activity consists of the following task:
 - 7.2.2.3.5.1 The following shall be determined and ensured: the functional completeness of the software items against their requirements and the physical completeness of the software items (whether their design and code reflect an up-to-date technical description).
- **7.2.2.3.6 Release management and delivery.** This activity consists of the following task:
 - 7.2.2.3.6.1 The release and delivery of software products and documentation shall be formally controlled. Master copies of code and documentation shall be maintained for the life of the software product. The code and documentation that contain safety or security critical functions shall be handled, stored, packaged, and delivered in accordance with the policies of the organizations involved.

There are standards providing even more detail for selected processes

- These standards are plug-compatible and provide additional detail:
 - ISO/IEC/IEEE 14764, Software maintenance
 - ISO/IEC/IEEE 15026, Software assurance
 - ISO/IEC/IEEE 15289, Information items (documentation)
 - ISO/IEC/IEEE 15939, Measurement
 - ISO/IEC/IEEE 16085, Risk management
 - ISO/IEC/IEEE 16326, Project management
- There are many other process standards that are generally supportive although not yet completely plug-compatible.
 - Annex F of 15288 describes the relationship of each process to 5 IEEE standards.
 - Annex G of 12207 describes the relationship of each process to 30 IEEE standards.
 - For example, IEEE Std 828, SW Configuration Management
- (All of the mentioned standards are either published or very close.)

Understandable, Credible Conformance Claims

Conformance Clause

■ 2 Conformance

— 2.2 Full conformance

- ... Full conformance is achieved by demonstrating that *all of the requirements* of the *declared set of processes* have been satisfied *using the outcomes as evidence*.

— 2.3 Tailored conformance

- When this International Standard is used as a basis for establishing a set of processes that do not qualify for full conformance, *the clauses of this International Standard are selected or modified* The tailored text ... is declared. Tailored conformance is achieved by demonstrating that requirements for the processes, as tailored, have been satisfied using the outcomes as evidence.

- You can claim conformance on a process-by-process basis.
- You can be flexible in implementing the requirements if you achieve the outcomes.

Why is Tailored Conformance a Bad Thing?

- **Tailoring allows the deletion of any provision deemed as not suitable to the job at hand.**
 - Irresponsible providers delete anything that is costly or inconvenient.
 - Responsible providers make small deletions.
- **But, both get to make the same claim – “Tailored Conformance.”**
 - A reviewer has to comb through hundreds of pages to find what has been deleted.
- **It is clearer to implement and claim Full Compliance to a core set of processes.**

Ability to Grow

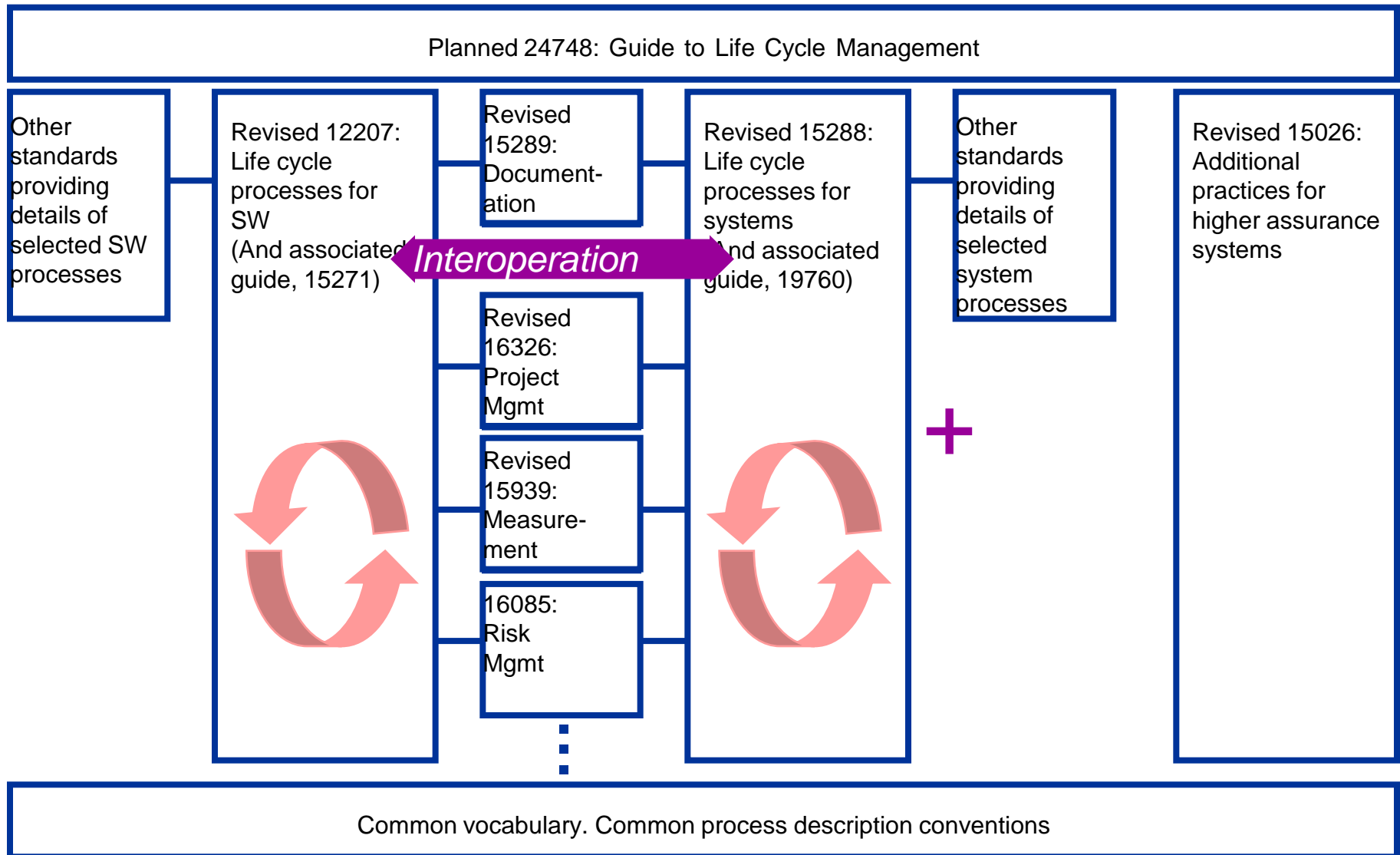
Ability to Grow

- **Adding processes**
 - As time goes on, you can select additional processes from the standards for implementation.
- **Increasing level of detail**
 - Starting at the level of outcomes, you can selectively implement the more detailed activities and tasks and the supplementary standards if appropriate.
- **Integrating software and systems engineering**
 - Most of the processes in 12207 include permission to use a more general systems engineering process from 15288.
- **Improving capability level**
 - The processes, as described in 12207 and 15288, do not require capability above level 1. ISO/IEC 15504, Process Assessment, provides the mechanism for assessing increased capability.
- **Compatibility**
 - Plug-compatible systems and software life cycle processes
 - Plug-compatible with a growing set of standards providing more detail
 - Generally compatible with the large collections of IEEE and ISO/IEC standards
 - Though reorganized, the 2008 versions of both 12207 and 15288 are backward-compatible to their previous versions.

A number of additional standards are harmonized with 12207/15228

- These standards provide a uniform context:
 - ISO/IEC/IEEE 24748-1, Guide for life cycle management
 - ISO/IEC/IEEE 24765, Vocabulary
 - Freely available at <http://www.computer.org/sevocab>
- These standards are plug-compatible and provide additional detail for selected processes:
 - ISO/IEC/IEEE 14764, Software maintenance
 - ISO/IEC/IEEE 15026, Software assurance
 - ISO/IEC/IEEE 15289, Information items (documentation)
 - ISO/IEC/IEEE 15939, Measurement
 - ISO/IEC/IEEE 16085, Risk management
 - ISO/IEC/IEEE 16326, Project management
- There are many other process standards that are generally supportive although not yet completely plug-compatible.
 - Annex F of 15288 describes the relationship of each process to 5 IEEE standards.
 - Annex G of 12207 describes the relationship of each process to 30 IEEE standards.
 - For example, IEEE Std 828, SW Configuration Management
- (All of the mentioned standards are either published or very close.)

Relationship of Key LC Process Standards



**Widely applicable and
adaptable**



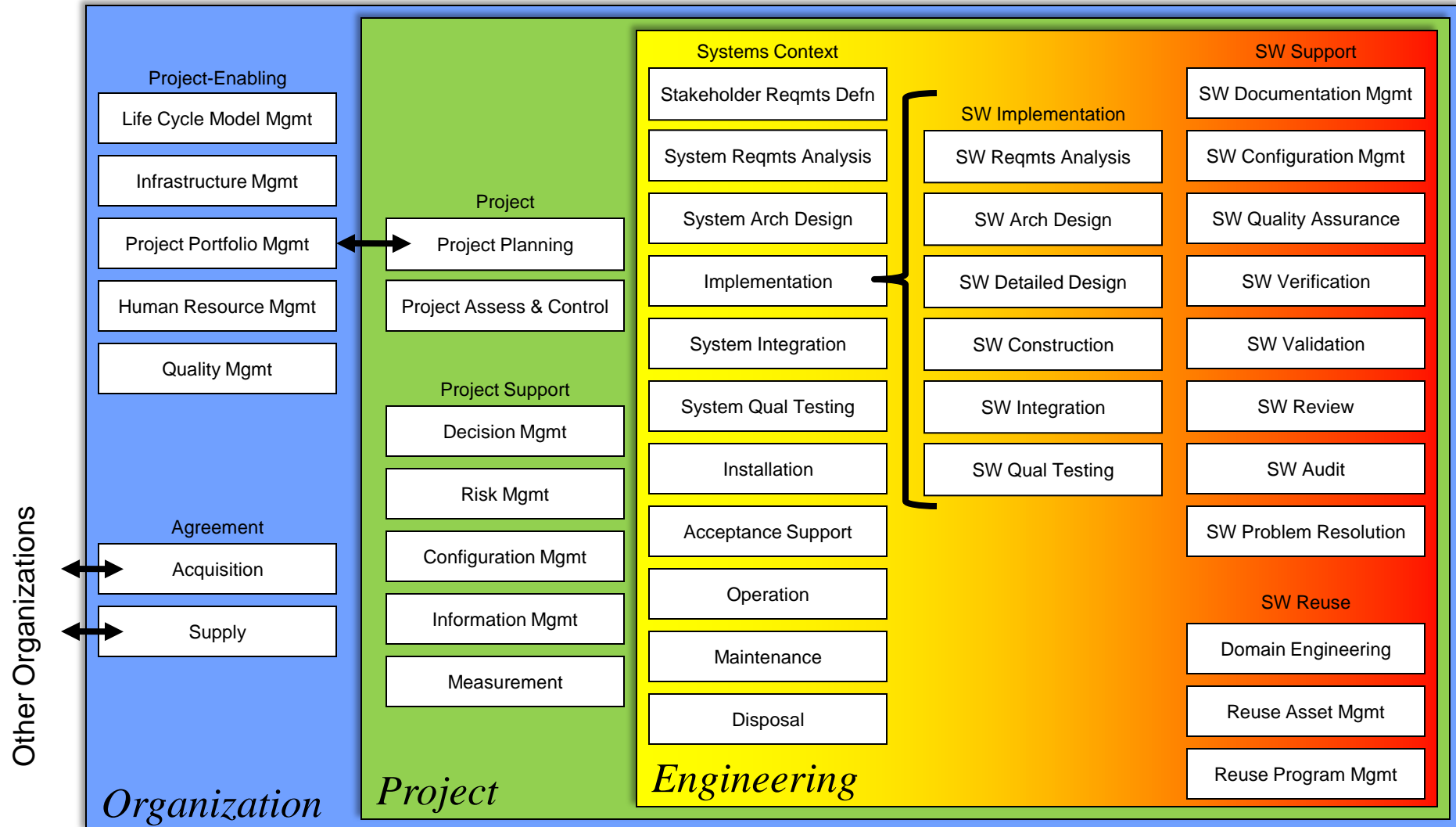
I need a process that isn't in the two standards

- **Actually, you don't ... With one exception*, all of the processes are already in the standard.**

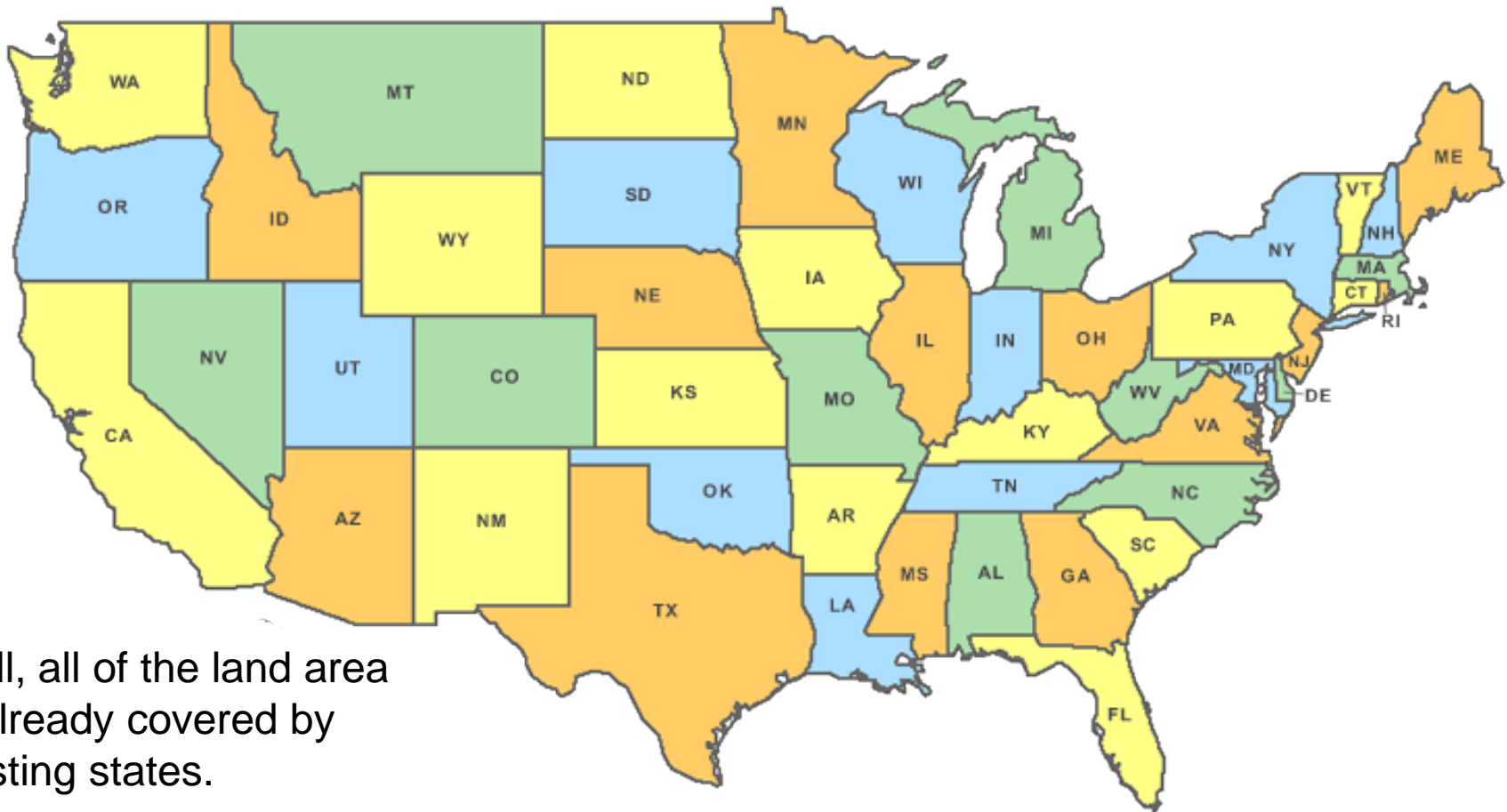


*** *The standards do not have a complete set of organizational processes. They include only the processes necessary to enable projects.***

I Need another process



I Need another state

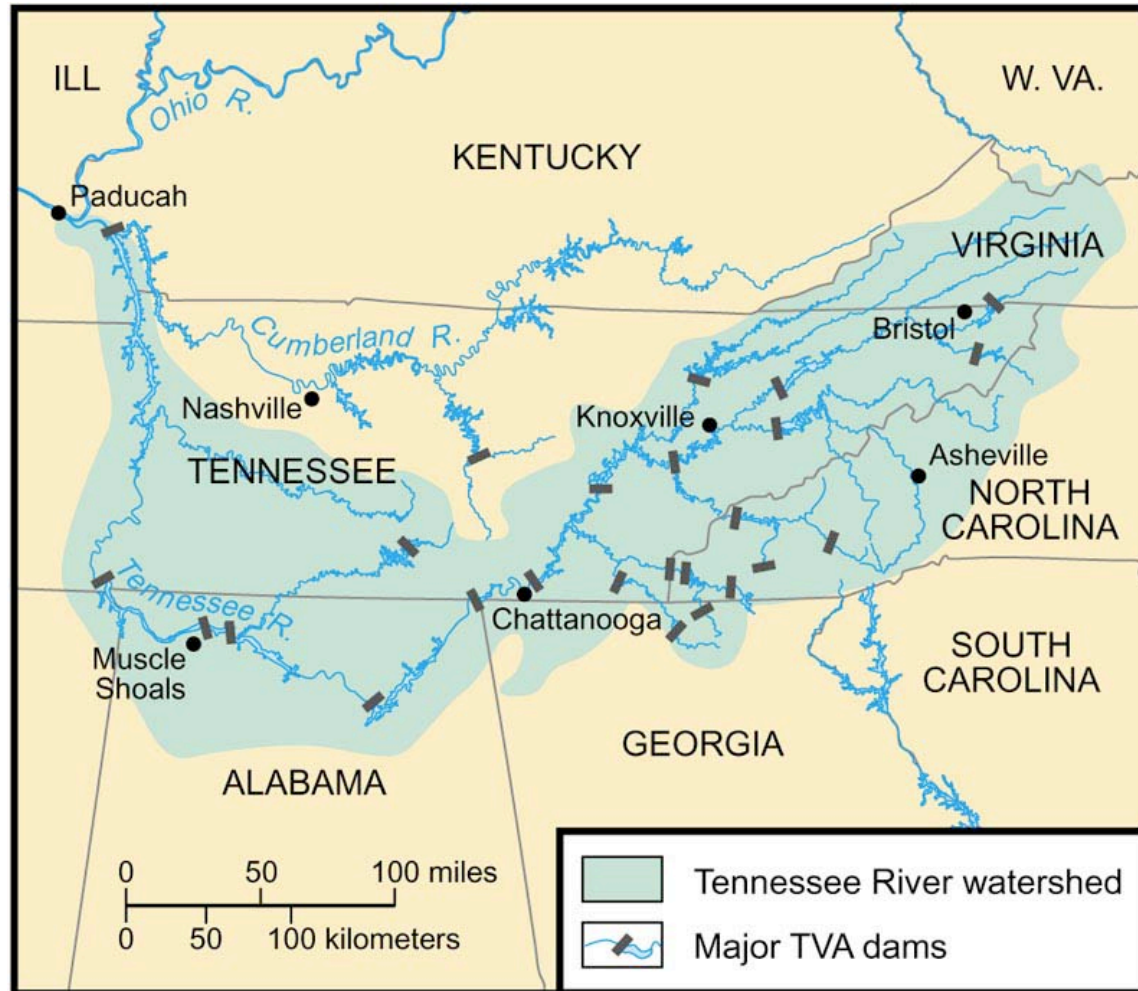


Well, all of the land area
is already covered by
existing states.

Similarly, all of the outcomes and activities in the life cycle of system or software are already provided by processes in the standards.



But the process I need isn't the same as any of the ones in the standards!



THE TENNESSEE VALLEY AUTHORITY

Process View

- The *process viewpoint* concept provides for particular engineering interests.
- A process view gathers in a single place the set of process activities that directly and succinctly address a concern.
- Like a process, the description of a process view includes a statement of purpose and outcomes.
- Unlike a process, the description of a process view does *not* include activities and tasks.
- Instead, the description includes guidance explaining how the outcomes can be achieved by employing the activities and tasks of the various processes in ISO/IEC 15288 and ISO/IEC 12207.

Application of process views

- Annex D of 15288 provides a process view for Specialty Engineering,
 - e.g., availability, maintainability, reliability, human factors – in fact, any of the “ilities”.
- Annex E of 12207 provides a process view for Usability (as described in ISO 13407, *Human-Centred Design*)
- The planned ISO/IEC/IEEE 15026-4 is a process view for systems and software assurance.
- Users of 12207 and 15288 can write their own process views
 - If they follow a few rules, then process views can be assessed for capability by ISO/IEC 15504 just like any other process.

Summary: ISO/IEC/IEEE 12207 and 15288...

- ... provide ***a broad set of coherent, cohesive processes*** that satisfy a variety of needs.
- ... provide ***varying levels of detail*** and information where to get more.
- ... provide ***conformance criteria that customers can easily understand*** and that separate responsible claimants from irresponsible ones.
- ... support ***adding processes, adding detail, and improving capability without causing incompatibility***.
- ... provide processes that are ***widely applicable***, yet ***capable of adaptation***.



Where can I get these standards and other information?

- ISO/IEC/IEEE 24765, Vocabulary, is available for free:
 - <http://www.computer.org/sevocab>
- All IEEE standards (including ones developed jointly with ISO/IEC) are available in MITRE's subscription to the IEEE Xplore digital library:
 - <http://ieeexplore.ieee.org/Xplore/dynhome.jsp?tag=1>
 - There is a special search category for standards permitting easy search via number.
- MITRE's customers can purchase standards:
 - IEEE: <http://standards.ieee.org> (click on **Shop** near top right)
 - ISO/IEC (via ANSI): <http://webstore.ansi.org/>
- For more information, contact me: Jim Moore, moorej@mitre.org
- Also, my 2006 book isn't *completely* obsolete yet ☺
 - James W. Moore, The Road Map to Software Engineering: A Standards-Based Guide, John Wiley / IEEE Computer Society Press, 2006, ISBN-10: 0471683620, ISBN-13: 978-0471683629

