Standards for Systems and Software Engineering: What Works?

Annette Reilly and Edward Addy
November 10, 2020
Annette Reilly, Ph.D., CSEP, PMP

IEEE Computer Society Representative to ISO/IEC JTC 1/SC7, Systems and software engineering

Co-editor of ISO/IEC/IEEE 12207, Systems and software engineering—Software life cycle processes

Editor of ISO/IEC/IEEE 15289, Systems and software engineering—Content of life-cycle information items (documentation)

Editor of ISO/IEC/IEEE 24765, Systems and software engineering—Vocabulary [www.computer.org/sevocab](http://www.computer.org/sevocab)

Member, IEEE Computer Society Standards Activities Board, Executive Committee of Systems and Software Engineering (S2ESC) Standards Committee, Society for Social Implications of Technology (SSIT) Standards Committee, and IEEE Standards Association Open Source Committee

IEEE Senior Life Member, IEEE Computer Society Golden Core

Lockheed Martin systems engineering manager and senior staff (retired)

annette.reilly@computer.org
“At lectures, symposia, seminars, or educational courses, an individual presenting information on IEEE standards shall make it clear that his or her views should be considered the personal views of that individual rather than the formal position, explanation, or interpretation of the IEEE.”

IEEE-SA Standards Board Operation Manual (subclause 5.9.3)
IEEE Systems and Software Engineering Standards

- Standards and IEEE
- What’s changing in systems/software engineering and its standards
- Key standards for systems/SW engineering
  - Vocabulary
  - Process descriptions
  - Life cycle processes for systems and software
  - Information management
  - IEEE 1012 Validation and Verification-Edward Addy
- New standards:
  - IEEE 7000 Ethically Aligned Design
  - IEEE 2675 DevOps
Standards and IEEE

- IEEE is the world’s largest technical society
  - 39 Technical Societies
  - Over 415,000 members
  - 44 Technical Committees
  - 72 active Standards Committees
  - Over 1000 active standards projects
  - Over 1200 active standards

- **IEEE systems and software engineering standards development is individually-based in an open process**
Where do standards come from?

- Issued by a respected, authoritative entity
  - IEEE – Computer Society and Standards Association
    - Individual based or Entity-based balloting
  - International Standards Organization (ISO)
  - International Electrotechnical Commission (IEC)
    - Nation-based balloting

- National standards organizations
  - American National Standards Organization (ANSI)
  - National Institute of Standards and Technology (NIST)
  - Other national organizations (BSI, Standards Bureau of Canada, etc.)

- CMMI Institute and Software Engineering Institute (SEI)
- Project Management Institute (Guidance)
- W3C (web standards)
- OASIS, OMG, TOG, DITA

For systems and software engineering, IEEE has a Partner Standards Development Organization (PSDO) agreement with ISO/IEC JTC 1 for development and adoption of identical standards
Beyond Systems and Software Engineering:
IEEE Computer Society standards

- Communications protocols: Ethernet IEEE 802™, audio, video coding
- Languages (POSIX IEEE 1003) and notations (VHDL, STIL)
- Design automation, smart manufacturing, AI
- Cryptography, Blockchain
- Cloud computing
- Learning technology, Test technology

Today we are focusing on Systems/Software Engineering standards.
IEEE Systems/SW Engineering Standards

- **S2ESC- Systems and Software Engineering Standards Committee**

- For over 25 years, IEEE S2ESC has engaged in joint standards development with ISO/IEC JTC 1/SC7 Software and systems engineering

- As of October 2020, there are 53 IEEE S2ESC standards
  - 14 IEEE-unique
  - 5 IEEE-adopted from ISO/IEC JTC 1
  - 2 ISO adopted from IEEE
  - 32 Joint ISO/IEC/IEEE

- Projects for 26 standards are underway in S2ESC
  - 17 of these are joint projects with ISO/IEC JTC 1/SC7
  - Note: Some projects are both completed and underway, i.e. completion followed by revision

- **Joint standards development for consistency and global applicability**
Why standards matter-- and what works?

What’s within reach that’s standardized?

What’s your favorite standard?
Why have standards?

- Support interoperability
- Support reliability
- Further world trade
- Promote consistent products
- Allow repeatable processes and process improvement
- Basis for regulations, contracts, and audits
Tensions in standards development

Proprietary and competing solutions

Advanced research theories

Old standards

National/regional interests
When is it really a standard?

- “... [A] formal document that establishes uniform engineering or technical criteria, methods, processes and practices” (Wikipedia)
- Not proprietary, tool-bound, or vendor-specific
- The result of consensus agreement from a balance of stakeholders
- Maintained by a recognized, impartial standards-producing organization
- Can be normative (mandatory) or guidance
- Open participation from all interested stakeholders
What’s different in systems/software engineering standards?

▸ System/product/service (specifications)
▸ Principles (attributes of IT governance)
▸ People (professional and organizational certification)
▸ Processes (from input to outcomes)

Which is more important to standardize: systems/software products, software engineering processes, software engineers?
How do standards get developed?

- Project proposal and approval
- Formation of a working group
- Development and review of drafts by a working group or fast-track of an existing document
- Ballot by interested parties
  - Possibility for public review and comment
- Revise and reballot
  - 24-48 months start to publish
- Final review and approval
- Publish, maintain, sell
- Periodic reviews – update, withdraw, stabilize
**Systems engineering**: interdisciplinary approach governing the total technical and managerial effort required to transform a set of customer needs, expectations, and constraints into a solution and to support that solution throughout its life.

**Software engineering**: application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software.

Source: ISO/IEC/IEEE 24765
What’s changing in systems engineering

- Systems thinking
- System of systems
- Model-based systems engineering (MBSE)
  - Advanced tool support
- Product line and variability engineering
- Virtual systems
- The system is software
What’s changing: software engineering

- Internet, big data, software as a service, the cloud, virtual systems, AI
- Highly complex integrated systems of systems
- Software construction through object-oriented, encapsulated, containerized methods
- Different ways of implementing software vs one-off custom SW development:
  - COTS integration, use of open source, services
- Using continuous, concurrent processes and methods
  - Agile, Dev Ops, integration, test-driven development
- Automation of software methods and tools
- Concern for security, data privacy
What has changed in the concept of software systems

1995 and 2008 Systems view of software:

- Hierarchical top-down system: Design the system, then design the software parts of the system
- Hierarchical view of software architecture (high-level, then detailed design, component to unit, then write code)
- Testing builds from unit to component to integrated system with a qualification test

2017 Software Engineering view

- The software is the system of interest.
- Non-software can be treated as system elements or infrastructure platform
- Software architecture, design, implementation, and test are continuous (DevOps)
- Views of software architecture use [encapsulated functions, objects, containers]
The Systems/SW Engineering Process Standards Landscape

ISO/IEC/IEEE 12207:2017

Management Systems
ISO 9001 Quality
ISO/IEC 20000 Service
ISO/IEC 27000 Security
ISO/IEC 19770 IT Asset Mgmt

Individual Processes
Information Mgmt: 15289
Verification/Validation
IEEE 1012...
Risk Management 16085

Capability Assessment

ISO/IEC/IEEE 24765 Vocabulary
www.computer.org/sevocab

ISO/IEC/IEEE 24774 Process Description


15288:2015 and 12207:2017
Uniform set of processes with identical process purpose and outcomes
System concepts, tasks, activities, and notes will vary

© Annette Reilly, 10 Nov 2020
The harmonized 12207/15288 process model

4 process groups
- 30 processes

* New
# Split or renamed
ISO/IEC/IEEE 24765-Vocabulary

- Latest version is freely available at computer.org/sevocab
- Over 240 sources, 6900 definitions, up to 800 changes annually

Welcome to SEVOCAB: Software and Systems Engineering Vocabulary

Find authoritative definitions for software and systems engineering terms in SEVOCAB. A project of the IEEE Computer Society and ISO/IEC JTC 1/SC7, SEVOCAB includes definitions from international standards. You can search for a term as defined in the standards, or for all the definitions in a source standard. To give you an understanding of related concepts, SEVOCAB will return any definition for the term, as well as all the definitions that use the term.
Changes in process descriptions: ISO/IEC/IEEE 24774

- From a project view to an organizational view
  - Processes are organizational assets

- Recognition of a variety of life-cycle models

- Cleaner delineation of processes
  - Processes occur concurrently
  - Processes involve strategy, planning, execution, reporting and control, and improvement

- Processes rather than procedures
  - Why and what, not so much how or who or when

New version due 1Q 2021
Generic process activities

- Strategize and Plan
- Perform
  - Do activities and tasks
- Evaluate and decide
  - Check, act, improve
- Manage outcomes and outputs
  - Trace and reuse work products and information

Now: Conformance to process outcomes
OR conformance to tasks and activities
Focus on why: purpose and outcomes

Process interrelationships

Technical Processes
Clauses 6.4.1 - 6.4.14

System Analysis
Clause 6.4.6

System Deployment and Use
Transition
Clause 6.4.10
Operation
Clause 6.4.12
Maintenance
Clause 6.4.13
Disposal
Clause 6.4.14

System Realization
Validation
Clause 6.4.11
Verification
Clause 6.4.9
Integration
Clause 6.4.8
Implementation
Clause 6.4.7


× “The” life-cycle model

× Complete, normalized input-activity-output models
  • Outcomes are not the same as Outputs

× Complete logical model linking various processes (waterfall or spaghetti nest?)
  • Processes can and should be done concurrently

× Process Assessment Model

× Capability-maturity model or improvement process
  • Officially, it’s all Level 1

× Complete listing of required work products, outputs, information items (documentation) or artifacts
  • Flexible listings (Annex B) of possible work products
## Information and content management trends

<table>
<thead>
<tr>
<th>Was</th>
<th>Now</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Item Description with outline</td>
<td>Contents</td>
</tr>
<tr>
<td>Exhaustive list of plans, procedures, reports for every process</td>
<td>Information management system (dashboard and alerts); Select and agree on needed internal and deliverable documents</td>
</tr>
<tr>
<td>Documented, hierarchical design descriptions</td>
<td>Integrated tools tracking and hosting requirements, design models, software objects, test scripts, test results, versions, problems/defects</td>
</tr>
<tr>
<td>Multiple plans (documents)</td>
<td>Feature toggles</td>
</tr>
<tr>
<td></td>
<td>Strategy and planning</td>
</tr>
</tbody>
</table>
ISO/IEC/IEEE 15289 Information items

- Identifies information items for each process in ISO/IEC/IEEE 15288 and ISO/IEC/IEEE 12207
- Information items can be combined or subdivided
- Information can be repurposed and reused throughout the life cycle
- Identifies contents of records and generic information items:
  - Description, plan, policy, procedure, report, request, specification
- Identifies contents of 80+ specific information items
  - Does not include template outlines
Software and Systems Engineering Standards Committee

1012 IEEE Standard for System, Software, and Hardware Verification and Validation

Edward Addy, PhD, PMP
Chair, IEEE CS S2ESC
Chair, P1012 Working Group

10 November 2020
Software and Systems Engineering Standards Committee (S2ESC)

• Many interesting and important standards under S2ESC

• This presentation will focus on one of the earliest standards developed by the Computer Society in software and systems engineering:

  1012
  IEEE Standard for Verification and Validation of Systems, Software, and Hardware
Origin of Verification and Validation

• Verification and Validation has roots in systems engineering principles of satellite systems and missile launch systems in 1960s and 1970s
• Verification and Validation processes are interrelated and complimentary
  • Each has activities performed throughout life cycle
  • Share methods and tools
• Because of interrelationship, Verification and Validation are often cited as “V&V”
Meaning of Verification and Validation

• Verification process provides objective evidence for whether the products successfully complete each life cycle activity
  • “the product is built correctly”

• Validation process provides objective evidence for whether the products satisfy intended use and user needs
  • “the correct product is built”
First Version of 1012

• Original version of 1012 published in 1986

• Contributors to the first version were from the defense industry, medical device area and nuclear power industry

• Most volunteers from U.S., with some from U.K. and Canada

• First version focused on the content of the Software V&V Plan
Next Versions of 1012

• Two versions published in 1998 and 2004

• Standard became focused on the activities and tasks of the software V&V processes

• Aligned with ISO/IEC 12207 Software Life Cycle Processes (now jointly developed with IEEE)

• Working group participants included same industries as the original version, plus transportation, space, and non-defense federal government
Latest Versions of 1012

• 2012 and 2016 versions increased scope by including V&V of systems and hardware in addition to software

• Aligned with the software engineering processes of ISO/IEC/IEEE 12207 and the systems engineering processes of ISO/IEC/IEEE 15288

• Decreased participation in working group from defense and medical devices, increased participation from nuclear power; increased international participation especially from China
Users of 1012

• 1012 is one of the most widely used systems and software engineering standards

• U.S. government organizations that are mandating the use of 1012 include:
  - DoD
    - Nuclear weapon systems
    - Satellite communication systems
  - IRS
  - Homeland Security
  - FDA (Radiation therapy devices)
  - FAA (air traffic control)
  - NASA
    - Manned space missions
    - Critical interplanetary space missions
  - NRC
    - Nuclear Power Plant Instrumentation & Controls
  - State/local government IT projects
Independence of V&V

• A key element of V&V is determining the level of independence appropriate between those involved in the development of the system or its elements and those who are performing V&V on the system products.

• The parameters of independence are:
  • Technical
  • Managerial
  • Financial

• When all three parameters are fully satisfied, the effort is called Independent Verification and Validation (IV&V).
Tailoring of V&V

• The V&V effort should be tailored to the individual system, since not all systems need the same level of V&V

• This tailoring is performed using the Integrity Level of the system and its elements
  • Systems with a higher integrity level need a larger set of V&V activities that are performed with more rigor and intensity

• 1012 describes the set of tasks to be performed using a four-level schema, but allows other schemas to be used
V&V in different life cycle models

• V&V tasks are identified to perform assessments of the products produced from the other technical life cycle processes

• The life cycle processes are arranged by the organization responsible for the system to describe the system life cycle model

• 1012 can be applied using any life cycle model that is based on life cycle processes
S2ESC Standards—What’s New
IEEE 2675 DevOps-Building Reliable and Secure Systems Including Application Build, Package and Deployment

- DevOps principles applied in a secure and reliable way throughout the life-cycle
  - Mission first
  - Customer focus and stakeholder involvement
  - Continuous everything: continuous integration/continuous delivery
    - Build the pipeline,
  - Systems thinking and risk-based approach
  - Organizational culture: leadership, communication and collaboration, knowledge management, adaptation and resilience

- Applies a DevOps view to each of the 12207/15288 life cycle processes

- Anticipated publication: February 2021
IEEE 7000 - Model Process for Addressing Ethical Concerns during System Design

- Identifying, analyzing, and resolving ethical issues early in the life cycle or for new versions of products or services
- A process standard: Applicable during concept exploration, ethical values elicitation and prioritization, requirements definition, and design
- Emphasis on transparency and risk reduction
- Choice of values relevant to the culture where the system will be deployed
  - Improving the value proposition and reducing risk.
IEEE 7000 concepts

Value cluster contains
Core value

Values can be derived from, e.g.:

Value in human conduct:
Virtue

Principle applied to a person’s actions:
Personal maxim

System is a Value bearer
System features and functions show Value dispositions

DRAFT pending final balloting
Anticipated publication mid-2021

Value Lead Performs Conceptual Value Analysis (CVA)

Value demonstrator translates into
Ethical value requirement (EVR)
Value-based system requirement

Existing Ethical principle

Value-based Requirements translate into
Ethical risk-based design
Get involved with standards

- Have models for your process and products
- Use standards at work
  - Get IEEE standards at http://www.techstreet.com/ieee
- Become a standards reviewer, baloter, or working group member
- Join IEEE- Standards Association and IEEE-Computer Society
Backup

Source: Dario Sabljak/Shutterstock
The 12207:1995 Process Model

5. PRIMARY LIFE CYCLE PROCESSES

5.1 Acquisition
5.2 Supply
5.3 Operation
5.4 Development
5.5 Maintenance

6. SUPPORTING LIFE CYCLE PROCESSES

6.1 Documentation
6.2 Configuration Management
6.3 Quality Assurance
6.4 Verification
6.5 Validation
6.6 Joint Review
6.7 Audit
6.8 Problem Resolution

7. ORGANIZATIONAL LIFE CYCLE PROCESSES

7.1 Management
7.2 Infrastructure
7.3 Improvement
7.4 Training

3 process groups
17 processes
The 12207:2008 process model

- 4 System Context process groups
- 25 systems processes
- 3 software-specific process groups
- 18 software processes
Processes removed from the 2008 process model for 12207:2017

© Annette Reilly, 10 Nov 2020 | 47