

# STANDARDS AND GUIDANCE FOR SECURE ORGANIZATIONAL PROCESSES

**Paul Croll**  
**CSC**  
**Fellow**  
**September 26, 2011**

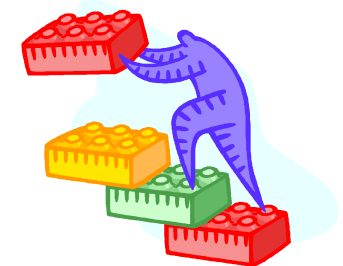
**10<sup>th</sup> Annual Mid-Atlantic Software  
Quality & Program Management  
Conference**



## Enterprise Risk Management – Balancing Assurance Costs

- NIST Special Publication 800-30, Risk Management Guide for Information Technology Systems describes risk management for IT systems as a process that balances the operational and economic costs of protective measures to achieve mission-essential security capabilities
- NIST Special Publication 800-27, Engineering Principles for Information Technology Security (A Baseline for Achieving Security), Revision A, recognizes that elimination of all risk is not cost-effective
  - *A cost-benefit analysis should be conducted for each proposed control. In some cases, the benefits of a more secure system may not justify the direct and indirect costs. Benefits include more than just prevention of monetary loss; for example, controls may be essential for maintaining public trust and confidence*

***Principle 5: Reduce risk to an acceptable level***



## Risk Hierarchy for Enterprise Security

- Legal and Regulatory Risk
  - This class of risk addresses risks associated with failures regarding compliance with legal or regulatory requirements
  - Consequences may include fines, civil or criminal prosecution, prohibitions against provision of products to the market place.
- Operational Risk
  - This class of risk addresses both external and internal risk
    - External risks associated with failures of provided products in their operational environments,
    - Internal risks associated with failures in the engineering processes producing such products.
  - Consequences may include delivered exploitable vulnerabilities that result in harm to users, their systems, or their data
- Reputational Risk
  - This class of risk is linked with legal and regulatory, operational, and competitive risk
  - It addresses risks associated with damages to the organization's reputation in the market place resulting from legal and regulatory breaches and operational failures
  - Consequences include loss of standing in the market place and mistrust on the part of current and potential customers.
- Competitive Risk
  - This class of risk addresses risks associated with loss of stature with respect to competitors.
  - Consequences include loss of market share and potential difficulty entering new markets.
- Financial Risk
  - This class of risk addresses risks associated with monetary loss
  - Consequences include loss of revenue, negative impact on stock prices, and diminishing shareholder confidence.
- Strategic Risk
  - This class of risk is linked with all the other risk classes below it in the hierarchy
  - It addresses risks associated with failures to meet the strategic goals and objectives of the organization



## Governance for Software Assurance



*Directing and controlling an organization to establish and sustain a culture of security in the organization's conduct (beliefs, behaviors, capabilities, and actions).*

*Governing for enterprise security means viewing adequate security as a non-negotiable requirement of being in business.*

*Source: Julia Allen, Governing for Enterprise Security, CMU/SEI-2005-TN-023, June 2005*

## The Governance Context for Assurance

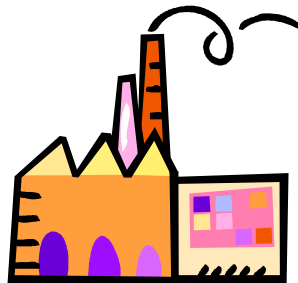
- In the U. S. Federal marketplace alone, there are over two hundred governance documents related to system and software assurance
  - An example for the US DoD is provided on the next slide
- A recent U. S. Congressional Budget Office review estimated the cost of implementing the Federal Information Security Act of 2008 (FISMA) alone, designed to improve information security throughout the federal government, at US \$40 million in 2009 and about US \$570 million over the 2009-2013 period.
- These external governance requirements drive internal governance structures that must be both responsive and cost-effective, while providing value to all stakeholders





## Governance Classes

- In performing the engineering trades associated with system and software assurance, governance documents of various classes define compliance and conformance requirements that may constrain the trade space. These may include:
  - Legal and regulatory requirements
  - Industry standards
  - Client-imposed requirements
  - Internal guidelines



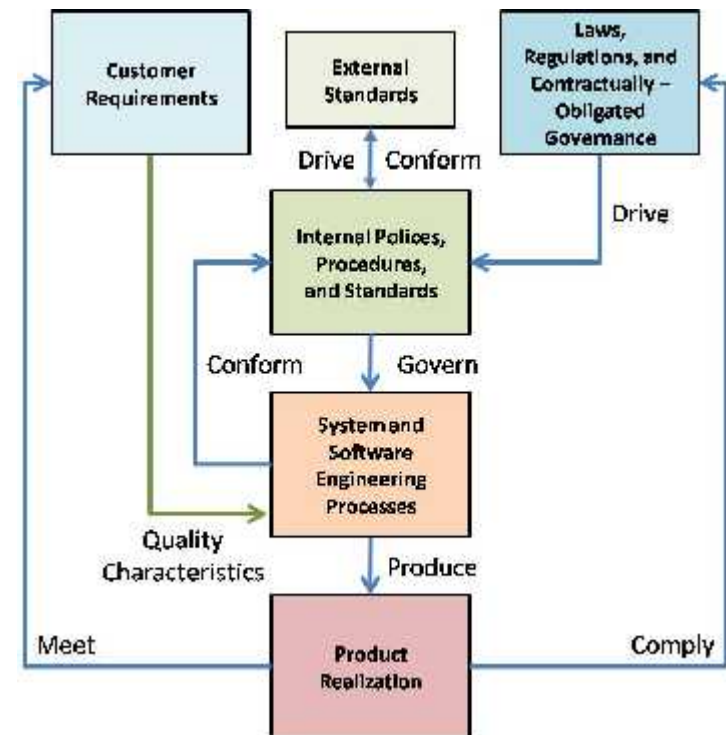
## Compliance vs. Conformance

- There is a difference between compliance and conformance
  - Compliance refers to mandatory adherence to laws, rules, and regulations
  - Conformance refers to voluntary adherence to standards and best practices.
- Compliance requirements and conformance objectives are addressed as part of an organization's business strategy through the development and promulgation of an internal governance structure consisting of:
  - Policies
  - Procedures
  - Standards
  - Practices
- These are aligned with external compliance and conformance drivers



## Governance in the Engineering Life Cycle

- Customer requirements for the system, defining the system's quality requirements, set the expectations for the system. It is against these quality requirements that engineering trades will be made.
- Applicable laws, regulations, and other contractually-obligated governance set the constraints bounding the engineering trade space.
- External standards drive Internal policies, procedures, and standards
- Internal policies, procedures, and standards institutionalize external governance requirements (as well as external standards and business best practices) and drive the engineering processes for producing systems and software
- Engineering processes produce the product by trading off internal governance requirements along with customer quality requirements, to facilitate optimization among quality characteristics and compliance with external governance requirements.

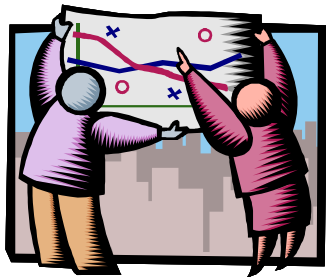


## Processes and Standards for Assurance



# Using Process Benchmarking and Standards to Engineer for Software Assurance

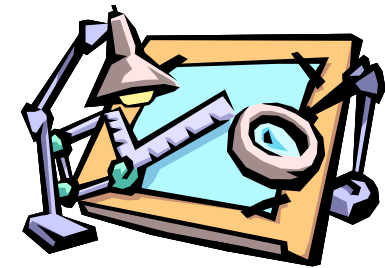
1. Understand Your Legal, Regulatory, and Business Requirements for Assurance



5. Measure Your Results – Assess Risk and Modify Your Processes as Necessary



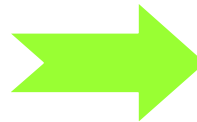
4. Build, or Refine, and Execute Your Assurance Processes



2. Look to models like the CMMI® for Process-Related Capability Expectations



3. Look to Standards for Assurance Process Detail



## CMMI ® V1.3 OPF SP 1.1 Includes Examples Of Standards For Addressing Cyber Challenges

### Subpractices

1. Identify policies, standards, and business objectives that are applicable to the organization's processes.

Examples of standards include the following:

- ISO/IEC 12207:2008 Systems and Software Engineering – Software Life Cycle Processes [ISO 2008a]
- ISO/IEC 15288:2008 Systems and Software Engineering – System Life Cycle Processes [ISO 2008b]
- ISO/IEC 27001:2005 Information technology – Security techniques – Information Security Management Systems – Requirements [ISO/IEC 2005]
- ISO/IEC 14764:2006 Software Engineering – Software Life Cycle Processes – Maintenance [ISO 2006b]
- ISO/IEC 20000 Information Technology – Service Management [ISO 2005b]
- Assurance Focus for CMMI [DHS 2009]
- NDIA Engineering for System Assurance Guidebook [NDIA 2008]
- Resiliency Management Model [SEI 2010c]

## CMMI® V1.3 OPF SP 1.1 Includes Examples Of Standards For Addressing Cyber Challenges

### Subpractices

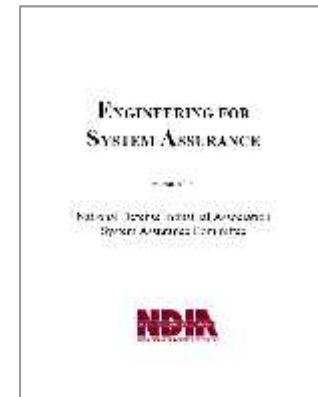
1. Identify policies, standards, and business objectives that are applicable to the organization's processes.

Examples of standards include the following:

- ISO/IEC 12207:2008 Systems and Software Engineering – Software Life Cycle Processes [ISO 2008a]
- ISO/IEC 15288:2008 Systems and Software Engineering – System Life Cycle Processes [ISO 2008b]
- ISO/IEC 27001:2005 Information technology – Security techniques – Information Security Management Systems – Requirements [ISO/IEC 2005]
- ISO/IEC 14764:2006 Software Engineering – Software Life Cycle Processes – Maintenance [ISO 2006b]
- ISO/IEC 20000 Information Technology – Service Management [ISO 2005b]
- Assurance Focus for CMMI [DHS 2009]
- NDIA Engineering for System Assurance Guidebook [NDIA 2008]
- Resiliency Management Model [SEI 2010c]

# DoD-Related Standards Based Guidance For Systems Assurance

- National Defense Industrial Association Guidebook on Engineering for System Assurance (<http://www.acq.osd.mil/sse/docs/SA-Guidebook-v1-Oct2008.pdf>)
  - Intended to supplement the knowledge of systems (and software) engineers who have responsibility for systems for which there are assurance concerns
    - General Guidance mapped to ISO/IEC/IEEE 15288, System Life Cycle Processes
    - DoD Specific Guidance, mapped to DoD Acquisition Life Cycle
      - Anti-Tamper
      - DAG Lifecycle Framework
      - Technology Development Phase
      - System Development & Demonstration Phase
      - Production, Deployment, Operations, & Support Phases
      - Supporting Processes
      - Periodic Reports
      - Supplier Assurance
      - Mappings
    - Correspondence with Existing Documentation, Policies, and Standards
      - Executive Policy, Services Standards, NIST/NSA (NIAP) Standards, GEIA, AIA, IEEE, ISO Standards, Best Practice (e.g., DHS/DOD SwABOK)
  - Adopted as NATO AEP-67, Engineering for System Assurance in NATO Programmes, February 2010



# NDIA System Assurance Guidebook – Mapped To ISO/IEC/IEEE 15288, System Life Cycle Processes

- Agreement Processes
  - Acquisition
  - Supply
- Project Processes
  - Project Planning
  - Project Assessment
  - Project Control
  - Decision-making
  - Risk Management
  - Configuration Management
  - Information Management
- **Assurance Case Process**

---

- Enterprise Processes
  - Acquisition
  - Enterprise Environment Management
  - Investment Management
- Technical Processes
  - Stakeholder Requirements Definition
  - Requirements Analysis
  - Architectural Design
  - Implementation
  - Integration
  - Verification
  - Transition
  - Validation
  - Operation
  - Maintenance
  - Disposal
- System Life Cycle Process Management
- Resource Management [including human resource training]
- Quality Management

## Guidebook Example – Architectural Design

- Define appropriate logical architectural designs and methodologies. When identifying the architectural elements and their interactions, consider the following general issues:
  - Attempt to separate elements that are highly critical from those that are not
    - Attempt to make the critical elements easier to assure by making them smaller, less complex, and more isolated from impact by other elements
  - Include defensive elements whose job is to protect elements from each other and/or the surrounding environment (e.g. potentially untrustworthy sources)
  - Individually secured elements are not enough; the composition of the elements and their interconnections also matter.
    - understanding the interrelationships between elements and their linkages will help in addressing potentially weak areas in the design.
  - Use defense-in-depth measures where appropriate
    - Use multiple independent layered mechanisms to protect a system's critical functionality to make it more difficult for attackers to succeed.
  - Beware of maximizing performance (including throughput) to the detriment of assurance
- Some specific approaches to consider when developing the architecture are:
  - Least privilege
  - Isolation/containment (e.g. sandboxes, layered interfaces, element wrappers, virtual machines)
  - Monitoring and response for both legitimate and illegitimate actions
  - Tolerance (resiliency)

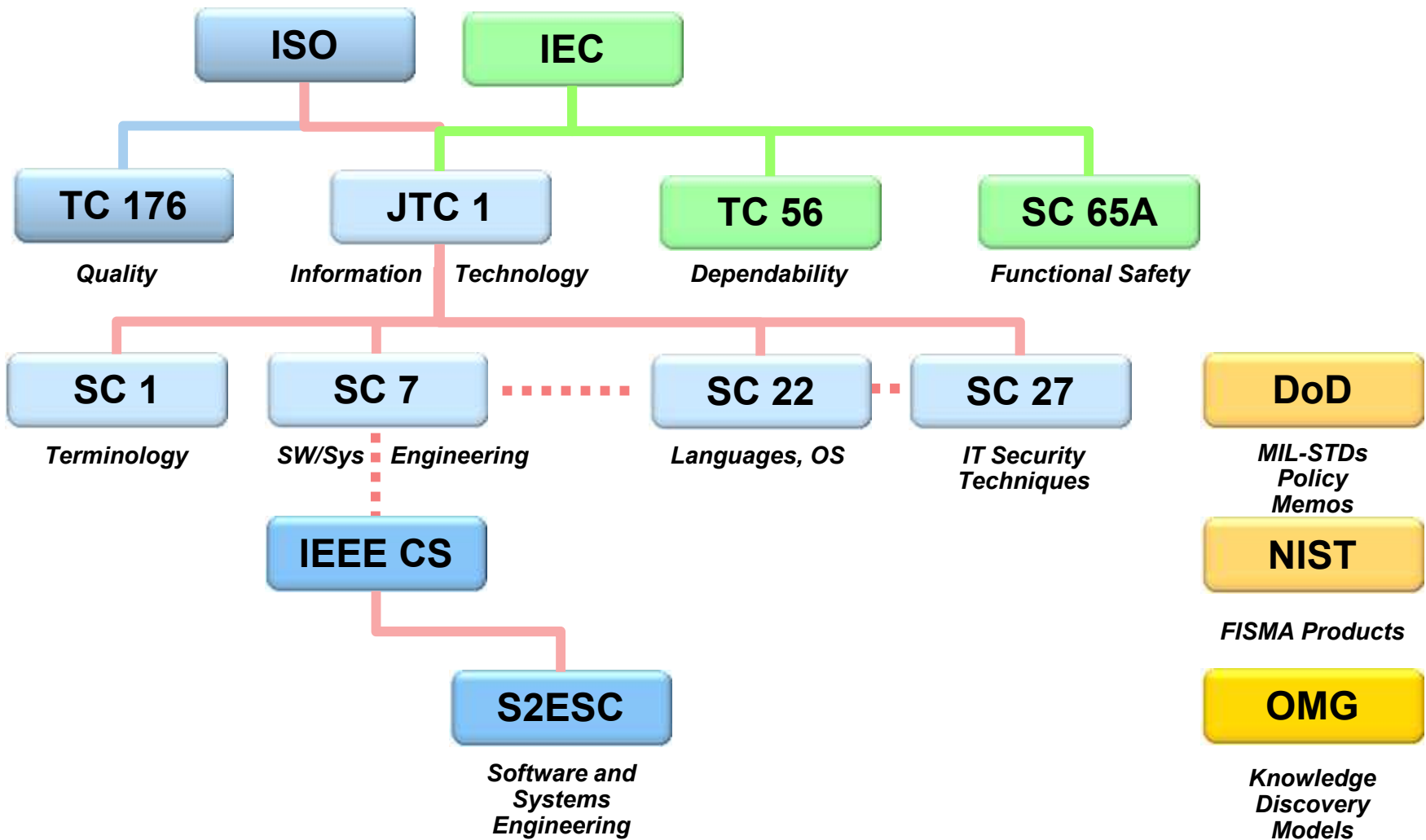
## Guidebook Example – Static Code Analysis – Preliminary Design Review (PDR) Outcomes

- Information security technology evaluation of all critical COTS/GOTS elements
  - Performed as part of the analysis of alternatives.
  - Includes an updated assurance case based on the design, and new weaknesses and vulnerabilities identified.
  - Results of static code analyses performed of GOTS/COTS components.
    - Which tools were used?
    - What weaknesses and vulnerabilities were discovered
- Specification of assurance-specific static analysis
  - Specification of assurance-specific static analysis and assurance-specific criteria to be examined during code reviews
    - Code reviews performed during implementation
    - Documented in the System Engineering Plan (SEP) and Software Development Plan (SDP)
    - Plan for training to use static analysis tools and for manual analysis
- Configuration management
  - For assurance, the preliminary configuration management plan must support traceability and protection of each configuration item, including requirements and architectural elements.
    - At what stages of the configuration management process will static code analysis be applied?
    - What configuration change events will trigger code analysis?
    - What components will be analyzed?
    - How will the results of the analyses be documented?

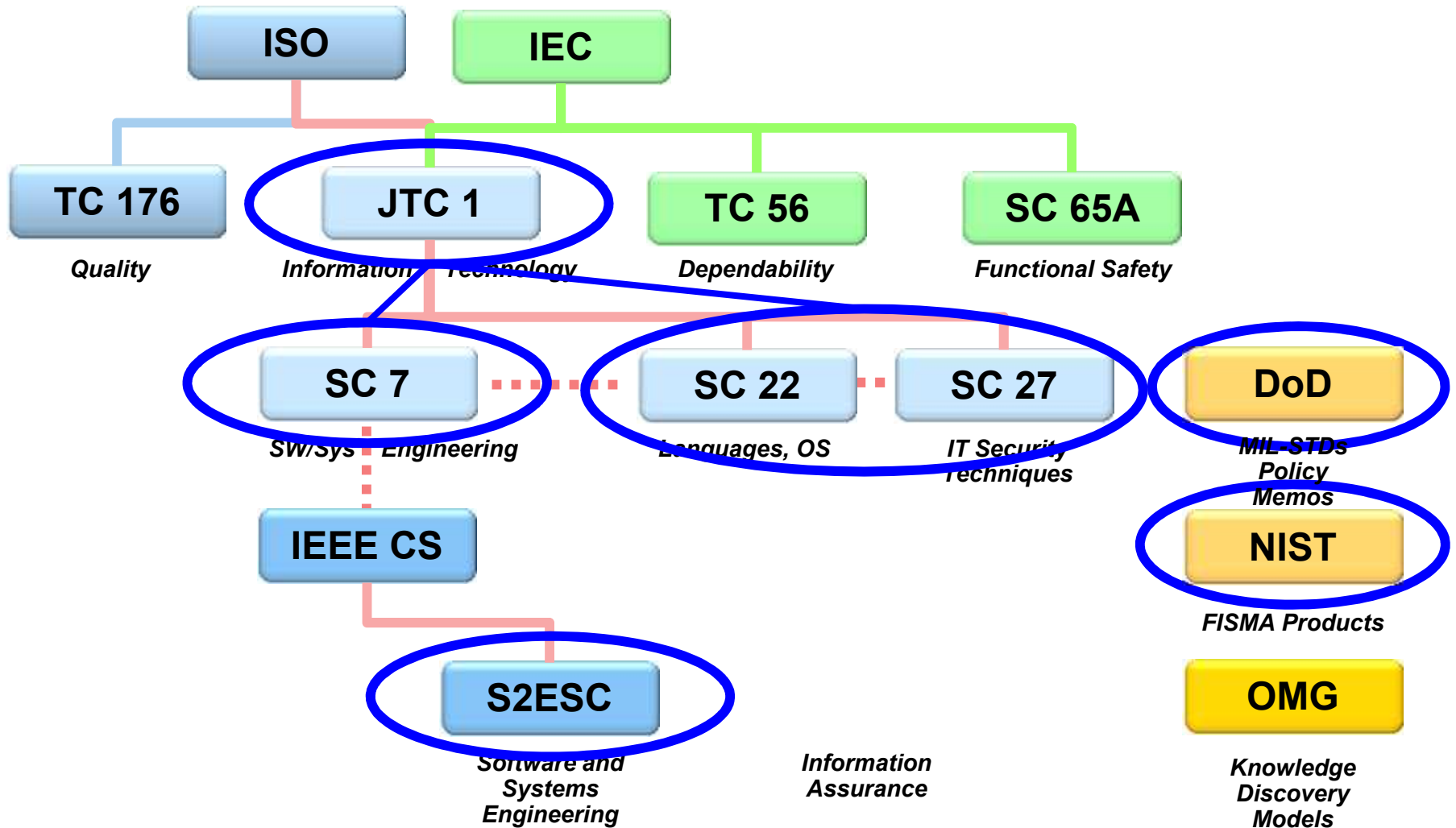
## Standards for Assurance



# Standards Organizations Supporting Assurance



# Standards Organizations Supporting Assurance



## ISO/IEC JTC1 Standards for Assurance

**SC 7**

**ISO/IEC/IEEE 15026**  
**System and Software Assurance**

**SC 22**

**ISO/IEC TR 24772**  
**Programming Language Vulnerabilities**

**SC 27**

**ISO/IEC 15408**  
**Common Criteria for IT Security Evaluation**

**ISO/IEC 21827**  
**System Security Engineering Capability Maturity Model (SSE CMM)**

**ISO/IEC 27000 series**  
**Information Security Management Systems (ISMS)**

**ISO/IEC WD 27036-3**  
**Supply Chain Risk Management (Preliminary Draft)**

# IEEE Standards for Systems and Software Assurance



**ISO/IEC/IEEE 15026**  
**System and Software Assurance**

**IEEE 730**  
**Software Quality Assurance Plans**

**IEEE 828**  
**Software Configuration Management**

**IEEE 830**  
**Software Requirements Specifications**

**IEEE 1008**  
**IEEE Standard for Software Unit Testing**

**IEEE Std 1012**  
**System and Software Verification and Validation**

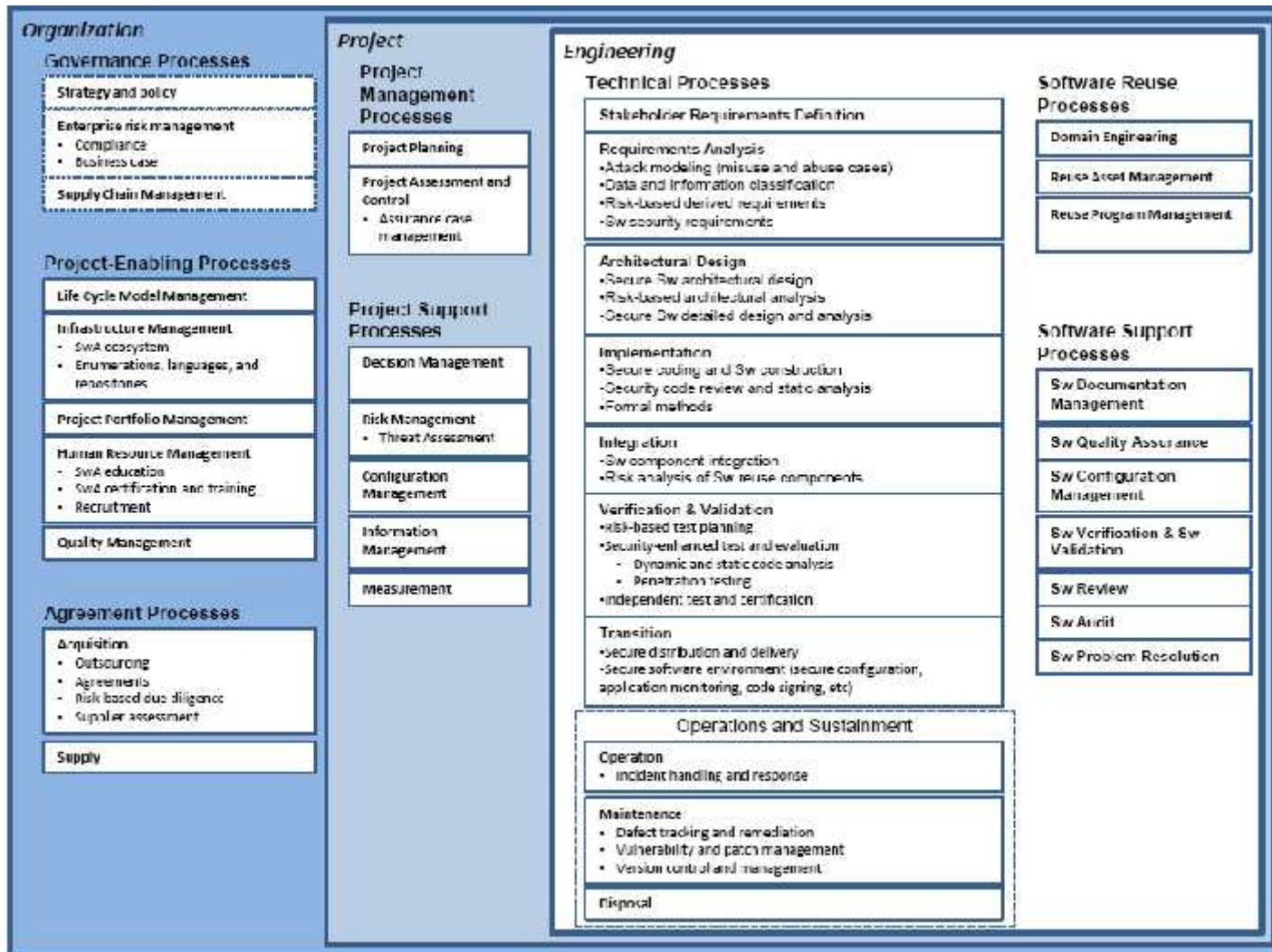
**IEEE Std 1016**  
**Software Design Descriptions**

**IEEE Std 1233**  
**System Requirements Specifications**

Over 40 standards in the  
S3ESC Collection  
[http://www.computer.org/port  
al/web/s2esc](http://www.computer.org/port<br/>al/web/s2esc)

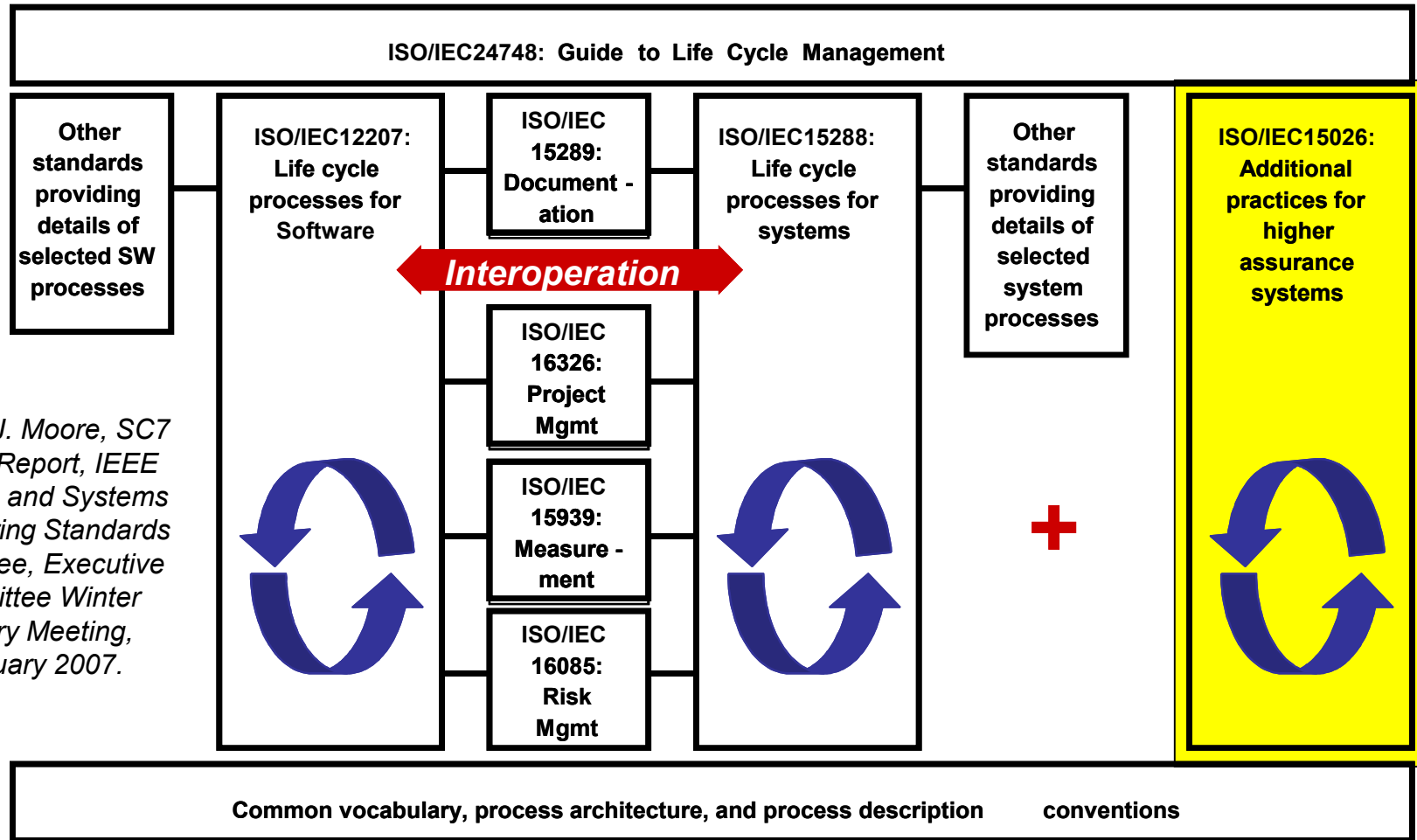
# The ISO/IEC/IEEE Life Cycle Process Framework Standards 15228 and 12207

SC 7  
S2ESC



# Assurance in the ISO/IEC JTC1/SC7/IEEE System and Software Life Cycles

SC 7  
S2ESC



Source: J. Moore, SC7 Liaison Report, IEEE Software and Systems Engineering Standards Committee, Executive Committee Winter Plenary Meeting, February 2007.

# ISO/IEC/IEEE 15026, System and Software Assurance

SC 7

S2ESC

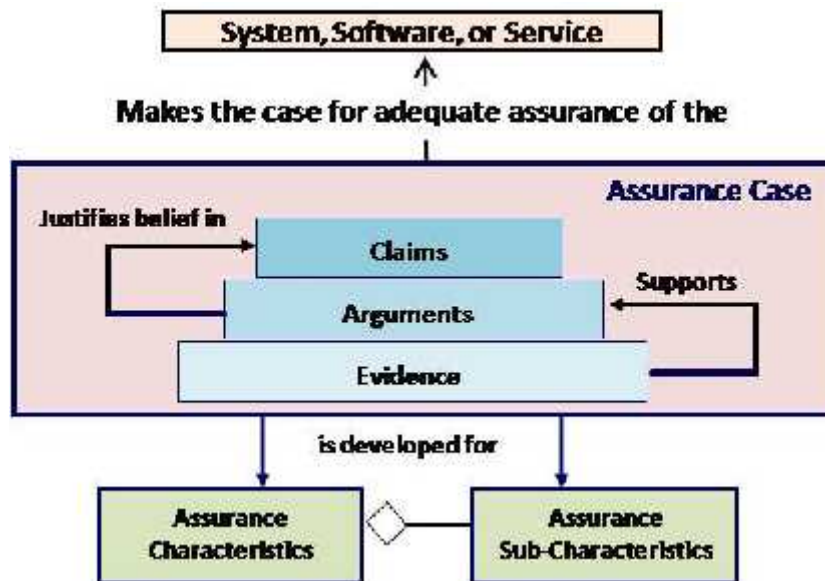
- A four-part standard
  - 15026-1: Concepts and vocabulary
    - Initially a Technical Report
  - 15026-2: Assurance case
    - Includes requirements on the assurance case content and the life cycle of the assurance case itself, as well as an informative clause on planning for the assurance case
  - 15026-3: System integrity levels (a revision of the 1998 standard)
    - Relates integrity levels to the assurance case and includes related requirements for their use with and without an assurance case
  - 15026-4: Assurance in the life cycle
    - Addresses requirements and guidance regarding how claims of assurance are treated in life cycle processes.

# The ISO/IEC/IEEE 15026 Assurance Case

- Set of structured assurance claims, supported by evidence and reasoning, that demonstrates how assurance needs have been satisfied.
  - Shows compliance with assurance objectives

- **Sub-parts**

- A high level summary
- Justification that product or service is acceptably safe, secure, or dependable
- Rationale for claiming a specified level of safety and security
- Conformance with relevant standards and regulatory requirements
- The configuration baseline
- Identified hazards and threats and residual risk of each hazard and threat
- Operational and support assumptions



- Attributes**
- Clear
  - Consistent
  - Complete
  - Comprehensible
  - Defensible
  - Bounded
  - Addresses all life cycle stages

## ISO/IEC TR 24772, Programming Language Vulnerabilities

SC 22

- A catalog of 60+ issues that arise in coding when using any language and how those issues may lead to security and safety vulnerabilities
- Cross-referenced to [CWE](#) (Common Weakness Enumeration database)
- Each discussion includes
  - Description of the mechanism of failure
  - Recommendations for programmers: How to avoid or mitigate the problem
  - Recommendations for standardizers: How to improve programming language specifications
- Second edition will add annexes specific to particular programming languages

## ISO/IEC 27001, Information Security Management Systems – Requirements

**SC 27**

- Specifies the requirements for establishing, implementing, operating, monitoring, reviewing, maintaining and improving a documented Information Security Management System (ISMS) within the context of the organization's overall business risks
- Specifies requirements for the implementation of security controls customized to the needs of individual organizations
- Designed to ensure the selection of adequate and proportionate security controls that protect information assets and give confidence to interested parties

# ISO/IEC 27002, Code of Practice For Information Security Management

**SC 27**

- Establishes guidelines and general principles for initiating, implementing, maintaining, and improving information security management in an organization
- Provides general guidance on the commonly accepted goals of information security management
- Contains best practices of control objectives and controls in the following areas of information security management:
  - security policy;
  - organization of information security;
  - asset management;
  - human resources security;
  - physical and environmental security;
  - communications and operations management;
  - access control;
  - information systems acquisition, development and maintenance;
  - information security incident management;
  - business continuity management;
  - compliance

## ISO/IEC 27003, Information Security Management System Implementation Guidance

SC 27

- Focuses on the critical aspects needed for successful design and implementation of an Information Security Management System (ISMS)
- Describes the process of ISMS specification and design from inception to the production of implementation plans, including:
  - Obtaining management approval to implement an ISMS
  - Project definition for implementation of an ISMS
  - How to plan an ISMS project

## ISO/IEC 27034, Application Security

SC 27

- Four part standard
  - Part 1 –Application security overview and concepts
  - Part 2 – Organization normative framework
  - Part 3 – Application security management process
  - Part 4 – Application security validation
- Process-based
  - Not a software application development standard, an application project management standard, nor a software development cycle standard
  - Provides general guidance that will be supported by more detailed methods and standards

# ISO/IEC 27036, Information Security for Supplier Relationships

SC 27

- Four part standard
  - Part 1 – Overview and Concepts
  - Part 2 – Generic Requirements
  - **Part 3 – ICT Supply Chain Risk Management**
    - Provides ICT supply chain product and service providers and acquirers with guidance on:
      - Gaining visibility into and managing the security risks caused by geographically dispersed ICT supply chains;
      - Integrating information security processes and practices into the system and software lifecycle processes, described in ISO/IEC 15288 and ISO/IEC 12207 while supporting information security controls, described in ISO/IEC 27002;
      - Responding to risks stemming from the global supply chain to IT products and services that may have an impact on the security of organizations using these products and services, including risks related to provided products and services in the ICT supply chain regardless of actors.
  - Part 4 – Guidelines for Security of Outsourcing .

## DoD Standards for System and Software Assurance

**DoD**

**DODD 8500.1, Information Assurance (IA)**

**DODI 8500.2, Information Assurance (IA) Implementation**

**DODI 8510.01, DOD IA Certification and Accreditation  
Process (DIACAP)**

**DODD 8570.01 IA Training, Certification, and Workforce  
Management**

**DISA Security Technical Implementation Guides (STIGS)**

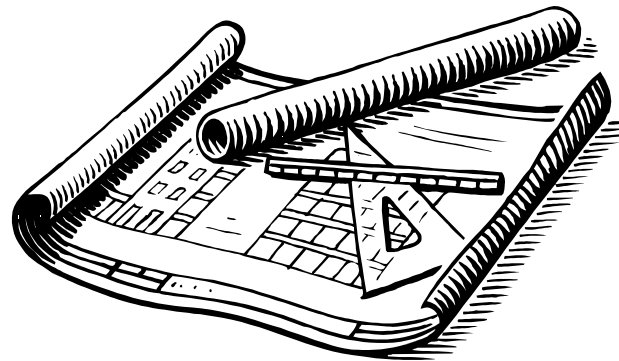
# Federal Information Security Management Act (FISMA) Implementation

NIST

- FIPS Publication 199, Standards for Security Categorization of Federal Information and Information System
- FIPS Publication 200, Minimum Security Requirements for Federal Information and Federal Information Systems
- NIST Special Publication 800-18 Revision 1, Guide for Developing Security Plans for Federal Information Systems and Organizations
- NIST Special Publication 800-30, Revision 1, Risk Management Guide for Information Technology Systems
- NIST Special Publication 800-37, Revision 1, Guide for Applying the Risk Management Framework to Federal Information Systems
- **NIST Special Publication 800-39, Enterprise-wide Risk Management**
- **NIST Special Publication 800-53 Revision 1, Recommended Security Controls for Federal Information Systems**
- NIST Special Publication 800-53A, Guide for Assessing the Security Controls in Federal Information Systems
- NIST Special Publication 800-59, Guide for Identifying an Information System as a National Security System
- NIST Special Publication 800-60, Guide for Mapping Types of Information and Information Systems to Security Categories

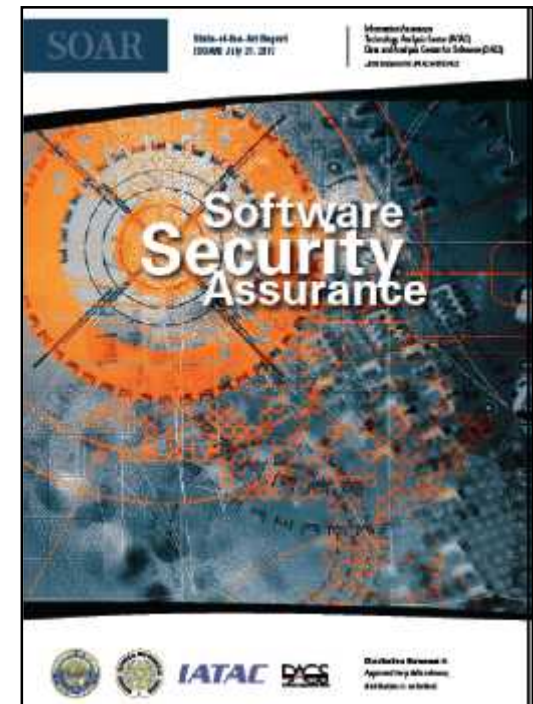
Source: <http://csrc.nist.gov/groups/SMA/fisma/overview.html>

## Additional Guidance for Assurance



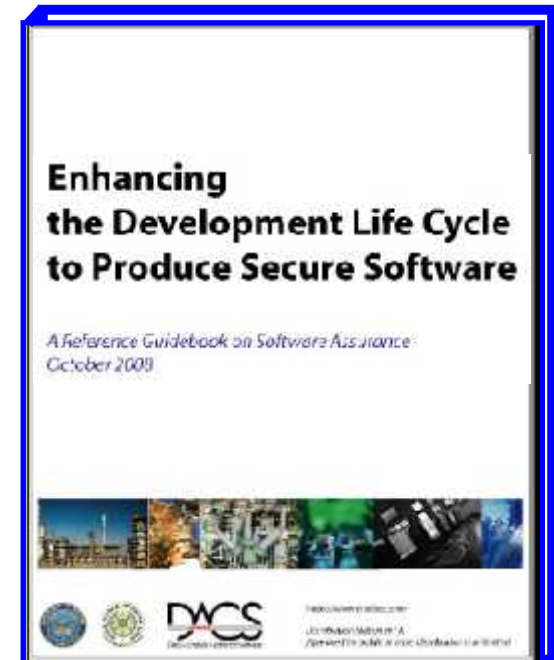
## State of the Art Report on Software Security Assurance

- An IATAC/DACS report identifying and describing the current state of the art in software security assurance, including trends in:
  - Techniques for the production of secure software
  - Technologies that exist or are emerging to address the software security challenge
  - Current activities and organizations in government, industry, and academia, in the U.S. and abroad, that are devoted to systematic improvement of software security
  - Research trends worldwide that might improve the state of the art for software security
- Available free via <http://iac.dtic.mil/iatac/download/security.pdf>



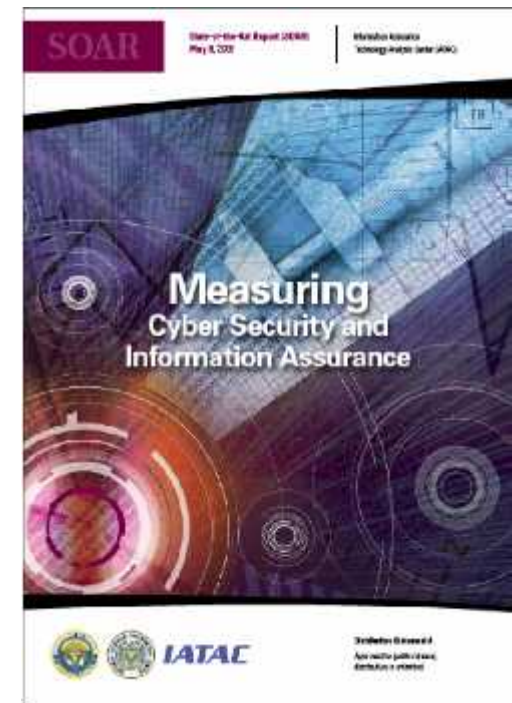
## Enhancing the Development Life Cycle to Produce Secure Software

- Describes how to integrate security principles and practices in software development life cycle
- Addresses security requirements, secure design principles, secure coding, risk-based software security testing, and secure sustainment
- Provides guidance for selecting secure development methodologies, practices, and technologies
- Available free via [https://www.thedacs.com/techs/enhanced\\_life\\_cycles/](https://www.thedacs.com/techs/enhanced_life_cycles/)



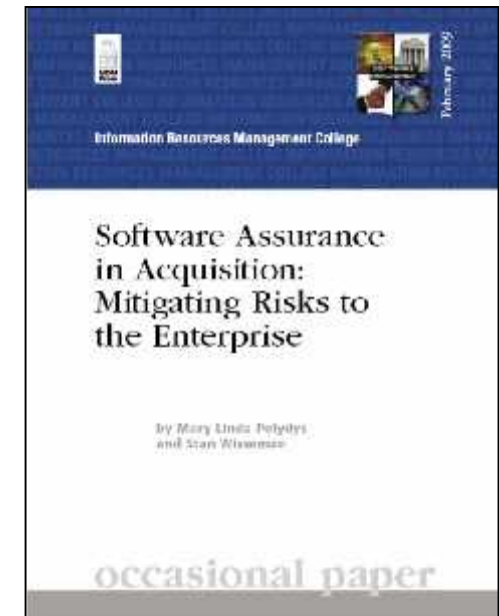
## Measuring Cyber Security and Information Assurance

- Provides a broad picture of the current state of cyber security and information assurance (CS/IA), as well as, a comprehensive look at the progress made in the CS/IA measurement discipline over the last nine years since IATAC published its IA Metrics Critical Review and Technology Assessment (CR/TA) Report in 2000
- Available free via
- <http://iac.dtic.mil/iatac/download/cybersecurity.pdf>



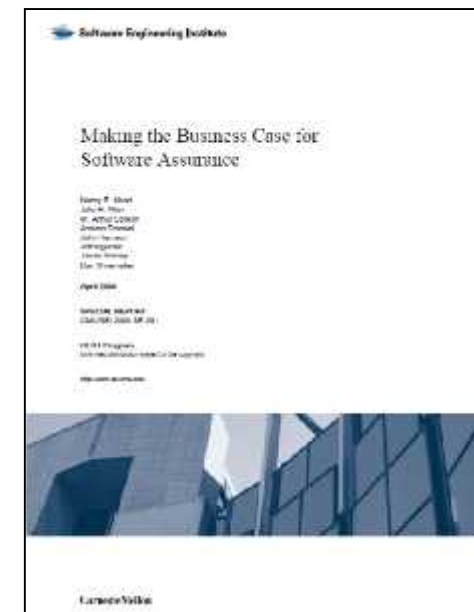
# Software Assurance in Acquisition: Mitigating Risks to the Enterprise

- Provides information on how to incorporate Software Assurance considerations in key decisions
  - How to exercise due diligence throughout the acquisition process relative to potential risk exposures that could be introduced by the supply chain
  - Includes practices that enhance SwA in the purchasing process
    - Due diligence questionnaires designed to support risk mitigation efforts by eliciting information about the software supply chain (these are also provided in [Word format](#) so they can be customized)
    - Sample contract provisions
    - Sample language to include in statements of work
- Pre-publication version available free via [https://buildsecurityin.us-cert.gov/swa/downloads/SwA\\_in\\_Acquisition\\_102208.pdf](https://buildsecurityin.us-cert.gov/swa/downloads/SwA_in_Acquisition_102208.pdf)
- Final version published by National Defense University Press, Feb 2009



## Making the Business Case for Software Assurance

- Provides background, context and examples for making the business case for software assurance:
  - Motivators
  - Cost/Benefit Models Overview
  - Measurement
  - Risk
  - Prioritization
  - Process Improvement & Secure Software
  - Globalization
  - Organizational Development
  - Case Studies and Examples
- Available free via
- <http://www.sei.cmu.edu/library/abstracts/reports/09sr001.cfm>



## Software Assurance: A Curriculum Guide to the Common Body of Knowledge to Produce, Acquire, and Sustain Secure Software

- Provides a framework intended to identify workforce needs for competencies, leverage sound practices, and guide curriculum development for education and training relevant to software assurance
- Available free via

<https://buildsecurityin.us-cert.gov/bsi/940-BSI/version/default/part/AttachmentData/data/CurriculumGuideToTheCBK.pdf>



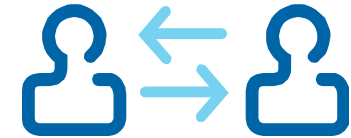
Software Assurance: A Curriculum Guide to the Common Body of Knowledge to Produce, Acquire and Sustain Secure Software

Software Assurance Workforce Education and Training Working Group

October 2007

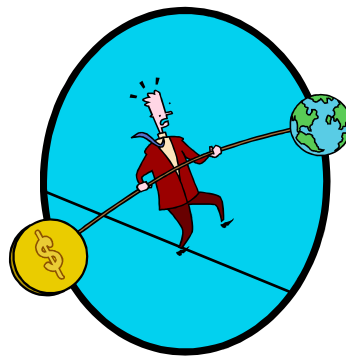


## Useful Links



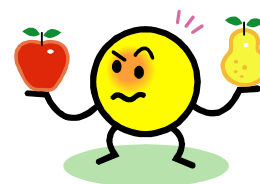
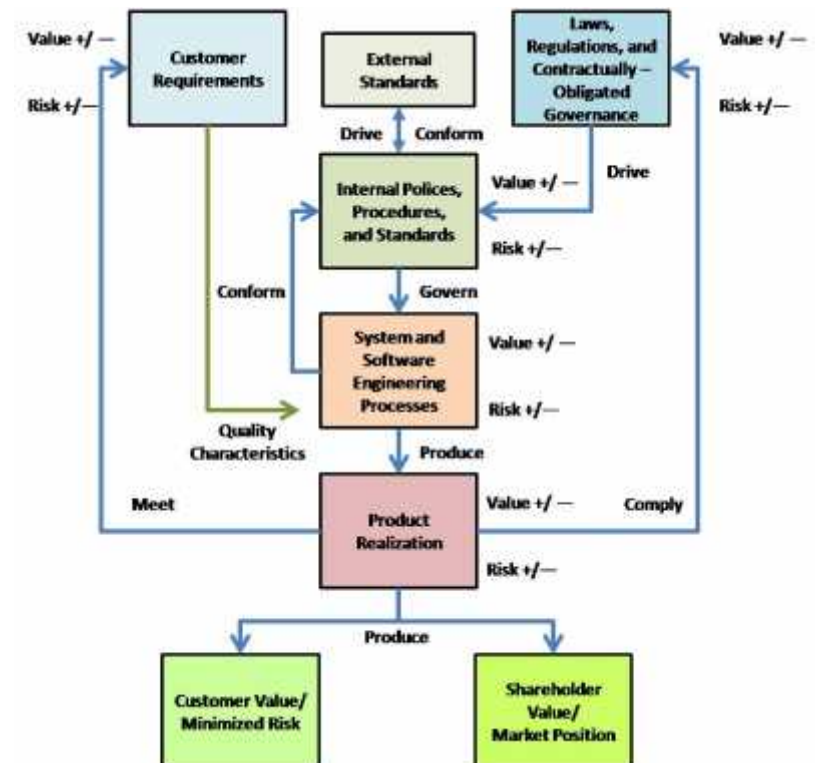
- **DHS Build Security In Web Site**
  - A wealth of software and information assurance information, including white papers on static code analysis tools
  - More information on Build Security In can be found at: <https://buildsecurityin.us-cert.gov/daisy/bsi/home.html>
- **Common Weaknesses Enumeration (CWE)**
  - This site provides a formal list of software weakness types created to Serve as a common language for describing software security weaknesses in architecture, design, or code
  - More information on CWEs can be found at:
    - <http://cwe.mitre.org/>
- **CWE/SANS Top 25 Most Dangerous Software Errors**
  - The 2010 CWE/SANS Top 25 Most Dangerous Software Errors is a list of the most widespread and critical programming errors that can lead to serious software vulnerabilities.
  - More information on the CWE/SANS Top 25 can be found at:
    - [http://cwe.mitre.org/top25/archive/2010/2010\\_cwe\\_sans\\_top25.pdf](http://cwe.mitre.org/top25/archive/2010/2010_cwe_sans_top25.pdf)
- **NIST SAMATE Static Analysis Tool Survey**
  - The National Institutes for Science and Technology (NIST), Software Assurance Metrics and Tool Evaluation (SAMATE) project, provides tables describing current static code analysis tools for source, byte, and binary code analysis
  - More information on SAMATE can be found at: <http://samate.nist.gov/>

# Rationalizing Governance, Engineering Practice, and Engineering Economics



## Key Questions in Implementing Processes and Standards to Deliver Value

- How does compliance with a particular external governance requirement impact organizational risk and value delivery?
- Where multiple external compliance requirements exist, have I examined their overlaps and chosen a compliance strategy that optimizes compliance while minimizing risk and maximizing value?
- Have I added value and reduced risk to my engineering processes through the policies, procedures, and standards I've adopted in compliance with those external governance requirements?
- Does my product provide value in the market place while limiting risk to acquirers and users?



## For More Information

**Paul R. Croll**

**Fellow**

**CSC**

**10721 Combs Drive**

**King George, VA 22485-5824**

**Phone: +1 540.644.6224**

**Fax: +1 540.663.0276**

**e-mail: [pcroll@csc.com](mailto:pcroll@csc.com)**

