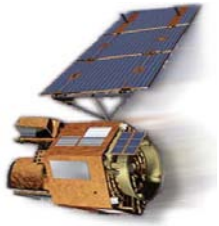
CSC
NORTH AMERICAN PUBLIC SECTOR

Rationalizing Governance, Engineering Practice, and Engineering Economics in the System and Software Assurance Trade Space

Paul R. Croll
Fellow
CSC
pcroll@csc.com






Outline

- The Governance Challenge
- Governance for System and Software Assurance
- Quality Characteristics in Engineering Trades
- System and Software Assurance Engineering Economics and Risk Management
- Rationalizing Governance, Engineering Practice, and Engineering Economics

CSC

NORTH AMERICAN
PUBLIC SECTOR

The Governance Challenge

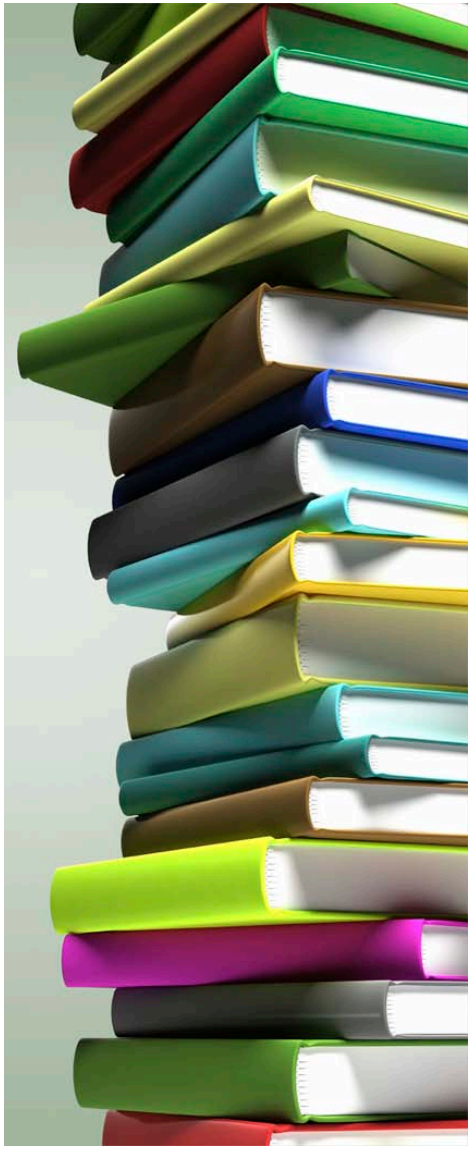
Trade Space Constraints

- Engineering systems and systems of systems (SoSs) is about trade-offs
- Generally such trade-offs focus on quality attributes associated with architecture, design, and implementation [1]
- From a system and software assurance perspective, the trade space is often constrained by a myriad of governance documents that may include public law, regulatory agency directives, both acquiring and supplying organizations' policies and procedures, as well as standards and best practices.
- These numerous documents may in some cases be duplicative in their reporting requirements, or may even conflict with each other. They not only constrain trade-offs, but directly impact system cost.

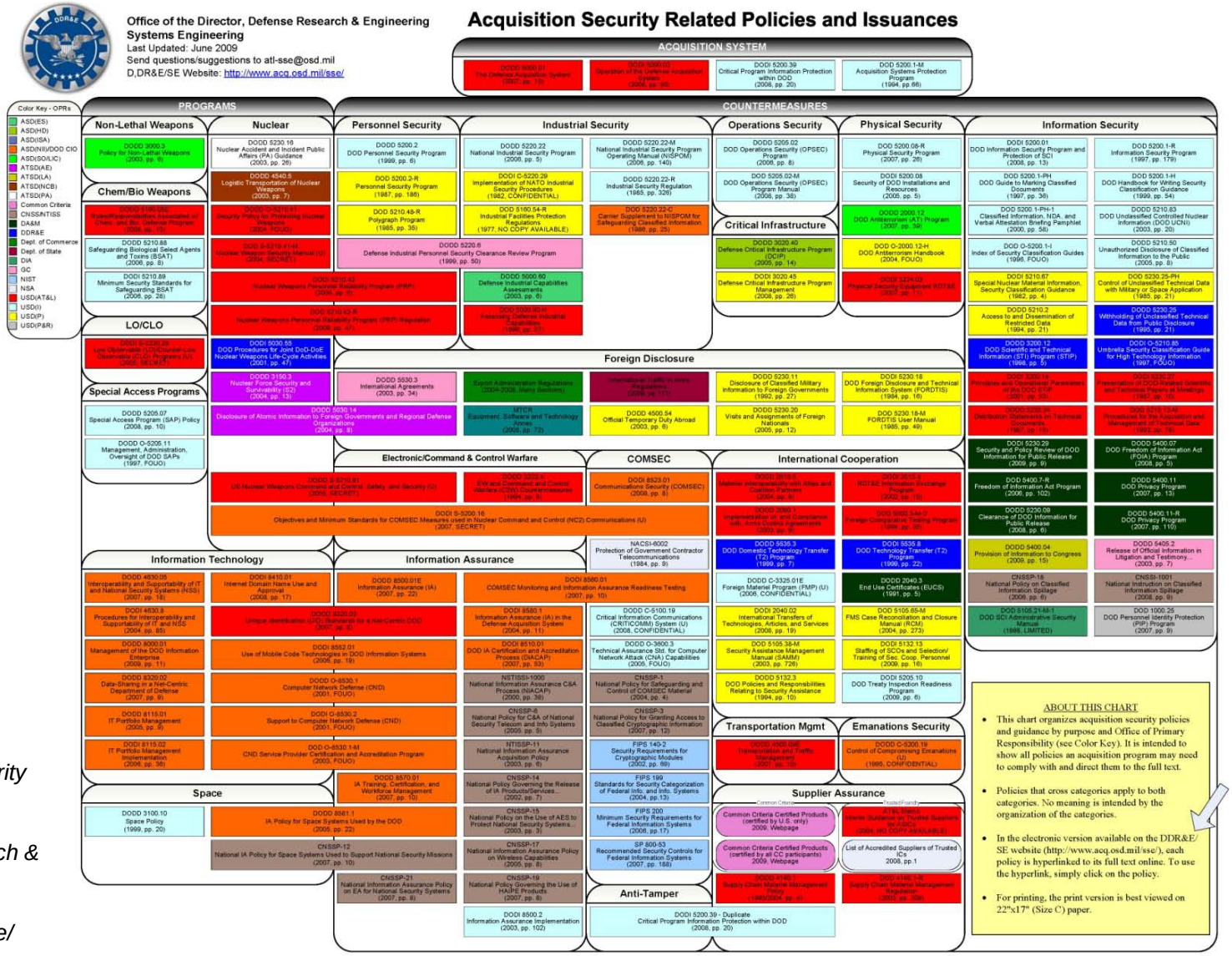
.

Governance Context

- 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 [2] 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



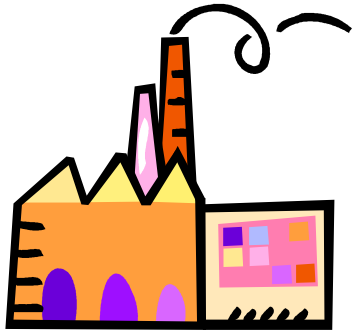
Example: US DoD Compilation of Security-Related Policy and Guidance



Source: Acquisition Security Related Policies and Issuances, Office of the Director, Defense Research & Engineering Systems Engineering, http://www.acq.osd.mil/sse/

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 [3]
  - 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

CSC

NORTH AMERICAN
PUBLIC SECTOR

Governance for System and Software Assurance

IT Governance Defined

- The Information Security and Control Association (ISACA) defines governance as:
  - *The set of responsibilities and practices exercised by the board and executive management with the goal of providing strategic direction, ensuring that objectives are achieved, ascertaining that risks are managed appropriately and verifying that the enterprise's resources are used responsibly* [4]
- Brotby [5] suggest five basic outcomes of information security governance:
  - Strategic alignment of information security with business strategy
  - Security risk management
  - Resource management related to the effective deployment of security knowledge and infrastructure
  - Performance measurement through information security governance metrics
  - Value delivery through optimization of information security investments

System and Software Assurance Defined

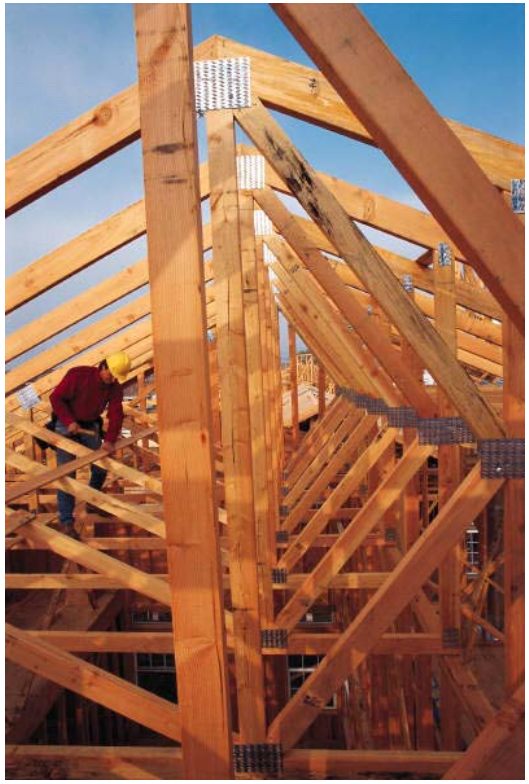
- These suggested outcomes may be generalized to engineering for system and software assurance.
- System assurance is defined as *the justified confidence that the system functions as intended and is free of exploitable vulnerabilities, either intentionally or unintentionally designed or inserted as part of the system at any time during the life cycle* [6].
- Similarly, software assurance is defined as *the level of confidence that software is free from vulnerabilities, either intentionally designed into the software or accidentally inserted at anytime during its lifecycle, and that the software functions in the intended manner* [7].

IT Governance Generalized to System and Software Assurance

- ***Strategic alignment with business strategy*** implies that
  - Organizational policies and enforcement mechanisms are in place to support effective engineering for system and software assurance
  - Costs and benefits are well understood
- ***Security risk management*** implies that
  - Security-related risks are identified and managed at all stages of the engineering life cycle
  - Such risks are communicated appropriately to stakeholders.
- ***Resource management*** implies that
  - Qualified, properly trained engineers are available throughout the engineering life cycle to adequately address systems and software assurance concerns
- ***Performance measurement*** implies,
  - For engineering processes, that they are meeting their targets for architecture and design resiliency, secure coding practices, and freedom from vulnerabilities
  - For the products produced, that the product meets its quality requirements
- ***Value delivery*** implies that
  - The value proposition associated with both the investment made in the engineering processes for system and software assurance, and the costs associated with architectural and design decisions, show clear benefit to both the organization, the acquirer and the users of the system

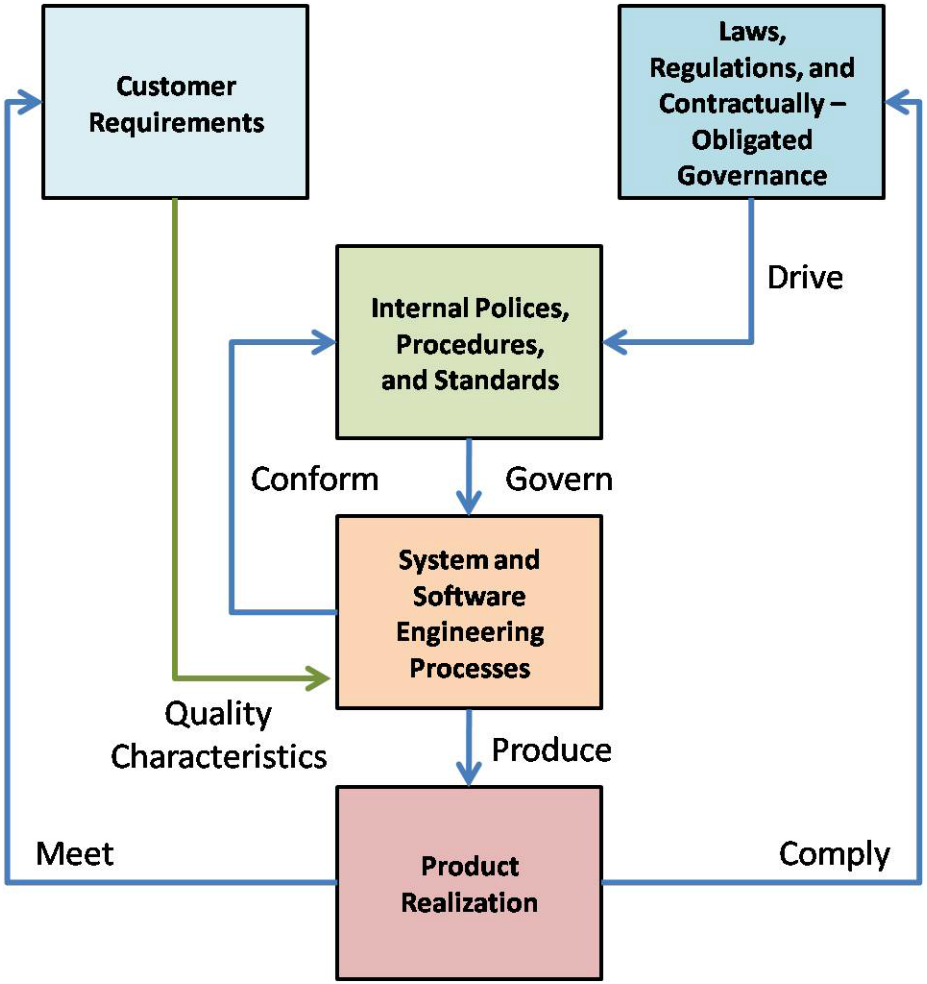
Security Governance Framework

- Brotby [5] further suggests that an effective security governance framework generally consists of:
  - A security risk management methodology
  - A security strategy linked with business objectives
  - An effective security organizational structure
  - Security policies that address control and regulation in the context of the security strategy
  - Security standards defining compliance with policy
  - A process for monitoring of compliance and for providing feedback on risk mitigation
  - A process for periodic evaluation and update of the governance framework in the context of changing risks and organizational objectives.



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 this trade space.
- **Internal policies, procedures, and standards** institutionalize external governance requirements (as well as business best practices) and drive the engineering processes for producing systems and software
  - Internal quality reviews will generally include reviews of conformance of a project's engineering processes to these established policies, procedures, and standards.
- **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.



CSC

NORTH AMERICAN
PUBLIC SECTOR

Quality Characteristics in Engineering Trades

Customer Defined Quality Requirements

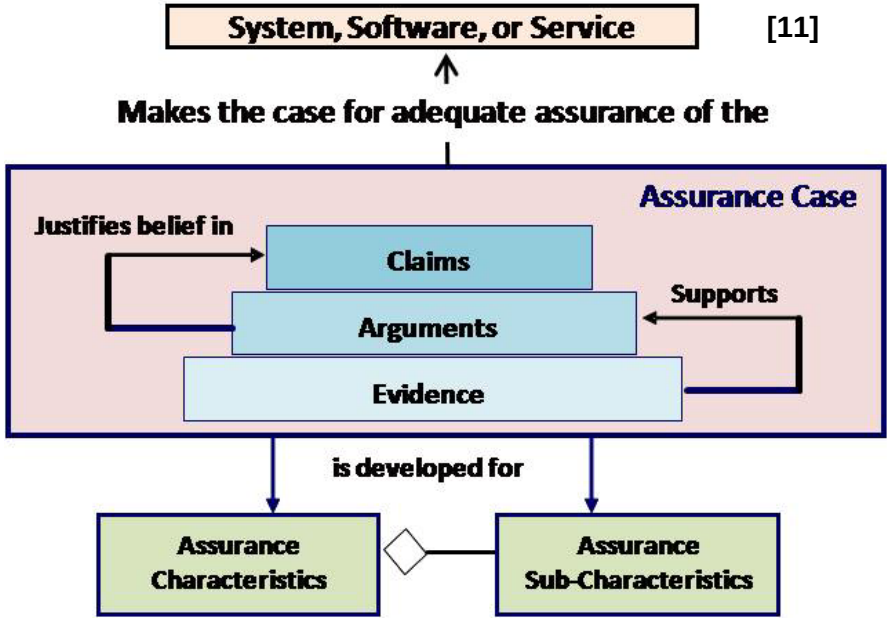
- Quality characteristics drive system and software architecture and design.
- Firesmith [8] describes two classes of quality characteristics important to the system and software assurance trade space:
  - Internal quality characteristics
  - External quality characteristics
- Internal characteristics are described as characterizing an internally visible quality of a system or architectural component when it is in the process of being developed, modified, or retired
  - Internal quality characteristics are of primary interest to developers and maintainers
- External characteristics are described as characterizing an externally visible quality of a system or architectural component when it is deployed and in service in its operational environment
  - External quality characteristics are of primary interest to users and operators
- Quality requirements in the trade space are specified in terms of these quality characteristics
- *Compliance with governance documents*, as is *security*, is only one of many quality characteristics which must be addressed when making architectural and design trades

A Quality Attribute Approach to System and Software Engineering Trades

- This approach [1] ensures that:
  - Customer quality requirements will have been distilled into drivers which will have shaped the system architecture and design.
  - Tradeoffs will have been made to optimize the realization of important quality characteristics, in concert with customer expectations.
  - The level of confidence that the resultant system will meet those expectations will be known.
  - Customers will be knowledgeable of any residual risk they are accepting by accepting the delivered system.
- The NDIA guidebook on Engineering for System Assurance [6], suggests using system assurance requirements, design constraints and system assurance critical scenarios for trade-off analysis, and documenting the results in an assurance case

The Assurance Case

- A detailed explanation of the recommended structure of an assurance case may be found in [9], and a discussion of its contents in [6] and [10].
- Claims made about a system's assurance characteristics must be supported by rational arguments to justify their belief
- In order for these arguments to be accepted, they must in turn be supported by sufficient evidence
- The assurance case is the means for communicating to stakeholders the degree of assurance achieved, with what confidence level, and with what residual risk



CSC

NORTH AMERICAN
PUBLIC SECTOR

System and Software Assurance Engineering Economics and Risk Management

Software Assurance Economic Models

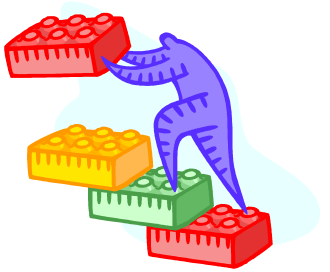
- Bailey et al [12] discuss four categories of IT valuation models, and thirteen specific models, that might be adapted assessing the cost and value of software assurance.
- Those most familiar in engineering and IT environments include:
  - ***Investment-Oriented Models*** like Microsoft's Rapid Economic Justification framework [13];
  - ***Cost-Oriented Models***, like Total Cost of Ownership [14];
  - ***Environmental/Contextual-Oriented Models***, like Balanced Scorecard [15]; and
  - ***Quantitative Estimation Models***, like CoCoMo II with Security Extensions [16]
- However, there is no one widely accepted model for determining the cost/benefit of investment in software assurance [17] [18]

Assurance Risk Management – Balancing Assurance Costs

- NIST Special Publication 800-30, *Risk Management Guide for Information Technology Systems* [19] 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* [20], 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

- MacKessy [21] posits that a "hierarchy of risk" framework, providing a flexible, multidimensional schema for analyzing quantitative and qualitative risk may be useful in address related classes of risk in the business enterprise



Risk Hierarchy for System and Software Assurance

- 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

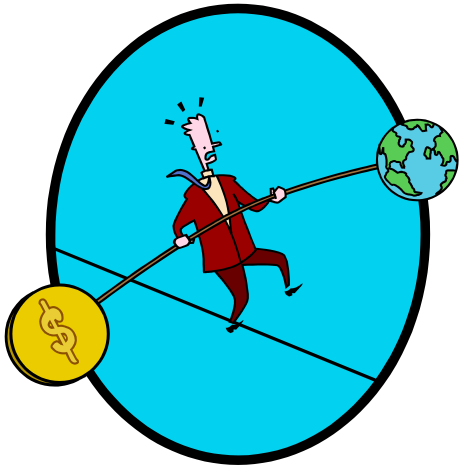
Value Delivery

- *Value delivery* implies that the value proposition associated with both the investment made in engineering processes for system and software assurance, and the costs associated with architectural and design decisions, shows clear benefit to both the organization and the acquirer and users of the system
- Value addresses the relationship between stakeholder needs and the resources used to satisfy them. Stakeholders will have different perceptions about what constitutes value
  - Value in the eyes of a regulatory agency may be viewed as compliance with directives
  - The organization's CEO and its shareholders may view value in terms of profit and market position
  - Acquirers or users of a system may perceive value in terms of expected performance and freedom from exploitable vulnerabilities
- The challenge is to understand and reconcile these differences without any negative impact on quality requirements.

CSC

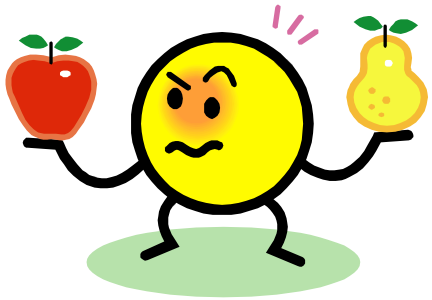
NORTH AMERICAN
PUBLIC SECTOR

Rationalizing Governance, Engineering Practice, and Engineering Economics



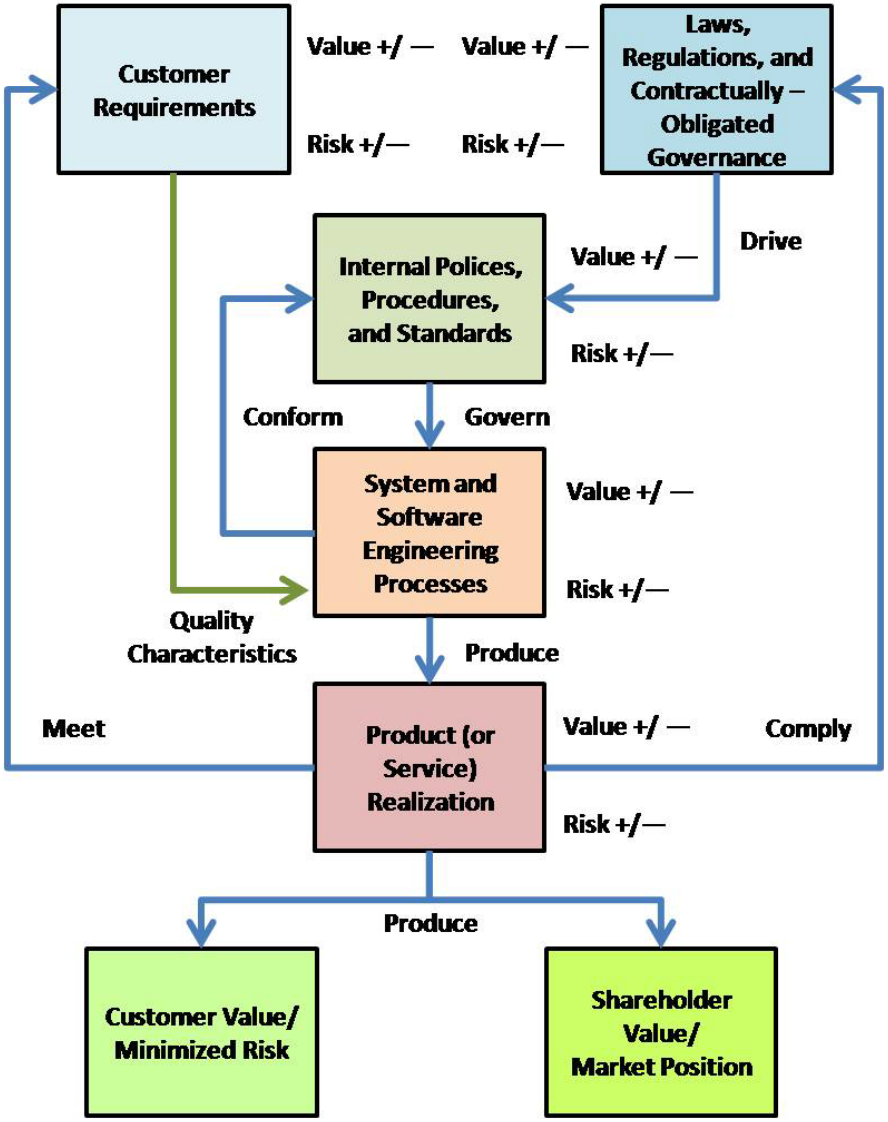
Rationalizing Governance, Engineering Practice, and Engineering Economics

- The previous discussion has touched on governance in the engineering life cycle, quality characteristics and their use in making engineering trades, models for assessing the cost and value of software assurance, assurance risk management, and value delivery
- The literature surveyed is abundant with models, equations, and checklists but comes to no consensus on the “best” approach for system and software assurance
- Several of the references cited provide more information for comparison of governance requirements as well as methods, tools, and techniques [5], [6], [8], [12], [18], and [22]
- The figure on slide 14, described earlier as depicting governance in the engineering life cycle, may be annotated to illustrate the touch points for rationalizing governance with risk and value



Rationalizing Governance in the Engineering Life Cycle

- Although simplistic, this figure depicts the necessary consideration of positive and negative impacts on value and risk throughout the engineering life cycle and the ultimate delivery of value to both the customer and the organization's shareholders
- Understanding of this value chain should be an integral part of an organization's approach to engineering projects
- This includes the impacts of external laws, regulations or other contractually-obligated governance requirements on internal policies procedures and standards that in turn govern the organization's engineering processes, as well as the impacts of those processes on the value chain.



Key Rationalization Questions

- 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?

Further research is needed to produce both qualitative and quantitative tools to facilitate such rationalization

References – 1 of 2

[1] P. Croll. Quality Attributes – Architecting Systems To Meet Customer Expectations, Proceedings of the 2nd Annual IEEE International Systems Conference. New York: Institute of Electrical and Electronics Engineers, April 2008.

[2] U. S. Congressional Budget Office. Congressional Budget Office Cost Estimate – S.3474, FISMA Act of 2008. Washington, DC, October 27, 2008.

[3] Tashi. Regulatory Compliance and Information Security Assurance, Proceedings of the 2009 International Conference on Availability, Reliability and Security. New York: Institute of Electrical and Electronics Engineers, 2009

[4] ISACA. CISM Review Manual, Information Security and Control Association, 2008.

[5] K. Brotby. Information Security Governance, A Guide for Boards of Directors and Executive Management, 2nd ed. ITGI, 2005.

[6] National Defense Industrial Association (NDIA) System Assurance Committee. Engineering for System Assurance. Arlington, VA: NDIA, 2008.

]7] Committee on National Security Standards. CNSS Instruction No. 4009, National Information Assurance Glossary, Ft. Meade, MD, Revised 2006.

[8] D. Firesmith, P. Capell, D. Falkenthal, C. Hammons, D. Latimer, and T. Merendino. The Method-Framework for Engineering System Architectures (MFESA): Generating Effective and Efficient Project-Specific System Architecture Engineering Methods. Boca Raton, FL: Auerbach Publications, 2009.

[9] ISO/IEC JTC1/SC7, CD 15026-2.4, Systems and software engineering — Systems and software assurance — Part 2: Assurance case. ISO, Geneva Switzerland, July 2009.

[10] J. Goodenough, H. Lipson, and C Weinstock. Arguing *Security - Creating Security Assurance Cases*. Pittsburgh, PA: Carnegie Mellon University, 2008. https://buildsecurityin.us-cert.gov/bsi/articles/knowledge/assurance/643-BSI.html#

[11] P. Croll. *Practices Make Perfect – Leveraging Your Engineering and Management Practices to Meet the Software Assurance Challenge*, 20th Annual Department of Defense Systems and Software Technology Conference, 2008.

References – 2 of 2

[12] J. Bailey, A. Drommi, J. Ingalsbe, N. Mead, and D. Shoemaker. Models for Assessing the Cost and Value of Software Assurance. Pittsburgh, PA: Carnegie Mellon University, 2008. https://buildsecurityin.us-cert.gov/daisy/bsi/articles/knowledge/business/684-BSI.html

[13] Microsoft. Rapid Economic Justification, Enterprise Edition: A Step-by-Step Guide to Optimizing IT Investments that Forge Alliances Between IT and Business. Redmond, Washington: Microsoft Corporation, 2005

[14] R. West, and L Daigle. Total Cost of Ownership: A Strategic Tool for ERP Planning and Implementation. Boulder, Colorado: EDUCAUSE Center for Applied Research, Research Bulletin, Volume 4, Issue 1, 2004.

[15] W. Van Grembergen and S. De Haes. Measuring and Improving IT Governance Through the Balanced Scorecard, Information Systems Control Journal, Volume 2. Information Systems Audit and Control Association Inc., 2005.

[16] E. Colbert and B. Boehm. Cost Estimation for Secure Software and Systems, USC-CSSE-2008-811. Los Angeles, California: Center for Systems and Software Engineering, University of Southern California, 2008. http://csse.usc.edu/csse/TECHRPTS/2008/usc-csse-2008-811/usc-csse-2008-811.pdf

[17] J. Allen. Making the Business Case for Software Assurance (SWA). SEPG Conference, 2009.

[18] A. Arora, et al. Estimating Benefits from Investing in Secure Software Development. Pittsburgh, PA: Carnegie Mellon University, 2008. https://buildsecurityin.us-cert.gov/daisy/bsi/articles/knowledge/business/267-BSI.html

[19] G. Stoneburner, A. Goguen, and A. Feringa. NIST Special Publication 800-30, Risk Management Guide for Information Technology Systems. Gaithersburg, MD: National Institute of Standards and Technology, 2002.

[20] G. Stoneburner, C. Hayden, and A. Feringa. NIST Special Publication 800-27, Engineering Principles for Information Technology Security (A Baseline for Achieving Security), Revision A. Gaithersburg, MD: National Institute of Standards and Technology, 2004.

[21] J. MacKessy. The Hierarchy of Risk A New Approach To Risk Management, The Investment Professional, Vol 2, No. 3. New York: The New York Society of Security Analysts, Inc., 2009.

[22] R. Venkataraman and J. Pinto. Cost and Value Management in Projects. Hoboken, New Jersey: John Wiley & Sons, Inc., 2008.

For More Information . . .

Paul R. Croll
CSC
10721 Combs Drive
King George, VA 22485-5824

Phone: +1 540.644.6224

Fax: +1 540.663.0276

e-mail: pcroll@csc.com