CISQ

Consortium for IT Software Quality™

Automating Software Quality Measurement with Standards

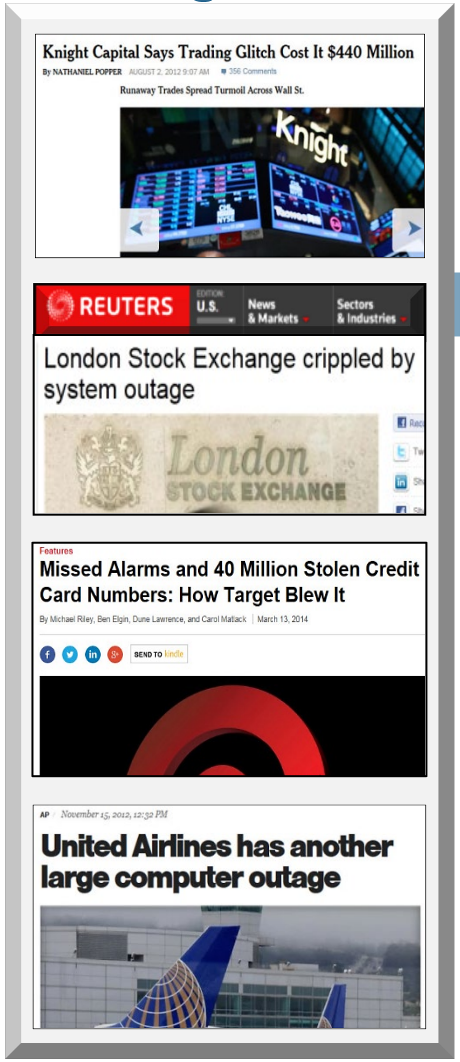
Dr. Bill Curtis

Founding Executive Director, CISQ

CISQ Webinar — May 15, 2019

The Era of Nine-Digit Defects

Nine Digit Defects



now affect

- Board of Directors
- CEO, COO, CFO
- Business VPs
- Corporate Auditors
- CIO

accountable for

- Governance
- Risk management
- Business Continuity
- Brand protection
- Customer experience

Evaluate Application Risk with CISQ Measures

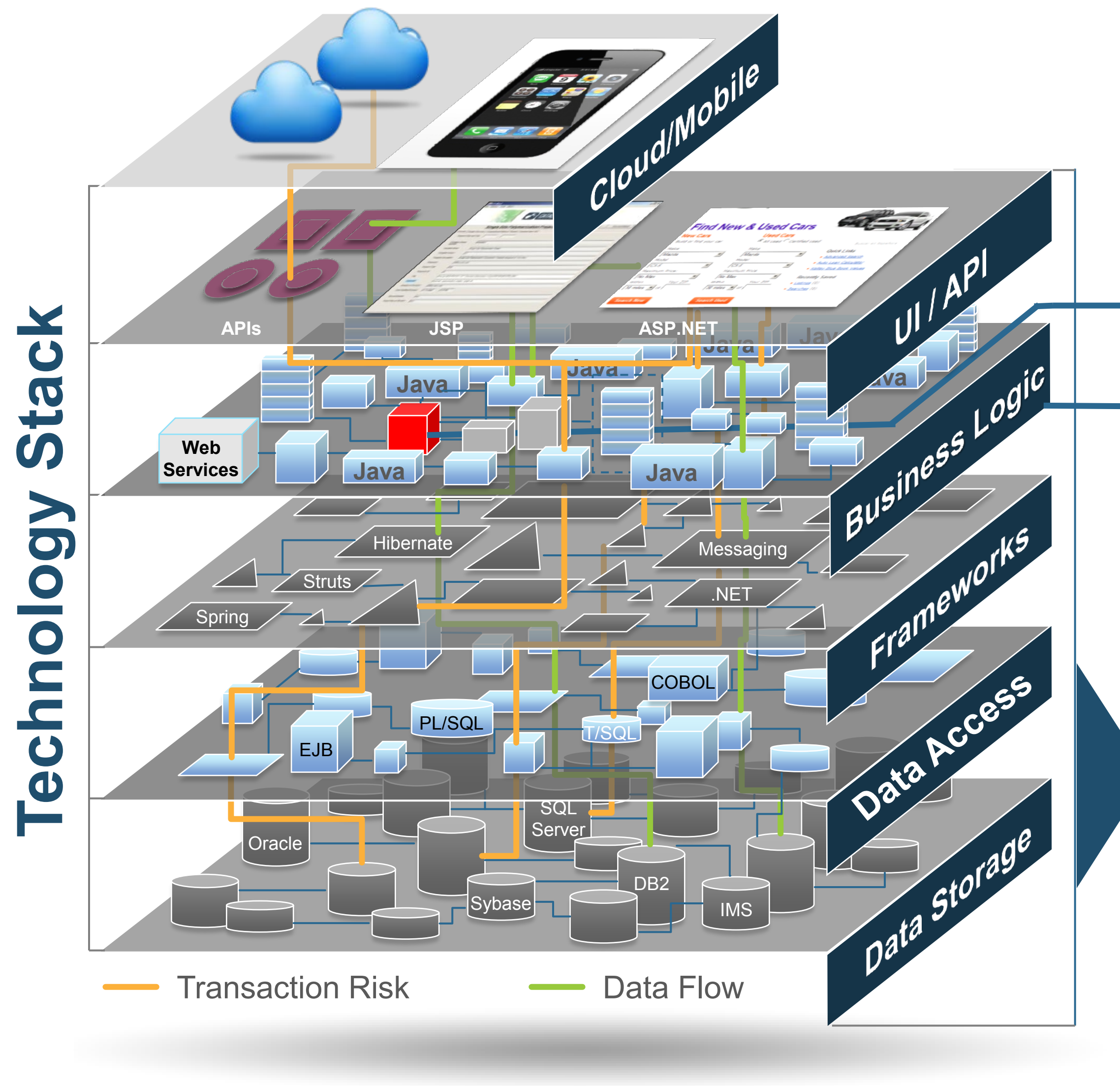
CISQ
Consortium for IT Software Quality

Why Do Software-Intensive System Projects Fail ?

Failure causes		Failure avoidance practices
Unrealistic expectations	→	Feasibility analysis
Incomplete requirements	→	Mission & business analysis
Weak contracts & control	→	**Acquisition management**
Unachievable schedules, budgets	→	Project & risk management
Changing requirements	→	Baseline management
Unsound architecture & coding	→	**Technical risk measurement**
Staff inexperience & turnover	→	Contractual personnel clauses
Truncated testing	→	**Quality management**
Botched deployments	→	**Acceptance management**

Software measurement

Modern Apps are a Technology Stack



1 Unit Level

- Code style & layout
- Expression complexity
- Code documentation
- Class or program design
- Basic coding standards
- Developer level

2 Technology Level

- Single language/technology layer
- Intra-technology architecture
- Intra-layer dependencies
- Inter-program invocation
- Security vulnerabilities
- Development team level

3 System Level

- Integration quality
- Architectural compliance
- Risk propagation
- Application security
- Resiliency checks
- Transaction integrity
- Function point
- Effort estimation
- Data access control
- SDK versioning
- Calibration across technologies
- IT organization level

What Is the Consortium for IT Software Quality ?

Carnegie Mellon Software Engineering Institute

OMG — OBJECT MANAGEMENT GROUP®

Co-founders

Dr. Paul Nielsen, CEO → **CISQ** ← Dr. Richard Soley, CEO

Develop specifications for automatable measures of software systems

↓

Gain approval as OMG standards

↓

Fast-track to ISO

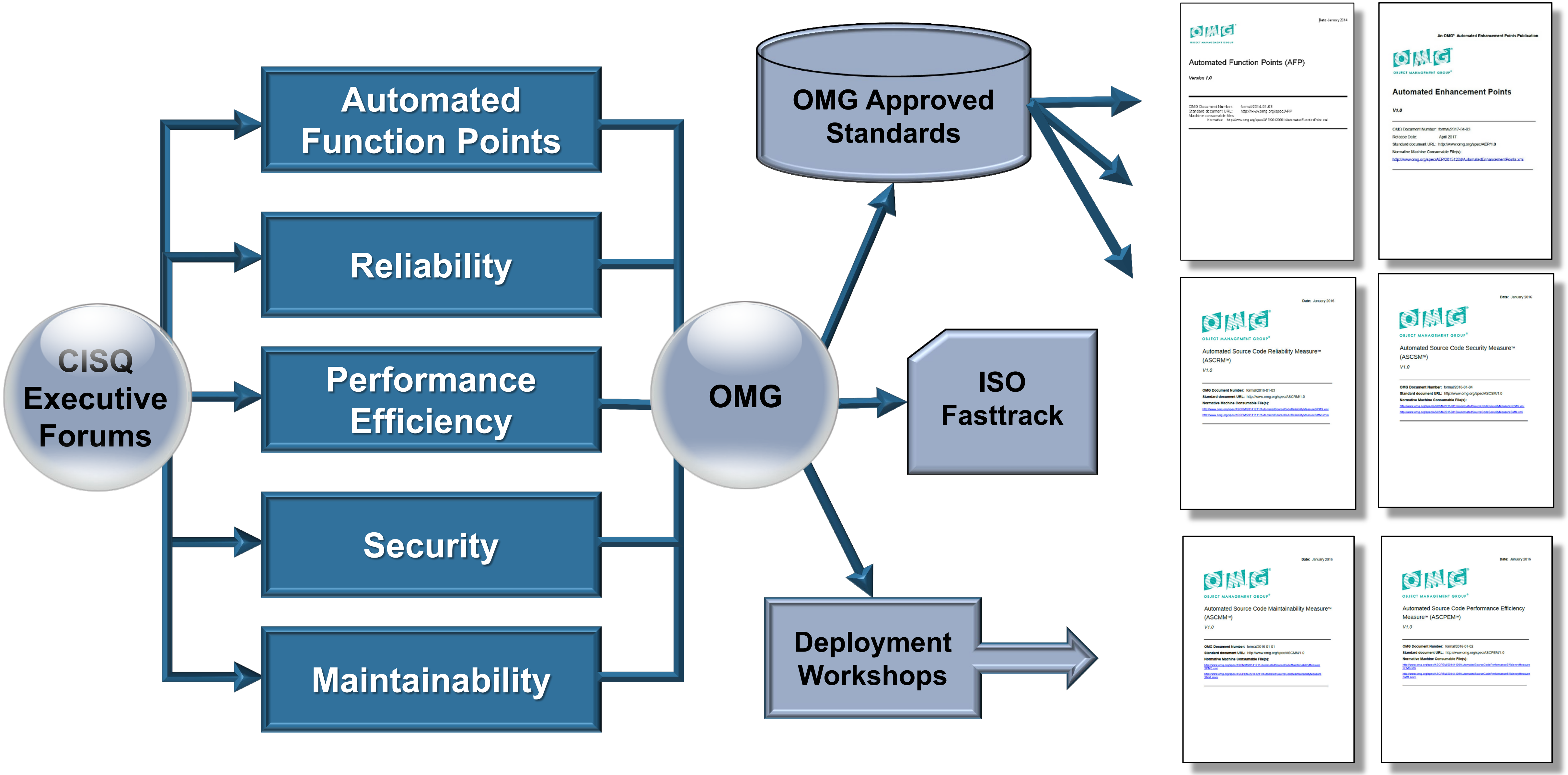
CISQ Sponsors

- Cognizant
- Synopsys
- Tech Mahindra
- SHPI
- CAST
- CGI
- USC CSSE
- Northrop Grumman

CISQ Partners

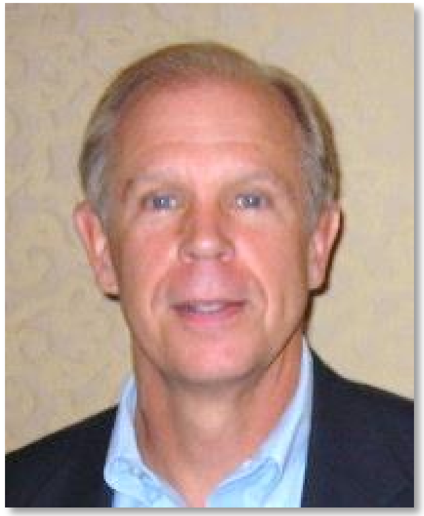
- Gartner
- QuEST Forum
- Forrester
- IAOP
- ITAAC
- TechWell
- AFCEA Washington, DC Chapter
- Making Software Better

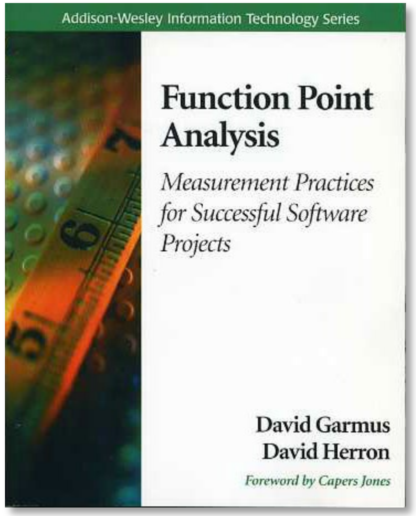
CISQ/OMG Standards Process

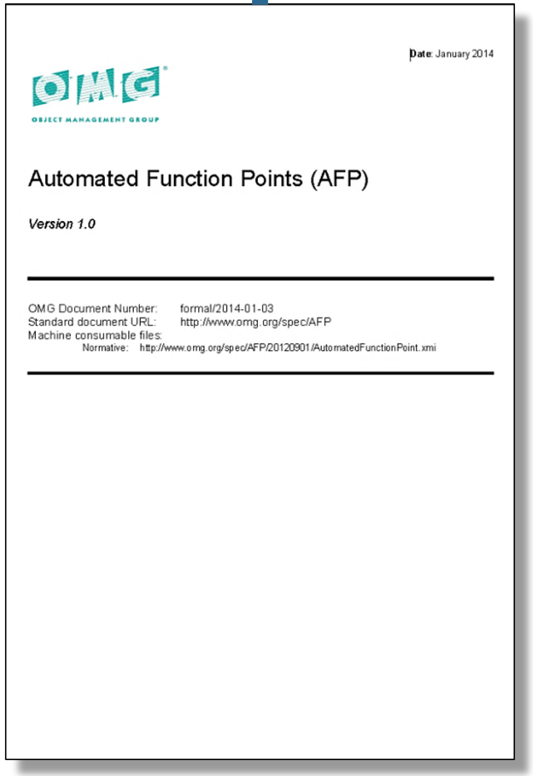


Automated Size Measurement

- **Mirrors IFPUG counting guidelines, but automatable**
- **Specification developed by international team led by David Herron of David Consulting Group**
- **Submitted thru OMG's fasttrack as ISO 19515**





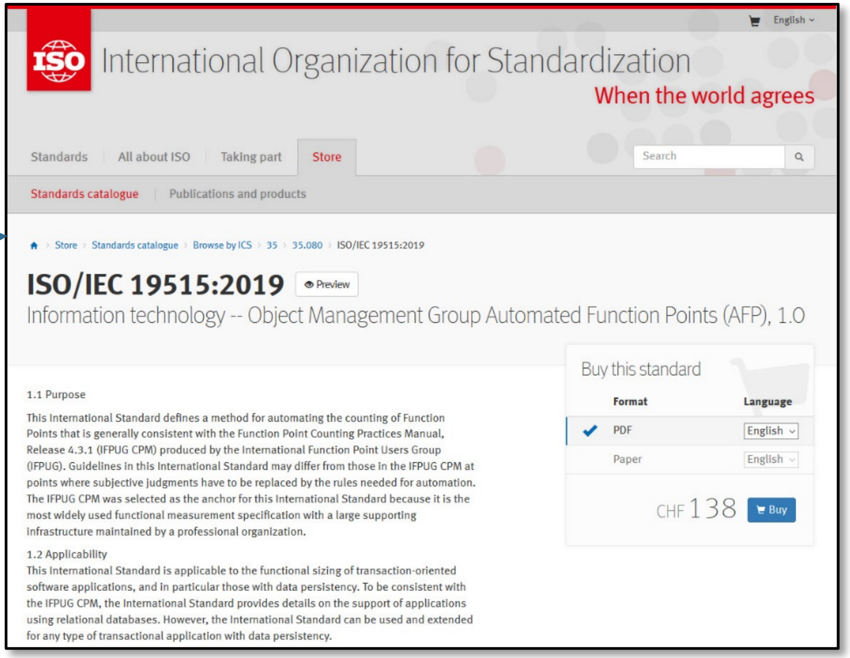

Date: January 2014

OMG
OBJECT MANAGEMENT GROUP

Automated Function Points (AFP)

Version 1.0

OMG Document Number: formal/2014-01-03
Standard document URL: http://www.omg.org/spec/AFP
Machine consumable files:
Normative: http://www.omg.org/spec/AFP/20120901/AutomatedFunctionPoint.xmi


English

ISO International Organization for Standardization
When the world agrees

Standards | All about ISO | Taking part | Store | Search

Standards catalogue | Publications and products

Store > Standards catalogue > Browse by ICS > 35 > 35.080 > ISO/IEC 19515:2019

ISO/IEC 19515:2019 Preview

Information technology -- Object Management Group Automated Function Points (AFP), 1.0

1.1 Purpose

This International Standard defines a method for automating the counting of Function Points that is generally consistent with the Function Point Counting Practices Manual, Release 4.3.1 (IFPUG CPM) produced by the International Function Point Users Group (IFPUG). Guidelines in this International Standard may differ from those in the IFPUG CPM at points where subjective judgments have to be replaced by the rules needed for automation. The IFPUG CPM was selected as the anchor for this International Standard because it is the most widely used functional measurement specification with a large supporting infrastructure maintained by a professional organization.

1.2 Applicability

This International Standard is applicable to the functional sizing of transaction-oriented software applications, and in particular those with data persistency. To be consistent with the IFPUG CPM, the International Standard provides details on the support of applications using relational databases. However, the International Standard can be used and extended for any type of transactional application with data persistency.

Buy this standard

Format	Language
PDF	English
Paper	English

CHF 138 Buy

Automated Function Points is now an ISO/IEC standard

CISQ Structural Quality Measures

Consortium for IT Software Quality

CISQ Structural Quality Measures

CISQ Structural Quality Measures		Example architectural and coding weaknesses included in the CISQ measures
Security	36 (37) weaknesses	• SQL injection • Cross-site scripting • Buffer overflow
Reliability	36 (38) weaknesses	• Poor exceptional handling • Deadlock • Improper synchronization
Performance Efficiency	16 (3) weaknesses	• Expensive loop operation • Un-indexed data access • Unreleased memory
Maintainability	30 (1) weaknesses	• Excessive coupling • Dead code • Hard-coded literals

An international team of experts selected the weaknesses to include in CISQ measures based on the severity of their impact on operational problems or cost of ownership.

Only weaknesses considered severe enough they must be remediated were included in the CISQ measures.

CISQ Structural Quality measures have been extended to embedded systems software

CISQ Measures Updated for Embedded Systems

CISQ
Consortium for IT Software Quality

CWE #	Descriptor	Weakness description
CWE-22	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')	The software uses external input to construct a pathname that is intended to identify a file or directory that is located underneath a restricted parent directory, but the software does not properly neutralize special elements within the pathname that can cause the pathname to resolve to a location that is outside of the restricted directory.
CWE-23	Relative Path Traversal	The software uses external input to construct a pathname that should be within a restricted directory, but it does not properly neutralize sequences such as ".." that can resolve to a location that is outside of that directory.
CWE-36	Absolute Path Traversal	The software uses external input to construct a pathname that should be within a restricted directory, but it does not properly neutralize absolute path sequences such as "/abs/path" that can resolve to a location that is outside of that directory.
CWE-77	Improper Neutralization of Special Elements used in a Command ('Command Injection')	The software constructs all or part of a command using externally-influenced input from an upstream component, but it does not neutralize or incorrectly neutralizes special elements that could modify the intended command when it is sent to a downstream component.
CWE-78	Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')	The software constructs all or part of an OS command using externally-influenced input from an upstream component, but it does not neutralize or incorrectly neutralizes special elements that could modify the intended OS command when it is sent to a downstream component.
CWE-88	Argument Injection or Modification	The software does not sufficiently delimit the arguments being passed to a component in another control sphere, allowing alternate arguments to be provided, leading to potentially security-relevant changes.

- **With all the functionality being embedded on chips, the line between embedded and IT software is blurring**
- **All CISQ weaknesses are now identified with CWE numbers (ITU-T X.1524; UN standards body)**
- **Some CISQ weaknesses presented in parent-child relationships**
- **Attempting to get CISQ quality measures referenced in revision of ISO/IEC 25023**

CISQ
Consortium for IT Software Quality

Increases in the Number of CISQ Weaknesses

Embedded extensions

Quality Attribute	Parent weaknesses	Child weaknesses	Previous weaknesses
Reliability	36	38	29
Security	36	37	22
Performance	16	3	15
Maintainability	30	1	20
Totals	118	79	86

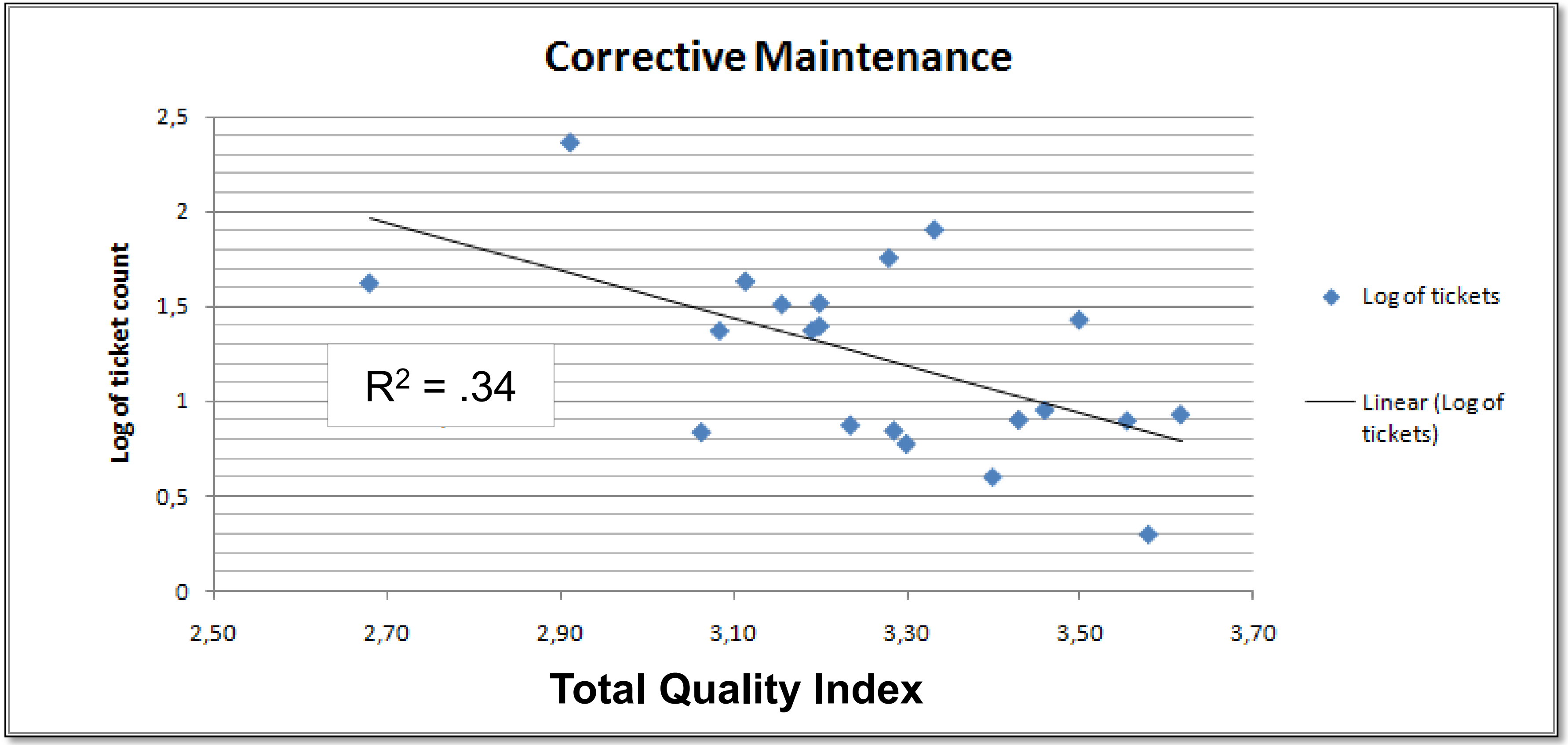
CISQ-like Measures Predict Incidents & Costs

CISQ
Consortium for IT Software Quality

Correlation of Total Quality Index and log of incidents for 21 applications in a large global system integrator

R^2 = .34
Total Quality Index accounts for 1/3 of variation in incidents

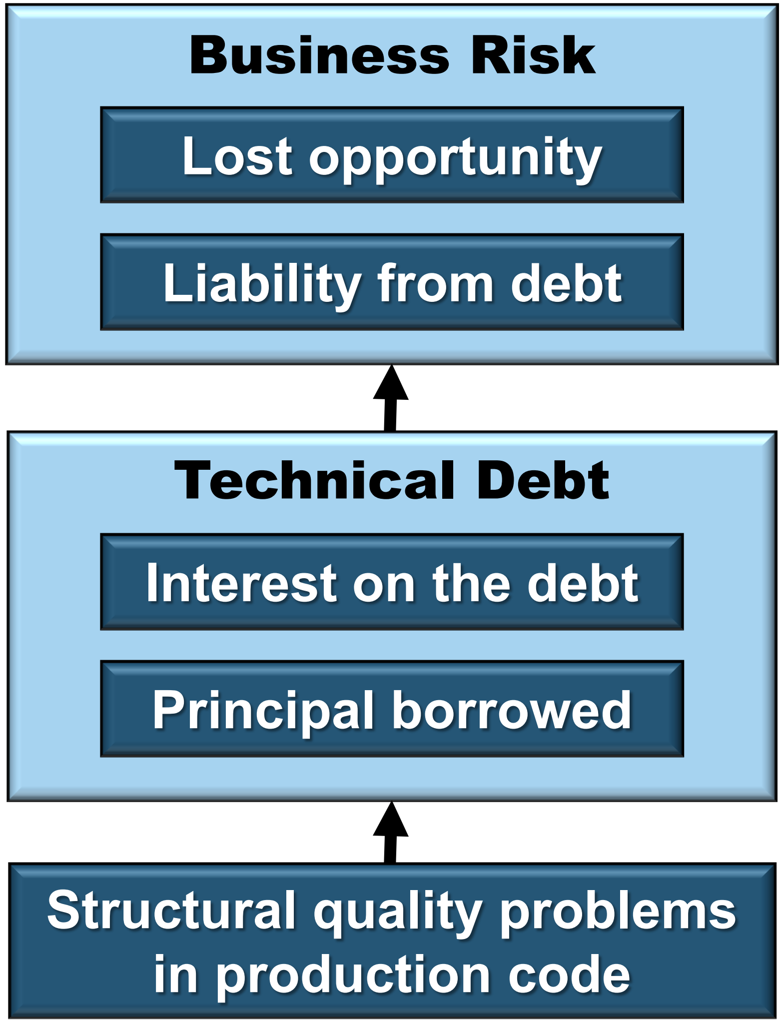
Increase in Total Quality Index of .24 decreased corrective maintenance effort 50%



CISQ
Consortium for IT Software Quality

The Technical Debt Metaphor

Technical Debt — **the future cost of repairing must-fix defects remaining in code at release, a component of the cost of ownership**



Lost opportunity—benefits that could have been achieved had resources been put on new capability rather than retiring technical debt

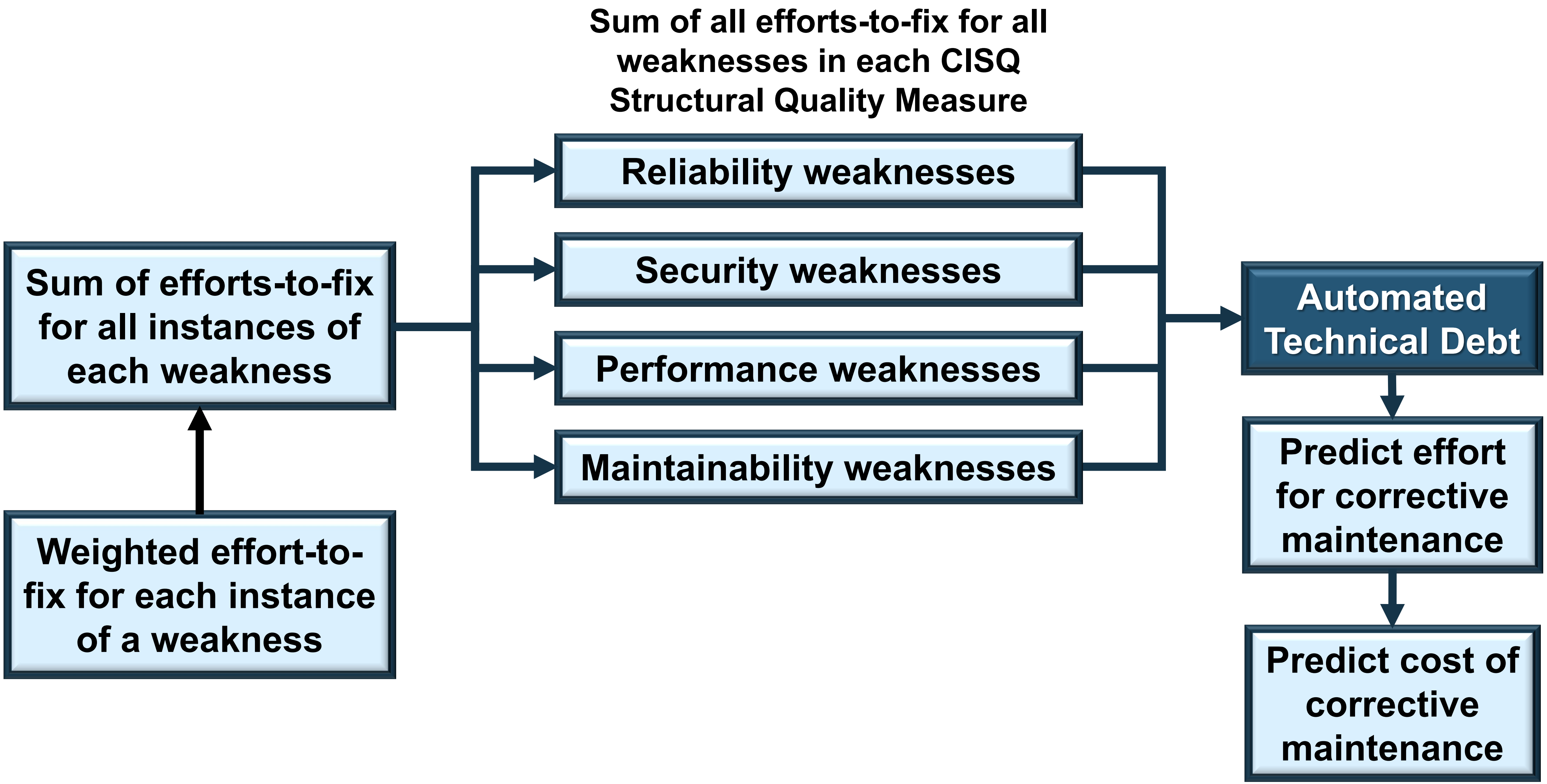
Liability—business costs related to outages, breaches, corrupted data, and other damaging incidents

Interest—continuing IT costs attributable to the violations causing technical debt, i.e, higher maintenance costs, greater resource usage, etc.

Principal—cost of fixing problems remaining in the code after release that must be remediated

Curtis, et al. (2012). IEEE Software.

Automated Technical Debt Measure



CISQ
Consortium for IT Software Quality

CISQ Roadmap

2010 | 2012 | 2014 | 2016 | 2018 | 2020 | 2022

Size

Automated Function Points → Automated Enhancement Points

Quality

ISO 25010
CWE → Automated Source Code Quality Measures
- Reliability
- Security
- Performance
- Maintainability

ISO 25023 ← (Reliability, Security, Performance, Maintainability)

Effort

Automated Technical Debt Measure

→ Acquisition Management
→ Risk & Cost Management
→ Transformation to Agile & Automation
→ Model-Based Software Engineering Measures
→ Certification
→ Due Diligence
→ Benchmarking

CISQ Conforms/Supplements ISO 25000 standards

- ISO/IEC 25010 defines a software product quality model of 8 quality characteristics
- CISQ conforms to ISO/IEC 25010 quality characteristic definitions
- ISO/IEC 25023 defines measures, but not automatable or at the source code level
- CISQ supplements ISO/IEC 25023 with automatable source code level measures

ISO/IEC 25010 — Software Product Quality

Functional Suitability	Reliability	Performance Efficiency	Operability	Security	Compatibility	Maintainability	Portability
Functional appropriateness	Maturity	Time behavior	Appropriateness	Confidentiality	Co-existence	Modularity	Adaptability
Accuracy	Availability	Resource utilization	Recognizability	Integrity	Interoperability	Reusability	Installability
Compliance	Fault tolerance	Compliance	Learnability	Non-repudiation	Compliance	Analyzability	Replaceability
	Recoverability		Ease of use	Accountability		Changeability	Compliance
	Compliance		Attractiveness	Authenticity		Modification stability	
			Technical Accessibility	Compliance		Testability	
			Compliance			Compliance	

CISQ automated structural quality measures are highlighted in blue

CISQ and the NIST Cybersecurity Framework

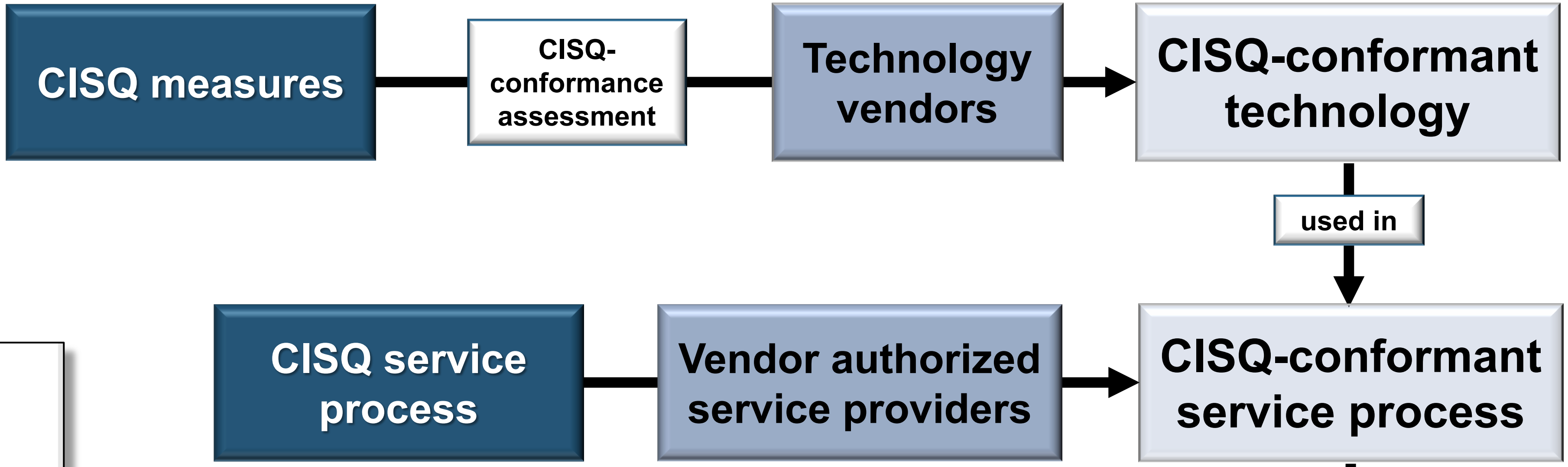
CISQ
Consortium for IT Software Quality

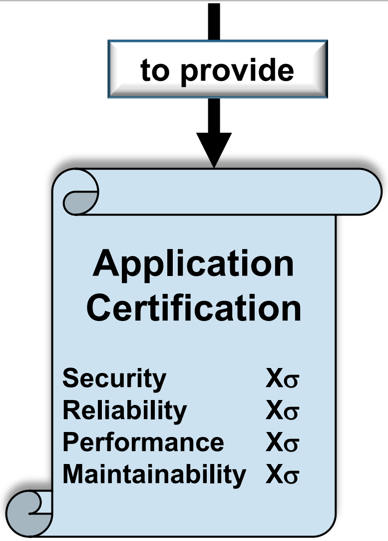
Function Unique Identifier	Function	Category Unique Identifier	Category
ID	Identify	ID.AM	Asset Management
		ID.BE	Business Environment
		ID.GV	Governance
		ID.RA	Risk Assessment
		ID.RM	Risk Management Strategy
		ID.SC	Supply Chain Risk Management
PR	Protect	PR.AC	Identity Management and Access Control
		PR.AT	Awareness and Training
		PR.DS	Data Security
		PR.IP	Information Protection Processes and Procedures
		PR.MA	Maintenance
		PR.PT	Protective Technology
DE	Detect	DE.AE	Anomalies and Events
		DE.CM	Security Continuous Monitoring
		DE.DP	Detection Processes
RS	Respond	RS.RP	Response Planning
		RS.CO	Communications
		RS.AN	Analysis
		RS.MI	Mitigation
		RS.IM	Improvements
RC	Recover	RC.RP	Recovery Planning
		RC.IM	Improvements
		RC.CO	Communications

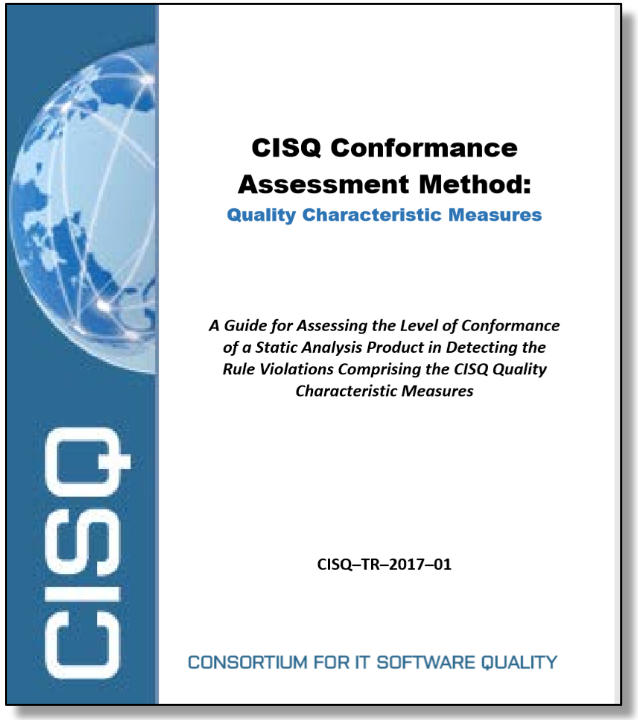
The CISQ Security measure (and others) can be used in numerous processes of the NIST Cybersecurity Framework. Some examples:

- ID.RA: **Empirical risk tolerance thresholds for software security**
- ID.SC: **Contractual SLAs and audits for software security**
- PR.IP: **Evaluation of software assets for security weaknesses**
- PR.MA: **Continual improvement of software security**
- DE.CM: **Periodic scans for software weaknesses**
- RS.CO: **Software security and weakness data are shared**
- RS.MI: **Security weaknesses are identified and mitigated**

Application Certification Using CISQ







- **CISQ/OMG**
  - only assess vendor conformance
  - do not certify applications
  - program initiated in 2017
- **Service providers**
  - use CISQ-conformant technology
  - in a CISQ-conformant service process
  - to provide application certifications

CISQ
Consortium for IT Software Quality

Trustworthiness Measures for Model-Based Engineering

Objective — Define quality measures based on counting severe architectural and design weaknesses that can be detected through analyzing formal models developed in Model-Based System Engineering (MBSE) languages and technologies.

Two Focii —

1. Quality of the architecture:
   - Architecture analysis might be the only way to find some weaknesses
   - Find other weaknesses earlier at the architectural level
2. Quality of the model of the architecture

Sources —

1. Architectural-level CWEs
2. Lists of architecture-level antipatterns
3. Vendor and system architect weakness lists or experiences

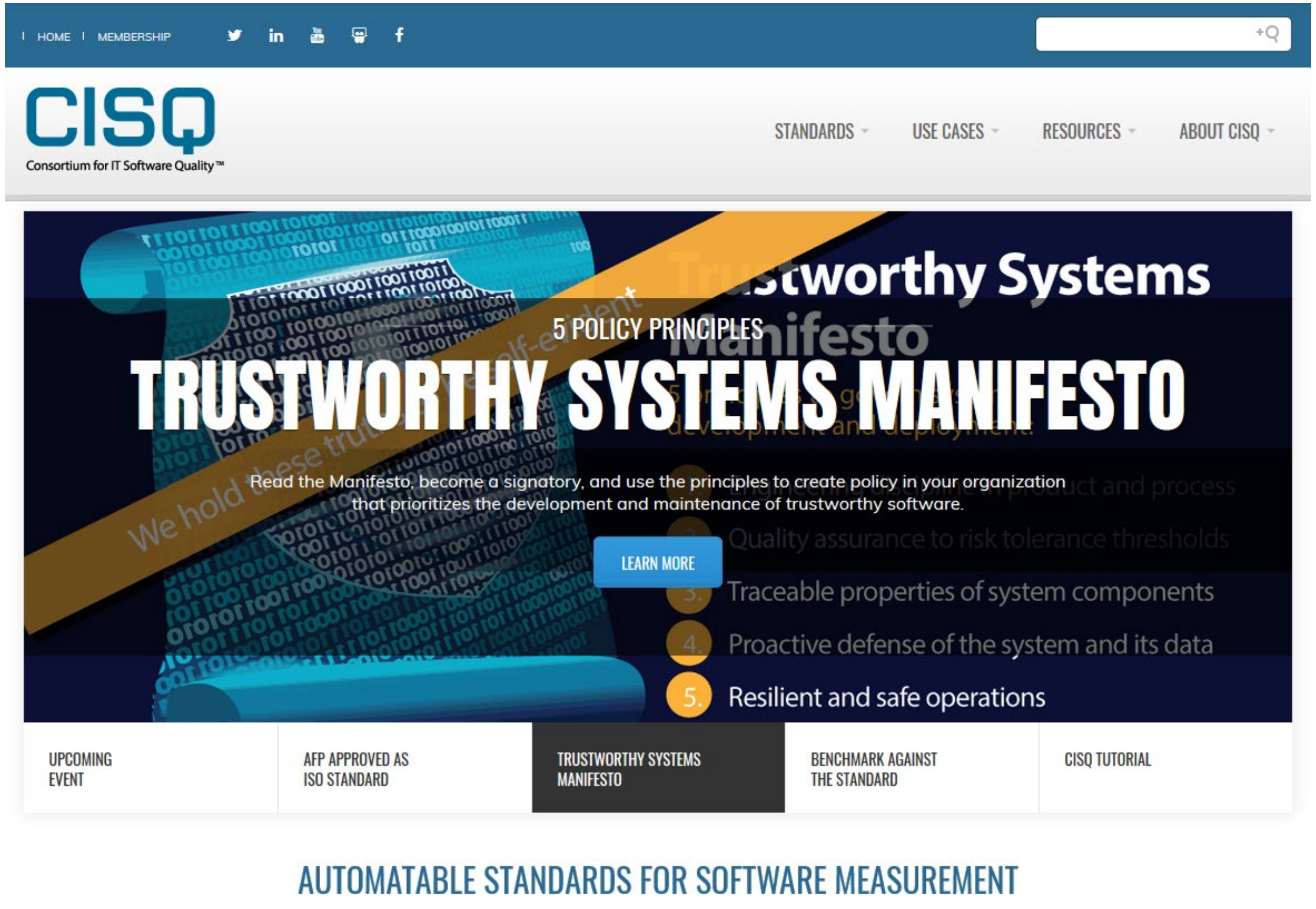
Trustworthy Systems Manifesto

As a greater portion of mission, business, and safety critical functionality is committed to software-intensive systems, these systems become one of, if not the largest source of risk to enterprises and their customers. Since corporate executives are ultimately responsible for managing this risk, we establish the following principles to govern system development and deployment.

1. **Engineering discipline in product and process**
2. **Quality assurance to risk tolerance thresholds**
3. **Traceable properties of system components**
4. **Proactive defense of the system and its data**
5. **Resilient and safe operations**

CISQ Membership Is Free — www.it-cisq.org

CISQ
Consortium for IT Software Quality



Over 2000 individual members from large software-intensive organizations: