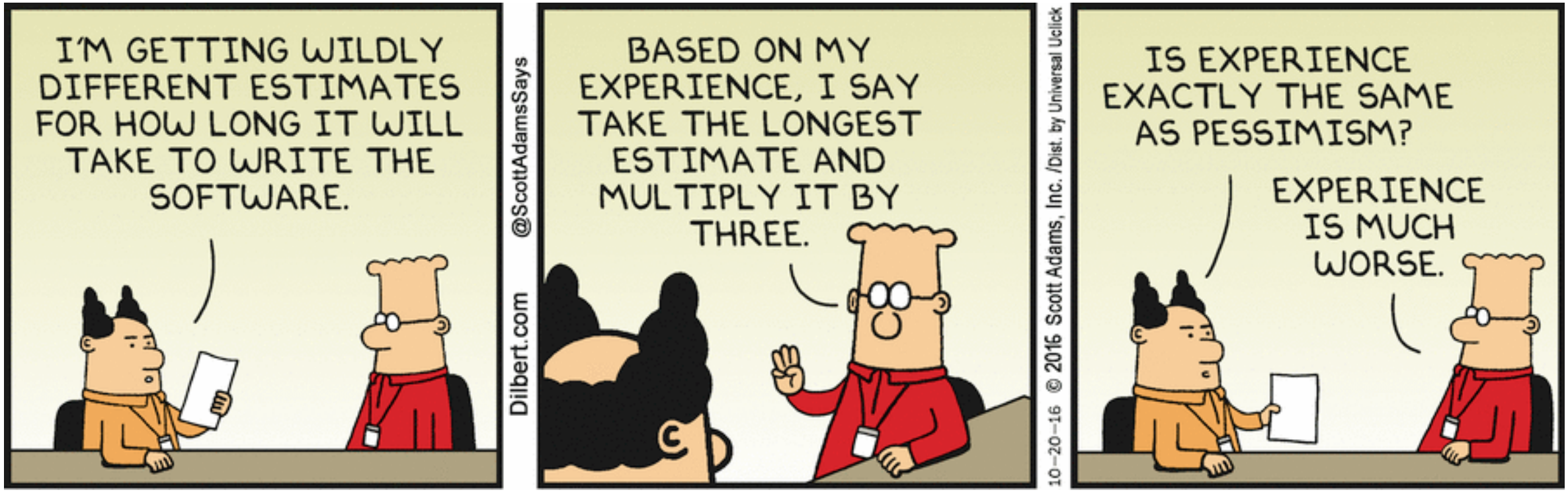
Presented at the 2018 ICEAA Professional Development & Training Workshop - www.iceaaonline.com

Establishing Standards as the Basis for Effective Measurements and Affordability

Pete Pizzutillo

CISQ
Consortium for IT Software Quality

Sound familiar?

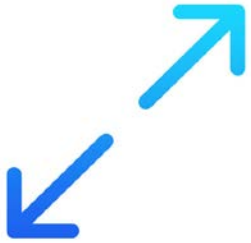


Agenda

- About CISQ
- Why measure software
- Review CISQ standards
- Measuring the software supply chain
- Affordable Software: Maintainability & Technical Debt
- Measuring security
- Certifying software
- Getting involved with CISQ

Introducing CISQ Standards

CISQ is an IT leadership group that has developed international OMG® standards for automating the measurement of software from the source code.



the **size** of a code base

for measuring development productivity

its **structural quality**

security, reliability, performance efficiency, maintainability

technical debt

critical violations of good coding and architectural practice that live in the code

CISQ Mission

To develop computable measures and anti-patterns to be used for evaluating multi-tier IT application software:

- **Establish a computable software quality standard** for IT applications with scoring guidelines
- **Recommend measurement thresholds** against which minimally acceptable levels of quality and other attributes of business application software can be assessed.
- **Develop baselines for benchmarking** application quality, productivity, cost, and other attributes across application domains and industry segments.
- **Conduct case study research** with consortium sponsors validating application metrics and their business value.
- **Provide a source of application measurement expertise** to consortium sponsors.

About CISQ Leadership, Members and Partners

Dr. Bill Curtis
Executive Director

Leads CISQ working groups
American lead on ISO 25000 standards
Led development of Capability Maturity Model

Co-founders:

OMG Object Management Group®

Software Engineering Institute
Carnegie Mellon University

Sponsors on Governing Board:

Synopsys® CAST Cognizant CGI Tech Mahindra SHPI Northrop Grumman

Over 2000 members; large SW-intensive organizations:

accenture, U.S. Department of Homeland Security, amazon, AXA, at&t, MITRE, Infosys, BOSCH, ISG, Boeing, aetna, Capgemini, Booz | Allen | Hamilton, Credit Suisse, Capital One, IAOP, Gartner, HCL, citi, Duke Energy, DreamWorks, Huawei, Atos, Dell, ca technologies, FedEx, Discovery Health, DCG, Cisco, Fidelity Investments, CenturyLink, IBM, Allianz, Barclays

Why measure software?

Visible Symptoms

Invisible Root Causes

TEST

STRUCTURAL ANALYSIS

Why measure software?

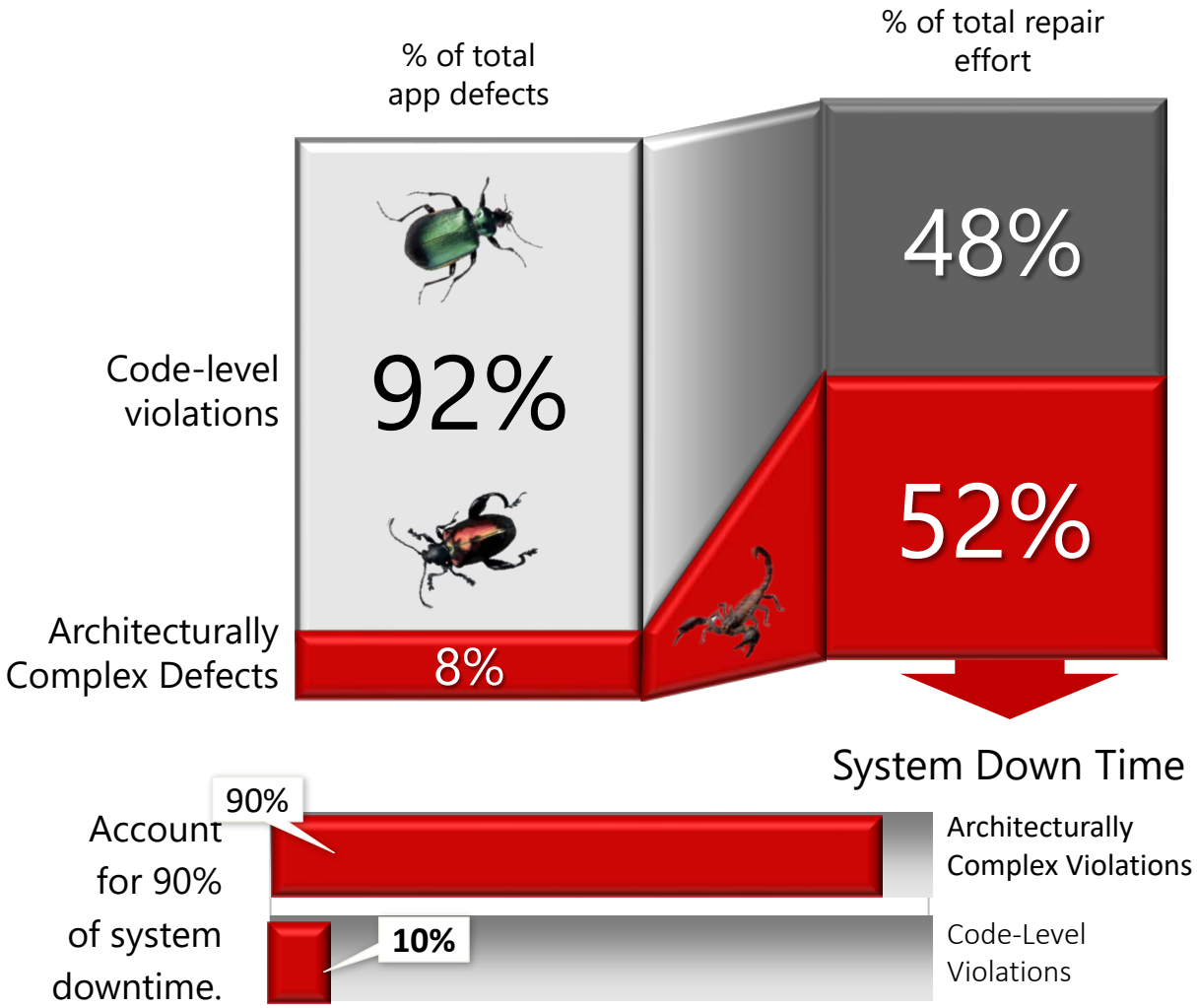
Visible Symptoms

Invisible Root Causes

TEST

STRUCTURAL ANALYSIS

Structural flaws involve interactions among multiple components and application layers and are undetectable by traditional testing.

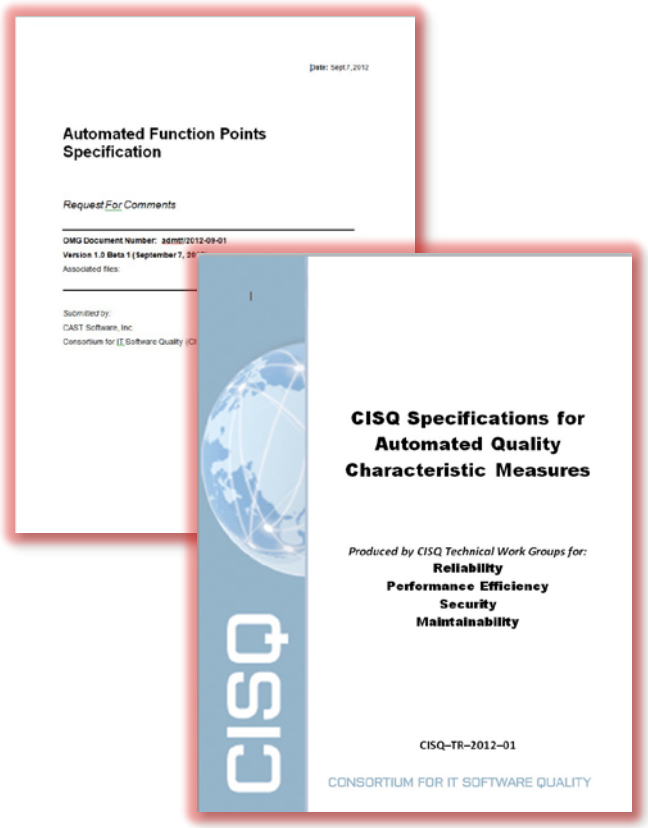


Source: Code Complete, Steve McConnell

Software is a deep mystery to many

- Why isn't software modeled like other large, complex engineering systems?
  - Airbus wouldn't bend a single piece of metal (or carbon fiber) before simulating the part, manufacturing the part, and maintaining the part
- Why isn't software acceptance subject to quality metrics like other component parts?
  - Boeing wouldn't accept a single fastener without checking against design specifications

OMG Standards Developed by CISQ



Standards available for free at:

- www.omg.org/spec
- www.it-cisq.org/standards

Software Sizing

Automated Function Points	Measures the functional size of software
Automated Enhancement Points	Measures the size of both functional and non-functional code in one measure

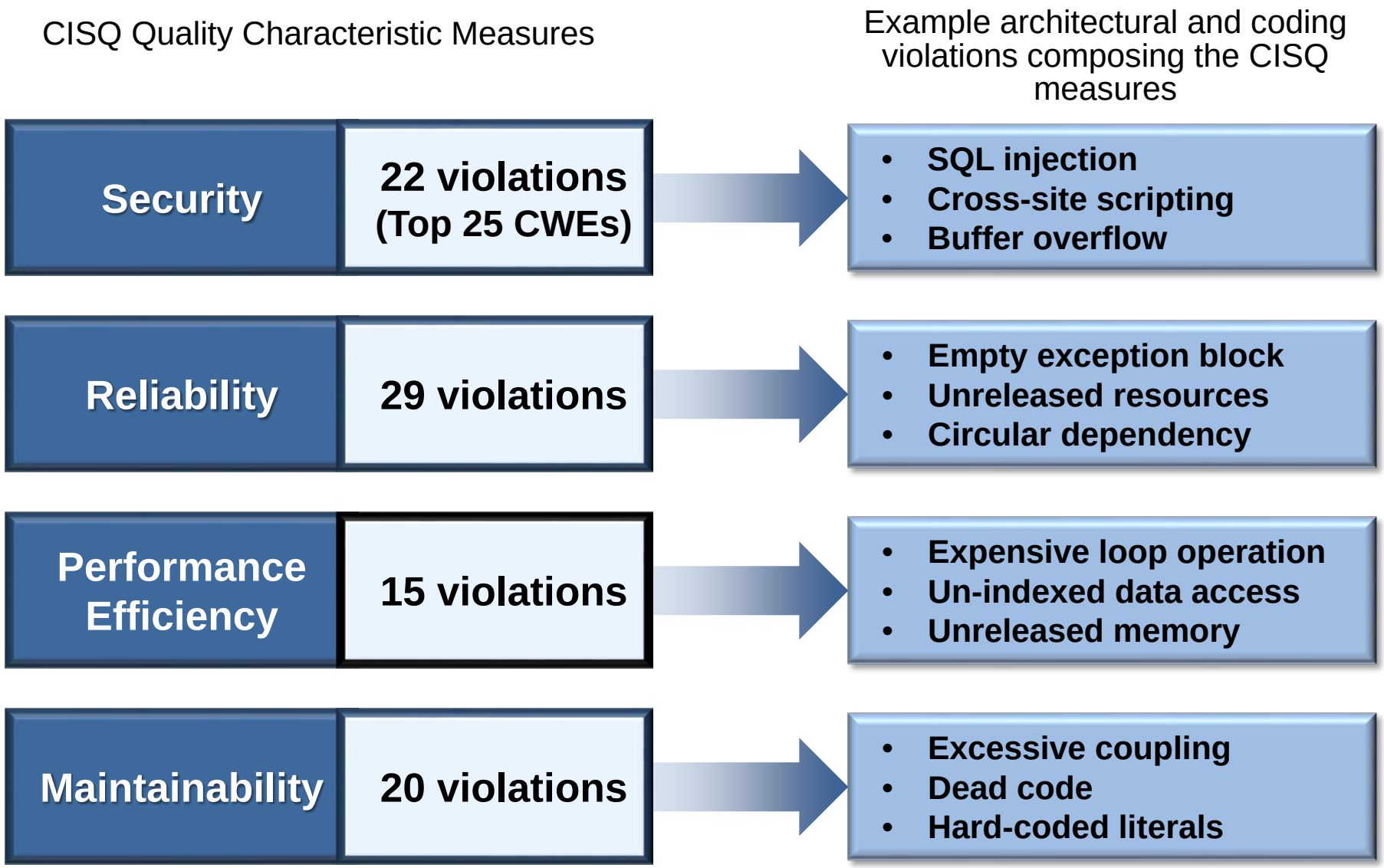
Code Quality

Security	Measures 22 violations in source code representing the most exploited security weaknesses in software – CWE/Sans Institute Top 25 Most Dangerous Security Errors, OWASP Top 10
Reliability	Measures 29 violations in source code impacting the availability, fault tolerance, and recoverability of software
Performance Efficiency	Measures 15 violations in source code impacting response time and utilization of processor, memory, and other resources
Maintainability	Measures 20 violations in source code impacting the comprehensibility, changeability, testability, and scalability of software

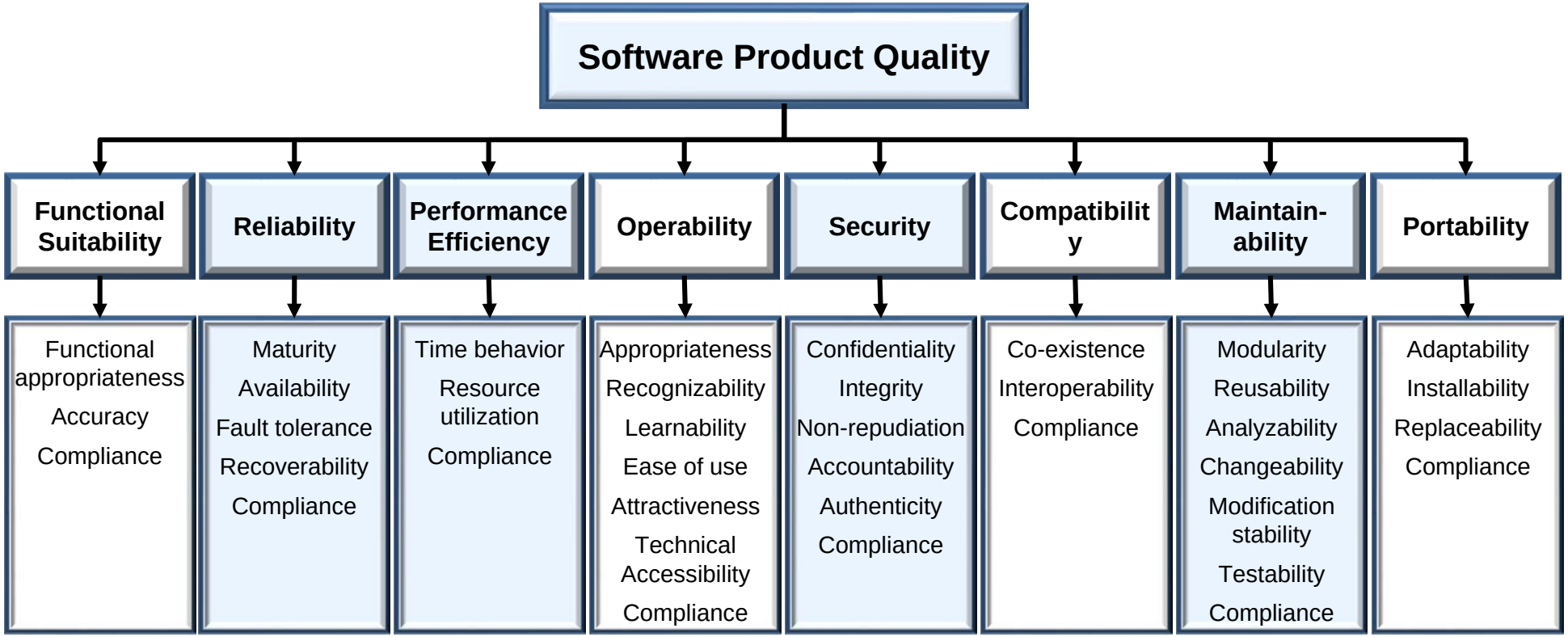
Technical Debt

Automated Technical Debt	A measure of corrective maintenance effort due to violations (weaknesses) remaining in a software application

Defining 'good software'



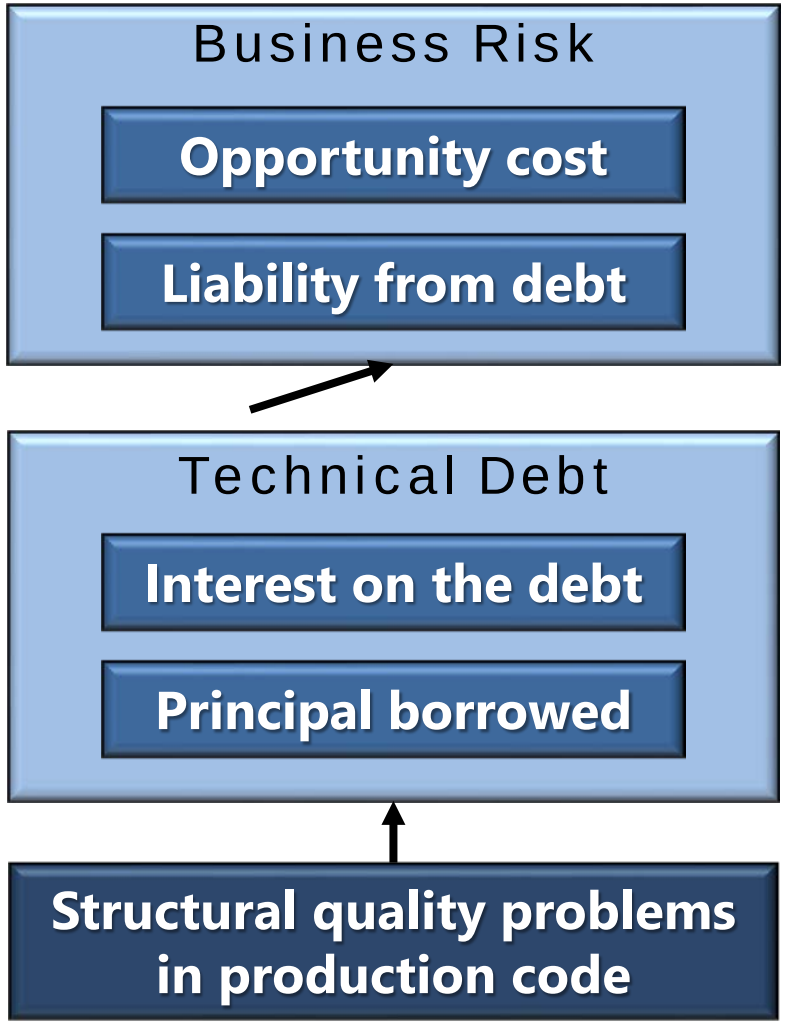
Conforms to / Supplements ISO 25000 Series



- ISO 25000 series replaces ISO/IEC 9126 (Parts 1-4)
- ISO 25010 defines quality characteristics and sub-characteristics
- CISQ conforms to ISO 25010 quality characteristic definitions
- ISO 25023 defines measures, but not at the source code level
- CISQ supplements ISO 25023 with source code level measures

The Technical Debt Metaphor

Technical Debt - the future cost of defects remaining in code at release, a component of the cost of ownership



Opportunity cost—**benefits that could have been achieved had resources been put on new capability rather than retiring technical debt**

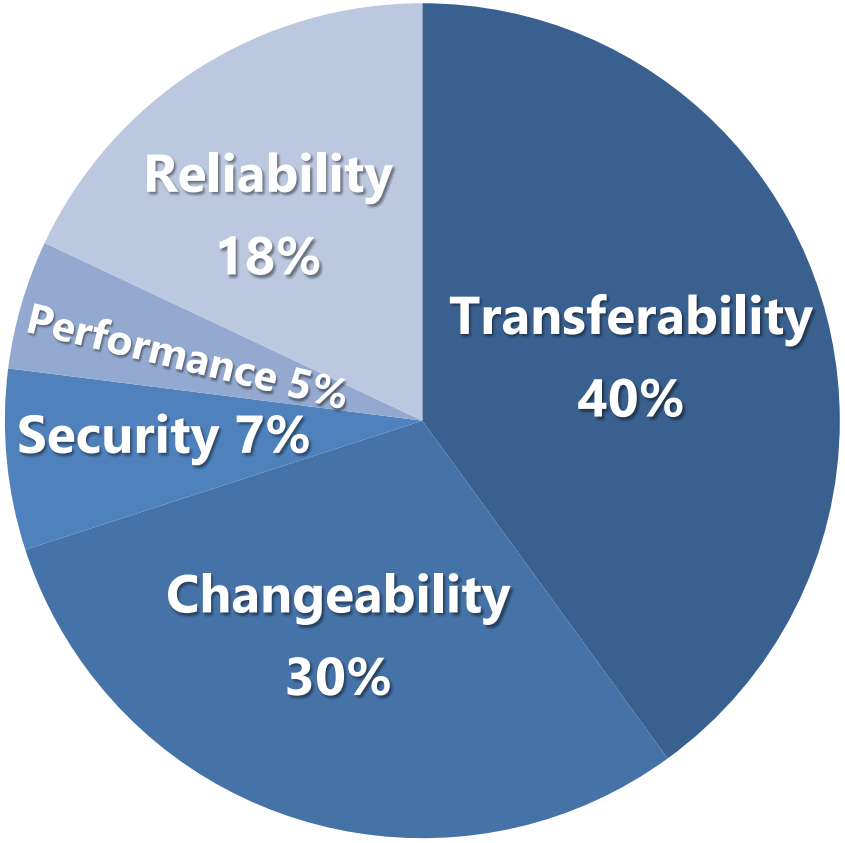
Liability—**business costs related to outages, breaches, corrupted data, etc.**

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**

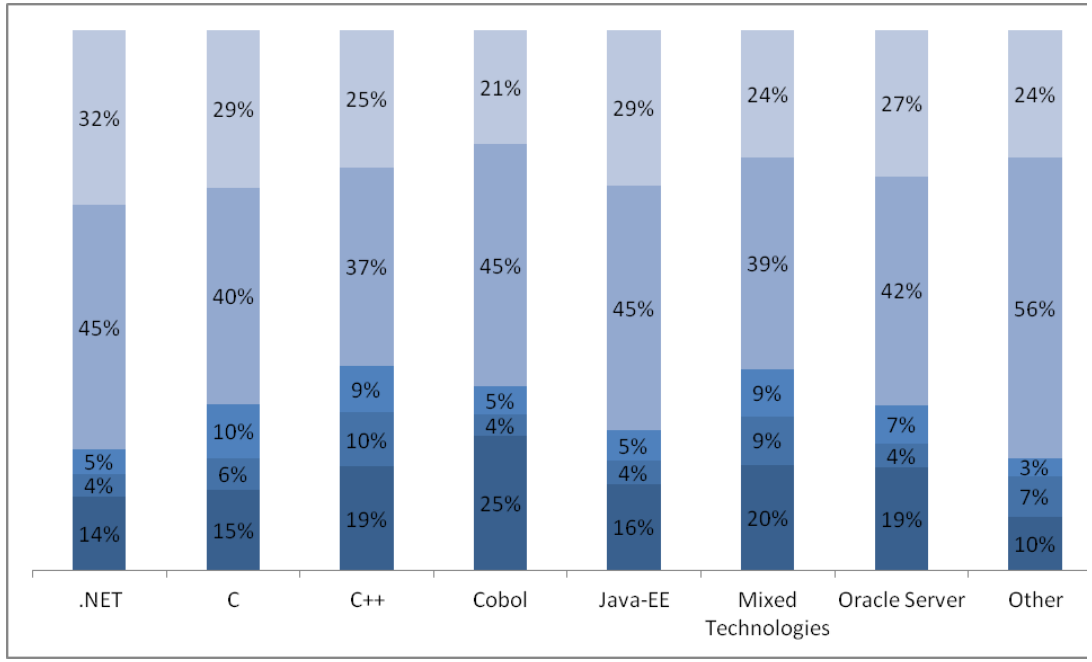
Curtis, et al. (2012). IEEE Software.

Tech Debt: Maintainability and Risk



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

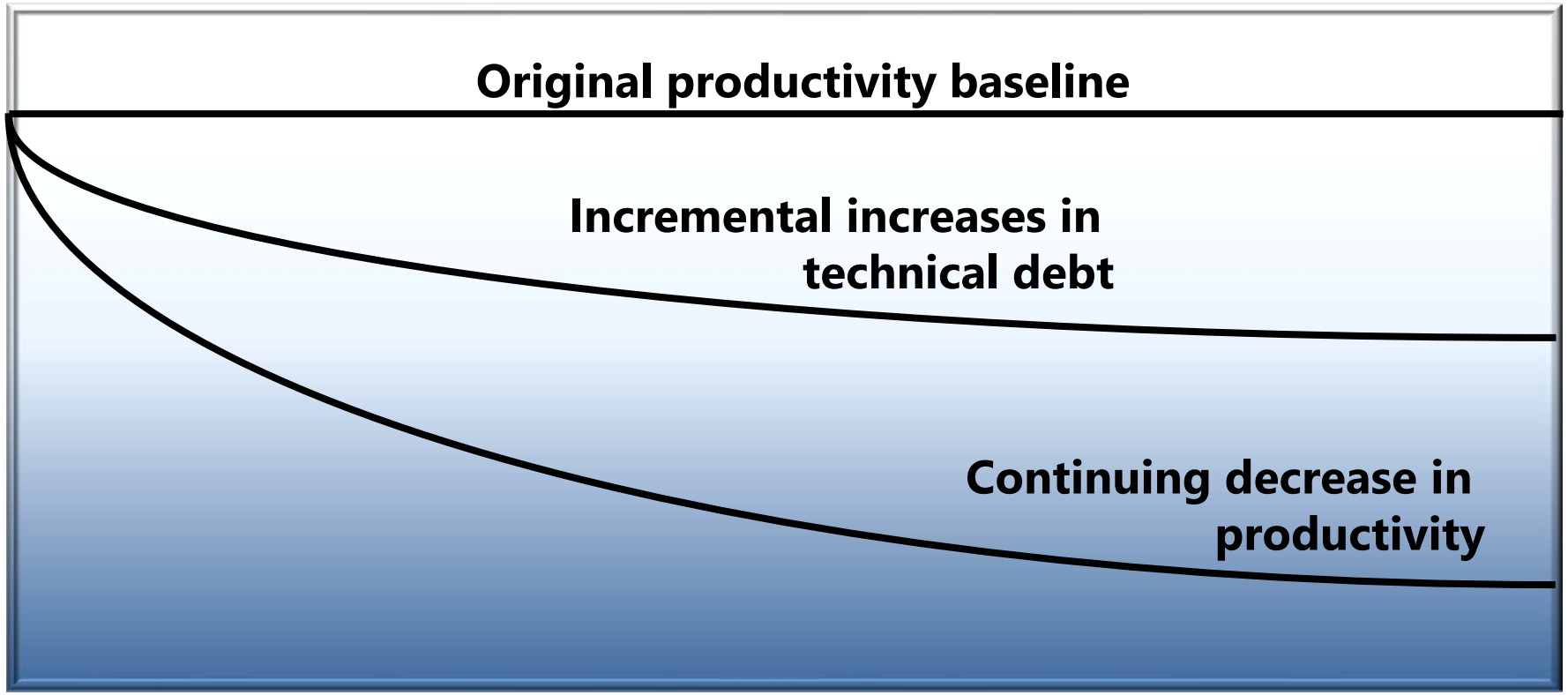
- **70% of Technical Debt are factors that drive cost**
  (Transferability, Changeability)
- **30% of Technical Debt creates operational risk**
  (Robustness, Performance, Security)
  - Sometimes called Security Debt or Risky Debt
- **Health Factor proportions are mostly consistent across technologies**



Technical Debt Affects Developer Productivity

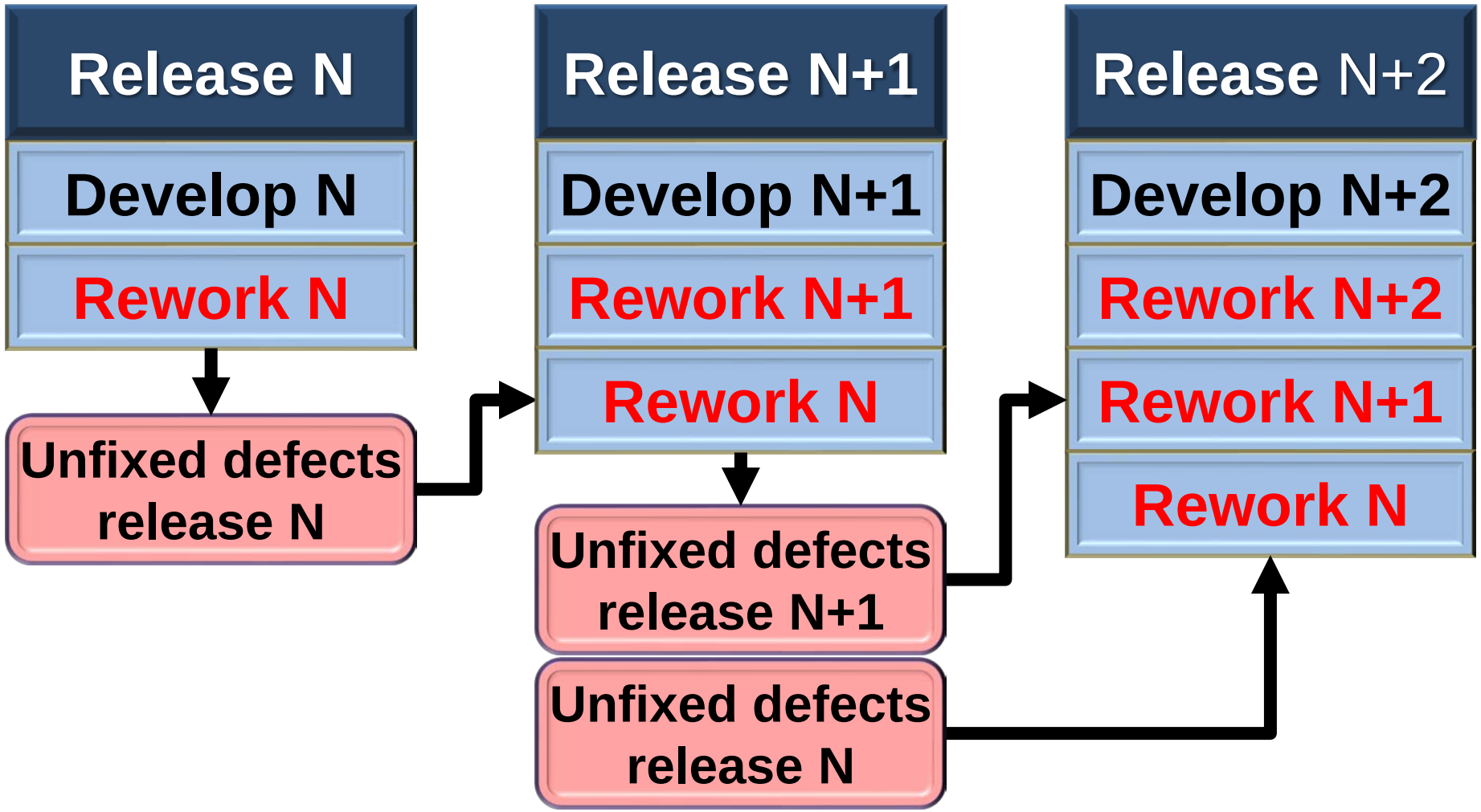
Assumption: Productivity is a stable number

Reality: Productivity is a monotonically decreasing function of releases



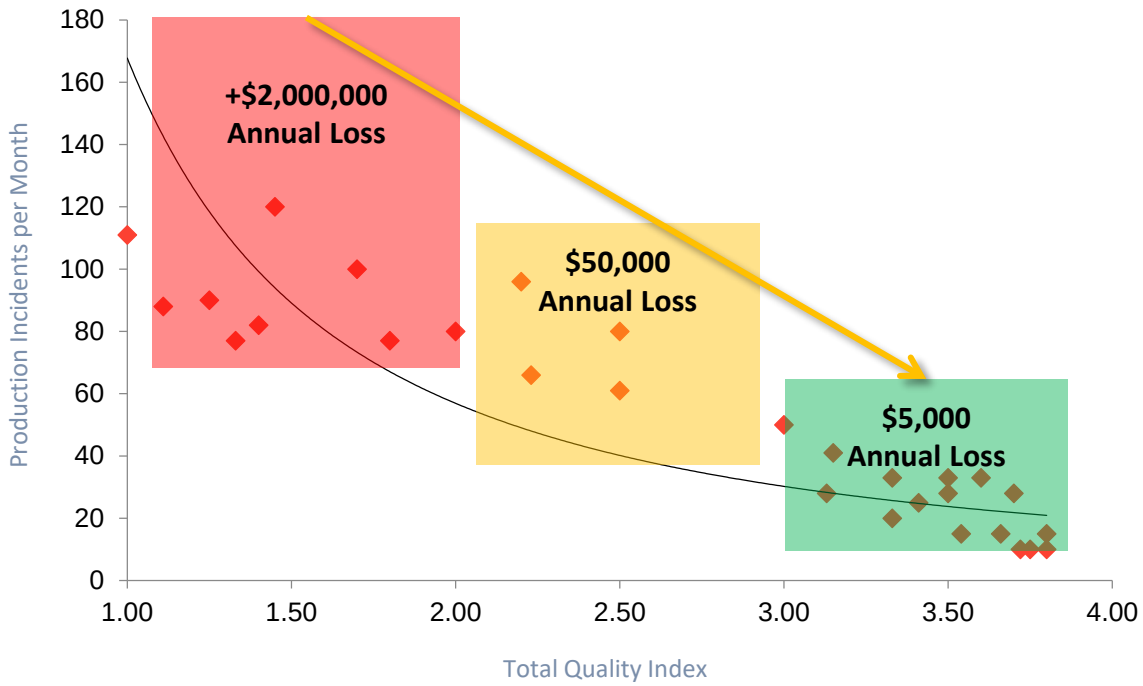
Unless you take action !!!

Rework is Technical Debt



No "functional" credit for rework → **Charge rework effort against the release where defect was injected**

Financial impact of technical debt

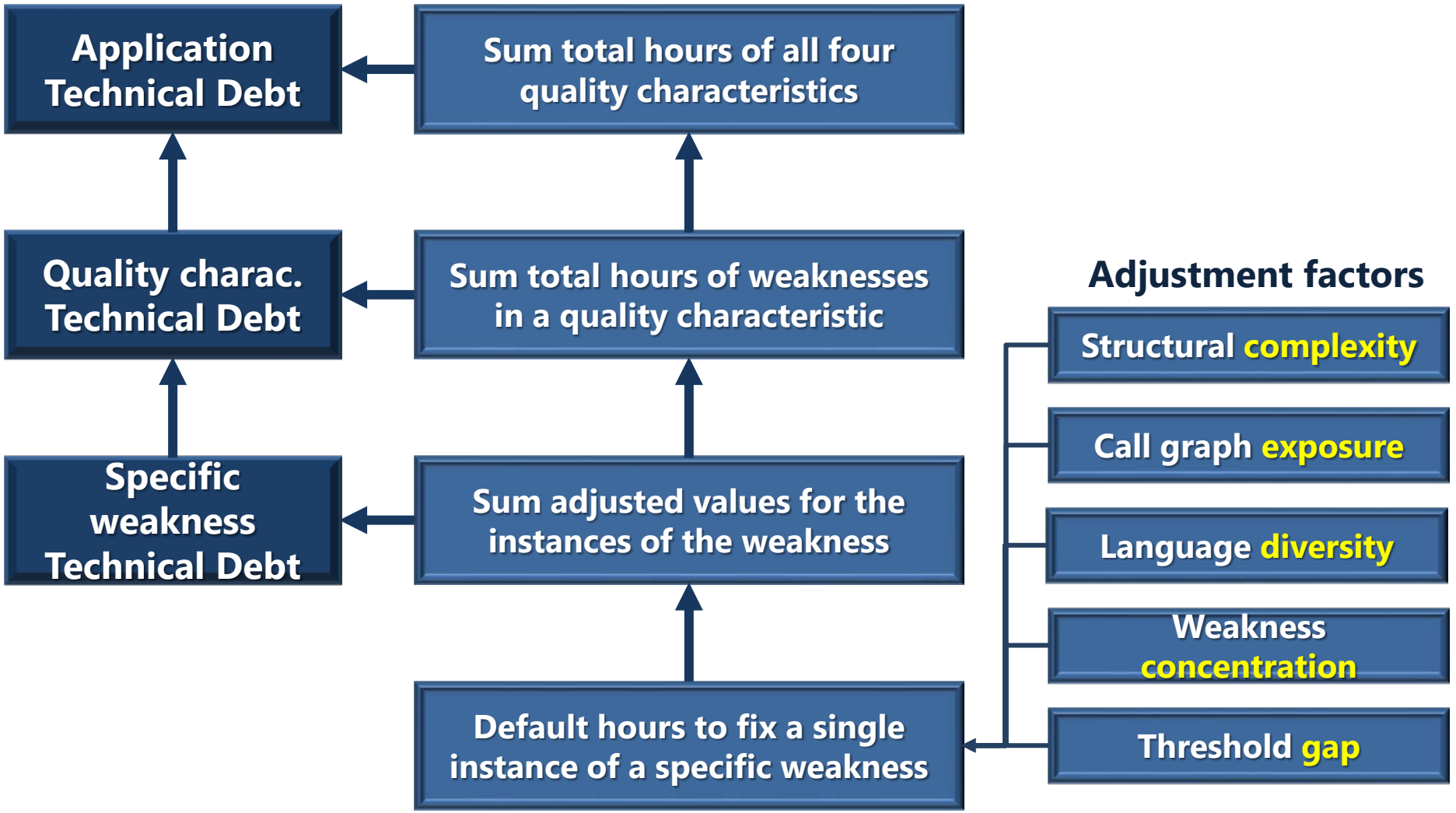


Challenges:

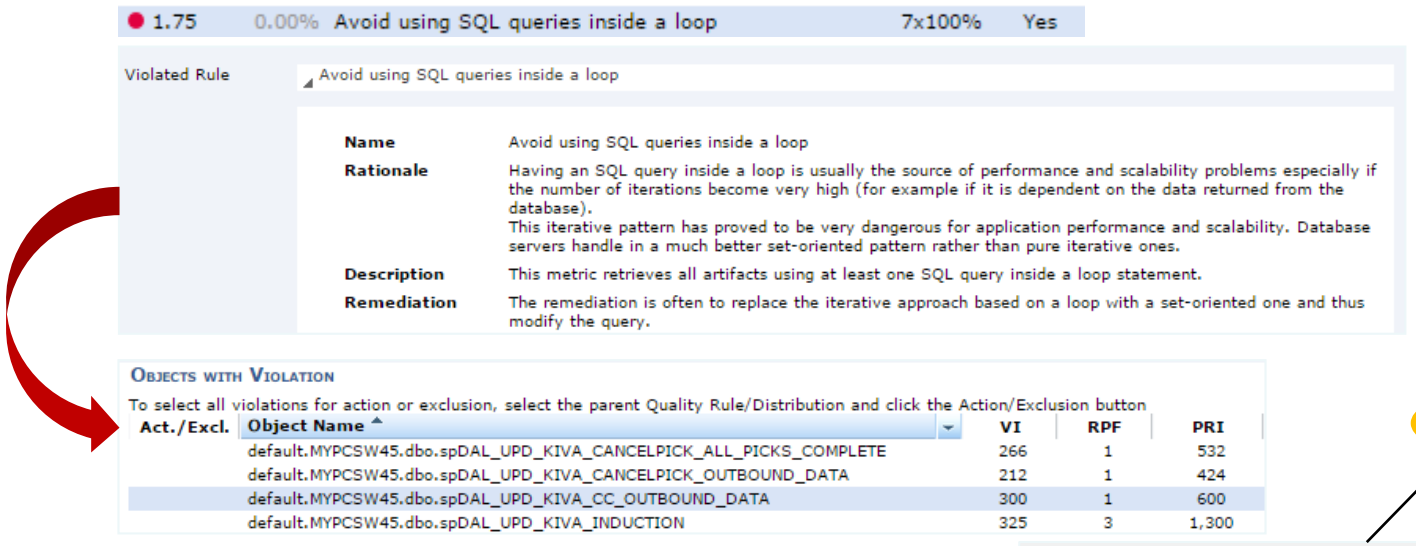
- Pervasive quality and incident issues on multiple key applications
- Instituting broad quality improvement initiative across the ADM organization

- +30 application analyzed
- Application with lower technical debt have far fewer production incidents and lower financial losses
- Strong correlation between overall structural quality of application and the fewer production incidents and overall financial impact to organization.

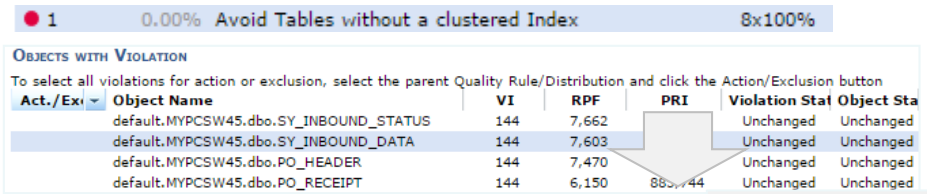
CISQ Technical Debt Calculation



Calculating technical debt is software estimation



1) Intensive background task



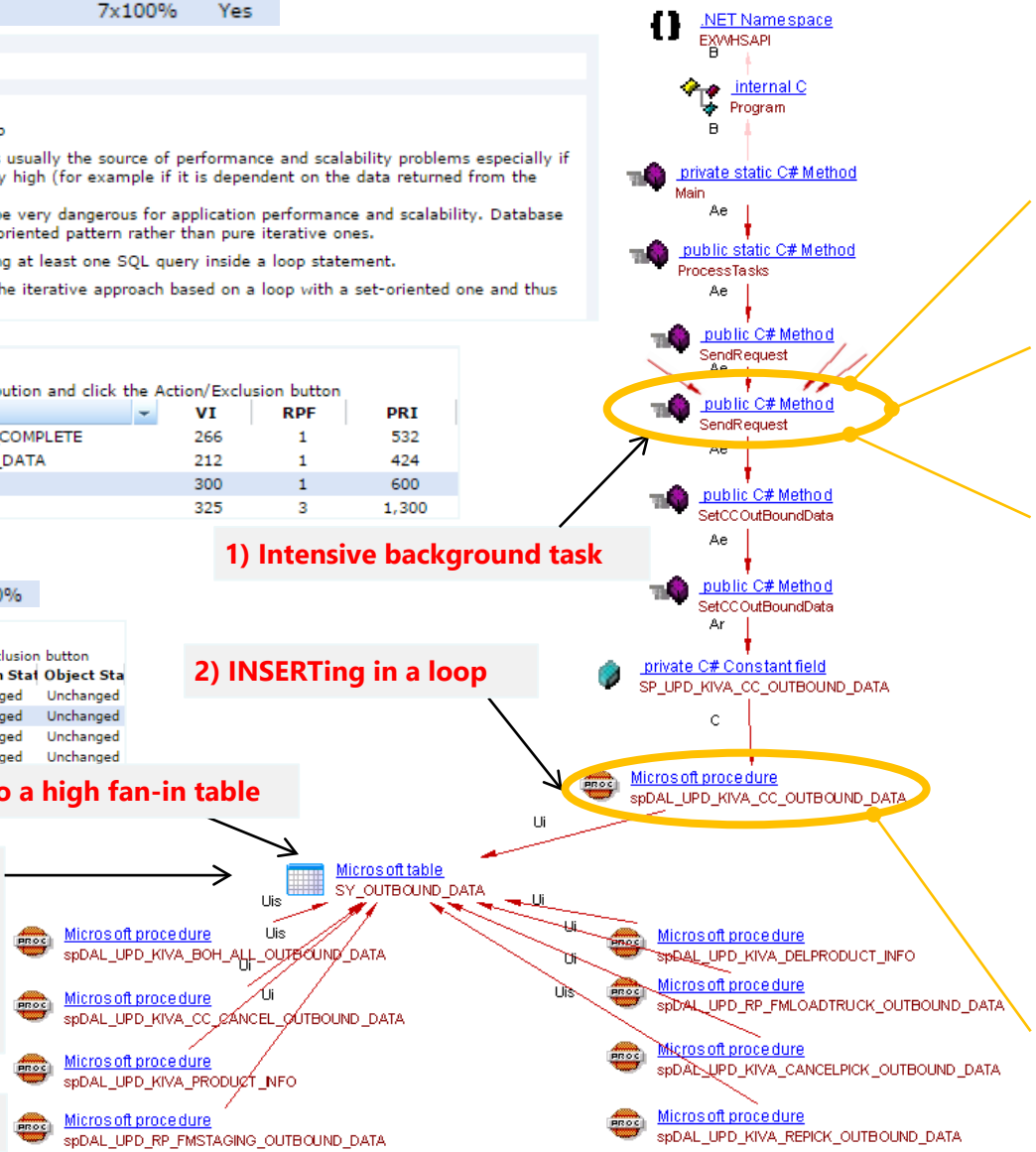
2) INSERTing in a loop

3) Into a high fan-in table

4) with no Clustered Index and no Partitions

→ all INSERTS go to the last page of the table, creating a locking **HOTSPOT**

1) + 2) + 3) + 4) = High lock contention Risk



How complex is this component?

How much higher is complexity than needed?

How many objects calling this component?

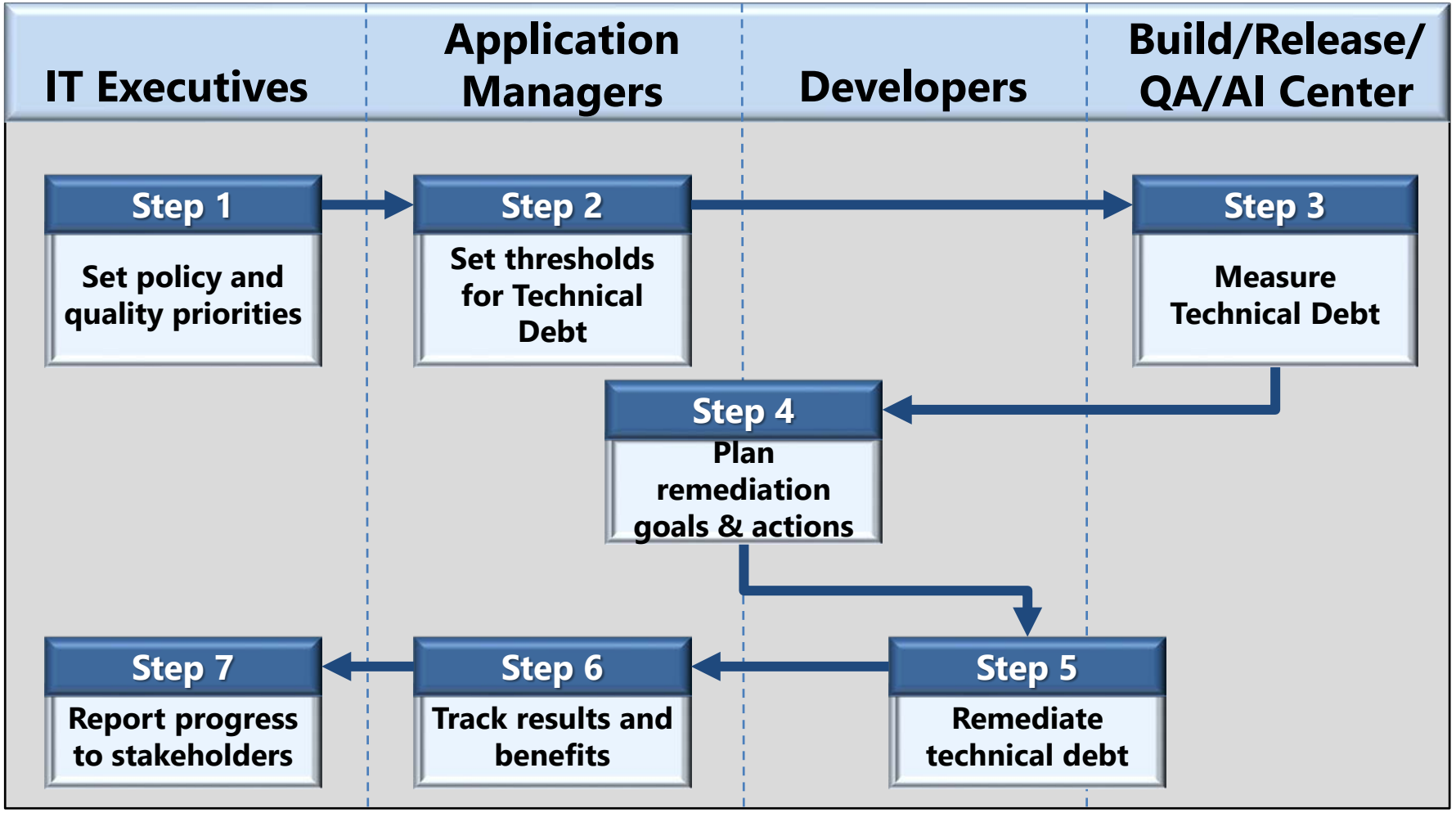
How many components do I need to fix?

How many languages?

Are there other weaknesses in there?

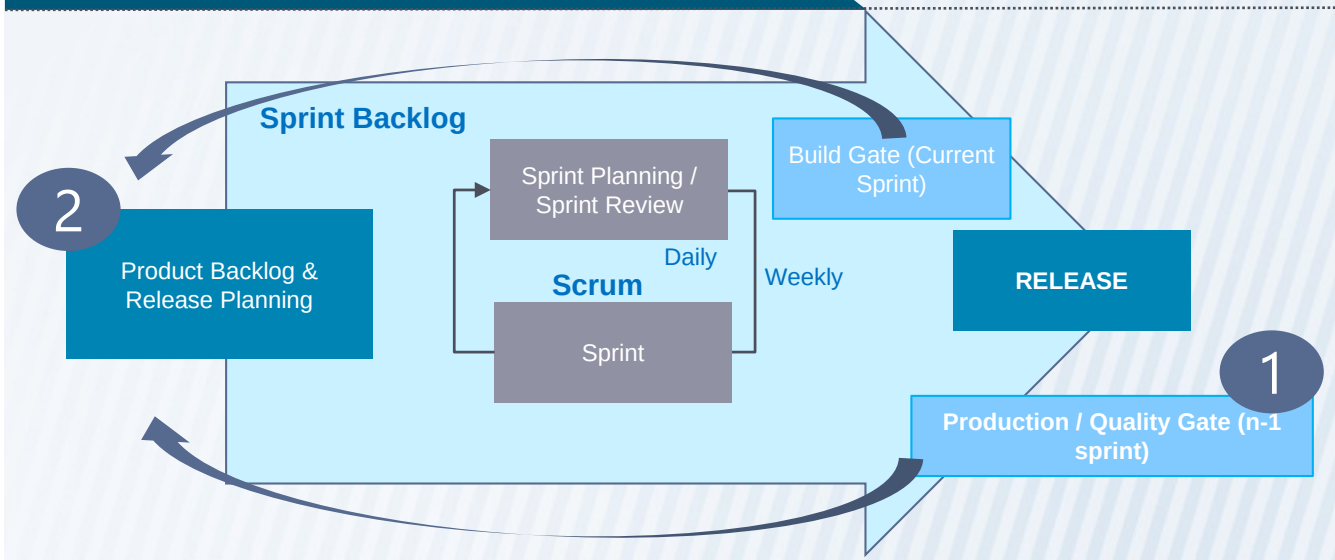
Technical Debt as an Affordability Management Process

- **Managing Technical Debt is a process that must be integrated into the SDLC**
- **Executives must protect time for removing Technical Debt by policy**
- **Technical Debt should be measured, tracked, and reported regularly**
- **Failure to manage Technical Debt builds legacy daily!**



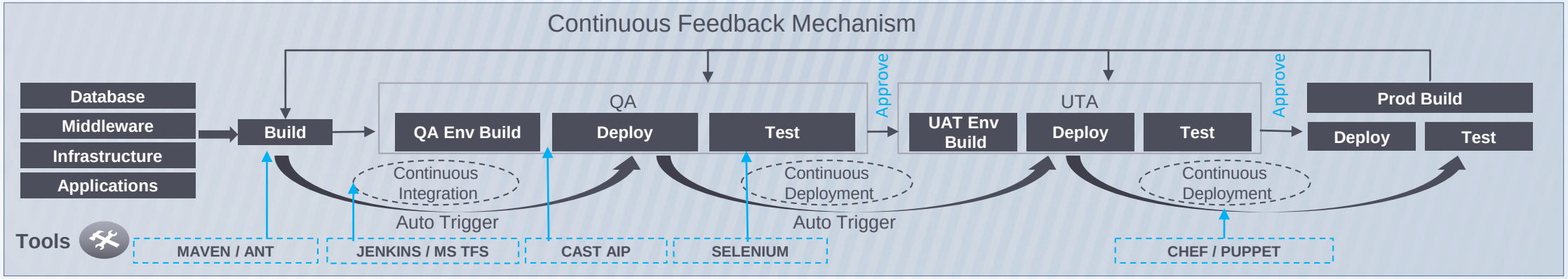
Measuring / Managing Tech Debt in Agile & DevOps

Agile Sprints



1. Prevent new technical debt by **"shift left"** and changing developer behavior
2. **Allocate** up to **10%** of sprint cycle to fix technical debt that goes into the product backlog

Automated Pipeline



Security breaches lead to bad headlines

Facts & Figures

- Estimated average annualized cost of cybersecurity is $11.7M
- 22.7% increase in cost of cybersecurity in a year
- Estimated average number of security breaches each year is 130
- 27.4% increase in average annual number of security breaches
- Forbes - cybercrime will cost approximately $6 trillion per year on average through 2021

Attack Types vs. security investment

Annualized Cost for different types of security attack

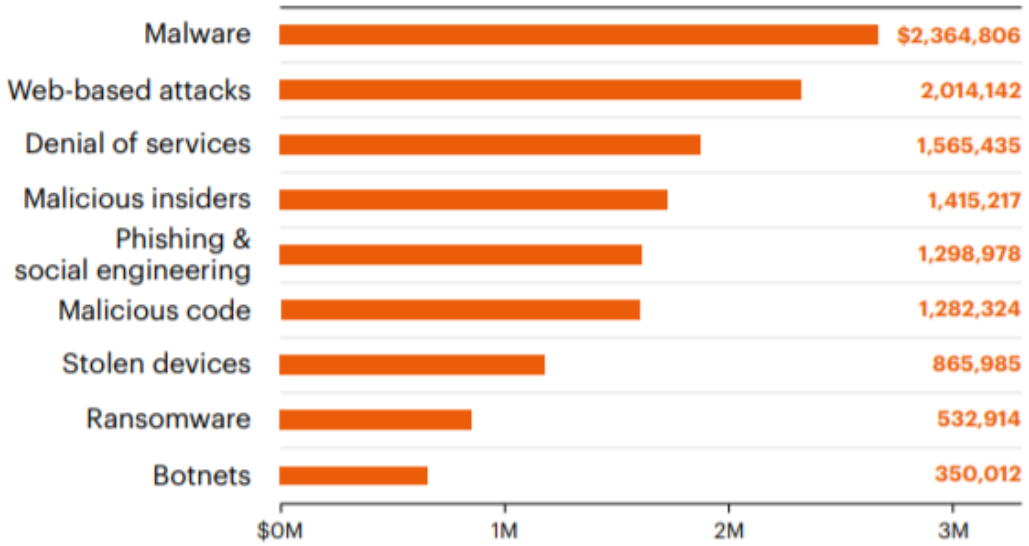
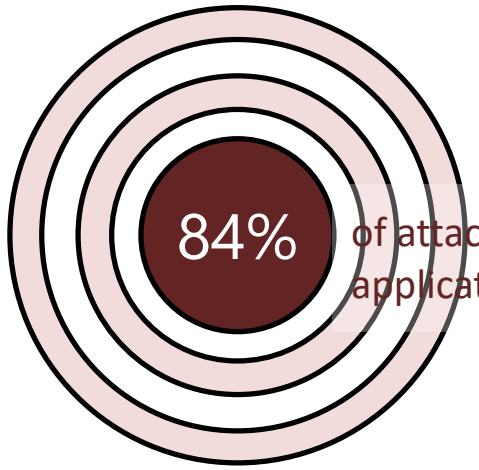


FIGURE 13
Total annualized cyber crime cost for attack types
US$ millions

Legend
Consolidated view
n = 254 separate companies

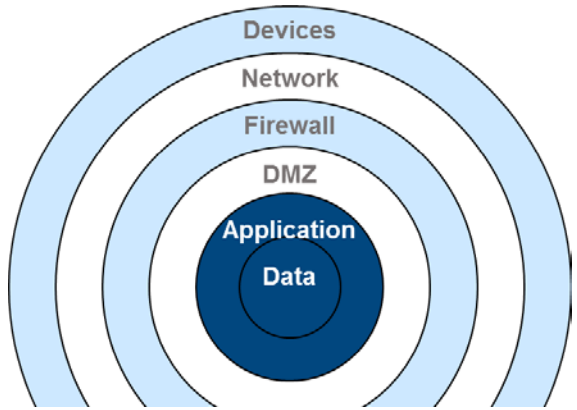
Source - Accenture



of attacks target application layer

Source: Gartner

"Up to 70% of CWEs are actually quality defects." Source: SEI



- Security breaches due to web-based attacks, malicious insiders, and malicious code are on the rise and costs due to these breaches are significantly high
- Yet spending on application and data security tends to be lowest

1 : 23 Ratio by which spend on perimeter outstrips application security

50% security issues due to design flaws

Flaws of omission

Occurs due to ignorance of a security requirement or potential threat

Ex - store a password in a file without encryption.

Flaws of commission

Design decision which can lead to undesirable consequences

Ex – client side authentication

Flaws of realization

The design is correct, but implementation suffers from coding mistakes

Ex – input sanitization

IEEE
Advancing Technology for Humanity

"Architectural flaws are results of inappropriate design choices in early stages of software development, incorrect implementation of security patterns, or degradation of security architecture over time."

Security Measures

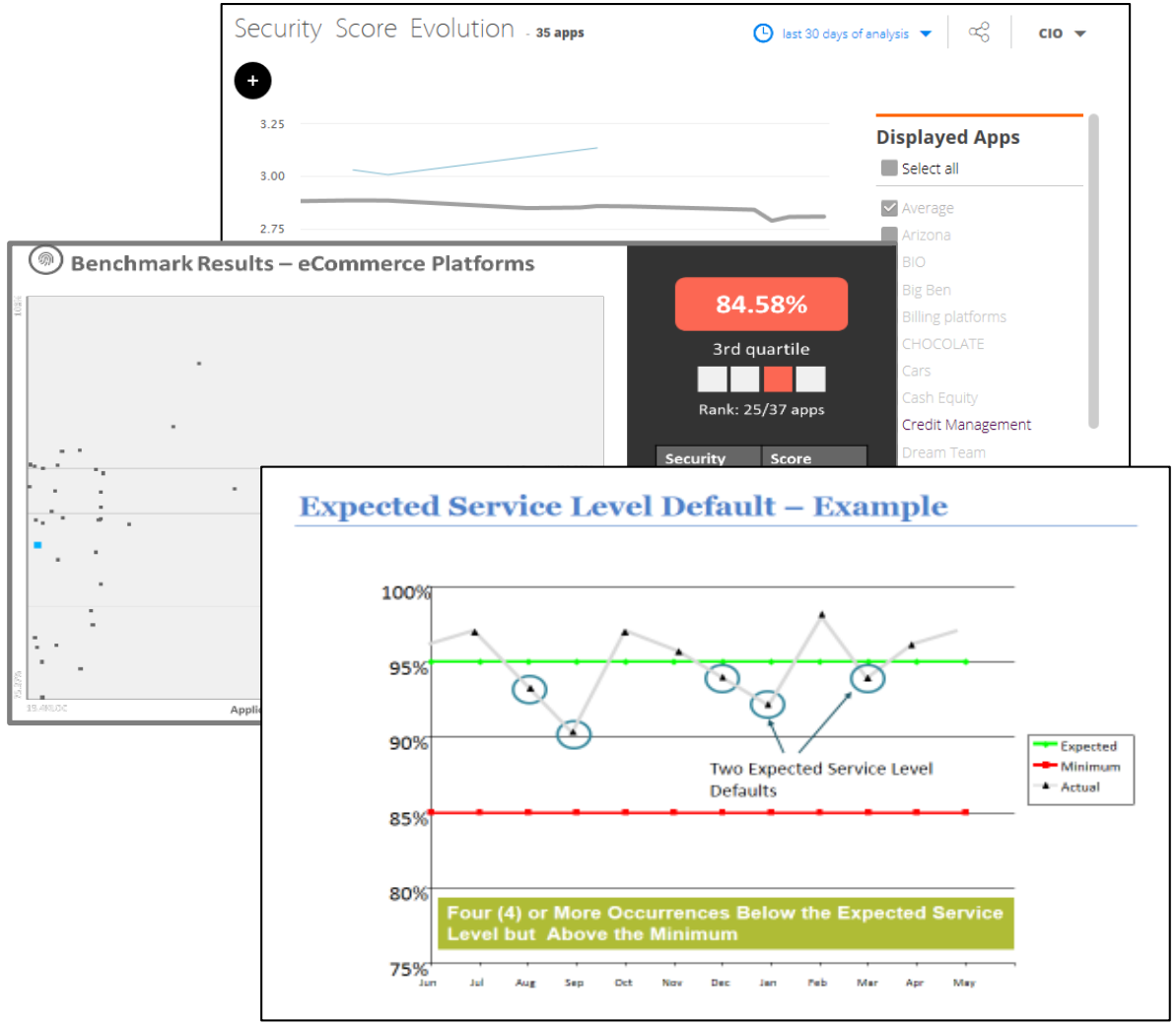
Robert Martin
MITRE

Common Weakness Enumeration
cwe.mitre.org

- CWE-22 Path Traversal Improper Input Neutralization
- CWE-78 OS Command Injection Improper Input Neutralization
- CWE-79 Cross-site Scripting Improper Input Neutralization
- CWE-89 SQL Injection Improper Input Neutralization
- CWE-120 Buffer Copy without Checking Size of Input
- CWE-129 Array Index Improper Input Neutralization
- CWE-134 Format String Improper Input Neutralization
- CWE-252 Unchecked Return Parameter of Control Element Accessing Resource
- CWE-327 Broken or Risky Cryptographic Algorithm Usage
- CWE-396 Declaration of Catch for Generic Exception
- CWE-397 Declaration of Throws for Generic Exception
- CWE-434 File Upload Improper Input Neutralization
- CWE-456 Storable and Member Data Element Missing Initialization
- CWE-606 Unchecked Input for Loop Condition
- CWE-667 Shared Resource Improper Locking
- CWE-672 Expired or Released Resource Usage
- CWE-681 Numeric Types Incorrect Conversion
- CWE-706 Name or Reference Resolution Improper Input Neutralization
- CWE-772 Missing Release of Resource after Effective Lifetime
- CWE-789 Uncontrolled Memory Allocation
- CWE-798 Hard-Coded Credentials Usage for Remote Authentication
- CWE-835 Loop with Unreachable Exit Condition ('Infinite Loop')

Measure Security Risk

- CISQ provides standard security measures for reliable, consistent measurement and insight to management.
- Recommendations for secure software development:
  - Measure and trend level of software security
  - Provide benchmarks to industry
  - Sourcing governance
  - Estimate the security debt of critical applications
- Compliance to Standards



Measuring the software supply chain

Measuring the software supply chain

- 10 out of the top application service providers use CISQ internally for ADM (Application Development and Maintenance) measurement and industrialization
- The main sourcing partners will recognize and appreciate CISQ analytics

Deploying latest ADM standards

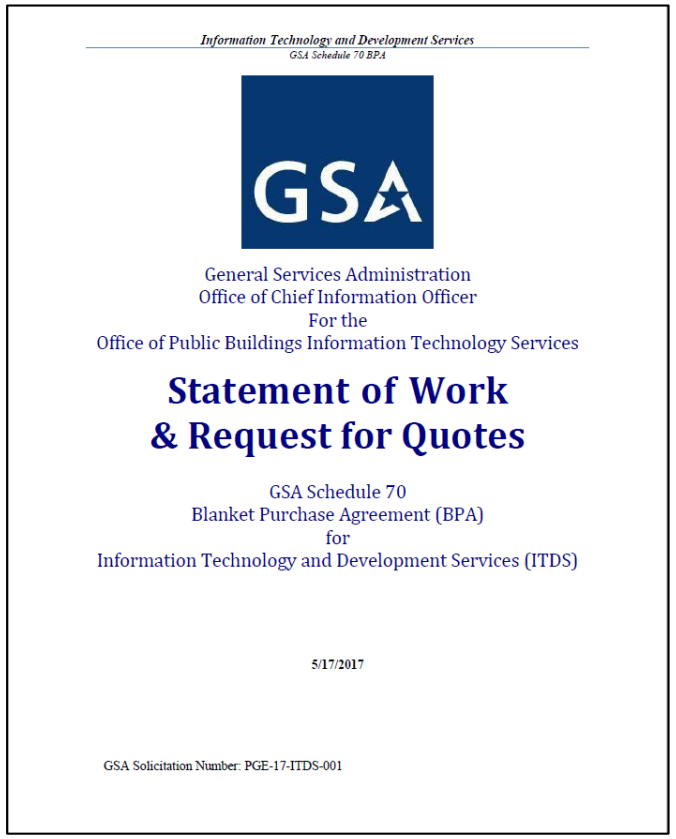
- OMG Automated Function Points Spec
- OMG System Level Guidance
- OMG Automated Quality Characteristics Spec
  - Reliability, Performance Efficiency, Security, and Maintainability

ADM vendor management based on outcomes can be deployed in five steps:

1. RFPs
2. Scorecarding
3. Policy
4. SLAs
5. Acceptance Criteria

Standard, objective measurement creates visibility

Sample RFP



Information Technology and Development Services
GSA Schedule 70 BPA

GSA

General Services Administration
Office of Chief Information Officer
For the
Office of Public Buildings Information Technology Services

Statement of Work
& Request for Quotes

GSA Schedule 70
Blanket Purchase Agreement (BPA)
for
Information Technology and Development Services (ITDS)

5/17/2017

GSA Solicitation Number: PGE-17-ITDS-001

CISQ has been referenced by the U.S. General Services Administration (GSA), formally citing CISQ requirements in a Information Technology (IT) statement of work from the Office of the CIO for the Office of Public Buildings. GSA is an independent agency of the U.S. government that supports general services of Federal agencies.

See page 21, section 5.9 in GSA's document, Schedule 70 Blank Purchase Agreement for IT and Development Services…

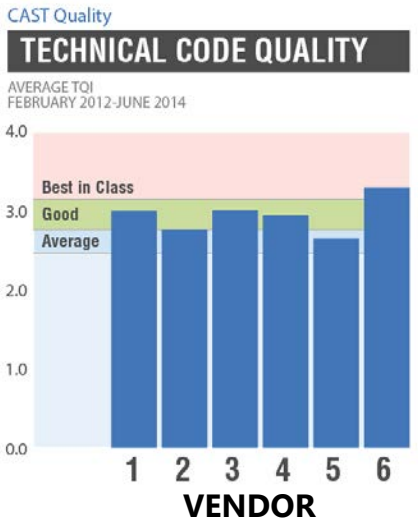
"PB-ITS (Project Based IT Services) is seeking to establish code quality standards for its existing code base, as well as new development tasks. As an emerging standard, PB-ITS references the Consortium for IT Software Quality (CISQ) for guidance on how to measure, evaluate and improve software."

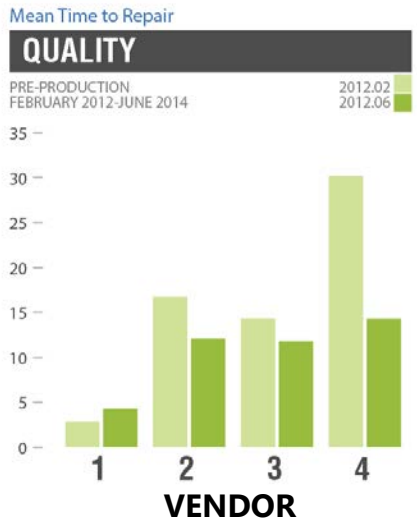
Sample vendor scorecard

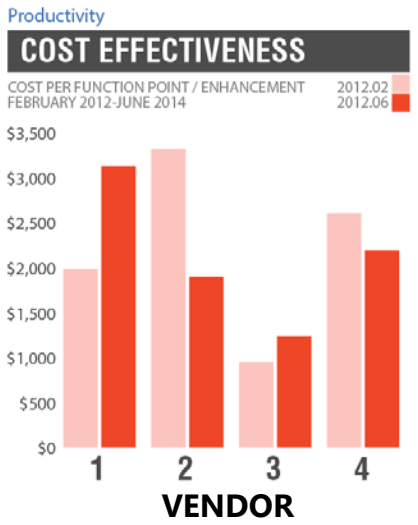
Scorecard Service Providers

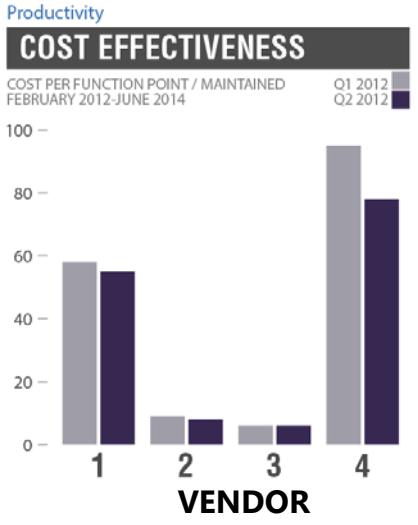
Outsourcer	CISQ-86	Reliability	Performance Efficiency	Security	Maintainability
VENDOR 1	2.59	3.16	2.34	3.01	1.99
VENDOR 2	2.81	2.78	2.78	3.12	2.34
VENDOR 3	2.59	1.67	3.54	2.98	1.76
VENDOR 4	3.06	3.12	3.11	2.79	3.11
VENDOR 5	2.83	2.56	2.88	3.03	2.56
VENDOR 6	2.90	3.76	2.89	2.97	2.55

Monitor Performance Over Time









Sample Critical Service Level Matrix

At Risk Amount and Allocation of Risk

Total Billing Per Release : $1,000,000
Total At Risk Amount (10% of Bill) : $100,000
Total Risk Pooler: 100%

10% is for example

Application Name	Tier 1 Metrics (Critical Service Levels)	At Risk Multiplier	Risk Allocation	At Risk Amount
OMS			30%	
	Security Findings	50%		$15,000
	Reliability Findings	30%		$9,000
	Application Pain Violations	20%		$6,000
		100%		$30,000
CRM			10%	
	Security Findings	30%		$3,000
	Reliability Findings	30%		$3,000
	Application Pain Violations	40%		$4,000
		100%		$10,000
AMSS			20%	
	Security Findings	50%		$10,000
	Reliability Findings	30%		$6,000
	Application Pain Violations	20%		$4,000
		100%		$20,000
SDP			20%	
	Security Findings	50%		$10,000
	Reliability Findings	30%		$6,000
	Application Pain Violations	20%		$4,000
		100%		$20,000
Enabler			20%	
	Security Findings	50%		$10,000
	Reliability Findings	30%		$6,000
	Application Pain Violations	20%		$4,000
		100%		$20,000

Amount service provider has at risk on this individual Service Level is 30% * 50% * $100K = $15,000

Any time there is a default, the at-risk amount will be applied

Incentive is given to service provide equivalent to the at risk amount if they exceed the Expected Service Level by 5% of the delta between the then current Expected and Perfection

Credits / Incentives are settled at the Annual Reset

Software supply chain resources

Website area for Vendor Management use case

- http://it-cisq.org/vendor-management/

Whitepaper about the concept of using CISQ metrics in SLAs

- http://it-cisq.org/wp-content/uploads/2015/07/Using-Software-Measurement-in-SLAs-Integrating-CISQ-Size-and-Structural-Quality-Measures-into-Contractual-Relationships.pdf

Whitepaper with detailed step-by-step instructions for putting CISQ metrics in SLAs

- http://it-cisq.org/wp-content/uploads/2017/04/CISQ-Rec-Guide-Effective-Software-Quality-Metrics-for-ADM-Service-Level-Agreements.pdf

Sample acceptance criteria using CISQ metrics

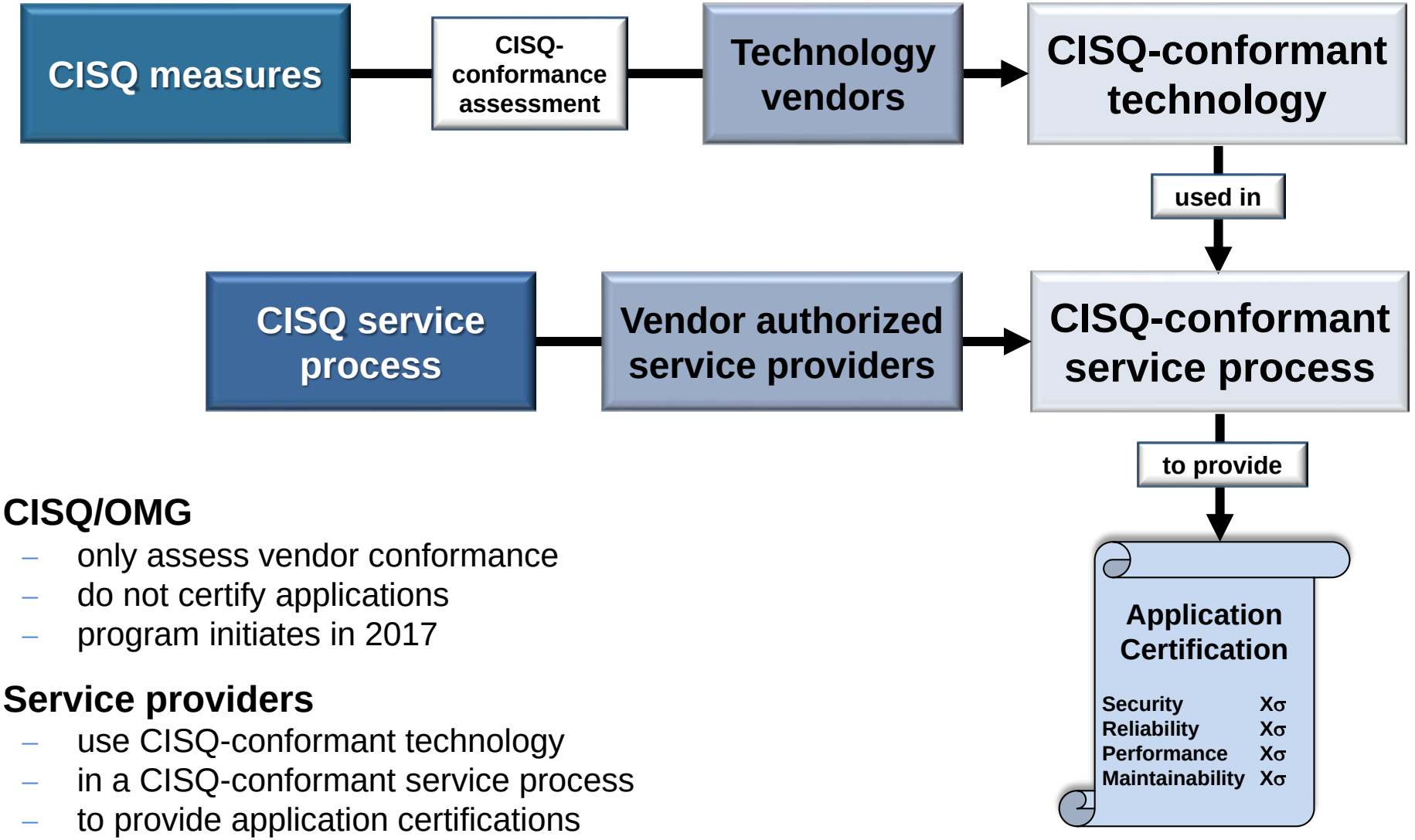
- http://it-cisq.org/wp-content/uploads/2017/06/Sample-Acceptance-Criteria-with-CISQ-Standardized-Metrics.pdf

Sample RFP from U.S. General Services Administration (GSA) that uses CISQ as part of it's requirement for quality software

- http://it-cisq.org/wp-content/uploads/2017/06/ITDSBPASOWFINALV420170517.pdf
- Go to section 5.9, page 21 of 73

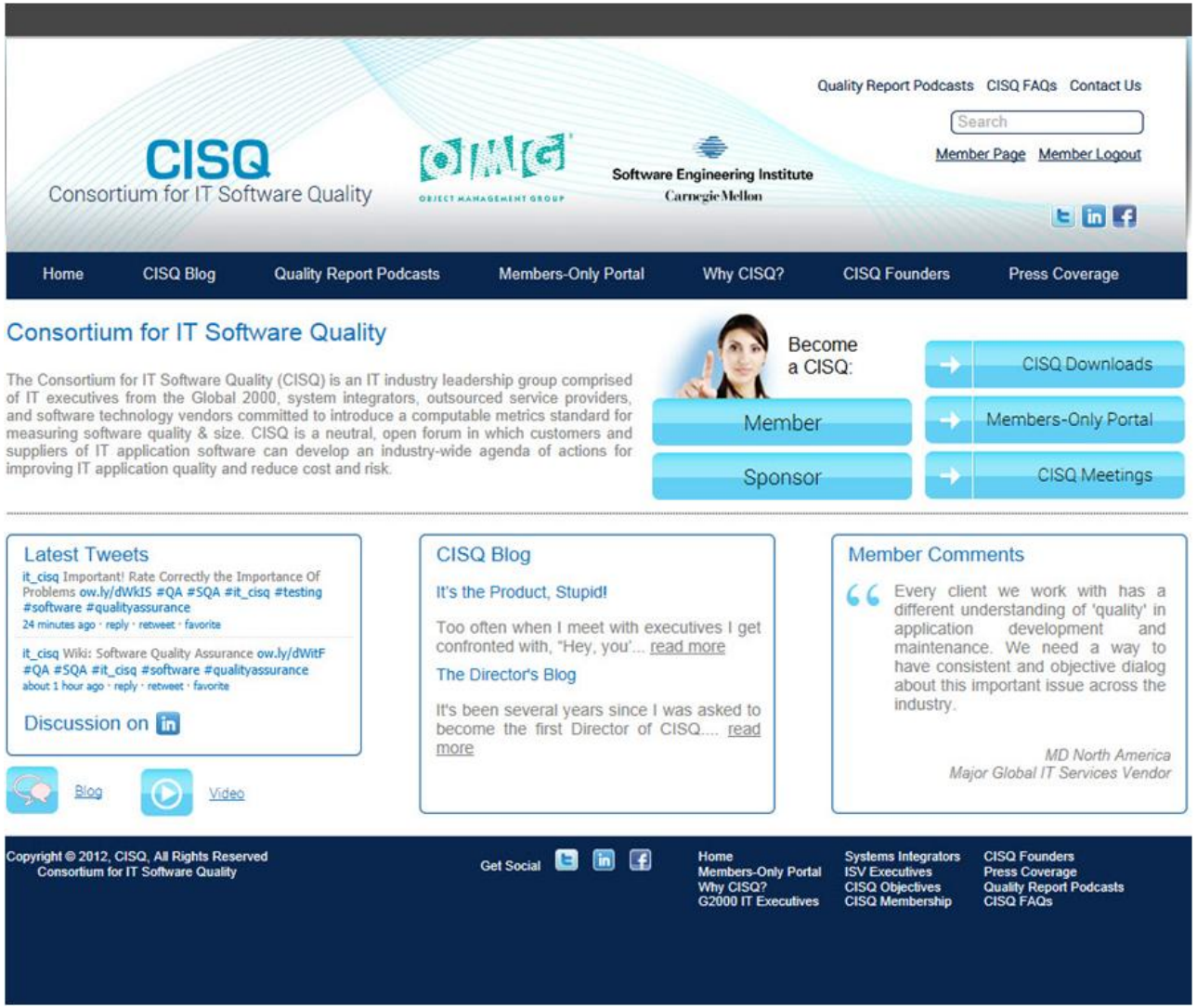
Certifying software

Certifying software



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

Come Learn about CISQ



More on CISQ: http://it-cisq.org/

More on OMG: http://www.omg.org/

