How to Address Both the Technical and Management Issues in Developing Software Which is Secure from Cyber-attacks – A Case Study

NDIA

2018 Agile in Government

Girish Seshagiri

20535

"I cannot define the real problem, therefore I suspect there's no real problem, but I'm not sure there's no real problem."

Richard Feynman

Topics

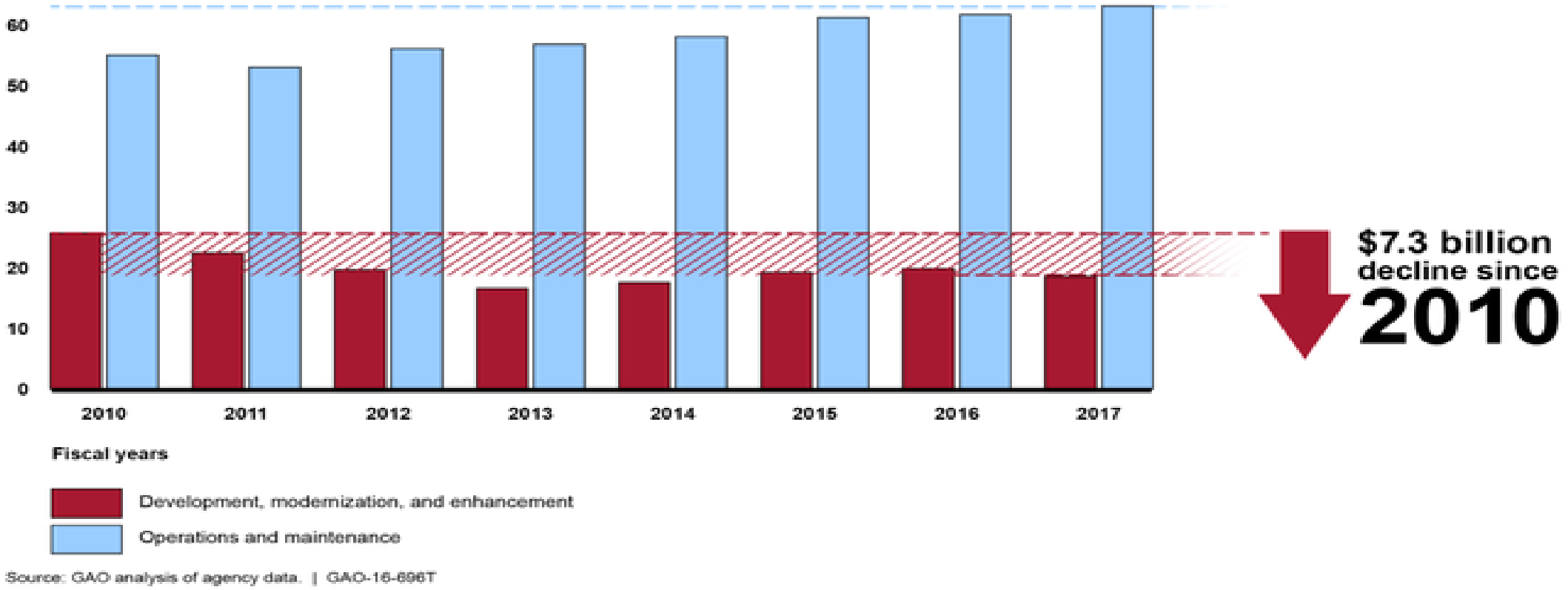
- IT Challenges Faced by the Government
- Challenges in the Agile Manifesto
- Software Engineering and Cybersecurity
- The Immutable Laws of Software Development and Cognitive Dissonance
- Managing Secure Software Development
- Case Study

Federal IT Spend in 2015

The **federal** government **spent** more than 75 percent of the total amount budgeted for information technology (IT) for fiscal year 2015 on operations and maintenance (**O&M**) investments. Specifically, 5,233 of the government's approximately 7,000 IT investments are **spending** all of their funds on **O&M** activities.

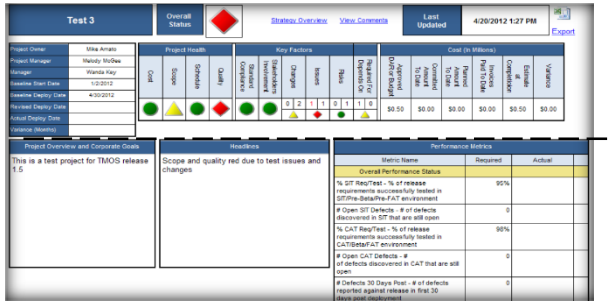
Source: https://www.gao.gov/assets/680/677436.pdf

Federal IT Spend in 2017



Agile Project Planning and Tracking

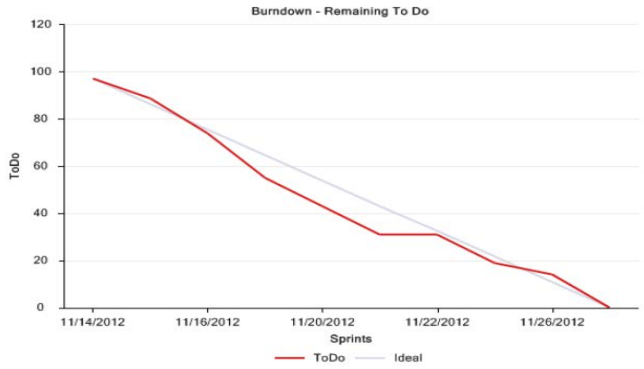
Metrics



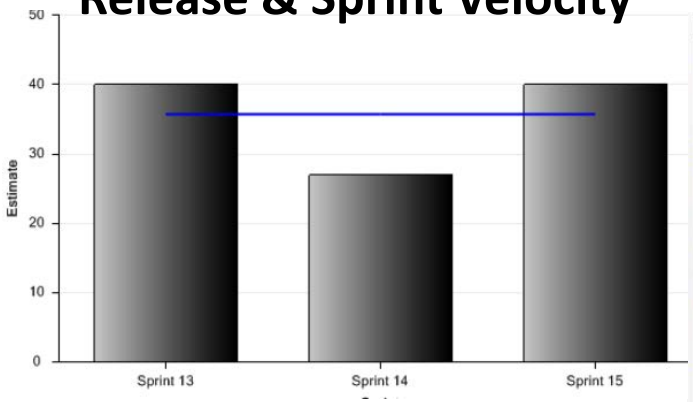
Agile Milestone Tracking

Scorecard Indicator	WBS	Task Name	Baseline Start	Baseline Finish	%	Planned Duration	Planned Start	Planned Finis
	1	Sample Agile Project Plan Template	NA	NA	0%	91 days?	Fri 10/21/11	Fri 2/
Initiate and Plan	1.1	Initiate and Plan	NA	NA	0%	10 days	Fri 10/21/11	Thu 1
Sprint 0	1.2	Sprint 0 - Analysis and Design	NA	NA	0%	10 days?	Mon 10/24/11	Fri 11
Development Sprints	1.3	Development - Sprint 1 through n	NA	NA	0%	30 days?	Mon 11/21/11	Fri 12/
SIT	1.4	Systems Integration Test	NA	NA	0%	1 day?	Mon 10/24/11	Mon 1(
CAT	1.5	Customer Acceptance Test	NA	NA	0%	1 day?	Mon 10/24/11	Mon 1(
Governance	1.6	Governance	NA	NA	0%	1 day?	Mon 10/24/11	Mon 1(
Release	1.7	Release	NA	NA	0%	1 day?	Mon 10/24/11	Mon 1(

Daily Burndown



Release & Sprint Velocity



Sprint Task board

CISQ Consortium for IT Software Quality

Challenges in the Agile Manifesto

Individuals and interactions over processes and tools

1. Occasionally becomes an excuse for undisciplined work
2. Agile methods are processes—shorter time scales require greater discipline
3. Agile and DevOps cannot be executed without tools chains
4. Agile focuses on teams (tribal) rather than on organizational capability

Working software over comprehensive documentation

1. Beware the temptation to short-circuit architecture and design
2. Bad architectures cannot be refactored

Customer collaboration over contract negotiation

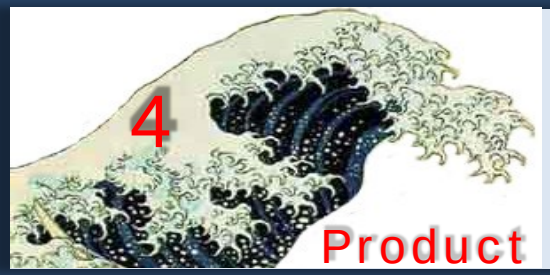
1. Most often achieved when both are mature

Responding to change over following a plan

1. Control of commitments is critical for dependable, trustworthy software
2. Customers must own responsibility for controlling change

CISQ
Consortium for IT Software Quality

4th Wave in Software Engineering



What: Architecture, Structural measures, Reuse

When: 2002➔

Why: Improve engineering of software products

What: CMM, ITIL, PMBOK, Agile

When: 1990-2002

Why: Improve software management and discipline

What: Design methods, CASE tools

When: 1980-1990

Why: Give developers better aids to construct systems

What: 3rd & 4th generation languages, structured programming

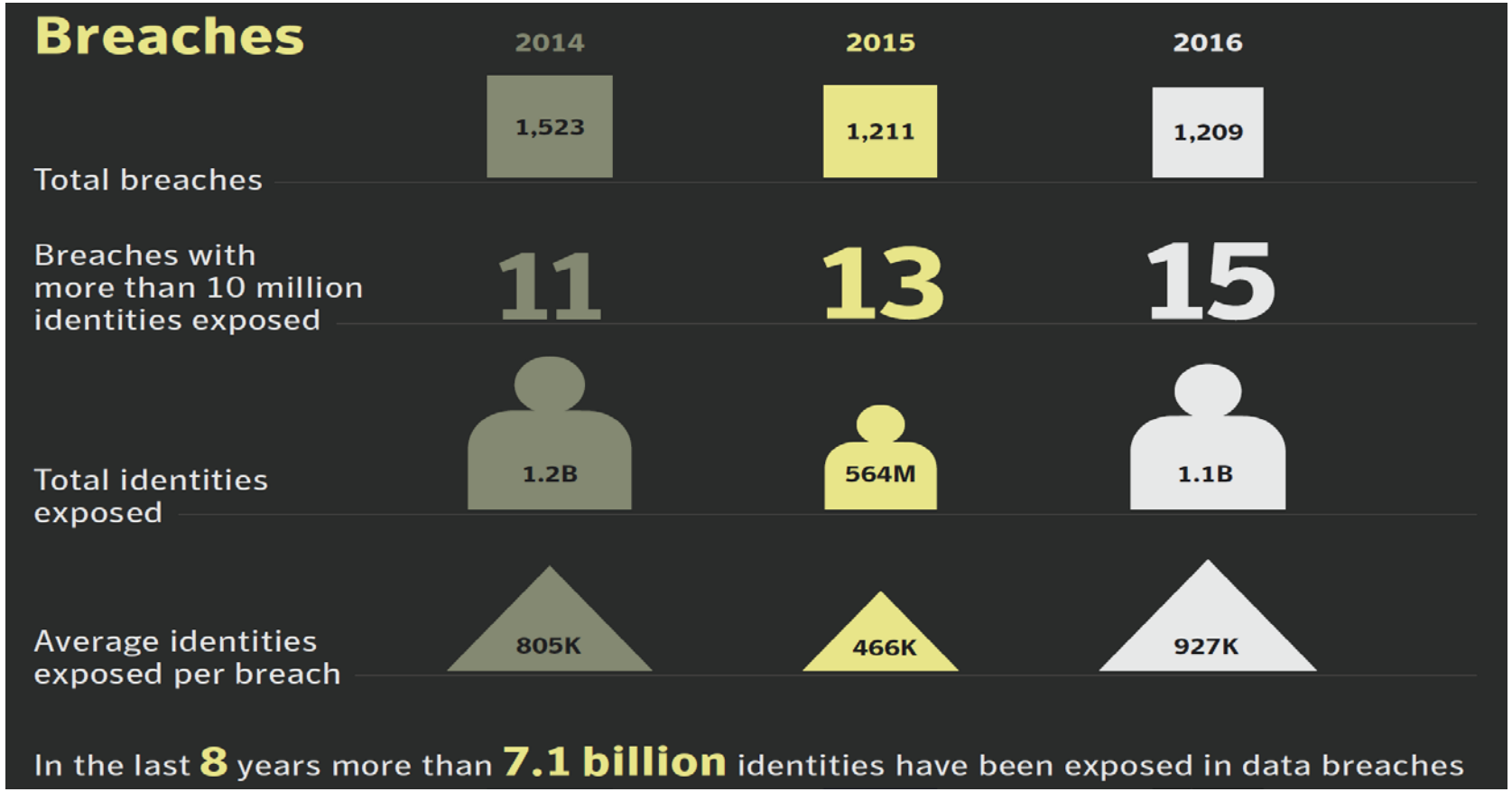
When: 1965-1980

Why: Give developers greater power for expressing programs

Software Engineering's Persistent Problems

- Exponential rise in cybersecurity vulnerabilities due to defective software
- Unacceptable cost, schedule, and quality performance of Enterprise Resource Planning (ERP) and legacy systems modernization projects
- The number one cost driver in software projects – cost of finding and fixing bugs (i.e. scrap and rework)
- Arbitrary and unrealistic schedules leading to a culture of "deliver now, fix later"
- Absence of work place democracy and joy in work

Personal Identity Breaches

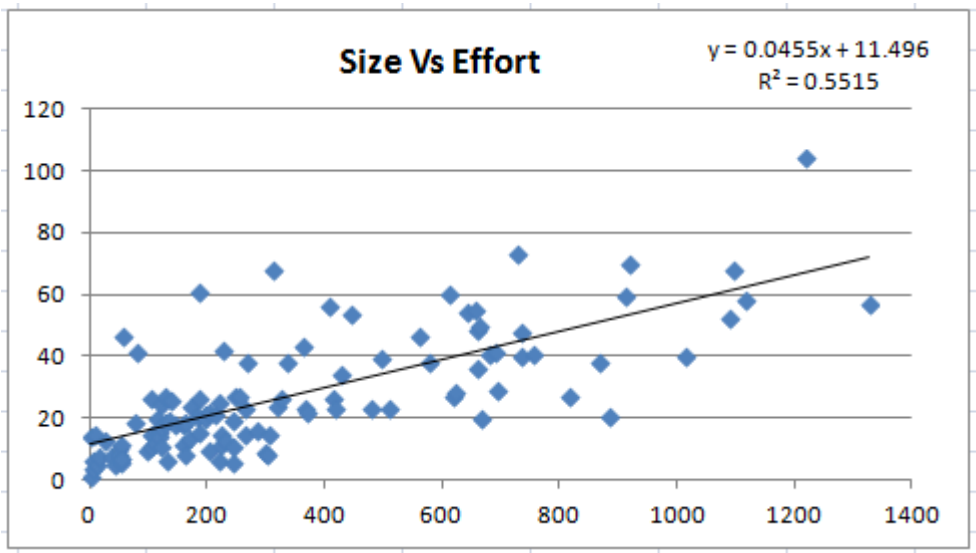


Cybersecurity

- **Defective software** is insecure
  - 90% of attacks are successful by exploiting defects in the software application layer
  - 1 in 20 software defects are vulnerabilities that can be exploited to launch cyberattacks
  - "If you have a quality problem, you have a security problem"
- **Consequences of poor quality software**
  - Impacts - Democracy, loss of life and limb besides just financial loss
  - Potentially more catastrophic than bridge falling down
- Cannot **rely on testing alone** to find and remove software defects
  - Common misconception – "if it passes test, it must be OK"
  - Root cause of "Deliver now, Fix later" culture, technical debt, increase in total ownership cost in many agile projects
- **Reducing vulnerabilities** - number one goal for every agile software team
- High priority national goal to move from reactive to proactive – **from threat detection to threat prevention**

Immutable Laws of Software Development – 1

The number of development hours will be directly proportional to the size of the software product



Immutable Laws of Software Development – 2

When acquirers and vendors both "guess" as to how long a project should take, the acquirers' "guess" will always win

Customers' Dilemma

- Want their product now at zero cost.
- Due to time-to-market pressures, time frames are arbitrary and unrealistic for the software team to produce a product that works.

Developers' Choices

- Try to "guess" what it would take to win the business.
- Or make a commitment based on a plan and what the organization can do based on organization historic data.

Immutable Laws of Software Development – 3

When management compresses schedule arbitrarily, the project will end up taking longer

Schedule/Quality Trade-off				
	Default	10% Compression	20% Compression	10% Extension
Duration Mths	25.9	23.3	20.7	28.5
Defect Count	1,033	1,316	1,715	849
% Change		27.4%	66.0%	-17.8%

Cognitive Dissonance

The term cognitive dissonance is used to describe the *feelings of discomfort* that result from holding two conflicting beliefs.

When there is an inconsistency between beliefs and behaviors, something must change in order to eliminate or reduce the dissonance.

Cognitive dissonance often has a powerful influence on our behaviors and actions.

Commitment Example

- **Belief:** Manager believes that if the team says it is impossible to meet the customer's arbitrary and unrealistic delivery deadline, it probably is.

- **Behavior:** Manager commits the team to the customer's arbitrary and unrealistic deadline knowing that the team will not meet it.

- **Strategy to reduce or minimize dissonance:** Focus on belief that offering monetary incentives, paid overtime etc., will improve productivity to meet the committed deadline. This belief outweighs the dissonant belief or behavior.

Transformation Goal

Combine high maturity CMMI Maturity Level 5 with agile team process to create software which is secure from cyberattacks by self-managed teams and trained, certified, and empowered developers

Transformation Principles

- Adopted from Dr. Deming's fourteen key principles which revolutionized manufacturing

1. Improve constantly and forever the system of production and service to improve quality, reduce vulnerabilities, minimize technical debt, and lower total cost (Development + O&M)
2. Compete on the basis of guaranteed quality and proven track record of lowest cost (Development +O&M)
3. Cease dependence on test and fix (rework) cycles to achieve quality
4. Institute training on the job
5. Institute leadership – Job of team leads and managers should be to help developers perform their best
6. Drive out fear by empowering developers to negotiate realistic and aggressive cost, schedule, and quality commitments instead of arbitrary and unrealistic schedules that all know will not be met
7. Remove barriers that rob developers and team leads the right to pride of workmanship

Managing for Secure Software Development – 1

- Develop workforce formally trained in disciplined software practices including secure coding, estimation, design, and measuring and managing quality
- Start each project right with a jelled team capable of making disciplined commitments that the team knows it can meet
- Proactively create work product standards, guidelines, design patterns, and checklists and consistently apply them
- Support the teams in collecting, analyzing, and reporting product and process data—size, effort, schedule, and quality
- Systematically apply secure coding practices and verify that they have been followed

Managing for Secure Software Development – 2

- Continuously incorporate best practices from industry sources such as CISQ, OWASP, (ISC)^2, and CWE/SANS
- Motivate developers to put the highest-quality code components into test, striving for 90 – 100% defect-free components with zero cybersecurity vulnerabilities
- Require teams to report status weekly
  - Precise and accurate data on planned versus actual size, effort, schedule, earned value and defects injected and removed

Decision Process to Reduce Vulnerabilities

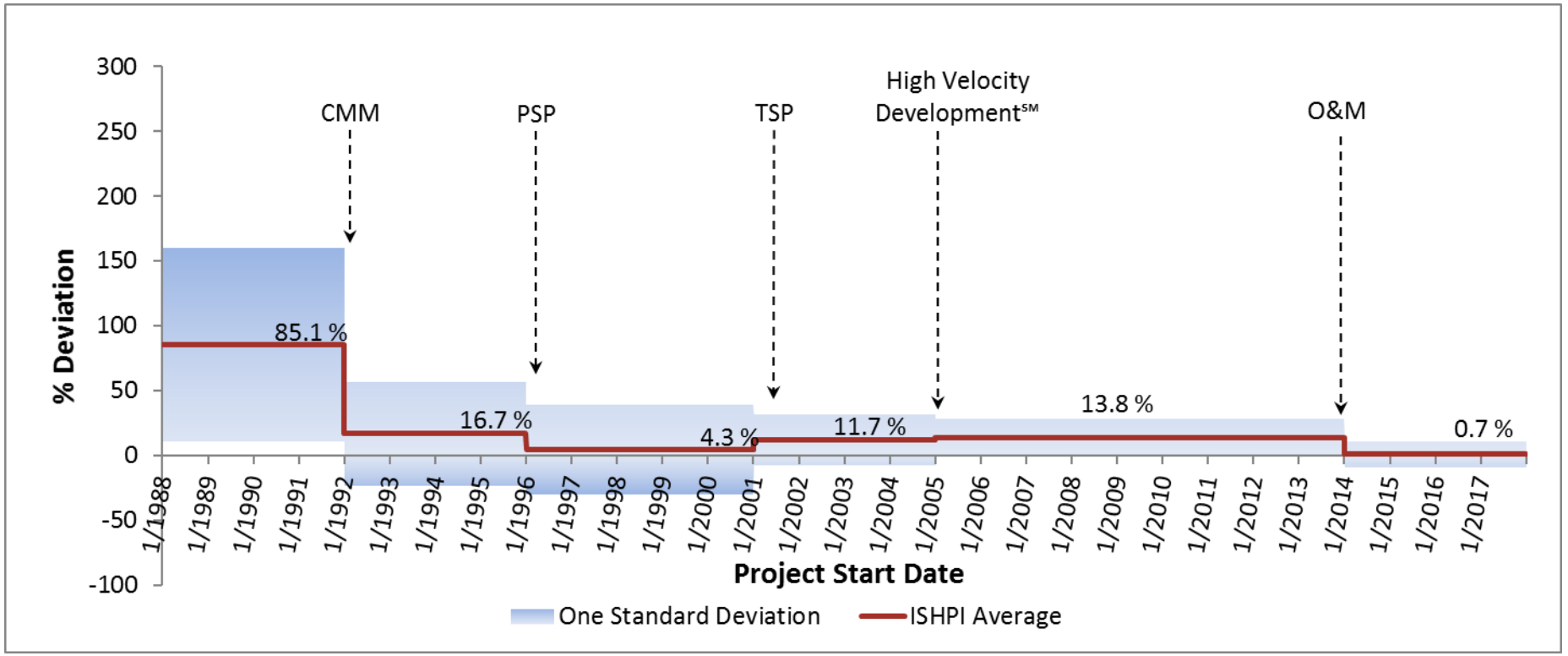
- Size, effort, schedule deviation
  - Review process adherence and test and rework due to poor quality
- Earned value deviation
  - Review hours worked against plan, task dependencies
  - "What if" scenario using process performance model
- Planned versus actual defect profile
  - Review personal and peer review yields – design, code
  - Estimate defect counts and defect densities for downstream steps using quality projection model

Customer Benefits of Optimizing Process and Agile Development

- Frequent incremental deliveries that meet customer business goals
- Dramatically reduces and practically eliminates cyber security incidents attributable to poor quality software code
- Reduces software operations & maintenance (O&M) costs by more than half
- Fewer bugs to fix in production software making available a larger percentage of O&M spend for new features and enhancements
- Enables lifetime warranty

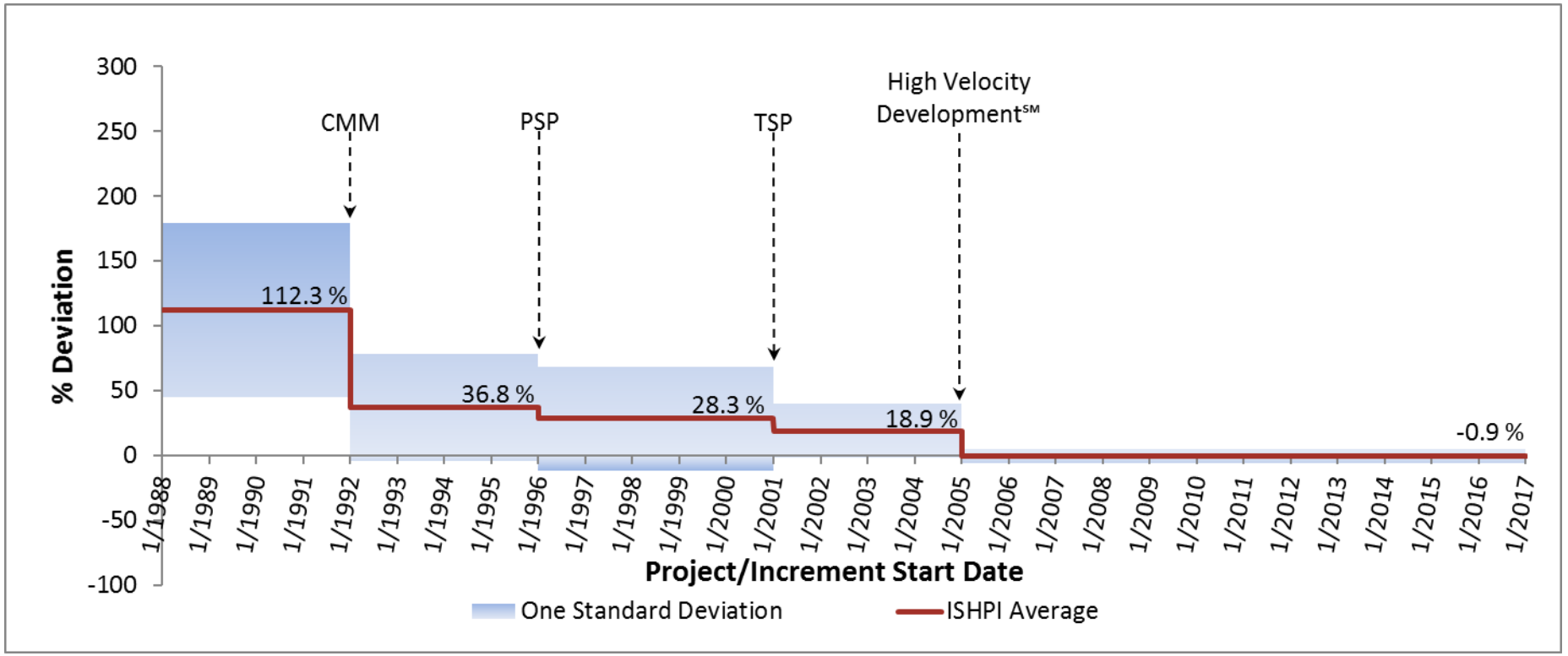
ISHPI Capabilities

Effort Deviation – Development Phases



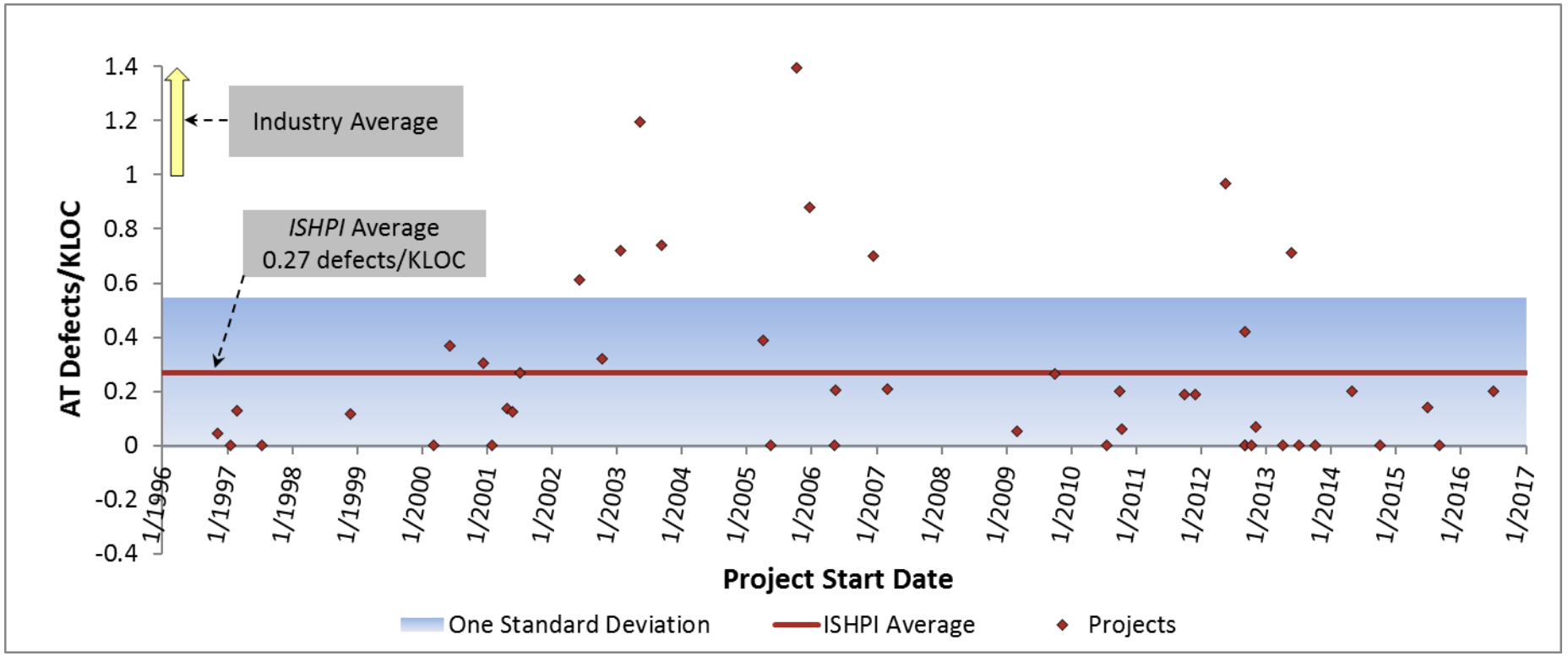
ISHPI Capabilities

Schedule Deviation – Development Phases



ISHPI Capabilities

Acceptance Test Defect Density



ISHPI Capabilities Summary

Industry, Best in Class, and *ISHPI*

Performance Metrics	Industry Average	*ISHPI* Average
Schedule deviation	> 50%	-0.9%
Acceptance test defects in delivered product (100,000 Source Lines of Code)	> 100	13.9
% of design and code inspected	< 100%	100%
Customer's time to accept 100,000 LOC product	> 4 months	< 5 weeks
% of defects removed prior to system test	< 60%	> 90%
% of development time in rework fixing system test defects	> 33%	< 10%
Cost of quality	> 50%	< 33%

Contact

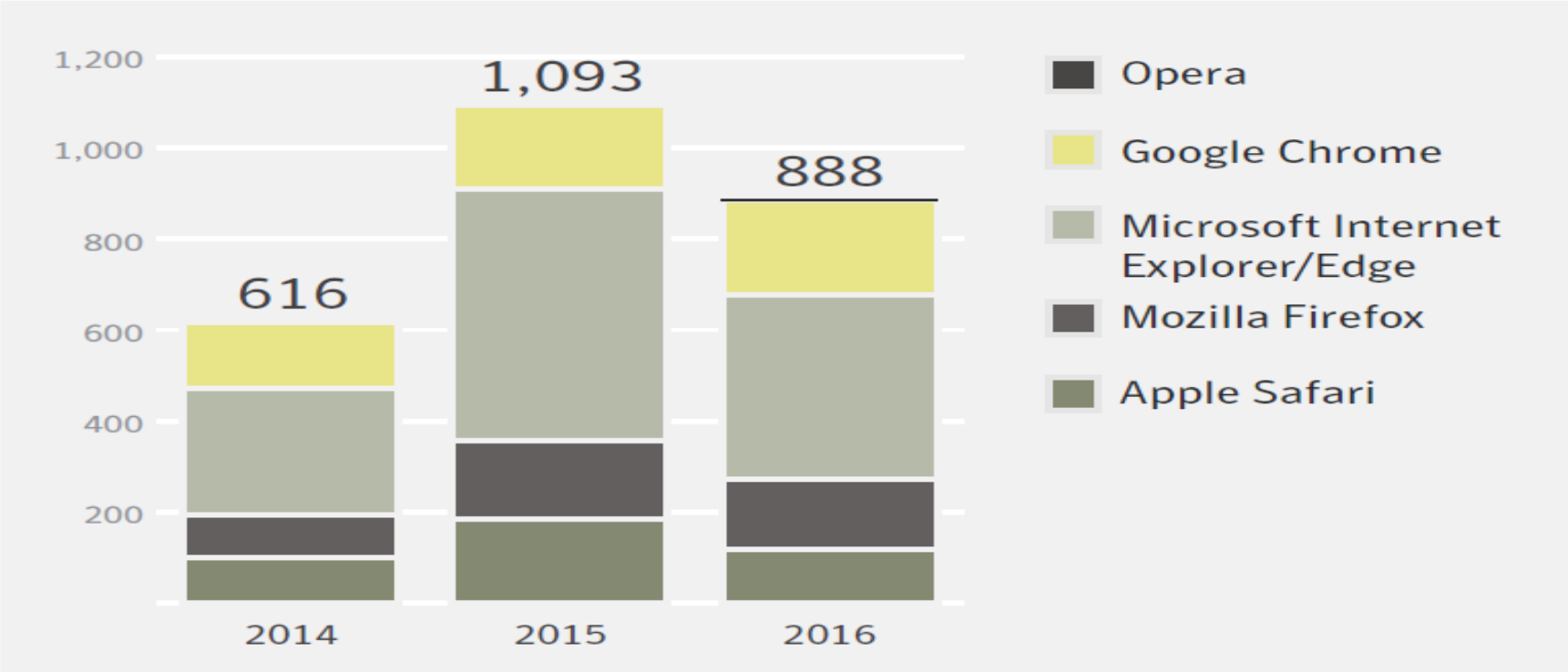
Girish Seshagiri

girish.seshagiri@ishpi.net

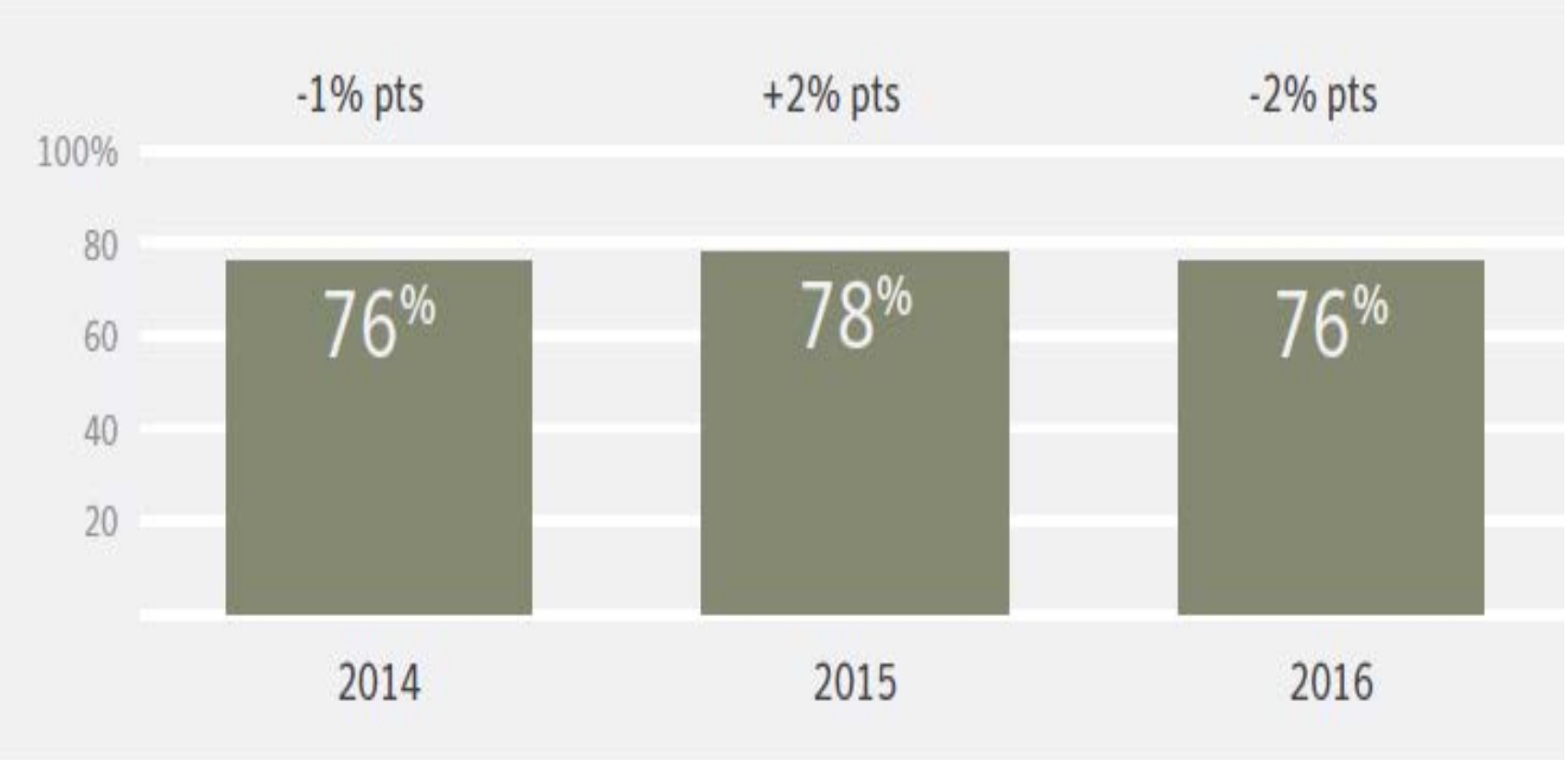
(703) 426-2790

Backup Material

Why We Are Here
Browser Vulnerabilities



Why We Are Here
Websites with Vulnerabilities



Patch and Pray - 1

Microsoft Patch Tuesday

On Tuesday, September 12, Microsoft released fixes for more than 80 security issues in multiple products, including Windows, Office, Microsoft .NET Framework, Flash, Internet Explorer, and Edge.

Patch and Pray - 2

Apache Struts Vulnerability Exploited in Equifax Breach

Equifax has acknowledged that the massive breach that exposed personal information of as many as 143 million people was due to a failure to apply a patch for a vulnerability in Apache Struts. A patch for the flaw was released on March 6, 2017. The Equifax breach occurred in "mid-May" 2017.

Patch and Pray - 3

Adobe Security Updates

Adobe has released updates to address security issues in Flash Player, ColdFusion, and RoboHelp for Windows. The Flash updates, available for Windows, Mac, Linux, and Chrome OS, address two critical memory corruption flaws. The ColdFusion update includes fixes for four flaws, and the RoboHelp update fixes two flaws

Patch and Pray - 4

Patches dominate the Top of the News this week.

Bill Murray speaks for many when he writes: "The cost to tolerate or remediate a design, recording, or coding error goes up exponentially with the time to its discovery. **There is something fundamentally wrong with an industry in which the toleration, indeed the institutionalization, of late discovery and remediation of error is as it is in ours."**

And John Pescatore offers a path toward fixing that fundamental flaw in the software industry, describing a future where larger buyers of software (government, for example, along with the Business Roundtable) set a much higher bar for application security testing at multiple stages of the development life cycle, both for custom software they develop and for every package they buy, with **substantial contractual penalties for vendors who fail**.

What is at Stake - 1

Some US States Are Going Back to Paper Ballots

In the wake of rising concerns about the security of electronic voting systems, several US states are returning to the use of paper ballots for their elections. Georgia will pilot a paper-ballot system in elections this fall.

FDA Approves Pacemaker Patch, Announces Recall of Abbott/St. Jude Medical Devices

The US Food and Drug Administration (FDA) has announced a recall of more than 450,000 pacemakers because they require a firmware update to address several security issues. The recall applies to several models of pacemakers manufactured by Abbott, formerly known as St. Jude Medical. Patients must visit their doctor's office where the update can be installed while the device is in backup mode. The flaws could be exploited to gain unauthorized access to vulnerable devices and issue commands to modify the pacemaker's settings and functionality.

What is at Stake - 2

Car Safety Vulnerability Lies in the Way CAN Handles Error Messages

A vulnerability in the Controller Area Network (CAN) that exists in most new automobiles could be exploited to shut down components of the car, including safety systems. Any component connected to the car's CAN bus could be affected. The issue is not one that can simply be patched because it lies in the CAN bus messaging protocol standard. Components that send too many error messages are disconnected from the CAN, so if attackers can spoof error messages to appear to be coming from a targeted component, that component could be shut off from the CAN.

What is at Stake -3

National Infrastructure Advisory Council Report - A Pre 9-11 Moment

"There is a narrow and fleeting window of opportunity before a watershed, 9/11-level cyber attack, [for the nation] to organize effectively and take bold action," said the US National Infrastructure Advisory Council report. The report lists 11 recommendations, including "establish separate, secure communications networks specifically designated for the most critical cyber networks; ... identify best-in-class scanning tools and assessment practices; ... [and] establish clear protocols to rapidly declassify cyber threat information."

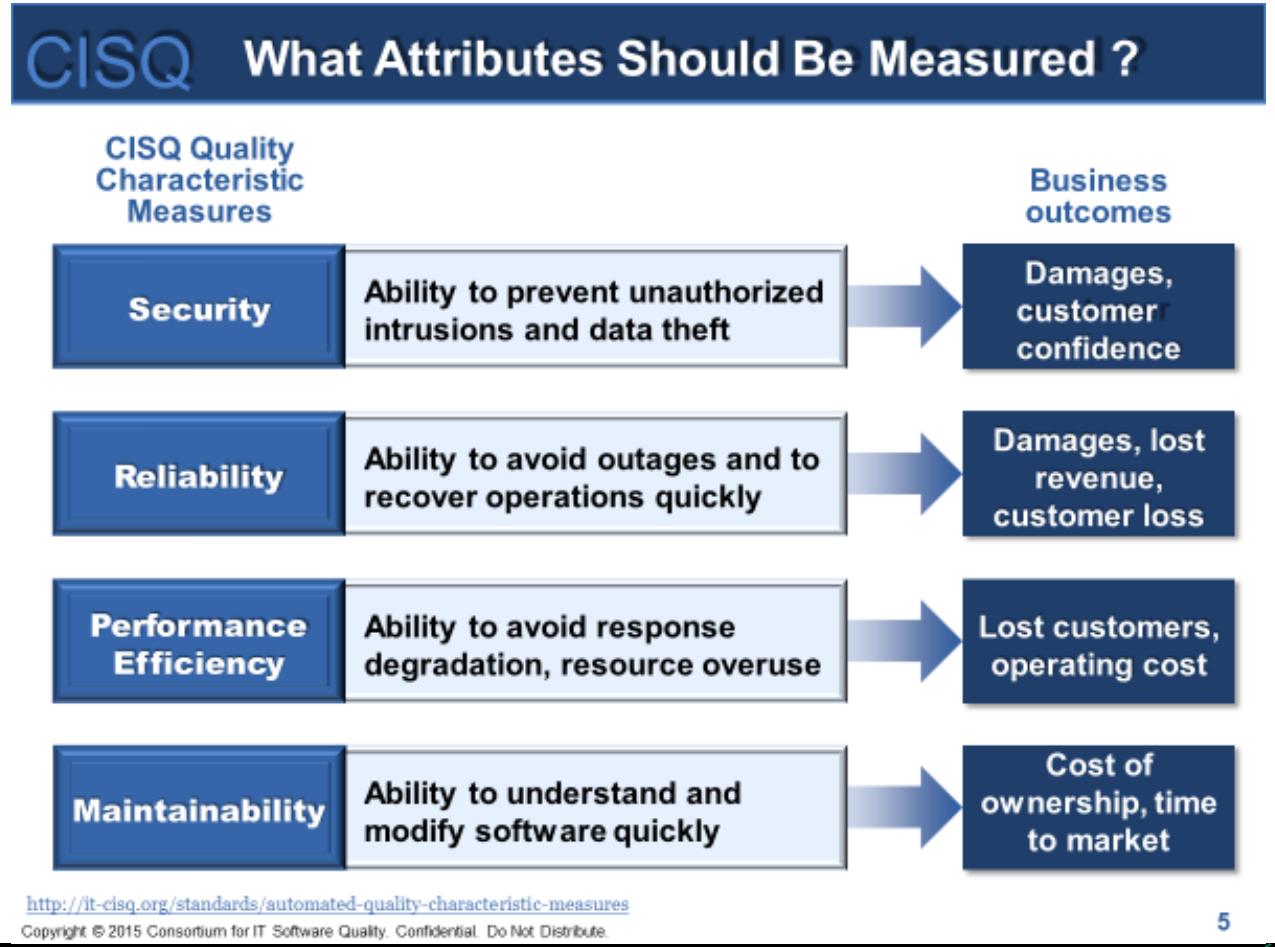
Unique Point in Time – 1

- Software is the most important technology contributing to global high standards of living
- Cyber-attacks are increasing; 90% of attacks result from exploiting defects in software
- The cybersecurity spend estimated to grow to $170 billion by 2020
- 50 billion devices coming online during the next 5 years (IOT)
- Government is dependent on legacy code which is insecure
- Government will have >30% staff turnover due to boomers retiring

Unique Point in Time – 2

- GAO has added the Management of Information Technology (IT) Acquisitions and Operations to high risk list
- Many agencies are convinced that agile/Scrum is the silver bullet solution
- Majority of the government's IT spend is for operations & maintenance; majority of the maintenance spend is for corrective maintenance
- There is no evidence that software project management capability has improved during the past 50 years

CISQ Quality Characteristic Measures



CISQ Quality Characteristic Violations