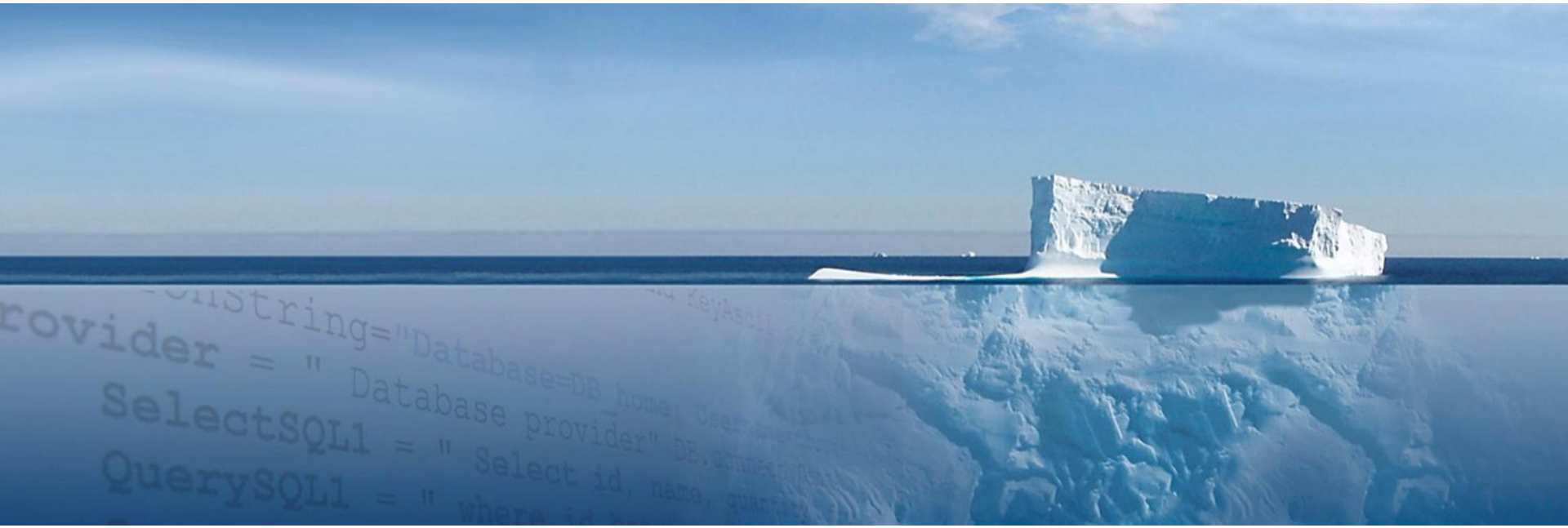
CAST

ACHIEVE INSIGHT. DELIVER EXCELLENCE.



The Global State of Structural Quality in Business Applications

Dr. Bill Curtis
Sr. VP & Chief Scientist, CAST Software
Director, Consortium for IT Software Quality (CISQ)

CEE–SEC(R)
November 1, 2012

Bad Structural Quality Is Expensive

© 2012 CAST

Testing is Not Enough

review articles

Who could fault an approach that offers greater credibility at reduced cost?

BY DANIEL JACKSON

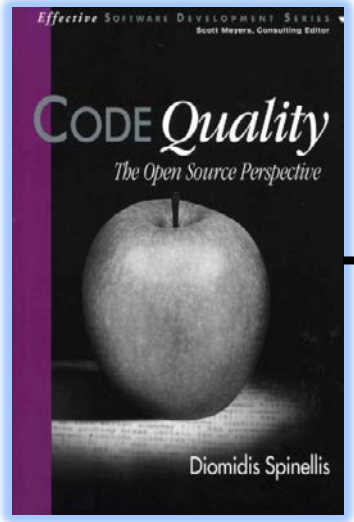
A Direct Path to Dependable Software

SOFTWARE PLAYS A fundamental role in our society, bringing enormous benefits to all fields. But because many of our current systems are highly centralized and tightly coupled, [1] we are also susceptible to massive and coordinated failure.

A Chicago hospital lost its entire pharmacy database one night, and it was only able to reconstruct medication records for its patients by collecting paper printouts from nurses' stations. In their report on this incident, [2] Richard Cook and Michael O'Connor concluded: "Accidents are signals sent from deep within the system about the sorts of vulnerability and potential for disaster that lie within." Similar signals have been sent, for example, in the fields of electronic voting, [3] air traffic control, [4] nuclear power, [5] and energy distribution. [6] The growing tendency to embed

"As higher levels of assurance are demanded…testing cannot deliver the level of confidence required at a reasonable cost."

"The correctness of the code is rarely the weakest link."



"…a failure to satisfy a non-functional requirement can be critical, even catastrophic…non-functional requirements are sometimes difficult to verify. We cannot write a test case to verify a system's reliability…The ability to associate code to non-functional properties can be a powerful weapon in a software engineer's arsenal."

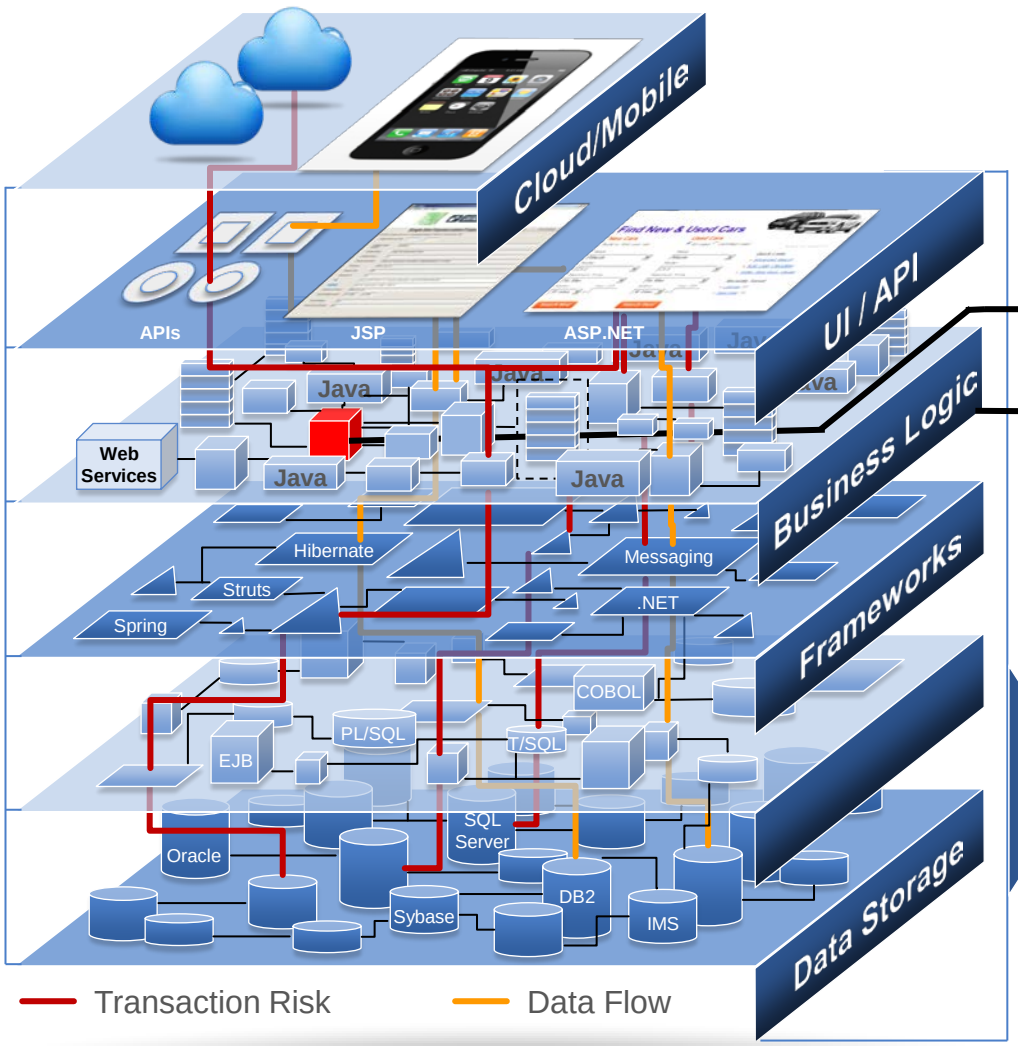
© 2012 CAST

Software Quality Iceberg

Source: Code Complete, Steve McConnell

© 2012 CAST

3 Levels of Structural Quality Analysis



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

© 2012 CAST

Software Engineering's 4th Wave

What: Architecture, Structural Quality, Reuse
When: 2002➔
Why: Ensure software is constructed to standards that meet its lifetime demands

What: CMM/CMMI, ITIL, PMBOK, Agile
When: 1990-2002
Why: Provide a more disciplined environment for professional work using best practices

What: Design methods, CASE tools
When: 1980-1990
Why: Give developers better methods and tools for constructing software systems

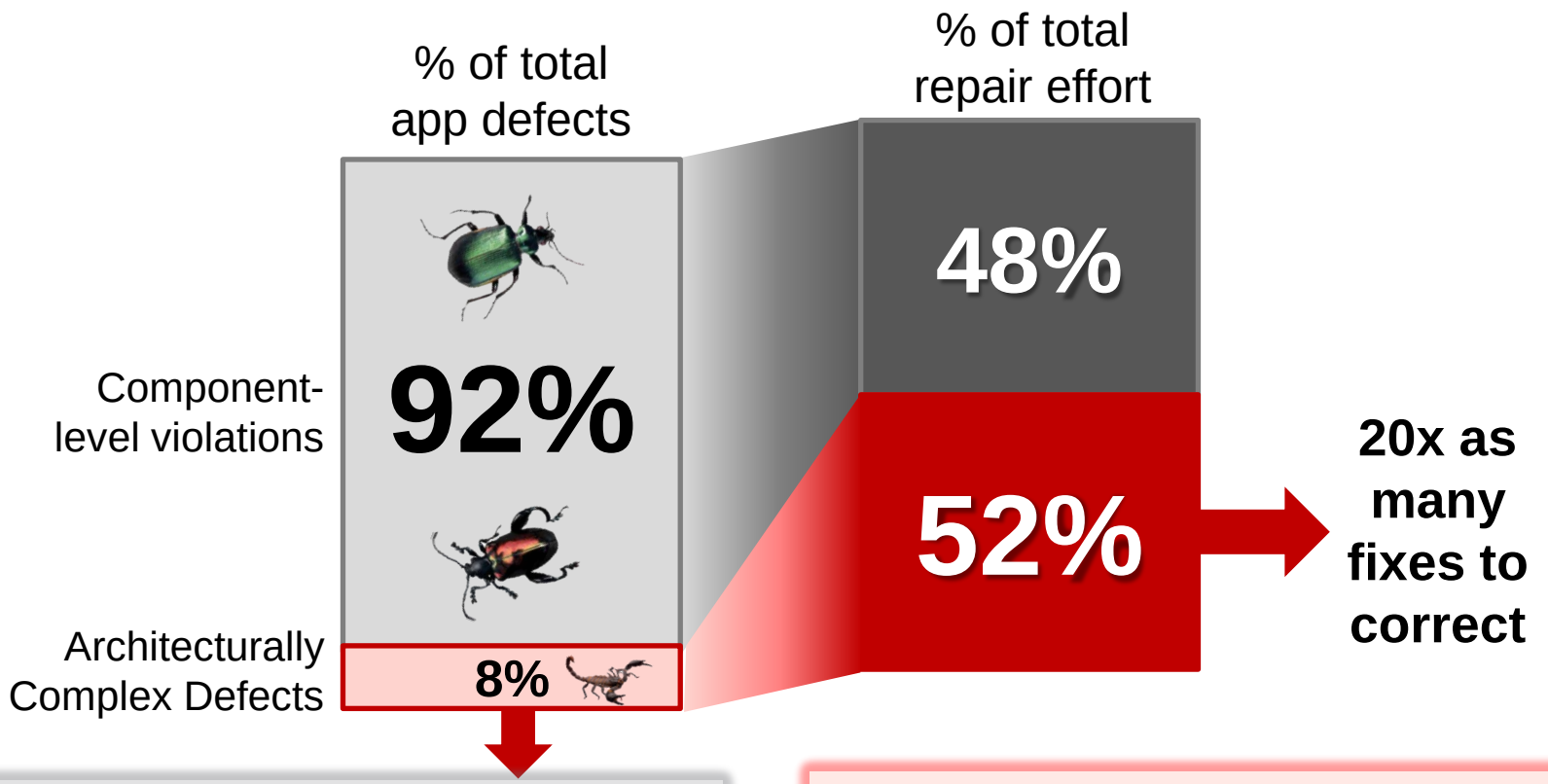
What: 3rd & 4th generation languages, structured programming
When: 1965-1980
Why: Give developers greater power for expressing and understanding their programs

© 2012 CAST

Architecturally Complex Defects

Architecturally Complex Defect

A structural flaw involving interactions among multiple components that reside in different application layers

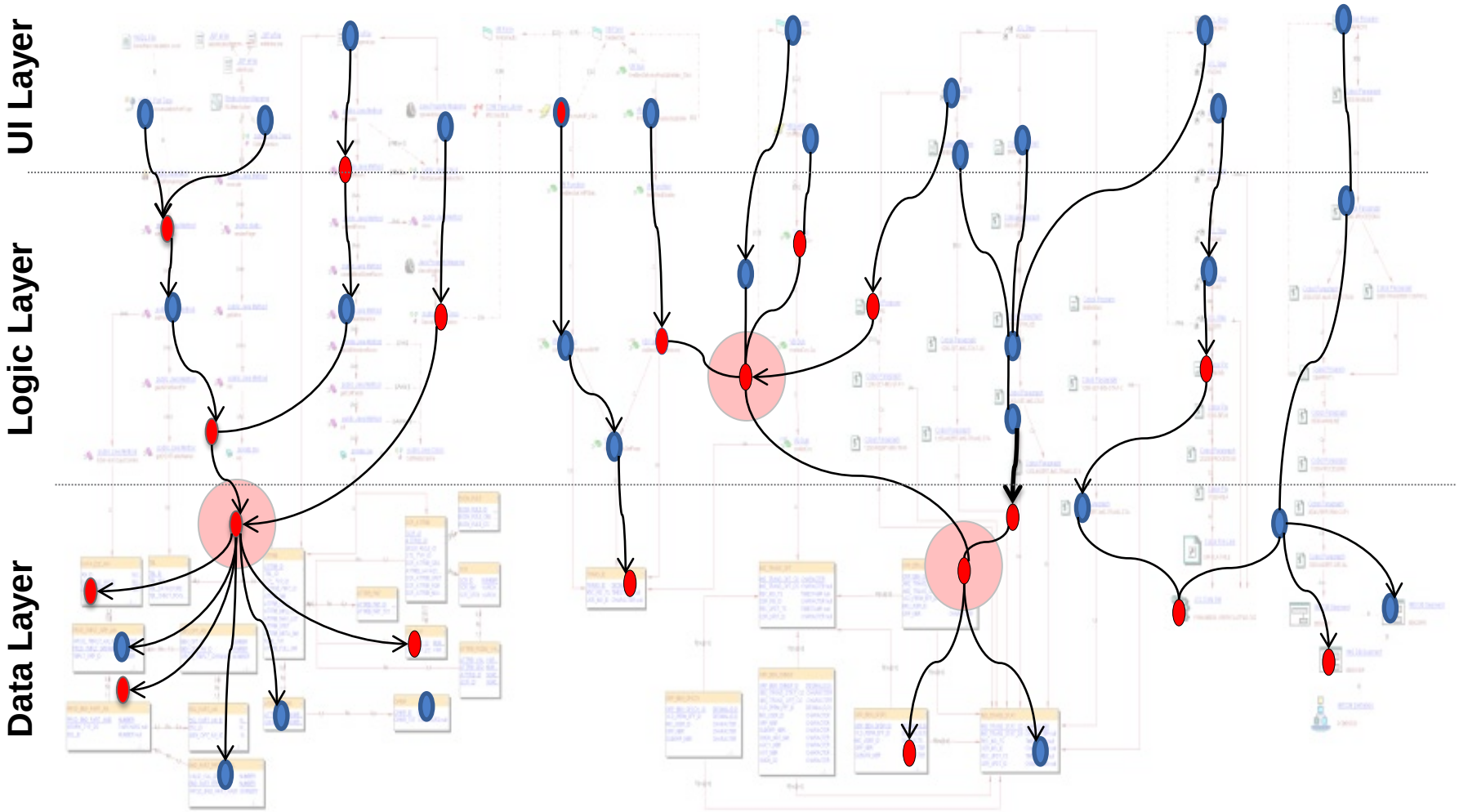


80% of architecturally complex defects touch an Architectural Hotspot—a badly designed component causing problems

Architectural hotspots provide a roadmap for remediating the worst risk, rework, and cost drivers

© 2012 CAST

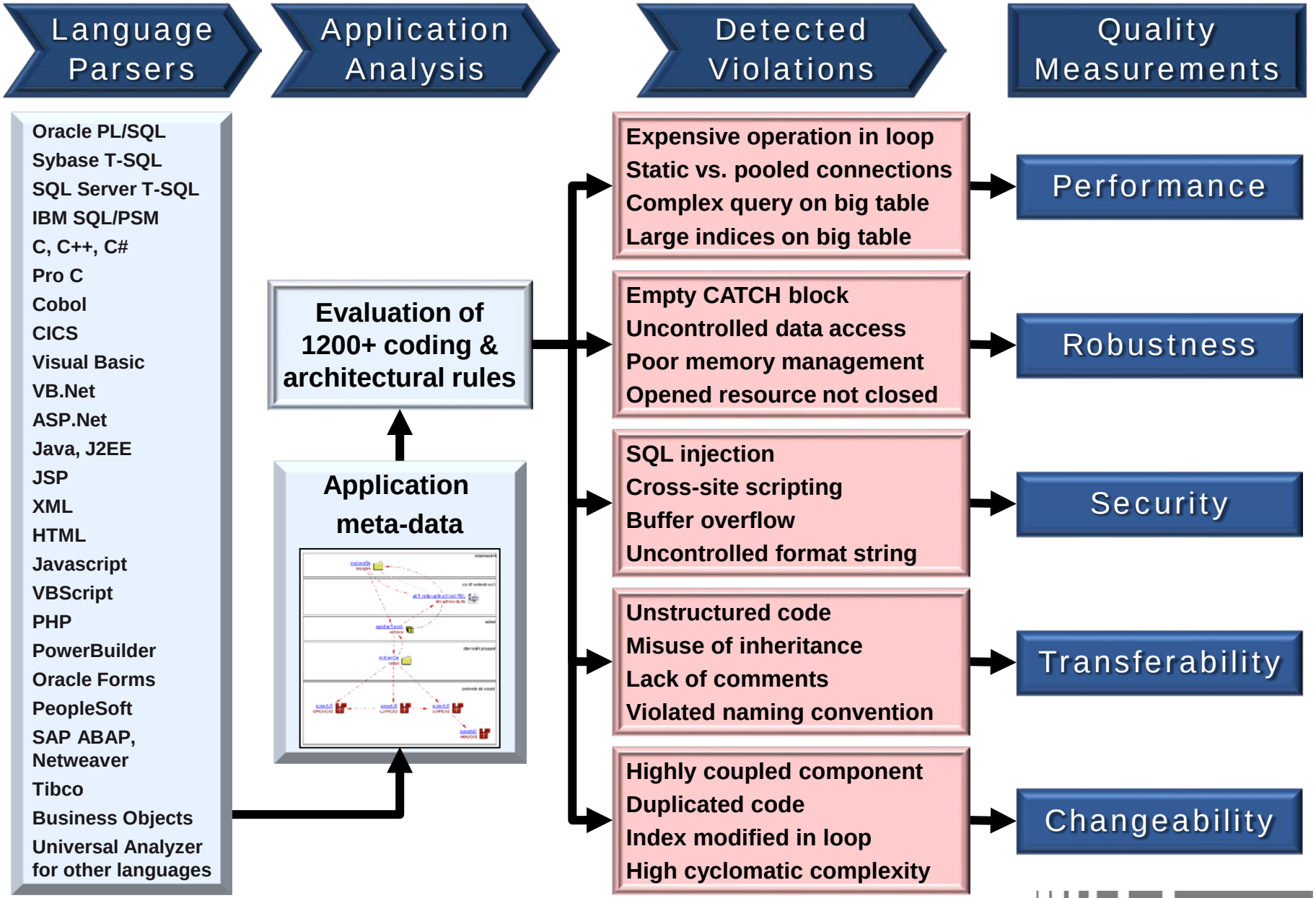
Detecting Architectural Hotspots



System Level analysis allows detection architectural hotspots

© 2012 CAST

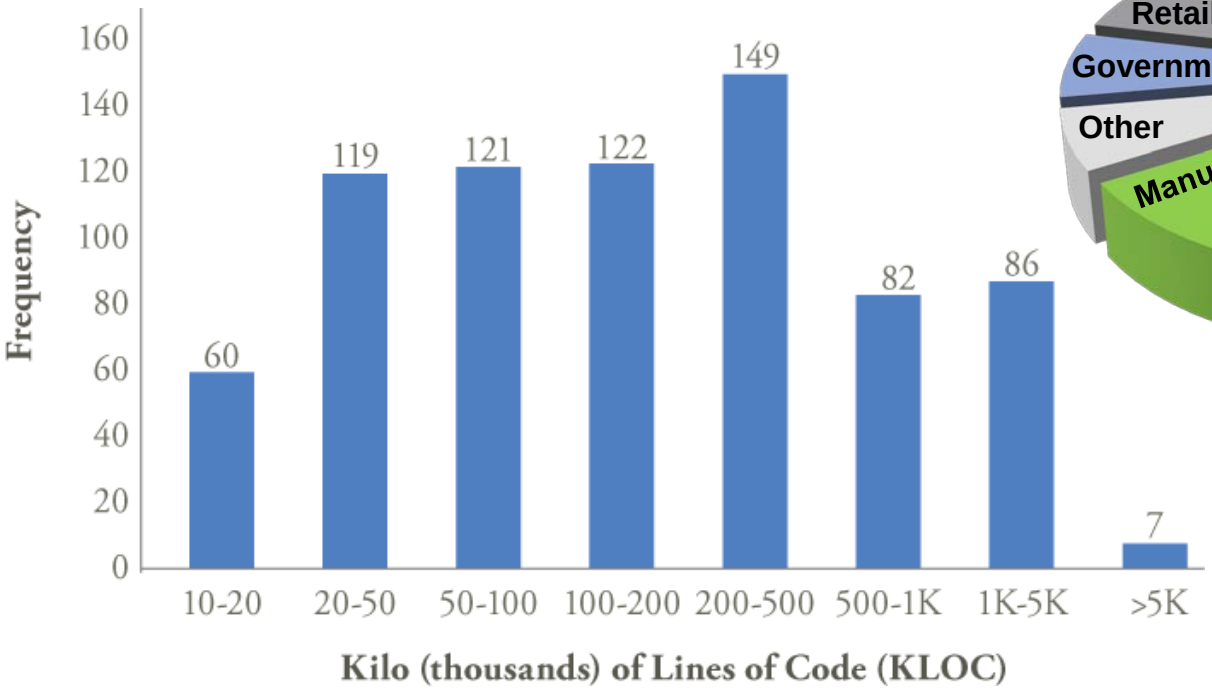
CAST's Application Intelligence Platform

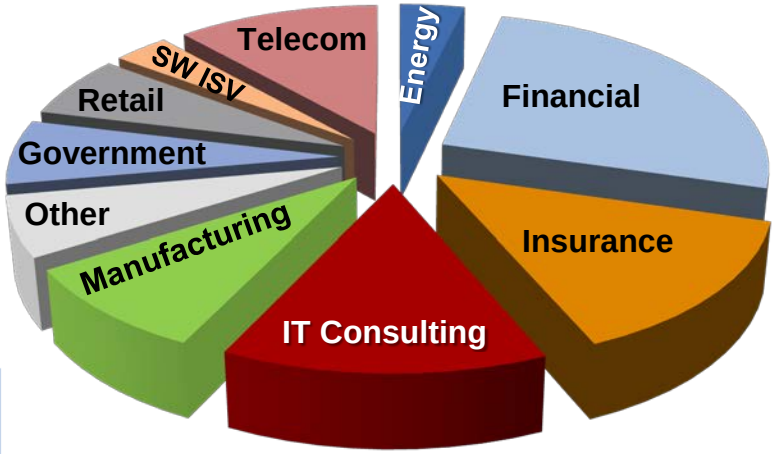


 © 2012 CAST

Appmarq Structural Quality Repository

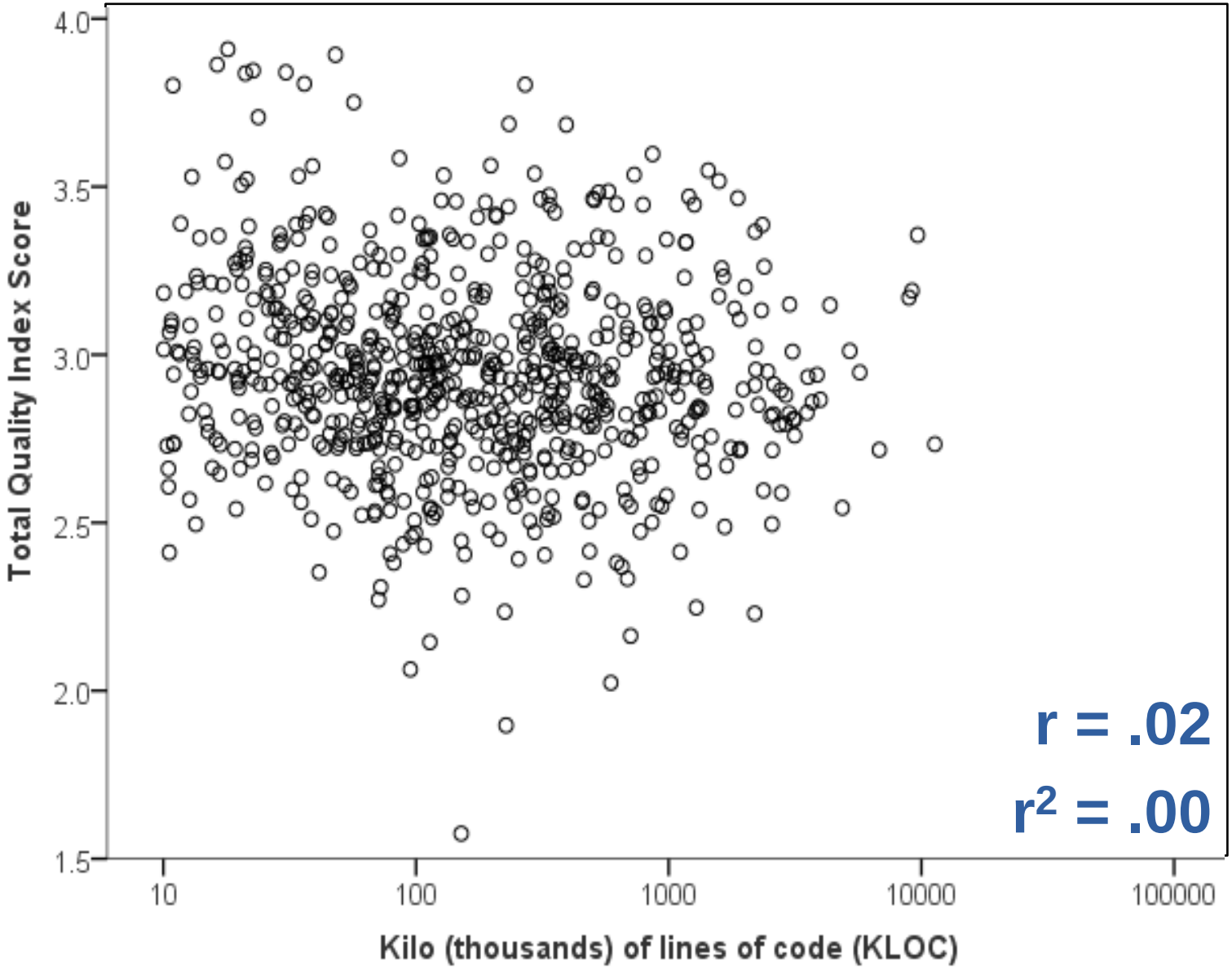
- CAST's repository of structural quality data
  - **745 Applications**
  - **160 Companies, 14 Countries**
  - **321,259,160 Lines of Code; 59,511,706 Violations**





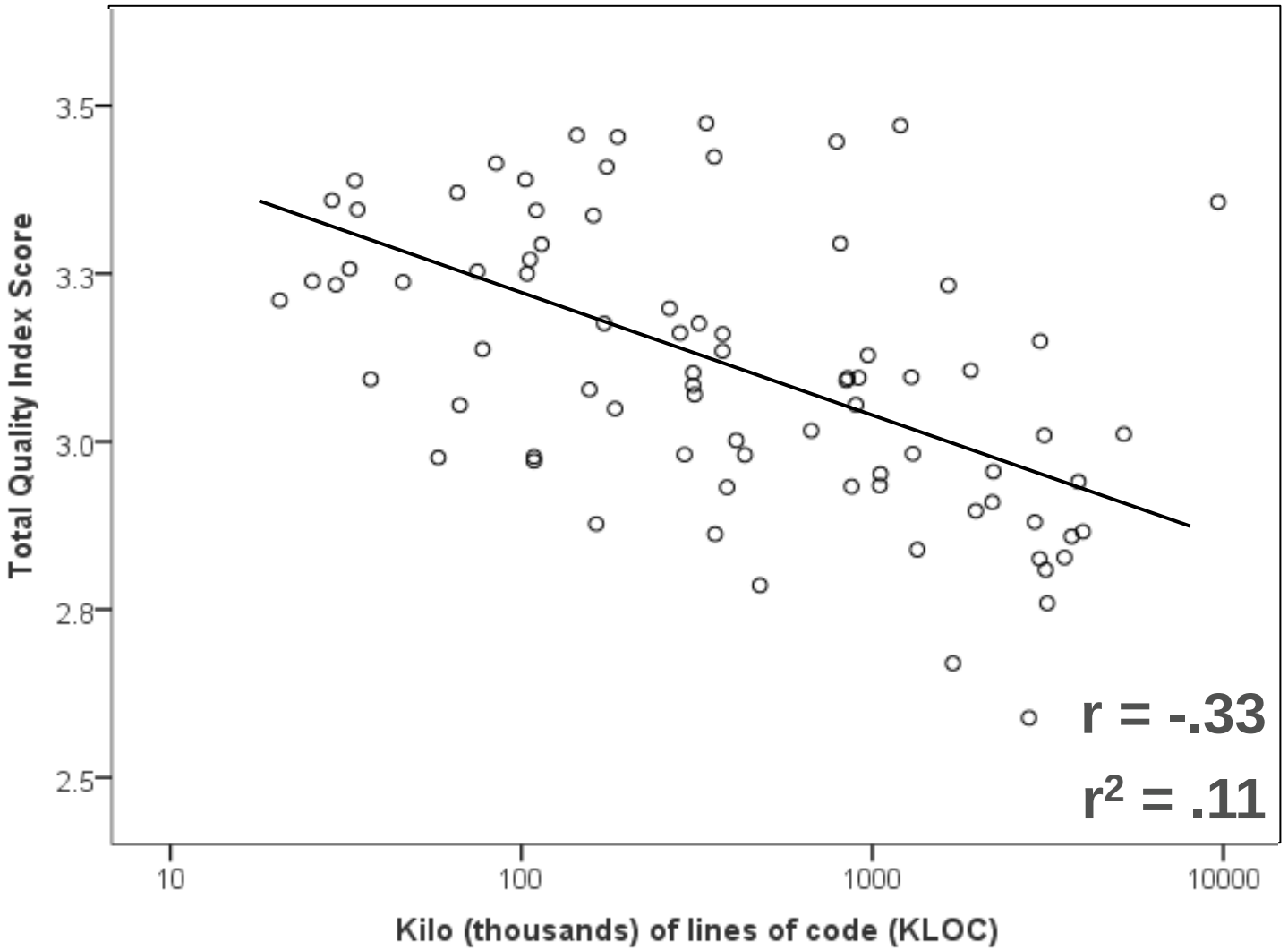
© 2012 CAST

Structural Quality Unrelated to Size



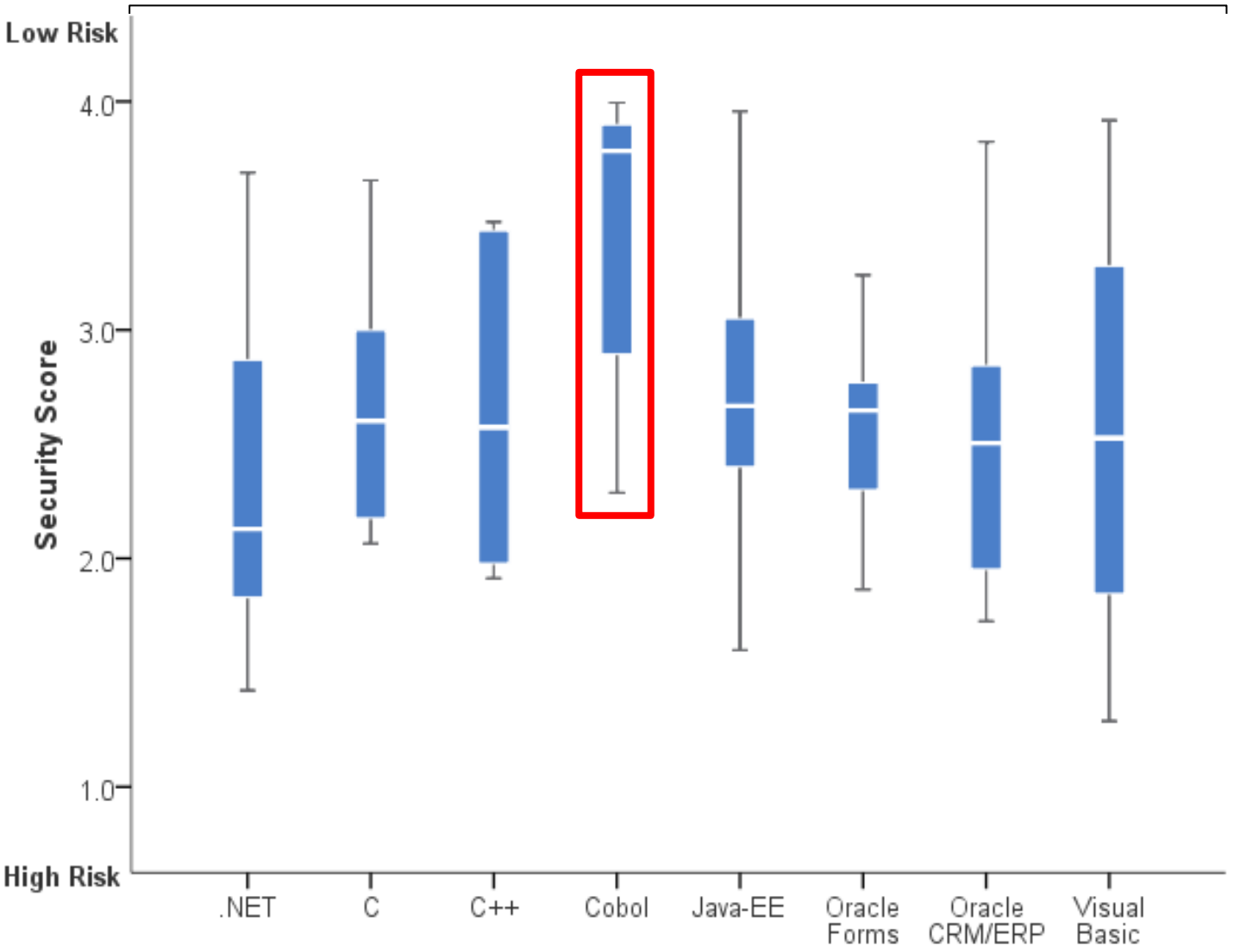
© 2012 CAST

…Except for COBOL



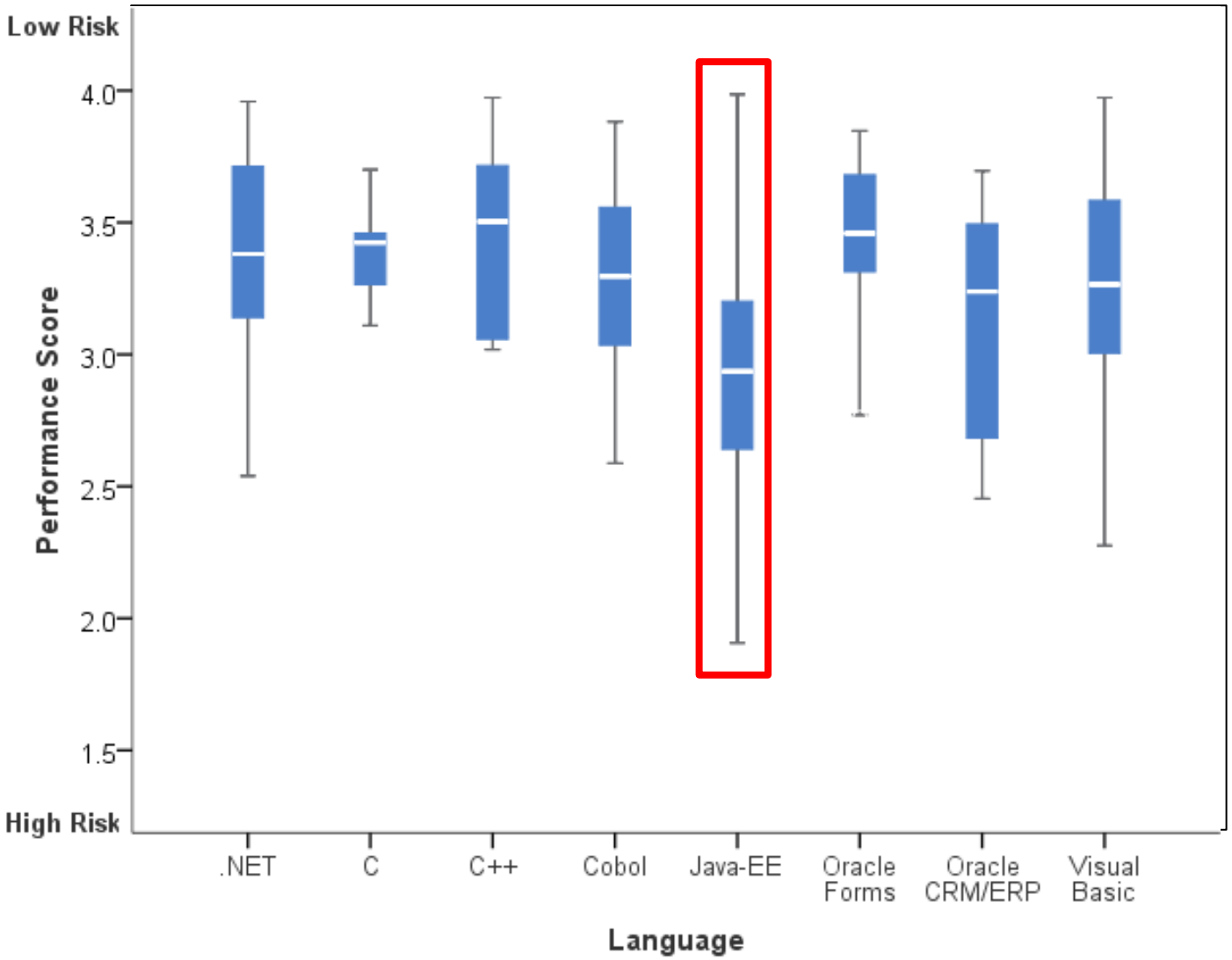
© 2012 CAST

Security Scores by Language



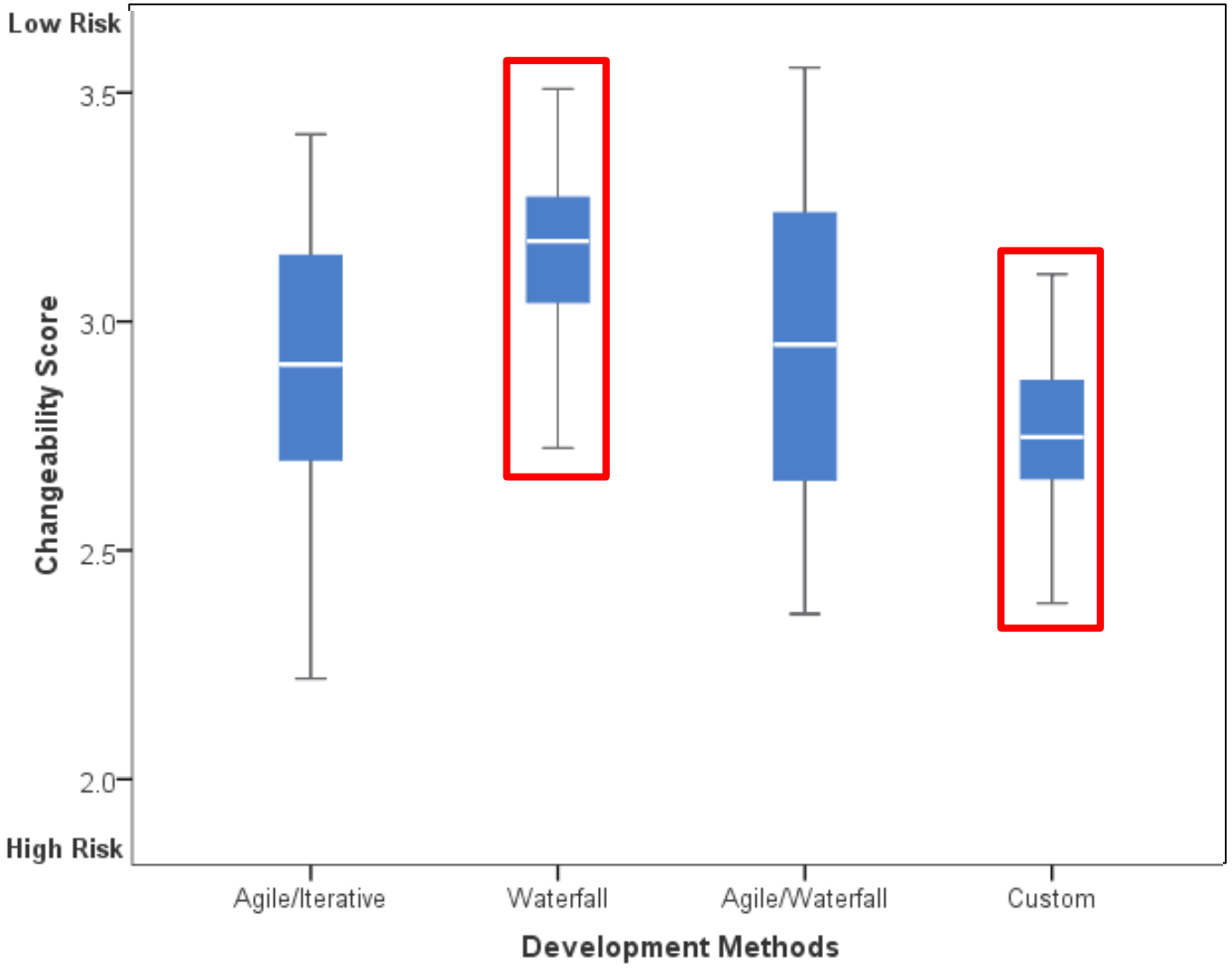
© 2012 CAST

Performance Scores by Language



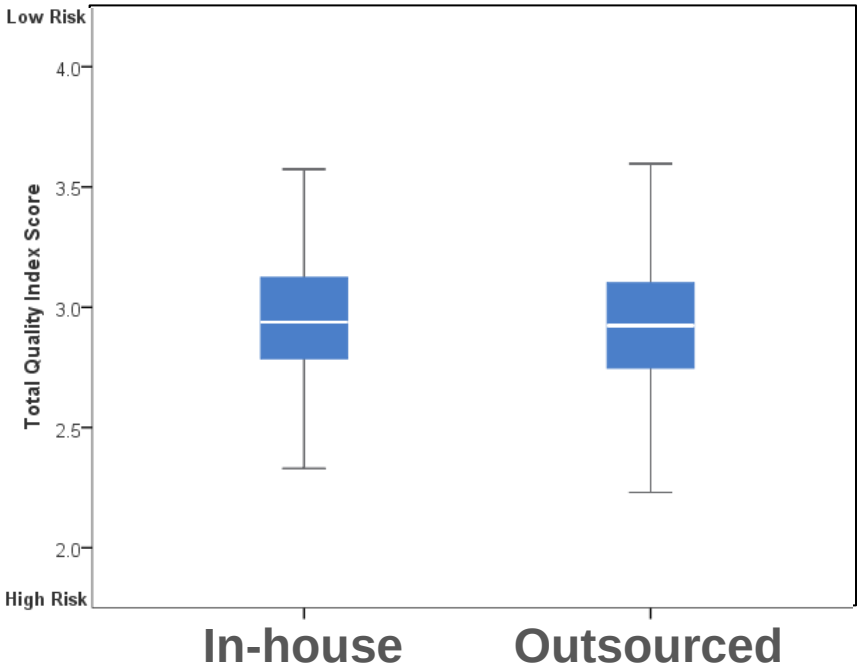
© 2012 CAST

Changeability Scores by Method

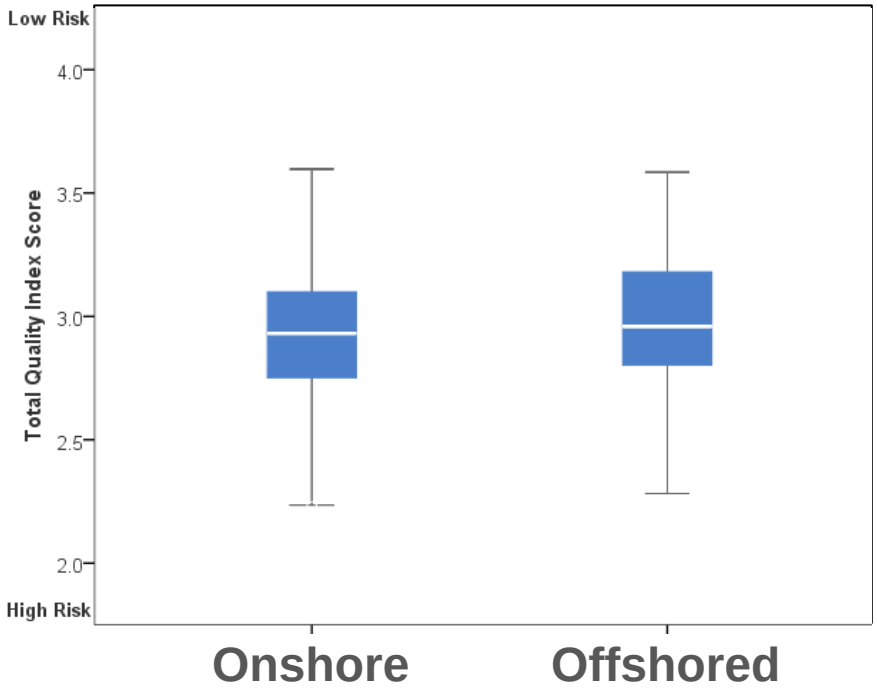


© 2012 CAST

No Differences by Source or Shore



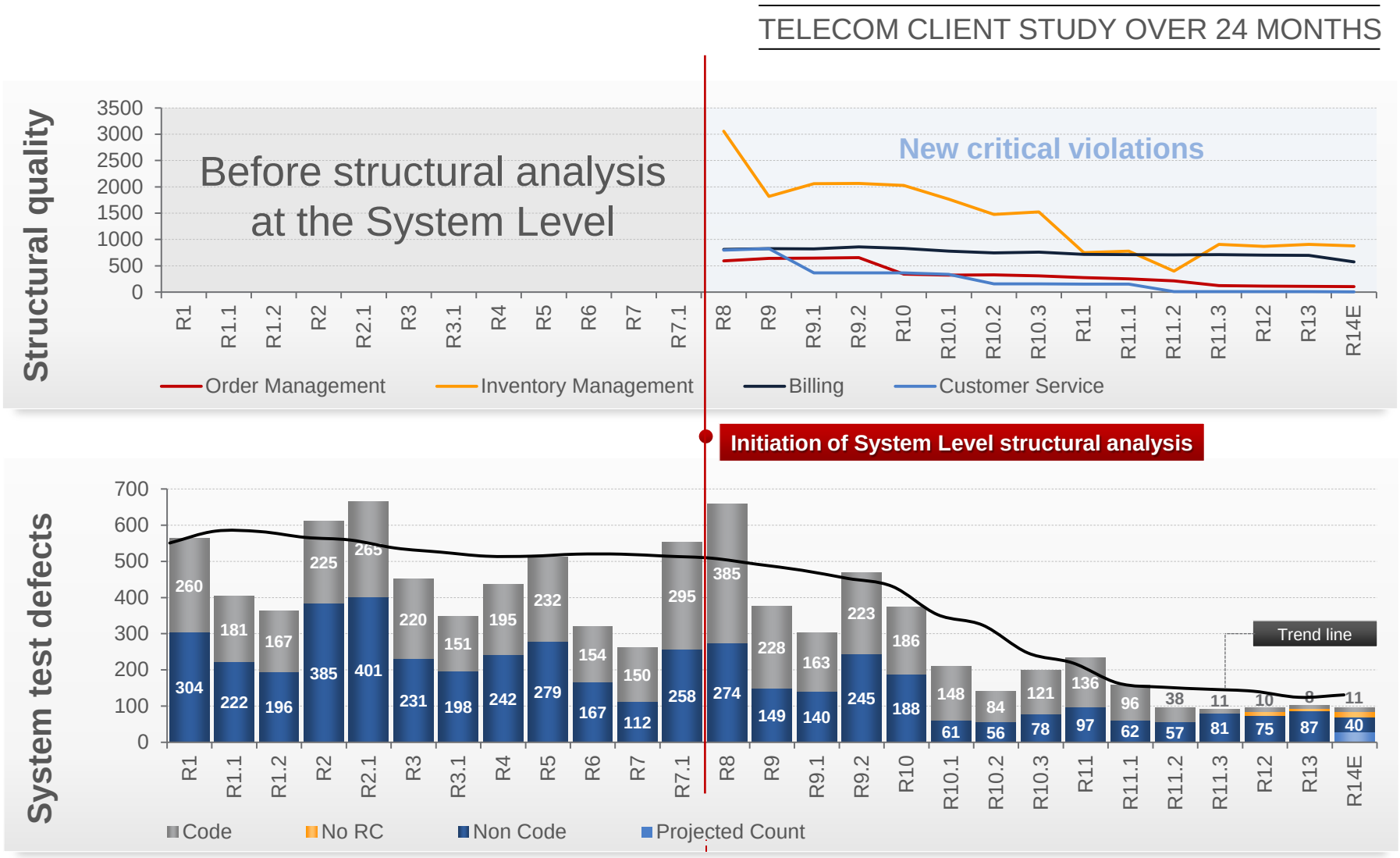
SOURCE



SHORE

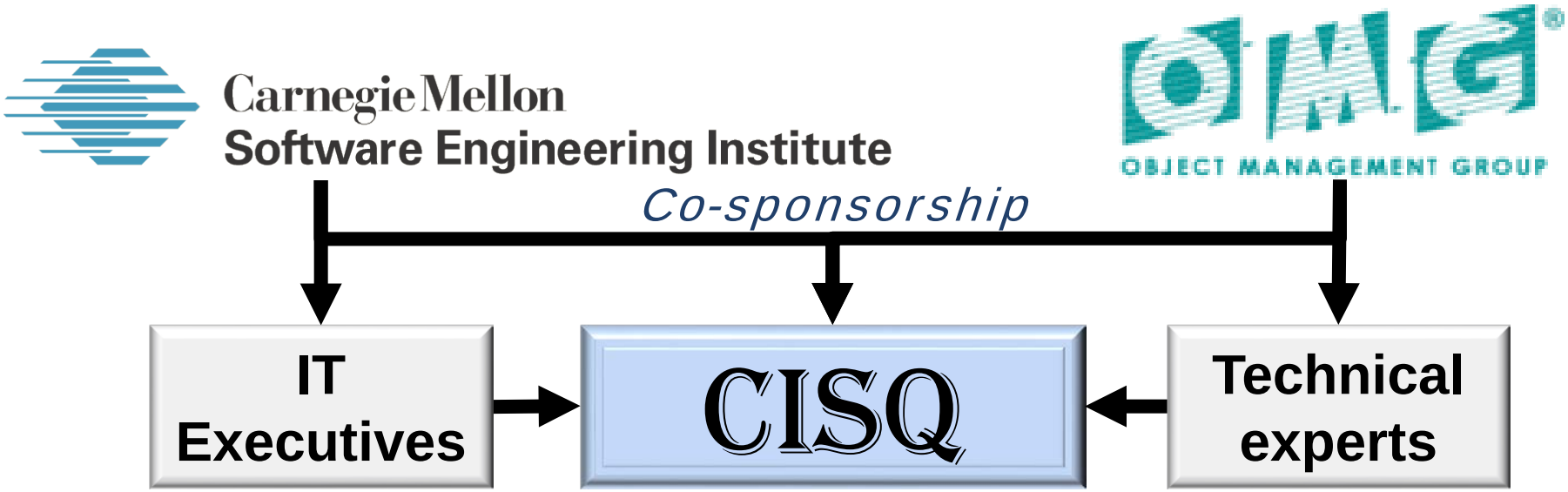
© 2012 CAST

Structural Analysis Reduces Risks/Costs



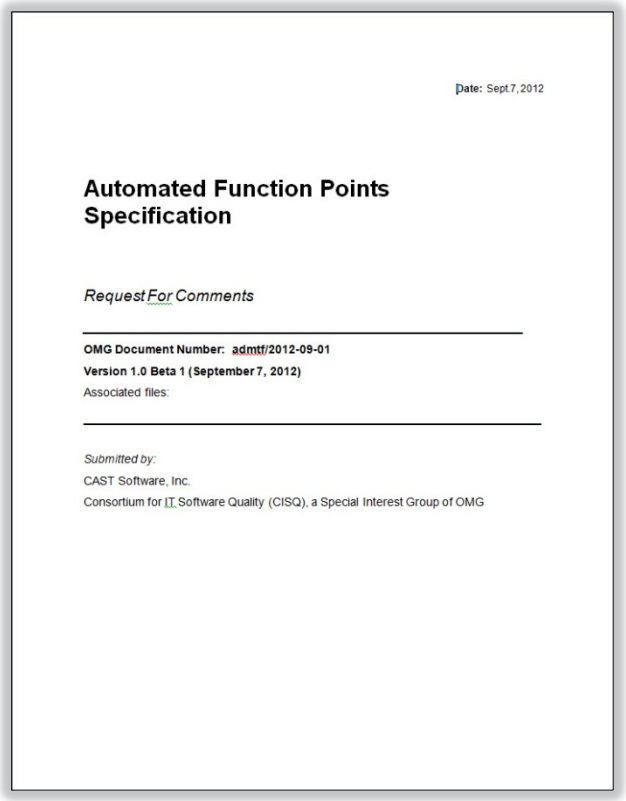
© 2012 CAST

Consortium for IT Software Quality



 © 2012 CAST

Recent CISQ Events



Date: Sept.7, 2012

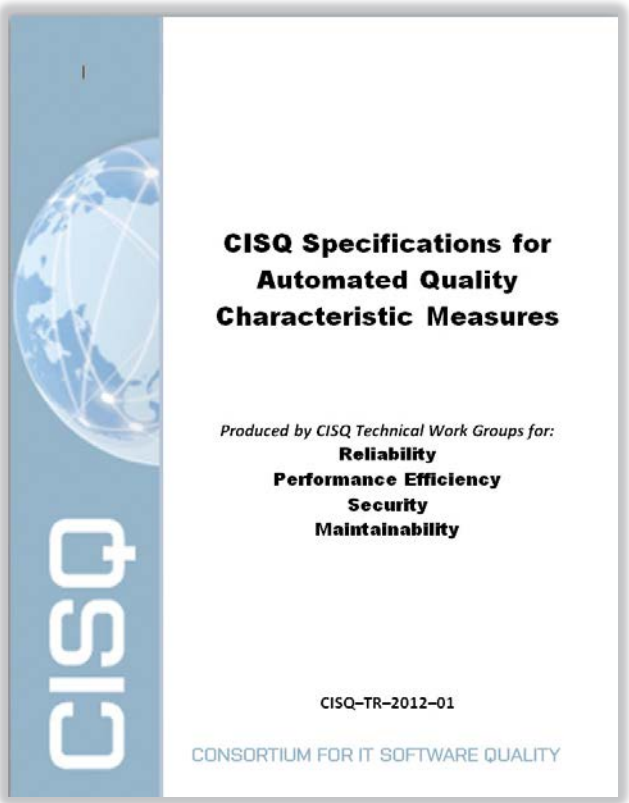
Automated Function Points Specification

Request For Comments

OMG Document Number: admtf/2012-09-01
Version 1.0 Beta 1 (September 7, 2012)
Associated files:

Submitted by:
CAST Software, Inc.
Consortium for IT Software Quality (CISQ), a Special Interest Group of OMG

OMG Architecture Board approved the Automated Function Point spec for a 3-month public review leading to final approval at the Dec. OMG meeting



On 9/12 CISQ released specifications for automated measures of Reliability, Performance Efficiency, Security, and Maintainability

© 2012 CAST

Global Trends

- Reducing operational risk is valued over reducing cost of ownership
- Code produced with Agile methods could have higher cost of ownership
- Sourcing and shoring choices do not affect structural quality
- Structural quality measures are being used as Service Level Agreements in contracts

But.....

- STRUCTURAL QUALITY MUST IMPROVE

© 2012 CAST