Model Based System Engineering (MBSE) Quality

Dave Norton
Executive Director
What We Do

- CISQ Executive Forums
  - Automated Function Points
  - Reliability
  - Performance Efficiency
  - Security
  - Maintainability

OMG

OMG Approved Standards

ISO Fasttrack

Deployment Workshops

©2020 CISQ
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 that they must be remediated were included in the OMG standards. The OMG standards are being submitted to ISO 25000.

### Typical Areas for CISQ

<table>
<thead>
<tr>
<th>CISQ Structural Quality Measures</th>
<th>Example architectural and coding weaknesses included in the CISQ measures</th>
</tr>
</thead>
</table>
| Security                         | • SQL injection  
                                | • Cross-site scripting  
                                | • Buffer overflow  
| Reliability                      | • Empty exception block  
                                | • Unreleased resources  
                                | • Circular dependency  
| Performance Efficiency           | • Expensive loop operation  
                                | • Un-indexed data access  
                                | • Unreleased memory  
| Maintainability                  | • Excessive coupling  
                                | • Dead code  
                                | • Hard-coded literals  

- **Security**: 74 weaknesses (Top 25 CWEs)
- **Reliability**: 74 weaknesses
- **Performance Efficiency**: 18 weaknesses
- **Maintainability**: 29 weaknesses

©2020 CISQ
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 Information Software Quality (CISQ) for guidance on how to measure, evaluate and improve software.”
Focus on Outcomes

- Higher Productivity
- Business Outcomes
- Reduces Risk
- Agility
- Improves Quality
CISQ Membership
Recap on MBSE
<table>
<thead>
<tr>
<th>Model</th>
<th>Based</th>
<th>Systems</th>
<th>Engineering</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ontology</td>
<td>Analysis</td>
<td>Operational</td>
<td>Process</td>
</tr>
<tr>
<td>Abstraction</td>
<td>Decisions</td>
<td>Engineering</td>
<td>Practices</td>
</tr>
<tr>
<td>Semantics</td>
<td>Planning</td>
<td>CPS</td>
<td>Methods</td>
</tr>
<tr>
<td>Syntax</td>
<td>Risk</td>
<td>SoS</td>
<td>Frameworks</td>
</tr>
<tr>
<td>Algorithms</td>
<td>Collaboration</td>
<td>Social</td>
<td>Maturity Models</td>
</tr>
<tr>
<td>Viewpoints</td>
<td>Implementation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Authoritative</td>
<td>Support</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
OMG SysML Narrowing the Conceptual Gap

1. Structure

2. Behavior

3. Requirements

4. Parametrics

Analysis Models

System Model

Requirements

V&V Models

Hardware Models

Software Models
The Model is **Marked** up ready for transformation,
MBSE Is For Life, Not Just Initial Development

Hi-Fidelity MBSE Models Have Greater Value and Utility
SAFe has the concept of MBSE.

You can use MBSE without a enterprise agile framework, but it is harder.
Why The Increasing Interest?
MBSE Is Common In IoT and CPS

- Enforcement
- Traffic Flow
- Payment Systems
- Mobile Device(s)
- Parking Sensors
- Ground Water Sensor
- Automatic Beer Stocking
- Automotive (Various)
- Dementia Care Tracking
So What's The Problem?
Model Generated Code Needs To Be Of High Quality

```java
package my;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class TestServlet extends HttpServlet implements Servlet {
    static final long serialVersionUID = 1L;

    public TestServlet() {
        super();
    }

    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        doPost(request, response);
    }

    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        response.getWriter().println("blah");
    }
}
```

How do I create a profile to format Java code in Eclipse?
A Digital Ecosystem is a System of Systems – A Weakness In One Can Be A Weakness In All

For Example In Banking Over 21% Of Incidents 3rd Party Related (UK FCA 2019)
MBSE & Digital Twins Are The New Attack Vector, The Greater The Model Fidelity The Greater The To An Attacker
Bottom Line – Poor Model Leads To Poor Outcomes

- Higher Carbon Foot Print
- Poor Performance
- Harder To Maintain
- Higher TCO
- Poor Reliability
- Poor Security
- Low Trustworthiness
So What Is CISQ Doing About It?
**CWE** is an enumeration (list) of software architecture, design, or code weaknesses.

Weaknesses are defined as flaws, bugs, faults, or other errors, that create vulnerabilities that can be exploited by both internal and external forces.

Weaknesses can be found in software implementation, code, design, and architecture.
CWE Analysis Already Best Practice At The Code Level

• **OWASP Top 10 Vulnerabilities**—most critical web application security risks – CWEs & CVEs

• **OWASP Application Security Verification Std v4.0** – 14 categories guide automated unit & integration tests – most all verification checks have corresponding CWEs

• **SANS/CWE Top 25** — most commonly encountered cyber weakness enumerators (CWEs),

• **CISQ Object Management Group (OMG)** Automated Source Code Measures for technical debt & structural quality (Security, Reliability, Performance Efficiency & Maintainability) – all based on CWEs
Shift-Left – Move Code Weakness and Vulnerability Analysis Into The Model

CWE Discovered At Code Level Using Static Analysis
Formalize Normative Model Specification For The CWE’s

CWE-284: Improper Access Control

**Weakness ID:** 284  
**Abstraction:** Class, Structure: Simple

**Description:** The software does not restrict or incorrectly restricts access to a resource from an unauthorized actor.

**Extended Description:** Access control involves the use of several protection mechanisms such as:
- Authentication (proving the identity of an actor)
- Authorization (ensuring that a given actor can access a resource), and
- Accountability (tracking of activities that were performed)

When any mechanism is not applied or otherwise fails, attackers can compromise the security of the software by gaining privileges, reading sensitive information, executing commands, evading detection, etc.

There are two distinct behaviors that can introduce access control weaknesses:
- Specification: incorrect privileges, permissions, ownership, etc. are explicitly specified for either the user or the resource (for example, setting a password file to be world-writable, or giving administrator capabilities to a guest user). This action could be performed by the program or the administrator.
- Enforcement: the mechanism contains errors that prevent it from properly enforcing the specified access control requirements (e.g., allowing the user to specify their own privileges, or allowing a syntactically-incorrect ACL to produce insecure settings). This problem occurs within the program itself, in that it does not actually enforce the intended security policy that the administrator specifies.

CWE Visible In The Model
The New Standard Will Allow CWE Analysis Before Anything Is Generated

1. Structure
2. Behavior
3. Requirements
4. Parametrics

[CISQ logo]
Geek Alert – MBSE Quality Model
What Will The Standard Give You ?

• Model Validation Earlier In the MBSE Life Cycle - 1:40 to 1:60 ratio in cost compared to code review and testing
• A Way Of Certifying The MBSE Environment Regard Generated (CWE) Code Weakness
• Certify Supplier MBSE Quality Against (CWE) Code Weakness
• Consistent Model Validation Across The Ecosystem
• Improved Quality, Lower Risk and Happier Customer
Help Us Develop The Next Generation Of Digital Standards

Individual Membership
Stay updated on this work and network with members in the community. Individual membership is free.

• Subscribe to CISQ's email list
• Receive updates on the standards
• Receive technical guidance documents
• Receive event invitations

Corporate Membership
Contribute to the standards and participate in deployment activities. Sponsorship is open to companies, government agencies, not-for-profit, and academic institutions.

• Team members participate in working groups
• An executive joins the Governing Board
• Your organization is listed as a supporter of all CISQ events, including complimentary passes and an exhibit table
• See [benefits of corporate membership](#)
Thank You

David Norton
Executive Director
david.norton@it-cisq.org