Contracting Best Practice

Lower Risk and Improve Outcomes with Suppliers by Using Software Structural Quality Standards

Author: David Norton – Executive Director, CISQ

Table of Contents

Please Note: The Consortium for Information & Software Quality™ (CISQ™) and Object Management Group® (OMG®), its managing organization, cannot give legal advice and the following is shared in good faith for example purposes only. As part of due diligence, we recommend all contracts are reviewed by a competent contracts lawyer.

Introduction

The software structural quality measures developed by CISQ and standardized by the Object Management Group® (OMG®) are for the automated analysis of software quality. The standards are intended to lower risk to the business, ICT function, and end user by removing structural code weaknesses during software development.

The following document outlines example contracting clauses and related standards for software structural quality measures and controls suitable for inclusion in new software development or enhancement contracts with suppliers. The recommendations may also be used for product contracting where there may be continuous delivery of features.

It should be noted this contracting best practice is not an exhaustive list of software structural quality measures; however, it does cover the most critical structural weaknesses and related vulnerabilities in source code. This document can be supplemented with other suitable structural measures if necessary.

The contracting recommendations in this document should be aligned with the overall testing and security clauses of the master contract.

General Recommendations for Pre-Contracting

Standards Compliance

The supplier should inform the customer as to the level of compliance they hold, or claim to hold, against relevant OMG standards of Software Structural Quality and Technical Debt, hereby referred to as “The Structural Standards.”

In the case of non-compliance, the supplier should demonstrate they can deliver the required quality measurement capability provided by “The Structural Standards” by other verifiable means.

The customer has the right to audit for compliance and/or request relevant documentation from the supplier in support of their statement of compliance.

Level	Description	Compliance
Level 1	Holds independently certified compliance to standard	Full
Level 2	Demonstrable self-certified auditable compliance to standard	Full
Level 3	Generally In Accordance (GIA) with only immaterial departures from standard, if any	Full
Level 4	Significant material departures from standard	Partial
Level 5	No compliance to standard	None

Pre-Contract Systems Assessment

It is considered best practice when dealing with existing systems for the systems to undergo technical architecture and code structural review before contracting negotiation and work commences. The review process is required to establish baseline quality levels to set fair and realistic contract quality and productivity levels and target incentive thresholds.

Systems assessment may be undertaken using either a manual review process or a systematized code and asset analysis, or any combination of the two.

Contracting and Structural Quality

Measurement of Software Structural Quality

For the purpose of this contract, software structural quality will be measured at the code level using static code analysis.

To ensure consistency and adherence to best practices, the OMG structural quality standards will be applied to measure Reliability, Security, Performance Efficiency, Maintainability and Technical Debt.

- Reliability measures the risk of potential application failures and the stability of an application when confronted with unexpected conditions (ISO/IEC 25010).
- Performance Efficiency assesses characteristics that affect an application's response behavior and use of resources under stated conditions (ISO/IEC 25010).
- Security assesses the degree to which an application protects information and data so that persons or other products or systems have the degree of data access appropriate to their types and levels of authorization (ISO/IEC 25010).
- Maintainability represents the degree of effectiveness and efficiency with which a product or system can be modified by the intended maintainers (ISO/IEC 25010).
- Technical Debt measures the system attributes known to lead to critical structural weaknesses in production code that must be fixed to reduce cost and lower IT and business risk.

It should be noted that the Contracting and Structural Quality section (this section) does not include the full security contracting clauses which are detailed in section x.x. In the event of any conflict between satisfying the quality section y.y and security section x.x, the security section shall take precedent.

Automated software quality measurement and reporting can be complemented with additional standards such as ISO/IEC 25010:2011 Systems and software engineering — Systems and Software Quality Requirements and Evaluation (SQuaRE) — System and Software Quality Models. The use of any complementary software quality standards must be agreed upon with the customer.

Delivered Quality

All code developed for the customer by the supplier must be analyzed for structural quality in adherence with Standards of Software Structural Quality section x.1 of this document, unless an exception is granted.

The supplier will ensure for each software deliverable submitted for acceptance, all of the applicable structural quality thresholds set forth in Software Structural Quality Limits section a.a are satisfied.

With each deliverable submitted for acceptance, the supplier shall provide evidence of compliance with the above requirement and shall sign-off on the statement of compliance.

The statement of quality compliance must be signed by an agreed upon and named individual from the supplier's organization.

The supplier can only invoice for code that has reached the agreed quality level of quality and agreed exceptions (if any).

3rd party components that will not be tested for structural quality by the supplier must be agreed upon in advance and listed in structural quality exceptions.

No payment will be authorized until the statement of quality conformance has been received by the customer with supporting evidence of compliance, and its receipt has been acknowledged by the customer.

Exceptions to structural quality analysis are allowed with agreement from the customer. Any exception must be agreed upon in advance with the customer and clearly marked in the invoice as “Non-conformant to OMG Quality Standards.”

Customer Right to Reject Delivery

If the measurement under any agreed metric does not satisfy one or more of its corresponding standards, the customer shall have the right not to accept the non-compliant software and any other software that interoperates with or is affected by the non-compliant software, even if previously accepted.

The supplier shall correct the software so that the software satisfies all applicable standards and redeliver the corrected software within 28* days.

The supplier will keep the customer apprised of its progress toward resolution of any non-compliance. If the supplier is unable to promptly resolve a problem, the supplier will immediately notify the customer. This notice will include (i) a reasonable explanation for the delay, and (ii) a good faith schedule and plan for correction.

*Recommendation, will vary with the complexity and size of the system

Standards of Software Structural Quality

The following OMG standards will be used for software structural quality measurement based on automated static code analysis.

- Automated Source Code CISQ Reliability Measure
  https://www.omg.org/spec/ASCQM/
- Automated Source Code CISQ Performance Efficiency Measure
  https://www.omg.org/spec/ASCQM/
- Automated Source Code CISQ Security Measure
  https://www.omg.org/spec/ASCQM/
- Automated Source Code CISQ Maintainability Measure
  https://www.omg.org/spec/ASCQM/
- Automated Technical Debt Measure
  https://www.omg.org/spec/ATDM/

It is the supplier’s responsibility to ensure any tools used in static code analysis support the above standards.

Software Structure Target Quality and Limits

Structural Quality

For each software deliverable submitted for acceptance, it must meet the following levels of structural and code quality as defined by the relevant standard.

Automated Source Code CISQ Reliability Measure	
https://www.omg.org/spec/ASCQM/	
The Automated Source Code CISQ Reliability Measure contains 74 critical coding and architecture weaknesses that must be avoided for **Reliability**. The Automated Source Code CISQ Reliability Measure is based on The MITRE Corporation Common Weakness Enumeration and CWE identifiers. See Common Weakness Enumeration (CWE) for more detail on each CWE. The source code should NOT contain these 74 critical weaknesses known to severely impact reliability.	
Target Quality	**0 (Zero) Reliability weaknesses as defined in the standard.**

Automated Source Code CISQ Performance Efficiency Measure	
https://www.omg.org/spec/ASCQM/	
The Automated Source Code CISQ Performance Efficiency Measure contains 18 critical coding and architecture weaknesses that must be avoided for **Performance Efficiency**. The Automated Source Code CISQ Performance Measure is based on The MITRE Corporation Common Weakness Enumeration and CWE identifiers. See Common Weakness Enumeration (CWE) for more detail on each CWE. The source code should NOT contain these 18 critical weaknesses known to severely impact performance.	
Target Quality	**0 (Zero) Performance Efficiency weaknesses as defined in the standard.**

Automated Source Code CISQ Security Measure	
https://www.omg.org/spec/ASCQM/	
The Automated Source Code CISQ Security Measure contains 74 critical coding and architecture weaknesses that must be avoided for **Security**. The Automated Source Code CISQ Security Measure is based on The MITRE Corporation Common Weakness Enumeration and CWE identifiers. See Common Weakness Enumeration (CWE) for more detail on each CWE. The source code should NOT contain these 74 critical weaknesses known to severely impact security.	
Target Quality	**0 (Zero) Security weaknesses as defined in the standard.**

Automated Source Code CISQ Maintainability Measure	
https://www.omg.org/spec/ASCQM/	
The Automated Source Code CISQ Maintainability Measure contains 29 critical coding and architecture weaknesses that must be avoided for **Maintainability**. The Automated Source Code CISQ Maintainability Measure is based on The MITRE Corporation Common Weakness Enumeration and CWE identifiers. See Common Weakness Enumeration (CWE) for more detail on each CWE. The source code should NOT contain these 29 critical weaknesses known to severely impact maintainability.	
Target Quality	**0 (Zero) Maintainability weaknesses as defined in the standard.**

Technical Debt

The following technical debt standard requires a risk-based approach to its acceptance level. Preferably, the limit should be zero technical debt defects. However, with agreement from the customer accepting a higher level of risk to the business, lower quality thresholds can be agreed upon.

The customer and supplier must agree to the impact of each of the technical debt characteristics defined in the **Automated CISQ Technical Debt Measure** standard based on the needs of the business and level of risk. This must be done in advance of delivery and as a documented list with each technical debt characteristic and its agreed severity.

- A **Critical weakness** is a discrepancy in the code from the standard that is deemed to be hazardous or unsafe, and with serious impact.
- A **Major weakness** is a discrepancy in the code from the standard that is likely to create failure of the system for its intended purpose.
- A **Minor weakness** is a discrepancy in the code from the standard, but one that is not likely to affect the usability of the system.

The severity of the technical debt characteristics may be changed during delivery with agreement from both parties.

Automated CISQ Technical Debt Measure		
https://www.omg.org/spec/ATDM/		
Severity	**Target Quality**	**Remediation Period**
Critical weakness	0 (Zero) of code	N/A
Major weakness	Less than 5% of code	5 Days
Minor weakness	Less than 10% of code	28 Days

Deliverables that are above their target technical debt quality level will trigger the penalty clause whereby all relevant non-compliance code must be within compliance within the agreed period. Failure to do so will result in the relevant penalty being applied.

Exclusion

The following artifacts are excluded contractually from software structural quality analysis with agreement from the customer.

Excluded artifacts must not affect the software structural quality of the remainder of the system under development.

All exclusions must be agreed in advance of contract signing. Exclusion may be granted in development with agreement with the customer.

Exclusions	**Reason**	**Will SQA be undertaken outside of the contract**

Penalties

If the supplier fails to meet the agreed base quality level for *three consecutive invoicing periods or *three consecutive sprints in the case of agile-based delivery, the penalty clause defined is section “a.a” shall be triggered.

If overall average quality for the program is below the agreed quality base level at the completion of the contract, the penalty clause defined is section “a.a” shall be triggered.

*Example periods

Note: If agreed base quality has not been met as defined above, and (a) the supplier can show the root cause was due to customer-owned activities, and (b) if issues with the aforementioned activities were raised with the customer at the earliest opportunity, then the penalties clause shall be waived with customer agreement.

Structural Quality Monitoring

Based on the "The Structural Standards," it is expected quality data be made available to the customer on a continuous basis, and the customer can review said data when required with 24 hours prior notice.

When structural quality data cannot be made available to the customer continuously, it must be reported (a) when the code enters functional testing, and (b) before delivery and invoicing to the customer as outlined in the "Delivered Quality" section of this document.

See the CISQ whitepaper, "Effective Software Quality Metrics for ADM Service Level Agreements," for reference https://www.it-cisq.org/adm-sla/index.htm

Independent Arbitration

Either the customer or supplier can engage a mutually agreed upon 3rd party to arbitrate on the structural quality of delivered code as defined by the "The Structural Standards."

The supplier agrees that if they trigger the independent arbitration process, the invoice payment will be withheld until an independent arbitration process is complete. The supplier also agrees to pay all reasonable costs related to the independent arbitration process.

The customer agrees that if they trigger the independent arbitration process, to pay any related supplier invoices within 28 days, and to seek repayments from the supplier if arbitration is in the customer's favor.

Customer and supplier agree to abide by independent arbitration.

Termination

In the event that the supplier is unable to achieve compliance with applicable quality levels defined within the relevant quality section of this document by the date(s) required in the Development Agreement/SOW, then the customer shall have the right to terminate the Development Agreement and obtain a refund of any amounts previously paid by the customer.

Unless and until such termination occurs, the supplier shall, at no additional cost or expense to the customer, continue to attempt to remedy any non-compliance with quality levels using the process described in Section X remediate action, until the agreed quality level is achieved.