

# Maintainability Weakness Descriptions

This document presents descriptions of the 20 weaknesses contained in the CISQ Automated Quality Characteristic Measure for Maintainability. These descriptions have been simplified from their description in the published OMG® specification that used formalisms from other OMG meta-models to specify the weaknesses for representation in machine-processable XMI notation. The tables below present each weakness with its unique CISQ identifier, a brief descriptive name, and a fuller description of the weakness presented as a recommendation for remediation.

## Maintainability Weaknesses

a measure of the extent to which software contains weaknesses that make software hard to understand or change, resulting in excessive maintenance time and cost as well as higher defect injection rates.

CISQ identifier	Descriptor	Remediation
ASCMM-MNT-1	Control transferred outside of switch statement	Remove instances where the control flow is transferred outside a switch statement (e.g., depending on the technology, by using 'go to', 'continue', or 'break' statements)
ASCMM-MNT-2	Excessive inheritance from concrete classes	Remove instances where a class inherits from too many concrete classes (default threshold value for the maximum number of concrete class Inheritances is 1).
ASCMM-MNT-3	Hard-coded literals	Remove instances where a literal value is used to initialize a variable, field, member, etc. (exceptions are simple integers and a static constant variable, field, member, etc.)
ASCMM-MNT-4	Excessive coupling	Remove instances where a function, method, procedure, stored procedure, sub-routine, etc. has a Fan-Out value that is too large, that is, with too many references to other objects within the application. (default threshold value for the maximum number of references to other objects within the application is 5)
ASCMM-MNT-5	Condition value update within loop	Remove instances where a value of a local variable, field, member, etc. used in the condition of a loop is updated within the loop body
ASCMM-MNT-6	Excessive commented-out code	Remove instances where a function, method, procedure, stored procedure, sub-routine, etc. contains too much commented-out code (default threshold value for the maximum percentage of commented-out instructions is 2%).
ASCMM-MNT-7	Circular dependencies among modules	Remove instances where a module has references that cycle back to itself (for example, in JAVA this pattern means cycles between packages)

<b>ASCMM-MNT-8</b>	Excessively large file	Remove instances where a file has too many lines of code (default threshold value for the maximum number of lines of code is 1000)
<b>ASCMM-MNT-9:</b>	Too many/few horizontal layers	Remove instances where a model of the architectural layers of an application contains too many or too few horizontal layers (excluding the vertical utility layers) based on comparison to a threshold value. The default threshold value for the minimal number of horizontal layers is 4, and the default value for maximal number of horizontal layers is 8. Note: <ul style="list-style-type: none"> <li>This is a 'control' on the architectural model used in ASCMM-MNT-10 and ASCMM-MNT-12 to avoid defining a model designed to 'pass' these other weaknesses</li> </ul>
<b>ASCMM-MNT-10</b>	Layer-spanning component	Remove instances where a function, method, procedure, stored procedure, sub-routine, etc. is part of two architectural layers as defined in a model of the application's architectural layers
<b>ASCMM-MNT-11</b>	Excessive cyclomatic complexity	Remove instances where a function, method, procedure, stored procedure, sub-routine, etc. has a Cyclomatic Complexity that is too large (default threshold value for maximum Cyclomatic Complexity is 20)
<b>ASCMM-MNT-12</b>	Layer-skipping calls	Remove instances where a function, method, procedure, stored procedure, sub-routine, etc. from a higher horizontal layer directly calls a function, method, procedure, stored procedure, sub-routine, etc. in a lower horizontal layer that is not adjacent to the upper layer making the call, as defined in a model of the application's architectural layers (this excludes the vertical utility layers that can be referenced from any horizontal layer)
<b>ASCMM-MNT-13</b>	Excessive parameterization	Remove instances where a function, method, procedure, stored procedure, sub-routine, etc. has too many parameters in its signature (default threshold value for the maximum number of parameters is 7)
<b>ASCMM-MNT-14</b>	Control element with excessive data operations	Remove instances where a function, method, procedure, stored procedure, sub-routine, etc. has too many SQL or file operations (default threshold value for the maximum number of SQL or file operations is 7)
<b>ASCMM-MNT-15</b>	Public data element	Remove instances where a variable, field, member, etc. is declared as public
<b>ASCMM-MNT-16</b>	Cross element data access	Remove instances where a method from a class accesses a field, or member from another class
<b>ASCMM-MNT-17</b>	Excessive inheritance levels	Remove instances where a class inheritance level is too large (default threshold value for maximum Inheritance levels is 7)

<b>ASCMM-MNT-18</b>	Excessive child classes	Remove instances where the class number of children of a class is too large (default threshold value for maximum number of children of a class is 10)
<b>ASCMM-MNT-19</b>	Element redundancy	Remove instances of copy-paste between functions, methods, procedures, stored procedures, sub-routines, etc.
<b>ASCMM-MNT-20</b>	Dead code	Number of instances where a function or method is unreferenced by any other code item in the application (the application determines the scope of the search for code items that could call a function or method element)