

Release Notes for the MISRA-C:2004 Compliance Module - M2CM-2.2-QAC-7.0

Unless otherwise noted, all materials provided in this release are Copyright (c) 2007 Programming Research.

This is a release of M2CM solely for use with QA C 7.0 for Linux-RH72.

All usage of this software must be in accordance with the terms of a current licence agreement with Programming Research as supplied with QA C.

Contents

[Introduction](#)

[Installing M2CM](#)

[Compliance Module Components](#)

[Message Configuration](#)

[Project Configuration](#)

[Uninstalling M2CM-2.2-QAC-7.0](#)

[Implementation Changes](#)

Introduction

This document provides an introduction to the MISRA-C:2004 Compliance Module for Linux-RH72 (M2CM). The package is designed to enforce compliance with the MISRA-C:2004 Subset programming standard in conjunction with QA C 7.0.

QA C is a software tool that performs static analysis on source code. Invariably a process of configuration is required in order to introduce QA C effectively to a particular software development environment. It is important to have QA C configured so as to restrict the output of warning messages to those that are of particular concern. M2CM is one such configuration, it configures QA C to enforce compliance with the MISRA-C:2004 coding rules.

For some MISRA-C:2004 coding standard rules which are not enforceable directly by QA C, additional checks are provided through a post-analysis program. QA C provides a standard capability for including such "Post-Analysis Processing" when a source file is analysed. The "post-processor" may incorporate any number of additional checks that are required by the programming standard. These checks can be programmed using the analysis output files (.met) generated by the QA C parsing process.

Some rules impose constraints on the complete project rather than the code in a single translation unit. Where this is the case the compliance module includes a global checks report.

Please note that this version of M2CM is designed to operate with QA C version(s) 7.0 and will not

function correctly with earlier releases.

Installing M2CM

1. You may install M2CM directly from your release CD. Alternatively, if you wish to install from a directory on a local disc or fileserver, insert the CD, and copy its contents to a suitable location.
2. Create a temporary directory in which to unload the M2CM files and attach to it. For example:

```
mkdir /tmp/m2cm
```

```
cd /tmp/m2cm
```

3. Locate the file M2CM-2.2-QAC-7.0.tar and unpack into your temporary directory. For example:

```
tar -xvf /cdrom/M2CM-2.2-QAC-7.0.tar
```

This creates a number of directories in your current directory and the file install.cm.

4. Ensure that the QACPATH environment variable is defined as the QA C home directory. QACPATH is normally set by sourcing the .profile or .cshrc file found in the QA C home directory.
5. Execute the install.cm Bourne shell script. For example:

```
cd /tmp/m2cm
```

```
./install.cm
```

This procedure will install the M2CM files in the \$QACPATH/m2cm directory. On successful completion of install.cm M2CM installation is complete.

6. You are advised to take copies of your M2CM personality files before introducing any changes you wish to make.

Compliance Module Components

M2CM can only be installed as an add-on to QA C. When M2CM is installed, the components are copied to a directory called m2cm within the QA C installation directory. The compliance module contains a number of elements which configure QA C and also extend the range of analysis checks. The following table describes the most important files:

Directory	Files	Description
m2cm/bin	.cshrc	Script to initialise command line analysis.
	.profile	Alternative script to initialise command line analysis.
	about_m2cm	M2CM version details report program.

	qac++.return_codes	Command line interface program configuration.
	qac.return_codes	Command line interface program configuration.
	qacsa_m2cm	Secondary-analyser program.
	qacsa_m2cm.return_codes	Secondary-analyser program configuration.
	qaw	Command line interface program.
	qaw.options	Command line interface program configuration.
	qaw.options.properties	Command line interface program configuration.
m2cm/messages	qac.usr.m2cm	User message file.
	nnnn.html	Replacement HTML help files which supplement standard QA C help text with cross-references to MISRA-C:2004 rules.
	rule*.html	HTML files which outline the implementation details of each MISRA-C:2004 rule. These are referenced by the HTML Annotated Source and serve as an online reference to the MISRA-C:2004 standard.
m2cm/doc	M2CM-2.2-QAC-7.0-Coding_Standard_Manual.html	Manual for coding standard.
	M2CM-2.2-QAC-7.0-Coding_Standard_Manual.pdf	Manual for coding standard.
	M2CM-2.2-QAC-7.0-Coding_Standard_Review.html	Coding standard review document.
	M2CM-2.2-QAC-7.0-QAC-Message_Usage_Summary.html	Summary of usage of QAC messages.
	M2CM-2.2-QAC-7.0-QAC-Message_Usage_Summary.pdf	Summary of usage of QAC messages.
	M2CM-2.2-QAC-7.0-Rule_Enforcement.html	Rule enforcement report.
	M2CM-2.2-QAC-7.0-Rule_Enforcement.pdf	Rule enforcement report.
	M2CM-2.2-QAC-7.0-Rule_Xreference.html	Compliance matrix.
	M2CM-2.2-QAC-7.0-Rule_Xreference.pdf	Compliance matrix.
	QAW_user_guide.pdf	User guide for command line interface.
	M2CM-2.2-QAC-7.0-Linux-RH72-Release_Notes.html	Release notes.
	M2CM-2.2-QAC-7.0-Linux-RH72-Release_Notes.pdf	Release notes.
m2cm/personalities		

m2cm.p_s	Message personality.
m2cm.p_a	Analysers personality.
m2cm.p_c	Compiler personality.
m2cm/projects/m2cm_examples/*	Directory containing example source directory, personalities and output directory. NB some compliance modules do not have a projects directory.

Personality Files

Three personality files are provided in the m2cm/person directory. Of these the most important is the message personality which restricts the generation of messages to those which are relevant to the enforcement of the MISRA-C:2004 rules.

Post-Analysis Program

Where rules are not enforceable directly by QA C, additional checks may be provided through a post-analysis program. The M2CM post-analysis program qacsa_m2cm is located in the m2cm/bin directory.

User Message File

The M2CM user message file qac usr.m2cm is located in the m2cm/messages directory. It defines a new message structure, new messages and cross reference information.

HTML files

The m2cm/messages directory contains replacement HTML files which provide cross-reference links from a message to the MISRA-C:2004 rules.

Global Checks Report

Where coding standard rules impose constraints on the code of the complete project rather than the code in a single translation unit, coding standard enforcement is achieved using the Cross Module Analysis capability of the QA C product. CMA reports violations of project wide rules.

Message Configuration

Message Groups and Levels

In the MISRA-C:2004 document, the rules are arranged in chapters organised on language / coding standard topic.

QA C has 10 levels of messages in its standard configuration. M2CM introduces a level 4, M2CM.

Within Level 4 there are message groups corresponding to the MISRA-C:2004 Rules. A rule may be enforced by one or more QA C messages, by a post-analysis check, by the global checks report or by a combination of these elements. Some of the messages present in Level 4 are intrinsic to QA C and

some are implemented through post-analysis checks supplied with M2CM.

Duplicate Messages

QA C messages are usually reviewed from within the message personality dialog. A QA C message which is associated with M2CM enforcement will be found in Level 4 under the appropriate rule, but it will also be visible elsewhere in its non-M2CM message level. In other words, the composition of other levels is not changed by installing M2CM; relevant messages are simply duplicated within Level 4 under the appropriate rule.

The enforcement of the ISO language standard is a foundational element of QA C. It is implemented using a large number of messages which reside in various groups under levels 6, 7, 8 and 9. These messages check for syntax errors, constraint errors, and language constructs whose results depend on behaviour described as unspecified, undefined or implementation defined.

Some messages within the M2CM configuration may be associated with more than one rule. This occurs because there is not always an exact correspondence between QA C messages and MISRA-C:2004 rules and also because some of the rules include a degree of overlap in the way that they are intended to function. Note that if you disable a message in the message personality, you disable all occurrences of it in any group or level.

Project Configuration

Having installed the supplied software, you will need to make changes to the configuration of existing projects.

Message Personality

Three personality files are supplied with the M2CM package. Of these the message personality `m2cm.p_s` is the most important. By configuring a project with the supplied personality, analysis is reconfigured to use:

1. A subset of the QA C messages appropriate to enforcement of the MISRA-C:2004 rules.
2. A new message structure reflecting the MISRA-C:2004 rules. This is defined by the user message file `qac usr.m2cm`.
3. Post-analysis checks to enforce a number of the MISRA-C:2004 rules.

Compiler and analyser personalities are also supplied as part of the M2CM package. It may be helpful to merge settings from these files into existing personality files.

Uninstalling M2CM-2.2-QAC-7.0

If you wish to uninstall your M2CM-2.2-QAC-7.0 configuration, perform the following actions:

1. Ensure that `$QACPATH` is correctly set
2. `cd $QACPATH`
3. Delete the M2CM installation:

```
rm -rf m2cm
```

Implementation Changes

Secondary Analysis updated to fix CR12406.

PR:QA and the PR:QA logo are registered trademarks of Programming Research.
QA C, QA C++, QA Fortran, QA J and QA Manager are trademarks of Programming Research.
All other products or services mentioned in this document are covered by the trademarks, service marks, or product names as designated by the companies who market those products or services.