



QA C 7.1 Release Notes

March 2008

Contents

1.	OVERVIEW	2
2.	C-99 LANGUAGE FEATURES	2
3.	GCC LANGUAGE EXTENSIONS	2
4.	MISRA-C:2004	3
4.1.	Rule 6.05	3
4.2.	Rule 8.3	3
4.3.	Rule 8.5	3
4.4.	Rule 13.4	3
4.5.	Rule 14.3	3
4.6.	Rule 15.0	3
4.7.	Rule 16.3	3
4.8.	Rule 16.9	3
4.9.	Rule 19.12	4
5.	WINDOWS GUI IMPROVEMENTS	4
5.1.	Relative Path and Environment Variable Support	4
5.2.	Project Operation	4
5.3.	Personality Representation and Saving	4
5.4.	Miscellaneous Improvements	4
6.	MESSAGE CHANGES	5
6.1.	New messages	5
6.2.	Messages relocated to a different message group	7
6.3.	Messages relocated to a different message level	9
7.	CR SUMMARY	10

1. OVERVIEW

The principle features of QAC 7.1 include the following:

- Parsing support for additional selected ISO-C99 language features
- Parsing support for additional selected GCC language extensions.
- Additional messages to improve enforcement of the MISRA-C:2004 coding guidelines.
- Support for environment variables and relative paths in project files and personality files.

2. C-99 language features

Parsing support has been introduced for a number of additional features of the ISO-C99 language. These include

- Variable length arrays
- Compound literals
- Designated Initializers
- Declarations in for statements
- Variadic macros

However it remains a primary focus of QAC to be able to enforce compliance to the ISO:C90 standard and all non-C90 language features are identified with a suitable language extension message. QAC now supports a range of C99 language features as well as other language extensions and so, for the sake of clarity, those features which are consistent with C99 are now classified under a separate level 6 message group called “ISO C99 Language Features”.

The ISO:C99 standard is more generous in the environmental limits which it imposes for strict conformance. A number of additional messages have been implemented to enforce some of these limits rather than the C90 limits. For this reason, the level 6 message group “Conformance limits” has been superseded by 2 new level 6 message groups “ISO C90 Conformance limits” and “ISO C99 Conformance limits”.

3. GCC Language Extensions

Parsing support has been added for various language constructs which are supported in GCC but are not acceptable in either ISO:C90 or ISO:C99. These include

- Named argument lists in variadic macros
- Empty initializers
- Empty struct declarations
- Arrays of zero length

4. MISRA-C:2004

Additional messages have been implemented which provide improved enforcement of the following rules:

4.1. Rule 6.05

Bit-fields of type signed int shall be at least 2 bits long

Message 3665: "Unnamed bit-field consisting of a single bit declared with a signed type."

4.2. Rule 8.3

For each function parameter the type given in the declaration and definition shall be identical, and the return types shall also be identical.

Message 0606: "Object '%s' is declared using typedefs which are different to those in a previous declaration."

Message 0624: "Function '%s' is declared using typedefs which are different to those in a previous declaration."

4.3. Rule 8.5

There shall be no definitions of objects or functions in a header file.

Message 3480: "Object/function '%s', with internal linkage, has been defined in a header file."

4.4. Rule 13.4

The controlling expression of a for statement shall not contain any objects of floating type.

Message 3342: "Control expression of 'for' loop is a floating point comparison."

4.5. Rule 14.3

Before preprocessing, a null statement shall only occur on a line by itself; it may be followed by a comment provided that the first character following the null statement is a white-space character.

Message 3138: "Null statement is located close to other code or comments."

4.6. Rule 15.0

Rule 15.0 does not exist in the original MISRA-C:2004. The new rule number has been adopted as a way of identifying the requirement to conform to the switch statement syntax specified under section 6.15. This syntax is considered normative and is fully enforced in QAC 7.1 with the addition of the following message.

Message 3234: "Declarations precede the first label in this switch construct."

4.7. Rule 16.3

Identifiers shall be given for all of the parameters in a function prototype declaration.

Message 1335: "Parameter identifiers missing in function prototype declaration."

Message 1336: "Parameter identifiers missing in declaration of a function type."

4.8. Rule 16.9

A function identifier shall only be used with either a preceding &, or with a parenthesised parameter list, which may be empty.

Message 3635: "Function identifier used as a pointer without a preceding & operator."

4.9. Rule 19.12

There shall be at most one occurrence of the # or ## preprocessor operators in a single macro definition.

Message 0884: "Order of evaluation of multiple # operators is unspecified."

5. Windows GUI Improvements

5.1. Relative Path and Environment Variable Support

QA C now fully supports relative paths and environment variables in its project file and in personality files attached to the project. This feature greatly enhances the potential for project sharing across teams, storage and retrieval of projects into arbitrary locations, and helps to bind personalities better to their host projects.

The relative path concept is that paths can be saved relative to the location of the project file itself (the root location). Relative paths and environment variables are applied during project saves in a single convenient dialog window.

5.2. Project Operation

To assist in situations where the output path does not exist (for example, where project contents excluding the output directory were saved and restored from the source repository), there is now an option in the 'Project Path Validation' dialog to force-create output paths rather than ask for a replacement location.

As an enhancement to the auto-create functionality, it is now possible to create output path locations that are relative to each source location, rather than creating a sibling folder branch that matches the source branch. This is a needed feature for saving output in certain source repository structures. Also, especially for auto-created projects with lots of empty project folders to prune, there is now a status display showing progress of project folder pruning to ease concerns about the GUI hanging.

(As a separate change, the Unix GUI also supports the creation of output folders matching the source structure.)

5.3. Personality Representation and Saving

We have addressed the possible confusion over making personality file changes that were not attached to the current project. When a personality is saved that was not already attached to the project, a dialog will ask whether to apply this new personality to folders in the current project. Within this dialog, there is a selection mechanism displaying all project folders to assist in this assignment choice.

To further visualize the available personalities, the Configuration | Personalities dialog displays all personalities installed in the product, including compliance installations as well as those assigned to the current project. Various operations are possible, including editing, assigning to the current project, and making a personality the default for new projects.

5.4. Miscellaneous Improvements

- Analysis now automatically detects and takes advantage of dual-processor machines, speeding up file analysis by nearly double on such architectures.
- All browser windows (Relationships, Function Structure, Metrics and Demographics) now contain a menu item to allow easy navigation between browser windows and the main window.
- A new menu item, File | Reopen | Cleanup, will remove any project listings that were not located (which are shown as disabled in the menu list).
- A 30-minute idle time license release feature has been added to the GUI. When enabled and the system (note: not just the application) is idle for 30 minutes, a forced license release is performed to allow other users to capture a license. The GUI user will be presented with an opportunity to reacquire a license, which is of course dependent on a license being available.

6. Message Changes

6.1. New messages

Msg	Message Text	CR
0292	[I] Source file '%s' has comments containing characters which are not members of the basic source character set ('\$', '@' or ''').	11567
0299	[I] Source file '%s' includes #pragma directives containing characters which are not members of the basic source character set.	11106
0320	[C99] Declaration within 'for' statement.	924
0321	[C] Declaration within 'for' statement defines an identifier '%s' which is not an object.	924
0322	[C] Illegal storage class specifier used in 'for' statement declaration.	924
0371	[L] Nesting levels of blocks exceeds 127 - program does not conform strictly to ISO:C99.	910
0372	[L] Nesting levels of conditional inclusion exceeds 63 - program does not conform strictly to ISO:C99.	910
0375	[L] Nesting of parenthesized expressions exceeds 63 - program does not conform strictly to ISO:C99.	12561
0380	[L] Number of macro definitions exceeds 4095 - program does not conform strictly to ISO:C99.	910
0388	[L] '#include "%s"' causes nesting to exceed 15 levels - program does not conform strictly to ISO:C99.	910
0390	[L] Number of members in 'struct' or 'union' exceeds 1023 - program does not conform strictly to ISO:C99.	910
0391	[L] Number of enumeration constants exceeds 1023 - program does not conform strictly to ISO:C99.	12561
0392	[L] Nesting of 'struct' or 'union' types exceeds 63 - program does not conform strictly to ISO:C99.	12561
0606	Object '%s' is declared using typedefs which are different to those in a previous declaration.	11450
0624	Function '%s' is declared using typedefs which are different to those in a previous declaration.	11450
0670	[S] Function defined with invalid function header syntax.	1855
0884	Order of evaluation of multiple # operators is unspecified.	12025
0904	[S] Invalid designator.	918
0940	[C] Illegal usage of a variably modified type.	904
0941	[C] A variable length array may not be initialized.	904
0942	[U] A * can only be used to specify array size within function prototype scope.	904
0943	[C] Jump to label '%s' is a jump into the scope of an identifier with variably modified type.	904
0944	[C] The label '%s' is inside the scope of an identifier with variably modified type.	904
0945	[C99] WARNING. Operand of sizeof has a type which is a variable length array type.	904
1033	[C] The identifier <code>_VA_ARGS_</code> may only be used in the replacement list of a variadic macro.	12161
1034	[E] Macro defined with named variable argument list. This is a language extension.	11276
1035	[E] No macro arguments supplied for variable argument list. This is a language extension.	1501
1036	[E] Comma before <code>##</code> ignored in expansion of variadic macro. This is a language extension.	1501
1037	[E] Arrays of length zero are a language extension.	12294
1038	[E] The sequence <code>"## _VA_ARGS_"</code> is a language extension.	1501
1041	[E] Empty aggregate initializers are a language extension.	10961
1051	[C99] A variable length array has been declared.	904
1052	[C99] A variable length array of unspecified size has been declared.	904
1053	[C99] Designators have been used in this initialization list.	918
1054	[C99] A compound literal has been used.	917
1335	Parameter identifiers missing in function prototype declaration.	11338
1336	Parameter identifiers missing in declaration of a function type.	11338
1690	Null preprocessing directive encountered.	10949
1691	Null preprocessing directive encountered in excluded section of code.	10949
2020	Final switch clause does not end with an explicit 'break' statement.	11541

3138	Null statement is located close to other code or comments.	11355
3234	Declarations precede the first label in this switch construct.	11453
3342	Control expression of 'for' loop is a floating point comparison.	12024
3470	The operand of 'sizeof' is not an expression which designates either an object or a type.	11464
3480	Object/function '%s', with internal linkage, has been defined in a header file.	11876
3635	Function identifier used as a pointer without a preceding & operator.	11454
3665	Unnamed bit-field consisting of a single bit declared with a signed type.	11286
3675	Function parameter declared with type qualification which differs from previous declaration.	11443

6.2. Messages relocated to a different message group

Some existing messages have been relocated to a different message group.

Msg	Old Group	New Group	Message Text
0180	Lang_ext	C99_ext	[C99] Use of ll for conversion specifier.
0268	Constraint	Syntax	[S] Comment open at end of translation unit.
0269	Constraint	Syntax	[S] Unexpected end of file.
0410	ISO_Limits	ISO_C90_Limits	[L] Nesting of parentheses exceeds 32 - program does not conform strictly to ISO:C90.
0428	Maj_Func	Min_Func	Function identifier should probably be followed by ().
0604	Lang_ext	C99_ext	[C99] Declaration appears after statements in a compound statement.
0609	ISO_Limits	ISO_C90_Limits	[L] Number of modifiers in declarator exceeds 12 - program does not conform strictly to ISO:C90.
0611	ISO_Limits	ISO_C90_Limits	[L] Nesting of 'struct' or 'union' types exceeds 15 - program does not conform strictly to ISO:C90.
0612	ISO_Limits	ISO_C90_Limits	[L] Size of object '%s' exceeds 32767 bytes - program does not conform strictly to ISO:C90.
0614	ISO_Limits	ISO_C90_Limits	[L] Number of block scope identifiers exceeds 127 in a block - program does not conform strictly to ISO:C90.
0633	Syntax	Lang_ext	[E] Empty structures and unions are a language extension.
0634	Min_Array	ISO_ImplDef	[I] Bit-field is not declared explicitly as unsigned or signed.
0635	ISO_ExpU	Lang_ext	[E] Bit-fields not of type int, signed int or unsigned int are not supported in ISO-C.
0639	ISO_Limits	ISO_C90_Limits	[L] Number of members in 'struct' or 'union' exceeds 127 - program does not conform strictly to ISO:C90.
0647	ISO_Limits	ISO_C90_Limits	[L] Number of enumeration constants exceeds 127 - program does not conform strictly to ISO:C90.
0715	ISO_Limits	ISO_C90_Limits	[L] Nesting of control structures (statements) exceeds 15 - program does not conform strictly to ISO:C90.
0739	ISO_Limits	ISO_C90_Limits	[L] Number of 'case' labels exceeds 257 - program does not conform strictly to ISO:C90.
0776	ISO_Limits	ISO_C90_Limits	[L] External identifier matches other identifier(s) (e.g. '%s') in first 6 characters - program does not conform strictly to ISO:C90.
0778	ISO_Limits	ISO_C90_Limits	[L] Identifier matches other identifier(s) (e.g. '%s') in first 31 characters - program does not conform strictly to ISO:C90.
0810	ISO_Limits	ISO_C90_Limits	[L] '#include "%s"' causes nesting to exceed 8 levels - program does not conform strictly to ISO:C90.
0815	ISO_Limits	ISO_C90_Limits	[L] '#include <...>' file name does not conform strictly to ISO:C90.
0816	ISO_Limits	ISO_C90_Limits	[L] '#include "..."' file name does not conform strictly to ISO:C90.
0828	ISO_Limits	ISO_C90_Limits	[L] Maximum '#if...' nesting exceeds 8 levels - program does not conform strictly to ISO:C90.
0850	ISO_ExpU	C99_ext	[C99] Macro argument is empty.
0857	ISO_Limits	ISO_C90_Limits	[L] Number of macro definitions exceeds 1024 - program does not conform strictly to ISO:C90.
0858	ISO_Limits	ISO_C90_Limits	[L] Number of macro parameters exceeds 31 - program does not conform strictly to ISO:C90.
0859	ISO_Limits	ISO_C90_Limits	[L] Number of arguments in macro call exceeds 31 - program does not conform strictly to ISO:C90.
0875	ISO_Limits	ISO_C90_Limits	[L] String literal exceeds 509 characters - program does not conform strictly to ISO:C90.
0880	ISO_ExpU	Min_Prepro	Order of evaluation of # and ## operators is unspecified.
0881	ISO_ExpU	Min_Prepro	Order of evaluation of multiple ## operators is unspecified.

0930	Lang_ext	C99_ext	[C99] Trailing comma at the end of an enumerator-list.
1001	Constraint	Lang_ext	[E] '#include %s' is a VMS extension.
1002	Constraint	Lang_ext	[E] '%s' is not a legal identifier in ISO C.
1011	Lang_ext	C99_ext	[C99] Use of C++ comment '//'.
1018	Lang_ext	C99_ext	[C99] Using LL suffix.
1027	Lang_ext	C99_ext	[C99] Using type 'long long'.
1030	Lang_ext	C99_ext	[C99] Macro defined with variable argument list.
1031	Lang_ext	C99_ext	[C99] Initializer for 'struct', 'union' or array type is not a constant expression.
1317	Min_Enum	Maj_Enum	Value not in the enumeration type.
1401	Min_Enum	Maj_Enum	Argument passed is constant enum not in enum type.
1402	Min_Enum	Maj_Enum	Assignment of constant enum not in enum type.
1403	Min_Enum	Maj_Enum	Return of constant enum not in enum type.
1411	Min_Enum	Maj_Enum	Argument passed is a numeric constant but an enum is expected.
1412	Min_Enum	Maj_Enum	Constant expression assigned to an enum.
1413	Min_Enum	Maj_Enum	Return of numeric constant when enum expected.
1421	Min_Enum	Maj_Enum	Argument passed is enum object of different type.
1422	Min_Enum	Maj_Enum	Assignment of enum object of different type.
1423	Min_Enum	Maj_Enum	Return of enum object of different type.
1441	Min_Enum	Maj_Enum	Argument passed is non enum type when enum expected, and may be out of range.
1442	Min_Enum	Maj_Enum	Expression assigned to an enum.
1443	Min_Enum	Maj_Enum	Return of non enum type to enum, and may be out of range.

6.3. Messages relocated to a different message level

Some existing messages have been relocated to a different message level.

Msg	Old Level	New Level	Message Text
0268	8	9	[S] Comment open at end of translation unit.
0269	8	9	[S] Unexpected end of file.
0428	3	2	Function identifier should probably be followed by ().
0633	9	6	[E] Empty structures and unions are a language extension.
0634	2	6	[I] Bit-field is not declared explicitly as unsigned or signed.
0635	7	6	[E] Bit-fields not of type int, signed int or unsigned int are not supported in ISO-C.
0850	7	6	[C99] Macro argument is empty.
0880	7	2	Order of evaluation of # and ## operators is unspecified.
0881	7	2	Order of evaluation of multiple ## operators is unspecified.
1001	8	6	[E] '#include %s' is a VMS extension.
1002	8	6	[E] '%s' is not a legal identifier in ISO C.
1317	2	3	Value not in the enumeration type.
1401	2	3	Argument passed is constant enum not in enum type.
1402	2	3	Assignment of constant enum not in enum type.
1403	2	3	Return of constant enum not in enum type.
1411	2	3	Argument passed is a numeric constant but an enum is expected.
1412	2	3	Constant expression assigned to an enum.
1413	2	3	Return of numeric constant when enum expected.
1421	2	3	Argument passed is enum object of different type.
1422	2	3	Assignment of enum object of different type.
1423	2	3	Return of enum object of different type.
1441	2	3	Argument passed is non enum type when enum expected, and may be out of range.
1442	2	3	Expression assigned to an enum.
1443	2	3	Return of non enum type to enum, and may be out of range.

7. CR Summary

The following table lists summarises the change requests (CRs) that have been implemented in QAC 7.1.

CRs have been categorised into 3 types (column "T"):

C – A significant change has been implemented to existing behaviour.

F – A fix of a bug or problem feature.

N – New functionality has been introduced.

Some CRs are associated with more than one category and in some cases, the distinction between categories is a little indistinct. The classification should therefore be treated as a guide only.

CR	T	Resolution
904	N	Implementation of support for variable array dimensions as defined in ISO C99. 0940: [C] <i>Illegal usage of a variably modified type.</i> 0941: [C] <i>A variable length array may not be initialized.</i> 0942: [U] <i>A * can only be used to specify array size within function prototype scope.</i> 0943: [C] <i>Jump to label '%s' is a jump into the scope of an identifier with variably modified type.</i> 0944: [C] <i>The label '%s' is inside the scope of an identifier with variably modified type.</i> 0945: [C99] <i>WARNING. Operand of sizeof has a type which is a variable length array type.</i> 1051: [C99] <i>A variable length array has been declared.</i> 1052: [C99] <i>A variable length array of unspecified size has been declared.</i>
910	N	Implementation of support for selected ISO C99 environmental limits. 0371: [L] <i>Nesting levels of blocks exceeds 127 - program does not conform strictly to ISO:C99.</i> 0372: [L] <i>Nesting levels of conditional inclusion exceeds 63 - program is non-conforming.</i> 0380: [L] <i>Number of macro definitions exceeds 4095 - program is non-conforming.</i> 0388: [L] <i>#include "%s" causes nesting to exceed 15 levels - program is non-conforming.</i> 0390: [L] <i>Number of members in 'struct' or 'union' exceeds 1023 - program is non-conforming.</i>
917	N	Implementation of support for compound literals as defined in ISO C99. 1054: [C99] <i>A compound literal has been used.</i>
918	N	Implementation of support for designated initializers as defined in ISO C99. 1053: [C99] <i>Designators have been used in this initialization list.</i> 0904: [S] <i>Invalid designator.</i>
924	N	Implementation of support for declarations in clause 1 of a for statement as permitted in ISO C99. 0320: [C99] <i>Declaration within 'for' statement.</i> 0321: [C] <i>Declaration within 'for' statement defines an identifier '%s' which is not an object.</i> 0322: [C] <i>Illegal storage class specifier used in 'for' statement declaration.</i>
927	N	Implementation of support for macros with a variable number of arguments as permitted in ISO C99.
938	N	An empty argument supplied to a macro is now parsed and interpreted according to ISO-C99. Message 0850 has been modified accordingly. 0850: [C99] <i>Macro argument is empty.</i>
1258	N	Recognise and apply Environment Variables and Relative Paths in project and personality files.
1501	N	Implementation of support for alternative specification of variadic macros using a named argument list. 1035: [E] <i>No macro arguments supplied for variable argument list. This is a language extension.</i> 1036: [E] <i>Comma before ## ignored in expansion of variadic macro.</i> 1038: [E] <i>The sequence '## __VA_ARGS__' is a language extension.</i>
1855	F	New message introduced to identify faulty syntax in the definition of a function. 0670: [S] <i>Function defined with invalid function header syntax.</i>
1876	F	3406: <i>Object/function '%s', with external linkage, has been defined in a header file.</i> Message 3406 was generated incorrectly on a tentative definition in a header file which was followed by an explicit definition in a source file.
10949	N	New messages introduced to identify null preprocessor directives (i.e. #) 1690: <i>Null preprocessing directive encountered.</i> 1691: <i>Null preprocessing directive encountered in excluded section of code.</i>

CR	T	Resolution
10961	N	Implementation of support for a language extension which permits empty initializers. <i>1041: [E] Empty aggregate initializers are a language extension.</i>
10975	F	3229: File scope static, '%s', is written but never used. Message 3229 was not generated when the object was only referenced in a single function.
10994	F	In situations where the initial declaration of a type was incomplete, QAC sometimes failed to recognize when the type became complete through a subsequent declaration. This resulted in spurious messages, for example: 0478: [C] 'sizeof' cannot be applied to an object of unknown size.
10996	F	0495: [C] Left operand of '%', '<<', '>>', '&', '^' or ' ' must have integral type. 0496: [C] Right operand of '%', '<<', '>>', '&', '^' or ' ' must have integral type. 0560: [C] Left operand of '<=<', '>=>', '&=', ' =', '^=' or '%=' must have integral type. 0559: [C] Right operand of '<=<', '>=>', '&=', ' =', '^=' or '%=' must have integral type. Messages 0495 and 0496 were sometimes being issued in place of messages 0560 and 0559
11007	C	Unix GUI: Correct a bug exhibited when multiple file selection is made in all GUI browsers using Ctrl & Shift & End.
11013	F	0605: [C] A declaration must declare a tag or an identifier. 3313: No definition has been found for structure/union tag '%s'. Message 3313 was sometimes issued instead of 0605 when encountering an empty declaration at file scope.
11044	F	1279: An L" suffix has been appended to a hex constant." Message 1279 was issued incorrectly when parsing a hex constant within a #if or #elif preprocessing directive.
11106	N	0289: [I] Source file '%s' has preprocessing tokens containing characters which are not members of the basic source character set. A new message 0299 is now generated instead of message 0289 when non-standard characters are encountered in a #pragma directive. 0299: [I] Source file '%s' includes #pragma directives containing characters which are not members of the basic source character set.
11131	C	336: Macro defined as an octal constant. 339: Octal constant used According to the strict definition of the ISO-C language standard, any integer literal starting with 0 is octal rather than decimal, however we choose not to generate messages 336 and 339 for integer constants of value zero and by implication treat them as decimal constants. In QAC 7.1, a literal constant consisting of a single zero is treated as decimal but literal constants consisting of <i>multiple zero's</i> are now considered octal in nature and will trigger generation of messages 336 and 339.
11158	C	Message 2468 is no longer generated for a loop control variable which is declared volatile. 2468: Control variable, %s, with file scope, not modified inside loop.
11187	C N	The scope of message suppression implemented through #PRQA_MESSAGES_OFF is normally limited to the current file and other files which are included from the current file. This behaviour has been modified in the case of a forceinclude file so that suppression can be applied to the translation unit using suppression #pragma's located in a forceinclude file.
11191	F	Unix GUI: Metric Browser correctly reads larger met file.
11276	N	Support has been implemented for variadic macros defined with a named variable argument list. <i>1034: [E] Macro defined with named variable argument list. This is a language extension.</i>
11286	F	New message implemented to identify unnamed bit-fields of a signed type consisting of a single bit. <i>3665: Unnamed bit-field consisting of a single bit declared with a signed type.</i>
11302	F	Message 309 was not generated when casting a function pointer to an integer type of insufficient size. <i>0309: [U] Integral type is not large enough to hold a pointer value.</i>
11338	N	New QAC messages implemented to identify function prototypes which include unnamed parameters. <i>1335: Parameter identifiers missing in function declaration.</i> <i>1336: Parameter identifiers missing in function type declaration.</i>

CR	T	Resolution
11355	N	New message implemented to identify null statements which violate MISRA-C2 Rule 14.3 <i>3138: Null statement is located close to other code or comments.</i>
11358	F	Message 3396 generated incorrectly when a cast is applied to an operand of a ternary operator. <i>3396: Ternary operator used with another binary operator. Extra parentheses recommended.</i>
11389	F	Configuration of type 'char' with a size greater than 8 triggered spurious messages.
11404	C	Spurious function metric values (AVGS and STMT) have been removed from the met file
11440	F	<i>0883: Include file code is not protected against repeated inclusion</i> Message 883 was generated incorrectly when the header file contained a more complex conditional inclusion structure. Superseded by CR 12647.
11443	N	QAC does not identify when a parameter in a function declaration is differently qualified to the type in the function definition. <i>3675: Function parameter declared with type qualification which differs from previous declaration.</i>
11450	N	New messages have been implemented to identify when multiple declarations of the same object or function are not typedef compatible. <i>0606: Declaration of '%s' is not typedef compatible with previous declaration.</i> <i>0624: Declaration of function '%s()' is not typedef compatible with previous declaration.</i>
11453	N	New message implemented to identify declarations at the start of a switch statement. <i>3234: Declarations precede the first label in this switch construct.</i>
11454	N	New message implemented to identify use of a function name as a pointer when not preceded by the & operator. <i>3635: Function identifier used as a pointer without a preceding & operator.</i> Message 0428 has been reclassified as a level 2 message. <i>0428: Function identifier should probably be followed by ().</i>
11464	N	New message implemented to identify an operand to the sizeof operator which is neither an object nor a type. <i>3470: The operand of 'sizeof' is not an expression which designates either an object or a type.</i>
11530	N	Use of the sizeof operator in a preprocessing expression now works with an operand formed from a typedef.
11541	N	New message to identify when the final clause of a switch statement is not terminated with a break statement. <i>2020: Final switch clause does not end with an explicit 'break' statement.</i>
11567	N	<i>0288: [I] Source file '%s' has comments containing characters which are not members of the basic source character set.</i> Message 0288 is normally generated when non-standard characters are encountered in comments. In future a new message, 0292, will be generated instead, when either of three specific characters '\$', '@' or "'" is encountered in a comment. <i>0292: [I] Source file '%s' has comments containing characters which are not members of the basic source character set ('\$ ', '@' or '"').</i>
11591	C	The '-nomsg' configuration option has been changed so that it is applied to the resulting set of messages after the '-only' configuration option has been applied.
11628	N	Unix GUI: Ordering of files in the project can be changed by selecting the column heading.
11672	N	Unix GUI: Auto-create project and sub-folder includes the 'Replicate Source Tree Structure in Output Paths' option.
11708	C	Windows GUI: A warning dialog is now raised if attempting to open an invalid project file.
11717	F	<i>0410: [L] Nesting of parenthesized expressions exceeds 63 - program does not conform strictly to ISO:C99.</i> Message 0410 was not generated correctly for some types of expression.
11740	F	<i>3628: Octal escape sequences used in a character constant or string literal.</i> Certain octal escape sequences were not identified with message 3628.
11746	C	Unix GUI: External editor can now be used for all file view operations.
11816	F	<i>0883: Include file code is not protected against repeated inclusion</i> Message 883 was generated incorrectly when a protected header file was included more than once using a different (but equivalent) path specification. CR superseded by CR 12647.
11839	C	Unix GUI: HTML internal viewer has been removed and is replaced by Firefox as default.

CR	T	Resolution
11845	N	Parsing support has been added for a C language extension which allows structure and union types to be defined with no members. Message 0633 has been reworded <i>0633: [E] Empty structures and unions are a language extension.</i>
11862	F	QAC failed to generate an appropriate message when incorrect syntax was encountered in a #pragma. New message implemented: <i>3104: #pragma '%s' has invalid arguments and has been ignored.</i>
11876	N	New message implemented to identify the definition of an object or function with internal linkage in a header file. <i>3480: Object/function '%s', with internal linkage, has been defined in a header file.</i>
11908	F	<i>2003: The preceding non-empty 'case' or 'default' clause does not end with an explicit 'break' or 'return' statement.</i> Message 2003 was not generated on the final clause in a switch statement if the switch statement was itself nested.
11040	C	Unix GUI: Permit spaces in file names
11981	F	<i>2003: The preceding non-empty 'case' or 'default' clause does not end with an explicit 'break' or 'return' statement.</i> Message 2003 was not generated following a non-empty switch clause with no break statement if the code did not result in side-effects.
12020	C	Both GUIs: 'Collapse Folders' menu function now raises a confirmation dialog as an irreversible action.
12024	N	New message implemented to identify floating point operations in the controlling expression of a "for loop". <i>3342: Control expression of 'for' loop is a floating point comparison.</i>
12025	N	New message implemented to identify usage of multiple # operators in a macro. <i>0884: Order of evaluation of multiple # operators is unspecified.</i>
12091	N	Windows GUI: Concurrent analysis on a dual processor machine is now supported in the GUI.
12095	F	Unix GUI: Fixed failure when file is removed from the project while in the analysis queue.
12113	F	Fix for bug in the handling of message suppressions.
12139	F	The column location in <LIT> records in the met file was incorrectly numbered from zero instead of one.
12142	F	Improvements to the text and associated help pages associated with a number of messages.
12161	N	New message implemented to identify misuse of the identifier __VA_ARGS__ . <i>1033: [C] The identifier __VA_ARGS__ may only be used in the replacement list of a variadic macro.</i>
12271	F	<i>0309: [U] Integral type is not large enough to hold a pointer value.</i> Message 0309 was not generated when converting a function pointer.
12278	C	Unix GUI: Generate .txt and .html files on calling errdsp, mirroring Windows functionality.
12290	F	Unix GUI: Relationship browser mishandled recursive function calls.
12294	N	Parsing support has been added for arrays of zero length – a language extension. <i>1037: [E] Arrays of length zero are a language extension.</i>
12295	F	<i>1031: [E] Initializer for 'struct', 'union' or array type is not a constant expression.</i> Message 1031 was not always generated as expected
12313	F	Unix GUI: xqac terminated when a subfolder was deleted during analysis.
12314	F	Unix GUI: In some cases, not possible to add "Metric Values" in Function Metrics Browser.
12321	C	Windows GUI: In the internal ASCII Text Viewer font dialog, the Apply button had no effect and is removed.
12325	C	Configuration Option 'ForceInclude' now caters for multiple entries.
12339	F	Macro replacement sometimes performed incorrectly when substituting an empty macro argument.
12370	F	When an object was defined in a header file with a tentative definition, the "inc" field in the <DEFINE> DI record was incorrectly set to "N" instead of "I" and the record was not suppressed by applying the -q option.
12429	C	Windows GUI: Common temporary files in the GUI (settings.via and filelist.lst) are now replaced with unique file names for each process, although retained afterwards as residue with the above names for convenience.

CR	T	Resolution
12430	C	Windows GUI: Extension of the auto-create functionality to permit output paths relative to each source folder.
12431	N	Windows GUI: After editing and saving a personality, a dialog will ask whether to apply this to folders in the current project.
12432	N	Windows GUI: Relationships, Function Structure, and Metrics windows operate in pseudo-MDI mode, with a Window menu item to navigate between them.
12437	F	Fix for problem in dataflow analysis which resulted in a program hang under some specific conditions.
12447	F	The line number specified in the metfile DEFINE record for a function parameter named in the declaration but unnamed in the subsequent definition was incorrect.
12455	F	Windows GUI: Correction of a bug where the file list remained active in Relationships/Structure/Metrics browsers after pressing Cancel.
12456	N	Windows GUI: A new menu item added under File Reopen to cleanup (remove from the list) non-existent projects from the Recent list.
12471	C	Windows GUI: Offer to force-create output paths rather than attempt a path substitution when loading a project.
12485	C	Windows GUI: Handle (escape) the '\$' character in paths for project configuration.
12486	C	Redesign and modernise the Windows GUI folder selection dialog.
12490	C	Unix GUI: Add %R option to custom reports.
12513	C	Unix GUI: External Editor allows all settings to be configurable.
12531	N	Windows GUI: Display both product installation and added compliance personality folders in the Configuration Personalities dialog.
12532	N	Windows GUI: Add configuration and timing logic for a 30-minute idle license release.
12533	F	<i>2003: The preceding non-empty 'case' or 'default' clause does not end with an explicit 'break' or 'return' statement.</i> Message 2003 is no longer generated if the case clause is terminated with a call to a non-returning function (as specified with #pragma PRQA_NOT_RETURN).
12540	C	Windows GUI: Provide running status for the auto-create function. This is to provide the user with details of empty project folders that are being pruned, and avoid the suspicion that the GUI is hanging.
12561	N	C99: Implementation of additional environmental limits <i>0375: [L] Nesting of parenthesized expressions exceeds 63 - program does not conform strictly to ISO:C99.</i> <i>0391: [L] Number of enumeration constants exceeds 1023 - program does not conform strictly to ISO:C99.</i> <i>0392: [L] Nesting of 'struct' or 'union' types exceeds 63 - program does not conform strictly to ISO:C99.</i>
12577	F	<i>0268: [S] Comment open at end of translation unit.</i> <i>0269: [S] Unexpected end of file.</i> Messages 0268 and 0269 are now classified as syntax errors.
12647	F	<i>0883: Include file code is not protected against repeated inclusion</i> The implementation of message 0883 has been redesigned in order to address the problems identified in CR 11440 and CR 11816.
12648	F	The behaviour of the '-ForceInclude' configuration option has been changed. In the absence of an absolute path specification, the location of the file is now assumed to be relative to the "current directory" rather than the directory containing the source file.
12705	F	<i>1512: Identifier '%s' with external linkage has separate declarations in multiple translation units.</i> CMA Message 1512 was sometimes generated incorrectly.