autowarefoundation / autoware Public

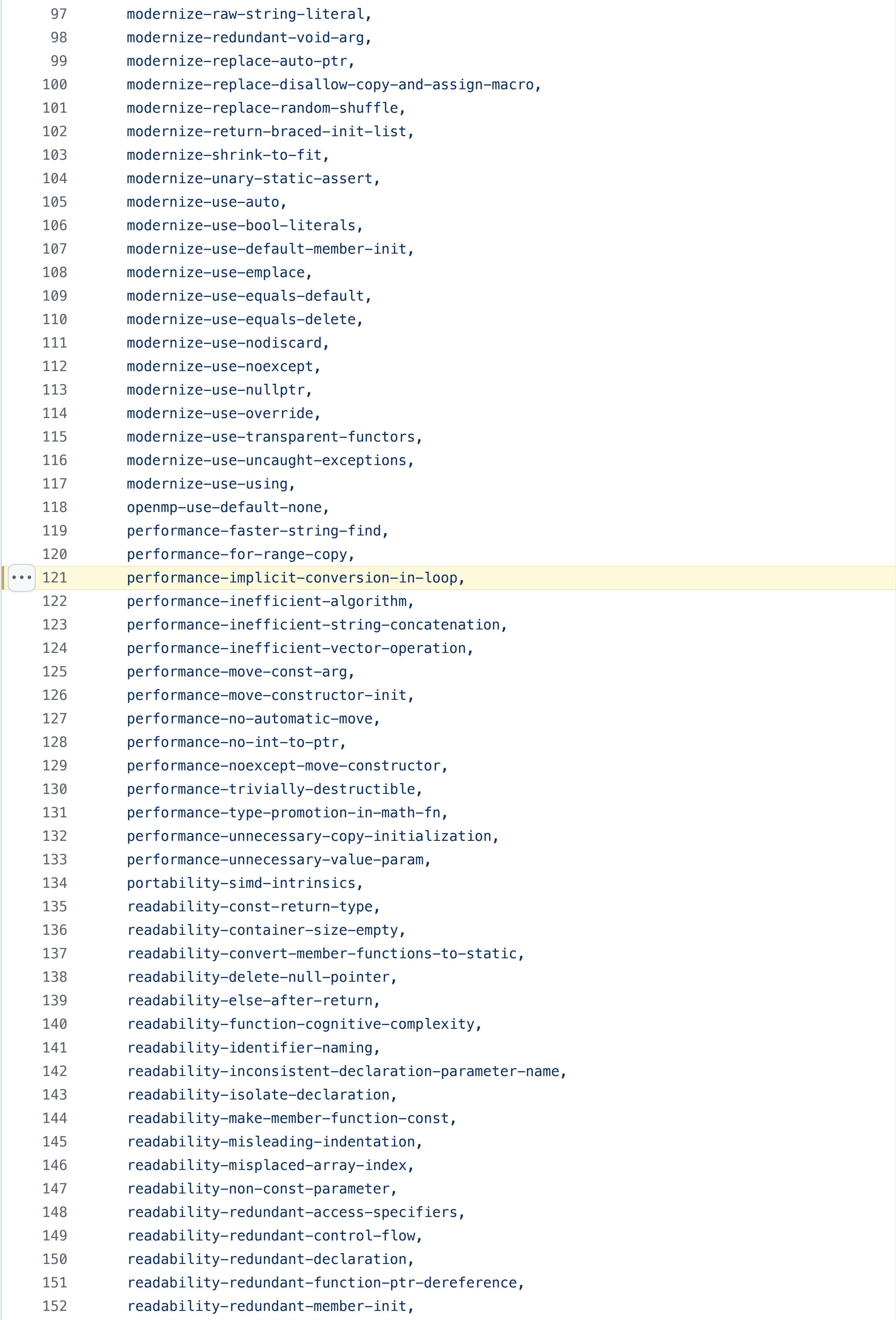
Code Issues 42 Pull requests 13 Discussions Actions Security ···

497c9c4 autoware / .clang-tidy Go to file ···

yhisaki feat(.clang-tidy): add ExtraArgs (#5439) ··· ✓ 3591106 · last week

462 lines (457 loc) · 16.2 KB ·

```
Checks: "
  -*,
  boost-use-to-string,
  bugprone-argument-comment,
  bugprone-assert-side-effect,
  bugprone-bad-signal-to-kill-thread,
  bugprone-bool-pointer-implicit-conversion,
  bugprone-branch-clone,
  bugprone-copy-constructor-init,
  bugprone-dangling-handle,
  bugprone-dynamic-static-initializers,
  bugprone-exception-escape,
  bugprone-fold-init-type,
  bugprone-forward-declaration-namespace,
  bugprone-forwarding-reference-overload,
  bugprone-inaccurate-erase,
  bugprone-incorrect-roundings,
  bugprone-integer-division,
  bugprone-lambda-function-name,
  bugprone-macro-parentheses,
  bugprone-macro-repeated-side-effects,
  bugprone-misplaced-operator-in-strlen-in-alloc,
  bugprone-misplaced-widening-cast,
  bugprone-move-forwarding-reference,
  bugprone-multiple-statement-macro,
  bugprone-not-null-terminated-result,
  bugprone-parent-virtual-call,
  bugprone-posix-return,
  bugprone-signed-char-misuse,
  bugprone-sizeof-container,
  bugprone-sizeof-expression,
  bugprone-string-constructor,
  bugprone-string-integer-assignment,
  bugprone-string-literal-with-embedded-nul,
  bugprone-suspicious-enum-usage,
  bugprone-suspicious-memset-usage,
  bugprone-suspicious-missing-comma,
  bugprone-suspicious-semicolon,
  bugprone-suspicious-string-compare,
  bugprone-swapped-arguments,
    bugprone-terminating-continue,
    bugprone-throw-keyword-missing,
    bugprone-too-small-loop-variable,
    bugprone-unchecked-optional-access,
    bugprone-undefined-memory-manipulation,
    bugprone-undelegated-constructor,
    bugprone-unhandled-self-assignment,
    bugprone-unused-raii,
    bugprone-unused-return-value,
    bugprone-use-after-move,
    bugprone-virtual-near-miss,
    cppcoreguidelines-avoid-goto,
    cppcoreguidelines-init-variables,
    cppcoreguidelines-interfaces-global-init,
    cppcoreguidelines-macro-usage,
    cppcoreguidelines-narrowing-conversions,
    cppcoreguidelines-no-malloc,
    cppcoreguidelines-pro-bounds-pointer-arithmetic,
    cppcoreguidelines-pro-type-const-cast,
    cppcoreguidelines-pro-type-cstyle-cast,
    cppcoreguidelines-pro-type-member-init,
    cppcoreguidelines-pro-type-reinterpret-cast,
    cppcoreguidelines-pro-type-static-cast-downcast,
    cppcoreguidelines-pro-type-union-access,
    cppcoreguidelines-slicing,
    cppcoreguidelines-special-member-functions,
    google-build-explicit-make-pair,
    google-build-namespaces,
    google-build-using-namespace,
    google-explicit-constructor,
    google-global-names-in-headers,
    google-upgrade-googletest-case,
    hicpp-exception-baseclass,
    hicpp-multiway-paths-covered,
    hicpp-no-assembler,
    hicpp-signed-bitwise,
    llvm-namespace-comment,
    misc-definitions-in-headers,
    misc-misplaced-const,
    misc-new-delete-overloads,
    misc-non-copyable-objects,
    misc-redundant-expression,
    misc-static-assert,
    misc-throw-by-value-catch-by-reference,
    misc-unconventional-assign-operator,
    misc-uniqueptr-reset-release,
    misc-unused-alias-decls,
    misc-unused-parameters,
    misc-unused-using-decls,
    modernize-concat-nested-namespaces,
    modernize-deprecated-headers,
    modernize-deprecated-ios-base-aliases,
    modernize-loop-convert,
    modernize-make-shared,
    modernize-make-unique,
    modernize-pass-by-value,
```



```
    modernize-raw-string-literal,
    modernize-redundant-void-arg,
    modernize-replace-auto-ptr,
    modernize-replace-disallow-copy-and-assign-macro,
    modernize-replace-random-shuffle,
    modernize-return-braced-init-list,
    modernize-shrink-to-fit,
    modernize-unary-static-assert,
    modernize-use-auto,
    modernize-use-bool-literals,
    modernize-use-default-member-init,
    modernize-use-emplace,
    modernize-use-equals-default,
    modernize-use-equals-delete,
    modernize-use-nodiscard,
    modernize-use-noexcept,
    modernize-use-nullptr,
    modernize-use-override,
    modernize-use-transparent-functors,
    modernize-use-uncaught-exceptions,
    modernize-use-using,
    openmp-use-default-none,
    performance-faster-string-find,
    performance-for-range-copy,
    performance-implicit-conversion-in-loop,
    performance-inefficient-algorithm,
    performance-inefficient-string-concatenation,
    performance-inefficient-vector-operation,
    performance-move-const-arg,
    performance-move-constructor-init,
    performance-no-automatic-move,
    performance-no-int-to-ptr,
    performance-noexcept-move-constructor,
    performance-trivially-destructible,
    performance-type-promotion-in-math-fn,
    performance-unnecessary-copy-initialization,
    performance-unnecessary-value-param,
    portability-simd-intrinsics,
    readability-const-return-type,
    readability-container-size-empty,
    readability-convert-member-functions-to-static,
    readability-delete-null-pointer,
    readability-else-after-return,
    readability-function-cognitive-complexity,
    readability-identifier-naming,
    readability-inconsistent-declaration-parameter-name,
    readability-isolate-declaration,
    readability-make-member-function-const,
    readability-misleading-indentation,
    readability-misplaced-array-index,
    readability-non-const-parameter,
    readability-redundant-access-specifiers,
    readability-redundant-control-flow,
    readability-redundant-declaration,
    readability-redundant-function-ptr-dereference,
    readability-redundant-member-init,
    readability-redundant-smartptr-get,
    readability-redundant-string-cstr,
    readability-redundant-string-init,
    readability-simplify-boolean-expr,
    readability-simplify-subscript-expr,
    readability-static-accessed-through-instance,
    readability-static-definition-in-anonymous-namespace,
    readability-string-compare,
    readability-uniqueptr-delete-release"

  WarningsAsErrors: "
    boost-use-to-string,
    bugprone-dangling-handle,
    bugprone-fold-init-type,
    bugprone-inaccurate-erase,
    bugprone-incorrect-roundings,
    bugprone-misplaced-widening-cast,
    bugprone-sizeof-container,
    bugprone-sizeof-expression,
    bugprone-string-constructor,
    bugprone-suspicious-enum-usage,
    bugprone-suspicious-memset-usage,
    bugprone-suspicious-missing-comma,
    bugprone-suspicious-semicolon,
    bugprone-swapped-arguments,
    bugprone-unused-raii,
    bugprone-use-after-move,
    llvm-namespace-comment,
    misc-non-copyable-objects,
    misc-redundant-expression,
    misc-throw-by-value-catch-by-reference,
    misc-unused-alias-decls,
    misc-unused-parameters,
    misc-unused-using-decls,
    modernize-deprecated-headers,
    modernize-redundant-void-arg,
    modernize-use-bool-literals,
    modernize-use-emplace,
    modernize-use-equals-default,
    modernize-use-equals-delete,
    modernize-use-nullptr,
    modernize-use-override,
    modernize-use-using,
    performance-faster-string-find,
    performance-inefficient-algorithm,
    readability-make-member-function-const,
    readability-misleading-indentation,
    readability-misplaced-array-index,
    readability-string-compare"

  HeaderFilterRegex: ^(?!\/usr)(?!\/opt)
```

```yaml

CheckOptions:
  - key: bugprone-argument-comment.CommentBoolLiterals
    value: "0"
  - key: bugprone-argument-comment.CommentCharacterLiterals
    value: "0"
  - key: bugprone-argument-comment.CommentFloatLiterals
    value: "0"
  - key: bugprone-argument-comment.CommentIntegerLiterals
    value: "0"
  - key: bugprone-argument-comment.CommentNullPtrs
    value: "0"
  - key: bugprone-argument-comment.CommentStringLiterals
    value: "0"
  - key: bugprone-argument-comment.CommentUserDefinedLiterals
    value: "0"
  - key: bugprone-argument-comment.IgnoreSingleArgument
    value: "0"
  - key: bugprone-argument-comment.StrictMode
    value: "0"
  - key: bugprone-assert-side-effect.AssertMacros
    value: assert
  - key: bugprone-assert-side-effect.CheckFunctionCalls
```

```
  - key: modernize-use-override.FinalSpelling
    value: final
  - key: modernize-use-override.IgnoreDestructors
    value: "0"
  - key: modernize-use-override.OverrideSpelling
    value: override
  - key: modernize-use-transparent-functors.SafeMode
    value: "0"
  - key: modernize-use-using.IgnoreMacros
    value: "1"
  - key: performance-faster-string-find.StringLikeClasses
    value: std::basic_string
  - key: performance-for-range-copy.AllowedTypes
    value: ""
  - key: performance-for-range-copy.WarnOnAllAutoCopies
    value: "0"
  - key: performance-inefficient-string-concatenation.StrictMode
    value: "0"
  - key: performance-inefficient-vector-operation.EnableProto
    value: "0"
  - key: performance-inefficient-vector-operation.VectorLikeClasses
    value: ::std::vector
  - key: performance-move-const-arg.CheckTriviallyCopyableMove
    value: "1"
  - key: performance-move-constructor-init.IncludeStyle
    value: google
  - key: performance-no-automatic-move.AllowedTypes
    value: ""
  - key: performance-type-promotion-in-math-fn.IncludeStyle
    value: google
  - key: performance-unnecessary-copy-initialization.AllowedTypes
    value: ""
  - key: performance-unnecessary-value-param.AllowedTypes
    value: .*Ptr;.*SharedFuture
  - key: performance-unnecessary-value-param.IncludeStyle
    value: google
  - key: portability-simd-intrinsics.Std
    value: ""
  - key: portability-simd-intrinsics.Suggest
    value: "0"
  - key: readability-function-cognitive-complexity.IgnoreMacros
    value: "1"
  - key: readability-else-after-return.WarnOnUnfixable
    value: "1"
  - key: readability-identifier-naming.NamespaceCase
```

```yaml
    value: lower_case
  - key: readability-identifier-naming.ClassCase
    value: CamelCase
  - key: readability-identifier-naming.PrivateMemberSuffix
    value: _
  - key: readability-identifier-naming.StructCase
    value: CamelCase
  - key: readability-identifier-naming.FunctionCase
    value: lower_case
  - key: readability-identifier-naming.VariableCase
    value: lower_case
  - key: readability-identifier-naming.GlobalConstantCase
    value: lower_case
  - key: readability-identifier-naming.GlobalConstantPrefix
    value: g_
  - key: readability-identifier-naming.ConstexprVariableCase
    value: lower_case
  - key: readability-inconsistent-declaration-parameter-name.IgnoreMacros
    value: "1"
  - key: readability-inconsistent-declaration-parameter-name.Strict
    value: "0"
  - key: readability-redundant-smartptr-get.IgnoreMacros
    value: "1"
  - key: readability-redundant-string-init.StringNames
    value: ::std::basic_string
  - key: readability-simplify-subscript-expr.Types
    value: ::std::basic_string;::std::basic_string_view;::std::vector;::std::array
  - key: readability-static-accessed-through-instance.NameSpecifierNestingThreshold
    value: "3"
```