C++ Standard Adaptive Features

Cpp11

- required_features :
  - constexpr limited
  - alignas 16
  - static_assert
  - enable_if
  - variadic_templates
  - range_based_for
  - is_trivially_copyable
  - numeric_limits epsilon

macros :
MATRIX_CONSTEXPR
MATRIX_NODISCARD empty
MATRIX_INLINE_VAR static constexpr
MATRIX_BIT_CAST memcpy based

Fully compatible baseline.
Provides fallback implementations
for newer features.

upgrade

Cpp14

- new_features :
  - relaxed_constexpr
  - generic_lambdas
  - binary_literals

macros :
MATRIX_CONSTEXPR14 constexpr

benefits :
more compile time operations
constexpr loops and conditionals
better optimization

Detection:
if __cplusplus >= 201402L

Impact:
small performance improvements
for compile time computations.

upgrade

Cpp17

- new_features :
  - nodiscard attribute
  - if_constexpr
  - inline_variables
  - structured_bindings

macros :
MATRIX_NODISCARD nodiscard
MATRIX_CONSTEXPR17 constexpr
MATRIX_INLINE_VAR inline
MATRIX_IF_CONSTEXPR if constexpr

benefits :
compile time branching
reduced code bloat
better warnings
improved metaprogramming

Detection:
if __cplusplus >= 201703L

Impact:
code size reduction
modest performance gains.

upgrade

Cpp20

- new_features :
  - bit_cast
  - constexpr_vector
  - likely_unlikely
  - consteval
  - constinit
  - concepts

macros :
MATRIX_CONSTEXPR20 constexpr
MATRIX_CONSTEVAL consteval
MATRIX_CONSTINIT constinit
MATRIX_LIKELY likely
MATRIX_UNLIKELY unlikely
MATRIX_BIT_CAST bit_cast

benefits :
safe type conversions
guaranteed compile time evaluation
branch prediction hints
type constraints
reduced undefined behavior

Detection:
if __cplusplus >= 202002L

Impact:
improved performance
better code generation
sanitizer friendly.

upgrade

Cpp23

- new_features :
  - if_consteval
  - multidimensional_subscript
  - extended_floating_point

macros :
MATRIX_CONSTEXPR23 constexpr
MATRIX_IF_CONSTEVAL

benefits :
better compile time detection
improved metaprogramming
enhanced type safety

Detection:
if __cplusplus >= 202302L

Impact:
incremental improvements
better compile time diagnostics.

upgrade

Cpp26

- new_features :
  - constexpr_sin
  - constexpr_cos
  - constexpr_sqrt
  - constexpr_cmath

macros :
MATRIX_CONSTEXPR26 constexpr
MATRIX_CONSTEXPR_TRIG constexpr

benefits :
compile time rotation matrices
arbitrary angle transforms
zero runtime trig cost

Detection:
if __cplusplus >= 202600L

Impact:
performance gains for
rotation heavy applications.