

Wording for `[[nodiscard]]` attribute.

Document No.: P0189R1

Revises: P0189R0

Project: Programming Language C++

Audience: Core

Author: Andrew Tomazos <andrewtomazos@gmail.com>

Date: 2016-02-29

Summary

A wording for the `[[nodiscard]]` attribute described in the previous revision is proposed for application to the C++17 working draft.

Wording

7.6.6 Nodiscard attribute

`[dcl.attr.nodiscard]`

1. The *attribute-token* `nodiscard` may be applied to the *declarator-id* in a function declaration or to the declaration of a class or enumeration. It shall appear at most once in each *attribute-list* and no *attribute-argument-clause* shall be present.
2. [Note: A nodiscard call is a function call expression that calls a function previously declared `nodiscard`, or whose return type is a possibly cv-qualified class or enumeration type marked `nodiscard`. Appearance of a nodiscard call as a potentially-evaluated discarded-value expression (Clause 5) is discouraged unless explicitly cast to `void`. Implementations are encouraged to issue a warning in such cases. This is typically because discarding the return value of a nodiscard call has surprising consequences. -- end note]

3. [Example:

```
struct [[nodiscard]] error_info { /*...*/ };
error_info enable_missile_safety_mode();
void launch_missiles();
void test_missiles() {
    enable_missile_safety_mode(); // warning encouraged
    launch_missiles();
}
```

```
error_info &foo();  
void f() { foo(); } // reference type, warning not  
encouraged  
-- end example]
```