

noreturn

Chapter 1 Safe Features

Misuse of `[[noreturn]]` on function pointers

Although the `[[noreturn]]` attribute is permitted to syntactically pertain to a function pointer for the benefit of external tools, it has no effect in standard C++; fortunately, most compilers will issue a warning:

```
void (*fp [[noreturn]])(); // no effect in standard C++; will likely warn
```

What’s more, assigning the address of a function that is not decorated with `[[noreturn]]` to an otherwise suitable function pointer that is so decorated is perfectly fine:

```
void f() { return; }; // function that always returns

void g()
{
    fp = f; // [[noreturn]] on fp is silently ignored.
}
```

Any reliance on `[[noreturn]]` to have any effect in standard C++ when applied to other than a function’s declaration is misguided.

See Also

- “Attribute Syntax” (§1.1, p. 12) explains that `[[noreturn]]` is a built-in attribute that follows the general syntax and placement rules of C++ attributes.

Further Reading

- The original proposal for this feature — elucidating its rationale and history — is presented in [svoboda10](#).
- Herb Sutter opines on this attribute being one of comparative few that cannot be ignored in [sutter12](#).