

# Slay Some Earthly Demons X

Document: n3409 Author: Martin Uecker Date: 2024-12-07

**Undefined Behavior:** Use of the value of a void expressions is UB. (J.2 (20), N3301)

**Analysis:** One can not do much with a void expression, so most cases which are made UB are already constraint violations elsewhere. Simple assignment constrains the types that can be used (and this covers other cases such as return). That void is an incomplete type also rules out many other uses (e.g, sizeof). Casts require a scalar type (except casts to void).

Lvalue conversion is UB for incomplete types, but this will be addressed elsewhere.

Possibilities of legitimately using void expressions (but not their value) are as operands to the cast, `_Generic`, and `typeof` operators. All seem to be ok and have defined behavior.

## Examples:

```
extern void x;

void foo(void)
{
    (void)(void)1;
    x;
    _Generic(x, void: 1);
    _Generic(x, typeof(x): 1);
}
```

Godbolt Link: [Example](#)

**Recommendation:** Simple remove the vacuous UB.

## Wording

### 6.3.3.2 void

~~1 The (nonexistent) value of a void expression (an expression that has type void) shall not be used in any way, and implicit or explicit conversions (except to void) shall not be applied to such an expression. **An expression that has type void is a void expression.** If an expression of any other type is evaluated as a void expression, its value or designator is discarded. (A void expression is evaluated for its side effects.~~

### 6.5.5 Cast operators

#### Constraints

~~2 Unless the type name specifies a void type, the type name shall specify atomic, qualified, or unqualified scalar type, and the operand shall have scalar type~~

**Acknowledgment:** The UB study group for comments.