Proposal for C2y
WG14 N3273

Title:	`alignof` of an incomplete array type
Author, affiliation:	Aaron Ballman, Intel
Date:	2024-06-13
Proposal category:	Issue
Target audience:	C programmers, mixed C and C++ source bases

Abstract: `alignof` on an incomplete array type is accepted in C++ and needlessly disallowed in C, so this proposes bringing the C behavior in line with the C++ behavior.

Prior Art: C++, Clang, ICC

`alignof` of an incomplete array type

Reply-to: Aaron Ballman (aaron@aaronballman.com)
Document No: N3273
Revises Document No: N3053
Date: 2024-06-13

Summary of Changes

N3273

- Removed part of the example and mention of it in surrounding prose

N3053

- Original proposal

Introduction and Rationale

In both C and C++, passing an array type as an operand of the `alignof` operator obtains the alignment of the array element type. Thus, so long as the array is an array of complete object types, the alignment can be determined. However, in C, there is a constraint that the `alignof` operator cannot be applied to an incomplete type. An array with no bounds is considered an incomplete type, and thus the following code results in a needless incompatibility between C and C++:

```
int alignment = alignof(int[]); // OK in C++, error in C
```

This was the subject of WG14 N1910, which was discussed at the 2018 Pittsburgh meeting and favorably received. I took the action item to champion this and it fell off my radar (mea culpa!).

Proposal

Because the completeness of the element type is all that is necessary to determine the alignment of an array type, the `alignof` operator can safely be used on an incomplete array of complete object types. This helps to unify both C and C++ in this corner case. Because C has always had this restriction, the change should not break any existing C code. Clang has been shipping with this behavior since Clang 3.5 (Sept 2014).

Because of the lack of opportunity to break valid C code and because C++ has always worked in practice in the way being proposed here, this paper should be added to the Extensions to Obsolete Versions of C list.

Proposed Straw Polls

Does WG14 want to adopt N3273 into C2y?

Does WG14 want to add N3273 to the Extensions to Obsolete Versions of C list?

Proposed Wording

The wording proposed is a diff from the working draft of WG14 N3047. Green text is new text, while ~~red~~ text is deleted text.

Modify 6.5.4.4p1:

The `sizeof` operator shall not be applied to an expression that has function type or an incomplete type, to the parenthesized name of such a type, or to an expression that designates a bit-field member. The `alignof` operator shall ~~not~~ be applied to a ~~function type or an incomplete type~~ complete object type or an array thereof.

Add a new Example to 6.5.4.4:

Example 4 The `alignof` operator can be applied to an incomplete array type, as shown in:

```
int a1 = alignof(int[]);    // Ok
int a2 = alignof(int[][1]); // Ok
```

Acknowledgements

I would like to recognize the following people for their help in this work: Rajan Bhakta, Pavel Morozkin, Erich Keane, Tom Honermann, and Shafik Yaghmour.

References

[N1910]
Potential DR: _Alignof Incomplete Arrays. Bhakta. https://www.open-std.org/jtc1/sc22/wg14/www/docs/n1910.htm