Clarifying the status of feature test macros

Document Number: **P0697 R0**
Date: 2017-06-18
Author: Herb Sutter (hsutter@microsoft.com)
Audience: EWG, WG21

Abstract

SD-6's current status as a WG21-authored but not WG21-technically-approved de facto specification is causing confusion in the marketplace. Also, it is incomplete with missing sections.

This paper proposes that WG21 make a decision on whether it wants to regularize SD-6 by turning it into a completed formal specification of some kind, or else retire it.

1 Background

Over the last several years, WG21 has agreed to let SG10 produce and maintain Standing Document 6 (SD-6). This was approved in plenary at two meetings, with no opposition; however, the polls were not the usual polls to approve a specific technical document, but rather polls to see if there was any opposition to SG10 having its own standing document whose contents are not formally reviewed outside SG10. In particular, when polled it was explained that:

- this would be a document that implementers who cared to implement feature tests could refer to in order to do so consistently; and
- there would be no attempt to turn this into a TS or other formal document that creates a formal requirement on implementers.

The result is that we have produced SD-6 as the only specification produced by WG21 (ever, to my knowledge) that has never been the subject of a committee technical plenary poll, or been formally approved by national bodies via ISO ballots.

Since then, several things have happened:

- *The camel's nose has entered the standard:* Part of SD-6 (`__has_include`) has since been put into the C++ standard as a normative requirement (see [cpp.cond]). Some WG21 experts and users, including two different library implementers/owners I have heard from, have expressed that they feel it has turned out to be odd or unuseful to be able to test for a whole header, but not for individual features.
- *It's starting to be encouraged/enforced in WG21:* Subgroups have started recommending (and even requiring) that new proposals contain feature test macros as a consideration (and sometimes a condition) of passing subgroup review. However, expectations about this seem to be at least partly-unwritten rules.

- *There is pressure to make this a portable de facto standard:* Because some implementations support feature test macros, users are starting to demand their availability in other implementations. On the one hand, customer demand for a feature is usually a positive sign of its usefulness. On the other hand, as a standardization body, WG21 normally either ignores de facto portability and leave it entirely to implementations or some other group or consortium, or turns it into de jure portability and issues a formal national body-approved report or specification.
- *We now have more experience with the feature tests:* They have now been tried in practice, so we know more about them to make a technical decision about their desirability and readiness.

2 The problems

There are two main issues:

First, SD-6's status is causing confusion in the marketplace, and to some extent within the committee, and we should clarify this by regularizing its status.

Second, the current version of the document (as of this writing, P0096R3 dated 2016-04-29) still appears incomplete, with "STUBS for sections expected to be filled in later," including the sections "Conditionally-supported constructs" and "C++98 features." If we want to keep this document, we should complete it.

3 Recommendation

WG21 needs to make a decision on whether it wants to turn SD-6 into a completed formal specification, or not.

I see the following main options, all of which are of the form "turn SD-6 into something different":

a) Retire SD-6 from the standing documents page.
b) Complete the document, and turn it into a Technical Report (TR) or Technical Specification (TS).
c) Complete SD-6, and move it into the C++20 IS working draft.
d) Complete SD-6, and turn it into a new kind of WG21-approved specification, namely a document that is a specification whose design and wording progresses as usual (in EWG/LEWG and CWG/LWG) and is approved by WG21 in plenary, but is not the subject of formal ISO ballots. In other words, this would be a TR or TS in every way except formal ISO ballot and publication. This would at least make WG21 consciously ratify SD-6 with its technical content, rather than leaving it in its current "sure, put in whatever you like, we're not going to review it" state.

We should plan to discuss this in Toronto. If anyone would like to suggest additional options, that would be welcome but please make them specific and be prepared to present them to the group.

4 Conclusion

This discussion should begin in EWG, but I would like it to lead to WG21 as a whole deciding whether SD-6 is something officially blessed by the committee that should be turned into something more formal, or should be abandoned, because its halfway-existence is causing confusion in the marketplace, and to some extent within the committee.