Doc. No. P2184R0
Date: 2020-06-10
Audience: all
Reply to: Bjarne Stroustrup (bs@ms.com)

Thriving in a Crowded and Changing World: C++ 2006–2020

Bjarne Stroustrup

Morgan Stanley and Columbia University

www.stroustrup.com

Over the last two years, I have written a paper for the ACM on the history of C++. It is now released even though the History Of Programming Languages (HOPL) conference has been postponed. The paper covers the earlier years, the ISO committee process, C++11, C++14, C++17, C++20, concepts, the uses of C++, my hopes for its future, and more:

Thriving in a Crowded and Changing World: C++ 2006–2020

BJARNE STROUSTRUP, Morgan Stanley and Columbia University.

By 2006, C++ had been in widespread industrial use for 20 years. It contained parts that had survived unchanged since introduced into C in the early 1970s as well as features that were novel in the early 2000s. From 2006 to2020, the C++ developer community grew from about 3 million to about 4.5 million. It was a period where new programming models emerged, hardware architectures evolved, new application domains gained massive importance, and quite a few well-financed and professionally marketed languages fought for dominance. How did C++ an older language without serious commercial backing ś manage to thrive in the face of all that?

This paper focuses on the major changes to the ISO C++ standard for the 2011, 2014, 2017, and 2020 revisions. The standard library is about 3/4 of the C++20 standard, but this paper's primary focus is on language features and the programming techniques they support.

The paper contains long lists of features documenting the growth of C++. Significant technical points are discussed and illustrated with short code fragments. In addition, it presents some failed proposals and the discussions that led to their failure. It offers a perspective on the bewildering flow of facts and features across the years. The emphasis is on the ideas, people, and processes that shaped the language.

Themes include efforts to preserve the essence of C++ through evolutionary changes, to simplify its use, to improve support for generic programming, to better support compile-time programming, to extend support for concurrency and parallel programming, and to maintain stable support for decades' old code.

> The ISO C++ standard evolves through a consensus process. Inevitably, there is competition among proposals and clashes (usually polite ones) over direction, design philosophies, and principles. The committee is now larger and more active than ever, with as many as 250 people turning up to week-long meetings three times a year and many more taking part electronically. We try (not always successfully) to mitigate the effects of design by committee, bureaucratic paralysis, and excessive enthusiasm for a variety of language fashions.
>
> Specific language-technical topics include the memory model, concurrency and parallelism, compile-time computation, move-semantics, exceptions, lambda expressions, and modules. Designing a mechanism for specifying a template's requirements on its arguments that is sufficiently flexible and precise yet doesn't impose run-time costs turned out to be hard. The repeated attempts to design "concepts" to do that have their roots back in the 1980s and touch upon many key design issues for C++ and for generic programming.
>
> The description is based on personal participation in the key events and design decisions, backed by the thousands of papers and hundreds of meeting minutes in the ISO C++ standards committee's archives.

The paper is academic, peer-reviewed, and aiming to document aims, principles, events, decisions, rationales, and more. It is a massive paper (168 pages) rather than a quick read:

https://www.stroustrup.com/hopl20main-p5-p-bfc9cd4--final.pdf

With this paper, C++ becomes the first and only language to be presented three times at HOPL and I become the first and only person to present three times.

HOPL happens every 15 years. The other papers are now accessible as the Proceedings of the ACM on Programming Languages, Volume 4, Issue HOPL, June 2020:

https://dl.acm.org/toc/pacmpl/2020/4/HOPL