

Remove Deprecated strstreams From C++26

Document #: P2867R0
Date: 2023-05-15
Project: Programming Language C++
Audience: Library Evolution Incubator
Revises: N/A
Reply-to: Alisdair Meredith
<ameredith1@bloomberg.net>

Contents

| | |
|------------------------------|----------|
| 1 Abstract | 1 |
| 2 Revision history | 1 |
| 2.1 R0: Varna 2023 | 1 |
| 3 Introduction | 1 |
| 4 Origin | 2 |
| 5 Analysis | 2 |
| 6 Proposal | 2 |
| 7 Wording | 2 |
| 8 Acknowledgements | 2 |
| 9 References | 3 |

1 Abstract

Annex D of the C++ Standard, deprecated features, maintains a limited and easily misused iostreams facility for `char *` strings in user-managed memory. This paper proposes the removing this facility from the C++ Standard Library as C++20 and C++23 have provided superior replacement facilities.

2 Revision history

2.1 R0: Varna 2023

Original version of this document, extracted from the C++23 proposal [P2139R2].

Key changes since that earlier paper:

- Rebased wording onto [N4944]

3 Introduction

At the start of the C++23 cycle, [P2139R2] tried to review each deprecated feature of C++, to see which we would benefit from actively removing, and which might now be better undeprecated. Consolidating all this

analysis into one place was intended to ease the (L)EWG review process, but in return gave the author so much feedback that the next revision of that paper was not completed.

For the C++26 cycle there will be a concise paper tracking the overall review process, [P2863R0], but all changes to the standard will be pursued through specific papers, decoupling progress from the larger paper so that delays on a single feature do not hold up progress on all.

This paper takes up the deprecated `strstream` facility from the original C++98 standard, D.15 [depr.str.strstreams].

4 Origin

The `char*` streams were provided, pre-deprecated, in C++98 and have been considered for removal before. The underlying principle of previous reviews is that this facility not be removed until suitable replacement functionality is available in the standard for users to migrate to.

5 Analysis

C++20 landed the ability to move strings efficiently out of stringstreams in [P0408R7]. C++23 landed the `spanstream` library [P0448R4], which is a candidate for the replacement functionality, so for C++26 we can seriously consider removing this legacy feature, the largest and oldest deprecated feature in the standard. The Zombie Names clause provides library vendors the ability to retain support for these libraries long after the standard has stopped specifying them. Therefore the preferred recommendation of this paper is to finally remove this library from the C++26 Standard.

There remains the alternative position that this facility has been a supported shipping part of the C++ Standard for almost 30 years when C++26 ships. If we have not made serious moves to remove the library in all that time, maybe we should consider undeprecating, and taking away the shadow of doubt over any code that reaches for this facility today.

We note that there are two open issues that should be resolved as part of any attempt to undeprecate this facility:

- [LWG3095] `strstreambuf` refers to nonexistent member of `fpos`, `fpos::offset`
- [LWG3109] `strstreambuf` is copyable

Both issues could be closed without further work if the `char *` streams are removed.

6 Proposal

Remove the long deprecated `char *` streams from C++26.

7 Wording

All wording is relative to [N4944], the latest working draft at the time of writing.

Add new identifiers to 16.4.5.3.2 [zombie.names].

ANNEX C WORDING TO FOLLOW

Strike all of D.15 [depr.str.strstreams] `char *` streams.

8 Acknowledgements

Thanks to Michael Parks for the pandoc-based framework used to transform this document's source from Markdown.

Huge thanks to Peter Sommerlad for doing all the hard work to make removal of this tricky legacy facility possible.

9 References

- [LWG3095] Billy O’Neal III. `strstreambuf` refers to nonexistent member of `fpos`, `fpos::offset`.
<https://wg21.link/lwg3095>

- [LWG3109] Jonathan Wakely. `strstreambuf` is copyable.
<https://wg21.link/lwg3109>

- [N4944] Thomas Köppe. 2023-03-22. Working Draft, Standard for Programming Language C++.
<https://wg21.link/n4944>

- [P0408R7] Peter Sommerlad. 2019-07-18. Efficient Access to `basic_stringbuf`’s Buffer.
<https://wg21.link/p0408r7>

- [P0448R4] Peter Sommerlad. 2021-03-01. A `strstream` replacement using `span` as buffer.
<https://wg21.link/p0448r4>

- [P2139R2] Alisdair Meredith. 2020-07-15. Reviewing Deprecated Facilities of C++20 for C++23.
<https://wg21.link/p2139r2>

- [P2863R0] Alisdair Meredith. 2023-05-15. Review Annex D for C++26.
<https://wg21.link/p2863r0>