

# Contiguous Containers Should Contain `.data()`

---

Document number: P1768R0

Date: 2019-06-17

Project: Programming Language C++, Library Evolution Working Group Incubator

Reply-to: Nevin “☺” Liber, [nliber@anl.gov](mailto:nliber@anl.gov)

## Table of Contents

<b>Introduction</b> .....	<b>1</b>
<b>Motivation and Scope</b> .....	<b>2</b>
<b>Impact On the Standard</b> .....	<b>2</b>
<b>Design Decisions</b> .....	<b>2</b>
<b>Acknowledgements</b> .....	<b>2</b>
<b>References</b> .....	<b>2</b>

## Introduction

All the contiguous containers could contain a `.data()` member. All the standard contiguous containers do contain a `.data()` member. This proposes that all the contiguous containers should contain a `.data()` member, so that all future contiguous containers would contain a `.data()` member.

## Motivation and Scope

During recent discussions on other proposals, it was observed that while all the existing contiguous containers (`basic_string`, `array` and `vector`) and contiguous views (`basic_string_view`, and `span`) contain `.data()` member functions to access the underlying contiguous range, it wasn't a requirement which could be counted upon.

This proposal wishes to remedy that by requiring contiguous containers contain `.data()` member functions, as it would help with generic code.

## Impact On the Standard

This is additive to the standard. However, it would break contiguous containers in the wild which do not as of yet have `.data()` member functions or use `.data()` member functions for other purposes.

## Design Decisions

While `ranges::data(...)` can synthesize the behavior from `begin` and `end`, users are far more likely to either call the member `.data()` or `std::data(E)`, even in generic code.

## Acknowledgements

This research was supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of two U.S. Department of Energy organizations (Office of Science and the National Nuclear Security Administration) responsible for the planning and preparation of a capable exascale ecosystem, including software, applications, hardware, advanced system engineering, and early testbed platforms, in support of the nation's exascale computing imperative. Additionally, this research used resources of the Argonne Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC02-06CH11357.

Thanks to Christian Trott for pointing out that `.data()` was not an actual requirement on contiguous containers.

## References

N4810 Working Draft, Standard for Programming Language C