



C++1x for Embedded Systems compact

18th – 21st October 2022



Driving embedded software quality



C++1x for Embedded Systems compact

Date:	18 th – 21 st October 2022
Duration:	4 days, each 9:00 am – 1:00 pm UTC/GMT+2
Location:	Online
Speaker:	Andreas Fertig
Language:	EN
Price:	€ 1,700 (plus the legally applicable tax of the recipient of the service)
1. Early bird (20%):	€ 1,360 (plus the legally applicable tax of the recipient of the service) <u>register before 30-Apr-2022</u>
2. Early bird (10%):	€ 1,530 (plus the legally applicable tax of the recipient of the service) <u>register before 30-Sep-2022</u>

Testimonials:

"Understandable explanation & collegial atmosphere :-)"

"Andreas Fertig is very competent and conveys the topic in an understandable and entertaining way. Merci!"

"Very good seminar, where illustrative examples are shown."

Benefits

The flexibility of the C++ language has improved even further with the new features of the C++11 standard and its successors. Some of these innovations are particularly interesting for embedded systems. For example it is now possible to move computations from run to compile time, resulting in a smaller and more performant codebase.

Are you also tired of writing ++i and still trying to get > and >= right? C++1x helps by leaving recurrent code of this type to the compiler. As a result, you gain more time to write really sophisticated code.

In this four-day seminar Andreas Fertig teaches how C++1x supports you to write faster and more robust code. After the seminar, you will write clearer and more precise code and will pass code reviews easier and more confidently, among other things.

Target group / Prerequisites

- ✓ Developers and architects of C++ based systems. Familiarity with the major features of C++ is required. Concepts and technical innovations of C++1x will be introduced.

Format

Interactive class with exercises. All attendees are encouraged to play with the course material during the class.

Overview

In this seminar, attendees C++1x familiarise themselves with the use of embedded systems. Although C++1x is not specifically designed for embedded systems, the language contains various valuable features for such systems. Participants gain knowledge to optimise compiled code size and performance. An example is `static_assert`. With this function, it is possible to check conditions at compile time. Calculations can also be done with `constexpr` functions at compile time. In both cases, these features positively affect runtime behaviour and code size.

The use of move semantics promises faster and more efficient copying operations. Participants will learn how move semantics work, how to use it, and in what situations the compiler will disable them.

C++1x for Embedded Systems compact

During the seminar, the participants will learn which new features of C++1x they can use in their codebase. Complex problems can be formulated more easily using the extended support of the compiler. The resulting code is often more robust and less error-prone. Participants who work with an older compiler,

without C++1x, will receive tips and suggestions on using new features in their code. After the seminar, participants write clearer and more precise code. For example, participants will pass code reviews easier and safer.

Topics Outline

- ✓ Clean and modern C++ code
- ✓ Do things at compile time: `constexpr`, `constexpr if`
- ✓ The right choice at the right time: preprocessor, `inline` or `constexpr`
- ✓ ROMability
- ✓ Interesting C++1x elements for embedded systems: `override`, `auto`, `literaloperator`, `nullptr`, `structured bindings`, `std::optional`
- ✓ A must: range based for loops
- ✓ Influence of C++1x to performance and code size
- ✓ Templates: the proper dose
- ✓ Usage of the new `initializerlist`
- ✓ Lambdas
- ✓ Move semantics

About the speaker Andreas Fertig

Andreas Fertig, CEO of Unique Code GmbH, is an experienced trainer and lecturer for C++ for standards 11 to 20. Andreas is involved in the C++ standardisation committee, in which the new standards are developed. At international conferences, he presents how code can be written better. He publishes specialist articles, e.g., for *iX* magazine, and has published several textbooks on C++. With C++ Insights (<https://cppinsights.io>), Andreas has created an internationally recognised tool that enables users to look behind the scenes of C++ and thus to understand constructs even better. Before working as a trainer and consultant, he worked for Philips Medizin Systeme GmbH for ten years as a C++ software developer and architect focusing on embedded systems.



Registration for the seminar: C++1x for Embedded Systems compact

Date: 18th – 21st October 2022
4 days, each 9:00 am – 1:00 pm UTC/GMT+2

Price: € 1,700 (plus the legally applicable tax of the recipient of the service)

1. Early bird (20%): € 1,360 (plus the legally applicable tax of the recipient of the service)
[register before 30-Apr-2022](#)

2. Early bird (10%): € 1,530 (plus the legally applicable tax of the recipient of the service)
[register before 30-Sep-2022](#)

Please send this registration form by

email to: seminare@qa-systems.de

Contact details attendee:

Company

Family name, First name

Department

Address

ZIP, City, Country

Phone/Fax

Email

VAT-ID No:

Invoice address (if different from above):

Company

Family name, First name

Department

Address

ZIP, City, Country

PO (if available)

I acknowledge that I have read and accept the terms and conditions for public online events.

Date

Name, authorized signature



QA Academy Terms and Conditions for Public Online Events

QA Academy Public Online Events

These Terms and Conditions apply to all Seminars and Training Courses where these are delivered online in web meetings.

Confirmation of registration

After registration you will receive an acknowledgement of receipt via email. The official confirmation with further technical instructions and invoice attached will be sent via email in time before the seminar or training course date.

Participation and participant license

Only the registered participant is entitled to participate in the online event.

The registered participant is required to keep confidential all event access information and is expressly not permitted to grant third parties access to this information. Event access information may only be passed on after prior registration and written approval from QA Systems. QA Systems must be informed immediately if the event access information is lost or there is a suspicion of misuse.

Only the registered participant is granted rights (participant license) to use the supplied event materials i.e. plug-ins and documents. The registered participant is expressly not permitted to store event materials on a company or web server, forward them by e-mail, reproduce them in any other way or make them accessible to any third parties.

Recording of image and sound

Any recording, transmission or other reproduction of the event content requires the prior written approval of QA Systems.

Costs/Payment

The fees for attending the seminars or training courses are due immediately without deduction after receipt of invoice, anyhow before the event date. All prices are offered without tax (plus legally applicable tax of the recipient of the service).

Cancellation by participant

- 6 or more weeks before the event, none of the fee will be charged
- 6 to 2 weeks before the event, 20% of the fee will be charged
- From 2 weeks before the event, 100% of the fee will be charged

Replacement of event participant is possible at any time free of charge

Cancellation by organizer

We ask for your understanding that due to organizational reasons we retain the right to cancellations at our discretion at any point in time before the seminar or training course is due to commence. In these cases we will endeavour to provide an alternative date. In case of cancellation by QA Systems, paid event fees are fully refunded.

QA Systems is not liable for any costs or charges incurred in the event that any seminar or training course is cancelled or postponed due to low registration numbers or unforeseen circumstances.