

1

Einleitung

1.1 Ist dieses Buch für Sie?

Als C++-Entwickler haben Sie natürlich mitbekommen, dass es seit 2011 ein runderneueretes „modernes“ C++ gibt, das sich in vielen Konzepten vom klassischen C++ unterscheidet und das seitdem regelmäßig weiterentwickelt wird, mit einem kleinen Update im Jahr 2014 und mit einer größeren Updates im Jahr 2017 (zum Zeitpunkt des Buchs so geplant).

Aber es geht Ihnen vielleicht wie den meisten: Sie stecken im Tagesgeschäft, haben Deadlines zu beachten und unvorhersehbare Fixes zu liefern. Bis jetzt blieb Ihnen keine Zeit, um sich die Neuerungen genauer anzusehen oder gar darüber nachzudenken, wie Sie sie in Ihrer Arbeit einsetzen wollen. Die vorhandenen Bücher zum modernen C++ sind alle recht umfangreich, erheblich umfangreicher jedenfalls als das ursprüngliche C++-Buch „Die C++-Programmiersprache“ von Bjarne Stroustrup aus dem Jahr 1985. Das alles zu lesen, zu verstehen sowie die Vor- und Nachteile abzuwägen, dafür hat die Zeit bisher einfach nicht gereicht.

Wenn diese Beschreibung auf Sie zutrifft, dann ist dieses Buch genau das richtige für Sie. Es strebt nicht danach, möglichst alle Aspekte des neuen Standards ausführlichst zu beschreiben, sondern gibt Ihnen so knapp wie möglich das nötige Wissen und Verständnis an die Hand, die neuen Möglichkeiten sofort einzusetzen. Es beginnt bei den Erläuterungen nicht bei Adam und Eva, sondern setzt voraus, dass Sie über ein solides Praxiswissen mit klassischem C++ verfügen. Andererseits erwartet es von Ihnen nicht, dass Sie die theoretischen Grundlagen von Programmiersprachen noch auswendig aufsagen können. Ein großer Wert liegt darauf, zu jeder Neuerung zu erläutern, wie und wann man sie einsetzen kann. Ab und zu steht hier auch einmal, dass Sie das eher nicht brauchen werden oder lieber die Finger davon lassen sollten.

Vom Aufbau her stehen die einzelnen Kapitel, von denen jedes genau ein neues Feature beschreibt, weitgehend für sich. So brauchen Sie dieses Buch nicht unbedingt von vorne bis hinten zu lesen, sondern können sich auch einfach die Themen heraussuchen, die Ihnen am interessantesten erscheinen. Weil natürlich manche Neuigkeiten auf anderen aufbauen, finden Sie an den entsprechenden Stellen die Verweise auf die nötigen Voraussetzungen.

1.2 Aufbau

Die Kapitel sind locker zu vier Themenbereichen zusammengefasst. Im ersten Teil „Sprache“ geht es um Erweiterungen an der Programmiersprache selbst. Hier werden hauptsächlich neue Schlüsselwörter oder neue Bedeutungen vorhandener Schlüsselwörter vorgestellt.

Anschließend kommt ein Teil über „Templates“, den Sie komplett überspringen können, wenn Sie keine Templates bei Ihrer Arbeit einsetzen. Hier sind sowohl Sprach- als auch Bibliothekserweiterungen zusammengefasst, die das Programmieren mit Templates betreffen. Früher oder später sollten Sie sich übrigens unbedingt mit diesem Themengebiet auseinandersetzen. Das Arbeiten mit Templates ist gerade durch die Erweiterungen seit C++11 erheblich angenehmer geworden. Dieses Buch will allerdings keine Einführung in die Templateprogrammierung sein.

Im dritten Teil geht es um die Erweiterungen der Standardbibliothek. Er ist mit „Bibliothek“ überschrieben und enthält Dutzende von neuen Funktionen und Klassen von ganz einfachen Dingen wie den Funktionen *next* und *prev* bis zu umfangreichen Bibliotheken wie der für Zufallszahlen oder Zeitpunkte und Zeitdauern.

Alles, was mit Multithreading und paralleler Programmierung zu tun hat, fasst der vierte Teil unter „Nebenläufigkeit“ zusammen. Auch hier kommen sowohl Spracherweiterungen als auch neue Bibliotheksklassen vor, weil sie zusammen ein sinnvolles Ganzes ergeben. Wenn diesem Ihr Hauptinteresse gilt, fangen Sie einfach damit an und folgen den Verweisen für die wenigen Spracherweiterungen und Bibliotheksklassen, die als Grundlage benötigt werden.

Im Anhang finden Sie ein Glossar, welches einige Fachbegriffe rund um C++ erläutert. Diese Begriffe gehören zwar nicht zu den Neuerungen seit C++11. Sie tauchen aber im Hauptteil auf und sind dann doch so speziell, dass sie nicht jeder sofort parat haben kann. Immer, wenn Sie den Pfeil vor einem Wort sehen wie in `→trivialer Typ`, dann wissen Sie, dass es dazu einen Glossareintrag gibt.

Die Beschreibung jedes Sprachmerkmals hat einen sich wiederholenden Aufbau. Zuerst kommt unter „Worum geht es?“ eine Kurzfassung, die in wenigen Sätzen erklärt, was das Feature tut. Darauf folgt unter der Überschrift „Was steckt dahinter?“ eine Motivation und Erläuterung, wo erklärt wird, wieso das Feature eingeführt wurde und wie es funktioniert. Als drittes kommen Tipps, wie Sie die Erweiterung am besten einsetzen, und wie nicht. Dieser Abschnitt ist mit „Wie setzt man es ein?“ überschrieben. In „Wer unterstützt es?“ folgen Hinweise zur Compilerunterstützung in Visual C++ und GCC. Abschließend runden Verweise auf den Header und gegebenenfalls weiterführende Literatur die Beschreibung ab.

Damit Sie die für Sie interessantesten Neuigkeiten möglichst schnell finden, gibt es neben einem umfangreichen Stichwortverzeichnis gleich anschließend eine Feature-Tabelle als Wegweiser. Hier finden Sie alle Erweiterungen von C++ mit einer Bewertung bezüglich Relevanz, Expertentum und Spezialwissen mit Verweis auf das jeweilige Kapitel.

1.3 Compiler

Bei den Compilern beschränkt sich dieses Buch auf die beiden meistverwendeten, nämlich Visual C++ unter Windows und GNU unter Linux. Viele Beispiele wurden sowohl mit MSVC 2015 Update 3 als auch GCC 5.4 bzw. 7.0 getestet. Manchmal war das aber nicht möglich, weil insbesondere MSVC 2015 bei den Features von C++17 noch große Lücken hat. Die Version 2017 war noch nicht veröffentlicht, als dieses Buch geschrieben wurde.

Insbesondere für GCC ist auch nicht vollständig dokumentiert, ab welcher Version eine bestimmte Erweiterung der Standardbibliothek unterstützt wird. Deshalb finden Sie bei der Beschreibung der Compilerunterstützung öfter Formulierungen wie „spätestens ab 4.8“. Das bedeutet dann,

dass laut Dokumentation und eigenen Tests das Feature in GCC 4.8 zur Verfügung steht, dass aber nicht bekannt ist, seit wann es schon existiert.

1.4 Schreibweisen

Dieses Buch wurde in der Originalfassung auf Deutsch geschrieben und bemüht sich deshalb, wo immer sinnvoll möglich, deutsche oder eingedeutschte Begriffe zu nutzen. Dies führt zwangsläufig zu einigen schwierigen Fällen. Zum Beispiel könnte man Compiler als Übersetzer verwenden, aber genau genommen besteht der Übersetzer aus Compiler und Linker zusammen. Statt Templates könnte man Schablonen sagen, aber dieser Begriff hat sich nicht genug eingebürgert, um ihn zu verwenden. Das Wort *variadisch* steht nicht im Duden, allerdings steht auch seine englische Entsprechung *variadic* nicht im Oxford Dictionary, so dass es hier verwendet wird (für *variadic templates*). Es gibt noch viele weitere Beispiele für Zweifelsfälle, sie wurden so gut gelöst, wie es eben möglich ist.

Generell wird in Programmierrichtlinien und Coding-Style-Guides empfohlen, auf die `using namespace`-Anweisung zu verzichten, und gerade bei Bezeichnern aus der Standardbibliothek das *std* immer mit dazuzuschreiben. Diese Regel wird in diesem Buch grundsätzlich beachtet, an einigen Stellen aber aus Platzgründen übergangen. In den Codebeispielen passen ca. 70 Zeichen in eine Zeile und Umbrüche führen zu sehr schlechter Lesbarkeit. Deshalb wurden lange Zeilen, wo immer möglich, gekürzt, unter anderem durch Weglassen von `std::`, durch Einsatz kürzerer Bezeichner, durch kürzere Literale für die Ausgabe, durch Einführung zusätzlicher Variablen usw. Diese Maßnahmen würden in echtem Code die Lesbarkeit verschlechtern und werden deshalb nicht empfohlen, im Buch jedoch verbessern sie die Übersichtlichkeit.

1.5 Dank

Dieses Buch hat der Autor nicht alleine geschaffen. Mein Dank geht an meine Frau Judith und meine Söhne Simon und Philipp, die nicht nur

viele Stunden auf Mann und Vater verzichten mussten, sondern auch auf den leistungsstärksten Computer unserer Familie. Peter Biereder, Jörg Roth und Kurt Holzinger haben sich durch das Manuskript gekämpft und geholfen, Fehler und Ungereimtheiten zu beseitigen. Sebastian Burkart und Martina Raschke vom Verlag haben das Projekt betreut und mir mit Rat und Tat zur Seite gestanden.