

C++11 ist da. Und jetzt?



Embedded Event 2012

Raphael Meyer

Was ist C++11?

JTC1 ?
SE22 ?
WG21 ?

Was ist C++11?

Joint ISO/IEC **T**echnical **C**ommittee **1**:
Information Technology

Subcommittee **22**: Programming
Languages

Working **G**roup **21**: C++

Was ist C++11?

25. März 2011

Die Working Group 21 des JTC1/SC22 genehmigt den Final Committee Draft.

12. August 2011

Der C++ Standard ISO/IEC 14882:2011 wird formell genehmigt.

September 2011

Der neue Standard wird veröffentlicht.

Was ist C++11?



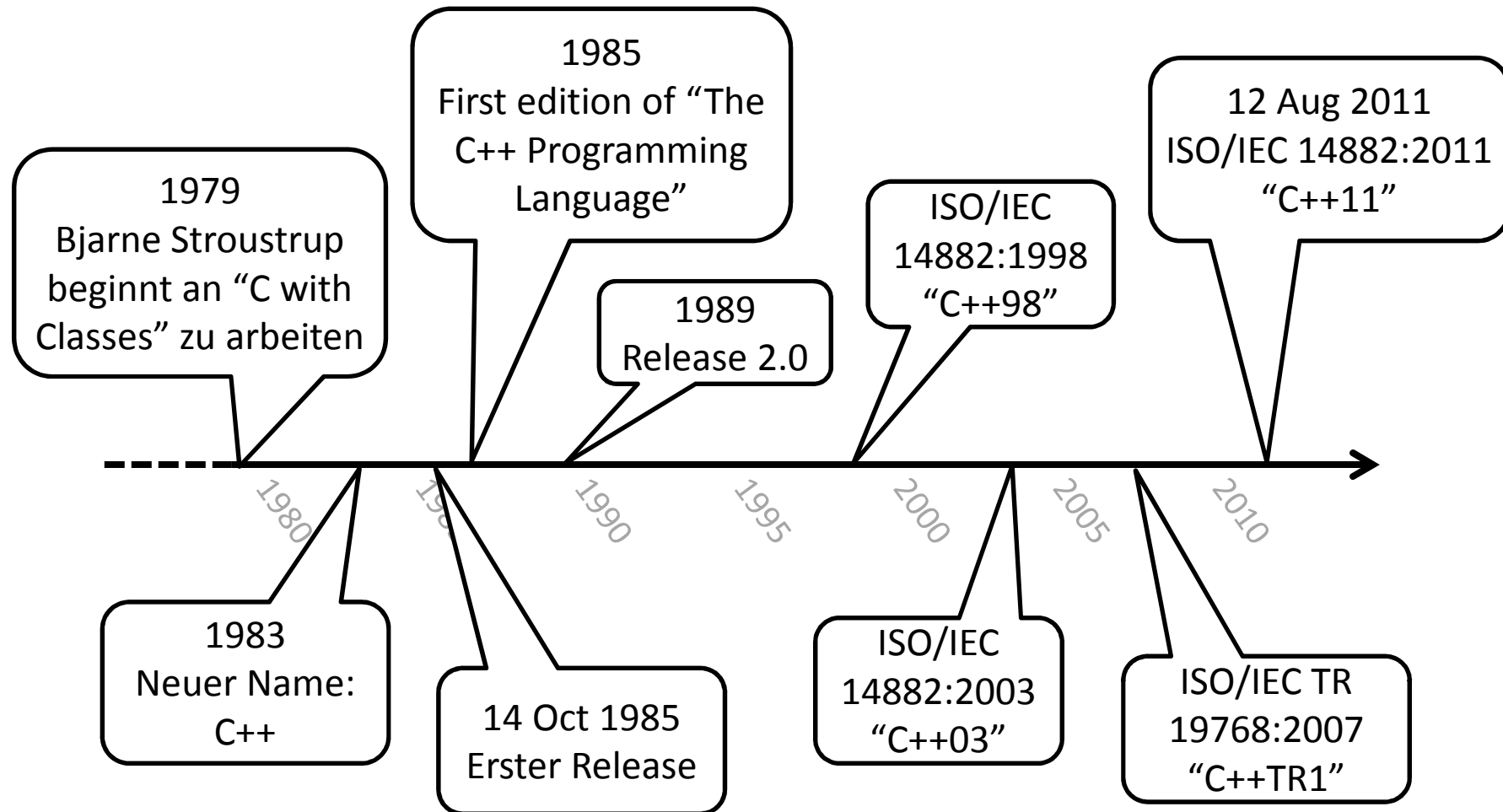
Bjarne Stroustrup

Die Frisur ist immer noch dieselbe, aber er ist heute moderner und gereifter.

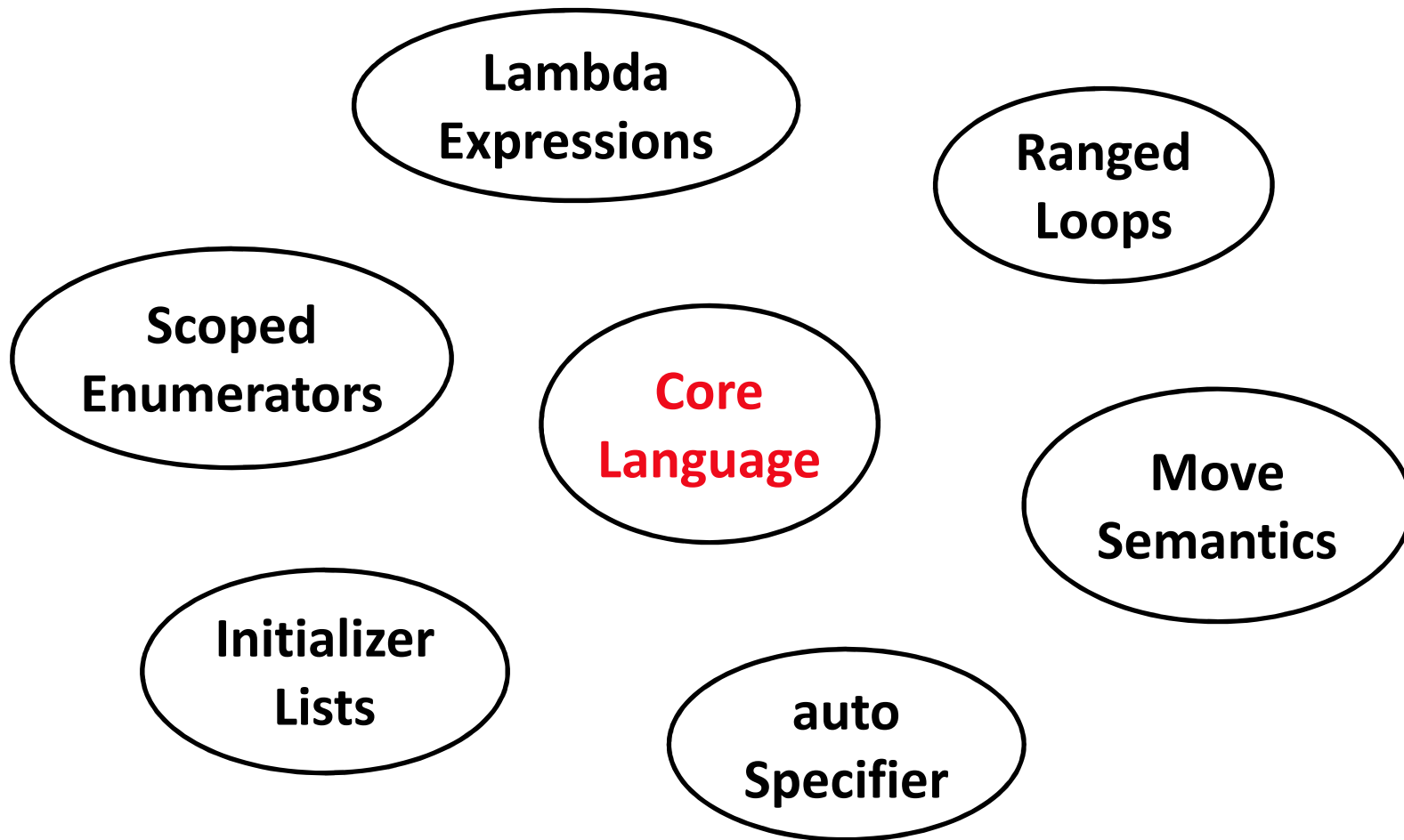
C++11

Es ist immer noch C++, aber moderner und gereifter.

C++ Entstehungsgeschichte



Was ist neu?



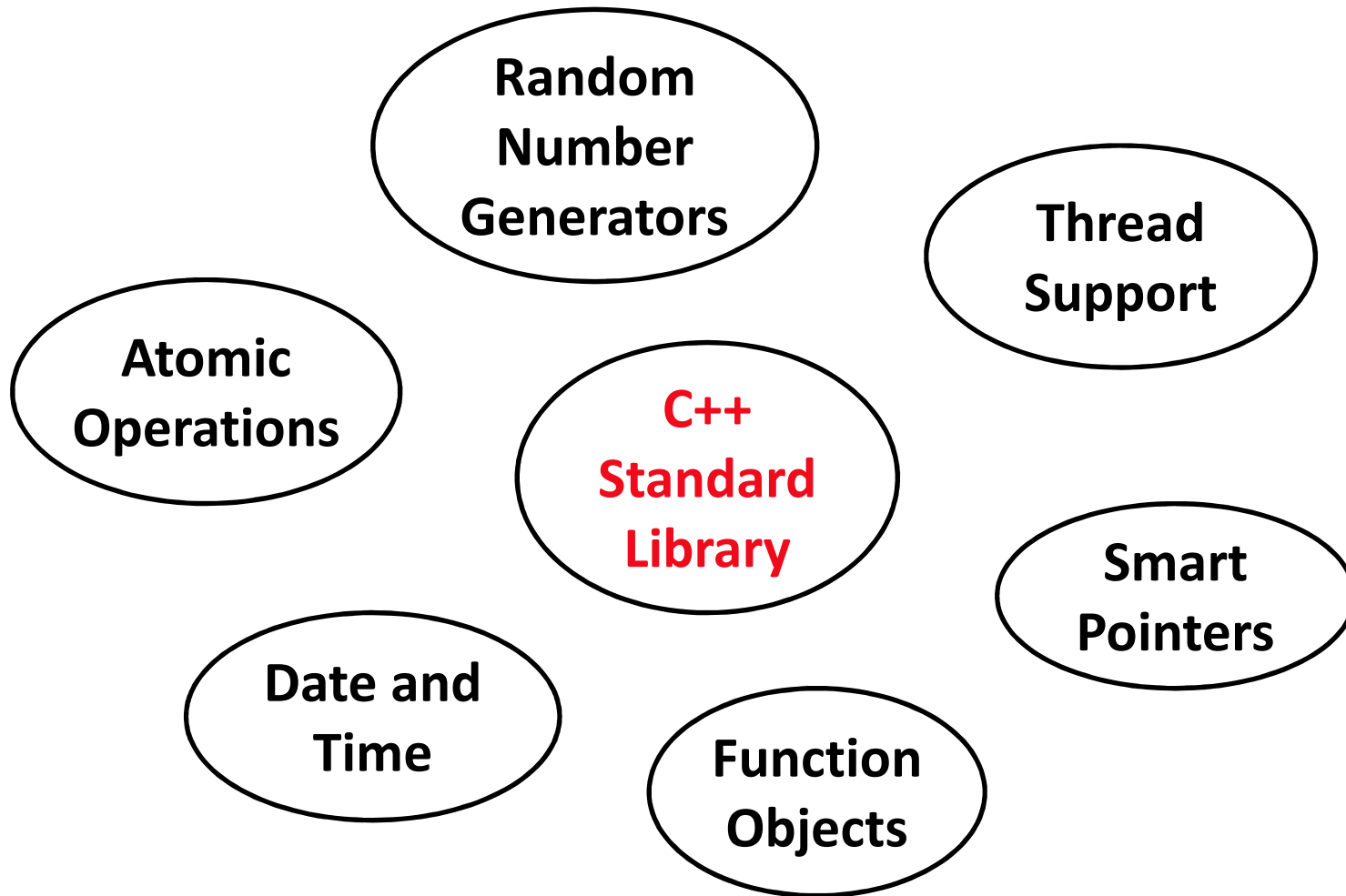
Was ist neu?

```
// Scoped Enumerator
enum class Ingredient { Rum, Cachaca, Lime, Sugar, Mint, Soda, Coconut, Pineapple };

// Initializer List
std::map<std::string, std::vector<Ingredient>> drinks {
    { "Caipirinha", { Ingredient::Cachaca, Ingredient::Lime, Ingredient::Sugar } },
    { "Mojito", { Ingredient::Rum, Ingredient::Lime, Ingredient::Mint, Ingredient::Sugar,
Ingredient::Soda } },
    { "PinaColada", { Ingredient::Rum, Ingredient::Coconut, Ingredient::Pineapple } }
};

std::cout << "Rum based:";
for(auto drink: drinks) { // Ranged Loop, auto Specifier
    if(std::count_if(drink.second.begin(), drink.second.end(),
        [](Ingredient i){ return i == Ingredient::Rum; }) > 0) { // Lambda Expression
        std::cout << " " << drink.first;
    }
}
std::cout << std::endl;
```

Was ist neu?



Was ist neu?

```
const int Threads = 1000; const int Shots = 1000;
```

```
std::vector<std::future<int>> simulations(Threads); // Futures
```

```
for(auto & sim: simulations) {
```

```
    sim = std::move(std::async(std::launch::async, [](){ // Move Semantics, Async  
        std::random_device rd; std::mt19937 gen(rd()); // Random Number Generator
```

```
        std::uniform_real_distribution<> dis(0, 1);
```

```
        int hits = 0;
```

```
        for(int i = 0; i < Shots; i++) {
```

```
            double x = dis(gen), y = dis(gen);
```

```
            if(x * x + y * y < 1.0) { hits++; }
```

```
        }
```

```
        return hits;
```

```
    });
```

```
}
```

```
int hits = 0;
```

```
std::for_each(simulations.begin(), simulations.end(),
```

```
    [&hits](std::future<int> & f){ hits += f.get(); });
```

```
std::cout << (4.0 * hits) / (Threads * Shots) << std::endl;
```

**Soll man jetzt
C++11
verwenden?**

**Ist die
Toolchain
schon bereit?**

Nein.

Ist die Toolchain schon bereit?

<http://wiki.apache.org/stdcxx/C++0xCompilerSupport/>

C++ 0x FEATURE	PAPER(S)	HP aCC	EDG eccp	GCC	Intel C++	MSVC	IBM XLC++	Sun/Oracle C++	C++ Builder 2009/10	Digital Mars C++	Clang
alignas	N2341										3.0
alignof	N2341			4.5					Yes		2.9
Atomic operations	N2427			4.4		11.0					3.1
auto	v0.9: N1984, v1.0: N2546		4.1(v0.9)	4.4(v1.0)	11.0(v0.9)	10.0(v0.9)	11.1 (V1.0)				Yes
C99 preprocessor	N1653			4.3	11.1		10.1	5.9		Yes	3.0
Concepts <i>[removed]</i>	N2617			ConceptGcc							3.0
constexpr	N2235			4.6	13.0		12.1				3.0
decltype	v1.0: N2343, v1.1: N3276		4.1(v1.0)	4.1	11.0(v1.0)	10.0(v1.1), 11.0(v1.1)	11.1 (V1.0)		Yes		3.0
Defaulted And Deleted Functions	N2346		4.1	4.4	12.0						3.0
Delegating Constructors	N1986			4.7			11.1				3.0
Explicit conversion operators	N2437			4.5	13.0		12.1		Yes		3.0
Extended friend Declarations	N1791		4.1	4.7	11.0	10.0***	V1R11,11.1				3.0
extern template	N1987	3, 5, 6		3.3	9	6.0	V1R11,11.1		Yes		Yes
Forward declarations for enums	N2764			4.6			12.1				3.1
Inheriting Constructors	N2540										3.1
Initializer Lists	N2672			4.4	13.0						3.1
	v0.9: N2550,				11.0(v0.9)	10.0(v1.0)					3.1

GCC

MSVC

llvm clang

Und die IDE?

Und die IDE?

Eclipse CDT

C++11 Unterstützung von Eclipse 4.2 (Juno) mit CDT 8.1 liegt im Vergleich zum gcc zwischen gcc 4.6 und 4.7.

Microsoft Visual C++

Die IDE unterstützt das, was auch der Compiler unterstützt.

Qt Creator

Version 2.5 unterstützt die wesentlichen Spracherweiterungen.

**Soll man
C++11 in
Zukunft
verwenden?**

**Sind denn die
Entwickler
schon bereit?**

**Wie soll ich
mich
vorbereiten?**

Wie soll ich mich vorbereiten?

- Boost libraries benutzen
- `std::tr1` aus der C++ Standard Library verwenden
- Prototypen-, Test- und Hobby-Projekte mit C++11
- C++11 lernen

Wo kann ich C++11 lernen?

Wo kann ich C++11 lernen?

bbv Academy C++11 Workshop geplant für 2013

<http://www.bbv.ch/> → bbv-Academy

The screenshot shows the homepage of bbv Software Services. At the top left is the logo 'bbv Software Services'. On the right, there are links for 'Home', 'Sitemap', 'Kontakt', and language options 'de' and 'en'. Below the header is a large banner image. A navigation bar contains the following menu items: 'Branchen', 'Dienstleistungen', 'Kompetenzen', 'Unternehmen', 'Karriere', 'bbv-Academy', and 'Blog'. A search bar with the text 'Suchen...' and a 'SUCHE' button is located on the right side of the navigation bar. The main content area is titled 'HOME' and features several articles. The first article is titled 'bbv Software Services AG – Ihr Partner von Consulting' and the second is 'bbv Software Services AG – Ihr Partner für innovativ'. A callout box with a black border and a white background points to the 'bbv-Academy' link in the navigation menu. The callout box contains the text 'bbv-Academy'.

Wo kann ich C++11 lernen?

<http://en.cppreference.com/>

cppreference.com Log in / create account

Page Discussion View View source History

C++ reference

C++98, C++03, C++11

Language <ul style="list-style-type: none">PreprocessorKeywordsOperator precedenceEscape sequencesASCII chartFundamental types	Strings library <ul style="list-style-type: none">basic_stringNull-terminated byte stringsNull-terminated multibyte stringsNull-terminated wide strings	Input/output library <ul style="list-style-type: none">basic_streambufbasic_filebufbasic_stringbufios_basebasic_iosbasic_istreambasic_ostreambasic_iostreambasic_ifstreambasic_ofstreambasic_fstreambasic_istringstreambasic_ostringstreambasic_stringstreamI/O manipulatorsC-style I/O
Headers	Containers library <ul style="list-style-type: none">array (C++11) - vector - dequelist - forward_list (C++11)set - multisetmap - multimapunordered_set (C++11)unordered_multiset (C++11)unordered_map (C++11)unordered_multimap (C++11)stack - queue - priority_queue	Localizations library
Concepts	Algorithms library	Regular expression library (C++11)
Utilities library <ul style="list-style-type: none">Type supportDynamic memory managementError handlingProgram utilitiesDate and timebitsetFunction objectspair - tuple (C++11)	Iterators library	Atomic operations library (C++11)
	Numerics library <ul style="list-style-type: none">Common mathematical functionsComplex numbersPseudo-random number generation	Thread support library (C++11)

Links - Libraries

Wo kann ich C++11 lernen?

Presentation Materials: Overview of the New C++ (C++11)

Scott Meyers

[PDF]

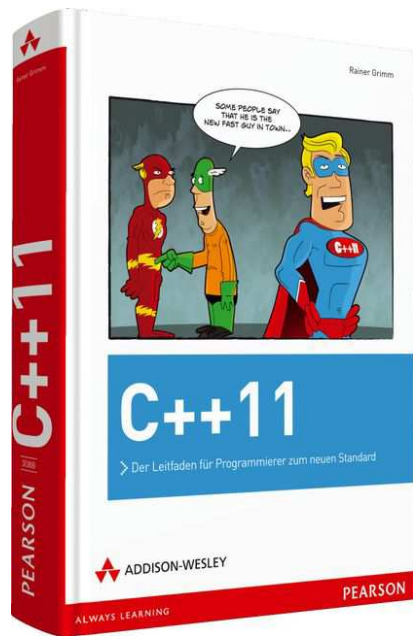
http://www.artima.com/shop/overview_of_the_new_cpp/



Wo kann ich C++11 lernen?

C++11

Der Leitfaden für Programmierer zum neuen Standard
Rainer Grimm



Rainer Grimm ist auch der Author der C++11 Serie im Linux
Magazin: “Modernes C++ in der Praxis”

Wo kann ich C++11 lernen?

C++11 programmieren

60 Techniken für guten C++11-Code

Torsten T. Will

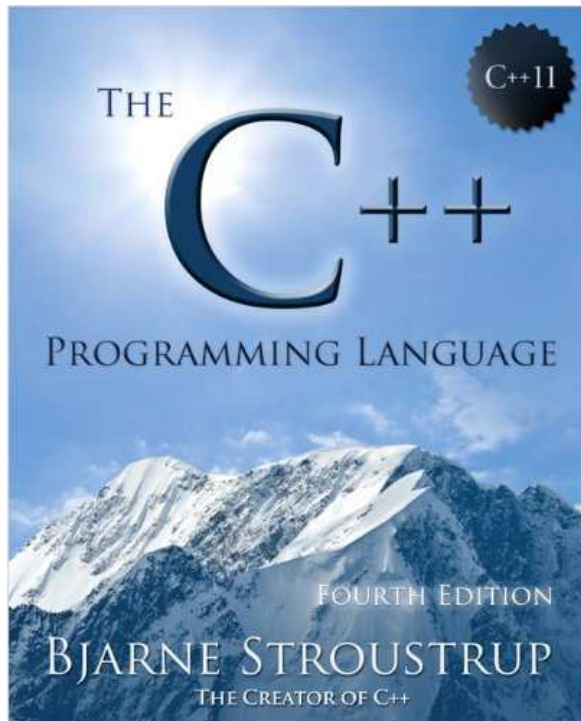


Wo kann ich C++11 lernen?

The C++ Programming Language (4th Edition)

Bjarne Stroustrup

Voraussichtliches Erscheinungsdatum: März 2013

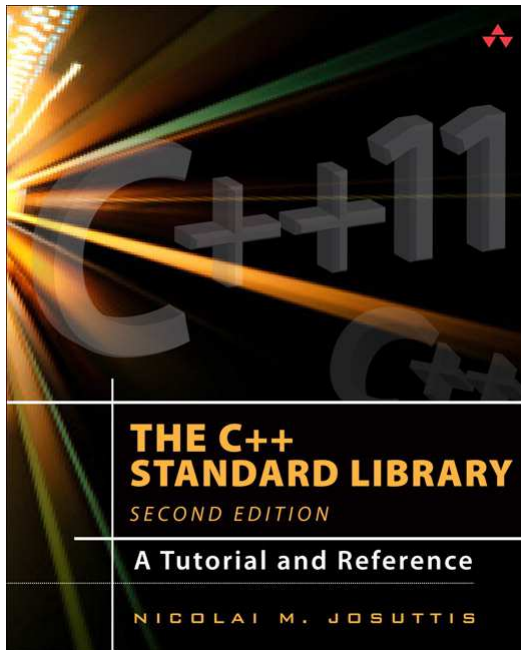


Wo kann ich C++11 lernen?

The C++ Standard Library (2nd Edition)

A Tutorial and Reference

Nicolai M. Josuttis



Enthält auch eine kurze Einführung (20 Seiten) über das Neue in der Core Language.

Fragen?

Danke!

Raphael Meyer

info@bbv.ch

www.bbv.ch