

Modernes C++ in der Praxis

Within C++, there is a much smaller and cleaner language struggling to get out.

-Bjarne Stroustrup

And no that smaller cleaner language is not Java or C#

-Bjarne Stroustrup nach
dummer Frage.

Immer eine zündende Idee.



Wer bin ich und was mache ich eigentlich hier?

- 📈 2001 als Austauschschüler Startup gegründet
- 📈 Autor der Kvasir SFR abstraction library
- 📈 Mitglied im ISO C++ working group SG14 Low latency, game design and embedded
- 📈 Gegner der Startuparmut im Ruhrgebiet
- 📈 C++ Evangelist

Warum C++?

The good thing about C++ is that it's a language for smart people
The bad thing about C++ is that it's a language for smart people

- 🔴 Efficiency
- 🔴 Static checking
- 🔴 Multi paradigm
- 🔴 Unglaubliche Mengen existierender Code
- 🔴 C++ ist ein internationales Abkommen
- 🔴 Werkzeugkasten für die Erstellung von Abstraktionen

Aktuelle Trends

- Massively multicore
- Functional programming
- Reliability/Quality is king
- Hardware awareness
- Hardware specific (SIMD, GPU, ARM-Cortex)
- Idiot proof APIs
- Optimizer aware coding style

Was ist Effizienz?

Üblicherweise rechnet man mit einer insgesamt Produktivität – also inklusive aller Projektstätigkeiten – von 10 bis 50 Lines of Code je Mitarbeiter und Tag. - Steve McConnell







- 🔴 Arbeit / Sekunde
- 🔴 Arbeit / Watt, sowohl in kleinen als auch in riesengroßen Systeme wichtig
- 🔴 Arbeit / € (Minimum: Hardware / Hardwarekosten)
- 🔴 Features / Failure in the field (Support, Sachschaden, Tote)
- 🔴 Programmiererproduktivität ist schwer messbar

Was ist static checking?

Checking code deeply requires understanding the code's semantics. The most basic requirement is that you parse it. Parsing is considered a solved problem. Unfortunately, this view is naïve, rooted in the widely believed myth that programming languages exist. - Allan McInnes

- 🔴 Static heißt vor/ohne Laufzeit
- 🔴 `char* + float = error` ist schon eine Art static checking
- 🔴 Die Kunst eine gute API zu erstellen besteht unter anderen darin so viele Fehler zur compile time zu erkennen wie möglich
- 🔴 Es gibt über die Semantik der Programmiersprache hinaus Werkzeuge, die eine statische Prüfung ermöglichen

Hardware awareness?

-  Caching architecture
-  Instruction reordering / pipelining
-  Prefetcher
-  Branch prediction
-  Data driven design
-  Profile guided optimization

Ressourcenverwaltung

- 🔴 Garbage collection nicht immer optimal
- 🔴 C++ besser als sein Ruf
- 🔴 Es existieren größtenteils automatische Speicherverwaltungsstrategien
- 🔴 “C++ Core Guidelines”

Was macht der Optimizer?

- 🔴 Unglaubliche Effizienzsteigerung
- 🔴 Leicht verwirrbar
- 🔴 Kann nur das optimieren was er sieht
- 🔴 Ermöglicht viele static checking und meta programming idioms

Was bedeutet Qualität in der Informatik?

*Informatiker ist ein Beruf ohne Berufsehre
-Odin Holmes*








- 📡 Von Tamagotchi bis IOT hat sich erstaunlich wenig getan
- 📡 Unit testing, leider nicht allgegenwärtig
- 📡 Nomenklatur wichtig (Entwurfsmuster etc.)
- 📡 Say it in code
- 📡 Guess and Pray oft noch Industriepraxis!



Functional Programming...



...ist dein Freund!

-  Unit testing ist einfach
-  Optimizerfreundlich
-  Race conditions unmöglich
-  Ressourcenverwaltung automatisch (stack based)
-  Static checking friendly
-  High intrinsic security
-  Highly generic