

---

## Liste der noch zu erledigenden Punkte

■ Danke an Mutter, Vater und alle Profs . . . . .	4
Abbildung: Baby Laptop Bild . . . . .	4



HOCHSCHULE BREMEN

BACHELORARBEIT

THESIS

---

# Konzeption und Implementierung einer Makrosprache in C++

---

*Autor:*

Roland JÄGER

360 956

14. März 2016

# Inhaltsverzeichnis

<b>Allgemeines</b>	<b>3</b>
Eidesstattliche Erklärung . . . . .	3
Danksagung . . . . .	4
<b>1 Einleitung</b>	<b>6</b>
1.1 Problemfeld . . . . .	6
1.2 Ziele der Arbeit . . . . .	6
1.3 Hintergründe und Entstehung des Themas . . . . .	6
1.4 Struktur der Arbeit, wesentliche Inhalte der Kapitel . . . . .	6
<b>2 Anforderungsanalyse</b>	<b>6</b>
2.1 Diskussion des Problemfeldes . . . . .	6
2.2 Anforderungen an die angestrebte Lösung . . . . .	6
<b>3 Konzeption</b>	<b>6</b>
3.1 Syntax . . . . .	6
3.2 Level 1 – Grundarchitektur . . . . .	6
3.3 Level 2 – Logik / primitive Rückgabewerte . . . . .	6
3.4 Level 3 – Komplexe Rückgabewerte . . . . .	6
<b>4 Exemplarische Realisierung</b>	<b>6</b>
4.1 Tokenizer . . . . .	6
4.2 Abstrakter Syntaxbaum . . . . .	6
4.3 Parser . . . . .	6
4.4 Interpreter . . . . .	6
4.5 Makro . . . . .	6
<b>5 Evaluation</b>	<b>6</b>
<b>6 Zusammenfassung und Ausblick</b>	<b>6</b>
6.1 Ausblick . . . . .	6
<b>7 Literatur</b>	<b>7</b>
<b>8 Anhänge</b>	<b>7</b>

## Allgemeines

### Eidesstattliche Erklärung

Ich, Roland Jäger, Matrikel-Nr. 360 956, versichere hiermit, dass ich meine Bachelorarbeit mit dem Thema

*Konzeption und Implementierung einer Makrosprache in C++*

selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe, wobei ich alle wörtlichen und sinngemäßen Zitate als solche gekennzeichnet habe. Die Arbeit wurde bisher keiner anderen Prüfungsbehörde vorgelegt und auch nicht veröffentlicht.

Bremen, den 14. März 2016

---

Roland Jäger

## Danksagung

Danke an Mutter, Vater und alle  
Profs



Fehlende  
Abbildung

Baby Laptop Bild



# 1 Einleitung

## 1.1 Problemfeld

## 1.2 Ziele der Arbeit

## 1.3 Hintergründe und Entstehung des Themas

## 1.4 Struktur der Arbeit, wesentliche Inhalte der Kapitel

# 2 Anforderungsanalyse

## 2.1 Diskussion des Problemfeldes

## 2.2 Anforderungen an die angestrebte Lösung

# 3 Konzeption

## 3.1 Syntax

## 3.2 Level 1 – Grundarchitektur

## 3.3 Level 2 – Logik / primitive Rückgabewerte

## 3.4 Level 3 – Komplexe Rückgabewerte

# 4 Exemplarische Realisierung

## 4.1 Tokenizer

## 4.2 Abstrakter Syntaxbaum

## 4.3 Parser

## 4.4 Interpreter

## 4.5 Makro

# 5 Evaluation

# 6 Zusammenfassung und Ausblick

## 6.1 Ausblick

zeichnet sind, kann das gesamte Macro parallel ausgeführt werden.

- Debugger / Stepping  
Es könnte ein Interface angeboten werden, mit dem man durch die Ausführung eines Macros Schritt für Schritt gehen kann.
- C++17 `std::string_view`  
for better parsing performance and less memory consumption.

## 7 Literatur

- [1] *cppreference.com*. 2015. URL: <http://en.cppreference.com/w/cpp> (besucht am 29.12.2015).
- [2] Jacques Ferber. „Computational reflection in class based object-oriented languages“. In: *ACM Sigplan Notices*. Bd. 24. 10. ACM. 1989, S. 317–326.
- [3] Erich Gamma u. a. *Design Patterns: Elements of Reusable Object-oriented Software*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1995. ISBN: 0201633612.
- [4] Roman R Redziejowski. „Parsing expression grammar as a primitive recursive-descent parser with backtracking“. In: *Fundamenta Informaticae* 79.3-4 (2007), S. 513–524.
- [5] Elizabeth Scott und Adrian Johnstone. „GLL parsing“. In: *Electronic Notes in Theoretical Computer Science* 253.7 (2010), S. 177–189.
- [6] *The C++ Standards Committee*. 2016. URL: <http://www.open-std.org/JTC1/SC22/WG21/> (besucht am 17.01.2016).
- [7] Steve Vinoski. „A time for reflection“. In: *Internet Computing, IEEE* 9.1 (2005), S. 86–89.

## 8 Anhänge