



Hochschule Osnabrück  
University of Applied Sciences

# Implementierung eines Source-To-Source-Kompilierers zur Optimierung von CSS-Dateien im Webstack

Oliver Erxleben (oliver.erxleben@hs-osnabrueck.de)

Sergej Hert (sergej.hert@hs-osnabrueck.de)

Jörn Voßgröne (joern.vossagroene@hs-osnabrueck.de)

Hochschule Osnabrück  
Ingenieurwissenschaften und Informatik  
Informatik - Mobile und Verteilte Anwendungen  
Compilerbau - Sommersemester 2013

13. Juni 2013

## Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Motivation . . . . .	1
1.2	Zielsetzung . . . . .	2
1.3	Begriffsdefinitionen . . . . .	2
1.3.1	Web-Stack . . . . .	2
<b>2</b>	<b>Cascading Style Sheets</b>	<b>2</b>
2.1	Allgemeine Verwendung . . . . .	2
2.1.1	Browserunterstützung . . . . .	2

2.2	Syntax . . . . .	2
2.3	Grammatik . . . . .	2
<b>3</b>	<b>Optimierung</b>	<b>2</b>
3.1	Analyse . . . . .	2
<b>4</b>	<b>Messungen</b>	<b>2</b>
4.1	Grundlage . . . . .	2
<b>5</b>	<b>Ergebnis</b>	<b>2</b>

## Abbildungsverzeichnis

## Listings

### **Zusammenfassung:**

Die vorliegende Ausarbeitung wurde in LaTeX verfasst und ist eine gemeinsame Arbeit von Oliver Erxleben, Sergej Hert und Jörn Voßgröne an der Hochschule Osnabrück / University of Applied Sciences im Fachbereich Ingenieurwissenschaften und Informatik für das Fach Compilerbau im Sommersemester 2013. Die Arbeit beschäftigt sich mit der Optimierung von Cascading Stylesheets für Webseiten.

Die Arbeit gliedert sich in mehrere Abschnitte. Im ersten Kapitel wird der Hintergrund des Themas beschrieben und es werden wichtige Fachbegriffe definiert. Ausserdem werden die Anforderungen an die Software aufgestellt.

Im zweiten Teil, Cascading Style Sheets, werden die Sprache, dessen Verwendung, sowie die zugrundeliegende Grammatik erläutert und es wird beschrieben wie Webbrowser diese interpretieren.

Der dritte Teil erläutert das Vorgehen der Optimierung. Neben der Syntaxanalyse und dem Optimieren der Knoten, wird auch die Implementierung eines Kommandozeilenwerkzeugs zur Steuerung der Optimierungen erläutert.

Der Abschnitt Messungen erklärt das Vorgehen für Tests.

Im abschließenden Abschnitt wird die Arbeit resümiert, das Ergebnis zusammengefasst und gewonnene Erfahrungen geschildert.

# 1 Einleitung

## 1.1 Motivation

Im Alltag eines Frontend-Entwicklers, eines Mitarbeiters an einem Web-Projekt oder des Entwicklers für das grafische User Interface kommt es nicht selten vor, dass die Beschreibung der grafischen Elemente durch Cascading Stylesheets geschieht. Noch seltener sind diese Style Angaben fehlerfrei. Sei es aufgrund von Zeitdruck, unterschiedlichen Entwicklern oder durch nachträgliches Bugfixing, oft sind CSS-Regeln inkonsistent aufgestellt, zum Beispiel existieren noch Regeln, die gar nicht mehr im DOM der Seite zu finden sind. Es werden Regeln mehrmals überschrieben und es wird nicht auf Optimierung von Selektoren geachtet.

Seit 2010 berücksichtigt der Page Ranking Algorithmus von Google auch die Ladezeiten für Websites. Seiten, die neben SEO ein gutes Page Ranking erhalten, werden demnach auch durch ihre Ladezeiten bestimmt. Die Ladezeiten spielen zwar im Vergleich mit SEO nur eine kleine Rolle, können aber zu einem besseren Ergebnis beitragen (Weitere Informationen: <http://googlewebmastercentral.blogspot.de/2010/04/using-site-speed-in-web-search-ranking.html>). Ladezeiten von mobilen Websites und Web-Anwendungen bzw. Seiten, die über mobile Internetverbindungen geladen werden, sollten schnell und nur wenig Daten übertragen, um ein konsistentes Benutzererlebnis zu gewährleisten. Statistiken zeigen, dass Benutzer auf Websites eher verbleiben wenn diese schnell geladen werden und der Benutzer schnell Informationen abrufen oder mit der Anwendung interagieren kann.

Im Rahmen der Hausarbeit für das Fach Compilerbau im Sommersemester 2013 im Master-Studiengang Informatik - Verteilte und Mobile Anwendungen an der Hochschule Osnabrück / University of Applied Sciences soll ein Werkzeug entwickelt werden mit dem sich Stylesheets optimieren lassen. Dazu soll ein Kommandozeilentool entwickelt werden, womit CSS-Optimierungen gesteuert und ausgegeben werden können.

## 1.2 Zielsetzung

## 1.3 Begriffsdefinitionen

### 1.3.1 Web-Stack

# 2 Cascading Style Sheets

## 2.1 Allgemeine Verwendung

### 2.1.1 Browserunterstützung

## 2.2 Syntax

## 2.3 Grammatik

# 3 Optimierung

## 3.1 Analyse

## 3.2

# 4 Messungen

## 4.1 Grundlage

# 5 Ergebnis