



Hochschule für  
Wirtschaft und Recht Berlin  
Berlin School of Economics and Law

## **Berliner Schriften zu modernen Integrationsarchitekturen**

Hrsg.: Prof. Dr.-Ing. habil. Andreas Schmietendorf

**Hochschule für Wirtschaft und Recht Berlin**

Fachbereich II

Wirtschaftsinformatik – Systementwicklung

Maximilian Bieleke

## **Performanceoptimierung in Single-Page-Applications**

# Bachelor Thesis

---

## Performanceoptimierung in Single-Page-Applications

---

Maximilian Bieleke

vorgelegt am 19.07.2021  
an der  
Hochschule für Wirtschaft und Recht Berlin

**Von:** Maximilian Bieleke  
**Fachbereich:** Duales Studium Wirtschaft · Technik  
**Fachrichtung:** Wirtschaftsinformatik  
**Studienjahrgang:** 2018  
**Semester:** 6  
**Ausbildungsbetrieb:** DB Systel GmbH  
**Erstgutachter:** Prof. Dr. Andreas Schmietendorf  
**Zweitgutachter:** Marek Meyer  
**Betreuer Betrieb:** Marek Meyer  
**Unterschrift Betreuer:** \_\_\_\_\_



**Berliner Schriften zu  
modernen Integrationsarchitekturen**

herausgegeben von  
Prof. Dr.-Ing. habil. Andreas Schmietendorf  
Hochschule für Wirtschaft und Recht Berlin, FB II

Band 26

**Maximilian Bieleke**

**Performanceoptimierung in Single-Page-Applications**

Shaker Verlag  
Düren 2021

### **Bibliografische Information der Deutschen Nationalbibliothek**

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

Copyright Shaker Verlag 2021

Alle Rechte, auch das des auszugsweisen Nachdruckes, der auszugsweisen oder vollständigen Wiedergabe, der Speicherung in Datenverarbeitungsanlagen und der Übersetzung, vorbehalten.

Printed in Germany.

ISBN 978-3-8440-8315-6

ISSN 1867-7088

Shaker Verlag GmbH • Am Langen Graben 15a • 52353 Düren

Telefon: 02421 / 99 0 11 - 0 • Telefax: 02421 / 99 0 11 - 9

Internet: [www.shaker.de](http://www.shaker.de) • E-Mail: [info@shaker.de](mailto:info@shaker.de)

## **Vorwort**

Die vorliegende Abschlussarbeit beschäftigt sich mit vielfältigen Performanceaspekten von Single-Page-Applications aus Sicht der Softwareentwicklung. Im Unterschied zu klassischen Webanwendungen, bei denen zwischen verlinkten Webseiten navigiert wird, erfolgt bei dieser Art von Anwendungen die algorithmische Verarbeitung der Präsentationsschicht mit Hilfe von nur einer HTML-Basisseite innerhalb des Webclients (z.B. Browsers, Web-Apps). Benötigte inhaltliche Veränderungen der Basisseite werden dynamisch vom Webserver nachgeladen, ohne jedoch - wie im Falle einer verlinkten Webseite - einen neuen Sitzungszustand zu erzeugen.

Aus diesem Architekturprinzip resultieren vielfältige Vorteile, wie z.B. die Reduktion der Kommunikation zwischen Webclient und Webserver, die Entlastung des Webserver, woraus eine verbesserte Skalierbarkeit resultiert, aber auch die Entkopplung der zu entwickelnden Client- und Serverkomponenten. Für die Anwender entsprechender Lösungen vergegenwärtigen sich die Vorteile in einer schnelleren Interaktionsfähigkeit, grafisch moderneren Nutzerschnittstellen oder auch in den einhergehenden Offline-Fähigkeiten der so entwickelten Anwendungen.

Gerade im Zusammenhang mit dem Einsatz von Single-Page-Applications in großen und heterogen gewachsenen IT-Architekturen gilt es den aufgezeigten Vorteilen die Nachteile Single-Page-Applications gegenüberzustellen und hinsichtlich des Einsatzzweckes zu bewerten. Im Zusammenhang mit den Performanceeigenschaften leistet die vorliegende Abschlussarbeit dafür einen wichtigen Beitrag. Ausgangspunkt der Analysen waren dementsprechend initiale Ladezeiten, Anforderungen an eingesetzte Browsersysteme, Implikationen des JavaScript-Frameworks sowie das Ressourcenmanagement auf den für die Webclients eingesetzten Hardwaresystemen. Mit Hilfe einer umfangreichen Analyse werden vielfältige Ansätze zum Performance-Monitoring und -Tuning recherchiert und anhand konkreter Messbeispiele im Detail bewertet.

Mit der Erstellung eines Konzepts zum Performance-Tuning zu entwickelnder Single-Page-Applications unterbreitet der Autor schließlich einen generischen Leitfaden für die praktische Anwendung. Dabei wird u.a. auf das Monitoring von Latenz oder auch Memory-Leaks eingegangen, aber auch auf Optimierungen innerhalb des Quellcodes (JS, CSS, DOM). Auf der Grundlage eines Industrieprojekts erfolgt schließlich die schrittweise Validierung des vorgeschlagenen Konzepts zum Performance-Tuning.

Berlin, November 2021

Andreas Schmietendorf  
*Betreuer der Abschlussarbeit*



## **Abstract**

Ziel der vorliegenden Arbeit ist es, ein allgemeingültiges und frei nutzbares Konzept für Performanceoptimierungen in Single-Page-Applications zu entwerfen. Dabei wird folgende Forschungsfrage gestellt: Wie lässt sich Performance in Single-Page-Applications optimieren? Zur Beantwortung der Forschungsfrage wurden einleitend Grundlagen im Bereich Performance-Tuning, Webperformance und Single-Page-Applications erläutert. Danach wurden Metriken, Messtechniken und gängige Problemfelder für Webperformance anhand des aktuellen Forschungsstandes analysiert und auf Basis dessen ein Konzept entwickelt. Dieses wurde abschließend zur Verifizierung an einem Realbeispiel verprobt. Entstanden ist ein aus drei Schritten bestehendes Konzept. Hierzu zählen Vorbetrachtung, initiale Messung und Optimierung. Für jeden Schritt wurden Handlungsanweisungen und Checklisten bereitgestellt, um eine Durchführung zu ermöglichen. Anhand des Konzeptes konnte die Performance einer produktiven Anwendung, in einem Projekt bei der DB Systel GmbH, in einzelnen Teilen deutlich verbessert werden. Dies zeigt die Wirksamkeit von gezielten Maßnahmen für Performanceoptimierungen mit Hilfe des Konzeptes. Auch wenn gleichzeitig nicht alle Problembereiche genauer analysiert werden konnten.



## Inhaltsverzeichnis

<b>Abstract</b>	<b>III</b>
<b>Akronyme</b>	<b>VII</b>
<b>Abbildungsverzeichnis</b>	<b>IX</b>
<b>1 Einleitung</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Vorgehensweise und Ziel der Arbeit . . . . .	2
<b>2 Grundlagen</b>	<b>3</b>
2.1 Was ist Performance? . . . . .	3
2.1.1 Was ist Web-Performance? . . . . .	3
2.1.2 Rendering- und Runtime-Performance . . . . .	6
2.2 Auswirkungen der Performance auf eine Anwendung . . . . .	8
2.3 Was ist eine Single Page Application? . . . . .	9
2.4 Was ist Performance Tuning? . . . . .	13
<b>3 Analyse bestehender Ansätze zur Performanceoptimierung</b>	<b>15</b>
3.1 Monitoring von Performance in Single-Page-Applications . . . . .	15
3.1.1 Analyse bestehener KPIs . . . . .	15
3.1.2 Tools zur Messung von Performance . . . . .	16
3.2 Performance Tuning mit JavaScript . . . . .	21
3.2.1 Performante Interaktion mit dem DOM . . . . .	22
3.2.2 JavaScript-Schleifen und Funktionsbasierte Iteration . . . . .	25
3.2.3 Weitere Ansätze und Empfehlungen . . . . .	28
3.3 Weitere Performance Tuning Ansätze im Webkontext . . . . .	30
3.3.1 Optimierungen mit CSS . . . . .	30
3.3.2 Spezielle Optimierungen in Single-Page-Applications . . . . .	33
<b>4 Erstellung des Konzeptes</b>	<b>37</b>
4.1 Vorbetrachtungen . . . . .	37
4.2 Bestandsaufnahme und initiale Messung . . . . .	39
4.3 Optimierung des Codes . . . . .	40
<b>5 Einschätzung der Ergebnisse und Ausblick</b>	<b>43</b>
<b>Literaturverzeichnis</b>	<b>45</b>

<b>A</b>	<b>Anhang</b>	<b>51</b>
A.1	Skalierte SPA-Architektur . . . . .	51
A.2	Vergleich der Ausführungszeit von JavaScript-Schleifen . . . . .	51
A.3	Beispielimplementierung von Duffs Device . . . . .	52
A.4	Beispiel einer JavaScript Closure . . . . .	53
A.5	Performanceunterschiede bei CSS-Selektoren . . . . .	53