

MARTIN DUSCHEK, 67664, 16MI1-B

EVOLUTION VON CODE BEI MAJOR-RELEASES VON  
PROGRAMMIERSPRACHEN

EVOLUTION VON CODE BEI MAJOR-RELEASES  
VON PROGRAMMIERSPRACHEN

MARTIN DUSCHEK, 67664, 16MI1-B

**HITWK**

Hochschule für Technik,  
Wirtschaft und Kultur Leipzig

am Beispiel von PHP

Juli 2019

Martin Duschek, 67664, 16MI1-B: *Evolution von Code bei Major-Releases von Programmiersprachen*, am Beispiel von PHP, © Juli 2019

# INHALTSVERZEICHNIS

---

## I EINLEITUNG

1	EINLEITUNG	2
1.1	Motivation	2
1.2	Aufgabe	2
1.3	Aufbau	2
2	GRUNDLAGEN	3
2.1	IEEE-Standard Softwarezyklen	3
2.2	Software Maintenance	3
2.3	PHP	3

## II PRAKTIKUMSBERICHT

3	MAJOR-RELEASE PHP7	5
3.1	Neue Funktionen	5
3.2	Veraltete Funktionen	5
3.3	Geänderte Funktionen	5
4	UNTERSUCHUNG GEEIGNETER MITTEL	6
4.1	Lauffähigkeit historischen Codes	6
4.1.1	Codeverwaltung	6
4.1.2	Lokale Entwicklungsumgebung	6
4.1.3	Continuous Integration mittels Containern	6
4.2	Erkennung des zu ändernden Codes	6
4.2.1	Manuelle Erkennung	6
4.2.2	Automatisierte Erkennung	6
4.3	Refactoring	6
4.3.1	Unit-Tests	6
4.3.2	Search & Replace	6
4.3.3	Wrapping	6

## LISTINGS

---

## Teil I

### EINLEITUNG

## EINLEITUNG

---

Am 03. Dezember 2015 erschien mit PHP 7.0.0 das erste Major-Release seit elf Jahren. Damit einhergehend wurde die Einstellung der Weiterentwicklung der vorhergehenden Version 5 für den 10. Januar 2019 angekündigt. Der Entwicklungsstopp führt dazu, dass Sicherheitslücken in der Implementation der alten Version nicht mehr geschlossen werden, was wiederum dazu führt, dass bereits ausgelieferte Software angreifbar wird sobald neue Lücken gefunden werden.

Derzeit setzen 79,1% der 10 Millionen meistgenutzten Webseiten PHP als serverseitige Programmiersprache ein, davon 61,5% PHP in der veralteten Version 5<sup>1</sup>. Diese Installationen können allesamt als unsicher eingestuft werden. Seit der letzten Veröffentlichung unter Version 5 wurden vier neue Schwachstellen veröffentlicht<sup>2</sup>, die in unterstützten Versionen bereits geschlossen wurden.

### 1.1 MOTIVATION

### 1.2 AUFGABE

Ziel dieser Arbeit ist die Evaluierung verschiedener Techniken und Technologien, die ein Upgrade der Programmiersprache in Softwareprojekten einfacher und nachhaltig gestalten oder erst in effizienter Weise ermöglichen.

### 1.3 AUFBAU

---

<sup>1</sup> W3Techs, „Usage statistics of PHP for websites“, <https://w3techs.com/technologies/details/pl-php/all/all>

<sup>2</sup> CVE details, „PHP 5.6.40 Security Vulnerabilities“, [https://www.cvedetails.com/vulnerability-list/vendor\\_id-74/product\\_id-128/version\\_id-298516/PHP-PHP-5.6.40.html](https://www.cvedetails.com/vulnerability-list/vendor_id-74/product_id-128/version_id-298516/PHP-PHP-5.6.40.html)

## GRUNDLAGEN

---

### 2.1 IEEE-STANDARD SOFTWAREZYKLEN

Der Standard **IEEE 14764** beschreibt den Prozess der Wartung von Software bis zu deren Einstellung. Darin wird unter Anderem beschrieben, welche Schritte bei der Migration von Software zu befolgen sind, sobald diese an eine neue Umgebung angepasst werden muss.

### 2.2 SOFTWARE MAINTENANCE

### 2.3 PHP

**PHP** (kurz für "PHP: Hypertext Preprocessor") ist eine Skriptsprache, welche seit 1994 entwickelt wird und seit 1995 Open-Source bereitgestellt wird. Obwohl **PHP** viele Einsatzzwecke abdeckt, wird es hauptsächlich dazu genutzt, dynamische Websites zu programmieren.



## Teil II

# PRAKTIKUMSBERICHT

## MAJOR-RELEASE PHP7

---

### 3.1 NEUE FUNKTIONEN

### 3.2 VERALTETE FUNKTIONEN

### 3.3 GEÄNDERTE FUNKTIONEN

## UNTERSUCHUNG GEEIGNETER MITTEL

---

### 4.1 LAUFFÄHIGKEIT HISTORISCHEN CODES

#### 4.1.1 *Codeverwaltung*

#### 4.1.2 *Lokale Entwicklungsumgebung*

#### 4.1.3 *Continuous Integration mittels Containern*

### 4.2 ERKENNUNG DES ZU ÄNDERNDEN CODES

Um alten Code migrieren zu können, müssen alle Stellen gefunden werden, die in ihrer ursprünglichen Form in der neuen Umgebung nicht lauffähig wären.

#### 4.2.1 *Manuelle Erkennung*

#### 4.2.2 *Automatisierte Erkennung*

### 4.3 REFACTORING

#### 4.3.1 *Unit-Tests*

#### 4.3.2 *Search & Replace*

#### 4.3.3 *Wrapping*