

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Kontext . . . . .	1
1.2	Problemdefinition . . . . .	3
1.3	Anwendungsszenarien . . . . .	4
1.4	Beitrag dieser Arbeit . . . . .	5
1.5	Struktur der Arbeit . . . . .	6
1.6	Hinweise zur erstellten Software . . . . .	7
1.7	Hinweise zu Sprache und Notation . . . . .	7
<b>2</b>	<b>Grundlagen der Überwachung und Instrumentierung von Softwaresystemen</b>	<b>9</b>
2.1	Klassifizierung . . . . .	9
2.1.1	Ansatzpunkte für Überwachung und Instrumentierung . . . . .	10
2.1.2	Sammlung von Analysedaten . . . . .	11
2.1.3	Auswertung und Analyse . . . . .	13
2.1.4	Zusammenfassende Darstellung . . . . .	13
2.2	Instrumentierung . . . . .	15
2.2.1	Instrumentierungspunkte . . . . .	15
2.2.2	Statische Instrumentierung . . . . .	16
2.2.3	Dynamische Instrumentierung . . . . .	17
2.3	Automatisierte Überwachung . . . . .	17
2.3.1	Regelkreis-basierte Überwachung . . . . .	18
2.3.2	Autonomic Computing . . . . .	22
2.4	Begriffsbestimmung: Ereignis und Ereignisstrom . . . . .	25
2.4.1	Definition: Ereignis . . . . .	25
2.4.2	Definition: Ereignisstrom . . . . .	27
2.5	Zusammenfassung . . . . .	28

<b>3</b>	<b>Der Windows Monitoring Kernel</b>	<b>29</b>
3.1	Zielstellung der Implementierung . . . . .	29
3.1.1	Anforderungen . . . . .	29
3.1.2	Architektur . . . . .	30
3.2	Ereigniserzeugung . . . . .	31
3.2.1	Grundlagen . . . . .	31
3.2.2	Betriebssystemkernereignisse . . . . .	33
3.2.3	Anwendungsspezifische Ereignisse . . . . .	35
3.2.4	Ereignisse in DLLs . . . . .	38
3.2.5	Ereignisse in Treibern . . . . .	38
3.3	Ereignisaufzeichnung . . . . .	38
3.3.1	Pufferverwaltung . . . . .	38
3.3.2	Synchronisation . . . . .	40
3.3.3	Logdateiverwaltung . . . . .	42
3.4	Werkzeuge . . . . .	43
3.4.1	Steuerung der Ereignisaufzeichnung . . . . .	43
3.4.2	Auswertung von Logdateien . . . . .	43
3.5	Evaluation . . . . .	46
3.5.1	Testumgebung und Methodologie . . . . .	46
3.5.2	Messungen und Analysen . . . . .	47
3.5.3	Vergleich mit alternativen Systemen . . . . .	50
3.6	Zusammenfassung . . . . .	52
<b>4</b>	<b>Online Verarbeitung von Ereignissen</b>	<b>53</b>
4.1	Zielstellung der Implementierung . . . . .	53
4.2	Beschreibung von Mustern in Ereignisströmen . . . . .	54
4.2.1	Ereignistypen . . . . .	55
4.2.2	Konstellation . . . . .	58
4.2.3	Relationen . . . . .	62
4.2.4	Weitere Sprachelemente . . . . .	64
4.3	Automaten zur Mustererkennung . . . . .	65
4.3.1	Grundlagen . . . . .	65
4.3.2	Automatenerzeugung . . . . .	67
4.3.3	Compiler . . . . .	75
4.4	Laufzeitumgebung zur Mustererkennung . . . . .	80

4.4.1	Abarbeitungsmodell . . . . .	81
4.4.2	Verarbeitung von Ereignissen . . . . .	82
4.4.3	Verwaltung von Regeln . . . . .	84
4.4.4	Pufferverwaltung und Synchronisation . . . . .	85
4.4.5	Systemaufruf Schnittstelle . . . . .	88
4.5	Reaktion auf erkannte Muster . . . . .	89
4.5.1	Kernelmode Skripte . . . . .	89
4.5.2	Usermode Callbacks . . . . .	93
4.6	Werkzeuge . . . . .	96
4.7	Evaluation . . . . .	97
4.7.1	Messungen und Analysen . . . . .	97
4.7.2	Vergleich mit alternativen Systemen . . . . .	99
4.8	Zusammenfassung . . . . .	100
<b>5</b>	<b>Fallstudien und Leistungsbewertung</b>	<b>101</b>
5.1	Einleitung . . . . .	101
5.2	Analyse der Bearbeitung von Anfragen an einen Webserver . . . . .	101
5.2.1	Instrumentierte Ausführung . . . . .	101
5.2.2	Ergebnisse . . . . .	102
5.2.3	Diskussion . . . . .	105
5.3	Analyse von Vorgängen im Betriebssystemkern . . . . .	105
5.3.1	Bootvorgang . . . . .	105
5.3.2	Quantumlängen und Scheduling . . . . .	107
5.3.3	Diskussion . . . . .	109
5.4	Softwareprüfstand . . . . .	110
5.4.1	Erkennung von Wartezeiten . . . . .	111
5.4.2	Erkennung von Fehlern . . . . .	112
5.4.3	Analyse von Synchronisationsvorgängen . . . . .	112
5.4.4	Analyse von Seitenzugriffsfehlern . . . . .	114
5.4.5	Überwachung von Annahmen und Grenzwerten . . . . .	114
5.4.6	Diskussion . . . . .	115
5.5	Adaption des Betriebssystems . . . . .	115
5.6	Online Verarbeitung von Ereignissen - Komplexitätsanalyse . . . . .	116
5.6.1	Charakterisierung von Ereignisströmen . . . . .	117
5.6.2	Modellierung der Ereignisstromverarbeitung . . . . .	119

5.6.3	Diskussion . . . . .	123
5.7	Weitere Untersuchungen . . . . .	123
5.7.1	Fein-granulare Zuteilung von CPU Zeit . . . . .	123
5.7.2	Instrumentierung von Spinlocks . . . . .	126
5.8	Zusammenfassung . . . . .	130
<b>6</b>	<b>Verwandte Arbeiten</b>	<b>131</b>
6.1	Instrumentierungstechniken . . . . .	131
6.1.1	Klassifikation . . . . .	131
6.1.2	Implementierungen . . . . .	131
6.1.3	Diskussion - Einordnung des WMK . . . . .	138
6.2	Verarbeitung von Ereignisströmen . . . . .	139
6.2.1	Forschungskontext . . . . .	139
6.2.2	Sprachen zur Beschreibung von Ereigniskonstellationen . . . . .	141
6.2.3	Diskussion - Einordnung der entwickelten Laufzeitumgebung . . . . .	143
6.3	Zusammenfassung . . . . .	143
<b>7</b>	<b>Zusammenfassung und Ausblick</b>	<b>145</b>
	<b>Literaturverzeichnis</b>	<b>149</b>
<b>A</b>	<b>WMK Ereignistypen</b>	<b>157</b>