

RESEARCH

Andreas Kohne

Cloud- Föderationen

SLA-basierte
VM-Scheduling-Verfahren



Springer Vieweg

Cloud-Föderationen

Andreas Kohne

Cloud-Föderationen

SLA-basierte
VM-Scheduling-Verfahren

Andreas Kohne
Dortmund, Deutschland

Zugl.: Dissertation, Technische Universität Dortmund, 2017

ISBN 978-3-658-20972-8 ISBN 978-3-658-20973-5 (eBook)
<https://doi.org/10.1007/978-3-658-20973-5>

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

Springer Vieweg

© Springer Fachmedien Wiesbaden GmbH, ein Teil von Springer Nature 2018

Das Werk einschließlich aller seiner Teile ist urheberrechtlich geschützt. Jede Verwertung, die nicht ausdrücklich vom Urheberrechtsgesetz zugelassen ist, bedarf der vorherigen Zustimmung des Verlags. Das gilt insbesondere für Vervielfältigungen, Bearbeitungen, Übersetzungen, Mikroverfilmungen und die Einspeicherung und Verarbeitung in elektronischen Systemen.

Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Werk berechtigt auch ohne besondere Kennzeichnung nicht zu der Annahme, dass solche Namen im Sinne der Warenzeichen- und Markenschutz-Gesetzgebung als frei zu betrachten wären und daher von jedermann benutzt werden dürften.

Der Verlag, die Autoren und die Herausgeber gehen davon aus, dass die Angaben und Informationen in diesem Werk zum Zeitpunkt der Veröffentlichung vollständig und korrekt sind. Weder der Verlag noch die Autoren oder die Herausgeber übernehmen, ausdrücklich oder implizit, Gewähr für den Inhalt des Werkes, etwaige Fehler oder Äußerungen. Der Verlag bleibt im Hinblick auf geografische Zuordnungen und Gebietsbezeichnungen in veröffentlichten Karten und Institutionsadressen neutral.

Gedruckt auf säurefreiem und chlorfrei gebleichtem Papier

Springer Vieweg ist ein Imprint der eingetragenen Gesellschaft
Springer Fachmedien Wiesbaden GmbH und ist Teil von Springer Nature
Die Anschrift der Gesellschaft ist: Abraham-Lincoln-Str. 46, 65189 Wiesbaden, Germany



Widmung:

Ich widme diese Arbeit meiner Frau Linda
und meiner Familie.

Danksagung

Diese Arbeit, die neben meinem Beruf bei der Firma Materna GmbH in Dortmund entstanden ist, konnte nur mit der Hilfe und unermüdlichen Unterstützung vieler entstehen. An dieser Stelle möchte ich mich bei allen, die mich über die Jahre hinweg begleitet, unterstützt und ermuntert haben ausdrücklich bedanken:

An erster Stelle möchte ich mich bei meiner Frau Linda bedanken. Sie hat mich fünf Jahre unterstützt und mir die nötige Zeit für diese Arbeit zugestanden.

Weiterhin möchte ich meiner Familie danken, die mich immer unterstützt und darin bestärkt hat, dieses Projekt zu vollenden.

Ich möchte mich bei Prof. Dr.-Ing. Olaf Spinczyk und beim Lehrstuhl 12 der Informatik an der Technischen Universität Dortmund bedanken, die durch ihre Anregungen diese Arbeit an vielen Stellen entscheidend geprägt haben.

Ebenfalls bedanke ich mich bei der Materna GmbH, die es mir ermöglicht hat, diese Arbeit zu schreiben. Ganz besonders danke ich meinen Kollegen Franz-Josef Stewing und Dr. Ingo Lück. Sie haben mir die Möglichkeit gegeben, das Thema intern weiterzuentwickeln und standen immer mit Rat und Tag zur Seite. Zusätzlich bedanke ich mich bei den ehemaligen Kollegen aus der Abteilung Academic Cooperations: Marco Pfahlberg, Marc Spor, Damian Pasternak, Julian Gaedeke und Marcel Krüger. Sie haben mit ihrer Arbeit und ihren Ideen die Arbeit maßgeblich beeinflusst. Ich bedanke mich weiterhin bei Dr. Thomas Kalinke und dem RZ-Team für die freundliche Zusammenarbeit und die Bereitstellung der Workload Traces.

Ich danke Dr. Lars Nagel von der Universität Mainz. Er hat diese Arbeit von Anfang an begleitet und sie an vielen Stellen positiv beeinflusst.

Abschließend möchte ich mich bei allen bedanken, die hier nicht namentlich erwähnt wurden, mich aber trotzdem auf die ein oder andere Weise bei dieser Arbeit unterstützt haben.

Vielen Dank!

Andreas Kohne

Inhaltsverzeichnis

Abbildungsverzeichnis	XV
Tabellenverzeichnis	XVII
Abkürzungen	XXI
1 Einleitung	1
1.1 Cloud-Föderationen	3
1.2 Service Level Agreements	4
1.3 Cloud Scheduling	5
1.4 Ziel der Arbeit	9
1.5 Wissenschaftliche Fragestellungen	9
1.6 Forschungsbeitrag	10
1.7 EASI CLOUDS	13
1.8 Beitrag Anderer	14
1.9 Aufbau der Arbeit	15
I Grundlagen und verwandte Arbeiten	17
2 Cloud Computing	19
2.1 Entwicklung	19
2.2 Service-Arten	24
2.3 Umsetzungsarten	26
3 Service Level Agreements	29
3.1 Cloud SLAs	30
3.2 Automatische SLA-Verarbeitung	31
3.3 SLAs in Forschung und Industrie	32

4	Cloud Föderationen	35
4.1	Inter-Clouds	35
4.2	Cloud Broker	39
4.3	Inter-Cloud-Forschung	43
5	Cloud Scheduling	47
5.1	VM-Scheduling	48
5.2	Zentrales Scheduling	49
5.3	Dezentrales Scheduling	50
5.4	Multi-Level Scheduling	50
5.5	Scheduling in Cloud-Umgebungen	51
II	FederatedCloudSim	55
6	Simulation von Cloud Systemen	57
6.1	Simulationsanforderungen	60
6.2	Frameworkauswahl	64
6.3	CloudSim	68
6.3.1	Aufbau	69
6.3.2	Simulationsablauf	75
6.3.3	Fehlende Funktionalitäten	76
7	Das FCS-Framework	81
7.1	Konzeption	81
7.2	Aufbau	84
7.3	Konfiguration von FederatedCloudSim	86
7.4	Simulationsablauf	88
7.5	Overprovisioning	92
7.6	Host-Energiemodell	94
7.7	Service Request	96
7.8	Initiale VM-Verteilung	97
7.9	Service Level Agreements (SLAs)	100
7.9.1	SLAs und FCS	100
7.9.2	SLA-Manager	105

7.10	Workload-Verarbeitung	109
7.11	Föderation	110
7.12	Virtuelle Cloud Service Provider (vCSPs)	111
7.13	VM-Scheduling	112
7.14	CloudAccount	117
7.14.1	Das Finanzmodell	118
7.15	Auktionsplattform	123
7.15.1	Auktionen in FederatedCloudSim	125
7.16	Vertrauensfaktor	130
7.17	Monitoring und Logging	131
7.17.1	Monitoring	131
7.17.2	End-Log	132
7.17.3	Accounting-Log	134
7.17.4	Migrations-Log	136
7.17.5	Gephi	136
7.18	Zusätzliche Werkzeuge	140
7.19	Relevante Metriken	141
7.20	Performance-Analyse des Frameworks	146
8	Auswahl von Eingabedaten	151
8.1	Grid und Cloud Workloads	153
8.2	Reale Cloud Workload Traces	157
8.2.1	Bitbrains Datensatz	157
8.2.2	Materna Datensatz	159
III	Scheduling-Strategien	161
9	Modell	163
9.1	Das initiale Scheduling	165
9.2	Das SLA-basierte Scheduling	167
9.3	Taxonomie	170
10	DC-Scheduling	181
10.1	Simulationsszenarien	182

10.2	Reaktive Scheduler	186
10.2.1	FirstFit	186
10.2.2	HPGWF	187
10.2.3	HPGOP	189
10.2.4	HPGBF	189
10.2.5	MMBF	190
10.2.6	MMOP	191
10.2.7	MMWF	191
10.2.8	Evaluation der Intra-DC-Scheduler	191
10.3	Initiale VM-Verteilung	196
10.4	Proaktive Scheduler	199
10.4.1	Energieeffizientes VM-Scheduling	199
10.4.2	Lastausgleichendes VM-Scheduling	205
10.4.3	Kombination der proaktiven Scheduler	207
10.5	SLA-Warnzeitverzögerung	209
10.6	Kleine vs. große Hosts	211
10.7	Kreuzvalidierung	215
10.8	Einordnung in die Taxonomie	219
10.9	Zusammenfassung	220
11	CSP-Scheduling	225
11.1	Simulationsszenarien	228
11.2	FirstFit	230
11.3	MML	230
11.4	SDHG	232
11.5	SDHL	232
11.6	SDLG	233
11.7	SDLL	233
11.8	SLLG	234
11.9	SLLL	234
11.10	Untersuchung mit zwei DCs	235
11.11	Untersuchung mit drei DCs	239
11.12	Kreuzvalidierung	242
11.13	Einordnung in die Taxonomie	244
11.14	Zusammenfassung	245

12 Föderations-Scheduling	249
12.1 Simulationsszenarien	249
12.2 FirstFit	252
12.3 FirstFitGreedy	253
12.4 FairFit	254
12.5 CheapestFit	254
12.6 RelianceScope	254
12.7 CheapestReliability	255
12.8 Auktions-Scheduler	256
12.8.1 2RoundVmCheapestFit	256
12.8.2 2RoundVmCheapestReliability	256
12.9 Evaluation Szenario 1	257
12.10 Evaluation Szenario 2	259
12.11 Evaluation Szenario 3	262
12.12 Evaluation Szenario 4	264
12.13 Kreuzvalidierung	268
12.14 Häufige VM-Migrationen	271
12.15 Grafische Simulationsauswertung mit Gephi	273
12.16 Einordnung in die Taxonomie	274
12.17 Zusammenfassung	274
13 Zusammenfassung	279
14 Ausblick	285
A Anhang I: Weiterführende Informationen	313
A.1 Virtualisierung	313
A.2 Service-Arten	315
A.3 Service Level Agreements	317
A.4 Scheduling	318
A.5 Simulationsarten	319
A.6 CloudSim-Erweiterungen	320
A.7 Untersuchung von Finanzen im Cloud-Umfeld	323
A.8 CloudAccount Implementierung	325
A.9 Auktionen	327

A.10	Zusätzliche Werkzeuge für FCS	330
A.10.1	FCS Configurator	330
A.10.2	TraceShortener	332
A.10.3	TraceWorkloadGenerator	332
A.10.4	Workloadparser	333
A.10.5	WorkloadFormatter	333
A.10.6	SWF-Modifier	334
B	Anhang II: FCS-Konfiguration und Log-Dateien	335
B.1	Finanzielle Konfiguration	335
B.1.1	Simulationen weiterer Workloads	336
B.2	Finanzkonfiguration für Szenario 2	337
B.3	Finanzkonfiguration für Szenario 4	337
B.4	Trace Files	338
B.5	Trace-Analyse	342
B.6	Simulationskonfiguration	349
B.6.1	XML-Konfiguration	350
B.6.2	Properties	354
B.6.3	SWF	357
B.7	Das End-Log	358
B.8	Das Accounting-Log	362
B.9	Das Migrations-Log	366
C	Eigene Veröffentlichungen	373

Abbildungsverzeichnis

1.1	Beispielhafter Aufbau einer Cloud-Föderation	5
1.2	Grafische Darstellung des SLA-Ablaufs	6
1.3	Der Multi-Level-Scheduler	7
2.1	Vergleich: Grid Computing / Cloud Computing	21
2.2	Grafische Darstellung der unterschiedlichen Cloud Service- Arten	25
4.1	Abbildung der Inter-Cloud-Taxonomie	37
4.2	Das Inter-Cloud Modell	44
6.1	Abstrakter Prozess zur Erstellung eines Simulators	59
6.2	Ablauf einer allgemeinen Simulation	61
6.3	Elemente für die Simulation einer Cloud-Föderation	62
6.4	Eingabe und Ausgabe der Simulation	65
6.5	Schematischer Aufbau von CloudSim	73
6.6	Ablauf einer Simulation in CloudSim	76
7.1	Schematischer Ablauf der schrittweisen Entwicklung von FCS	83
7.2	Schematischer Aufbau von FederatedCloudSim	86
7.3	Schematische Darstellung des Simulationsablaufs	89
7.4	Beispielhafte Darstellung des RAM-Overprovisionings	93
7.5	Grafische Darstellung des Energiemodells für Hosts	95
7.6	Schematische Darstellung einer VM-Erstellung	97
7.7	Darstellung der initialen VM-Verteilstrategien	99
7.8	Das zentrale Föderations-Scheduling mit vCSPs	113
7.9	Grafische Darstellung des Scheduling-Ablaufs mit SLA- Manager und Consolidation Calls	116

7.10	Das CloudAccount-Modell	119
7.11	Der Accounting-Ablauf	122
7.12	Modell des Ablaufs von Auktionen in FCS und ihre Integration in das Scheduling der Föderationsebene . .	127
7.13	Die grafische Benutzeroberfläche von Gephi	137
7.14	Grafische Auswertung der VM-Migrationen einer Si- mulation mit fünf CSPs mit je drei DCs in Gephi . . .	139
7.15	Ergebnisse der FCS-Performance-Analyse	150
9.1	Das initiale Scheduling	166
9.2	Das SLA-basierte Scheduling	168
9.3	Taxonomie der Cloud-Scheduler	172
9.4	Systematik der Scheduling-Zielfunktion	179
10.1	Grafische Darstellung des Simulationsszenarios der Intra- DC-Ebene	183
10.2	Grafische Darstellung der Veränderung der aktiven Host-Anzahl durch den proaktiven EE-Scheduler über einen Monat	203
10.3	Vergleich der technischen Metriken	222
10.4	Vergleich der finanziellen Metriken	223
11.1	Grafische Darstellung des Simulationsszenarios der Inter- DC-Ebene mit einem CSP und drei DCs	231
12.1	Grafische Darstellung des Simulationsszenarios der Inter- CSP-Ebene mit fünf CSPs und je drei DCs	251
12.2	Grafische Auswertung der VM-Migrationen einer Fö- derationssimulation mit Gephi	275
12.3	Vergleich der CSP-Finanzergebnisse (gerundet) mit und ohne Föderationsteilnahme	277
A.1	Darstellung von Hypervisor Typ 1 und Typ 2	314
B.1	Auszug aus Grid Trace KTH-SP2-1996-0	341
B.2	Auszug aus Materna Cloud Trace	342

Tabellenverzeichnis

- 7.1 Tabellarische Darstellung der Ergebnisse der FCS-Performance-Analyse 148

- 10.1 Zusammenfassung der DC-Scheduling-Ergebnisse ohne weitere Verbesserungen 192
- 10.2 Zusammenfassung der DC-Scheduling-Ergebnisse mit SLA-basierter initialer VM-Verteilung 196
- 10.3 Zusammenfassung der DC-Scheduling-Ergebnisse mit RAM-Ressourcen-basierter initialer VM-Verteilung . . 198
- 10.4 Zusammenfassung der Simulationsergebnisse des MMWF-Schedulers in Kombination mit dem proaktiven EE-Scheduler 202
- 10.5 Zusammenfassung der Simulationsergebnisse des MMWF-Schedulers in Kombination mit dem proaktiven LB-Scheduler 206
- 10.6 Zusammenfassung der Simulationsergebnisse des MMWF-Schedulers in Kombination mit den proaktiven EE- und LB-Schedulern 208
- 10.7 Zusammenfassung der Ergebnisse zur Ermittlung der Auswirkungen unterschiedlicher SLA-Warnzeiten auf die Simulationsergebnisse 209
- 10.8 Zusammenfassung der Simulationsergebnisse mit unterschiedlichen Host-Konfigurationen 212
- 10.9 Zusammenfassung der durchschnittlichen Anzahl der VMs pro Host für die unterschiedlichen Experimente . 214

10.10	Zusammenfassung der Ergebnisse der Intra-DC-Scheduler inklusive EE-Erweiterung und initialer VM-Verteilung nach RAM-Ressourcen für den Bitbrains RnD Trace (Monat 1) mit 500 VMs	216
10.11	Zusammenfassung der Ergebnisse der Intra-DC-Scheduler inklusive EE-Erweiterung und initialer VM-Verteilung nach RAM-Ressourcen für den Materna Trace (Monat 1) mit 520 VMs	217
10.12	Tabellarische Klassifizierung der vorgestellten Intra-DC-Scheduler anhand der Taxonomie	219
11.1	Zusammenfassung der Ergebnisse der Inter-DC-Scheduler für zwei DCs inklusive EE-Erweiterung und initialer VM-Verteilung nach RAM-Ressourcen für den Bitbrains Trace FastStorage mit 1.250 VMs	235
11.2	Zusammenfassung der Ergebnisse der Inter-DC-Scheduler für drei DCs inklusive EE-Erweiterung und initialer VM-Verteilung nach RAM-Ressourcen für den Bitbrains Trace FastStorage mit 1.250 VMs	240
11.3	Zusammenfassung der Ergebnisse der Inter-DC-Scheduler für drei DCs inklusive EE-Erweiterung und initialer VM-Verteilung nach RAM-Ressourcen für den Bitbrains Trace RnD (Monate 1-3) mit 1.500 VMs	244
11.4	Zusammenfassung der Ergebnisse der Inter-DC-Scheduler für drei DCs inklusive EE-Erweiterung und initialer VM-Verteilung nach RAM-Ressourcen für den Materna Trace (Monate 1-3) mit 1.594 VMs	245
11.5	Tabellarische Klassifizierung der vorgestellten Inter-DC-Scheduler anhand der Taxonomie	246
12.1	Föderationsszenario mit fünf CSPs	250
12.2	Zusammenfassung der Simulationsergebnisse mit fünf CSPs und drei DCs pro CSP aus Sicht von CSP-1	257
12.3	Zusammenfassung der Simulationsergebnisse mit fünf CSPs und einem DC pro CSP aus Sicht von CSP-1	260

12.4	Zusammenfassung der Ergebnisse zur Untersuchung des Kaufs zusätzlicher Hosts	263
12.5	Zusammenfassung der Simulationsergebnisse der vertrauensbasierten Scheduler	265
12.6	Zusammenfassung der Simulationsergebnisse mit fünf CSPs und einem DC pro CSP aus Sicht des CSPs mit dem Bitbrains RnD Traces (Monat 1) mit 500 VMs . .	269
12.7	Zusammenfassung der Simulationsergebnisse mit fünf CSPs und einem DC pro CSP aus Sicht des CSPs mit dem Materna Traces (Monat 1) mit 520 VMs	270
12.8	Tabellarische Klassifizierung der vorgestellten Inter-CSP-Scheduler anhand der Taxonomie	276
B.1	Tabellarische Zusammenfassung der finanziellen Parameter für die Simulation der Intra-DC-Scheduler und den Bitbrains Fast Storage Trace	336
B.2	Zusammenfassung der finanziellen Simulationsparameter für den Bitbrains RnD Trace (Monat 1) mit 500 VMs	337
B.3	Zusammenfassung der finanziellen Simulationsparameter für den Materna Trace (Monat 1) mit 520 VMs . .	338
B.4	Zusammenfassung der finanziellen Simulationsparameter für das Föderationsszenario 2 mit fünf CSPs	339
B.5	Zusammenfassung der finanziellen Simulationsparameter für das Föderationsszenario 4 mit fünf CSPs	339
B.6	Auszug aus einer SWF-Datei	358
B.7	Ausschnitt aus einem Migrations-Log	367
B.8	Auszug aus einer Gephi-Nodes-Datei mit den ersten fünf Hosts	368
B.9	Auszug aus einer Gephi-Edges-Datei mit den ersten fünf Migrationsverbindungen	368

Abkürzungen

ACCORDS	Advanced Capabilities for CORDS
API	Application Programming Interface
ASCII	American Standard Code for Information Interchange
AWS	Amazon Web Services
BDA	BestDangerAverage
BMBF	Bundesministerium für Bildung und Forschung
BPM	Business Process Management
BRITE	Boston university Representative Internet Topology gEnerator
Capex	Capital expenditures
CDMI	Cloud Data Management Interface
CERN	Conseil Européen pour la Recherche Nucléaire
CEx	Cloud Exchange
CF	Cheapest Fit
CIS	Cloud Information Service
CLOUDS	Cloud Computing and Distributed Systems
CORDS	CompatibleOne Resource Description System
CPU	Central Processing Unit

CR	Cheapest Reliability
CS	CloudSim
CRM	Customer Relationship Management
CSP	Cloud Service Provider
DC	Data Center
EE	Energy Efficiency
EOL	End of Life
FairF	Fair Fit
FCFS	First Come First Served
FCS	FederatedCloudSim
FF	First Fit
FFG	First Fit Greedy
GB	Gigabyte
GHz	Gigahertz
GUI	Graphical User Interface
HPGBF	Highest Potential Growth Best Fit
HPGOP	Highest Potential Growth Overprovisioning
HPGWF	Highest Potential Growth Worst Fit
IaaS	Infrastructure as a Service
IDE	Integrated Development Environment
IdM	Identity Management

IoT	Internet of Things
IP	Internet Protocol
IT	Informationstechnologie
ITSM	IT Service Management
JDK	Java Development Kit
kWh	Kilowattstunde
LB	Load Balancer
MB	Megabyte
MHz	Megahertz
MIPS	Million Instructions Per Second
MMBF	Minimize Migrations Best Fit
MML	Minimize Migrations Local
MMOP	Minimize Migrations Overprovisioning
MMWF	Minimize Migrations Worst Fit
NAS	Network Attached Storage
NIST	National Institute of Standards and Technology
NS2	Network Simulator 2
OCCI	Open Cloud Computing Interface
OLA	Operational Level Agreement
Opex	Operational expenditures
OVF	Open Virtualization Format

PaaS	Platform as a Service
PC	Personal Computer
RAM	Random Access Memory
REST	Representational State Transfer
RS	Reliance Scope
RTE	Runtime Environment
RZ	Rechenzentrum
SaaS	Software as a Service
SAN	Storage Area Network
SDHG	SlaDangerHighGlobal
SDHL	SlaDangerHighLocal
SDLG	SlaDangerLowGlobal
SDLL	SlaDangerLowLocal
SLA	Service Level Agreement
SLLG	SlaLevelLowGlobal
SLLL	SlaLevelLowLocal
SLO	Service Level Objective
SOA	Service Oriented Architecture
SSO	Single-Sign-On
SWF	Standard Workload Format
vCSP	virtueller Cloud Service Provider

UC	Underpinning Contract
UT	Uptime
VM	Virtuelle Maschine
VMM	Virtual Machine Monitor
VPN	Virtual Private Network
WSLA	Web Service Level Agreement
XML	Extensible Markup Language
XSD	XML Schema Definition



1 Einleitung

Cloud Computing hat sich in den letzten Jahren von einem Hype-Thema zu einem ernstzunehmenden Wirtschaftszweig entwickelt (vgl. [17, 67]). Das Cloud Computing bietet dabei die Möglichkeit, schnell und flexibel IT-basierte Dienste zu bestellen, zu nutzen oder wieder abzubestellen. Ein großer Vorteil gegenüber einer starren lokalen IT sind die flexiblen Abrechnungsmodelle, die eine nutzungsgenaue Bezahlung erlauben. Diese Modelle werden auch als Pay-as-you-go oder On-Demand bezeichnet. Dadurch entsteht ein großer Nutzen für den Kunden, da er sich nicht mehr um die eigentliche Dienstleistung kümmern muss und sich somit auf den eigentlichen Zweck des Dienstes konzentrieren kann. Obwohl das Cloud Computing immer größeren Zuspruch findet, wächst es in Deutschland langsamer als im internationalen Vergleich (vgl. [126]). Dies liegt an den hohen Sicherheits- und Datenschutzerfordernissen und den damit einhergehenden Sicherheitsbedenken. Weiterhin muss zwischen der privaten und der beruflichen Nutzung von Cloud-Diensten unterschieden werden. Im privaten Umfeld ist die Nutzung von Cloud-Diensten vor allem durch die große Verbreitung von Smartphones und Tablets und deren Apps (meist Client-Server-Applikationen mit Cloud-Anbindung) bereits zur Normalität geworden. Durch diesen selbstverständlichen Umgang mit cloudbasierten Diensten im privaten Umfeld steigen auch die Anforderungen an die berufliche IT. Diese Entwicklung wird als Consumerization bezeichnet (vgl. [93]). Die internen IT-Dienstleister (vor allem großer Firmen und Behörden) müssen sicher immer öfter mit der Konkurrenz aus der Cloud vergleichen lassen. Somit steigt der Druck auf die IT-Abteilungen, entweder selbst eine private Cloud (engl.: private cloud) anzubieten, oder sich doch (in Teilen) der öffentlichen Cloud (engl.: public cloud) zu öffnen.

Die Nutzungszahlen von Cloud-Diensten sind in den letzten Jahren rapide gestiegen. Laut einer Studie der Firma RightScale, in der 1060 Unternehmen weltweit befragt wurden, lag die internationale Nutzung von Cloud-Diensten im Jahr 2016 bei 95% (6% reine private cloud, 71% hybrid cloud (Mischung aus private und public cloud) und 18% public cloud) unter den befragten Unternehmen (vgl. [127]). Eine Studie von KPMG gemeinsam mit der BITKOM aus dem Jahr 2015 besagt, dass 44% der deutschen Unternehmen Cloud-basierte Diensten nutzen und weitere 24% der befragten Unternehmen in der nahen Zukunft Cloud-Dienste nutzen wollen (vgl. [121]). Diese Zahlen deuten darauf hin, dass die einfache Nutzung von IT-Diensten aus der Cloud in den nächsten Jahren sicher noch weiter steigen wird.

Die Nutzung von Cloud-basierten Diensten soll so einfach wie möglich sein, wobei der eigentliche Ausführungsort und die Ausführungsumgebung für den Endkunden meist unbekannt sind. Das heißt, dass nicht in jedem Fall klar ist, wo die Daten abgelegt und wo sie bearbeitet werden. Die anbieterseitige Produktion von Cloud-Diensten ist dagegen höchst komplex. Es herrscht ein enormer Kostendruck und eine große Konkurrenz zwischen den Cloud-Diensteanbietern (engl. Cloud Service Provider (CSP)). Große CSPs wie Amazon, Microsoft, IBM und Google verteilen riesige Rechenzentren (engl. Data Center (DC)) über den gesamten Globus, um ihren Kunden einen möglichst lokalen Service mit niedrigen Latenzzeiten bieten zu können (vgl. [71]). Weiterhin kann durch die Geolokalität der DCs in Teilen sichergestellt werden, dass rechtliche Vorgaben der jeweiligen Landesregierungen eingehalten werden.

Die Verwaltung und der Betrieb solcher verteilten DC-Infrastrukturen ist sehr komplex. Cloud-Infrastrukturen lassen sich heutzutage nur noch über eine in weiten Teilen automatisierte IT bewerkstelligen. Das Ziel der CSPs ist eine möglichst vollständige Automatisierung von der Dienstbestellung über Online-Service-Portale, über die Provisionierung der benötigten IT-Infrastruktur und Applikationen, die Verwaltung der laufenden Systeme, bis hin zur Fehlerbehebung und Deprovisionierung (engl. End of Life (EOL)). Ein hohes Maß an Automatisierung

garantiert auch eine gleichbleibend hohe Service-Erbringung, da alle Schritte exakt reproduzierbar sind.

Eine Basistechnologie, die eine kommerzielle Nutzung vom Cloud-Diensten erst ermöglicht hat, ist die Virtualisierung. Dabei werden über eine spezielle Software, den sogenannten Hypervisor, die durch einen physischen Server bereitgestellten Ressourcen (wie Central Processing Unit (CPU), Random Access Memory (RAM), Speicherplatz und Netzwerkanbindungen) Software-seitig partitionierbar. Dies erlaubt es, mehrere VMs (virtuelle Maschinen) parallel auf einem physischen Server auszuführen (vgl. [136]). Hierdurch steigt die lokale Ressourcenausnutzung immens, da die Kopplung von einem Dienst auf einen Server aufgehoben wird und mehrere Dienste gleichzeitig die zur Verfügung gestellten Ressourcen nutzen können. Dies erlaubt den CSPs einen flexiblen und gewinnbringenden Dienstbetrieb. Ein großer Vorteil von VMs besteht darin, dass sie im laufenden Betrieb zwischen verschiedenen Servern migriert werden können. So können automatisch lokale Lastspitzen auf einzelnen physischen Servern damit ausgeglichen werden, dass eine oder mehrere VMs auf weitere Server migriert werden können.

1.1 Cloud-Föderationen

In den letzten zwei bis drei Jahren setzte sich ein weiterer Trend durch: Cloud-Föderationen. Dabei handelt es sich um einen Zusammenschluss mehrerer CSPs mit dem Ziel, ihren jeweiligen Kunden eine größere Anzahl an Diensten anbieten und bei Ressourcenengpässen VMs in ein DC eines föderierten Partners migrieren zu können (vgl. [36]). Der Benutzer bekommt davon nichts mit. Cloud-Föderationen stecken zurzeit in der freien Wirtschaft noch in den Kinderschuhen. Dies liegt vor allem daran, dass die großen Cloud-Anbieter meist kein Interesse an Föderationen haben. Sie wollen lieber alle Dienste selbst erbringen und expandieren deswegen weiter. Cloud-Föderationen werden eher den Bereich der kleinen und mittleren CSPs revolutionieren, da diese durch einen Zusammenschluss im Grunde nur Vorteile haben: Sie kön-

nen ihre Ressourcen flexibel verwalten und mehr Dienste anbieten. Bei Ressourcenengpässen können sie schnell und flexibel auf Ressourcen der Partner zurückgreifen. Außerdem kann durch ein optimiertes Ressourcenmanagement die Auslastung der lokalen Ressourcen und damit der Gewinn gesteigert werden. Ein Beispiel für den Zusammenschluss von zwei großen CSPs ist die Ende 2016 bekanntgegebene Kooperation von Amazon und VMware (vgl. [159]). Beide Unternehmen bieten ihren Kunden weltweit verteilte Cloud-DCs zur Ausführung von VMs an. Kunden beider CSPs können jetzt VMs zwischen ihren lokalen DCs und den DCs der beiden CSPs vollautomatisch hin und her migrieren. Dies ist die erste große wirtschaftlich betriebene Cloud-Föderation.

In Abbildung 1.1 wird ein Föderationsszenario grafisch dargestellt. Es zeigt mehrere CSPs mit einem DC (CSP 1) oder mehreren DCs (CSP 2) und den darin vorhandenen Servern. Weiterhin sind hier bereits die drei Scheduling Ebenen dar gestellt (Intra-DC-Scheduler, Inter-DC-Scheduler, Inter-CSP-Scheduler), die im weiteren Verlauf näher vorgestellt werden. Zusätzlich ist ein Kunde (engl. Customer) abgebildet, der eine Dienstanforderung (engl. Service Request) an einen virtuellen CSP (vCSP) schickt. Virtuelle CSPs werden ebenfalls später weiter erläutert (vgl. Kapitel 7.12).

1.2 Service Level Agreements

Eine wichtige Voraussetzung für eine nachvollziehbare Dienstleistung ist ein gültiger Vertrag zwischen Kunde und CSP. Dieser Vertrag regelt die funktionalen und nicht-funktionalen Eigenschaften, die ein CSP seinem Kunden während der Ausführung eines Dienstes zusichert. Solch ein Vertrag wird als Service Level Agreement (SLA) bezeichnet. Über einen SLA können zum Beispiel bestimmte Ressourcenanforderungen, Durchsatzgeschwindigkeiten, Verfügbarkeiten, Ausführungsländer oder Wiederherstellungszeiten geregelt werden (vgl. [166]). Jeder dieser einzelnen Punkte wird als Service Level Objective (SLO) bezeichnet. Zur Laufzeit eines Dienstes müssen die vereinbarten SLOs permanent überwacht werden. Dafür ist ein feingranulares Monitoring

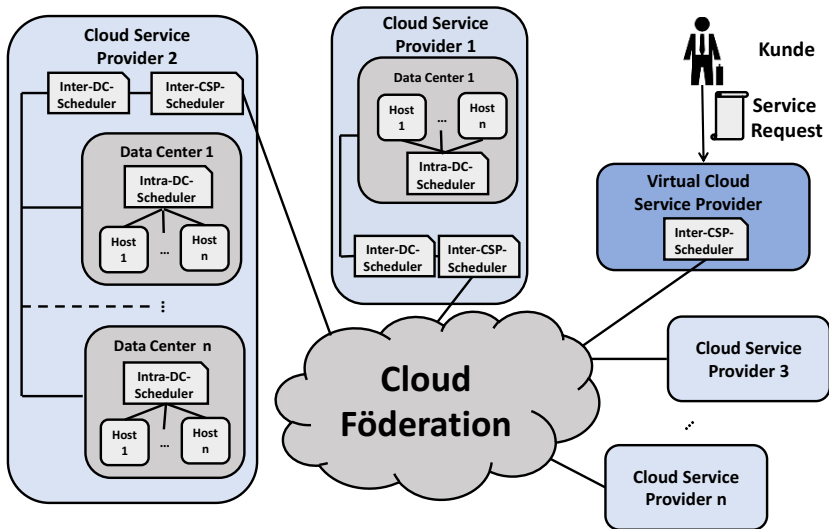


Abbildung 1.1: Beispielhafter Aufbau einer Cloud-Föderation (in Anlehnung an [92])

und Reporting notwendig. Der Kunde muss zu jeder Zeit einsehen können, ob der vereinbarte SLA noch eingehalten wird, oder ob ein oder gar mehrere SLOs gebrochen wurden. Sollte dies der Fall sein, so muss der CSP eine vorher ebenfalls im SLA definierte Strafe wegen des Vertragsbruchs an den Kunden zahlen. Diese Strafzahlung wird auch als Pönale bezeichnet. In Abbildung 1.2 wird die Zusammensetzung und der Lebenszyklus einer SLA grafisch dargestellt.

1.3 Cloud Scheduling

Die Verwaltung der VMs ist ein komplexer Prozess, der auf verschiedensten Ebenen abläuft. Vor allem die Platzierung der einzelnen VMs auf die physikalischen Server innerhalb eines gegebenen DCs stellt eine große Herausforderung dar. Hier spielen unterschiedlichste Faktoren mit. So sollen zum einen die lokalen Ressourcen optimal

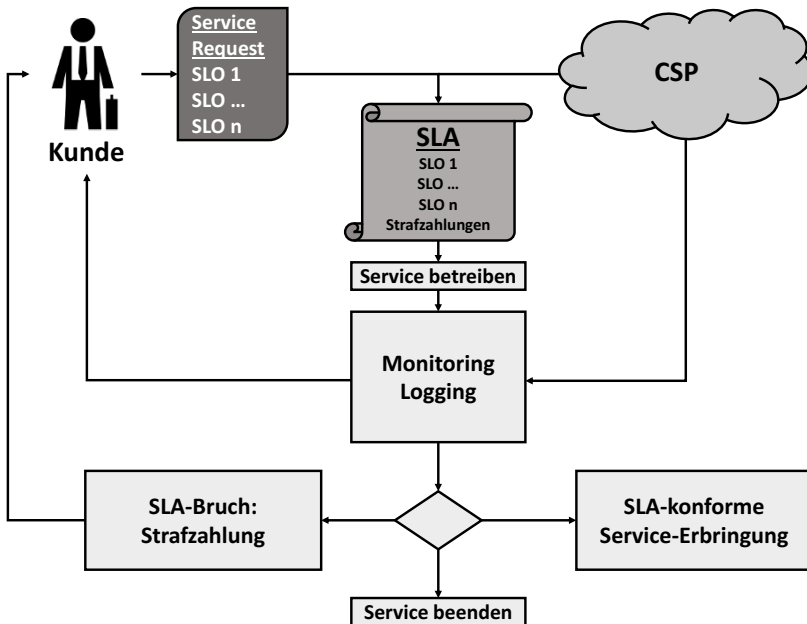


Abbildung 1.2: Grafische Darstellung des SLA-Ablaufs

ausgenutzt werden. Das bedeutet, dass die Server mit möglichst vielen VMs bestückt werden sollen, um den Gewinn zu maximieren. Zum anderen sollen die SLAs eingehalten werden, da sonst zum einen Strafzahlungen anfallen und zum anderen die Kundenzufriedenheit sinkt. Zusätzlich können noch viele weitere Ziele mit einbezogen werden: Zum Beispiel eine gleichmäßige Ressourcenauslastung, ein möglichst niedriger Energieverbrauch oder geringe Latenzen zum Kunden. Um all dies zu erreichen, müssen die VMs initial (also bei ihrer Erstellung) und zur Laufzeit möglichst optimal auf die vorhandenen Ressourcen verteilt werden. Dieser Vorgang wird als Scheduling bezeichnet. Jeder CSP kann dabei unterschiedliche Ziele innerhalb seiner DCs verfolgen und somit unterschiedliche Scheduling-Strategien einsetzen.

Im Bereich der Cloud-Föderationen verteilt sich das Scheduling auf insgesamt drei Ebenen. Auf der ersten Ebene (Intra-DC-Ebene)

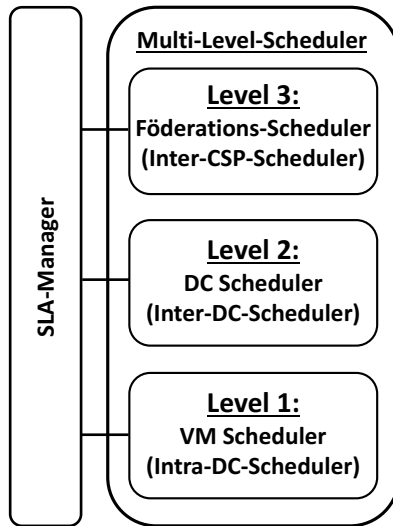


Abbildung 1.3: Der Multi-Level-Scheduler (in Anlehnung an [116])

muss anhand einer gegebenen Scheduling-Strategie festgelegt werden, welche VMs auf welchen Servern ausgeführt werden. Dabei kann es zur Laufzeit zu Ressourcenengpässen auf einzelnen Servern kommen. In diesem Fall muss der lokale Scheduler anhand seiner Strategie entscheiden, welche VM(s) auf andere Server migriert werden müssen, um wieder einen ausgeglichenen Ressourcenzustand herbeiführen zu können. Wenn der Scheduler das Problem durch eine oder mehrere lokale VM-Migrationen lösen kann, wird der Betrieb normal fortgesetzt. Kann der Scheduler das Problem nicht lokal lösen und besitzt der CSP keine weiteren DCs, so kommt es zu einem oder mehreren SLA-Brüchen. Besitzt der CSP aber noch weitere DCs, so kann auf der zweiten Scheduling-Ebene (Inter-DC-Ebene) ein DC gesucht werden, welches die zu migrierenden VMs aufnehmen kann. Auch auf dieser Ebene kann es wieder unterschiedlichste Strategien geben. In einer Cloud-Föderation kommt noch eine dritte Ebene (Inter-CSP-Ebene) hinzu, da sich verschiedene CSPs gegenseitig Ressourcen zur Verfü-