

# HANSER



## Inhaltsverzeichnis

Stefan Edlich, Achim Friedland, Jens Hampe, Benjamin Brauer, Markus  
Brückner

NoSQL

Einstieg in die Welt nichtrelationaler Web 2.0 Datenbanken

ISBN: 978-3-446-42753-2

Weitere Informationen oder Bestellungen unter

<http://www.hanser.de/978-3-446-42753-2>

sowie im Buchhandel.

# Inhalt

<b>Geleitwort</b> .....	<b>XIII</b>
<b>Vorwort</b> .....	<b>XV</b>
Vorwort zur 2. Auflage .....	XVII
<b>1 Einführung</b> .....	<b>1</b>
1.1 Historie .....	1
1.2 Definition und Diskussion .....	2
1.3 Kategorisierung von NoSQL-Systemen .....	5
1.3.1 Key/Value-Systeme .....	7
1.3.2 Column-Family-Systeme .....	7
1.3.3 Document Stores .....	8
1.3.4 Graphdatenbanken .....	8
1.4 Fazit.....	9
<b>2 NoSQL – Theoretische Grundlagen</b> .....	<b>11</b>
2.1 Map/Reduce.....	12
2.1.1 Funktionale Ursprünge.....	12
2.1.2 Phasen und Datenfluss .....	17
2.1.3 Komponenten und Architektur.....	19
2.1.4 Anwendungsbereiche und Implementierungen .....	22
2.1.5 Praktisches Beispiel .....	24
2.1.6 Zusammenfassung.....	28
2.2 CAP und Eventually Consistent .....	30
2.2.1 Konsistenzmodell relationaler Datenbanken .....	30
2.2.2 CAP-Theorem.....	31
2.2.3 Alternatives Konsistenzmodell: BASE .....	33
2.3 Consistent-Hashing .....	36
2.4 Multiversion Concurrency Control.....	40
2.5 Vector Clocks .....	43
2.6 Paxos.....	47
2.7 REST.....	51
2.7.1 Web-scale?.....	51

2.7.2	Bausteine .....	52
2.7.3	Entwurf von REST-Applikationen .....	56
2.7.4	Skalierung von REST-Systemen .....	59
2.7.5	Fazit .....	61
<b>3</b>	<b>Wide Column Stores .....</b>	<b>63</b>
3.1	HBase .....	64
3.1.1	Überblick .....	64
3.1.2	Allgemeines .....	65
3.1.3	Datenmodell .....	65
3.1.4	Installation .....	67
3.1.5	CRUD-Operationen .....	69
3.1.6	Fortgeschrittene Abfragen mit Map/Reduce .....	74
3.1.7	Skalierung und Konfiguration .....	77
3.1.8	Replikation .....	79
3.1.9	Bewertung .....	80
3.2	Cassandra .....	82
3.2.1	Allgemeines .....	82
3.2.2	Installation .....	83
3.2.3	Datenmodell .....	84
3.2.4	CRUD-Operationen .....	88
3.2.5	Abfragen in Cassandra .....	91
3.2.6	Replikation und Skalierung .....	93
3.2.7	Bewertung .....	94
3.3	Amazon SimpleDB .....	96
3.3.1	Allgemeines .....	97
3.3.2	Datenmodell .....	97
3.3.3	Datensicherheit und Datenschutz .....	98
3.3.4	Installation .....	99
3.3.5	CRUD-Operationen .....	100
3.3.6	Zugriff mit dem AWS Toolkit für Eclipse .....	111
3.3.7	Replikation und Skalierung .....	114
3.3.8	Bewertung .....	114
<b>4</b>	<b>Document Stores .....</b>	<b>117</b>
4.1	CouchDB .....	118
4.1.1	Überblick .....	118
4.1.2	Beschreibung .....	118
4.1.3	Datenmodell .....	119
4.1.4	View-Modell .....	120
4.1.5	Zugriffskontrolle .....	120
4.1.6	Installation .....	121

4.1.7	CRUD-Operationen.....	123
4.1.8	Erstellen von Views .....	127
4.1.9	Replikation .....	128
4.1.10	Skalierung.....	129
4.1.11	CouchApps .....	129
4.1.12	Bewertung .....	130
4.2	MongoDB.....	131
4.2.1	Überblick.....	131
4.2.2	Datenmodell.....	133
4.2.3	Installation .....	134
4.2.4	CRUD-Operationen.....	135
4.2.5	Fortgeschrittene Abfragen und Map/Reduce.....	140
4.2.6	Skalierung.....	143
4.2.7	Replikation .....	145
4.2.8	Bewertung .....	148
<b>5</b>	<b>Key/Value-Datenbanken .....</b>	<b>151</b>
5.1	Redis .....	152
5.1.1	Überblick.....	152
5.1.2	Allgemeines .....	152
5.1.3	Installation .....	153
5.1.4	CRUD-Operationen.....	154
5.1.5	Redis Hashes.....	160
5.1.6	Redis Messaging .....	161
5.1.7	Zugriff aus anderen Programmiersprachen.....	162
5.1.8	Replikation und Konfiguration .....	164
5.1.9	Skalierung.....	166
5.1.10	Bewertung .....	166
5.2	Chordless .....	168
5.2.1	Überblick.....	168
5.2.2	Allgemeines .....	169
5.2.3	Installation .....	170
5.2.4	CRUD-Operationen.....	171
5.2.5	Fortgeschrittene Abfragen.....	173
5.2.6	Skalierung, Replikation und Konfiguration.....	176
5.2.7	Bewertung .....	177
5.3	Riak.....	179
5.3.1	Überblick.....	179
5.3.2	Allgemeines .....	179
5.3.3	Installation .....	182
5.3.4	CRUD-Operationen.....	182
5.3.5	Zugriff aus anderen Programmiersprachen.....	184

5.3.6	Abfragen und Links in Riak .....	187
5.3.7	Riak Search .....	189
5.3.8	Skalierung, Replikation und Konfiguration.....	190
5.3.9	Bewertung .....	191
5.4	Membase .....	193
5.4.1	Überblick .....	193
5.4.2	Allgemeines .....	193
5.4.3	Installation .....	194
5.4.4	Architektur und Konzepte .....	196
5.4.5	Das Memcached-Protokoll .....	199
5.4.6	Zugriff aus anderen Programmiersprachen.....	199
5.4.7	Skalierung und Replikation .....	201
5.4.8	Der Moxi .....	202
5.4.9	Die TAP-Schnittstelle.....	203
5.4.10	Bewertung .....	204
<b>6</b>	<b>Graphdatenbanken .....</b>	<b>207</b>
6.1	Mathematische und technische Grundlagen .....	209
6.1.1	Das allgemeine Graphenmodell.....	209
6.1.2	Das Property-Graph-Modell.....	210
6.1.3	Relationen höherer Ordnung .....	213
6.1.4	Repräsentation von Graphen .....	217
6.1.5	Traversierung von Graphen .....	219
6.1.6	Indizierung von Graphdaten.....	221
6.1.7	Skalierung mittels Replikation und Partitionierung .....	222
6.1.8	Graph Query Languages .....	225
6.1.9	Vergleich mit anderen Datenmodellen.....	225
6.1.10	Zusammenfassung.....	228
6.2	Der Tinkerpop Graph Processing Stack.....	230
6.2.1	Blueprints.....	231
6.2.2	Blueprints.NET .....	234
6.2.3	Frames .....	236
6.2.4	Pipes und Pipes.NET .....	237
6.2.5	Gremlin.....	240
6.2.6	Pacer.....	244
6.2.7	Rexster .....	244
6.3	AllegroGraph.....	248
6.3.1	Datenmodell .....	249
6.3.2	Installation .....	250
6.3.3	CRUD-Operationen mit Java .....	251
6.3.4	RDF-Reasoning und Konsistenzkriterien.....	257
6.3.5	Bewertung .....	259

6.4	DEX .....	260
6.4.1	Beschreibung .....	261
6.4.2	Datenmodell .....	262
6.4.3	Installation .....	262
6.4.4	CRUD-Operationen.....	262
6.4.5	Graphoperationen.....	264
6.4.6	DEX-Skripting.....	265
6.4.7	DEX-Shell.....	267
6.4.8	Bewertung .....	268
6.5	HyperGraphDB.....	269
6.5.1	Datenmodell .....	269
6.5.2	Installation .....	270
6.5.3	CRUD-Operationen.....	271
6.5.4	Graphoperationen.....	274
6.5.5	Peer-To-Peer-Framework .....	274
6.5.6	Bewertung .....	274
6.6	InfiniteGraph .....	276
6.6.1	Beschreibung .....	276
6.6.2	Datenmodell .....	277
6.6.3	Installation .....	278
6.6.4	CRUD-Operationen.....	278
6.6.5	Graphoperationen.....	282
6.6.6	Bewertung .....	283
6.7	InfoGrid .....	284
6.7.1	Beschreibung .....	285
6.7.2	Datenmodell .....	285
6.7.3	Installation .....	286
6.7.4	CRUD-Operationen.....	286
6.7.5	Models.....	289
6.7.6	Bewertung .....	290
6.8	Neo4j.....	290
6.8.1	Installation .....	291
6.8.2	CRUD-Operationen mit der Java-API.....	292
6.8.3	CRUD-Operationen mit JRuby.....	296
6.8.4	HTTP/REST-Schnittstelle .....	297
6.8.5	Cypher .....	300
6.8.6	Replikation und Skalierung .....	301
6.8.7	Bewertung .....	301
6.9	sones GraphDB.....	302
6.9.1	Datenmodell .....	303
6.9.2	Installation .....	304
6.9.3	CRUD-Operationen.....	304
6.9.4	Bewertung .....	308

6.10	Weitere graphorientierte Ansätze.....	309
6.10.1	Twitters FlockDB.....	309
6.10.2	Google Pregel.....	311
6.10.3	Apache Hama.....	315
6.10.4	Microsoft Trinity .....	317
6.10.5	Die VertexDB-Familie .....	321
6.10.6	Filament.....	324
<b>7</b>	<b>OrientDB.....</b>	<b>329</b>
7.1	Überblick.....	329
7.2	Allgemeines .....	330
7.3	Datenmodell.....	330
7.4	Struktur der Datenablage.....	332
7.5	Datenbanksicherheit .....	332
7.6	Installation.....	333
7.7	CRUD-Operationen DocumentDB.....	336
7.8	CRUD-Operationen ObjectDB.....	340
7.9	CRUD-Operationen GraphDB.....	343
7.10	Datenbankschemata.....	347
7.11	Transaktionen .....	347
7.12	HTTP-REST-Schnittstelle.....	348
7.13	Replizierung .....	351
7.14	Hosting über NuvolaBase .....	353
7.15	Bewertung.....	353
<b>8</b>	<b>Weitere NoSQL-Datenbanken .....</b>	<b>355</b>
8.1	Wide Column Stores.....	356
8.1.1	Hypertable.....	356
8.1.2	Cloudera .....	357
8.2	Document Stores.....	358
8.3	Key/Value/Tupel-Stores.....	359
8.3.1	Amazon Dynamo .....	359
8.3.2	Dynomite und KAI .....	360
8.3.3	Voldemort .....	360
8.3.4	Scalaris .....	363
8.3.5	Die Tokyo-Produktfamilie .....	364
8.3.6	Weitere Key/Value-Systeme .....	366
8.4	Google App Engine Storage.....	367
8.5	Weitere ‚Soft‘-NoSQL-Lösungen.....	368

<b>9</b>	<b>Orientierung im Datenbankraum .....</b>	<b>371</b>
9.1	Grundlegende Gedanken.....	372
9.2	Datenanalyse .....	373
9.3	Transaktionsmodell.....	375
9.4	Performancesapekte.....	376
9.5	Abfrageanforderungen.....	377
9.6	Architektur .....	378
9.7	Weitere nicht-funktionale Anforderungen .....	379
9.8	Anwendungsfälle für NoSQL-Datenbanken .....	381
9.9	Fazit.....	382
	<b>Register .....</b>	<b>385</b>