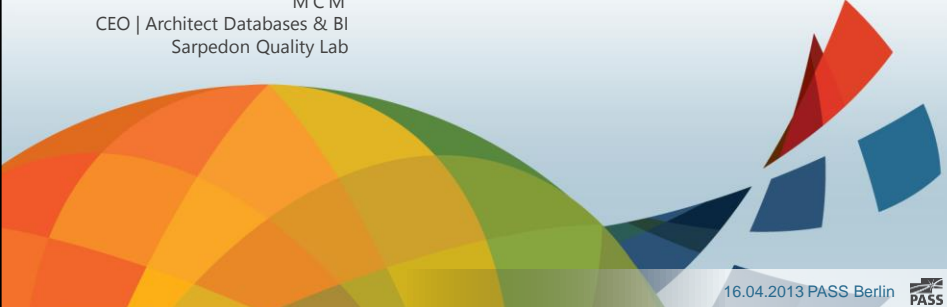



From Locks to Deadlocks - Concurrency in SQL Server


Concurrency mit Fokus auf Lese-Operationen

Andreas Wolter
MCM
CEO | Architect Databases & BI
Sarpedon Quality Lab



16.04.2013 PASS Berlin 

Andreas Wolter



Microsoft CERTIFIED Master

- Consultant, Trainer & Speaker
Microsoft Certified Master SQL Server (MCM)
 - SQL Server
 - Business Intelligence & Datawarehousing
 - Performance Tuning
 - Security
- PASS-Chapter Leader Köln/Bonn/Düsseldorf

Kontakt: andreas.wolter@SarpedonQualityLab.com
Blog: www.vb-magazin.de/forums/blogs/andreaswolter
Web: www.SarpedonQualityLab.com
XING: www.xing.com/profile/Andreas_Wolter2
Linkedin: www.linkedin.com/in/andreaswolter
Twitter: @AndreasWolter

Microsoft CERTIFIED Master	SQL Server® 2008	Microsoft CERTIFIED IT Professional	Database Developer 2008 Database Administrator 2008 Business Intelligence Developer 2008	Microsoft CERTIFIED Trainer	Database Development Database Administration Systems Administration
-----------------------------------	------------------	--	--	------------------------------------	---

PASS

Publikums-check

(mehrfach-Meldungen möglich)

(Prod) SQL Server Version?

- 2000 - 2005
- 2008 (☺)
- 2008 R2 (☺ ☺)
- 2012 (☺ ☺ ☺)

- SQL-Entwickler
- Anwendungsentwickler
- Administrator



Abstract

In dieser Session geht es um essentielle Mechanismen im SQL Server, die sowohl Administrator als auch Entwickler beeinflussen und kennen sollten: Warum muss SQL Server Objekte sperren, welche Auswirkungen kann das haben (Performance und Datenintegrität) und wie wir darauf Einfluss nehmen können.

Das Ganze natürlich wie immer reich an Demos – zieht Euch warm an ;-)



Themen

- Wann und was passiert bei Lock-Escalation?
- Was passiert bei zu vielen Sperren auf ein Object? Was ist zu viel?
- Können Indexe Blockaden verhindern?
- Fehler Nummer 1205. - Wie und wobei Deadlocks auftreten können und wie man sie beheben kann.
- Warum NOLOCK bei geschäftskritischen Daten keine gute Idee ist.
- Snapshot Isolation Transaction Level - und welche Auswirkungen hat das auf meine Datenstrukturen?
- Was hat Table-Design mit Sperren zu tun? (Stichwort "Large Databases")

16.04.2013 PASS Berlin



Demo

Locking
& Blocking

16.04.2013 PASS Berlin



Concurrency Probleme, die durch Locking verhindert werden

- Lost updates
- Uncommittete Abhängigkeiten (dirty read)
- Inconsistent analysis (non-repeatable read)
- Phantom reads
- Missing und double reads

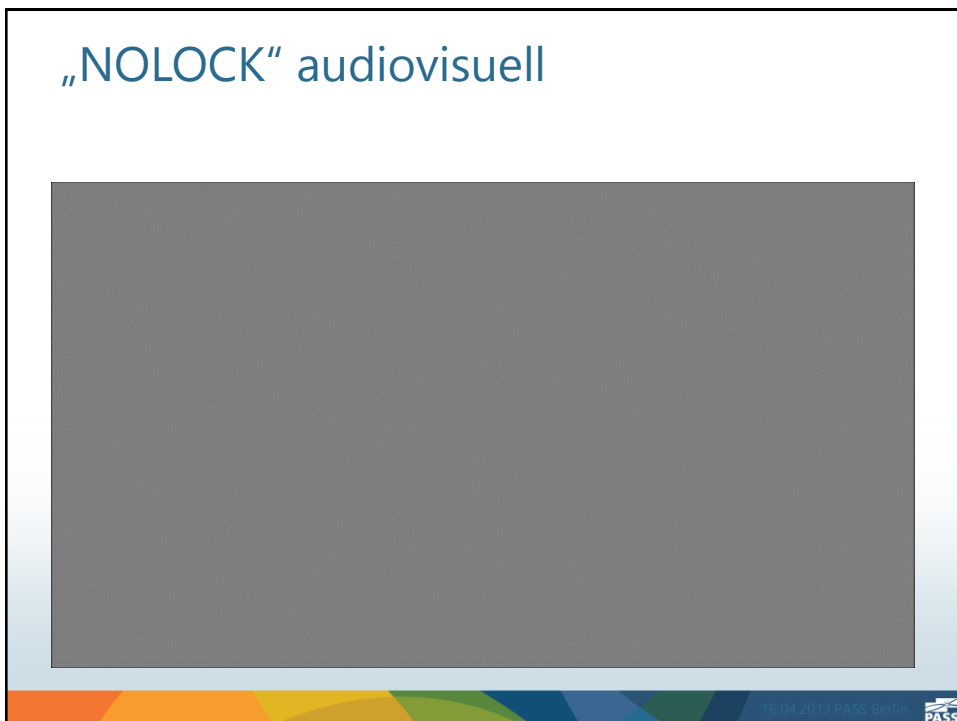


Lock Compatibility Matrix

	Existing granted mode					
Requested mode	IS	S	U	IX	SIX	X
Intent shared (IS)	Yes	Yes	Yes	Yes	Yes	No
Shared (S)	Yes	Yes	Yes	No	No	No
Update (U)	Yes	Yes	No	No	No	No
Intent exclusive (IX)	Yes	No	No	Yes	No	No
Shared with intent exclusive (SIX)	Yes	No	No	No	No	No
Exclusive (X)	No	No	No	No	No	No

16.04.2013 PASS Berlin





Möglichkeiten, Sperren zu umgehen

- NOLOCK Table hint

- Gleichbedeutend mit READ UNCOMMITTED
- Uncommittete DatenProbleme:
 - Falsche oder fehlende Updates
 - Fehlende Daten
 - Doppelte Daten
- Auf Fehler 601 prüfen

Could not continue scan with NOLOCK due to data movement.
Scan mit NOLOCK konnte aufgrund einer Datenverschiebung nicht fortgesetzt werden.

- Timeout

- standardmäßig setzt SQL Server KEIN Timeout
- Auf Fehler 1222 prüfen
- Kein Auto-Rollback!

- Readpast

- Gesperrte Datensätze werden „übersprungen“
- Am besten geeignet für Queues

16.04.2013 PASS Frankfurt



NOLOCK erzeugt
beachtet keine
Sperren

Tatsächlich??

16.04.2013 PASS Berlin



Lock Escalation

- Tritt ein, wenn ein Statement(!) mehr als 5000 Locks auf ein Objekt hält.
 - Wenn das nicht geht, wird alle 1250 Neue Sperren wiederholt versucht.
- Maximal 60% Memory werden für Locks verwendet.
- Bei 40 % Memory usage wird ebenfalls eskaliert. (steuerbar mit der "locks" Configuration, aber deprecated)
- Ebenfalls mit Traceflags beeinflussbar

Eskalation geschieht immer von

- Row/Key oder Page -> Table oder von Row/Key/Page -> Partition
- Nicht von Key -> Page oder von Partition -> Table – Das ist ein Mythos!

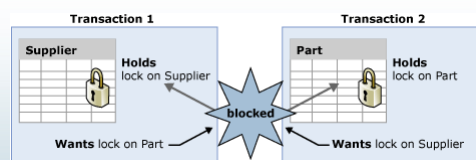
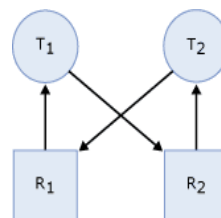
10.04.2013 PASS Berlin



Deadlocks

Varianten:

- cycle deadlock
- conversion deadlock



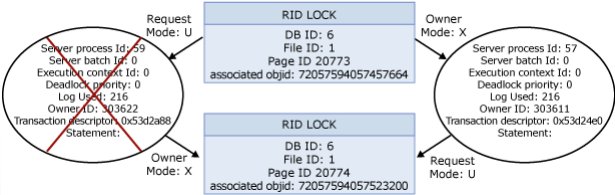
10.04.2013 PASS Berlin



Was beim deadlock passiert



Und so sieht das in Profiler/Xevents aus



Deadlock Monitor

- Der Standardintervall ist 5 Sekunden.
- Wenn der lock monitor thread einen deadlock findet, fällt der detection interval auf bis 100 milliseconds, abhängig von den auftretenden deadlocks.
- Wenn ein deadlock entdeckt wurde, wird beim nächsten darauf folgenden lock wait eine weitere deadlock detection durchgeführt.

16.04.2013 PASS Berlin



Deadlock Analyse

- Bis SQL 2005 mit Traceflag 1222 bzw 1204/1205
- Seit SQL Server 2008 über die Extended Event session „system-health“
(siehe auch meine Vorträge „*Hasta la vista, Profiler*“ – *Tracing mit Extended Events* in den RGs Hamburg und Köln-Bonn-Düsseldorf sowie SQLSaturdays in München 2012, Holland 2013, Denmark 2013)

16.04.2013 PASS Berlin



Traceflag 1222 Dump in Errorlog

```

File Edit Format View Help
2010-09-17 10:04:25.99 spid225 deadlock-list
2010-09-17 10:04:25.99 spid225 deadlock victim=process54f390
2010-09-17 10:04:25.99 spid225 process=1157
2010-09-17 10:04:25.99 spid225 process id=process54f390 taskpriority=0
2010-09-17 10:04:25.99 spid225 logused=632 waitresource=KEY: 7:72057594043432960 (020068e8b274) waittime=418
2010-09-17 10:04:25.99 spid225 ownerid=5739915 transactionname=user_transaction lasttranstarted=2010-09-
17T10:04:15.477 XDES=0x53fced0 lockmode=U schedulerid=2 kpid=11976 status=suspended
2010-09-17 10:04:25.99 spid225 spid=54 sbid=0 ecid=0 priority=0 tranount=2 lastbatchstarted=2010-09-
17T10:04:15.457 lastbatchcompleted=2010-09-17T09:59:00.830 clientapp=Microsoft SQL
Server Management Studio - query hostname=KEN-PC hostpid=14948 loginname=KEN-PC\KEN
isolationlevel=read committed (2) xactid=5739915 currentdb=1 locktimeout=4294967295
clientoption=671090784 clientoption2=390200
2010-09-17 10:04:25.99 spid225 executionstack
2010-09-17 10:04:25.99 spid225 frame procname=adhoc line=9 stmtstart=58
sqlhandle=0x020000004bd8112b27af24e75bfff654951e599e6466d02f0
2010-09-17 10:04:25.99 spid225 UPDATE [Adventureworks].[Person].[Address] set
[AddressLine1] = ([AddressLine1])+@1 WHERE [AddressID]=@2
2010-09-17 10:04:25.99 spid225 frame procname=adhoc line=9 stmtstart=302
sqlhandle=0x0200000094172711570c69fb5c87ecddcac21dabd16cf8
2010-09-17 10:04:25.99 spid225 UPDATE Adventureworks.Person.Address
SET AddressLine1 = AddressLine1 + ' b '
2010-09-17 10:04:25.99 spid225 WHERE AddressID = 2
2010-09-17 10:04:25.99 spid225 inputbuf
2010-09-17 10:04:25.99 spid225 BEGIN TRANSACTION
2010-09-17 10:04:25.99 spid225 UPDATE Adventureworks.Person.Address
SET AddressLine1 = AddressLine1 + ' a '
2010-09-17 10:04:25.99 spid225 WHERE AddressID = 1
2010-09-17 10:04:25.99 spid225 WAITFOR DELAY '00:00:10'
2010-09-17 10:04:25.99 spid225 UPDATE Adventureworks.Person.Address
SET AddressLine1 = AddressLine1 + ' b '
2010-09-17 10:04:25.99 spid225 WHERE AddressID = 2
2010-09-17 10:04:26.00 spid225 process id=process54f40 taskpriority=0
logused=632 waitresource=KEY: 7:72057594043432960 (010086470766) waittime=2157
ownerid=5739988 transactionname=user_transaction lasttranstarted=2010-09-
17T10:04:18.740 XDES=0x13ebcc10 lockmode=U schedulerid=2 kpid=8176 status=suspended
2010-09-17 10:04:26.00 spid225 spid=55 sbid=0 ecid=0 priority=0 tranount=2 lastbatchstarted=2010-09-
17T10:04:18.737 lastbatchcompleted=2010-09-17T09:59:11.057 clientapp=Microsoft SQL
Server Management Studio - query hostname=KEN-PC hostpid=14948 loginname=KEN-PC\KEN
isolationlevel=read committed (2) xactid=5739988 currentdb=1 locktimeout=4294967295
clientoption=671090784 clientoption2=390200
2010-09-17 10:04:26.00 spid225 executionstack
2010-09-17 10:04:26.00 spid225 frame procname=adhoc line=9 stmtstart=58
sqlhandle=0x020000004bd8112b27af24e75bfff654951e599e6466d02f0
2010-09-17 10:04:26.00 spid225 UPDATE [Adventureworks].[Person].[Address] set
[AddressLine1] = ([AddressLine1])+@1 WHERE [AddressID]=@2
2010-09-17 10:04:26.00 spid225 frame procname=adhoc line=9 stmtstart=302
sqlhandle=0x020000002433a2df2a7689cc0dad1e6d99274077605917
2010-09-17 10:04:26.00 spid225 UPDATE Adventureworks.Person.Address
SET AddressLine1 = AddressLine1 + ' b '
2010-09-17 10:04:26.00 spid225 WHERE AddressID = 1
2010-09-17 10:04:26.00 spid225 inputbuf

```

Deadlock-Vermeidung/Minimierung

Empfohlene Reihenfolge (je später desto kritischer)

- (1) Isolation Level beachten/planen
- (2) Einheitliche Reihenfolge beim Objektzugriff in Dev-Richtlinien
- (3) Covering Indexes
- (4) Frühzeitiges Sperren (schon beim Select)
- (5) Isolation Level wechseln

Wenn das nicht langt:

- (1) Präferierte Opfer/Sieger festlegen (DEADLOCK_PRIORITY)

In komplexen Umgebungen mit diversen Zugriffswegen lassen sich Deadlocks nicht garantiert vermeiden!

Deswegen, liebe Entwickler: defensiv entwickeln, nach Fehlern schauen.

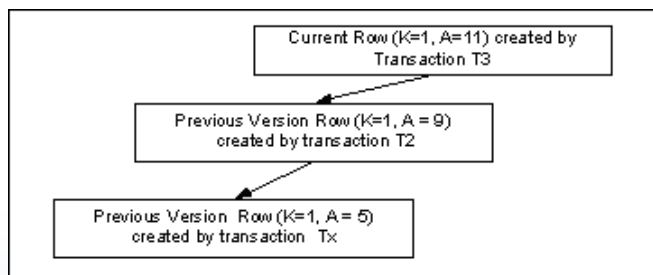
Snapshot isolation und read committed snapshot isolation (RCSI)

- "readers do not block writers, and writers do not block readers" (Leser blockieren keine Schreiber und Schreiber blockieren keine Leser)
- Basieren auf *row versioning*
 - version store in Tempdb
- SI vermeidet
 - Dirty reads
 - non-repeatable reads
 - phantom reads
 - , ohne Locks zu benötigen: Leser erhalten einen Snapshot der Daten, wie er zu dem Startzeitpunkt der Transaktion vorlag
- RCSI vermeidet:
 - Dirty reads
- Jedoch nicht
 - non-repeatable reads oder phantoms

10.04.2013 PASS Berlin



Version store



Die version store (unter SI und RCSI) wird immer gepflegt – egal ob es aktive Leser gibt, oder nicht!

10.04.2013 PASS Berlin



RCSI

- RCSI verwendet weniger tempdb space als SI.
- RCSI funktioniert für distributed transactions; SI nicht.
- RCSI führt nicht zu update Konflikten.
- RCSI erfordert keine Änderung der Applikation

10.04.2013 PASS Berlin



Aber Achtung

SELECTS können durch lange version-chains verlangsamt werden
Tempdb wird stark beansprucht
bei allen UPDATE, DELETE, und einigen Insert Statements

Row versioning verbreitert jede row um 14 bytes.

Bei Konflikten kann es zu Rollback von UPDATE(!)-Operationen kommen

Wenn in Tempdb kein Platz mehr ist, schlagen Select-statements (nicht die Updates) fehl



Fazit und weiterführende Gedanken

Entwickler sollten ein gutes Verständnis für das Verarbeiten von Transaktionen durch SQL Server haben

Unterschied zwischen Locking und Blocking

Blocking und Deadlocking

Systeme beobachten (möglichst ohne Deadlocks Traceflag 1222) und richtige Schlüsse ziehen

10.04.2013 PASS Berlin



Ressourcen

Lock Compatibility

<http://msdn.microsoft.com/en-us/library/ms186396%28v=sql.105%29.aspx>

Lock Modes

[http://technet.microsoft.com/en-us/library/ms175519\(v=SQL.105\).aspx](http://technet.microsoft.com/en-us/library/ms175519(v=SQL.105).aspx)

Lock Escalation

[http://msdn.microsoft.com/en-us/library/ms184286\(v=SQL.105\).aspx](http://msdn.microsoft.com/en-us/library/ms184286(v=SQL.105).aspx)

Working with tempdb in SQL Server 2005

<http://technet.microsoft.com/en-us/library/cc966545.aspx>

Detecting and Ending Deadlocks

<http://msdn.microsoft.com/en-us/library/ms178104.aspx>



From Locks to Dead-locks - Concurrency in SQL Server



Mail: andreas.wolter@SarpedonQualityLab.com
Blog: www.vb-magazin.de/forums/blogs/andreaswolter
Web: www.SarpedonQualityLab.com
XING: www.xing.com/profile/Andreas_Wolter2
Linkedin: www.linkedin.com/in/andreaswolter
Twitter: @AndreasWolter



SQL Server Master-Class Seminare 2013

- ❖ Kein (PASS) Essentials oder MOC-Kurs
- ❖ Tiefe Einblicke in SQL Server Internas
- ❖ Real world Szenarien
- ❖ Auch ideal zur Vorbereitung für den Microsoft Certified Master für SQL Server -> das einzige Trainingsangebot dieser Art in Deutschland

Für Alle,
die es genau wissen wollen



Limitierte Teilnehmerzahl

Die neuesten Informationen sofort aus erster Hand erhalten:
info@SarpedonQualityLab.com

Exklusiver 10% Rabatt-Code: „PASS2013“



SQL Server Master-Class Seminare 2013

bitly.com/WQe3Ev

- ❖ (XE1) **Workshop Tracing mit Extended Events in SQL Server** (1 Tag)
Frankfurt a.M. 17.5.2013 (keine Plätze mehr frei) **Zusatztermin: 13.6.2013**
- ❖ (XE2) **Workshop Fortgeschrittene Techniken für Tracing mit mit Extended Events in SQL Server** (1 Tag)
Frankfurt a.M. 18.5.2013, **Zusatztermin: 14.6.2013**
- ❖ (IX) **Indexe, Statistiken und Partitionierung** (2 Tage)
Frankfurt a.M. 11.6. - 12.6.2013
- ❖ ..weitere Themen in Vorbereitung:
 - **Concurrency - Transaktionen, Isolation Level und Sperren** (1 Tag)
 - **Optimierung von Prozeduren und Funktionen** (1 Tag)
 - **Performance und Analysetechniken & -Tools + Workshop** (2 Tage)
 - **Baselining & Benchmarking** (1 Tag)
 - **Sicherheitsworkshops** (Essentials, Vertiefung für Entwickler, Vertiefung für Administratoren, je 1 Tag)
 - **Wiederherstellungsstrategien und Techniken** (1 Tag)
 - **Workshop Hochverfügbarkeit** (2 Tage)
 - **Beyond Relational mit SQL Server - FileStream, FileTable, FullTextSearch, Geospatial** (1 Tag)
 - **Workshop Replikation** (1 Tag)

www.SarpedonQualityLab.com/SQL_Master-Classes.htm

Exklusiver 10% Rabatt-Code: „PASS2013“




Dozent: MCM Andreas Wolter

Super-Early Bird bis 8.5.

Early Bird bis 21.5.

Nicht nur „Best Practices“
sondern
„Was Wie Warum“

Master-Class **Workshop Tracing mit Extended Events** in SQL Server (1 Tag) **Zusatztermin 13.6.2013**

- ❖ Themenauszug:
 - ❖ Architektur und Execution Lifecycle
 - ❖ Extended Events Objekte
(Packages, Events, Predicates, Actions, Targets, Maps)
 - ❖ Arbeiten mit dem Extended Events Addin für SQL 2008
 - ❖ Arbeiten mit der Extended Events GUI SQL 2012
 - ❖ Grundlegende Targets (ring_buffer, event_file,
 - ❖ Auslesen von Event-Daten mit T-SQL und Xquery
 - ❖ Troubleshooting häufiger Szenarien mit Extended Events
(Deadlocks, Query-Waits, historical deadlock analysis, Errors)
- ❖ Weitere Infos: info@SarpedonQualityLab.com

von Anfang an dabei sein
auf die gering invasive Technik setzen

Exklusiver 10% Rabatt-Code: „PASS2013“

Master-Class **Workshop Fortgeschrittene Techniken für Tracing mit Extended Events** in SQL Server (1 Tag)

- ❖ Themenauszug:
 - ❖ Exportieren von Event Daten
 - ❖ Verwenden fortgeschrittener Targets
 - ❖ Arbeiten mit multiplen Targets
 - ❖ Überwachen verschiedener fortgeschrittener Szenarien (Performance, Ghost-cleanup,...)
 - ❖ Arbeiten mit dynamischen Predikaten (Abhängigkeiten zur Laufzeit)
 - ❖ Mapping Extended Events zu SQLTrace
 - ❖ Analyse von Page Splits, Compression, Multi-Deadlock-Victims
- ❖ Weitere Infos: info@SarpedonQualityLab.com

Zusatztermin
14.6.2013

Exklusiver 10% Rabatt-Code: „PASS2013“

Master-Class **Indexe, Statistiken und Partitionierung** (2 Tage)

- ❖ Themenauszug:
 - ❖ Speicherformen von HEAPS und Indizes
 - ❖ Clustered Indexes & Nonclustered Indexes
 - ❖ Statistiken
 - ❖ Datenzugriffsmuster
 - ❖ Selektivität und "Tipping Point"
 - ❖ Covering Indexes
 - ❖ Index-Konsolidierung
 - ❖ Index-Strategien
 - ❖ Partitionierung
 - ❖ Filtered Indexes
 - ❖ Filtered Statistics
 - ❖ Fragmentierung & Defragmentierung von Indexen
 - ❖ Columnstore Indexes
- ❖ Internas in Funktionsweise, Analysemöglichkeiten, Indexstrategien & Design
- ❖ Weitere Infos: info@SarpedonQualityLab.com

11.6. - 12.6.2013

Exklusiver 10% Rabatt-Code: „PASS2013“

Master-Class **Workshop Concurrency** (1 Tag)

- ❖ Themenauszug:
 - ❖ Transaktionen (Grundlagen, „Nested“, Savepoints, Mark)
 - ❖ Isolation Levels
 - ❖ Locking (Granularität, Kompatibilität, Escalation, Blocking)
 - ❖ Konfliktvermeidungsstrategien
 - ❖ Deadlocks
 - ❖ Strategien zur Minimierung von Deadlocks
 - ❖ Snapshot Isolation & Row Versioning
 - ❖ Konfliktvermeidungsstrategien
 - ❖ Latching
 - ❖ Spinlocks
- ❖ Internas in Funktionsweise, Analysetechniken & Lösungswege

❖ Weitere Infos: info@SarpedonQualityLab.com

Exklusiver 10% Rabatt-Code: „PASS2013“

SQL Server Health Check

„Master-Certified“



❖ 3 Standard-Level

❖ **Base Config & Maint.**

- ❖ Windows und SQL Server Konfiguration (FILE-System, Features, CPU, Memory Allocation,...)
- ❖ Check der Wartung
- ❖ High-Level Check Wait-Stats, Plan Cache,...
- ❖ Aufwand: ca. 3-4 Std

390,-*

❖ **Performance Baseline**

- ❖ Base Config & Maint. +
- ❖ Storage layout & Config
- ❖ Indexes, Statistics, Queries
- ❖ Aufwand: ca. 10 Std

990,-*

❖ **Performance Analysis**

- ❖ Performance Baseline +
- ❖ Ausführliche Performance-Analyse+Auswertung
- ❖ Aufwand: ca. 20 Std

2400,-*

- ❖ Weitergehende Analysen je nach System und Wunsch
- ❖ Disaster Recovery & SLA-Compliance-Checks je nach System auf Anfrage
- ❖ Mengenrabatt auf Anfrage.

❖ Weitere Infos: info@SarpedonQualityLab.com



Microsoft
CERTIFIED
Master

Exklusiver 10% Rabatt-Code: „UnlockPASSBerlin2013“

*pro SQL Server Instanz, Preise in EUR zzgl. UmSt.



**SARPEDON**
QUALITY LAB

MICROSOFT® CERTIFIED SINCE 2000

Microsoft
CERTIFIED
Master

Sarpedon Quality Lab: Ihr Spezialist für Datenbanken und Business Intelligence auf Basis von SQL Server Technologien

Seit Juni 2012 sind wir das erste und bislang einzige von Microsoft unabhängige Unternehmen im Deutschsprachigen Raum mit einem Microsoft Certified Master für SQL Server (MCM) an der Spitze des Teams*! [Mehr dazu hier.](#)

Sie suchen Unterstützung bei der Umsetzung Ihrer Ziele?
Sie sind noch unschlüssig, auf welche Technologien Sie setzen sollen?
Fragen Sie uns!
Wir haben beste Erfahrungen mit Produkten des Marktführers für Business Intelligence Produkte: Microsoft.

* Stand April 2013