SQL Server: die Bedeutsamkeit des Datenbanken Kompatibilitätslevels

Kathegorie
SQL Server

Aaron
Priesterroth

Noch vor der Veröffentlichung des SQL Server 2014 wurde dem Datenbanken Kompatibilitätslevel nicht besonders viel Aufmerksamkeit zugeteilt – zumindest wenn es darum ging, die Performanz des Servers zu verbessern.

Im Gegensatz zu dem Datenbanken-Datei-Level (die automatisch angepasst und geupdatet wird, wenn eine Datenbank auf einer Instanz, die eine neuere SQL Server Version ausführt, wiederhergestellt wird), kann das Datenbank Kompatibilitätslevel mit einem einfachen Befehl verändert werden:

```
SET COMPATIBILITY LEVEL = [LEVEL]
```

Dabei ist wichtig zu verstehen, dass das Kompatibilitätslevel einer Datenbank beliebig angepasst werden kann. Was wiederum in vielen Fällen (wie beispielsweise nach einer Migration auf einen neueren SQL Server) dazu führt, dass die Datenbanken Kompatibilitätslevel beibehalten und nicht angepasst werden. Im Allgemeinen sorgt dies erst einmal für keine großen Probleme, erst wenn ein neues Feature auf einer Datenbank benötigt wird, das aktuelle Kompatibilitätslevel dies allerdings verhindert.

Mit SQL Server 2012 und abwärts wurde das Datenbanken Kompatibilitätslevel hauptsächlich für die Regulation der neuen Features einer neueren Version und dem Abschalten veralteter Features genutzt. Zusätzlich wurde darüber eine bessere "backwards application compatibility" mit den älteren SQL Server Versionen gewährleistet. Falls die Zeit für einen vollständigen Regressions-Test nicht vorhanden war, konnte einfach das vorherige Kompatibilitätslevel genutzt werden, bis die benötigte Zeit frei war.

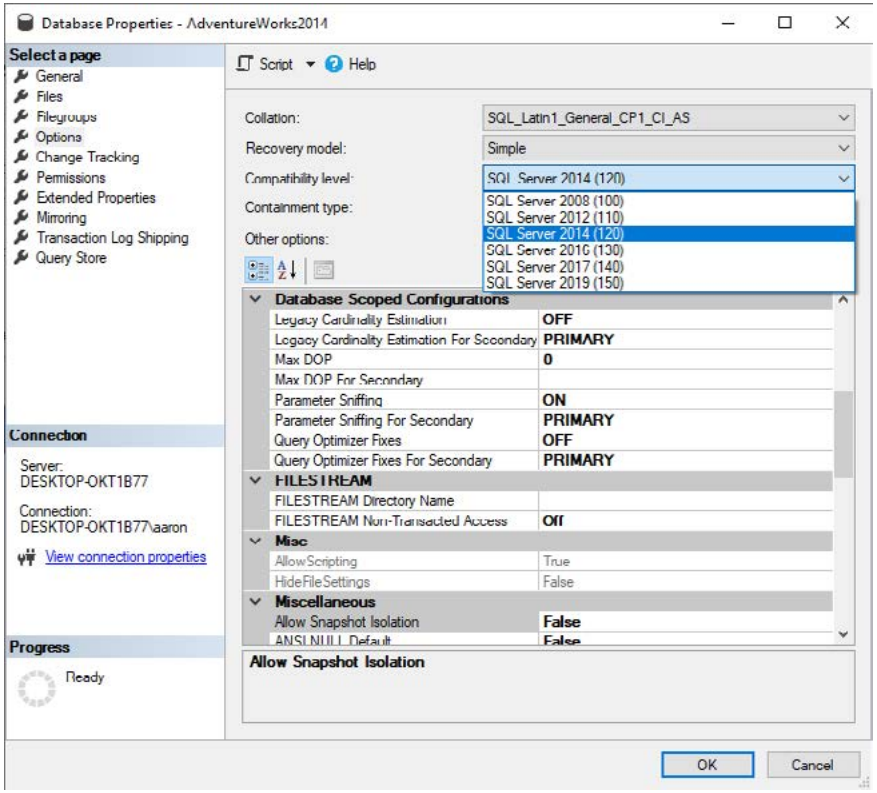
Mainzer Datenfabrik GmbH
Rotterdamer Platz 2, 55120 Mainz
info@madafa.de
madafa.de
+49 6131 3331612

Die folgende Tabelle zeigt die unterschiedlichen SQL Server Versionen mit ihren zugehörigen und unterstützten Kompatibilitätsleveln:

SQL Server Version	Datenbank Engine Version	Kompatibilitätslevel	Unterstützte Kompatibilitätslevel
SQL Server 2019	15	150	150, 140, 130, 120, 110, 100
SQL Server 2017	14	140	140, 130, 120, 110, 100
SQL Server 2016	13	130	130, 120, 110, 100
SQL Server 2014	12	120	120, 110, 100
SQL Server 2012	11	110	110, 100, 90
SQL Server 2008R2	10.5	100	100, 90, 80
SQL Server 2008	10	100	100, 90, 80
SQL Server 2005	9	90	90, 80
SQL Server 2000	8	80	80

Eine neue Datenbank erstellen

Immer wenn eine neu Datenbank erstellt wird, erhält diese automatisch das Kompatibilitätslevel der SQL Server Installation. Beispielsweise bekommt eine Datenbank, die auf SQL Server 2017 erstellt wird, automatisch das Kompatibilitätslevel 140 zugewiesen. Eine Ausnahme stellt die Model System-Datenbank dar. Wurde das Kompatibilitätslevel angepasst, erhalten alle neuen Datenbanken das Kompatibilitätslevel der Model Datenbank.



Kompatibilitätslevel einer Datenbank anpassen in SSMS

Mainzer Datenfabrik GmbH
Rotterdamer Platz 2, 55120 Mainz
info@madafa.de
madafa.de
+49 6131 3331612

Datenbank Anbinden oder Wiederherstellen

Wird eine bereits existierende Datenbank auf einer neueren Version von SQL Server wiederhergestellt, verändert sich das Kompatibilitätslevel der Datenbank nicht, außer das Kompatibilitätslevel der Datenbank wird nicht mehr von der SQL Server Installation unterstützt. In diesem Fall wird das Kompatibilitätslevel der Datenbank auf den niedrigsten unterstützten Wert des SQL Servers auf dem sie sich befindet angepasst. Wird also beispielsweise eine SQL Server 2005 Datenbank auf einem SQL Server 2017 wiederhergestellt, würde das Kompatibilitätslevel für diese Datenbank auf 100 gesetzt werden.
Dieses Verhalten ist erst einmal nichts Besonderes. Interessant wird es allerdings, wenn man betrachtet, was passiert, wenn eine Datenbank auf Kompatibilitätslevel 120 oder höher gesetzt wird. Die durch Kompatibilitätslevel 120 hinzugefügten Neuerungen haben einen enormen Einfluss auf die Performanz der Datenbank.

Datenbank Kompatibilitätslevel 120

Mit Kompatibilitätslevel 120 wurde der "neue" Cardinality Estimator (CE) eingeführt. Durch ihn war eine generelle Verbesserung der Ausführung von Abfragen zu beobachten.
Der Grund, warum dieser CE als "neu" (in Anführungszeichen) bezeichnet wird, ist, weil jede neue SQL Server Version seit SQL Server 2014 Verbesserungen an CE und Abfrage-Optimierung mit sich gebracht hat. Diese neueren, relevanteren Änderungen werden seit SQL Server 2016 mit CE120 für Kompatibilitätslevel 120, CE130 für Kompatibilitätslevel 130, CE140 für Kompatibilitätslevel 140 und CE150 für Kompatibilitätslevel 150 bezeichnet.

Datenbank Kompatibilitätslevel 130

Mit der Benutzung von SQL Server 2016 oder neuer und dem Kompatibilitätslevel 130 wird automatisch der CE130 benutzt, welcher eine Reihe von performanzrelevanten Änderungen mit sich bringt.
Die Effekte der globalen Trace-Frags 1117, 1118 und 2371 greifen, wenn das Kompatibilitätslevel 130 verwendet wird. Gleichzeitig erhält man den Effekt der globalen Trace-Flag 4199 für alle Abfragen-Optimierungs-Hotfixe, die vor der Veröffentlichung von SQL Server 2016 RTM herausgebracht wurden.
Mit SQL Server 2016 wurde zusätzlich die sog. "database scroped configuration" Optionen eingeführt: sie geben dem Benutzer die Möglichkeit Anpassungen an einer Datenbank vorzunehmen, die vorher nur Instanz-weit durchgeführt werden konnten. Die beiden wichtigsten Optionen sind dabei:

- **LEGACY_CARDINALITY_ESTIMATION**
  Diese Option stellt ein, dass für die Datenbank der "alte" CE verwendet wird. Es ist ein Äquivalent zur Trace-Flag 9481, betrifft aber nur eine spezielle Datenbank, nicht die gesamte Instanz.
  Sie ermöglicht es, eine Datenbank auf Kompatibilitätslevel 130 zu betreiben, gleichzeitig aber den "alten" CE zu verwenden.

- **QUERY_OPTIMIZER_HOTFIXES**
  Diese Option stellt ein, dass alle Abfrage-Optimierungs-Hotfixes, die vor der Veröffentlichung von SQL Server 2016 RTM erschienen sind, verwendet werden, sollte das Kompatibilitätslevel 130 verwendet werden (ohne die Trace-Flag 4199). Wird die Option mit der TF 4199 verwendet, erhält man zusätzlich alle Hotfixes, die nach der Veröffentlichung von SQL Server 2016 RTM herausgegeben wurden.
  Mit dem Service Pack 1 von SQL Server 2016 wurde zusätzlich die "USE HINT" Abfrage-Hinweise eingeführt. Diese sind wesentlich verständlicher als die alten "QUERYTRACEON" Abfrage-Hinweise, die in SQL Server 2014 verwendet werden mussten. Dies gewährleistet eine noch feinere Kontrolle über das Verhalten des Optimierens des zugehörigen Kompatibilitätslevels und die Version des CE, die verwendet wird. Um eine Liste von validen "USE HINT" Namen für die exakten SQL Server Version zu erhalten, kann folgendes Kommando verwendet werden:

```
sys.dm_exec_valid_use_hints
```


Mainzer Datenfabrik GmbH
Rotterdamer Platz 2, 55120 Mainz
info@madafa.de
madafa.de
+49 6131 3331612

Datenbank Kompatibilitätslevel 140

Mit der Benutzung von SQL Server 2017 oder neuer und dem Kompatibilitätslevel 140 wird automatisch CE140 benutzt, der wieder eine Reihe von performanzrelevanten Änderungen mit sich bringt. Mit SQL Server 2017 wurden die neuen sog. "adaptive query processing" Features eingeführt, die ebenfalls standardmäßig mit dem Kompatibilitätslevel 140 verwendet werden. Diese Features beinhalten unter anderem das sog. "batch mode memory grant feedback", "batch mode adaptive joins" und die "interleaved execution"

Nähere Informationen zu den mit SQL Server 2017 eingeführten Neuerungen findest Du hier in unseren Artikeln:

- Showplan Erweiterung in SQL Server 2017
- Automatisierte Plan Korrektur in SQL Server 2017
- Adaptive Abfrage Verarbeitung in SQL Server 2017

Datenbank Kompatibilitätslevel 150

Wie bei den vorherigen Versionen wird mit der Benutzung von SQL Server 2019 automatisch das Kompatibilitätslevel 150 und der CE150 verwendet. Und auch mit SQL Server 2019 sind wieder eine Vielzahl von Verbesserungen und Neuerungen bezüglich der Performanz eingeführt worden. Ein Paradebeispiel ist beispielsweise das sog. "scalar UDF inling", das automatisch das Inlining für viele skalare UDF Funktionen in einer Datenbank übernimmt.
Ein weiteres Beispiel ist das sog. "intelligent query processing feature", das eine Übergruppe der adaptiven Abfrage-Verarbeitung von SQL Server 2017 dargestellt. Dieses neue Feature beinhaltet sog. "table variable deferred compilation", "approximatie query processing" und "batch mode on rowstore".

Konklusion

Das Migrieren einer Datenbank wird mir den modernen SQL Server Versionen im vergleich zur "legacy" Anwendung immer komplizierter. Das liegt vor allem an den großen Veränderungen, die mit den Kompatibilitätsleveln, den CE, usw. einhergehen. Aber gerade aus diesem Grund ist es besonders wichtig, sich mit der neuen SQL Server Version, auf die eine Datenbank migriert werden soll, vertraut zu machen und den gesamten Prozess ausgiebig zu durchdenken.
Der von Microsoft empfohlene Vorgang zur Migration einer Datenbank lautet folgendermaßen: es soll auf die neuste SQL Server Version geupgradet, das Kompatibilitätslevel der Datenbank allerdings beibehalten werden. Als Nächstes sollte der "Query-Store" auf jeder Datenbank eingeschaltet sowie Informationen bezüglich der Workloads gesammelt werden. Erst dann sollte das Kompatibilitätslevel der Datenbank auf die höchste Stufe angepasst werden. Zu guter Letzt kann der Query-Store benutzt werden, um Performanz-Regressionen zu verhindern, indem der letzte gute Plan erzwungen wird. Von einer "blinden" Migrationen ist dringend abzuraten!
Ein neuerer, modernerer Ansatz ist das Testen einer Datenbank mit samt den dazugehörigen Anwendungen auf einem bestimmten Kompatibilitätslevel, statt sich nach der SQL Server Version zu richten. Wird diese Datenbank also in Zukunft auf unterschiedliche, neuere Versionen migriert, bleibt die Performanz dank des Kompatibilitätslevel erhalten.

compatibility level, datenbank, kompatibilitätslevel, migration, mssql, sql server

Mainzer Datenfabrik GmbH
Rotterdamer Platz 2, 55120 Mainz
info@madafa.de
madafa.de
+49 6131 3331612