Parallele Verarbeitung

MSSQL

Bei MSSQL ermöglichen die parallelen Abfragen die Abfrageausführung und Indexvorgänge für Computer zu optimieren, die über mehrere Mikroprozessoren verfügen.

MSSQL nutzt weiterhin mehrere Betriebssystemthreads für die Ausführung von Abfragen oder Indexvorgängen, so können diese effizient und schnell ausgeführt werden.

Um eine Abfrage optimieren zu können, sucht MSSQL erst mal nach Abfragen und Indexvorgängen die für eine parallele Ausführung vorteilhaft sind. Danach fügt es für diese Verteilungsoperatoren in denn Abfrageausführungsplan ein, damit bereitet es die ausgewählten Abfragen für die parallele Ausführung vor. Ein Verteilungsoperator ermöglicht die Prozessverwaltung, die Neuverteilung der Daten und die Ablaufsteuerung. Er schließt außerdem die Untertypen Distribute Streams, Repartition Streams und Gather Streams ein.

Nachdem der Verteilungsoperator eingefügt wurde, ist das Ergebnis ein Plan für eine parallele Abfrageausführung. Dieser kann mehrere Threads verwenden, die genaue Anzahl der Threads wird während der Initialisierung definiert und durch die Komplexität des Plans und den Grad der Parallelität bestimmt. Der Grad der Parallelität die maximal zu verwendete Anzahl an CPUs.

SAP HANA Epress: nichts genaues gefunden

Spaltenorientiert

Bei spartenorientierten Datenbanken werden die Inhalte Spaltenweise physisch abgespeichert.

Die Daten sind aus Sicht das Betriebssystem in einer eindimensionalen Folge von Bytes angeordnet. Bei einer zeilenorientierten Datenbank werdne alle Datenwerte einer Zeile aneinander gehängt. Im Gegensatz dazu werden die Datenwerte einer spaltenorientierten Datenbank Spalte für Spalte aneinander gehängt.

Das führt zu einer Beschleunigung der Datenbereitstellung, da die Festplattenzugriffe verringert werden.

Das bringt Vorteile mit sich bei Anwendung wo z.B. Aggregate über große Zahlen oder ähnliches gebildet werden. Wenn ein Aggregat über viele Zeilen, aber nur wenig Spalten gebildet werden muss, kann man nun nur die gesuchten Spalten auslesen aber nicht mehr alle wie bei zeilenorientierten Datenbanken. Außerdem ist es effizienter wenn eine Spalte über alle Zeilen der Tabelle einen neuen Wert erhält, da man keine Rücksicht auf die Inhalte der anderen Spalten nehmen muss und so effizient schreiben kann.

Ein weiterer Vorteil ist das sich die Spalten orientierten Systeme sich selbst indexieren und so weniger Speicherplatz benötigt wird.

In der Praxis sind Spalten orientierte Systeme gut für OLAP- Aufgaben geeignet, da diese durch eine kleine Anzahl sehr komplexer Abfragen über alle Datensätze definiert sind.

Quellen:

<https://de.wikipedia.org/wiki/Spaltenorientierte_Datenbank>

https://technet.microsoft.com/de-de/library/ms178065(v=sql.105).aspx