

Implementation und Performance-Vergleich von Voraggregierungsstrategien für OLAP-Cubes

Einleitung: Range-queries in array-basierten OLAP-Cubes können durch geeignete Voraggregation der Werte beschleunigt werden, ohne dass dabei zusätzlicher Speicherplatz verwendet wird. Hierzu werden gewisse Basiswerte im Würfel durch aggregierte Werte ersetzt, so dass bei Range-Summen-Berechnungen möglichst wenige Zellen abgefragt werden müssen. Im Gegenzug müssen jedoch beim Update einer Basiszelle immer auch alle von dieser Zelle betroffenen anderen Zellen ebenfalls aktualisiert werden, so dass Updates mehr Zeit erfordern.

Es gibt unterschiedliche Strategien zur Voraggregation, wie z.B. die *Local Prefix Sum* [1], die *Relative Prefix Sum* oder den *Dynamic Data Cube* [2]. Teilweise lassen sich dabei auch Dimensionshierarchien effizient unterstützen. Zusätzlich können beim Ansatz des so genannten *Iterative Data Cube* [3] mehrere dieser Strategien miteinander kombiniert werden, indem für jede Dimension eine eigene Strategie gewählt werden kann.

Während die asymptotischen Laufzeiten für die obigen Ansätze in der Theorie gut verstanden und analysiert sind, stellt sich die Frage, wie sehr sich die genannten Ansätze mit ihren jeweiligen Tradeoffs zwischen Anfrage- und Update-Verhalten in der Praxis tatsächlich unterscheiden.

Thema: In dieser Bachelorarbeit sollen verschiedene Voraggregationstechniken im Hinblick auf ihre Performanz bei Range-Sum-Queries und Updates miteinander verglichen werden. Die in der Praxis häufig auftretenden Abfragen über Dimensionshierarchien sollen dabei speziell untersucht werden.

Vorgehensweise: In einem ersten Schritt müssen Testszenarien entwickelt werden, die sich an möglichst vielen Arten von real vorkommenden OLAP-Anfragen orientieren sollen. Dazu kann auf die Erfahrung von Jedox mit der OLAP-Engine *Palo* zurückgegriffen werden.

Im Anschluss soll eine den Szenarien entsprechende Testumgebung implementiert werden. Gegebenenfalls müssen außerdem die zu untersuchenden Datenstrukturen, die zum großen Teil bereits in einer Java-Implementation zur Verfügung stehen, an die Testszenarien angepasst werden.

Für die Tests selbst stehen Beispiel-Datenbanken von Jedox zur Verfügung, die verwendet werden sollen. Als Vergleichskriterium ist primär die reale Zeit der Anfrageverarbeitung relevant. Interessant sind aber auch die Zahl der Zellzugriffe bzw. der bei einer Operation ausgeführten Vergleichs- bzw. Rechenoperationen.

Die Testergebnisse sollen visuell aufbereitet und in der schriftlichen Arbeit ausführlich dargestellt werden.

Bewertungskriterien:

- Dokumentierter Code (Javadoc)
- Schriftliche Ausarbeitung (30-50 Seiten ohne Anhang)
- Abschlusspräsentation (ca. 45 Minuten)