Proc SQL: Passthrough-Facility für effizientes Datenmanagement bei komplexen großen Datenbeständen

Kaluscha, Rainer; Holstiege, Jakob; Krischak, Gert
Institut für Rehabilitationsmedizinische Forschung an der Universität Ulm
rainer.kaluscha@uni-ulm.de

Bei großen Datenbeständen mit komplexen Datenstrukturen ist für ein effizientes Arbeiten ein adäquates Datenmanagement erforderlich. Dabei erlauben ausgereifte relationale Datenbankmanagementsysteme die performante Abwicklung auch komplexer Abfragen auf großen Datenmengen.

SAS bietet mit den ACCESS-Modulen via Proc SQL oder Libname-Statements eine flexible und weitgehend datenbankunabhängige Unterstützung für den Zugriff auf solche Datenbanksysteme. Sind aber in einer komplexen Abfrage viele große Tabellen miteinander zu verbinden (joins), kann es aus Performance-Gründen angebracht sein, die Passthrough-Facility von Proc SQL zu nutzen, so dass der SQL-Code unverändert an das dahinterliegende Datenbankmanagementsystem durchgereicht und von diesem ausgeführt wird [1] . Zum einen ist in der Regel die Hardware eines Datenbankservers für solche Aufgaben ausgelegt; zum anderen verfügt die Datenbankmanagementsoftware über ausgefeilte Anfrageoptimierungstechniken (query optimizer), die z.B. die für die Tabellen verfügbaren Indexe und ihre Selektivität berücksichtigen. Gerade bei der Selektion kleinerer Stichproben aus einem großen Datenbestand kann dies erhebliche Auswirkungen haben.

Ferner kann auch in der Datenbank vorliegender Code [2] für Plausibilitätsprüfungen oder spezielle Auswertungen (stored procedures) genutzt werden, um eine doppelte Implementierung zu vermeiden.

An einem Beispiel aus der medizinischen Versorgungsforschung wird das Vorgehen illustriert. Ein großer anonymisierter Datenbestand mit mehreren Millionen Datensätzen ist in einer leistungsfähigen Oracle-Datenbank gespeichert. Die für die jeweilige Auswertung nötigen Daten werden aus den verschiedenen Tabellen selektiert und miteinander verknüpft, um sie anschließend in SAS zu analysieren. Der Import kann dabei über Proc SQL oder ein Libname-Statement mit entsprechend definierten Datenbankviews erfolgen. Ersteres hat dabei den Vorteil, dass Datenbankabfrage und SAS-Code für die Analyse zusammen gepflegt werden können; bei letzterem können zentrale Vorgaben, z.B. bez. Datenschutz, besser umgesetzt werden.

Verweise:

[1] Chapman T & Carleton L: SAS with Oracle: Writing Efficient and Accurate SQL.

Pacific Northwest SAS Users Group (PNWSUG), 2009.

Online: http://www.sascommunity.org/wiki/SAS_with_Oracle:_Writing_Efficient_and_Accurate_SQL

[2] Fogleman S: SAS and Oracle PL/SQL: Partners or Competitors?

Western Users of SAS Software (WUSS), 2008.

Online: http://www.wuss.org/proceedings08/08WUSS%20Proceedings/papers/cod/cod04.pdf