Exposé zur Studienarbeit

Speicherung linguistischer Korpora in Datenbanken

Thorsten Vitt*

August – Oktober 2004
Betreuung: Lukas Faulstich, Prof. Ulf Leser,
Forschungsverbund Linguistik – Informatik – Bioinformatik

1 Hintergrund

Im Projekt *Deutsch.Diachron.Digital (DDD)* [1] entsteht ein umfangreiches historisches Korpus des Deutschen: Eine Sammlung zahlreicher historischer und zeitgenössischer deutscher Texte, teils in verschiedenen, miteinander alignierten Versionen (etwa diplomatischer – an der physischen Repräsentation orientierter –, normalisierter und übersetzter Fassungen). Die Texte sind mit verschiedenen Annotationen versehen, etwa zur physischen und logischen Aufteilung oder in Form von Syntaxbäumen.

Linguistische Korpora liegen oftmals als XML-Dateien vor. Um mehrere Annotationsebenen in einem Dokument sowie nicht baumförmige Annotationen wie etwa aus dem TIGER-Korpus (Brants et al., 2002) zu unterstützen, wurde für DDD jedoch ein über die baumförmige Struktur von XML hinausgehendes Datenmodell auf der Basis von geordneten gerichteten azyklischen Graphen (ODAGs) entwickelt (Dipper et al., 2004). Gespeichert werden soll das Korpus in einem relationalen (bzw. objektrelationalen) Datenbanksystem mit einer Komponente zur Volltextsuche.

2 Zielsetzung

In der Studienarbeit sollen unterschiedliche Speicherverfahren für linguistische Korpora auf der Basis des o. g. ODAG-Datenmodells im Datenbankmanagementsystem

*vitt@informatik.hu-berlin.de
[1] http://www.deutschdiachrondigital.de

Oracle implementiert und in Bezug auf typische Anfragen bewertet werden:

generische Datenbankrepräsentation: Hier werden alle Elemente der unterschiedlichen Annotationsebenen gemeinsam in einer Tabelle verwaltet.

eine Tabelle pro Elementtyp (orientiert am Schema des Korpus): In dieser Variante werden für unterschiedliche Elementtypen der Annotationsebenen verschiedene Tabellen angelegt.

Für Anfragen auf der Menge aller Dokumente wird die Obermenge aller Elemente in unterschiedlichen Varianten realisiert:

- nicht,
- als View,
- als materialisierter View,
- nach Möglichkeit objektrelational über Vererbung von Tabellen.

XML-Repräsentation in einer XML-Datenbank, hier sollen die Möglichkeiten von Oracle XML DB zur Speicherung und Anfrage von XML-Daten genutzt werden (vgl. u. a. Murthy und Banerjee, 2003).

Darüberhinaus sollen generische Methoden zum Import und Export der Daten zwischen der Datenbank und einer XML-Repräsentation entworfen und implementiert werden.

3 Herangehensweise

3.1 Oracle XML DB

Zunächst werden die Möglichkeiten von Oracle XML DB in Bezug auf ihre Anwendbarkeit auf das ODAG-Datenmodell untersucht. Von Interesse sind hierbei insbesondere die Möglichkeiten zur Behandlung der über das XML-Datenmodell herausgehenden Eigenschaften der ODAGs: mehrere Elternknoten und Maschen im Graphen.

3.2 Relationale Varianten

Weiterhin werden die relationalen Varianten realisiert. Für den Entwurf des Datenbankschemas der Variante mit Tabellen pro Elementtyp werden dabei auch Ansätze aus der Literatur zur Speicherung von XML-Daten in relationalen Datenbanken betrachtet (etwa aus Florescu und Kossmann, 1999; Murthy und Banerjee, 2003), insbesondere pfadbasierte Ansätze (Prakash et al., 2004).

In diesem Zusammenhang werden auch die Auswirkungen der verschiedenen Realisierungen der Obermenge aller Elemente untersucht.

3.2.1 Objektrelationale Variante

Der intuitive Ansatz nach SQL:1999, die Obermenge als Supertabelle und die einzelnen Elemente als Subtabellen darunter anzulegen, ist in Oracle 9i mangels der Unterstützung von Subtabellen (vgl. Türker, 2003) so nicht umsetzbar.

Oracle bietet jedoch hierarchisch organisierbare *Object Views* (Oracle Corp., Kapitel 5) und weitere objektrelationale Features, deren Anwendbarkeit auf das ODAG-Modell untersucht wird.

3.3 Import / Export

Schließlich sollen geeignete Im- und Exportmethoden gefunden werden. Für den **Import** ist etwa die Generierung von SQL-Importscripten aus den Korpusdaten mit XSLT oder die Nutzung der Methoden zum XML-Import von Oracle XML DB ggf. mit Nachbearbeitung in der Datenbank zu untersuchen, für den **Export** die Nutzung entsprechender XML-Funktionen wie XMLELEMENT aus SQL/XML / SQL:2003 im DBMS mit anschließender Nachbearbeitung mittels XSLT.

Literatur

Brants, S., Dipper, S., Hansen, S., Lezius, W. und Smith, G. (2002): The TIGER Treebank. In *Proceedings of the Workshop on Treebanks and Linguistic Theories, September 20-21*. Sozopol, Bulgaria. URL http://www.ims.uni-stuttgart.de/projekte/TIGER/paper/treeling2002.pdf.

Dipper, S., Faulstich, L., Leser, U. und Lüdeling, A. (2004): Challenges in Modelling a Richly Annotated Diachronic Corpus of German. In *Workshop on XML-based richly annotated corpora*. Lisbon, Portugal.

Florescu, D. und Kossmann, D. (1999): A Performance Evaluation of Alternative Mapping Schemes for Storing XML Data in a Relational Database. Technical-Report, Inria. URL http://dodgers.fmi.uni-passau.de/~kossmann/Papers/xml.pdf.

Murthy, R. und Banerjee, S. (2003): XML Schemas in Oracle XML DB. In *Proceedings of the 29th VLDB Conference*. Berlin.

Oracle Corp. (2002): *Oracle9i Application Developer's Guide – Object-Relational Features*. URL http://download-west.oracle.com/docs/cd/B10501_01/appdev.920/a96594/adobjvew.htm#433584.

Prakash, S., Bhowmick, S. S. und Madria, S. (2004): SUCXENT: An Efficient Path-based Approach to Store and Query XML Documents. In *DEXA 2004*. To appear.

Türker, C. (2003): *SQL:1999 & SQL:2003 – Objektrelationales SQL, SQLJ & SQL/XML*. dpunkt.verlag, Heidelberg.