

Architektur und Prototyp eines Replication Proxy Server für die nutzerdefinierte Replikation von Datenbankinhalten

Thomas Müller

Friedrich-Schiller-Universität Jena, Lehrstuhl für Datenbanken und Informationssysteme, Ernst-Abbe-Platz 1-4, 07743 Jena
muellert@informatik.uni-jena.de

Zusammenfassung

Neuartige mobile Anwendungen, wie das mobile Reiseinformationssystem *HERMES*, stellen neue Anforderungen an Datenbanksysteme. Zu diesen Anforderungen gehört die nutzerdefinierte Auswahl von Daten sowie deren unverbundene Verfügbarkeit und Änderbarkeit. Zur Realisierung dieser Anforderungen wird von uns eine Architektur eingesetzt, deren zentraler Bestandteil ein *Replication Proxy Server*, kurz RPS, ist. Ziel meiner Diplomarbeit ist der Entwurf und die prototypische Umsetzung eines solchen RPS. Dazu werden in meiner Diplomarbeit zunächst die zentralen Aufgaben des RPS identifiziert und die RPS-Schnittstellen definiert. Daran schließt sich die Entwicklung einer modularen Architektur an, welche in einem folgenden Schritt prototypisch implementiert wird.

1 Einführung und Motivation

Durch die Erweiterung der heutigen Computertechnologie um eine neue Dimension, die Mobilität von Geräten und Nutzern, wurde die Voraussetzung für neue Arten von Anwendungen geschaffen. Diese neuartigen Anwendungen stellen wiederum neue Anforderungen an die Datenbanksysteme [Gol02].

Als Beispiel für eine solche „neuartige“ Anwendung sei hier das mobile Reiseinformationssystem *HERMES* genannt, welches seinen Nutzern Informationen über Orte, Sehenswürdigkeiten, Hotels etc. zur Verfügung stellt [Bau]. Der Unterschied zu traditionellen Touristenführern besteht darin, dass nur ein geringer Teil der Daten (z. B. geographische Informationen über Städte) zentral bereitgestellt wird. Der überwiegende Teil der Informationen wird von den mobilen Nutzern selbst, i. d. R. vor Ort, erstellt bzw. ergänzt. Dazu gehören beispielsweise Öffnungszeiten von Museen, aktuelle Eintrittspreise oder Restaurantkritiken. Als mobile Geräte sollen sowohl Geräte der Laptop- als auch der Handheldklasse genutzt werden, die Kommunikation soll drahtlos erfolgen.

Drahtlose Kommunikationsmittel sind teuer, besitzen eine geringe Bandbreite und sind zudem nicht an allen Orten verfügbar. Um den mobilen Nutzern sowohl die benötigten Daten als auch entsprechende Manipulationsmöglichkeiten jederzeit kostengünstig zur Verfügung stellen zu können, muss den mobilen Nutzern ein zeitweise unverbundenes Arbeiten ermöglicht werden. Die Fähigkeit zum zeitweise unverbundenen Arbeiten erweist sich also als eine zentrale Anforderung.

Die Grundlage einer verteilten und zeitweise unverbundenen Arbeit ist ein Dienst zur *Replikation* von Daten, also zur konsistenten Verwaltung von Kopien von Primärdaten. Die Besonderheit der hier betrachteten „neuartigen“ Anwendungen besteht gegenüber den traditionellen Anwendungen darin, dass die Datenauswahl deskriptiv durch den *Nutzer* bzw. die *Anwendung* erfolgen soll, weshalb hier anstelle einer herkömmlichen Replikationslösung eine **nutzerdefinierte Replikation** benötigt wird [Gol02].

Der prinzipielle bei der nutzerdefinierten Replikation auftretende Arbeitsablauf kann wie folgt beschrieben werden. Zunächst wählt der mobile Client (im verbundenen Zustand) die zu replizierenden Daten deskriptiv aus. Anschließend werden diese Daten zum mobilen Client übertragen,

welcher dann unverbunden auf diesen Daten arbeiten kann. Dem schließt sich eine Synchronisation der replizierten Daten an, d. h. es erfolgt eine beidseitige Änderungsübertragung, verbunden mit einer gegebenenfalls nötigen Konfliktbehandlung. Nun kann erneut unverbunden gearbeitet werden.

Die nutzerdefinierte Replikation wird von kommerziellen mobilen Datenbanksystemen bisher nur unzureichend unterstützt. Aus diesem Grund soll eine Architektur entwickelt werden, die das mobile Reiseinformationssystem und andere neue Anwendungen unterstützt. Zentraler Bestandteil dieser Architektur ist der *Replication Proxy Server*, kurz RPS. Der RPS stellt den mobilen Nutzern die vom Replikationsadministrator zur Replikation freigegebenen Daten zur Verfügung, verarbeitet die von den mobilen Nutzern gestellten Replikations- bzw. Synchronisationsanforderungen und behandelt die bei der Replikation auftretenden Konflikte. Zu den weiteren Aufgaben des RPS gehört die Verwaltung der zur nutzerdefinierten Replikation benötigten Hilfsdaten.

2 Schnittstellen des RPS

Der RPS besitzt sowohl eine Schnittstelle zum Replikationsadministrator als auch zu den mobilen Clients. Diese Schnittstellen lassen sich durch die Dienste beschreiben, die der RPS dem Replikationsadministrator bzw. den mobilen Clients zur Verfügung stellt.

Mit Hilfe der dem Replikationsadministrator bereitgestellten Dienste kann dieser festlegen, welche Daten den mobilen Clients zur Replikation angeboten werden und welche Konfliktbehandlungsmethoden [Lie] dabei eingesetzt werden. Den mobilen Clients werden im Wesentlichen Dienste zum Anfordern von Daten bzw. zur Synchronisation angeboten.

Das Anfordern der zum lokalen unverbundenen Arbeiten benötigten Daten erfolgt durch den mobilen Client deskriptiv mittels der Anweisung `CREATE REPLICATION VIEW`. Einerseits spezifiziert der mobile Client dabei über eine `SELECT`-Anweisung, welche Daten repliziert werden sollen. Andererseits kann der mobile Client auch Zusicherungen für die ausgewählten Daten anfordern, z. B. ein exklusives Änderungsrecht. Zudem besteht für den mobilen Client die Möglichkeit, eine Parameterisierung der Konfliktbehandlungsmethoden vorzunehmen [Lie]. Da aus Platzgründen auf eine genaue Beschreibung dieser Anweisung verzichtet werden muss, soll ihre Wirkung anhand eines einfachen Beispiels veranschaulicht werden.

Es werde angenommen, dass sich ein auf dem Weg nach Leipzig befindender Reisender über die Leipziger Gastronomie informieren will. Dies bedeutet, dass auf dem mobilen Client sämtliche Daten über Leipziger Restaurants zum unverbundenen Lesen benötigt werden. Da die Daten lokal nicht geändert werden sollen, ist weder die Anforderung von Zusicherungen noch eine Parameterisierung von Konfliktbehandlungsmethoden sinnvoll bzw. nötig. Die gewünschten Daten können vom mobilen Client (bzw. der Client-Anwendung), wie in der folgenden Abbildung vereinfacht dargestellt, zum unverbundenen Lesen angefordert werden. Dabei wird vorausgesetzt, dass die entsprechenden Daten vom RPS in den Tabellen *Restaurant* und *Bewertung* zur Replikation angeboten werden. Geeignete Attribute werden ebenfalls vorausgesetzt.

```
CREATE REPLICATION VIEW EssenInLeipzig
AS SELECT r.Name, b.Text
   FROM Restaurant r, Bewertung b
  WHERE b.RNr=r.Nr AND r.Ort='Leipzig'
```

EssenInLeipzig

Name	Text
Auerbachs Keller	„sehr gut“
Auerbachs Keller	„herrlich“
...	...

Die vom mobilen Client gestellte Anweisung wird auf dem RPS verarbeitet. Da im konkreten Fall weder Zusicherungen gefordert wurden noch eine Parameterisierung von Konfliktbehand-

lungsmethoden durchgeführt wurde, beschränkt sich die Verarbeitung der gestellten Anfrage im Wesentlichen auf die Ermittlung der zum mobilen Client zu übertragenden Daten. Dabei wird berücksichtigt, welche Daten bereits auf dem mobilen Client vorliegen und deshalb nicht nochmals übertragen werden müssen. Das für die Anwendung sichtbare Ergebnis der Anweisungsverarbeitung ist eine auf dem mobilen Client unverbunden lesbare View *EssenInLeipzig* (vgl. vorige Abbildung). Sofern die Daten jedoch auch zum unverbundenen Ändern angefordert werden, ist die Verarbeitung der Anweisung deutlich aufwendiger, da dann u. a. geprüft werden muss, ob eine Änderung prinzipiell überhaupt möglich ist, ob der Nutzer berechtigt ist, die Änderung durchzuführen und ob anderen mobilen Nutzern gewährte Zusicherungen der Durchführbarkeit der Änderung entgegenstehen.

3 Architektur und Prototyp

Ausgehend von den vom RPS bereitzustellenden Diensten wird eine modulare RPS-Architektur entwickelt, welche im Anschluss prototypisch implementiert wird.

Literatur

- [Bau] K. Baumgarten. Konzept und Realisierung eines Reiseinformationssystems. Studienarbeit, Institut für Informatik, Friedrich-Schiller-Universität Jena. In Vorbereitung.
- [Gol02] C. Gollmick. Konzept und Anforderungen der nutzerdefinierten Replikation zur Realisierung neuer mobiler Datenbankanwendungen. Jenaer Schriften zur Mathematik und Informatik Math/Inf/15/02, Institut für Informatik, Friedrich-Schiller-Universität Jena, September 2002.
- [Lie] M. Liebisch. Synchronisationskonflikte beim mobilen Datenbankzugriff: Vermeidung, Erkennung, Behandlung. Diplomarbeit, Institut für Informatik, Friedrich-Schiller-Universität Jena. In Vorbereitung.