

Replikation in mobilen Datenbanken*

Hagen Höpfner, *hoepfner@iti.cs.uni-magdeburg.de*
Otto-von-Guericke-Universität Magdeburg

1 Einleitung

Die Nutzerakzeptanz mobiler Telekommunikationsmedien hat in den letzten Jahren stetig zugenommen. Anhand der erzielten Preise bei der Versteigerung der UMTS-Lizenzen im Herbst 2000 ist deutlich zu erkennen, daß auch seitens der Telekommunikationsanbieter großes Interesse an dieser Technologie besteht. Mit dem zunehmendem Ausbau der mobilen Datennetze steigt auch die Anzahl möglicher Anwendungsgebiete verteilter mobiler Datenbanken. Die Entwicklung von immer leistungsfähigeren und benutzerfreundlicheren mobilen Computern führt dazu, daß zukünftig noch mehr Anwender solche Technologien einsetzen werden.

Ein Problem bei mobilen Endgeräten ist, daß sie nicht permanent am Netz angemeldet sein müssen. Dies kann zu einem an Empfangs- und Sendeproblemen oder schlicht an der Tatsache, daß der Nutzer sie ausschaltet, liegen. Die Replikation in dem beschriebenen Einsatzfeld erfordert neue Konzepte, um den dort auftretenden Anforderungen gerecht zu werden. Insbesondere ist hier eine *Adaptivität* der zu verwendenden Verfahren zu gewährleisten, die es ermöglicht, dynamische Aspekte der Infrastruktur, in der die Replikationsverfahren zum Einsatz kommen, zu berücksichtigen. Zum Beispiel erscheint es sinnvoll, Informationen zum zeitlichen Verhalten von Netzwerkknoten, Nutzern und der Netzwerktopologie auszuwerten, um davon ausgehend effiziente Replikationsstrategien zu verfolgen.

Im Rahmen der Diplomarbeit werden grundlegende Anforderungen an Replikationstechniken für mobile Datenbanken erarbeitet, insbesondere Verfahren zur Synchronisation mobiler Datenbanken, wie z.B. *Primary Copy* oder *Quorum*-Verfahren betrachtet und bereits existierende Replikationsverfahren wie z.B. Bayou [3] auf ihre Eignung für mobile Datenbanken hin untersucht.

2 Begriffe

Nach [6] wird im mobilen Umfeld zwischen *mobilen Knoten* (MK) und *statischen Knoten* (SK) unterschieden. *Basisknoten* (BK) sind SK, welche die Einwahl von MKen in das statische Netzwerk ermöglichen. MKen, welche sich in einem gewissen Umkreis eines BKen befinden, werden in *Zellen* zusammengefaßt. Der Begriff *Cluster* faßt allen Knoten zusammen, die sich in einer gemeinsamen geographischen Gebiet befinden. Ist ein MK offline, wird er auch *disconnected node* genannt.

3 Problemstellung

MKen sind nicht immer erreichbar (vgl. Abschnitt 2). Mobile Datenbanksysteme müssen die daraus resultierenden Probleme lösen. Ein hierbei zu berücksichtigender Aspekt ist die Replikation.

Replikation in verteilten Datenbanken umfaßt nach [1] die vier Bereiche Kopie-Update-Strategie (Datensynchronisation), Fehlerbehandlung, Synchronisation konkurrierender Zugriffe und Behandlung von Lesetransaktionen. In diesem Papier werden nur die ersten beiden Aufgaben der Replikation betrachtet. Die Kopie-Update-Strategien dienen hierbei der Datenkonsistenzsicherung im fehlerfreien Betrieb. Eine Fehlerbehandlung umfaßt Funktionen zum Abgleich der Daten

*Diese Arbeit wird von der DFG gefördert (SA 782/3-1)

nach einem Fehlerfall. Das heißt, daß zum Zeitpunkt der Ausführung einer Transaktion nicht alle Knoten erreichbar waren.

4 Replikation in mobilen Datenbanken

Aufgrund besonderen Anforderungen an mobilen Datenbanken und deren Eigenschaften ist der Einsatz spezieller Verfahren notwendig. MKen sind in der Regel offline und werden nur zur Datensynchronisation mit dem Netz verbunden. Primär-Kopie-Ansätze und progressive Verfahren [1] bilden die Grundlagen für Replikationsverfahren für mobile Datenbanken. Die folgenden Betrachtungen beschreiben einige solcher erweiterten Verfahren.

4.1 Provisorische Transaktionen

Dieser Ansatz wird beispielsweise bei der Zwei-Schichten-Replikation [2], welche ein erweitertes Primär-Kopie-Verfahren ist, eingesetzt. Ist ein MK offline, werden alle Transaktionen vorerst provisorisch durchgeführt. Dabei werden provisorische Daten erzeugt. Provisorische Transaktionen unterliegen sogenannten Bereichsregeln, welche den Gültigkeitsbereich dieser Transaktionen definieren. Beispielsweise kann somit sichergestellt werden, daß Kontoänderungen nur dann global durchgeführt werden dürfen, wenn dadurch der Kontostand nicht negativ wird. Die provisorischen Transaktionen werden nach der Wiedereinwahl auf die Basisstation übertragen. Dort werden diese, unter Beachtung der Bereichsregeln, global auszuführen. Die entsprechenden Ergebnisse werden an den MK zurück propagiert. Kann eine provisorische Transaktion nicht global ausgeführt werden, wird eine Fehlermeldung generiert. Anderenfalls wird diese Transaktion in eine Basistransaktion überführt.

Die Propagation von Transaktionen im statischen Netz erfolgt nach einem Standard-Primär-Kopie-Verfahren.

Ein weiteres Beispiel für die Verwendung von provisorischen Transaktionen ist das Bayou-System [8]. Diesem System bietet weitere Techniken zur Konflikterkennung bzw. -behebung bei der Synchronisation. Schreiboperationen werden um *per-write dependency checks* und *per-write merge-procedures* erweitert [7]. Schlägt bei der Synchronisation der Abhängigkeitstest fehl, wird versucht, den Konflikt durch die Mischfunktion zu beheben. Ist dies nicht möglich, erfolgt eine Fehlermeldung.

4.2 Snapshots

Ein weiterer Replikationsansatz ist die Haltung verschiedener Datenversionen. Das folgende Beispiel [5] illustriert die Funktionsweise von Snapshot-Verfahren. Gegeben sei der folgender Ablauf auf der Basistation:

1. Schreiben der beiden Attribute $x = 1$ und $y = 1$ und Abschließen der Transaktion
2. Lesen der Werte der beiden Attribute
3. Schreiben von $x = 2$ und Abschließen der Transaktion

Der MK liest die Daten $x = 1$ und $y = 1$ nach der Bestätigung der ersten Transaktion, geht offline, schreibt lokal $y = 3$ und bestätigt die entsprechende lokale Transaktion.

Bei der nächsten Synchronisation besteht offensichtlich ein Konflikt (BK: $x = 2$, $y = 1$; Transaktion auf MK hat $x = 1$ und $y = 1$ gelesen). Bei der Verwendung von Snapshots, sind beide Versionen ($V1 : \{x_0 = 1, y_0 = 1\}$, $V2 : \{x_1 = 2, y_0 = 1\}$) gespeichert. Diese Informationen können zur Behebung des Konfliktes genutzt werden. Die Schreiboperation kann folgendermaßen geändert werden:

1. Schreiben der beiden Attribute $x = 1$ und $y = 1$ und Abschließen der Transaktion

2. Lesen der Werte beiden Attribute $x = 1$ und $y = 1$ und der Werte dieser Attribute auf dem Client ($x = 1, y = 1$)
3. Übertragen des neuen y -Wertes vom Client auf den Server ($y = 3$) und Abschließen der Transaktion
4. Schreiben von $x = 2$ und $y = 3$ und Abschließen der Transaktion

Dieser Ansatz funktioniert nur, wenn der y -Wert auf dem BK nicht geändert wurde. Ein Verfahren, welches diesen Ansatz verfolgt, ist der Multiversion-Reconciliation-Algorithmus [4], bei welchem zusätzliche Konfliktbehebungsfunktionen eingeführt werden, die, wenn möglich, abhängig von einer Kostenfunktion (Kostenminimum) die Synchronisation zwischen Snapshots durchführen.

5 Fazit und Ausblick

Einige klassische Replikationsverfahren für verteilte, nicht-mobile Datenbanken bilden die Grundlage für Ansätze für die Replikation in mobilen Datenbanken, welche in diesem Papier kurz vorgestellt wurden.

Im Rahmen des DFG-Projektes „Adaptive Replikation von Daten in heterogenen mobilen Kommunikationsnetzen“ wird untersucht, inwieweit die in diesem Papier vorgestellten Replikationstechniken um Aspekte der Adaptivität erweitert werden können. Beispielsweise wird der Einsatz von Techniken zur Konfliktbehebung und -behandlung aus dem Bereich der Datenförderung im Kontext der Synchronisation von Replikation betrachtet. Ein weiter zu untersuchender Aspekt ist die Replikation zwischen MKen.

Literatur

- [1] P. Dadam. *Verteilte Datenbanken und Client/Server-Systeme — Grundlagen, Konzepte, Realisierungsstrukturen*. Springer-Verlag, Berlin, 1996.
- [2] J. Gray, P. Helland, P. O’Neil, and D. Shasha. The dangers of replication and a solution. *SIGMOD Record*, 25(2):173–182, 1996.
- [3] K. Petersen, M. J. Spreitzer, D. B. Terry, and M. M. Theimer. Bayou: Replicated database services for worldwide applications. In *Proceedings Seventh ACM SIGOPS European Workshop (EuroSIGOPS ’96)*, pages 275–280, Connemara, Ireland, September 1996. Association for Computing Machinery, Inc.
- [4] Shirish H. Phatak and B. R. Badrinath. Multiversion reconciliation for mobile databases. In *Proceedings of the 15th International Conference on Data Engineering, 23-26 March 1999, Sydney, Australia*, pages 582–589. IEEE Computer Society, March 1999.
- [5] Shirish H. Phatak and B.R. Badrinath. Transaction-centric reconciliation in disconnected databases. *ACM Monet Journal*, 53, 1999.
- [6] E. Pitoura and B. Bhargava. Maintaining consistency of data in mobile distributed environments. In *Proceedings of the 15th International Conference on Distributed Computing Systems*, pages 404–413, May 1995.
- [7] D. B. Terry, M. M. Theimer, K. Petersen, A. J. Demers, M. J. Spreitzer, and C. Hauser. Managing update conflicts in bayou, a weakly connected replicated storage system. In *Proceedings 15th Symposium on Operating Systems Principles (SOSP-15)*, pages 172–183, Cooper Mountain, Colorado, December 1995. Association for Computing Machinery, Inc.
- [8] M. M. Theimer, A. J. Demers, K. Petersen, M. J. Spreitzer, D. B. Terry, and B. B. Welch. Dealing with tentative data values in disconnected work groups. In *Proceedings of the Workshop on Mobile Computing Systems and Applications*, pages 192–195, Santa Cruz, California, December 1994. Institute of Electrical and Electronics Engineers.