

Transaction Logging

Mit der Auslieferung von R5 wurde innerhalb von Notes/Domino das so genannte Transaction Logging (die Transaktionsprotokollierung) eingeführt. Mit der Transaktionsprotokollierung verfolgte IBM/Lotus gleich mehrere Ziele: Performance-Verbesserungen, Schneller Wiederanlauf nach Server-Abbruch, und den Möglichkeiten inkrementeller Datensicherung.

Performance-Verbesserungen

Unter Verwendung der Transaktionsprotokollierung kann in einigen Fällen ein nicht unerheblicher Faktor an Performance-Verbesserungen erzielt werden. Der Grund dafür liegt darin, dass im Falle einer hohen Server-Auslastung Datenbank-Aktualisierungen zeitlich versetzt durchgeführt werden. Dies macht sich besonders dann bemerkbar, wenn für die Transaktionsprotokollierung ein Laufwerk verwendet wird, das nicht mit dem Laufwerk identisch ist, auf dem sich der Domino-Server befindet. Diese Informationen werden also im günstigsten Fall auf einem separaten Laufwerk zwischengespeichert. Ein Vorteil dieser schnellen I/O-Tätigkeit für dieses Zwischenspeichern liegt weiterhin darin, dass die Informationen zusammenhängend in komplette Sektoren auf der Platte geschrieben werden können und nicht, wie es im normalen Betrieb von Domino der Fall ist, bei dem in der Regel „chaotischen“ Schreiben von Datenbank-Updates in diverse freie Plattensektoren. Wohlgemerkt: Es handelt sich bei dieser Art der Speicherung um eine Art „Zwischenspeicherung“ von Datenbank-Änderungen (der erfahrene iSeries-Betreiber wird hier gewisse Parallelen mit dem Journaling auf der iSeries wieder finden; übrigens stand das Journaling der iSeries in der Tat Pate für die Transaktionsprotokollierung). Nachdem die Änderungen an einer Datenbank zu Zeiten hoher Server-Auslastung in den Zwischenbereich – als Transaktionsprotokoll – geschrieben worden sind, werden diese, sobald die Server-Last unter einen bestimmten Grenzwert zurückfällt, verarbeitet. Das heißt: Die Datenbank-Änderungen werden in die Datenbank geschrieben.

Schneller Wiederanlauf nach Server-Abbruch

Bisher bedeutet ein abnormales Server-Ende zum Teil erhebliche Wiederanlaufzeiten. Ursächlich ist der Grund in der Ausführung bestimmter Server-Tasks zu finden, die in einem solchen Fall ausgeführt werden müssen. Der Fixup-Task war bisher zwingend nach einem abnormalem Server-Ende auszuführen.

Führen wir uns die Funktion des Fixup-Tasks einmal kurz vor Augen: Nach einem Server-Abbruch wird jede auf dem Server befindliche Datenbank durchlaufen und jedes in diesen Datenbanken enthaltene Dokument verarbeitet. Bei der Verwendung der Transaktionsprotokollierung müssen nicht alle Dokumente aller Datenbanken verarbeitet werden, vielmehr wird analysiert, welche Datenbank-Änderungen zum Zeitpunkt des Server-Abbruchs noch nicht in der Datenbank weggeschrieben wurden. Diese nichtverarbeiteten Änderungen werden dann bei einem Server-Neustart in die Datenbanken geschrieben.

Durch die seit der Version 6 zur Verfügung stehenden Protokollierung auf Ansichts-/Ordnersebene können die Wiederanlaufzeiten in diesem Bereich zum Teil drastisch verkürzt werden. Doch dazu später mehr.

Möglichkeit inkrementeller Datensicherung

Datensicherungen im Notes/Domino-Umfeld waren in der Vergangenheit meist Komplettsicherungen und bei großen Datenmengen oft ein Problem. Mit der Einführung der Transaktionsprotokollierung können nun auch so genannte inkrementelle Datensicherungen durchgeführt werden. Dabei werden – basierend auf einer „Gesamtsicherung“ – an Folgetagen nur noch geänderte Daten gesichert. Gerade in Verbindung mit Domino auf der iSeries wirbt IBM mit der „neuen“ Möglichkeit der inkrementellen Datensicherung von Domino in Verbindung mit BRMS (Backup und Recovery Media Systems). Leider fehlt in einigen Quellen der Hinweis, dass für diese

Art der Datensicherung auf dem Domino-Server die Transaktionsprotokollierung aktiviert sein muss!

Zudem sind in Notes/Domino bzw. in Tools, die mit der Transaktionsprotokollierung zur Verfügung gestellt werden, keine Wiederherstellungswerkzeuge enthalten, mit denen in einem Schadensfall ein Wiederaufbau der Informationen möglich ist. Für solche Wiederherstellungs-Senarien sind entsprechende Produkte (wie zum Beispiel Tivoli/BRMS) erforderlich.

Grundsätzlich ist die Transaktionsprotokollierung bei einer Neuinstallation oder auch einer Migration deaktiviert. Eine Aktivierung kann in dem Server-Dokument des jeweiligen Domino-Servers erfolgen. Macht man es sich recht einfach, dann klickt man sich „schnell“ durch und „schon“ läuft die Transaktionsprotokollierung – dann aber oft mit dem einen oder anderen negativen Effekt.

Denn so einfach die Transaktionsprotokollierung einzurichten ist, so falsch kann deren Einrichtung auch sein. IBM hat zwar einen wirklich einfachen Installations- bzw. Aktivierungsprozess implementiert, weist aber zu wenig auf Konsequenzen oder eben Punkte hin, deren Beachtung immens wichtig sein können, was sich unter anderem aus Ausführungsgeschwindigkeits-Gesichtspunkten stark bemerkbar macht!

Deshalb gilt:

Richten Sie in jedem Fall die Transaktionsprotokollierung so ein, dass sich das Protokoll auf einer anderen Platte als der des Domino-Servers befindet – besser noch: auf einer anderen Platte, die von einem anderen Platten-Controller bedient wird. Damit kann bei Engpässen im Schreib-/Lesebereich der Platten eine Entlastung erreicht werden, die zum Teil erheblichen Einfluss auf die Gesamtleistung haben kann!

Doch beschäftigen wir uns zunächst mit den Grundlagen der Transaktionsprotokollierung:

Was ist eine Transaktion?

Um die Arbeitsweise der Transaktionsprotokollierung verstehen zu können, schauen wir uns zunächst an, was man unter einer Transaktion versteht:

Eine einzelne Transaktion beinhaltet eine Reihe von Aktionen, die an einer Datenbank durchgeführt werden. Beispiele für solche Transaktionen können sein:

- Öffnen eines Dokuments
- Lesen eines Dokuments
- Hinzufügen von Informationen in einem Dokument
- Verändern von Informationen in einem Dokument
- Speichern eines Dokuments

IBM hat in einigen Publikationen den Begriff der „erfolgreichen Transaktion“ genannt. Darunter versteht man unter R5 das Ausführen der Transaktionen, wie sie in der vorhergehenden Aufzählung enthalten sind, bis zu ihrem endgültigen Schreiben auf die Platte. Unter Domino 6 kann dieser Schreibprozess ausgedehnt werden – quasi bis zum Schreiben der Informationen auf Platte aus dem Transaktionsprotokoll heraus.

Wie arbeitet die Transaktionsprotokollierung?

Wie bereits beschrieben, sollte das Transaktionsprotokoll auf einem anderen Plattenbereich (physisch!) geführt werden als der Domino-Server (bzw. als andere schreib-/leseintensiven Anwendungen). Der Plattenplatz, der zusätzlich für die Protokollierung erforderlich ist, sollte ausreichend für die Aufzeichnungen der letzten zwei Tage sein. In der Praxis hat sich ein Plattenplatz in Höhe von 25 Prozent des für die Domino-Installation belegten Plattenplatzes bewährt. Dies hängt allerdings immer stark von den individuellen Gegebenheiten ab.

- Indikator

Zunächst benötigt jede Datenbank einen eindeutigen Indikator. Dieser wird bei der Transaktionsprotokollierung verwendet, um die Datenbank eindeutig identifizieren zu können. Dieser Indikator – eine eindeutige Datenbankinstanz-ID (DBIID) – wird jeder Datenbank bei einem Server-Neustart neu zugewiesen. Voraussetzung dafür ist die ODS-Struktur ab R5 (ODS41 – unter Domino 6 erstellte Datenbanken haben die ODS43-Struktur).

Diese DBIID wird bei jeder Transaktion, die in dem Transaktionsprotokoll aufgezeichnet wird, mit weggeschrieben und dient als Basis für die Zuordnung dieser Transaktion zur Datenbank. Aus diesem Grund ist die Eindeutigkeit der DBIID so immens wichtig.

Leider:

Einige Aktionen im Bereich der Administration von Domino/Notes-Datenbanken bewirken, dass die DBIID neu vergeben wird. Somit wird ein „Umhängen“ des Protokolls auf die neue DBIID erforderlich und diese wird dann im weiteren Verlauf automatisch für die Aufzeichnungen verwendet. Ein Beispiel für eine solche Datenbankaktion ist die Komprimierung einer Datenbank.

Problematisch in diesem Zusammenhang:

Sollte nach einer Neuerstellung einer DBIID ein Wiederanlauf unter Verwendung des Transaktionsprotokolls durchgeführt werden, dann ist eine Wiederherstellung der Informationen, die mit der alten DBIID protokolliert worden sind, nicht mehr möglich! Hier bleibt zu hoffen, dass IBM in diesem Bereich in den nächsten Release-Versionen Abhilfe schafft.

Nachfolgend noch einige Datenbankaktionen, bei denen die DBIID neu vergeben wird:

- Komprimierung einer Datenbank (mit Optionen) ☐ bereits oben beschrieben
- Erstmaliges Aktivieren der Transaktionsprotokollierung
- Ausführen des Server-Tasks „Compact“
- Ausführen des „Fixup“-Tasks
- Ändern des Pfads oder der Größe des Transaktionsprotokolls
- Verschieben einer Datenbank mit ODS41 oder höher von einem Server auf einen anderen

- Transaktionsprotokolldatei

Die Aufzeichnung der Änderungen wird in einer Protokolldatei vorgenommen. Diese Datei – genau genommen sind es mehrerer Dateien, die nach bestimmten Regeln erstellt werden – haben die Endungen „.txn“ und werden bei einem Server-Neustart formatiert und bei Bedarf initialisiert. Zusätzlich wird bei einem Server-Neustart eine Steuerdatei angelegt, in der die wesentlichen Elemente und Einstellungen automatisch eingetragen werden. Diese Steuerdatei trägt die Bezeichnung „nlogctrl.lfh“.

Betrachten wir nun den Ablauf der Transaktionsprotokollierung an einem kleinen Beispiel:

Ein Benutzer arbeitet mit einem Notes Client in seinem e-Mail-Postkorb. Dabei liest, erstellt und verändert er unterschiedliche e-Mails.

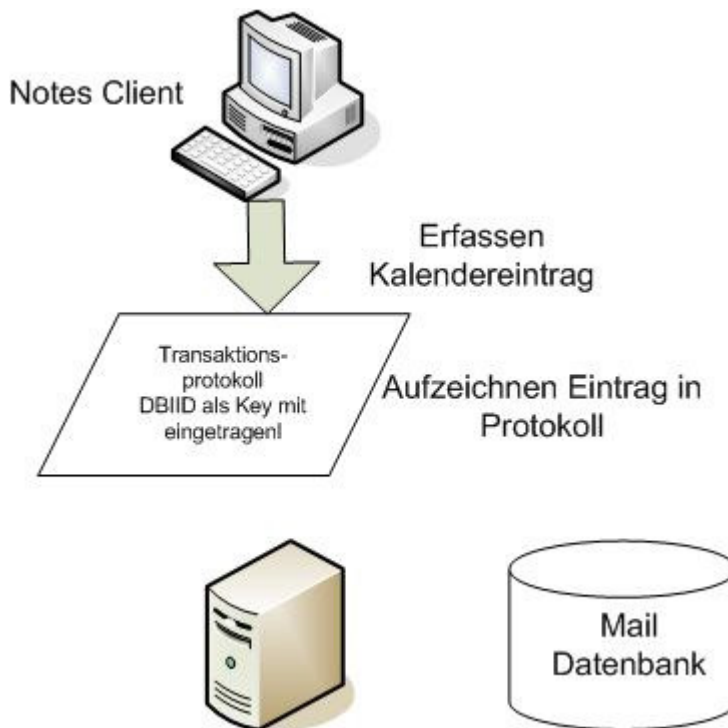
Beispielszenario:

Bei einem solchen Verarbeitungsvorgang mit einer e-Mail werden aus Sicht der Transaktionsprotokollierung folgende Tätigkeiten durchgeführt:

- Aufzeichnen der e-Mail Informationen in einem Transaktionsprotokoll
- Speichern der Informationen als Datenbank-Aktualisierung

Der Benutzer trägt nun einen Termin in seinen Notes-Kalender ein (dieser befindet sich im Notes-Standard ebenfalls in der Mail-Datenbank).

Der Kalendereintrag wird ebenfalls in dem Transaktionsprotokoll aufgezeichnet. Eine Speicherung des Termins in die Mail-Datenbank erfolgt noch nicht, denn der Server ist zu diesem Zeitpunkt stark ausgelastet. Deshalb werden die Leistungsanforderungen des Servers verteilt. Die Belastung durch die Schreib-/Leseprozesse wird auf einen späteren Zeitpunkt mit geringerer Server-Belastung verlagert, um dann automatisch gestartet zu werden.



002 Beispiel 1

Nun fällt der Server aus, ohne die Informationen des neuen Kalendereintrages in die Mail-Datenbank geschrieben zu haben.

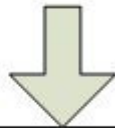
Mit Hilfe der im Transaktionsprotokoll aufgezeichneten Daten kann anhand der DBIID die Zugehörigkeit dieses Eintrages zur Mail-Datenbank ermittelt werden. Diese Information wird nun bei einem Server-Start für das Fortschreiben der offenen Aktionen aus dem Transaktionsprotokoll in die Mail-Datenbank verwendet. Für diese Datenbank sind die nach einem Server-Crash erforderlichen, zum Teil recht lange laufenden Prozesse nicht notwendig.

Nach dem Server-Neustart findet der Benutzer seinen Kalendereintrag in seiner Mail-Datenbank vor.

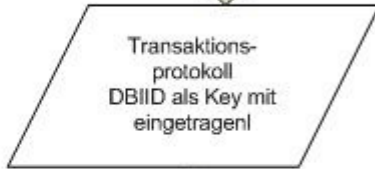
Serverneustart



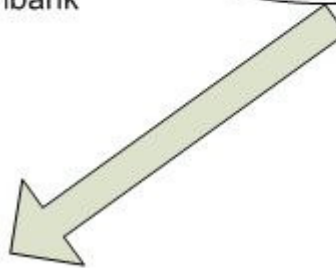
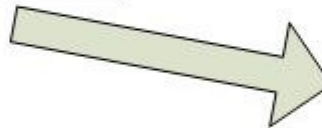
Verarbeiten
Transaktionsprotokoll
offene Einträge



Transaktions-
protokoll
DBIID als Key mit
eingetragen



Update in Domino
Datenbank



Client hat aktuelle Daten

003 Beispiel Teil 2

Diese grundsätzlichen Informationen zur Transaktionsprotokollierung werden wir in der nächsten Ausgabe vertiefen. Dann werde ich Ihnen zeigen, wie man mit bestimmten Einstellungen die optimale Leistung mit der Transaktionsprotokollierung erzielen kann.

Den Autor Jörg Zeig erreichen Sie unter ZeBIS
Beratung/Informationssysteme - Im Bühlsgarten 3 - 57223 Kreuztal - Tel.
(+49) 2732/892491 - e-Mail: HJZeig@ZeBIS.de