



# **Firebirds nbackup-Werkzeug**

Paul Vinkenoog, Mark Rotteveel, Alexey Kovyazin, Thomas Steinmaurer, Frank  
Ingermann, Martin Köditz

Version 1.6-de, 25. Juli 2020

# Inhaltsverzeichnis

|   |    |
|---|----|
| 1. Einleitung .....   | 2  |
| 2. Nbackup Funktionalität - Ein Überblick .....                                       | 3  |
| 2.1. Vorteile von nbackup .....   | 3  |
| 2.2. Einschränkungen von nbackup .....  | 3  |
| 3. Funktionen und Parameter .....   | 5  |
| 4. Backups erstellen und wiederherstellen .....                                       | 7  |
| 4.1. Vollständige Sicherungen .....   | 7  |
| 4.1.1. Vollständige Sicherung erstellen .....   | 7  |
| 4.1.2. Wiederherstellen einer vollständigen Sicherung .....                           | 9  |
| 4.2. Inkrementelle Backups .....  | 10 |
| 4.2.1. Inkrementelle Sicherungen erstellen .....                                      | 10 |
| 4.2.2. Wichtiger Leistungshinweis .....   | 11 |
| 4.2.3. Wiederherstellen von inkrementellen Sicherungen .....                          | 11 |
| 4.3. Sichern von Rohgerätedatenbanken .....   | 12 |
| 4.4. Datenbanktrigger unterdrücken (Firebird 2.1+) .....                              | 13 |
| 4.5. Direkte I/O (Firebird 2.1.4+) .....  | 13 |
| 4.6. Informationsoptionen (Firebird 2.5+) .....                                       | 13 |
| 4.7. Sicherungen auf Remote-Servern (Firebird 2.5+) .....                             | 14 |
| 4.8. Eine praktische Anwendung .....  | 16 |
| 4.9. Sollten Sie weiterlesen? .....   | 16 |
| 5. Sperren und entsperren .....   | 17 |
| 5.1. Sperren der Datenbank und Sicherung selbst erstellen .....                       | 17 |
| 5.2. Wiederherstellen eines Backups nach "nbackup -L" .....                           | 18 |
| 5.3. Internas .....   | 19 |
| 5.4. Sperren von Rohgerätedatenbanken .....   | 19 |
| 5.5. Sperren / Entsperrern der Datenbank bei der Sicherung virtueller Maschinen ..... | 20 |
| 6. Delta-Datei einstellen .....   | 21 |
| 7. Sicherungsverlauf .....  | 22 |
| 8. Technische Hintergrundinformationen .....  | 23 |
| Anhang A: Dokumentenhistorie .....  | 24 |
| Anhang B: Lizenzhinweise .....  | 27 |

# Kapitel 1. Einleitung

*Nbackup* ist ein Backup-Werkzeug, das seit Firebird 2.0 inkludiert ist. Es bietet Möglichkeiten, die in dem bekannten Firebird Backup-Tool *gbak* — bereits in Firebird vorhandenes Backup-Tool — nicht vorhanden sind. Es ersetzt jedoch nicht letzteres. Beide Programme haben ihre Stärken und Schwächen und werden sich noch für einige Zeit gegenseitig ergänzen.

# Kapitel 2. Nbackup Funktionalität - Ein Überblick

Mit nbackup können Sie zwei unterschiedliche Arten von Aufgaben durchführen:

1. Erstellung und Wiederherstellung von vollständigen und *inkrementellen* Sicherungen. Eine *inkrementelle Sicherung* beinhaltet nur Änderungen, die nach dem Zeitpunkt einer vorangegangenen Sicherung durchgeführt worden sind. Dies kann nur mit dem Firebird Befehlszeilenwerkzeug nbackup durchgeführt werden.
2. Sperren der Hauptdatenbankdatei, damit Sie sie anschließend selbst mit Kopier- oder Sicherungswerkzeugen Ihrer Wahl sichern können. In diesem Modus sichert nbackup nichts. Es werden nur die Bedingungen erstellt, unter denen Sie das Backup sicher selbst erstellen können. Auch hier ist eine Wiederherstellung vorgesehen. Das Sperren ist erforderlich, um gültige Sicherungen von Firebird-Datenbanken in virtuellen Maschinen durchzuführen: Ohne Sperren einer Datenbank vor dem Sichern der VM kann die Datenbankdatei in der VM-Sicherung inkonsistent sein. Siehe auch [Sperren / Entsperrern der Datenbank bei der Sicherung virtueller Maschinen](#).

Beide Modi können in einer aktiven Datenbank ausgeführt werden, ohne verbundene Benutzer zu behindern. Die erstellte Sicherung spiegelt immer den Status der Datenbank *zu Beginn der Operation* wider. Darüber hinaus können nur "SYSDBA" und der Datenbankeigentümer (und manchmal auch Betriebssystemadministratoren) eine Sicherung erstellen, aber jeder Benutzer kann eine Sicherung in einer neuen Datenbank wiederherstellen. In dieser Hinsicht unterscheidet sich nbackup nicht von gbak.

## 2.1. Vorteile von nbackup

- *Beide Modi*: hohe Geschwindigkeit (so hoch wie es Hardware und Betriebssystem zulassen), da nbackup die tatsächlichen Daten nicht berücksichtigt. Im Sicherungsmodus wird der Inhalt mehr oder weniger blind in die Sicherungsdatei geschrieben.
- *Backup/Restore-Modus*: Zeit- und Speicherplatzersparnis, da Sie nicht jedes Mal eine vollständige Sicherung durchführen müssen. Dies kann bei Datenbanken im Gigabyte-Bereich einen großen Unterschied machen.
- *Lock/Unlock-Modus*: Volle Freiheit bei der Auswahl von Sicherungs-, Kopier- und / oder Komprimierungswerkzeugen.

## 2.2. Einschränkungen von nbackup

- nbackup wird kein Sweep und auch keine Verkleinerung Ihrer Datenbank durchführen, so wie das bei gbak im Falle einer Sicherung mit anschließender Wiederherstellung ist.
- Sie können mit einem nbackup-Sicherungs/Wiederherstellungszyklus nicht den Eigentümer einer Datenbank ändern, so wie das mit gbak möglich ist.
- Nbackup kann keine *transportablen Sicherungen* erstellen, d.h. Sicherungen, die auf einer nicht-kompatiblen Plattform oder einer unterschiedlichen Serverversion wiederhergestellt werden

können.

- Zu diesem Zeitpunkt sollte nbackup nicht auf Mehrdateiendatenbanken ausgeführt werden. In Firebird 2.5 und höher können die Sicherungs- und Wiederherstellungsaufgaben jedoch auch remote über den Services Manager ausgeführt werden.
- Wird der Services Manager nicht verwendet (ab Firebird 2.5+), erfordert das Sichern mit nbackup den direkten Zugriff auf die Datenbankdatei. Bei gbak ist dies nicht der Fall.

## Kapitel 3. Funktionen und Parameter

Die folgende Tabelle gibt einen Überblick über die Parameter von nbackup. Wenn das Feld “Hinzugefügt” leer ist, ist der Parameter seit der Einführung von nbackup in Firebird 2.0.f vorhanden.

Tabelle 1. Nbackup-Parameter

| Parameter                  | Funktion   | Hinzugefügt |
|----------------------------|--|-------------|
| -B n database [filename]   | Erstellen von Level- <i>n</i> -Backups der Datenbank in eine Datei |             |
| -R database [filename ...] | Datenbank aus Sicherungsdatei (en) wiederherstellen                |             |
| -L database                | Datenbank sperren  |             |
| -N database                | Gesperrte Datenbank entsperren                                     |             |
| -F database                | Entsperren Sie die vom Benutzer wiederhergestellte Datenbank       |             |
| -S                         | Geben Sie die Größe der Datenbankseiten an (mit -L)                | 2.1         |
| -T                         | Datenbanktrigger unterdrücken (mit -B, -L, -N)                     | 2.1         |
| -D {on   off}              | Direkte I/O an/aus (mit -B)  | 2.1.4       |
| -U username                | Geben Sie den Benutzernamen an (mit -B, -L, -N)                    |             |
| -P password                | Passwort eingeben (mit -B, -L, -N)                                 |             |
| -FE filename               | Passwort aus Datei abrufen (mit -B, -L, -N)                        | 2.5         |
| -Z                         | Versionsinfo (allein oder mit -B, -R, -L, -N, -F)                  | 2.5         |
| -?                         | Hilfe (schaltet alle anderen Parameter aus)                        | 2.5         |

Abhängig von der gewählten Hauptfunktion (-B,-R, -L,-N oder -F) erfordert nbackup möglicherweise verschiedene Arten des Zugriffs auf die Datenbank: eine Firebird-Serververbindung, direkter Dateizugriff oder beides. Die folgende Tabelle enthält die Details:

Tabelle 2. Zugang erforderlich

| Parameter | Funktion          | Zugriff        |
|-----------|-------------------|----------------|
| -B        | Datensicherung    | Server + Datei |
| -R        | Wiederherstellung | Datei          |
| -L        | Sperren           | Server         |

| Parameter | Funktion   | Zugriff |
|-----------|--|---------|
| -N        | Entsperren (rückgängig -L)                           | Server  |
| -F        | Nach Wiederherstellung durch den Benutzer entsperren | Datei   |

Wenn Serverzugriff erforderlich ist (mit -B, -L und -N), muss der Benutzer entweder einen Firebird-Benutzernamen und ein Kennwort angeben (mit -U und -P/-FE oder über die Umgebungsvariablen ISC\_USER und ISC\_PASSWORD) oder vom Server aus anderen Gründen zugelassen werden (z. B. als root unter Posix oder durch vertrauenswürdige Authentifizierung unter Windows).

Wenn Dateisystemzugriff erforderlich ist (mit -B, -R und -F), muss der Benutzer über ausreichende Lese- und / oder Schreibrechte für die Datenbankdatei verfügen.

Wenn ausschließlich Dateisystemzugriff erforderlich ist (mit -R und -F), muss der Benutzer nicht über ein Firebird-Login verfügen und ein laufender Firebird-Server muss nicht vorhanden sein.

Bitte beachten Sie: Die obige Tabelle und der Text betreffen den Zugriff auf die *Datenbank*. Der Zugriff auf die Sicherungsdatei erfolgt — natürlich — immer auf Dateisystemebene.

# Kapitel 4. Backups erstellen und wiederherstellen

Um zu beginnen: nbackup.exe befindet sich im Unterverzeichnis bin Ihrer Firebird-Installation. Typische Speicherorte, zum Beispiel, sind: C:\Programme\Firebird\Firebird\_3\_0 oder C:\Programme\Firebird\Firebird\_2\_0\bin (Windows) oder /opt/firebird/bin (Linux). So wie bei den meisten Tools, die mit Firebird mitkommen, besitzt auch nbackup keine graphische Benutzeroberfläche. Sie müssen nbackup von der Kommandozeile aus starten (oder von einer Batchdatei oder einer Anwendung).



In einigen Umgebungen kann nbackup 2.0.3 und niedriger unter Umständen mit hoher Last Probleme verursachen, die zu Deadlocks oder sogar zu beschädigten Datenbanken führen. Obwohl diese Probleme nicht häufig auftreten, sind sie schwerwiegend genug, um ein Upgrade auf Firebird 2.0.4 oder höher zu rechtfertigen, wenn Sie nbackup bequem verwenden möchten. Wenn es sich um große Datenbanken unter Posix handelt, kann auch die Verwendung von direkter I/O einen Unterschied machen. Mehr dazu im Abschnitt *Direkter I/O*.

## 4.1. Vollständige Sicherungen

### 4.1.1. Vollständige Sicherung erstellen

Um eine vollständige Datenbanksicherung durchzuführen, verwenden Sie das folgende Kommando:

```
nbackup [-U user -P password] -B 0 database [backupfile]
```

Angenommen, die Datenbank befindet sich in C:\Data und nbackup.exe befindet sich im Windows-Suchpfad:

```
C:\Data> nbackup -B 0 inventory.fdb inventory_1-Mar-2006.nbk
```

Oder, wenn Firebird (ab Version 2.5) zum Beispiel unter einem anderem Port als dem Standard läuft:

```
C:\Data>nbackup -B 0 localhost/3051:C:\Data\inventory.fdb C:\Data\inventory-level-0-Jul-2020.nbk -user SYSDBA -pass masterkey
```

In Firebird 3.0 und höher druckt nbackup bei erfolgreichem Abschluss der Sicherung gekürzte Statistiken:



```
C:\Data> nbackup -B 0 inventory.fdb inventory_1-Mar-2006.nbk
time elapsed    0 sec
page reads     307
page writes    307
```

#### Anmerkungen:

- Der Parameter `-B` steht für Backup (Sicherung). Das *Backup-Level* 0 steht für eine vollständige Sicherung. Sicherungslevel größer als 0, werden für inkrementelle Sicherungen verwendet. Dies wird später behandelt.
- Anstatt eines Datenbankdateinamens können Sie auch einen Alias verwenden.
- Anstatt eines Dateinamens für die Sicherung können Sie auch `stdout` angeben. Dies wird die Sicherung an die Standardausgabe übergeben, von wo Sie die Sicherung an eine Bandarchivierung oder an ein Komprimierungstool weiterleiten können.
- Die Parameter `-U` (Benutzer) und `-P` (Kennwort) können weggelassen werden, wenn mindestens eine der folgenden Bedingungen erfüllt ist:
  - Die Umgebungsvariablen `ISC_USER` und `ISC_PASSWORD` wurden entweder auf `SYSDBA` oder auf den Eigentümer der Datenbank gesetzt.
  - Sie sind auf einem Posix-System als `root` angemeldet. Dies macht Sie standardmäßig zu `SYSDBA`.
  - Unter Windows: Die vertrauenswürdige Authentifizierung ist in der Datei `firebird.conf` aktiviert, und Sie sind bei dem Windows-Konto angemeldet, dem die Datenbank gehört. Dies ist in Firebird 2.1 und höher möglich.
  - Unter Windows: Die vertrauenswürdige Authentifizierung ist in der Datei `firebird.conf` aktiviert, und Sie sind als Windows-Administrator angemeldet. In Firebird 2.1 erhalten Sie automatisch `SYSDBA`-Rechte. In Firebird 2.5 und höher gibt es die zusätzliche Bedingung, dass `AUTO ADMIN MAPPING` in der Datenbank festgelegt wurde.

Aus Gründen der Klarheit und Kürze werden die Parameter `-U` und `-P` in den Beispielen nicht verwendet.

- Ab Firebird 2.5 können Sie anstelle von `-P` Kennwort auch `-FE` Dateiname verwenden. Dies führt dazu, dass `nbackup` das Kennwort aus der angegebenen Datei abrufen. Mit `-FE` erscheint das Passwort selbst nicht im Befehl und ist daher besser gegen Personen geschützt, die es andernfalls über den Befehlsverlauf, den Befehl `w` unter Unix oder aus einem Skript oder einer Batchdatei abrufen könnten.
- In Firebird 2.1 und höher kann das Auslösen von Datenbanktriggern verhindert werden, indem die Option `-T` angegeben wird. Für weitere Informationen, siehe auch [Datenbanktrigger unterdrücken](#).
- Ab Firebird 2.1.4 ist es möglich, die direkte I/O durch Angabe von `"-D on"` oder `"-D off"` zu aktivieren oder zu deaktivieren. Für Details und Hintergrundinfos, siehe [Direct I/O](#) in diesem Handbuch.
- Die verschiedenen Parameter (`-B`, `-U` usw.) können in beliebiger Reihenfolge auftreten. Natürlich

sollte jedem Parameter sofort ein eigenes Argument folgen. Im Fall von `-B` gibt es drei davon: Sicherungsstufe, Datenbank und Sicherungsdatei — in dieser Reihenfolge!

- Wenn der Parameter `-B` als letzter kommt, *können* Sie den Namen der Sicherungsdatei eglassen. In diesem Fall erstellt `nbackup` einen Dateinamen basierend auf dem Datenbanknamen, der Sicherungsstufe sowie dem aktuellen Datum und der aktuellen Uhrzeit. Dies kann zu einem Namenskonflikt (und einer fehlgeschlagenen Sicherung) führen, wenn zwei Sicherungsbefehle derselben Ebene in derselben Minute ausgegeben werden.



Verwenden Sie `nbackup` *nicht* für Datenbanken mit mehreren Dateien. Dies kann zu Beschädigung und Datenverlust führen, obwohl sich `nbackup` nicht über einen solchen Befehl beschwert.

### Einige Anmerkungen zu den Internas

Hinweis: Was hier folgt, ist nicht notwendigerweise ein Wissen, das Sie für die Verwendung von `nbackup` benötigen. Dieser kurze Abschnitt gibt nur einen groben und unvollständigen Überblick, was intern bei der Ausführung von `nbackup -B` geschieht:

1. Zuallererst wird die Hauptdatenbankdatei durch das Ändern eines internen Flags gesperrt. Von diesem Zeitpunkt an werden alle Änderungen in der Datenbank in eine temporäre Datei geschrieben, die auch als Differenzdatei oder *Deltadatei* bezeichnet wird. Standardmäßig wird die Delta-Datei im selben Ordner wie eine Datenbankdatei mit der zusätzlichen Erweiterung `.delta` erstellt, z. B. `MyDatabase.fdb.delta`
2. Danach wird die Sicherung durchgeführt. Bei der daraus resultierenden Sicherungsdatei handelt es sich nicht um eine reguläre Dateikopie, sondern eine Wiederherstellung muss ebenfalls mit `nbackup` durchgeführt werden.
3. Nach der Fertigstellung der Sicherung wird der Inhalt der Deltadatei mit der Hauptdatenbankdatei zusammengeführt. Im Anschluss daran wird die Sperre auf der Datenbankdatei wieder entfernt (das Flag wird wieder auf “normal” zurückgesetzt) und die Deltadatei wird gelöscht.

Die Funktionalität von Schritt 1 und 3 ist über zwei neue SQL-Anweisungen verfügbar: `ALTER DATABASE BEGIN BACKUP` und `ALTER DATABASE END BACKUP`. Entgegen den verwendeten Bezeichnungen führen diese Anweisungen nicht die eigentliche Sicherung durch, sondern sie stellen den Zustand her, unter dem die Hauptdatenbankdatei auf sichere Art und Weise gesichert werden kann. Um dies klar zu stellen: Sie brauchen diese Anweisungen nicht selbst ausführen, sondern `nbackup` wird das zum richtigen Zeitpunkt für Sie durchführen.

#### 4.1.2. Wiederherstellen einer vollständigen Sicherung

Eine vollständige Sicherung wird wie folgt wiederhergestellt:

```
nbackup -R database [backupfile]
```

Zum Beispiel:

```
C:\Data> nbackup -R inventory.fdb inventory_1-Mar-2006.nbk
```

Anmerkungen:

- Sie geben keinen Level bei der Wiederherstellung an.
- Der Parameter -R muss zum Schluß angegeben werden. Die Gründe dafür werden später klarer.
- Anstelle eines Datenbankdateinamens können Sie auch einen Alias verwenden.
- Falls die angegebene Datenbank bereits existiert und es keine Verbindungen zu dieser Datenbank gibt, dann wird die Datenbank ohne Warnung überschrieben! Falls Benutzer mit der Datenbank verbunden sind, dann wird die Wiederherstellung fehlschlagen und Sie bekommen eine Fehlermeldung angezeigt.
- Auch hier können Sie den Namen der Sicherungsdatei weglassen. Wenn Sie dies tun, werden Sie von nbackup zur Eingabe eines Dateinamens auffordern. (*Achtung! In Firebird 2.0.0 ist diese Funktion "Interaktive Wiederherstellung" fehlerhaft, sodass eine Fehlermeldung und eine fehlgeschlagene Wiederherstellung angezeigt werden. In 2.0.1 behoben.*)
- Das Wiederherstellen funktioniert ausschließlich auf Dateisystemebene und kann sogar ohne laufenden Firebird-Server durchgeführt werden. Alle über die Parameter -U und -P angegebenen Anmeldeinformationen werden ignoriert. Gleiches gilt für Passwörter, die aus einer Datei gelesen werden. Nbackup *versucht* jedoch, das Kennwort aus der Datei zu lesen, wenn der Parameter -FE vorhanden ist. Wenn ein Fehler auftritt, wird der gesamte Vorgang abgebrochen.

## 4.2. Inkrementelle Backups



Die inkrementelle Sicherungsfunktion wurde in Firebird 2.1.0 vollständig beschädigt und in 2.1.1 erneut behoben.

### 4.2.1. Inkrementelle Sicherungen erstellen

Um eine inkrementelle ("differenzielle") Sicherung durchzuführen, geben wir eine Sicherungsstufe größer als 0 an. Eine inkrementelle Sicherung der Stufe N enthält immer die Datenbankmutationen seit der letzten Sicherung der Stufe N-1.

Beispiele:

Einen Tag nach der vollständigen Sicherung (Stufe 0) erstellen Sie eine mit Stufe 1:

```
C:\Data> nbackup -B 1 inventory.fdb inventory_2-Mar-2006.nbk
```

Diese Sicherung enthält nur die Mutationen des letzten Tages.

Einen Tag später erstellen Sie eine weitere mit Level 1:

```
C:\Data> nbackup -B 1 inventory.fdb inventory_3-Mar-2006.nbk
```

Diese enthält die Mutationen der letzten zwei Tage seit der vollständigen Sicherung, nicht nur die seit der vorherigen Sicherung der Stufe 1.



Die vorherige inkrementelle Sicherung einer beliebigen Ebene muss vor dem Start der nächsten inkrementellen Sicherung abgeschlossen sein. Andernfalls führt nbackup nicht die gewünschte Sicherung durch und gibt den Fehler "Datenbank befindet sich bereits im physischen Sicherungsmodus" (engl. "Database is already in the physical backup mode") zurück.

Ein paar Stunden später erstellen wir ein Level-2-Backup:

```
C:\Data> nbackup -B 2 inventory.fdb inventory_3-Mar-2006_2.nbk
```

Dieses jüngste Backup enthält nur die Mutationen seit dem letzten Level-1-Backup, d.h. der letzten Stunden.

#### 4.2.2. Wichtiger Leistungshinweis

Vor Firebird 3.0 lesen alle inkrementellen Sicherungen (d.h. Stufe 1, 2 usw.) die gesamte Datenbankdatei, um die Änderungen zu lokalisieren und in einer Sicherungsdatei zu speichern. Dies kann die Datenbankleistung bei großen Datenbanken (100GB+) und langsamen Festplattensubsystemen beeinträchtigen (normalerweise kann bei Nicht-SSD-Laufwerken ein Geschwindigkeitseinbruch festgestellt werden). Ab Firebird 3.0 wird nur der geänderte Teil der Datenbankdatei gelesen, sodass inkrementelle Sicherungen der Stufe 1 und höher viel schneller sind und die Datenbankleistung weniger beeinträchtigen.



Alle [Anmerkungen](#), die eine vollständige Sicherung betreffen, treffen auch für inkrementelle Sicherungen zu.



Noch einmal: Verwenden Sie nbackup nicht für Mehrdateidatenbanken.

#### 4.2.3. Wiederherstellen von inkrementellen Sicherungen

Wenn Sie inkrementelle Sicherungen wiederherstellen wollen, dann müssen Sie die vollständige Kette der inkrementellen Sicherungsdateien angeben, beginnend mit der Level 0 Sicherung, bis zu dem Zeitpunkt, für den Sie eine Wiederherstellung wünschen. Die Datenbank wird immer von Grund auf erstellt; Schritt für Schritt. (Es handelt sich hier um ein schrittweises Hinzufügen der *inkrementellen Sicherungen*.)

Die formale Syntax dafür ist:

```
nbackup -R database [backup0 [backup1 [...] ] ]
```

Somit sieht die Wiederherstellung der Level-2 Sicherung aus unserem vorherigen Beispiel wie folgt aus:

```
C:\Data> nbackup -R inventory.fdb inventory_1-Mar-2006.nbk
inventory_3-Mar-2006.nbk inventory_3-Mar-2006_2.nbk
```

Natürlich wurde die Zeile nur zum Zwecke der besseren Lesbarkeit umgebrochen. In Wirklichkeit geben Sie das vollständige Kommando in einer Zeile ein und drücken am Schluß die [Enter]-Taste.

Anmerkungen (zusätzlich zu den [Anmerkungen bei der Wiederherstellung einer vollständigen Sicherung](#)):

- Da im Voraus nicht bekannt ist, wie viele Dateinamen dem -R Schalter nachfolgen (da wir keinen Level bei der Wiederherstellung angeben), wird nbackup alle Argumente nach -R als Namen der Sicherungsdateien ansehen. Das ist der Grund, warum keine anderen Parameter (-U oder -P) an die Liste der Dateinamen angehängt werden sollen.
- Es existiert kein formales Limit für die Anzahl der Backup-Levels, aber unter praktischen Gesichtspunkten, wird es kaum Sinn machen, dass man über 3 oder 4 hinaus geht.

### Nichtverbundene Aneinanderreihung

Was passiert, wenn Sie versehentlich eine Sicherungsdatei vergessen anzugeben, oder wenn Sie eine Serie an Dateien angeben, die keinen Bezug zueinander aufweisen? Stellen Sie sich vor, dass Sie im obigen Beispiel irrtümlich `inventory_2-Mar-2006.nbk` anstatt von `inventory_3-Mar-2006.nbk` verwenden. Beide sind Level-1 Sicherungsdateien, somit bekommen wir eine schöne Aneinanderreihung von "0, 1, 2"-Level Sicherungen. Unsere Level-2 Datei ist eine inkrementelle Sicherung von der Level-1 Sicherung vom 3. März, und nicht von der Sicherung vom 2. März.

Glücklicherweise kann so ein Versehen nie zu einer fehlerhaft zurückgesicherten Datenbank führen. Jede Sicherungsdatei hat ihre eigene eindeutige ID. Des Weiteren besitzt jede Level-1 (oder höher) Sicherungsdatei die ID der Sicherung, von der sie abstammt. Bei der Wiederherstellung überprüft nbackup zunächst die IDs daraufhin, ob es eine Aneinanderreihung gibt, die nicht zusammengehört. Ist dies der Fall, dann wird die Operation abgebrochen und Sie bekommen eine Fehlermeldung.

## 4.3. Sichern von Rohgerätedatenbanken

Firebird-Datenbanken müssen keine Dateien sein. Sie können auch auf einem sogenannten *Rohgerät* (engl. *raw device*) platziert werden, beispielsweise einer Festplattenpartition ohne Dateisystem. Die Frage, wo in solchen Fällen das `delta` platziert werden muss, wurde bei der Entwicklung von nbackup zunächst übersehen. Wenn sich die Datenbank auf Posix-Systemen z.B. auf `/dev/hdb5` befindet, kann es vorkommen, dass das Delta als `/dev/hdb5.delta` erstellt wurde. In Anbetracht der Art und des Zwecks des Verzeichnisses `/dev` und seines oft begrenzt verfügbaren Speicherplatzes ist dies unerwünscht.

Ab Firebird 2.1 weigert sich nbackup, mit Rohgerätedatenbanken zu arbeiten, es sei denn, ein expliziter Speicherort für die Delta-Datei wurde festgelegt. Die Vorgehensweise wird in [Delta-Datei](#)

[einstellen](#) weiter unten in diesem Handbuch erläutert.

## 4.4. Datenbanktrigger unterdrücken (Firebird 2.1+)

Firebird 2.1 führte das Konzept von *Datenbanktriggern* ein. Bestimmte Arten dieser Trigger können beim Herstellen oder Unterbrechen einer Datenbankverbindung ausgelöst werden. Im Rahmen des Sicherungsprozesses stellt nbackup eine reguläre Verbindung zur Datenbank her (in einigen Versionen sogar mehrmals). Um zu verhindern, dass Datenbanktrigger versehentlich ausgelöst werden, kann der neue Schalter `-T` verwendet werden. Beachten Sie, dass die entsprechenden Schalter in `gbak` und `isql -nodbtriggers` heißen (wir lieben Vielfalt hier bei Firebird).

## 4.5. Direkte I/O (Firebird 2.1.4+)

Ursprünglich verwendete nbackup direkte I/O nur, wenn eine Sicherung unter Windows NT erstellt wurde (und Nachfolger wie 2000, 2003 usw.). Auf allen anderen Betriebssystemen waren direkte I/O ausgeschaltet. Dies verursachte auf einigen Linux-Systemen Probleme, sodass in den Versionen 2.0.6 und 2.1.3 die direkten I/O auch unter Linux aktiviert wurden. Dies stellte sich jedoch für bestimmte andere Linux-Konfigurationen als problematisch heraus. In 2.1.4 und 2.5 wurde das ursprüngliche Verhalten wiederhergestellt, diesmal jedoch als Standard, der durch einen neu hinzugefügten Parameter überschrieben werden konnte: `-D`. Seine Verwendung ist wie folgt:

```
nbackup -B 0 cups.fdb cups.nbk -D on    -- direkte I/O an
nbackup -B 0 mugs.fdb mugs.nbk -D off  -- direkte I/O aus
```

Genau wie bei den Optionsschaltern selbst wird bei den Argumenten ON und OFF die Groß- und Kleinschreibung nicht berücksichtigt.

Direkte I/O werden nur beim Erstellen einer Sicherung angewendet, nicht während einer Wiederherstellung. Unter Windows wird dies durch Setzen von `FILE_FLAG_NO_BUFFERING` realisiert. Auf anderen Systemen werden `O_DIRECT` und `POSIX_FADV_NOREUSE` verwendet. Die beiden letzteren sind manchmal nicht verfügbar; in solchen Fällen werden sie (oder einer von ihnen) stillschweigend ausgelassen. Selbst wenn der Benutzer `-D on` explizit angegeben hat, führt dies nicht zu einer Warnung oder Fehlermeldung.

## 4.6. Informationsoptionen (Firebird 2.5+)

Neben den bereits erwähnten Parametern `-FE` und `-D` wurden in Firebird 2.5 auch die folgenden beiden eingeführt:

`-Z`

Zeigt einzeilige Versionsinformationen an. Diese Option kann unabhängig verwendet werden, aber auch in Kombination mit anderen Parametern wie `-B`, `-R`, `-L` usw.

`-?`

Zeigt eine Zusammenfassung der Verwendungs- und Befehlszeilenparameter von nbackup an. Achtung: Wenn diese Option vorhanden ist, werden alle anderen Parameter ignoriert!

## 4.7. Sicherungen auf Remote-Servern (Firebird 2.5+)

Nbackup selbst arbeitet nur mit lokalen Datenbanken. In Firebird 2.5 und höher können Sicherungen und Wiederherstellungen vom Typ nbackup auch remote über den Services Manager ausgeführt werden. Hierzu wird das Programm `fbsvcmgr.exe` auf dem lokalen Rechner verwendet; Es befindet sich im selben Ordner wie `nbackup.exe` und den anderen Firebird-Befehlszeilentools. Das erste Argument ist immer "Hostname: service\_mgr", wobei hostname der Name des Remote-Servers ist. Andere verfügbare Parameter sind:

```
-user username
-password password
-action_nbak
-action_nrest
-nbk_level n
-database database
-nbk_file filename
-nbk_no_triggers
-nbk_direct on|off
```

Das Erstellen eines vollständigen Backups auf dem Remote-Computer frodo sieht folgendermaßen aus:

```
fbsvcmgr frodo:service_mgr -user sysdba -password masterke
  -action_nbak -nbk_level 0
  -database C:\databases\countries.fdb -nbk_file C:\databases\countries.nbk
```

Und eine nachfolgende inkrementelle Sicherung:

```
fbsvcmgr frodo:service_mgr -user sysdba -password masterke
  -action_nbak -nbk_level 1
  -database C:\databases\countries.fdb -nbk_file C:\databases\countries_1.nbk
```

Um den ganzen Schebang wiederherzustellen:

```
fbsvcmgr frodo:service_mgr -user sysdba -password masterke
  -action_nrest -database C:\databases\countries_restored.fdb
  -nbk_file C:\databases\countries.nbk -nbk_file C:\databases\countries_1.nbk
```



Jeder der oben genannten Befehle sollte als einzelner Satz ohne Zeilenumbrüche eingegeben werden. Die Bindestriche vor den Parameternamen können weggelassen werden. Insbesondere bei langen Befehlen wie diesen kann es hilfreich sein, sie beizubehalten, damit Sie die einzelnen Parameter leicht identifizieren können (die Argumente erhalten keinen Bindestrich).

Anmerkungen:

- Der Services Manager erfordert immer eine Authentifizierung, sei es automatisch (root unter Posix, vertrauenswürdig unter Windows) oder explizit über die Parameter `-user` und `-password`. Die Umgebungsvariablen `ISC_USER` und `ISC_PASSWORD` werden nicht verwendet. `AUTO ADMIN MAPPING` in der Datenbank hat keine Auswirkung auf die Remoteverbindung (obwohl dies auch von der Konfiguration des Netzwerks abhängen kann).

Hinweis: Wenn die vertrauenswürdige Windows-Authentifizierung aktiviert ist, wird der Kontoname des Benutzers auf dem lokalen Computer an den Services Manager auf dem Remotecomputer übergeben. Wenn der Eigentümer der entfernten Datenbank ein Windows-Konto (z. B. `FRODO\PAUL`) und kein Firebird-Konto ist, *und* der Name des Windows-Kontos auf dem lokalen Computer mit dem Namen des Eigentümerkontos auf dem entfernten Computer identisch ist, wird der Anfragende als Datenbankeigentümer anerkannt und darf ein Backup erstellen. Dies könnte ein Sicherheitsrisiko darstellen, da selbst in lokalen Netzwerken der Benutzer `PAUL` auf einem Computer nicht unbedingt dieselbe Person ist wie der Benutzer `PAUL` auf einem anderen Computer.

- Das Wiederherstellen (`-action_nrest`) erfordert ebenfalls eine Authentifizierung, aber nach Überprüfung werden die Anmeldeinformationen in keiner Weise verwendet. Daher muss der Benutzer nicht der Datenbankeigentümer, `SYSDBA` oder Superuser sein. Bei der vertrauenswürdigen Windows-Authentifizierung muss der Benutzer auf dem Remotecomputer (auf dem sich die Datenbank befindet) überhaupt nicht vorhanden sein.

Diese schwache Authentifizierung impliziert ein weiteres potenzielles Sicherheitsrisiko. Angenommen, eine vertrauliche Datenbank wird nicht unterstützt, und die Sicherungen sind auf Dateisystemebene gut geschützt. Ein durchschnittlicher Benutzer kann die Datenbank dann nicht mit `nbackup` wiederherstellen, da `nbackup` im Benutzerprozessbereich ausgeführt wird. Wenn derselbe Benutzer jedoch den Namen und den Speicherort der Sicherung kennt oder diese analog erraten kann, kann er möglicherweise auf die Datenbank zugreifen, indem er sie von `fbsvcmgr` in einem öffentlichen Ordner wiederherstellen lässt. Schließlich ruft `fbsvcmgr` den Firebird-Server auf, der möglicherweise Zugriff auf das Backup auf Dateiebene hat. Natürlich gibt es Lösungen dafür, aber es ist wichtig, sich des Risikos bewusst zu sein.

- Der Services Manager kann auch lokal verwendet werden. In diesem Fall wird das erste Argument `service_mgr` ohne Hostnamen verwendet. Bei lokaler Verwendung hat `AUTO ADMIN MAPPING` den beabsichtigten Effekt. Dies gilt weiterhin, wenn Sie `localhost:` oder den Namen des lokalen Computers voranstellen. Die lokale Verwendung des Services Managers kann von Vorteil sein, wenn Sie keinen Dateisystemzugriff auf die Datenbank- und/oder Sicherungsdateien haben, der Firebird-Serverprozess jedoch. Wenn Sie über ausreichende Rechte verfügen, ist es praktischer, `nbackup` selbst mit seinen viel kürzeren Befehlen zu verwenden.
- Die Angabe von `-nbk_no_triggers` oder `-nbk_direct` mit `-action_nrest` führt zu einer Fehlermeldung. `Nbackup` selbst ist hier milder: Es ignoriert einfach die Parameter `-I` und `-D`, wenn sie im falschen Kontext verwendet werden.
- Anstelle eines Datenbankdateinamens können Sie auch einen Alias verwenden.
- Die Länge des Datenbankpfads (oder Alias) ist auf 255 Zeichen begrenzt.



## 4.8. Eine praktische Anwendung

Ein auf nbackup basierendes inkrementelles Sicherungsschema könnte folgendermaßen aussehen:

- Jeden Monat wird eine vollständige Sicherung mit Level 0 durchgeführt;
- Wöchentlich eine Level-1 Sicherung;
- Täglich eine Level-2 Sicherung;
- Stündlich eine Level-3 Sicherung.

Solange alle Sicherungen erhalten bleiben, können Sie die Datenbank für jeden beliebigen Zeitpunkt (im Stundenintervall), wiederherstellen. Für jede Wiederherstellungsaktion wird ein Maximum von vier Sicherungsdateien verwendet. Natürlich sollten Sie die Ausführungszeiten von aufwendigen Sicherungen zeitlich so planen, dass diese nicht zu Stoßzeiten ausgeführt werden. In diesem Fall könnten die Level-0 und 1 Sicherungen am Wochenende, und die Level-2 Sicherungen in der Nacht durchgeführt werden.

Falls Sie nicht immer alle Sicherungsdateien ewig aufbewahren wollen, dann können Sie auch ein Löschkonzept umsetzen. Zum Beispiel:

- Level-3 Sicherungen werden alle 8 Tage gelöscht;
- Level-2 Sicherungen jeden Monat;
- Level-1 Sicherungen nach sechs Monaten;
- Vollständige Sicherungen nach zwei Jahren, wobei die erste Sicherung von jedem Jahr aufbewahrt wird.

Dies ist natürlich nur ein Beispiel. Was in Ihrem speziellen Fall sinnvoll ist, hängt von der Anwendung, der Größe der Datenbank, den Operationen, usw... ab.

## 4.9. Sollten Sie weiterlesen?

Sie sind nun an einem Punkt angelangt, wo Sie alles über das Erstellen und das Wiederherstellen von vollständigen und/oder inkrementellen Sicherungen mit nbackup wissen. Sie brauchen nur dann weiterlesen, falls Sie andere Sicherungstools Ihrer Wahl für Ihre Firebird-Datenbank einsetzen möchten (siehe [Sperrern und entsperren](#)), oder falls Sie den Standardnamen oder den Speicherort der Deltadatei ändern wollen (siehe [Delta-Datei einstellen](#)).

Falls Sie keinen dieser Einsatzzwecke benötigen, dann wünschen wir Ihnen alles Gute bei der Verwendung von nbackup!

# Kapitel 5. Sperren und entsperren

Falls Sie Ihre eigenen Sicherungstools verwenden, oder wenn Sie nur eine Dateikopie erstellen wollen, dann hilft Ihnen nbackup mit dessen Sperr/Entsperr-Modi ebenfalls weiter. "Sperren" bedeutet hier, dass die Hauptdatenbankdatei vorübergehend eingefroren wird. Dies bedeutet allerdings nicht, dass keine Änderungen in der Datenbank durchgeführt werden können. Wie im Sicherungsmodus werden auch hier alle Änderungen in eine temporäre Deltadatei geschrieben. Unmittelbar nach dem Entsperren wird die Deltadatei mit der Hauptdatenbankdatei zusammengeführt.

Zur Erinnerung: nbackup.exe befindet sich im bin Unterverzeichnis Ihrer Firebird-Installation. Typische Speicherorte, zum Beispiel, sind: C:\Programme\Firebird\Firebird\_2\_0\bin (Windows) oder /opt/firebird/bin (Linux). Es existiert keine graphische Oberfläche. Sie müssen das Programm von der Kommandozeile aus starten (oder von einer Batchdatei oder einer Anwendung).

## 5.1. Sperren der Datenbank und Sicherung selbst erstellen

Eine typische Reihenfolge, um eine eigene Sicherung durchzuführen, kann wie folgt aussehen:

1. Sperren der Datenbank mit dem -L (Lock; Sperren) Schalter:

```
nbackup [-U user -P password] -L database
```

2. Kopieren/sichern/zippen Sie nun die Datenbank mit Ihren bevorzugten Sicherungstools. Die Erstellung einer einfachen Dateikopie ist ebenfalls möglich.
3. Entsperren Sie die Datenbank mit -N (uNlock; entsperren):

```
nbackup [-U user -P password] -N database
```

Die letzte Anweisung wird wiederum die Änderungen, die in die Deltadatei geschrieben wurden, mit der Hauptdatenbankdatei zusammenführen.

Die Sicherung beinhaltet die Daten, so wie diese zum Zeitpunkt der Sperre der Datenbank vorgefunden wurden, unabhängig davon, wie lange der Zustand der Sperre andauerte und wie lange Sie gewartet haben, bis die Sicherung tatsächlich durchgeführt wurde.

Anmerkungen:

- Anstelle eines Datenbankdateinamens können Sie auch einen Alias angeben.
- Die Parameter "-U" und "-P" können weggelassen werden, wenn die Envars ISC\_USER und ISC\_PASSWORD festgelegt sind, wenn Sie auf einem Posix-System als Root angemeldet sind oder wenn die vertrauenswürdige Authentifizierung unter Windows dies zulässt. Für weitere Informationen, siehe [Anmerkungen unter Vollständige Sicherungen erstellen](#).

- Ab Firebird 2.5 kann anstelle von `-P password` auch `-FE filename` verwendet werden.
- Sowohl `-L` als auch `-N` erstellen reguläre Datenbankverbindungen. In Firebird 2.1 und höher ist es sinnvoll den Schalter `-T` hinzuzufügen (siehe [Datenbanktrigger unterdrücken](#)).
- Wenn Sie eine Rohgerätedatenbank mit Firebird 2.1 oder höher sperren, kann die Option `-S` sehr hilfreich sein, siehe auch [Sperren von Rohgerätedatenbanken](#).
- Sie können optional `-Z` hinzufügen, um Versionsinformationen in die erste Zeile der Ausgabe zu drucken.



Was für das Sichern/Wiederherstellen gilt, gilt auch für die Sperr-/Entsperrschalter: Verwenden Sie sie nicht für Datenbanken mit mehreren Dateien. Lassen Sie `nbackup` in Datenbanken mit mehreren Dateien überhaupt nicht los, bis sich die Dinge geändert haben!

## 5.2. Wiederherstellen eines Backups nach “nbackup -L”

Eine Kopie einer gesperrten Datenbank ist selbst ebenfalls eine gesperrte Datenbank. Sie können sie also nicht einfach zurückkopieren und verwenden. Wenn Ihre ursprüngliche Datenbank verloren geht oder beschädigt wird und die selbst erstellte Kopie wiederhergestellt werden muss (oder wenn Sie die Kopie auf einem anderen Computer installieren möchten), gehen Sie wie folgt vor:

1. Kopieren/wiederherstellen/entpacken Sie die gesicherte Datenbankdatei mit Ihren dafür benötigten Werkzeugen.
2. Entsperren Sie nun die Datenbank, allerdings nicht mit dem `-N` Schalter, sondern mit `-F` (fixup):

```
nbackup -F database
```

Auch hier können Sie optional einen Alias anstelle eines Dateinamens verwenden und `-Z` für Versionsinformationen hinzufügen. Andere Optionen machen keinen Sinn.

Warum existieren mit `-N` und `-F` zwei Entsperrschalter?

- `-N` stellt sicher, dass irgendwelche Änderungen seit der Sperre mit `-L`, mit der Hauptdatenbankdatei zusammengeführt werden. Danach geht die Datenbank wieder in den normalen Lese/Schreib-Modus über und die temporäre Datei wird gelöscht.
- `-F` ändert nur das Zustandsflag der selbst-wiederhergestellten Datenbank auf “normal”.

Somit verwenden Sie:

- `-N` nachdem Sie eine eigene Kopie/Sicherung erstellt haben (um ein vorangegangenes `-L` wieder zurückzunehmen);
- `-F` nachdem Sie eine eigene Sicherung wiederhergestellt haben.



Es ist etwas unglücklich, dass der letzte Schalter mit `-F` für Fixup benannt wurde. Es wird in diesem Fall nichts hergerichtet/repariert, sondern es wird die Datenbank nur *entsperrt*. Andererseits führt der `-N` (uNlock) Schalter nicht nur ein Entsperren, sondern auch ein Fixup durch, nämlich die Integration der Änderungen von der Deltadatei in die Hauptdatenbankdatei. Damit müssen wir leben. Eine Gedankenstütze: Sie *können* `-F` als *nur kennzeichnen* betrachten.

## 5.3. Internas



Dieser Abschnitt enthält keine erforderlichen Kenntnisse, enthält jedoch einige zusätzliche Informationen, die Ihr Verständnis der verschiedenen Schalter vertiefen können.

`nbackup [parameter] -L` führt folgende Schritte durch:

1. Datenbankverbindung aufbauen;
2. Starten einer Transaktion
3. Aufruf von `ALTER DATABASE BEGIN BACKUP` (diese Anweisung wurde im Abschnitt [Weitere Informationen zu Sicherungen -B](#) diskutiert);
4. Die Transaktion committen;
5. Datenbankverbindung trennen.

`nbackup [parameter] -N` folgt den gleichen Schritten, allerdings mit einem “... `END BACKUP`” in Schritt 3.

`nbackup [parameter] -F` führt folgende Schritte aus:

1. Die wiederhergestellte Datenbankdatei wird geöffnet;
2. In dieser Datei wird das Zustandsflag von gesperrt (`nbak_state_stalled`) auf normal (`nbak_state_normal`) geändert;
3. Die Datei wird wieder geschlossen.



`nbackup -F` arbeitet rein auf Dateiebene und kann daher auch ohne einen laufenden Firebird Server ausgeführt werden. Jeder `-U` oder `-P` Parameter, der zum Kommando hinzugefügt wird, wird vollständig ignoriert.

## 5.4. Sperren von Rohgerätedatenbanken

Wie unter [Sichern von Rohgerätedatenbanken](#) beschrieben, können Probleme auftreten, wenn ein Delta für eine Datenbank auf einem Rohgerät erstellt werden muss. In Firebird 2.1 und höher weigert sich `nbackup` daher, mit Rohgerätedatenbanken zu arbeiten, es sei denn, es wurde zuvor ein expliziter Speicherort für die Delta-Datei festgelegt. Informationen zum Verfahren finden Sie unter [Delta-Datei einstellen](#) etwas weiter unten.

Es gibt noch ein weiteres Problem, wenn Sie ein Rohgerät sperren und kopieren: Sie kennen die

tatsächliche Größe der Datenbank nicht! Das Rohgerät kann 10 GB groß sein, aber die Datenbank belegt möglicherweise nur 200 MB dieses Speicherplatzes. Um zu vermeiden, dass das gesamte Gerät kopiert werden muss, um auf der sicheren Seite zu sein — was möglicherweise viel Zeit und Platz verschwendet –, hat Firebird 2.1 einen neuen Parameter für nbackup eingeführt: -S. Dieser Parameter ist nur in Kombination mit -L gültig. Wenn dieser vorhanden ist, schreibt nbackup die Datenbankgröße in Seiten nach dem Sperren der Datenbank in stdout. Da die Größe in Seiten angegeben ist, muss sie mit der Seitengröße der Datenbank multipliziert werden, um die tatsächliche Anzahl der zu kopierenden Bytes zu erhalten. Wenn Sie das Kopierdienstprogramm dd verwenden, können Sie die Seitengröße als (i)bs und die Ausgabe von nbackup -L -S als count angeben.

## 5.5. Sperren / Entsperren der Datenbank bei der Sicherung virtueller Maschinen

Die Verwendung von Sicherungswerkzeugen für virtuelle Maschinen ohne Vorbereitung der Datenbank für einen solchen Sicherungstyp kann zu einer beschädigten (d.h. nutzlosen) Sicherungskopie führen.

Der Firebird-Server verwendet intensiv seinen eigenen Cache im RAM, um den Betrieb zu beschleunigen, und implementiert komplexe Techniken, um die Datenbankkonsistenz zu jedem Zeitpunkt sicherzustellen. Backup-Tools für virtuelle Maschinen kennen den Cache von Firebird nicht und betrachten Datenbankdateien normalerweise nicht als Dateien mit wahlfreiem Zugriff.

Wenn die Sicherung der virtuellen Maschine abgeschlossen ist, hat die darin enthaltene Datenbankdatei den Status wie nach einem Hard-Reset der VM, und sehr oft ist eine solche Kopie nicht konsistent, also beschädigt. Die Wahrscheinlichkeit eines solchen Problems ist höher, wenn viele aktive Benutzer die Datenbank ändern oder wenn ein aktiver Sweep-Prozess läuft.

Solche inkonsistenten Sicherungen können in jeder virtualisierten Umgebung, einschließlich öffentlicher Clouds, auftreten.

Um eine gute Firebird-Datenbanksicherung mit dem VM-Sicherungstool zu erstellen, muss die Datenbankdatei vor der VM-Sicherung mit nbackup gesperrt und danach entsperrt werden. Normalerweise ermöglicht das VM-Sicherungstool die Ausführung benutzerdefinierter Skripts vor und nach der Sicherung, mit denen Sie Firebird-Datenbanken sperren / entsperren können.

## Kapitel 6. Delta-Datei einstellen

Standardmäßig befindet sich die Delta-Datei im selben Verzeichnis wie die Datenbank selbst. Der Dateiname ist ebenfalls derselbe, jedoch mit `.delta`. Dies ist normalerweise kein Problem, aber manchmal ist es wünschenswert oder sogar notwendig, den Ort zu ändern, z.B. wenn die Datenbank auf einem Rohgerät gespeichert ist. Nbackup selbst hat keine Möglichkeit, den Speicherort festzulegen. Dies muss über SQL erfolgen.

Stellen Sie mit jedem Client eine Verbindung zur Datenbank her, über die Sie Ihre eigenen SQL-Anweisungen eingeben und den folgenden Befehl eingeben können:

```
alter database add difference file 'pfad-und-dateiname'
```

Die benutzerdefinierbare Spezifikation der Deltadatei wird in der Datenbank gespeichert. Diese ist in der Systemtabelle `RDB$FILES` hinterlegt. Um das Standardverhalten wiederherzustellen, führen Sie die folgende Anweisung aus:

```
alter database drop difference file
```

Sie können beim Erstellen einer neuen Datenbank auch einen benutzerdefinierten Delta-Speicherort angeben:

```
create database 'path-and-dbname' difference file 'path-and-deltaname'
```



- Falls Sie in der `[ADD] DIFFERENCE FILE`-Anweisung nur den Dateinamen angeben, dann ist es wahrscheinlich, dass sich die Deltadatei *nicht* in dem selben Verzeichnis wie die Datenbank befinden wird, sondern im aktuellen Verzeichnis aus Sicht des Servers. Unter Windows kann das zum Beispiel das Systemverzeichnis sein. Die gleiche Logik gilt auch für relative Pfadangaben.
- Der gesamte Verzeichnispfad muss bereits vorhanden sein. Firebird versucht nicht, fehlende Verzeichnisse zu erstellen.
- Falls Sie die Spezifikation der Deltadatei ändern möchten, dann müssen Sie zuerst die alte Spezifikation mit `DROP` entfernen und die Neue mit `ADD` hinzufügen.

# Kapitel 7. Sicherungsverlauf

Die Firebird-Datenbank speichert einen Verlauf aller nbackup-Aktivitäten in der Systemtabelle RDB\$BACKUP\_HISTORY. Diese Informationen werden von nbackup selbst für die interne Verwaltung verwendet, können aber auch verwendet werden, um herauszufinden, wann die letzte Sicherung durchgeführt wurde, auf welcher Ebene und wie der Dateiname lautet.

Um beispielsweise die letzten 5 Sicherungen anzuzeigen, können Sie Folgendes verwenden:

```
SELECT RDB$BACKUP_ID, RDB$TIMESTAMP, RDB$BACKUP_LEVEL, RDB$GUID,
       RDB$SCN, RDB$FILE_NAME
FROM RDB$BACKUP_HISTORY
ORDER BY RDB$TIMESTAMP DESC
ROWS 5
```

Die Spalten in RDB\$BACKUP\_HISTORY sind:

| Spalte            | Beschreibung  |
|-------------------|---|
| RDB\$BACKUP_ID    | Primärschlüssel   |
| RDB\$TIMESTAMP    | Uhrzeit und Datum der Sicherung   |
| RDB\$BACKUP_LEVEL | Backup-Level  |
| RDB\$GUID         | GUID der Sicherung (zum Überprüfen der Abhängigkeiten zwischen Dateien) |
| RDB\$SCN          | Höchste Seitenmarkierung in der Sicherung                               |
| RDB\$FILE_NAME    | Dateiname der Sicherung   |

Eine Erläuterung des Feldes RDB\$SCN finden Sie im Abschnitt [Technische Hintergrundinformationen](#).

Der Inhalt der Tabelle RDB\$BACKUP\_HISTORY wird von gbak nicht gesichert und wiederhergestellt. Weitere Informationen finden Sie im Abschnitt [Technische Hintergrundinformationen](#).

# Kapitel 8. Technische Hintergrundinformationen

nbackup führt eine physische Sicherung der Datenbankseiten durch, indem Seiten kopiert werden, die seit der letzten Sicherung der unmittelbar vorhergehenden Ebene geändert wurden. Eine Sicherung der Stufe 0 kopiert alle Seiten, während eine Sicherung der Stufe 1 nur die Seiten kopiert, die nach der letzten Stufe 0 geändert wurden. Um die geänderten Seiten finden zu können, verwendet Firebird eine Markierung, die als *SCN* (kurz für Seiten-Scan) bezeichnet wird. Diese Zahl wird bei jeder Änderung des Sicherungsstatus erhöht. Für jede Sicherung mit nbackup gibt es drei Statusänderungen:

1. nbak\_state\_normal (kein Backup) zu nbak\_state\_stalled (Datenbank schreibt in Delta-Datei)
2. nbak\_state\_stalled zu nbak\_state\_merge (Zusammenführen der Delta-Datei zurück in die Datenbank)
3. nbak\_state\_merge zu nbak\_state\_normal (kein Backup)



Diese drei Statusänderungen treten auch dann auf, wenn die Sicherung fehlschlägt.

Der SCN der Datenbank vor dem Start der Sicherung wird zusammen mit der Sicherung aufgezeichnet. Das allererste Backup erhält SCN 0, das zweite 3 usw. Diese Nummer ist unabhängig von der Sicherungsstufe. Der SCN wird verwendet, um die Seiten einer Datenbank zu markieren. Also zum Beispiel:

| SCN | Bedeutung   |
|-----|---|
| 0   | Seiten vor einer Sicherung  |
| 1   | Seiten, die während der Sicherung in die Delta-Datei geschrieben/aktualisiert wurden                          |
| 2   | Seiten, die während des Zusammenführens der Delta-Datei in die Hauptsicherung geschrieben/aktualisiert wurden |
| 3   | Seiten, die nach Beendigung der ersten Sicherung + Zusammenführung geschrieben/aktualisiert wurden            |

Wenn eine Sicherung der Stufe 1 durchgeführt wird, sucht nbackup nach der Sicherung der letzten Stufe 0 und sichert alle Seiten mit einem SCN, der höher ist als der SCN dieser Sicherung der Stufe 0 (und so weiter).

Eine Sicherung und Wiederherstellung mit gbak stellt den Inhalt der Tabelle RDB\$BACKUP\_HISTORY nicht wieder her und setzt den SCN aller Seiten auf 0 zurück. Der Grund dafür ist, dass gbak anstelle einer physischen Sicherung eine logische Sicherung erstellt. Eine Wiederherstellung mit gbak schreibt also die gesamte Datenbank neu (und kann sogar die Seitengröße ändern). Dies macht frühere Sicherungen mit nbackup als Ausgangspunkt für nachfolgende Sicherungen bedeutungslos: Sie müssen mit einer neuen Stufe 0 beginnen.



# Anhang A: Dokumentenhistorie

Der genaue Dateiversionsverlauf wird im Firebird-Dokumentations-Git-Repository aufgezeichnet; siehe <https://github.com/FirebirdSQL/firebird-documentation>

## Versionsgeschichte

0.1 21. Okt. PV Erste Fassung.  
2005

1.0 1. Dez. PV "Beta"-Referenz in der Versionsinfo zu Beginn entfernt. Warnung bzgl. der  
2006 Angabe von Sicherungsdateinamen in Bezug auf die Verwendung mit nbackup -R geändert. "(oder wird)" im ersten Satz der Dokumentenhistorie entfernt.

In den Beispielen wurde C:\Datenbanken auf C:\Daten geändert, um Darstellungsprobleme im PDF Dokument mit der Hintergrundfarbe zu vermeiden.

Abschnitt *Setzen der Deltadatei* hinzugefügt, und den Abschnitt *Sollten Sie weiterlesen?* entsprechend abgeändert.

1.0 4. Dez. TS Deutsche Übersetzung basierend auf der englischen Dokumentenversion 1.0.  
2006 Feedback für die Übersetzung von Frank Ingermann und Paul Vinkenoog.

1.1 5. Mai PV *Backups erstellen und wiederherstellen*: Warnung vor Hochlastrisiken mit  
2008 nbackup 2.0.0-2.0.3 hinzugefügt.

*Vollständige Sicherung wiederherstellen*: Die falsche Aussage, dass nbackup eine vorhandene Datenbank überschreibt, wenn keine aktiven Verbindungen vorhanden sind, wurde korrigiert. Der kursive Text über den Fehler bei der interaktiven Wiederherstellung wurde in einen Hinweis geändert und dessen Korrektur in 2.0.1 erwähnt.

*Inkrementelle Sicherung*: In 2.1 wurde eine Warnung eingefügt, dass inkrementelle Sicherungen fehlerhaft sind.

*Datenbanktrigger unterdrücken (Firebird 2.1+)*: Neuer Abschnitt.

*Sollten Sie weiterlesen?*: Rechtschreibfehler beseitigt (engl. you → your).

## Versionsgeschichte

1.2 19. Sep 2011 PV Formatierung der Dokumentquelle: Geändert max. Zeilenlänge bis 100, ohne offene Zeilen.

Alle Abschnitte und Unterabschnitte haben jetzt eine ID.

*Einleitung*: Erster Satz bearbeitet.

*Nbackup features — eine Übersicht {zwei Doppelpunkte} Einschränkungen von nbackup*: Das zuvor zuletzt aufgelistete Listenelement wurde bearbeitet, um den Services Manager zu erwähnen. Listenelement zum direkten Dateizugriff hinzugefügt. Letzten Absatz entfernt.

*Funktionen und Parameter*: Neuer Abschnitt.

*Backups erstellen und wiederherstellen*: Leicht veränderter letzter Satz des ersten Abs. Erweiterte Warnung: Informationen zur Rolle der direkten I/O mit großen Datenbanken unter Posix hinzugefügt.

*Backups erstellen und wiederherstellen :: Vollständige Sicherungen :: Vollständige Sicherungen erstellen*: Korrigiertes und erweitertes Listenelement für die Parameter -U und -P. Listenelemente für den Parameter -FE (neu in 2.5), den Parameter -T (neu in 2.1) und den Parameter -D (neu in 2.5, Backport auf 2.1.4) hinzugefügt. In Listenelementen, die mit "Die verschiedenen Parameter" beginnen, lautet der Text in Klammern jetzt (-B, -U usw.), da viele neue Parameter hinzugefügt wurden.

*Backups erstellen und wiederherstellen :: Interna*: Kleine Anpassung (engl. image → impression).

*Backups erstellen und wiederherstellen :: Vollständige Sicherungen :: Wiederherstellen einer vollständigen Sicherung*: Die Parameter -U und -P wurden aus der Spezifikation entfernt. Listenelement für Aliase hinzugefügt. Separater Hinweis zum fehlgeschlagenen interaktiven Wiederherstellen auf kursiven Text im Listenelement selbst geändert. Listenelement hinzugefügt, dass Server nicht ausgeführt werden muss und Anmeldeinformationen ignoriert werden müssen.

*Backups erstellen und wiederherstellen :: Inkrementelle Backups*: Bearbeitete Warnung: Fix in 2.1.1 erwähnt.

*Backups erstellen und wiederherstellen :: Inkrementelle Backups :: Inkrementelle Sicherung wiederherstellen*: Die Parameter -U und -P wurden aus der formalen Syntax und dem 1. Listenelement entfernt.

*Backups erstellen und wiederherstellen :: Sichern von Rohgerätedatenbanken*: Neuer Abschnitt.

*Backups erstellen und wiederherstellen :: Datenbanktrigger unterdrücken*:

**Versionsgeschichte**

- |            |                  |        |  |
|------------|------------------|--------|--|
| 1.3        | 12. Okt.<br>2011 | PV     | <p><i>Funktionen und Parameter</i>: In der ersten Tabelle: selbst wiederhergestellt → vom Benutzer wiederhergestellt. In der zweiten Tabelle: Selbstwiederherstellung → Benutzerwiederherstellung.</p> <p><i>Sperren und entsperren</i> :: <i>Sperren der Datenbank und Sicherung selbst erstellen</i>: Abschnitt umbenannt <i>Sperren der Datenbank und Sicherung selbst erstellen</i>.</p> <p><i>Sperren und entsperren</i> :: <i>Wiederherstellen einer nach nbackup -L erstellten Sicherung</i>: 2. Liste in der 1. Auflistung: selbst wiederhergestellt → vom Benutzer wiederhergestellt.</p> |
| 1.4        | 18. Sep.<br>2014 | M<br>R | <p><i>Sicherungsverlauf</i>: Neuer Abschnitt</p> <p><i>Technische Hintergrundinformationen</i> Neuer Abschnitt</p>   |
| 1.5        | 27. Jun.<br>2020 | M<br>R | Konvertierung in AsciiDoc, geringfügige Bearbeitung von Texten   |
| 1.5-<br>de | 01. Jul.<br>2020 | M<br>K | Deutsche Übersetzung basierend auf der englischen Dokumentenversion 1.5.   |
| 1.6        | 25. Jul.<br>2020 | AK     | Absatz über VM-Sicherungen, Beispiel mit nicht standardmäßigem Port, Absatz mit Leistungshinweisen zur inkrementellen Sicherung  |
| 1.6-<br>de | 25. Jul.<br>2020 | M<br>K | Ergänzungen von Alexey Kovyazin in deutsche Übersetzung implementiert. Dokumenttitel angepasst.  |

## Anhang B: Lizenzhinweise

Der Inhalt dieser Dokumentation unterliegt der "Public Documentation License Version 1.0" (der "License"); die Dokumentation darf nur unter Respektierung dieser Lizenz genutzt werden. Kopien der Lizenz sind verfügbar unter <https://www.firebirdsql.org/pdfmanual/pdl.pdf> (PDF) und <https://www.firebirdsql.org/manual/pdl.html> (HTML).

Die Original-Dokumentation trägt den Titel *Firebird's nbackup tool*.

Der ursprüngliche Autor der Original-Dokumentation ist: Paul Vinkenoog.

Copyright © 2005-2020. Alle Rechte vorbehalten. Kontakt zum Original-Autor: <firstname> at <lastname> dot nl.

Mitwirkende: Thomas Steinmaurer; Mark Rotteveel; Alexey Kovyazin; Martin Köditz - siehe [Dokumenthistorie](#).