



## Firebird und IBExpert White Paper

# IBExpert: Wie macht man das? Metadaten und Daten automatisch abgleichen und dauerhaft aktives Backup!

Holger Klemt, Oktober 2022

## IBExpert: Wie macht man das?

**Teil 1: Automatischer Abgleich einer Datenbankstruktur zwischen mehreren Datenbanken**

**Teil 2: Automatischer Abgleich der Tabelleninhalte zwischen mehreren Datenbanken als dauerhaft aktives Backup**

## IBExpert: Wie macht man das?

### Automatischer Abgleich einer Datenbankstruktur zwischen mehreren Datenbanken

Mit IBExpert können Änderungen an einer Datenbank sehr einfach gegenüber einer anderen Datenbank abgeglichen werden. IBExpert erzeugt dafür vollautomatisch ein Script, das man dann auf der Zieldatenbank ausführen kann, um deren Metadaten zu aktualisieren.

1. Erzeugen Sie mit dem Script db1.sql aus dem Paket IBExpert Demo Database eine Referenzdatenbank, z.B. in c:\db\db1.fdb.
2. Registrieren Sie die Datenbank in IBExpert als alias db1.
3. Öffnen Sie die Datenbank.
4. Verschaffen Sie sich einen Eindruck von den bereits vorhandenen Tabellen, die Tabelle CUSTOMER enthält noch keine Demodaten.
5. Öffnen Sie die Prozedur INITALL, starten Sie diese mit F9 und geben Sie als Parameter 10000 ein.
6. Nachdem das durchgelaufen ist, vergessen sie nicht den COMMIT über dem Toolbutton.
7. Ein erneuter Blick in die Tabelle CUSTOMER zeigt nun zufällig erstellte Demodaten.
8. Schließen Sie die Datenbank-Verbindung.
9. Kopieren Sie die Datei c:\db\db1.fdb als c:\db\db2.fdb.
10. Registrieren Sie nun die neu erzeugte Datenbank db2 in IBExpert.



11. Obwohl wir noch 2 identische Datenbankdateien haben, starten wir schon mal mit Tools/Database Comparer den Abgleich der Datenbankstrukturen.
12. Tragen Sie db1 als Source ein und db2 als Target.
13. Alle Checkboxen im Dialog lassen Sie in der linken und mittleren Spalte angeklickt.
14. In der rechten Spalte sind alle Checkboxen aus, mit Ausnahme der Checkbox Verbose, die soll an bleiben.
15. Mit dem Start durch F9 werden alle Datenbankstrukturen der Source Datenbank db1 ausgelesen und mit denen der Target Datenbank db2 verglichen.
16. Sollte es irgendwelche Unterschiede geben, dann wird IBExpert ein Script erzeugen, um die Unterschiede auszugleichen.
17. Das funktioniert mit der Technik von IBExpert fast immer ohne Datenverlust und auch für sehr stark verschachtelte Objekte.
18. Nach dem Durchlauf zeigt das Ergebnis "No difference", weil ja beide Datenbanken auch noch völlig identisch sind.
19. Um nun in der Datenbank db1 eine Änderung zu machen, öffnen Sie die Datenbank db1 und führen in einem SQL-Fenster diesen Befehl aus

```
create table test1(id bigint not null primary key, txt varchar(80))
```

20. Mit F9 und einem COMMIT am Ende ist diese neue Tabelle nun in der Datenbank db1 angelegt, in Datenbank db2 aber noch nicht.
21. Schließen Sie den SQL-Editor wieder und starten Sie den noch offenen Database Comparer erneut.
22. Nach dem Durchlauf wird dieser Ihnen die fehlende Tabelle TEST in der Datenbank db2 anzeigen.
23. Beim Durchlauf erzeugt der Database Comparer immer nur intern ein Script, um die Metadaten anzupassen, führt das aber noch nicht aus.
24. Wenn etwas unterschiedlich ist, dann kann man das Script oben über die Toolbar (4. Button) in den Script Executive übergeben.
25. Im Script Executive kann man das dann mit F9 ausführen.
26. Nachdem es ausgeführt wurde, COMMIT nicht vergessen, und den Script Executive wieder schließen.
27. Wenn man nun den Database Comparer erneut startet, dann wird wieder "no difference" angezeigt, weil ja die Strukturen der Datenbanken wieder identisch sind.
28. Wie automatisiert man das nun? Schauen wir im Database Comparer auf den Reiter IBEBlock.
29. In diesem IBEBlock-Fenster stellt IBExpert immer die Scriptversion für alles zusammen, was mit IBExpert in der GUI zusammengestellt werden kann. Der IBEBlock-Reiter ist aber immer read-only, aber jede Checkbox wirkt sich dabei immer direkt auf die Parameter aus.
30. Um noch ein paar Anpassungen zu machen, kopieren wir den Inhalt der IBEBlock-Seite über die Zwischenablage in einen neu zu öffnenden SQL-Editor (F12 oder Tools/SQL Editor), um den dort wieder einzutragen.
31. Ersetzen Sie links von "-- Don't forget to specify file name for the result script" durch einen gültigen Dateinamen, z.B. 'c:\db\cmpres.sql' zwischen den Anführungszeichen.
32. Das was da nun im SQL-Editor steht, speichert man mit Rechtsklick - Save To File z.B. als c:\db\cmp.sql ab.
33. Der Database Comparer kann geschlossen werden.



34. Damit etwas passiert beim nächsten Durchlauf, erstellen wir eine weitere neue Tabelle in db1 im SQL-Editor

```
create table test2(id bigint not null primary key, txt varchar(80))
```

35. Laden Sie das Script wieder aus der gerade erzeugten Datei:\db\cmp.sql in den SQL Editor, z.B. mit Ctrl+L.
36. Starten Sie das Script mit F9.
37. Das Ergebnis liegt nun wie von uns vorgegeben in der Datei c:\db\cmpres.sql.
38. Damit dieses Ergebnis-Script kein einfacher SQL oder IBEBlock sein kann, brauchen wir dafür ein Script Executive, der mehrzeilige Scripte ausführen kann. Diesen Starten Sie mit Tools-Script Executive oder Ctrl+F12.
39. Mit Ctrl+L laden wir das Script aus der Datei c:\db\cmpres.sql. Wir sehen, dass die neue Tabelle test2 angelegt werden soll, es wird aber noch nicht ausgeführt!
40. Um den Abgleich zu automatisieren, erstellen wir mit Notepad eine ganz einfache Batchdatei c:\db\cmp.bat mit folgendem Inhalt

```
ibescript.exe cmp.sql  
ibescript.exe cmpres.sql
```

41. IBEScript.exe ist die Kommandozeilen-Version von IBExpert und kann sämtliche SQL und IBEBlock Scripte auch ohne Interaktion ausführen. IBEScript.exe ist bei der IBExpert Vollversion dabei, ebenso gibt es eine preisgünstige IBExpert Server Tools Version oder, für die Weitergabe als Softwarehersteller, auch eine IBExpert Distribution OEM Version.
42. Wir starten ein cmd.exe DOS-Fenster, gehen in den Pfad c:\db und starten dort cmp.bat.
43. Im ersten Schritt erstellt das Script eine Scriptdatei, mit der die Datenbank db2 abgeglichen wird und legt diese als cmpres.sql an.
44. Im zweiten Schritt führt das Script diese Datei aus, beginnt also umgehend mit den Anpassungen der Datenbank db2, damit diese die gleiche Struktur wie die Datenbank db1 hat.
45. Nach dem Durchlauf öffnen wir die Datei db2 in IBExpert erneut und sehen nun auch die neue Tabelle TEST2.
46. Wir schauen das Script c:\db\cmpres.sql an und sehen darin den CREATE TABLE Befehl, dieser wurde durch das Script aber schon ausgeführt.
47. Wir starten erneut die Batch cmp.bat im DOS-Fenster.
48. Wir schauen das Script c:\db\cmpres.sql an und sehen da nichts mehr drin, weil nun ja keine Differenzen mehr existieren, Script-Datei schließen nicht vergessen.
49. Nun kann die Batch per Scheduler täglich gestartet werden oder wann immer man das braucht.
50. Bekannte Einschränkungen bei der Metadatensynchronisierung: eigentlich keine, IBExpert kommt damit ziemlich abstrusen Datenbankkonstruktionen gut klar.



## IBExpert: Wie macht man das?

### Automatischer Abgleich der Tabelleninhalte zwischen mehreren Datenbanken als dauerhaft aktives Backup

Seit Firebird 4 ist eine Datenbankreplikation in Firebird enthalten, aber wenn der Umstieg auf Firebird 4 nicht so schnell in Frage kommt, ist ein sehr leistungsfähiger Datenbankabgleich auch für Firebird 2.5 und 3.0 machbar. Dieses hier vorgestellte Verfahren kann für den Abgleich von Master-Slave Datenbankkonfigurationen sehr einfach aufgesetzt werden, ohne dabei irgendwas an den Metadaten ändern zu müssen. Die benutzten Datenbanken db1 und db2 entsprechen dem oben erwähnten Beispiel.

1. Wir öffnen in IBExpert den Menüpunkt Tools/Table Data Comparer.
2. Als Master nehmen wir wieder Datenbank db1, Target ist wieder Datenbank db2.
3. Auf der ersten Seite kann man die Tabellen anklicken, die synchron sein sollen.
4. Wichtig: die Tabellen sollten einen Primärschlüssel haben oder eine Unique Constraint.
5. Wir synchronisieren zunächst mal die Tabellen CUSTOMER, PRODUCT, TEST1 und TEST2.
6. Damit der Abgleich möglichst schnell passiert, wird oben die Checkbox "online sync" aktiviert.
7. Wenn "online sync" aktiviert ist, startet der Abgleich sofort und es muss nicht zusätzlich noch eine SQL Ergebnisdatei geschrieben werden, daher entfernen wir den Dateinamen und den Pfad komplett.
8. In der rechten Spalte wählen wir den Generator ID an, auch dieser soll synchronisiert werden.
9. Auf der zweiten Seite im Table Data Comparer findet man nun viele Option. Üblicherweise braucht man davon nun mindestens oben die obersten 5 Checkboxes aktiv, das meiste weitere ist selbsterklärend.
10. Auch hier könnten wir das interaktiv in IBExpert starten, aber wir gehen wieder auf die Seite IBEBlock und speichern den Inhalt als C:\db\datacmp.sql ab.
11. Wir erstellen eine neue Batch c:\db\datacmp.bat mit diesem Inhalt

```
ibescript datacmp.sql
timeout 15
datacmp.bat
```

12. Wir starten die Batchdatei datacmp.bat.
13. Wir öffnen die Datenbank db1 und ändern dort etwas an einem Datensatz in der Tabelle CUSTOMER, COMMIT nicht vergessen.
14. Nach dem Durchlauf der Batchdatei, die sich immer wieder selbst nach timeout startet, schauen wir die Daten in der Datenbank db2 an und sehen die gleichen Werte in der Tabelle CUSTOMER.
15. Ebenso kann das nun mit neuen Datensätzen in den Tabellen TEST1 und TEST2 probiert werden.
16. Das Ganze funktioniert für alle Datentypen und auch für Blobs.
17. Wenn man komplexe Trigger hat, die selbst Daten erzeugen, dann sollte man diese ggf. in der Datenbank db2 mit ALTER TRIGGER ... inactive deaktivieren.
18. Foreign Keys usw. sollten ggf. auch synchron gehalten werden. IBExpert erkennt die erforderliche Reihenfolge sehr gut.
19. Das. o.a. Script ist als funktionales Beispiel gedacht und kann auch in der Realität als Echtzeit Zusatzbackup laufen, man sollte aber die Datenbankgröße und Anzahl der Records im Auge behalten.



20. Für sehr große Datenbanken mit Millionen von Datensätzen muss man das anders machen, das geht aber auch mit FB25 oder neuer ohne Probleme. Wir haben damit jahrelange Projekterfahrung und bieten transaktionssichere Echtzeit oder Near-time Replikation als individuelle Kundenprojekte an.
21. Als Runtime braucht man dafür mit dem o.a. Beispiel ggf. die IBExpert Server Tools, da ist die IBEScript.exe mit dabei und kann auch ohne IBExpert Vollversion beim Kunden laufen.
22. Praktischer Einsatz:
  - Kopiert eure Datenbank auf einen zweiten Rechner oder auf demselben Rechner in einen anderen Pfad (am Besten vorher den Firebird Serverdienst kurz beenden und nach dem Kopieren neu starten).
  - Erstellt mit der o.a. Technik ein IBEBlock Script für alle eure Tabellen mit der Produktionsdatenbank als Source und der Backup Datenbank als Target.
  - Deaktiviert alle Trigger in der Backup Datenbank (in IBExpert gibt es dafür einen Rechtsklick auf Trigger).
  - Erzeugt die Batchdatei und wählt den Timeout so, dass der für eure Zwecke passt, z.B. 3600 = stündlich.
  - Die Batchdatei vergleicht nun alle Tabellen, die einen Primärschlüssel oder Unique Constraint haben, in beiden Datenbanken und ihr habt ohne großen Aufwand ein dauerhaft mitlaufendes Backup.
  - Sollte nun eure Produktionsdatenbank nicht mehr erreichbar sein, dann braucht ihr nun die Backup-Datenbank zu öffnen, alle Trigger wieder aktivieren und ihr habt eine Datenbank, die ziemlich aktuell sein wird.
  - Für ganz wichtige Tabellen könnt ihr ein extra Script aufsetzen mit einem sehr kurzen oder gar keinem Timeout. Die dort erkannten Differenzen müssen dann zum Beispiel nicht auf die anderen Scripte warten, die zum Beispiel dann auch Blobs abgleichen usw.
  - Für sehr große Datenbanken oder Tabellen mit sehr vielen Datensätzen muss man das Laufzeitverhalten prüfen. Eine grobe Schätzung: Datenbanken bis 1 GB meistens kein Problem, bis 10 GB sollte der Abgleich tabellengruppenweise aufgeteilt werden und ab 100 GB muss man am Besten ein anderes Verfahren wählen oder sollte nur einen Teil abgleichen.
  - Ein langsamer Server wird mit diesem Verfahren ganz sicher nicht schneller. Prüft am Besten mit dem IBExpert Benchmark, ob euer Server überhaupt eine akzeptable Geschwindigkeit liefert, bevor ihr das Verfahren auf einer lahmen Kiste einsetzt.