

EASY-INTERBASE-REPLICATION

Replikation von Interbase-Datenbanken

Einfacher Ausschluß von Tabellen

Replikation zeit- oder ereignis- basierend

bidirektionalen Replikation

Benachrichtigung bei Replikationsfehlern durch Events

Einfache Replikation von Datenbanken ohne Programmierung

Replikation beliebig vieler Datenbanken

Ausschluss beliebiger Datensätze über SQL-Statements

Datensatz, Tabellen und Datenbanksynchronisation in beliebige Richtungen

Replikationskonflikte können vom Administrator über ein Protokoll gelöst werden

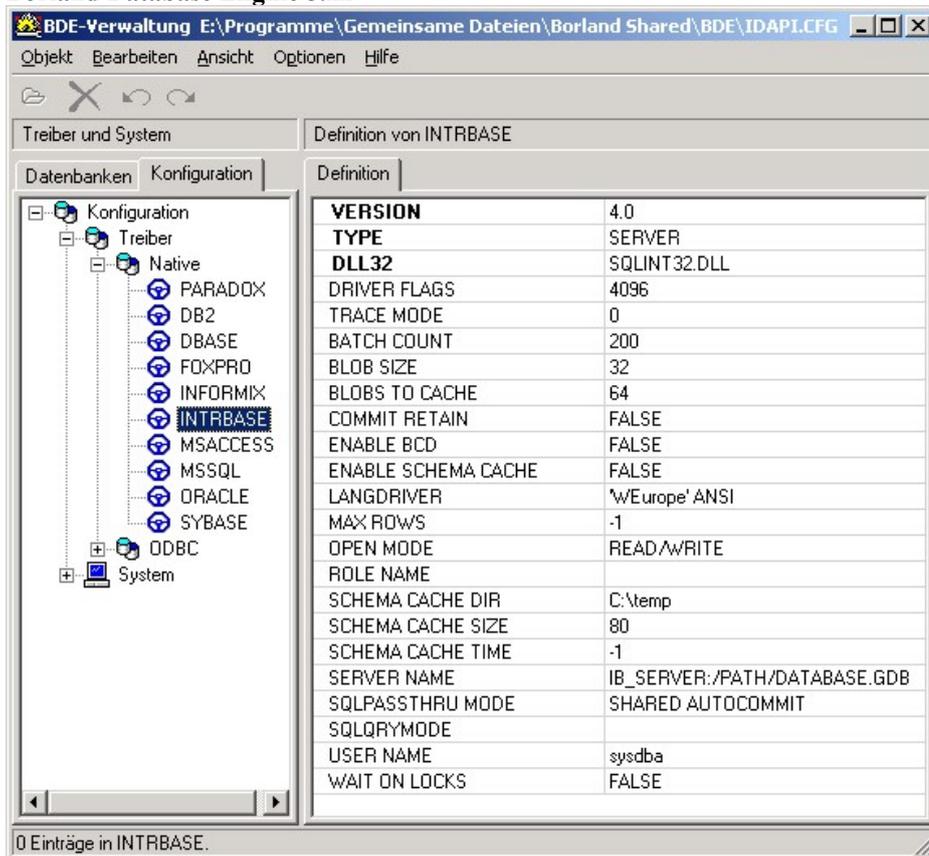
Inhaltsverzeichnis:

1.	SYSTEMVORAUSSETZUNGEN	3
1.1.	Software	3
1.2.	Vergabe von Primärschlüsseln zur Replikation	3
2.	SHAREWARE, VOLLVERSION UND REGISTRIERUNG	5
3.	ARBEITSWEISE	6
3.1.	Grundsätzliches.....	6
3.2.	Netzwerkeplikation zwischen festen Standorten	6
3.3.	Replikation von Datenbanken auf mobilen Computern und einer Netzwerkdatenbank	6
4.	ERSTELLUNG EINER REPLIKATIONSDEFINITION	7
4.1.	Replikation erstellen.....	7
4.2.	Replikation Ändern	7
5.	REPLIKATION ERGÄNZENDE EINSTELLUNGEN	8
5.1.	Öffnen der Datenbankverbindung und Definition der Einschränkungen.....	8
5.2.	Tabellenausschluss.....	8
5.3.	Datensatz einschränkung.....	8
6.	SYSTEMOBJEKTE	9
6.1.	Systemobjekte anlegen.....	9
6.2.	Systemobjekte entfernen	10
6.3.	Systemobjekte prüfen.....	10
6.4.	Systemobjekte	10
7.	SYNCHRONISATION	11
7.1.	Datenbanksynchronisation	11
7.2.	Einzeltablelle synchronisieren.....	13
7.3.	Datensatzsynchronisation.....	13
8.	REPLIKATION AKTIVIEREN	16
9.	REPLIKATION STARTEN	16
9.1.	Replikation im ständigen Netzwerkbetrieb.....	17
9.2.	Replikation von der Serverdatenbank zum mobilen Rechner	17
9.3.	Die Replication Machine als Anwendung im Einsatz.....	18
10.	FEHLERBENACHRICHTIGUNG	19
11.	FEHLERBEHANDLUNG	19
11.1.	Allgemeinfehler	20
11.2.	Behandlung von Datensatzfehlern	20

1. Systemvoraussetzungen

1.1. Software

▪ Borland Database Engine 5.xx



▪ Interbase-Server bzw. Firebird Server

▪ Datenbanken müssen die gleiche Struktur haben

In der vorliegenden Programmversion ist eine identische Struktur aller zu replizierenden Datenbanken erforderlich. Eine Zuordnung von Spaltenwerten ist nicht möglich bzw. entfällt.

▪ Datenbankübergreifende Eindeutigkeit des Primärschlüssels

1.2. Vergabe von Primärschlüsseln zur Replikation

Es werden nur Tabellen repliziert, die einen Primärschlüssel haben. Bei der Replikation werden auch die Primärschlüssel der jeweiligen Datenbank übernommen. Deshalb dürfen, sofern der Primärschlüssel nur aus einem Integerwert besteht, keine Trigger in folgender Form enthalten sein:

```
CREATE TRIGGER TRIGGERNAME FOR TABELLENAMEN BEFORE INSERT POSITION 0 AS BEGIN  
NEW.GENID = GEN_ID(SCHLUESSEL,1); END
```

In diesem Beispiel wäre in der Tabelle „Tabellenname“ ein Feld „GENID“ als Integer definiert, welches den Primärschlüssel beinhaltet und den Wert aus dem Generator Schlüssel holt. Diese Form der Primärschlüsselvergabe ist für eine Replikation nicht geeignet. Stattdessen sollte man die Eindeutigkeit des Primärschlüssels datenbankübergreifend sicherstellen.

Folgende Möglichkeiten bestehen:

Möglichkeit 1 (Fortlaufende Nummer und Hostname ergeben Primärschlüssel):

© Copyright by ECL Technologies & Consulting GmbH

Der Primärschlüssel sämtlicher Datenbanken wird aus zwei Teilen zusammengesetzt: Ein Teil ist ein durch Generatoren definierter Integerwert. Der zweite Teil ist der Hostname des jeweiligen Rechners, der ein Insert durchgeführt hat. Hostnamen müssen eindeutig sein und dürfen nicht doppelt vorkommen! Der Generator darf in diesem Fall nicht durch einen Trigger angestoßen werden. Der Wert wird aus dem Quellcode des Programms ermittelt und eingetragen.

```
SELECT GEN_ID(SCHLUESSEL,1) FROM RDB$DATABASE
```

Diese Abfrage muss also vom Clientprogramm aufgerufen werden, um den Wert zusammen mit dem Hostnamen des Rechners zu speichern, der den Datensatz erzeugt.

Möglichkeit 2 (Fortlaufende Nummer und Rechnernummer ergeben Primärschlüssel)

Wenn bereits eine Datenbank besteht, kann auch der alte Primärschlüssel um eine neue Spalte (z.B. „Rechner“) ergänzt werden. Hierbei muss das bestehende Programm so umgeschrieben werden, dass bei jeder Rechnerinstallation eine Rechnernummer mit in diese Spalte gespeichert wird, damit die Eindeutigkeit des Datensatzes sichergestellt ist.

2. Shareware, Vollversion und Registrierung

Zur Vereinfachung der Verbreitung und des Vertriebs haben wir uns entschlossen EASY-INTERBASE-REPLICATION als Shareware zum Download zur Verfügung zu stellen. Diese Version kann aber durch ein einfaches Freischaltprozedere in eine Vollversion umgewandelt werden. Voraussetzung für die Freischaltung ist das Registrieren des EASY-INTERBASE-REPLICATORS.

Solange die Software nicht Freigeschaltet ist, erscheint beim Start der Software sowie nach der Replikation von 1000 Datensätzen folgender Hinweis.

Nachdem bei uns ein Zahlungseingang stattgefunden hat erhalten Sie einen Aktivierungskkey bzw. eine Seriennummer mit der Sie die Software Freischalten können.



Wenn die Software bei der fin@time Software GmbH registriert ist, erhalten sie die Daten zur Freischaltung. Tragen Sie die Daten in folgenden Dialog ein und drücken den Schalter Freischalten.

Registrierung und Freischaltung

Name/Firma

Strasse

Land PLZ Ort

Seriennummer

Freischalten

Registrieren Sie diese Software noch heute im Internet unter
<http://www.fin-at-time.de/>

3. Arbeitsweise

3.1. Grundsätzliches

Beim Eintragen und Entfernen von Systemobjekten in den Datenbanken muss Exklusiv-Zugriff auf die Datenbanken bestehen. Kein anderer User darf mit der Datenbank verbunden sein. Vor dem Einsatz der Software bzw. dem Eintragen und Entfernen von Systemobjekten sollten auf jeden Fall eine Datensicherung Ihrer Datenbanken erstellt werden.

Die komplette Replikationssoftware (EASY-INTERBASE-REPLICATION) besteht aus drei getrennten Programmen: Die eigentliche Replikationssoftware (**easy Replication machine**) als Anwendung und Dienst, der Administrations-, Konfigurations- und Fehlermanagement-Software (**easy Interbase Replication**) sowie einem kleinen Trayicon-Programm zur Fehlerbenachrichtigung (**Error Event Messenger**)

Die Replikationssoftware selbst beinhaltet auch eine Interbase-Datenbank (IBREPL.GDB) welche die Konfigurationsdaten sowie das Fehlerprotokoll beinhaltet. Für diese Datenbank muß ein BDE-Alias eingerichtet werden, welcher den Namen IBREPL haben muß.

Wird eine getrennte Installation zwischen Administrationssoftware (am Client des Administrators) und der Replicationmaschine (am Server) erstellt, muss natürlich auf beiden Rechnern der Alias eingerichtet sein und auf ein und die selbe Datenbank zeigen. Die Benachrichtigungssoftware für Fehler (ErrorEventMessenger) muss in diesem Fall am PC des Administrators im gleichen Verzeichnis wie die Administrationssoftware installiert sein.

Nach Installation der Software sollten Sie für die zu replizierenden Datenbanken eindeutige Aliasnamen in Ihrer BDE definieren. Nach Start des Administrationsprogramms können Sie mit der Definition beginnen.

3.2. Netzwerkreplikation zwischen festen Standorten

Diese Art der Replikation kann entweder annähernd synchron oder asynchron erfolgen. Der asynchrone Abgleich erfolgt jeweils in fest definierten Intervallen, in denen die Warteschlangen abgearbeitet werden. Der synchrone Abgleich wird durch Datenbankevents angestoßen, welche die Replikationssoftware zum bearbeiten der Warteschlangen auffordert.

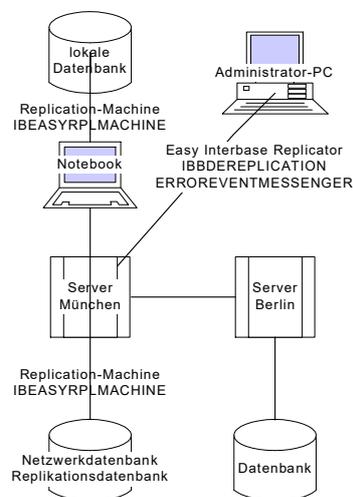
3.3. Replikation von Datenbanken auf mobilen Computern und einer Netzwerkdatenbank

Die Replikation von mobilen Computern erfolgt durch expliziten Aufruf bevor der Client das Netzwerk verläßt, bzw. wenn der Client wieder am Netzwerk angemeldet wird. Ein Anstoß der Replikation über ein DFÜ-Netzwerk ist selbstverständlich auch möglich. Die Konfigurationsdatenbank sollte Zentral am Server liegen. Auf jedem mobilen Computer muss ein Alias mit dem Namen IBREPL eingerichtet sein, der auf diese Datenbank zeigt. Die Replication-Machine wird am mobilen Computer installiert und vor Verlassen des Netzwerks bzw. nach erneuter Anmeldung im Netzwerk mit den entsprechenden Parametern ausgeführt.

Das nebenstehende Prinzipschaltbild zeigt folgende Konfiguration:

Am Server in München ist der zentrale Punkt. Dort werden die Replikationsdatenbank sowie die Sourcedatenbank gespeichert. Auf diesem Server läuft die Replication-Machine die den synchronen (eventbasierten) Abgleich mit der Datenbank in Berlin regelt.

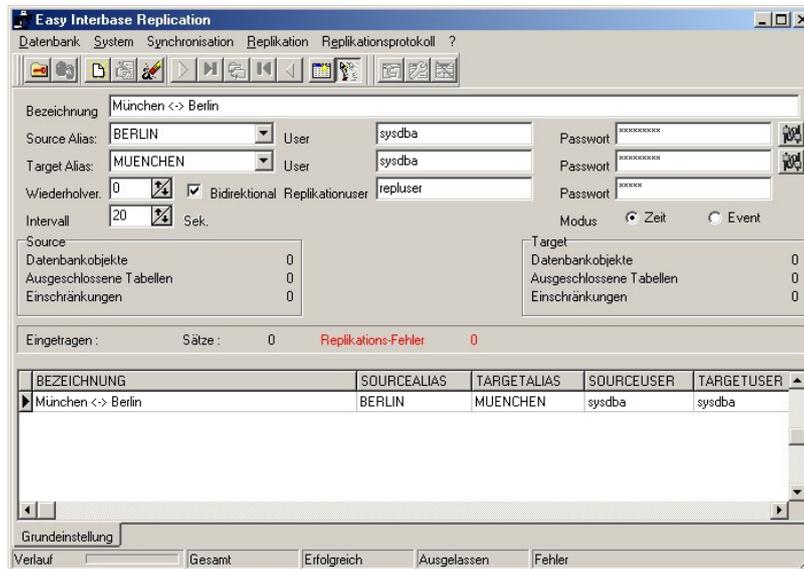
In einer zweiten Konfiguration ist die Verbindung zu einem Notebook dargestellt. Hierfür ist wieder die Datenbank in München die Sourcedatenbank und die Datenbank am Notebook die Targetdatenbank. Die Replication-Machine am Notebook, welche beim Anmelden und vor dem Abmelden einmalig angestoßen wird, regelt den Abgleich mit der Netzwerkdatenbank in München. Der Administrator kann von seinem PC die Fehlerprotokolle einsehen und bearbeiten.



4. Erstellung einer Replikationsdefinition

4.1. Replikation erstellen

Das Replikationsprozedere wird mit Hilfe der „easy Interbase Replication“ angelegt:



Schritt 1: Geben Sie Ihrer Replikation einen eindeutigen Namen, definieren Source- und Target-Alias sowie die entsprechenden Passwörter. Im Feld <Wiederholversuche> wird festgelegt, wie oft die Replikationssoftware im Fehlerfall versuchen soll, den Datensatz erneut zu replizieren.

Schritt 2: Wird eine Bidirektionale Replikation erstellt, dass Datensätze also in beide Richtungen repliziert werden sollen, muss ein Replikations-User eingerichtet werden. Dies dient der Kennzeichnung, damit Datensätze die von der Replikationssoftware erstellt, geändert oder gelöscht werden nicht wieder repliziert werden.

Hinweis: Der Replikations - User muss unbedingt bei jeder Datenbank (Target- und Source) eingerichtet werden und mit den Rechten zum Lesen und Schreiben in den Tabellen ausgestattet sein. Achten Sie darauf, diesen User erst nach dem Erstellen der Systemobjekte mit den Rechten zu versehen, da er auch Zugriff auf die Replikationswarteschlange benötigt, die vor der Erstellung der Systemobjekte noch nicht existiert.

Schritt 3: Definition des Replikationstyps. Es besteht die Möglichkeit, zwischen ereignisgesteuerter und intervallgesteuerter Replikation zu wählen. Im Falle der intervallgesteuerten Replikation muss das Timerintervall angegeben werden. Für die asynchrone Einmalreplikation (Mobiler PC und Netzwerk) spielt es keine Rolle, welchen Replikationstyp sie auswählen.

Schritt 4: Durch klick auf den Schalter <HINZUFÜGEN> wird die Definition gespeichert.



4.2. Replikation Ändern

Ändern: Zum Ändern und Löschen der Replikationsdefinition darf die Datenbank nicht geöffnet sein. Sobald eine Änderung in den Grunddaten der Definition gemacht wurde, kann über den Menüpunkt „Datenbank“-„Ändern“ bzw. dem entsprechenden Schalter auf der Toolbar die Änderung in der Konfigurationsdatenbank gespeichert werden.



Löschen: Über den Menüpunkt „Datenbank“-„Löschen“ oder dem entsprechendem Schalter auf der Toolbar kann der Konfigurationsdatensatz aus der Datenbank gelöscht werden.



Hinweis: Bevor Sie eine Konfiguration aus der Datenbank löschen, sollten Sie sich unbedingt vergewissern, dass keine Systemobjekte mehr in den jeweiligen Datenbanken eingetragen sind. Diese können nach dem Löschen des Konfigurationsdatensatzes nicht mehr entfernt werden!

5. Replikation ergänzende Einstellungen

5.1. Öffnen der Datenbankverbindung und Definition der Einschränkungen

Schritt:  Nach Erstellung der Definition muss für die Replikation die Verbindung zu den Datenbanken aufgebaut werden. Hierzu dient der Menüpunkt „Datenbank“ „Öffnen“ bzw. der entsprechende Schalter auf der Toolbar. Der gleiche Schalter bzw. Menüpunkt kann wieder zum schließen der Verbindungen verwendet werden. Änderungen an der Grundkonfiguration können nur bei geschlossener Verbindung gemacht werden.

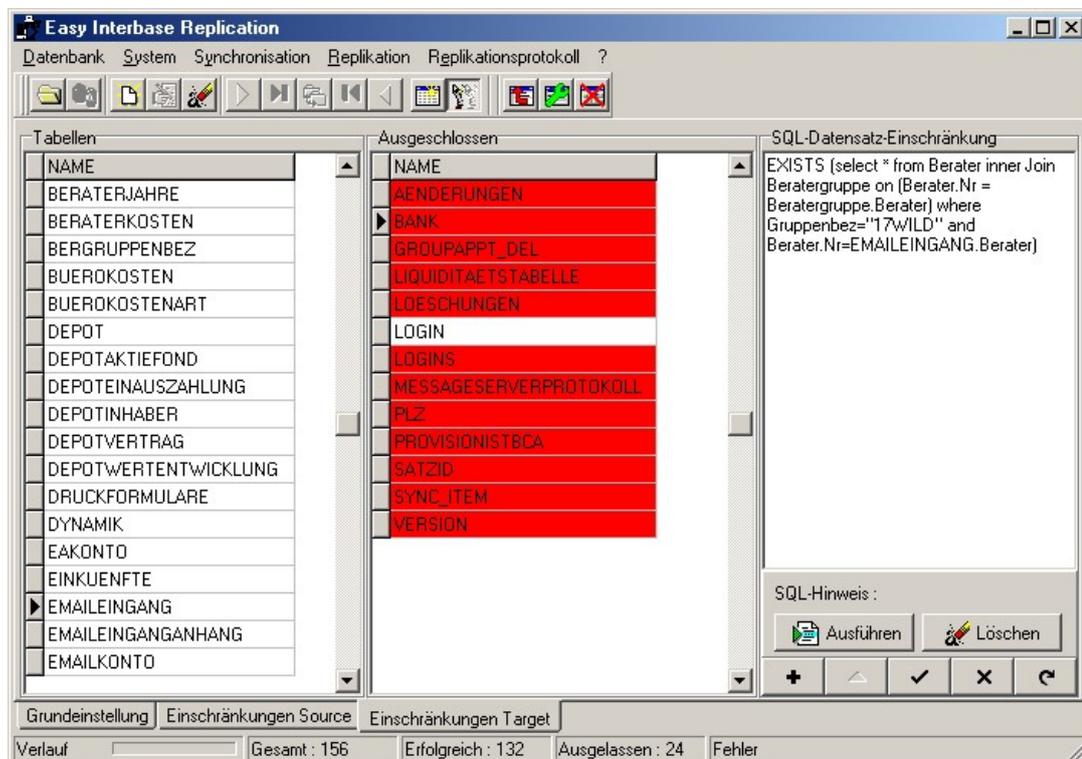
Hinweis: Gelingt die Verbindung zu den Datenbanken, wird die Konfigurationsdatenbank nach dem Öffnen mit den Primärschlüsseln der jeweiligen Tabellen gefüllt. Sind Datenbanktabellen ohne Primärschlüssel vorhanden, werden diese Tabellen automatisch zu den der Replikation ausgeschlossenen Tabellen hinzugefügt. **Tabellen ohne Primärschlüssel können nicht repliziert werden.**

5.2. Tabellenausschluss

Schritt: Nach der Erstellung der Konfigurationsdatenbank sind die Register „Einschränkungen Source“ und „Einschränkungen Target“ verfügbar. Für jede Seite können komplette Tabellen von der Replikation ausgeschlossen werden. Dies geschieht einfach via Drag and Drop, indem man den Tabellen-Namen in den Bereich „Ausgeschlossen“ zieht. Umgekehrt kann man natürlich ausgeschlossene Tabellen wieder zurück zu den Tabellen schieben, damit diese wieder zur Replikation hinzugefügt werden. Rot dargestellte Tabellen sind bereits ausgeschlossen!

5.3. Datensatzeinschränkung

Für jede Seite können nach frei definierten SQL-Befehlen Datensätze von der Replikation ausgeschlossen werden. Für jede Tabelle kann ein Ausdruck definiert werden:



In diesem Beispiel :

```
select * from emaileingang where EXISTS (select * from Berater inner Join Beratergruppe on (Berater.Nr = Beratergruppe.Berater) where Gruppenbez="17WILD" and Berater.Nr=EMAILEINGANG.Berater)
```

Natürlich können in diesem Where - Part Einfache wie Komplexe Ausdrücke stehen. Zum Testen des Ausdrucks kann der Schalter <AUSFÜHREN> betätigt werden. Ist der Ausdruck OK, wird eine Tabelle mit den Werten angezeigt, andernfalls eine Fehlermeldung. Zum speichern, löschen bzw. ändern eines Ausdrucks kann die Navigationsleiste verwendet werden.

6. Systemobjekte

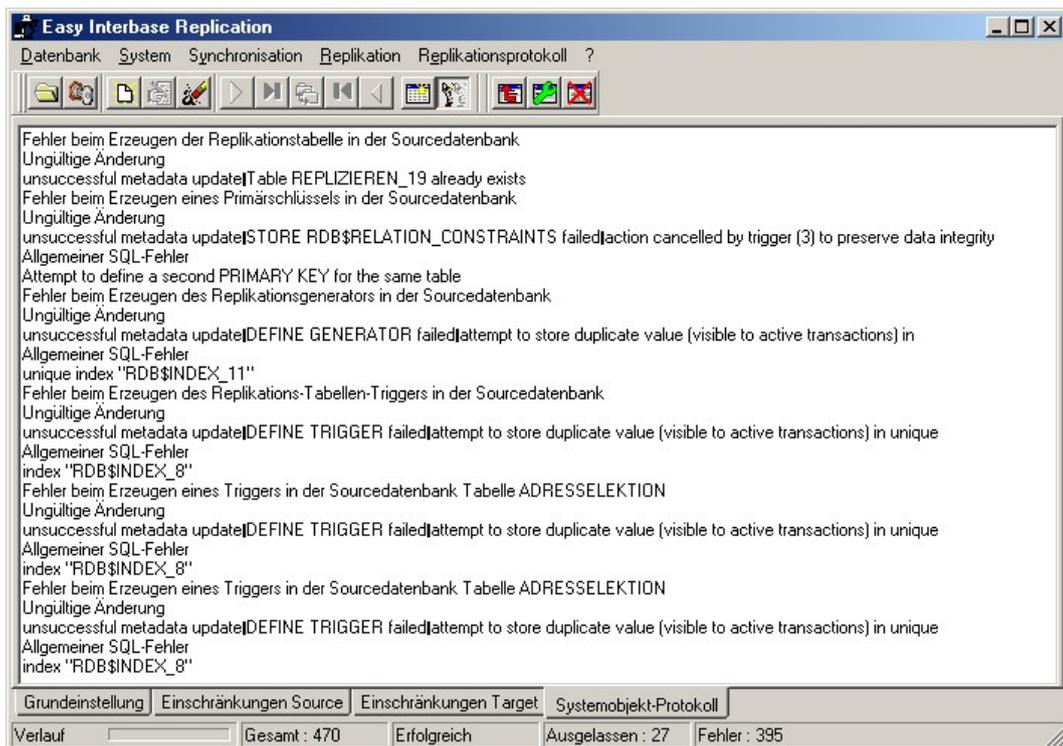
6.1. Systemobjekte anlegen

Hinweis: Bitte achten Sie darauf, beim eintragen und löschen von Systemobjekten Exklusivzugriff auf die Datenbanken zu haben. Vor dem Hinzufügen empfehlen wir Ihnen, eine Datensicherung der Datenbanken anzufertigen!

Schritt: Nachdem alle Einschränkungen und Ausschlüsse definiert wurden, müssen die Systemobjekte in die jeweiligen Datenbanken eingetragen werden. Dies geschieht über den Menüpunkt „System“ „Systemobjekte anlegen“ bzw. den entsprechenden Schalter auf der Toolbar.



Hinweis: Je nach Größe bzw. Umfang der Datenbanken kann dieser Vorgang einige Zeit in Anspruch nehmen. Der Verlauf bzw. der aktuellen Status kann in der Statusleiste beobachtet werden. Treten Fehler beim Eintragen auf, wird dies im entsprechenden Abschnitt in der Statusleiste angezeigt. Sollte dies der Fall sein, wird nach Abschluß des Vorgangs ein weiteres Register „Systemobjekt-Protokoll“ angezeigt, über welches Einsicht über die Art des Fehlers genommen werden kann.



Hinweis: In der Regel treten keine Fehler beim Eintragen der Systemobjekte auf! Sind Systemobjekte in die Datenbank eingetragen und der Schalter wird erneut betätigt, werden selbstverständlich für alle Objekte Fehler generiert, weil sie ja schon vorhanden sind (siehe Bild). Sollte sich die Datenbankstruktur ändern bzw. neue Tabellen hinzukommen, kann der Menüpunkt „System“ „Systemobjekte anlegen“ jederzeit neu betätigt werden. Die Fehleranzeige kann dann ignoriert werden. **Nach dem Eintragen der Systemobjekte müssen unbedingt die Userrechte für den Replikations-User vergeben werden.**

6.2. Systemobjekte entfernen

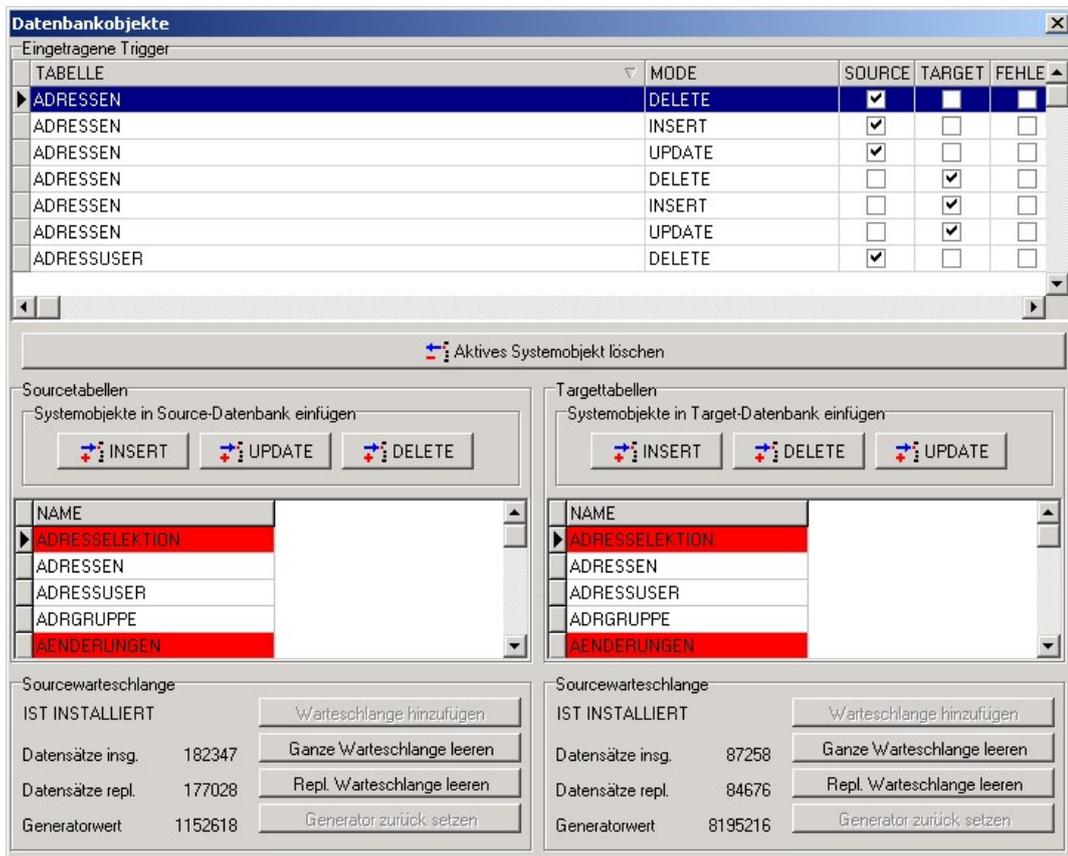
Schritt: Einmal eingetragene Systemobjekte lassen sich auch jederzeit über den Menüpunkt „System“-
 „Systemobjekte entfernen“ oder der entsprechende Schalter der Toolbar wieder aus der Datenbank entfernen. Dies ist vor allen Dingen dann wichtig wenn Sie an der Datenbankstruktur etwas verändern. Die Statusanzeige zeigt auch in diesem Fall den Verlauf wie den Status des Vorgangs an.

6.3. Systemobjekte prüfen

Schritt: Über den Menüpunkt „System“-„Prüfe Systemobjekte“ oder dem entsprechenden Schalter der Toolbar können die Systemobjekte überprüft und die Systemobjektdatenbank aktualisiert werden.


6.4. Systemobjekte

Systemobjekte sind Trigger, die den Replikationsablauf steuern. Alle angelegten Systemobjekte lassen sich auch Einzelnen verwalten. Hierfür wählt man den Menüpunkt „System“-„Objekte“. Dieser Menüpunkt dient dazu, einzelne Trigger aus der Definition bzw. den Datenbanken zu löschen bzw. hinzuzufügen.



The screenshot shows the 'Datenbankobjekte' dialog box. At the top, there is a table titled 'Eingetragene Trigger' with columns: TABELLE, MODE, SOURCE, TARGET, and FEHLE. The table contains several rows for the 'ADRESSEN' table with different modes (DELETE, INSERT, UPDATE) and checkboxes for SOURCE, TARGET, and FEHLE. Below the table, there are sections for 'Aktives Systemobjekt löschen', 'Sourcetabellen', and 'Targettabellen'. Each section has buttons for INSERT, UPDATE, and DELETE. Below these are two lists of table names, with 'ADRESSESELEKTION' and 'ÄNDERUNGEN' highlighted in red. At the bottom, there are two 'Sourcewarteschlange' sections, each with a 'Warteschlange hinzufügen' button and statistics for 'Datensätze insg.', 'Datensätze repl.', and 'Generatorwert'.

TABELLE	MODE	SOURCE	TARGET	FEHLE
ADRESSEN	DELETE	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
ADRESSEN	INSERT	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
ADRESSEN	UPDATE	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
ADRESSEN	DELETE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
ADRESSEN	INSERT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
ADRESSEN	UPDATE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
ADRESSUSER	DELETE	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Schritt: Die obere Tabelle beinhaltet sämtliche bereits eingetragenen Systemobjekte. In diesem Programmteil besteht nun die Möglichkeit, einzelne Trigger aus der Definition zu entfernen bzw. zur Definition hinzuzufügen. Auf diese Weise kann für jedes Ereignis (Hinzufügen, Ändern oder Löschen) und jede Tabelle eigens definiert werden, ob auf die jeweilige Aktion reagiert werden soll. Die Tabelle zeigt außerdem, in welcher der jeweiligen Datenbanken (Source oder Target) das Systemobjekt eingetragen ist bzw. ob beim Eintragen ein Fehler passiert ist.

Schritt: Die beiden unteren Tabellen stehen für die jeweiligen Tabellen der Datenbanken. In der Mitte sind die Schalter zum Hinzufügen der einzelnen Aktionen jeweils für Source und Target getrennt. Der Schalter in der Mitte zum Entfernen des Systemobjektes ist zum Löschen einzelner Systemobjekte gedacht.

Funktionsübersicht:

- Der Schalter „Entfernen“ in der Mitte des Dialoges löscht das aktuell markierte Systemobjekt der oberen Tabelle aus der Datenbank und der Definitionsdatenbank.
- Die linken Schalter fügen für die aktuell markierte Tabelle der Sourcetabellen das Systemobjekt in die Datenbank sowie in die Definitionsdatenbank ein.
- Die rechten Schalter fügen für die aktuell markierte Tabelle der Targettabellen das Systemobjekt in die Datenbank sowie die Definitionsdatenbank ein.
- Die Schalter im unteren Bereich sind eindeutig Bezeichnet und dienen zur Verwaltung der Warteschlange. Bevor Sie jedoch den kompletten Inhalt einer Warteschlange löschen sollten Sie sich im Klaren sein daß dadurch eventuell erforderliche Aktionen nicht mehr durchgeführt werden was unter Umständen eine Synchronisation der Datenbank zur Folge haben könnte.

7. Synchronisation

Bevor die Replikation von Datenbanken gestartet werden kann, muss die Datenbank synchronisiert werden.

Hinweis: Unter Synchronisation wird hier verstanden, dass die beiden Datenbanken einer replizierten Datenbank gleichkommen bzw. die Datenbestände identisch sind. Die Replikation arbeitet aktionsabhängig, d.h. dass jede Aktion (Ändern, Hinzufügen, Löschen) genau protokolliert wird. Diese Identität muss in beiden Datenbanken vorliegen. Für die Herstellung der Datenidentität wird die Synchronisation verwendet.

Die Synchronisation von Datenbanken kann also unabhängig vom Eintrag der Systemobjekte verwendet werden.

Natürlich kann hierfür auch eine Datapump verwendet werden. Die Verwendung einer herkömmlichen Datapump ist allerdings nur dann möglich, wenn keine Datensatz-Einschränkungen notwendig sind, da die meisten „Datapumps“ nur das jeweilige Kopieren von Tabellen ermöglichen.

Grundsätzlich bestehen 3 Möglichkeiten der Synchronisation:

- Die **Datenbanksynchronisation** beachtet alle Ausschlüsse von Tabellen sowie Datensatzeinschränkungen.
- Die **Synchronisation von Einzeltabellen** wiederum beinhaltet die Datensatzeinschränkungen.
- Mit der **Datensatzsynchronisation** lassen sich wie der Name schon sagt einzelne Datensätze synchronisieren.

7.1. Datenbanksynchronisation

Über den Menüpunkt „Synchronisation“- „Datenbank“ wird die Auswahl der drei verschiedenen Synchronisationsmethoden bereitgestellt. Der Aufruf ist ebenfalls über die Toolbar verfügbar.

- Komplette Synchronisation aller Tabellen in beide Richtungen
- Synchronisation aller Tabellen von Source nach Target
- Synchronisation aller Tabellen von Target nach Source

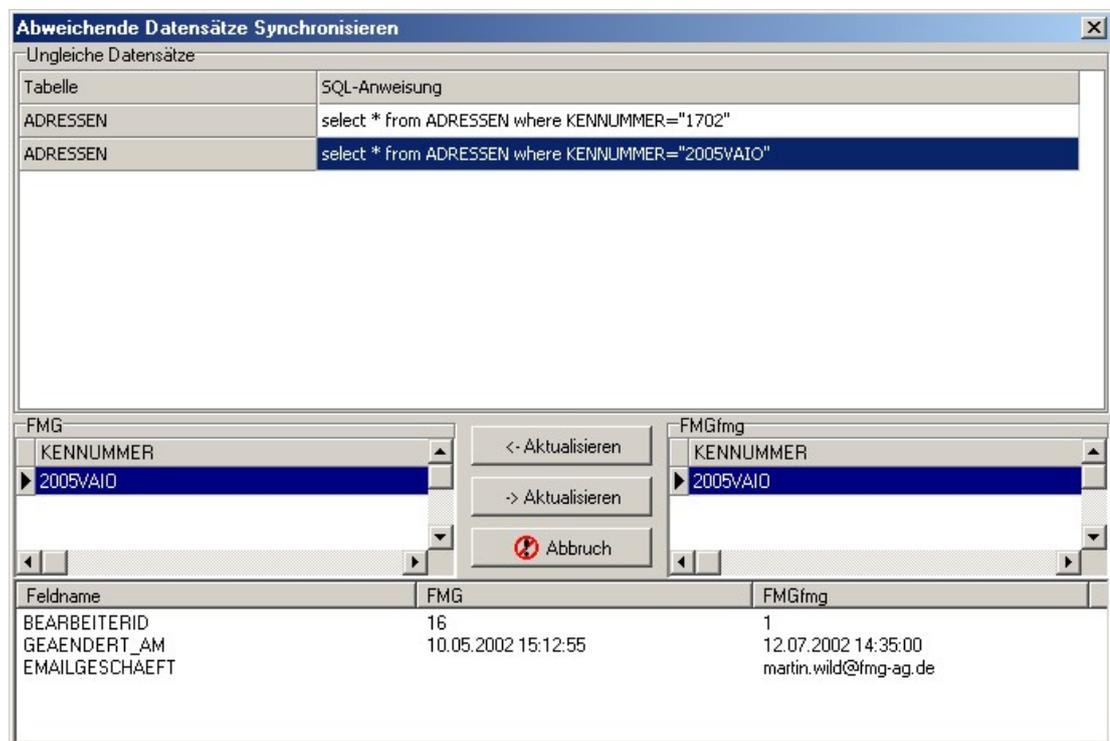


Unabhängig vom Synchronisationstyp werden alle Datensätze auf den gleichen Stand gebracht, die hinzugefügt, gelöscht oder geändert wurden. Zu beachten gibt es hierbei den Modus, mit dem voneinander abweichende Datensätze behandelt werden. Drei Optionen stehen zur Auswahl:

- Änderungsbestätigung nach Tabelle 
- Änderungsbestätigung nach Datenbank 
- Änderungen nach Synchronisationsrichtung 

Weichen Datensatzinhalte voneinander ab, legt der eingestellte Modus fest, wie diese Inhalte behandelt werden.

Hinweis: Am wenigsten Arbeit haben Sie mit *<Änderungen nach Synchronisationsrichtung>*. Dies bedeutet, dass alle abweichenden Datensätze der Lesedatenmenge in die Zieldatenmenge übernommen werden. Bei den zwei anderen Optionen erscheint je nach Einstellung (Nach jeder Tabelle bzw. nach Durchlauf einer Datenbankrichtung) folgender Dialog:



Die Tabelle im oberen Bereich beinhaltet die jeweiligen Abfragen, damit die abweichenden Datensätze anhand Ihrer Primärschlüssel gefunden werden. Beim Wechsel des Datensatzes wird die jeweilige Abfrage ausgeführt und die Ergebnisse der Abfrage in den Tabellen dargestellt.

Im unteren Bereich werden voneinander abweichende Felder der Datensätze dargestellt. Hier kann für jeden Datensatz die Synchronisationsrichtung festgelegt werden. Betätigt man einen Synchronisation-Schalter wird der jeweilige Datensatz in die gewählte Richtung synchronisiert. Der synchronisierte Datensatz verschwindet aus der Liste der abweichenden Datensätze. Nachdem der letzte abweichende Datensatz gleichgesetzt wurde schließt sich der Dialog automatisch und die Synchronisation wird fortgesetzt.

7.2. Einzeltabelle synchronisieren

Alle Ausführungen zur Synchronisation von Datenbanken gelten auch für die Synchronisation von einzelnen Tabellen. Eben auf eine Tabelle beschränkt. Über den Menüpunkt „Synchronisation“-„Einzeltabelle“ wie auch über die Toolbar könne sie die Synchronisation starten:

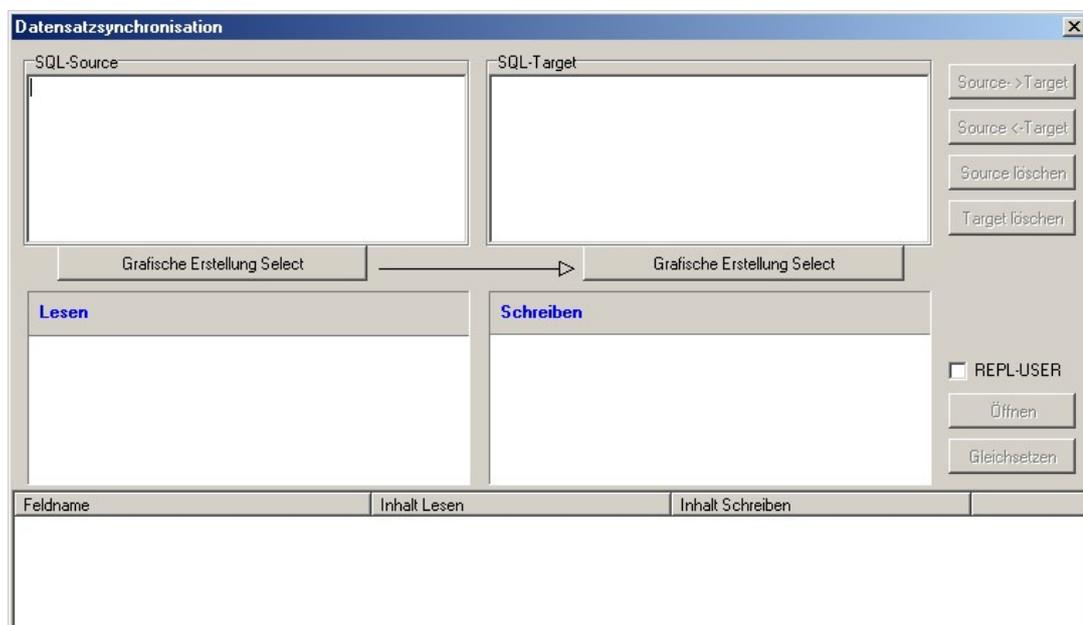
Source nach Target 

Target nach Source 

Vor dem Start der Synchronisation kann die zu synchronisierende Tabelle ausgewählt werden.

7.3. Datensatzsynchronisation

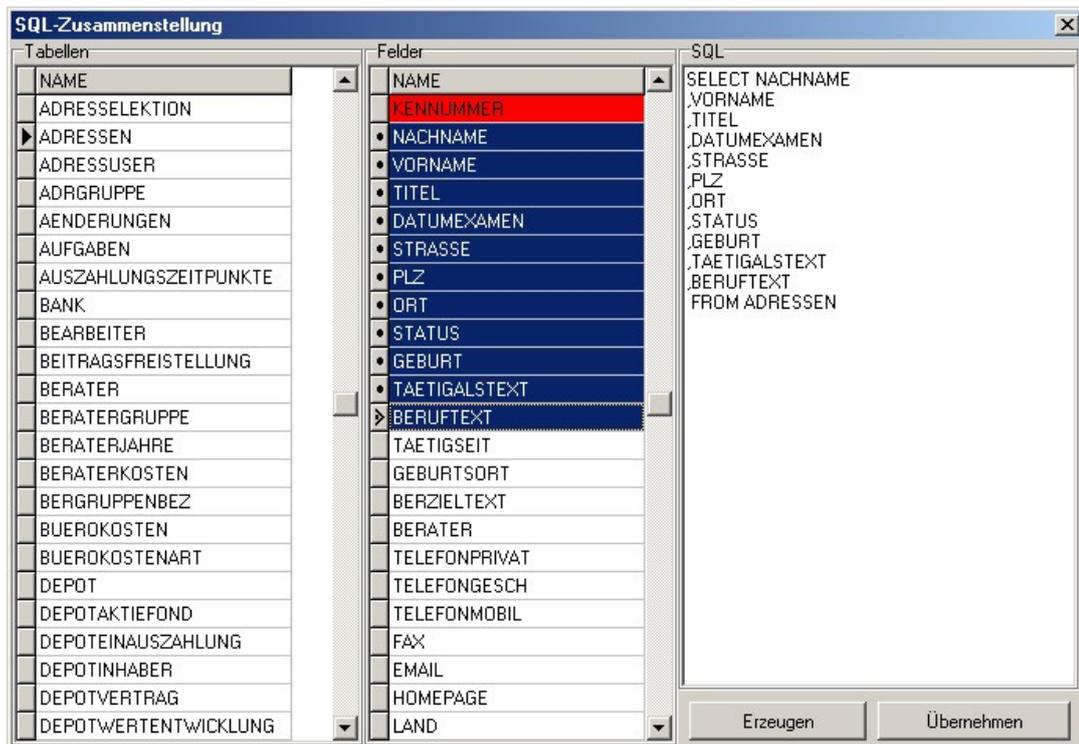
Mit Hilfe dieses Programnteils können Datensätze über Datenbanken hinweg abgeglichen werden. Über den Menüpunkt „Synchronisation“-„Datensatzsynchronisation“ wird folgender Dialog angezeigt:



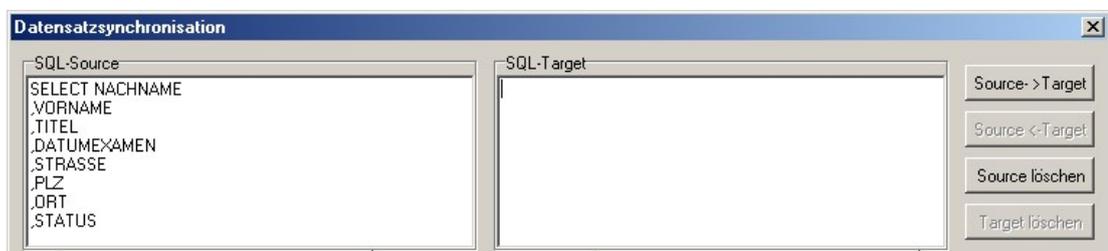
Über die Memofelder „SQL-Source“ bzw. „SQL-Target“ können die SQL-Statements eingegeben werden, die zum öffnen der Datenmenge benötigt werden. Es handelt sich dabei um Select-Abfragen.

Zum einen kann ein „Select * from adressen“ verwendet werden um beispielsweise Datensätze einer Adresstabelle abzugleichen. Je nachdem welche Spalten in der Select-Abfrage vorkommen werden diese Spalten abgeglichen.

Die nachfolgend abgebildete, grafische Erstellung der Select Abfrage soll die explizite Auswahl der Spalten erleichtern. Es können jeweils nur von der Struktur gleiche Tabellenspalten mit dem gleichen Namen und dem gleichen Typ abgeglichen werden.

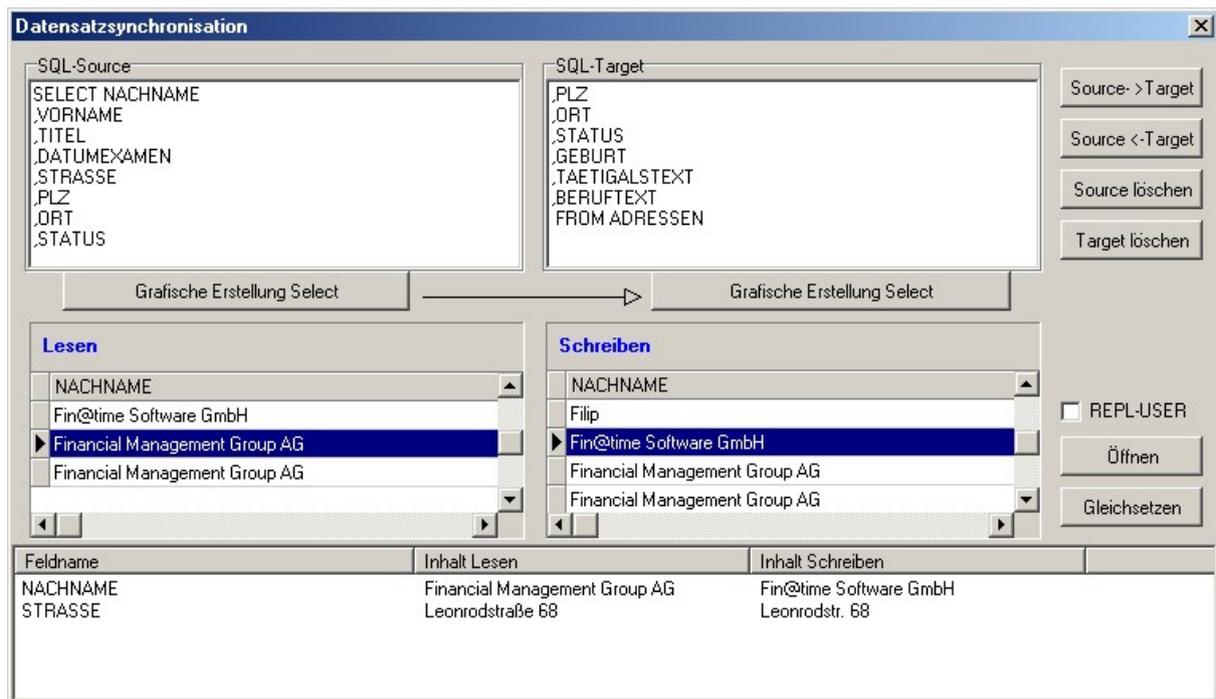


Zuerst wird die betreffende Tabelle ausgewählt. Die entsprechenden Felder werden in der rechten Tabelle angezeigt. Felder, die in Primärschlüsseln vorkommen, werden Rot dargestellt. In der Feldauswahl können nun beliebige Felder markiert und via Drag and Drop ins Memofeld gezogen werden bzw. der Schalter <Erzeugen> gedrückt werden. Über den Schalter <Übernehmen> kann der SQL-Ausdruck in die Datensatzreplikation übernommen werden. Der Dialog SQL-Zusammenstellung wird geschlossen.



Hinweis: Erst wenn für beide Datenbanken ein SQL-Ausdruck vorliegt, können die Abfragen geöffnet werden. Man kann über die Rechts angeordneten Schalter einfach SQL-Ausdrücke zwischen den beiden Memo-Feldern hin und her kopieren. Die angegebenen SQL-Ausdrücke dürfen für die Lesedatenmenge sowohl einen als auch mehrere Datensätze zurück liefern. Für die Datenmenge, in der geschrieben wird, kann auch kein Datensatz zurückgeliefert werden. In diesem Fall würde der Datensatz der Lesedatenmenge in die zu schreibende Datenmenge eingefügt werden. **Wichtig ist in diesem Fall, dass auch die Primärschlüssel-Spalten in beiden Datenmengen mit in den Select Statements enthalten sind.**

Nach dem erfolgreichen Öffnen der Abfragen kann in beiden Datenmengen über die jeweilige Tabelle navigiert werden. Über die Spaltentitel läßt sich die jeweilige Datenmenge Auf- und Absteigend sortieren.



Der untere Bereich zeigt die Felder der jeweiligen Datenmengen an die sich unterscheiden.

Hinweis: Vor dem „Gleichsetzen“ der Datensätze sollten Sie sich über die auszuführende Aktion genau bewußt sein. Müssen die Primärschlüssel mit übernommen werden? Sind in den Datenbanken Systemobjekte angelegt? Wenn Ja, ist es unter Umständen erforderlich, den Replikations-User für die Änderung zu verwenden damit die Änderung nicht wieder zurück repliziert wird.

Über den Schalter „Gleichsetzen“ wird der Lesedatensatz in die zu schreibende Datenmenge synchronisiert.

8. Replikation aktivieren

Bevor eine Replikationsdefinition verwendet werden kann, muss diese aktiviert werden. Erst wenn alle Einstellungen, Einschränkungen und Systemobjekte eingetragen wurden, sollte die Replikation als „aktiv“ markiert werden.

Schritt: Hierzu wird im Menü „Replikation“ – „Aktivieren“ ausgewählt oder der entsprechenden Schalter der Toolbar betätigt.  Aktivierte Replikationsdefinitionen werden in der Tabelle Rot dargestellt. Durch betätigen des gleichen Schalters wird der Eintrag wieder deaktiviert.

9. Replikation Starten

Es gibt zwei Programmvarianten:

Die erste Variante besteht aus einer Anwendung mit grafischer Oberfläche zur Verwendung unter Windows95,98 und Windows Millennium.

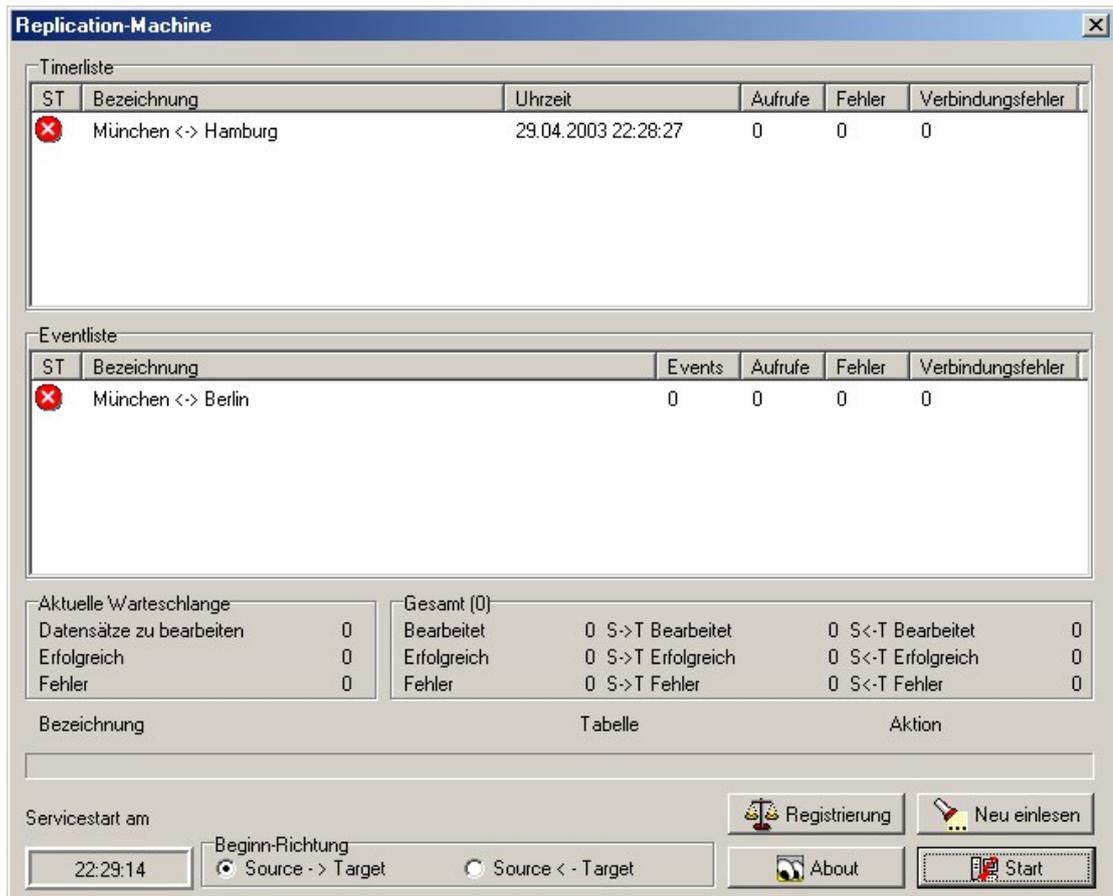
Die zweite Variante wird über das Menü Replikation-Serviceanwendung als Dienst gesteuert. Die Vorteile liegen auf der Hand:

Als Dienst bzw. Serviceanwendung auf einem Windows NT bzw. Windows 2000 System arbeitet die Replication Machine auch im abgemeldeten Zustand. Nach einem Absturz kann der Dienst automatisiert gestartet werden.

Die Installation des Dienstes (RPLSERVICE.EXE) kann entweder per Kommandozeile durchgeführt werden oder aber über das Menüsystem über Replikation – Serviceanwendung – Installieren. Der Dienst läßt sich von hier aus auch starten, stoppen und wieder deinstallieren.

Hinweis: Bitte beachten Sie, dass auch der Dienst in der Sharewareversion nur 1000 Datensätze repliziert. (siehe 2. Shareware, Vollversion und Registrierung)

Nach Programmstart zeigt sich die Replication - Machine als Programmfenster, ist aber noch inaktiv. Sie wird erst aktiviert bzw. gestartet durch drücken des Schalters.



Aus der Administrationssoftware heraus kann die Replication-Machine (Replikationssoftware) auf zwei verschiedene Arten gestartet werden:

9.1. Replikation im ständigen Netzwerkbetrieb

Die Replication-Machine startet, ohne zunächst die Replikation zu starten. Es werden alle aktiven Datenbankeinträge eingelesen und angezeigt. Der Start der Replikation erfolgt im Programm. Diese Art des Starts ist für die Replikation von Netzwerkdatenbanken mit festen Standorten gedacht.

9.2. Replikation von der Serverdatenbank zum mobilen Rechner

Zur Replikation von Serverdatenbanken mit mobilen Rechnern ruft man die Replikationssoftware mit Parametern derart auf, dass die Abarbeitung der Warteschlangen nur einmal abgearbeitet wird. Der Start würde von einem Notebook wie folgt durchgeführt werden:

IBEASYRPLMACHINE.EXE_(START)_(CONFIGID)

Syntax: Die in Klammern dargestellten Worte beinhalten die Parameter. Nach dem Programmnamen muss ein Leerzeichen folgen, dann das Wort START, wieder ein Leerzeichen und die CONFIGID der Datenbank die abgearbeitet werden soll.

Jede Replikation einer mobilen Einheit zu einer Serverdatenbank bedingt einen Eintrag in der Configdatenbank (siehe 3.1 Replikation erstellen). Jeder Datenbankeintrag in der Configtabelle erhält eine ID, eben den Primärschlüssel. Dieser Primärschlüssel dient zur Identifikation der abzuarbeitenden Replikation. Beim Start der Replication Machine aus Ihrer Datenbankanwendung muß dieser Primärschlüssel als Parameter (CONFIGID) verwendet werden.

Hinweis: Achten Sie bei einer bidirektionalen Replikation darauf daß beide Datenbank-Warteschlangen abgefragt werden.

Beispiel für die Config-ID 2:

```
SELECT COUNT(KENNUMMER) FROM REPLIZIEREN_2 WHERE REPLIZIERT="F"
```

Ergibt die Abfrage ein Ergebnis größer Null, muss die Replikationssoftware gestartet werden.

9.3. Die Replication Machine als Anwendung im Einsatz.

In diesem Beispiel ist eine timerbasierte Replikation zwischen Berlin und München sowie eine eventbasierte Replikation zwischen Hamburg und München definiert. Das Abarbeiten der Replikationswarteschlange kann unter Umständen - je nach Anzahl der Datensätze - einige Zeit in Anspruch nehmen. Der Replikationsvorgang lässt sich durch drücken des „Unterbrechen“ Schalters unterbrechen.

The screenshot shows the 'Replication-Machine' application window. It contains two main lists: 'Timerliste' and 'Eventliste'. Below these are statistics for the current queue and a summary of total operations. At the bottom, there are control buttons for 'Unterbrechen', 'Registrierung', 'Neu einlesen', 'Beginn-Richtung', 'About', and 'Stop'.

Timerliste					
ST	Bezeichnung	Uhrzeit	Aufrufe	Fehler	Verbindungsfehler
	München <-> Hamburg	29.04.2003 22:20:42	0	0	0

Eventliste					
ST	Bezeichnung	Events	Aufrufe	Fehler	Verbindungsfehler
	München <-> Berlin	0	0	0	0

Aktuelle Warteschlange		Gesamt (1)			
Datensätze zu bearbeiten	54621	Bearbeitet	54621	S->T Bearbeitet	54621
Erfolgreich	221	Erfolgreich	221	S->T Erfolgreich	221
Fehler	3	Fehler	3	S->T Fehler	3

Bezeichnung München <-> Hamburg Tabelle PROVISIONAUSZAHLUNC Aktion UPDATE

Servicestart am 29.04.2003 22:20:14

22:22:24 Source -> Target Source < - Target

10. Fehlerbenachrichtigung

Das beiliegende Programm ErrorEventMessenger ist ein Fehlerbenachrichtigungsprogramm. In der Regel sollte es am PC des Datenbank-Administrators installiert und über Autostart ausgeführt werden. Es ist eine Traybar-Icon-Anwendung welche nur dazu gedacht ist im Fehlerfall das Easy-Interbase-Replication-Verwaltungsprogramm sowie die Fehlerbehandlungsseite und den jeweiligen Fehlerdatensatz aufzurufen. Beim Öffnen der Benutzeroberfläche wird lediglich die Anzahl der abgefangenen Events, also Allgemeinfehler und Datensatzfehler angezeigt.

11. Fehlerbehandlung

Nur wenn Fehler für einen aktuell gewählten Replikations-Konfigurations-Datensatz vorhanden sind, werden die entsprechenden Fehlerregister angezeigt. Insgesamt ist die Fehler- wobei man auch sagen könnte Konflikt-Beseitigung eins der komplexesten Themen beim replizieren von Datenbanken. Wir möchten Ihnen mit EASY-INTERBASE-REPLICATION auch ein Produkt zum einfach Konfliktmanagement an die Hand geben. Wir unterscheiden zwischen Allgemeinfehlern (in der Regel Verbindungsfehler) und Datensatzfehlern. Entsprechend ist auch die Replikations-Fehler-Protokoll-Seite in zwei Bereiche aufgeteilt.

The screenshot shows the 'Easy Interbase Replication' application window. It has a menu bar with 'Datenbank', 'System', 'Synchronisation', 'Replikation', and 'Replikationsprotokoll'. Below the menu is a toolbar with various icons. The main area is divided into two sections: 'Allgemeinfehler (132)' and 'Datensatzfehler (5065)'. Each section contains a table of error records with columns for ID, ZEIT, TEXT, and SOURCEOF. Below each table are buttons for 'Filtern', 'Löschen', 'Alle löschen', and 'Detail'. At the bottom, there are tabs for 'Grundeinstellung' and 'Replikations-Fehler-Protokoll', and a status bar showing 'verlauf', 'Gesamt', 'Erfolgreich', 'Ausgelassen', and 'Fehler'.

ID	ZEIT	TEXT	SOURCEOF
159	06.05.2003 15:55:04	Connect zur Targetdatenbank!Unbekannte Datenbank	S
158	06.05.2003 15:49:47	Connect zur Targetdatenbank!Unbekannte Datenbank	S
157	06.05.2003 15:46:38	Connect zur Targetdatenbank!Unbekannte Datenbank	S
156	06.05.2003 15:43:11	Connect zur Targetdatenbank!Unbekannte Datenbank	S
155	04.05.2003 18:19:13	Öffnen der Replikationstabelle in Targetdatenbank!Allgemeiner SQL-Fehler	T

ID	ZEIT	TABELLE	TEXT	WIEDERHOLUNG	SOURCEOF
28481	07.05.2003 10:24:37	GROUPAPF	Fehler beim Löschen! Gelöschter Datensatz ist noch	2	T
28480	07.05.2003 10:23:16	GROUPAPF	Fehler beim Löschen! Gelöschter Datensatz ist noch	1	T
28479	07.05.2003 10:22:15	GROUPAPF	Fehler beim Löschen! Gelöschter Datensatz ist noch	0	T
28478	07.05.2003 10:19:29	GROUPAPF	Fehler beim Löschen! Gelöschter Datensatz ist noch	2	T
28477	07.05.2003 10:18:38	GROUPAPF	Fehler beim Löschen! Gelöschter Datensatz ist noch	1	T

Über die jeweiligen „Filtern“-Schalter lassen sich Filterdialoge aufrufen mit denen die jeweiligen Tabellen gefiltert werden können. Die Schalter Löschen, löschen jeweils den aktuellen Fehlerdatensatz in der Tabelle und der Schalter „Alle Löschen“ löscht jeweils die gesamte Tabelle bzw. gefilterte Datenmenge.

11.1. Allgemeinfehler

Bei den Allgemeinfehlern ist ein Schalter „Detail“ der zur detaillierten Ansicht des Fehlers führt. Bei Allgemeinfehlern kann nicht programmgesteuert auf den Fehler reagiert werden, nur Sie können anhand der detaillierten Fehlerbeschreibung gegen den Fehler vorgehen bzw. darauf reagieren. Nach Auftreten dieser Fehlerart werden aber auch keine Replikationswarteschlangen abgearbeitet, was bedeutet dass diese unbehandelt bleiben (Der Warteschlangenstapel bleibt erhalten bis der Fehler beseitigt wurde oder nicht mehr auftritt).

Beispiel :

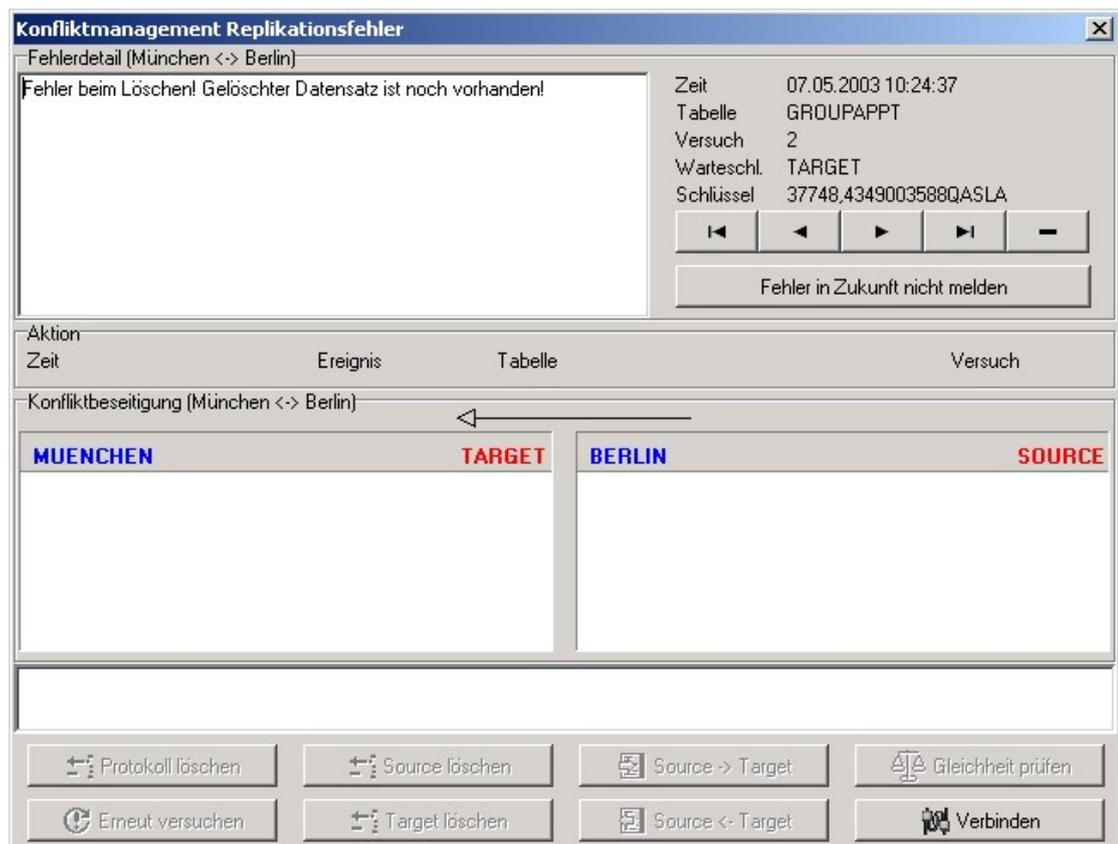


Dieser Fehler zeigt dass die Netzwerkverbindung zur Targetdatenbank unterbrochen war.

11.2. Behandlung von Datensatzfehlern

Hinweis: Grundsätzlich sollte die Replikationssoftware Fehlerfrei laufen. Einige Fehler können jedoch programmbedingt bzw. aufgrund der Zeitverschiebung bei der Replikation eintreten. Insofern kann es auch leicht möglich sein, dass bestimmte Fehler in Wirklichkeit keine Fehler sind. Es liegt nun an Ihnen, Fehler zu beseitigen bzw. an der Replikationssoftware mitzuteilen, wo es sich wirklich um Fehler handelt über die sie in Zukunft benachrichtigt werden wollen.

Durch drücken des Schalters Konfliktbeseitigung kann auf den aktuell markierten Fehler reagiert werden, der Konfliktmanagement-Dialog öffnet sich.



Nachfolgend eine Tabelle mit möglichen Datensatzfehlern:

Nummer	Aktion	Beschreibung
1	Hinzufügen	Fehler beim Posten des hinzugefügten Datensatzes
2	Hinzufügen	Datensatz in Ziel mit anderen Inhalten bereits vorhanden
3	Hinzufügen	Kein Lesedatensatz vorhanden
4	Hinzufügen	Mehrere Lesedatensätze vorhanden
5	Ändern	Fehler beim Posten des geänderten Datensatzes
6	Ändern	Fehler beim Hinzufügen wenn Einfügedatensatz nicht existiert!
7	Ändern	Es sind mehr als ein Datensatz im Ziel vorhanden!
8	Ändern	Kein Lesedatensatz vorhanden!
9	Ändern	Mehrere Lesedatensätze vorhanden
10	Löschen	Fehler beim Löschvorgang
11	Löschen	Kein Datensatz zum Löschen vorhanden
12	Löschen	Es wurden mehrere Datensätze gefunden
13	Löschen	Gelöschter Datensatz beim Lesen noch vorhanden
14	Suche Modul Source	Fehler beim Suchen nach dem Sourcedatensatz
15	Suche Modul Source	Fehler beim Suchen nach dem Targetdatensatz
16	Suche Modul Target	Fehler beim Suchen nach dem Sourcedatensatz
17	Suche Modul Target	Fehler beim Suchen nach dem Targetdatensatz

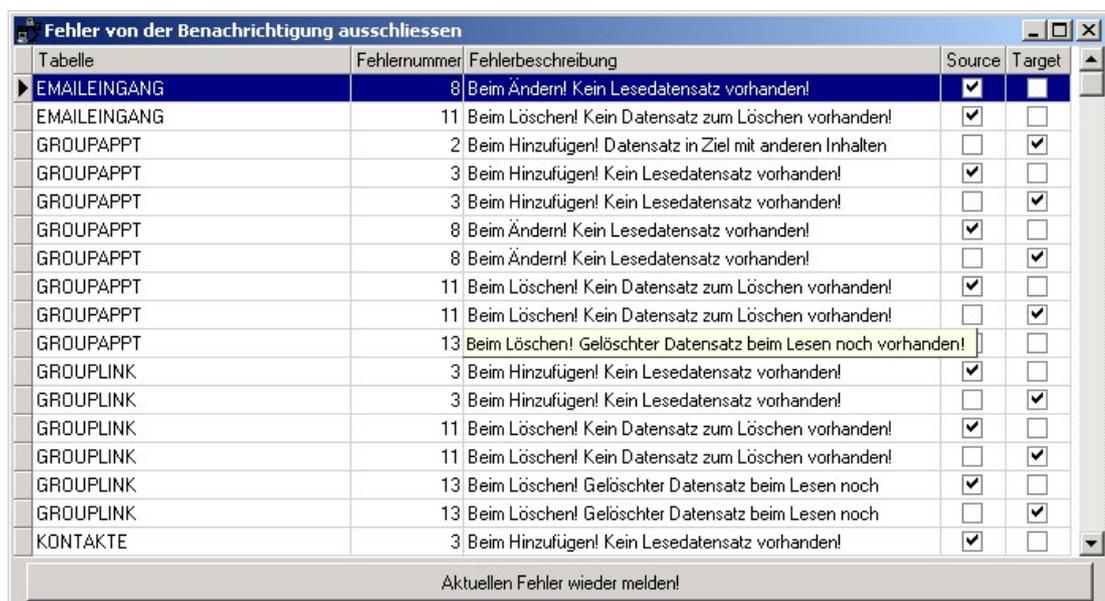
Tabelle Nr. 1.

Jede Fehlernummer kann Tabellenspezifisch von der Fehlerbenachrichtigung ausgeschlossen werden. Ein praktisches Beispiel:

Beispiel: Ein bestimmter Bereich einer Software wurde so programmiert, dass statt einer Änderung des Datensatzes dieser Datensatz erst gelöscht, und gleich danach wieder neu mit den gleichen Primärschlüsseln erzeugt wird. In diesem Fall wird natürlich jede Aktion in der Warteschlange aufgenommen. Der Löschvorgang würde einen Fehler mit der Nummer 13 auslösen (Gelöschter Datensatz beim Lesen noch vorhanden)

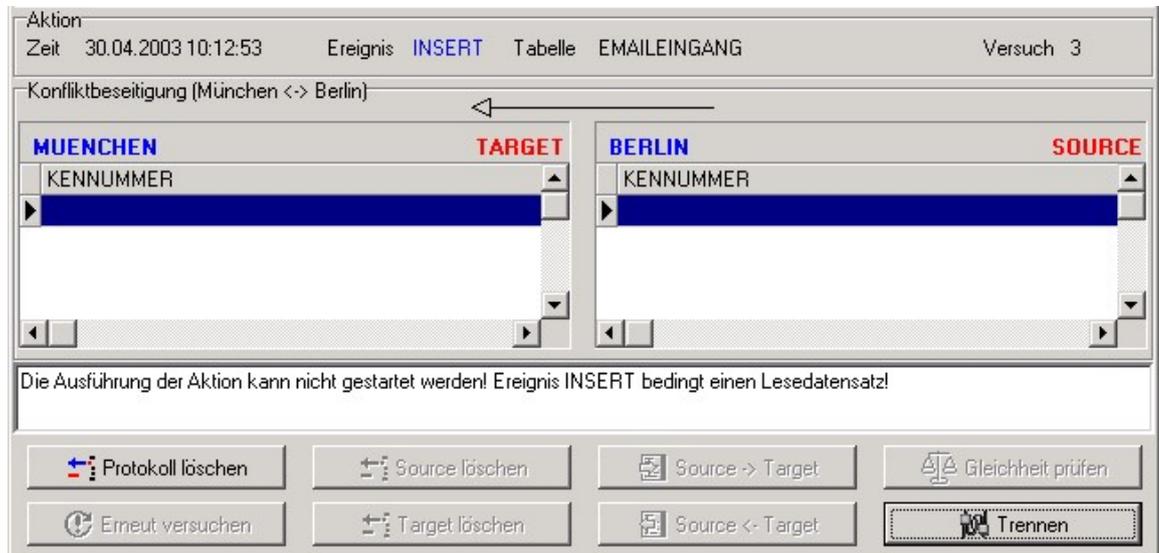
Anhand der fehlerbeschreibung sollten Sie eventuell schon entscheiden wie sie vorgehen, entweder die **Fehlerbenachrichtigung ausschließen:**

Der jeweilige Ausschluss läßt sich einfach durch betätigen des Schalters „Fehler in Zukunft nicht melden“ aus der Fehlerbenachrichtigung ausschließen. Der Ausschluß eines Fehles läßt sich jederzeit wieder rückgängig machen. Dazu muß man über das Hauptmenü – System die Funktion „Ausschlüsse Fehlerbenachrichtigung“ aufrufen.



Oder auf den Fehler reagieren:

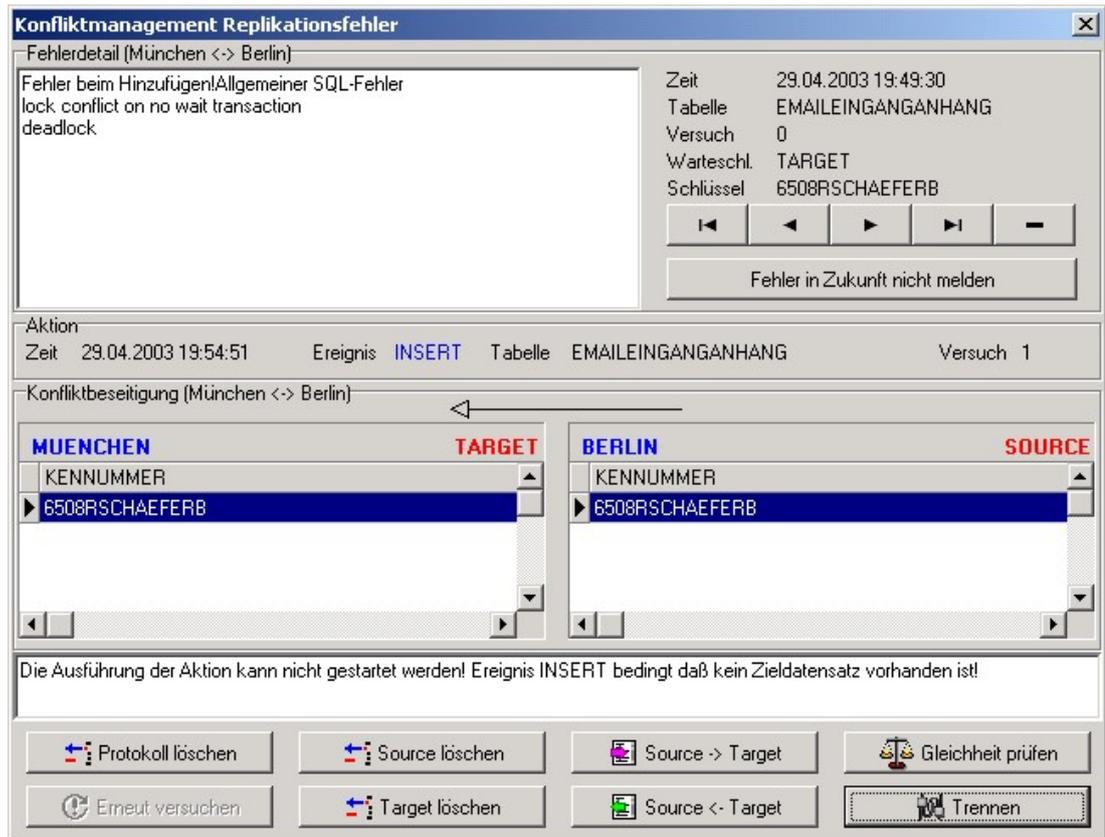
Dazu betätigt man zunächst den Schalter „Verbinden“, welcher die Verbindung zu den beteiligten Datenbanken herstellt und die jeweiligen Datensätze der Aktion in beiden Datenbanken sucht und anzeigt. Je nachdem ob ein erneuter Versuch möglich ist oder anders auf den Konflikt reagiert werden kann oder nicht werden die Schalter der Konfliktbeseitigung freigegeben oder inaktiv dargestellt



Dieses Beispiel zeigt eine INSERT-Aktion wo aber in keinem der beiden Datenmengen ein zugehöriger Datensatz gefunden wurde. Somit ist auch keine Aktion möglich außer das Protokoll zu löschen.

Hinweis: Bitte beachten Sie bei diesem Dialog auch die jeweilige Replikationsrichtung sowie die Anzeige der Aliasse und wer Source bzw. Target ist. Nur so kann mit den Schaltern der Konfliktbeseitigung Richtig umgegangen werden.

Der folgende Dialog beschreibt den klassischen Fall eines Deadlocks (Der Datensatz der repliziert werden soll ist von einem anderen User gesperrt). Sobald die Verbindung hergestellt wurde und wie in diesem Beispiel zwei Datensätze vorhanden sind ist ein wichtiger Aspekt die Datensätze auf Gleichheit zu prüfen!



Anhand der Information ob Datensätze in Ziel oder Target gleich oder nicht gleich sind können Sie entscheiden wie in dem Fall vorgegangen werden soll. Im obigen Fall kann außer der Funktion „Erneut Versuchen“ (Ein Insert in diesem Fall ist nicht Möglich da ja der Datensatz bereits in Source und Target vorhanden ist) entschieden werden was gemacht wird:

- Source löschen
- Target löschen
- Source nach Target
- Target nach Source

Nach erfolgreicher Abarbeitung einer Aktion wird der Fehlerdatensatz aus dem Protokoll entfernt.

Index

A

Abweichende Datensätze 12
Abweichende Felder 12, 15
Aktivieren 5, 16
Alias 6
Alle Tabellen 11
Allgemeinfehler 20
Arbeitsweise 6
Asynchrone Replikation 6
Aufrufparameter 17
Ausgeschlossene Tabellen 8

B

Benachrichtigung 6

D

Datapump 11
Datenbankstruktur ändern 10
Datenbanksynchronisation 11
Datenmengen 14
Datensätze 13
Datensatzeinschränkung 8
Datensatzfehler 20
Datensatzsynchronisation 13

E

Einmalreplikation 7
Einzelne Systemobjekte 10
Ereignisgesteuert 7
ErrorEventMessenger 6
Erstellen 7

F

Fehler beim Eintragen 9
Fehlerbehandlung 19
Fehlerbenachrichtigung 19, 21
Fehlernummern 21
Felder 14
Feste Standorte 6
Freischaltung 5

G

Generator 3
Getrennte Installation 6
Gleichsetzen 15
Grundsätzliches 6

H

Hostname 3

I

IBREPL.GDB 6
Intervallgesteuert 7

K

Komplette Synchronisation 11
Konfliktbeseitigung 20
Kopieren von Tabellen 11

L

Lesedatenmenge 14

M

Mobile Computer 6

N

Netzwerkbetrieb 17
Netzwerkverbindung 20

Ö

Öffnen 8

P

Parameter 17
Primärschlüssel 3, 14, 15

R

Registrierung 5
Replikation starten 16
Replikationsdefinition erstellen 7
Replikations-User 7, 9, 15

S

Select 13
Seriennummer 5
Server 6
Shareware 5
SQL-Zusammenstellung 14
Synchrone Replikation 6
Synchronisation 11
Synchronisationsrichtung 12
Synchronisationstyp 12
Systemobjekt entfernen 11
Systemobjekt hinzufügen 11
Systemobjekte 6, 15
Systemobjekte eintragen 9
Systemobjekte entfernen 10
Systemobjekte prüfen 10
Systemobjekte verwalten 10
Systemobjekt-Protokoll 9

T

Tabelle synchronisieren 13
Tabellenausschluss 8
Trigger 3, 10

U

Unterschiede 15

V

Verbinden 8
Vollversion 5
Voraussetzungen 3

W

Warteschlange abfragen 17