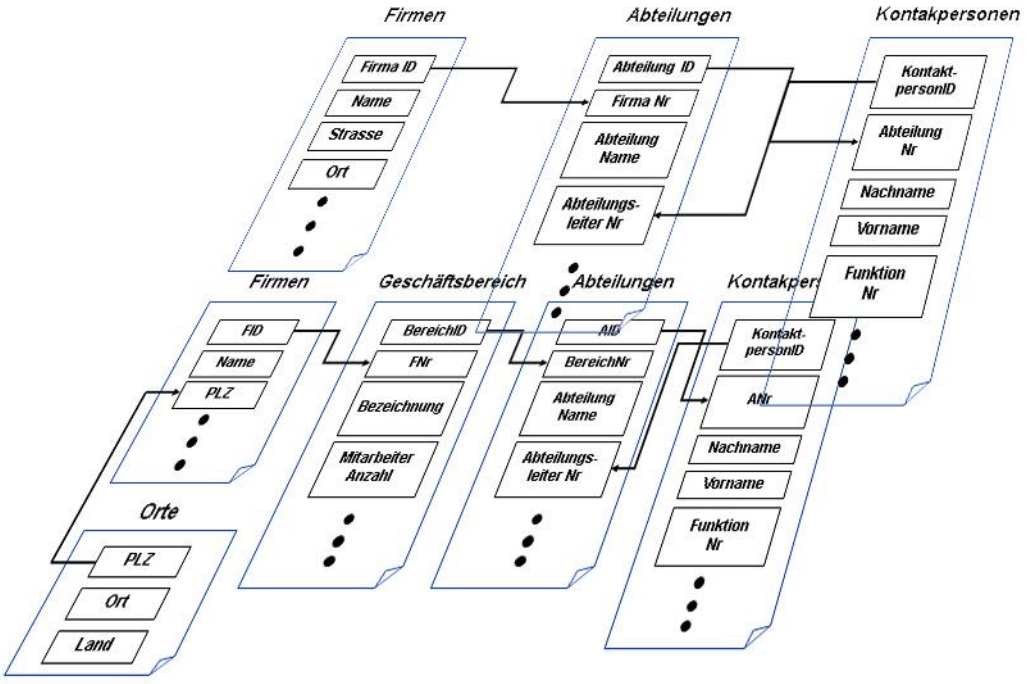
ComfortsQL



Schluss mit der redundanten Arbeit, wenn sich etwas an der Tabellenstruktur ändert

ComfortsQL-Relationic

Ihr Autopilot durch ein stets agiles relationales Datenmodell

Ein für die Zukunft stets gerüstetes Datenmodell

Unser "ComfortsQL-Relationic“ verkürzt und vereinfacht die DB-Programmierungen, unterstützt extrem die Wartung eines relationalen DBMS und ist die kostenschonende Lösung, unterschiedliche Datenbanken zu integrieren.

Selbst bei bereits im Einsatz befindlichen DB-Systemen, müssen bei auftretenden Änderungen an der Tabellenstruktur keine SQL-Statements mehr umgeschrieben werden, da diese automatisch an das aktuelle Datenmodell angepasst werden.

Das ComfortsQL-Relationic ist ein Softwaremodul für SQL-Anfragen.

Dieses Softwaremodul liest sich die Tabellenstrukturen und die Tabellen-Beziehungen eines Datenmodells in einen Graphen ein.

Zu jedem Ausführungszeitpunkt ergänzt das Softwaremodul die SQL-Statements um die aktuellen relationalen Operationen entsprechend dieses Graphen bzw. passt die betreffenden SQL-Statements an.

Das bedeutet in der Praxis, dass SQL-Statements ohne relationale Verarbeitungsbereiche formuliert werden können und bereits vorhandene SQL-Statements automatisch adaptiert werden.

Unsere Software ermöglicht somit, dass Änderungen am Datenmodell jederzeit und problemlos z.B. für Optimierung am Datenmodell aufgrund von auftretenden Geschwindigkeitsverlusten oder bei der Einbindung von neuen Softwarebereichen vorgenommen werden können.

Neue Anforderungen, Unternehmensumwelten oder Datenbanken und die daraus resultierenden Änderungen am Datenmodell können einfach integriert werden - ohne dass veraltete SQL-Statements umgeschrieben werden müssen.

Kein neuer SQL-Dialekt sondern automatisierte Relationen

Die ComfortsQL-Relationic nutzt „ComfortsQL-Statements“ zur Verarbeitung der Daten.

Ein „ComfortsQL-Statement“ ist ein von Relationen befreites SQL-Statement für das jeweilige DBMS.

Die verwendeten ComfortsQL-Statements sind aufgrund der wegfallenden Relationen-Angaben kürzer und übersichtlicher als die entsprechenden SQL-Statements.

Dies erleichtert das Programmieren von SQL-Statements bzw. das Adaptieren dieser entfällt, da die SQL-Statements automatisch an das aktuelle Datenmodell angepasst werden.

Das Kennzeichen eines „ComfortsQL-Statement“ ist ein CSELECT anstatt des SELECT-Befehls.

Dieser neue Befehl CSELECT ermöglicht die Formulierung von SQL-Statements ohne den FROM- und JOIN-Bereich.

Es müssen lediglich die Tabellen- und -spalten-Namen im ComfortsQL-Statement vorhanden

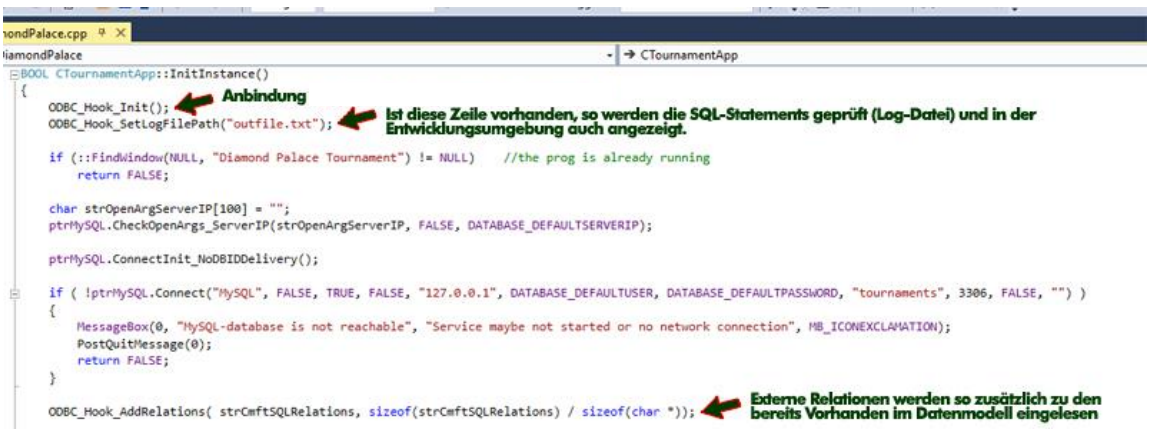
sein z.B. CSELECT Firmen.Firmenname, Firmen.Strasse, etc., die, falls es zu Änderungen am Datenmodell durch Namensänderung kommt, automatisch durch die aktuellen Tabellen-bzw. –spalten Bezeichnungen z.B. CSELECT FirmenLiebling.Firmenname, FirmenLiebling.Strasse ersetzt werden.

Die ComfortsQL-Relationic erkennt und behandelt Änderungen am Zugriffspfad (FROM, JOIN) in gleicher Weise.

Angenommen, der Zugriffspfad eines SQL-Statements auf ein konkretes Datenmodell findet von „A->B->C->D->E->F“ statt, und dieser wird nun auf ein anderes Datenmodell von „A->X->Y->D->H->I->F“ geändert,

so wird diese Änderung und somit das aktuelle Datenmodell automatisch durch unsere Software für das DB-System in Echtzeit übernommen und dem User zur Verfügung gestellt.

ComfortsQL-Relationic im ODBC-Treiber




```
BOOL CTournamentApp::InitInstance()
{
    ODBC_Hook_Init();
    ODBC_Hook_SetLogFilePath("outfile.txt");

    if (::FindWindow(NULL, "Diamond Palace Tournament") != NULL)    //the prog is already running
        return FALSE;

    char strOpenArgServerIP[100] = "";
    ptrMySQL.CheckOpenArgs_ServerIP(strOpenArgServerIP, FALSE, DATABASE_DEFAULTSERVERIP);

    ptrMySQL.ConnectInit_NoDBIDDelivery();

    if ( !ptrMySQL.Connect("MySQL", FALSE, TRUE, FALSE, "127.0.0.1", DATABASE_DEFAULTUSER, DATABASE_DEFAULTPASSWORD, "tournaments", 3306, FALSE, "") )
    {
        MessageBox(0, "MySQL-database is not reachable", "Service maybe not started or no network connection", MB_ICONEXCLAMATION);
        PostQuitMessage(0);
        return FALSE;
    }

    ODBC_Hook_AddRelations( strCmftSQLRelations, sizeof(strCmftSQLRelations) / sizeof(char *));
```

Die Software selbst ist in unserem ComfortsQL-Relationic-ODBC-Treiber integriert.

Dieser wird dem aktuellen ODBC-Treiber des DB-Systems vorgeschaltet. Die Performance-Kosten sind minimal, da praktisch alle Befehle direkt an den eigentlichen ODBC-Treiber weitergeleitet werden, außer jene, welche SQL-Statements als Parameter übergeben bekommen.

(Beispielsweise wird SQLFetch direkt weitergeleitet, aber SQLExecDirect wird von unserem ODBC-Treiber insofern verarbeitet, als dass ein mitgegebenes ComfortsQL-Statement zu einem für das aktuell zugrundeliegende Datenmodell umgebaut wird.)

Ein Entwickler verwendet seine ihm bekannten ODBC-Befehle wie gehabt weiter und braucht so auch keine neuen Befehle zu lernen.

Er kann von Relationen befreite und damit vom Datenmodell unabhängige SQL-Statements erstellen, da die Relationen im Anschluss auch für eine geänderte Datenstruktur automatisch wieder eingefügt werden.

Somit kann ein Umstieg auf das ComfortsQL-Relationic durch die Installation unseres ODBC-Treiber sehr einfach durchgeführt werden, um die Vorteile, die ein flexibles Datenmodell bietet, nutzen zu können und das in sämtlichen Programmiersprachen z.B. C / C++, PHP usw.

Features ComfortsQL-Relationic

- **Zukunftsorientiert und beständig**
  Auch wenn Computer und vor allem Server immer schneller werden, so verhalten sich Datenbanken trotzdem bei ansteigenden Datenmengen nicht so. Ein wesentlicher Grund dafür sind Datenmodelle, die bei der Applikationsentwicklung erstellt wurden und oftmals den gestiegenen Anforderungen der heutigen Zeit nicht mehr entsprechen.
  Durch unser Softwaremodul können Umstellungen an einem Datenmodell jederzeit vorgenommen werden, selbst bei Applikationen, welche an ein veraltetes Datenmodell gebunden sind. Somit sind Performance-Probleme schnell zu lösen und neue Anforderungen leicht implementiert.

- **Große Kostenersparnis bei Wartung und Projekten**
  Da das Umschreiben von SQL-Statements entfällt und die Applikationen zur Laufzeit automatisch an das neue Datenmodell angepasst wird, erspart man sich technisches Personal für diese Tätigkeit.

- **Kürzere Projektdauer**
  Neue Datenmodelle können, aufgrund der neugewonnenen Flexibilität der SQL-Statements, wesentlich schneller im Echtbetrieb eingeführt werden. Anfragen werden übersichtlicher und kürzer, da sie nur für die jeweilige Aufgabe direkt relevante Tabellen und deren Spalten enthalten.

- **Heterogene Datenbanken können einfach und kostenschonend integriert werden**

  Schritt 1:
  Es werden der ersten Datenbank alle relationalen Operationen der bereits vorhandenen SQL-Statements entfernt.

  Schritt 2:
  Es werden in die von Relationen bereinigten SQL-Statements die entsprechenden relationalen Operationen der zweiten Datenbank eingefügt.

  Ergebnis:
  Es wird die Abfrage auf Basis der mit den zweiten relationalen Operationen versehenen Datenbankanweisung ausgeführt.

- **ODBC-Treiber - leichter Umstieg - maximale Flexibilität und Sicherheit**
  Um die Vorteile ComfortsQL in eigenen Programmen nutzen zu können, reicht ein Verweisaustausch auf den ComfortsQL-Relationic-ODBC-Treiber anstatt des Datenbank-ODBC-Treibers. Der ODBC-Treiber liest automatisch alle für ihn relevanten Informationen ein, und benötigt für diesen Vorgang nur eine Leseberechtigung. Als Quelle dient die Datenbank direkt, welche mit einer Konfigurationsdatei auch erweitert werden kann

- **ComfortsQL-Relationic ersetzt die Mastertabelle**
  Unsere ComfortsQL-Relationic kann auch als virtuelle Mastertabelle verwendet werden, da der Zugriff auf alle Spalten der Tabellen des Datenmodells stets möglich ist aber keine kartesischen Produkte im Arbeitsspeicher gehalten werden müssen.

Systemvoraussetzungen

DBMS - Datenbankmanagementsysteme

Oracle ab Version 11.0
DB2 ab Version 9.5
MySQL ab Version 5.1
MsSQLServer ab Version 2008
Access ab Version 2003
SQLite ab Version 3.6

Intel Pentium ab 500 MHz bzw. gleichwertiger AMD-Prozessor
Arbeitsspeicher: 50 MB; **Festplattenplatz:** 10 MB
Betriebssysteme: Windows XP, Windows Vista, Windows 7, Windows 8, Windows 10, Windows Server 2000, 2003, 2005, 2008, 2010
Datenbanken: Zugangsdaten für Leseberechtigung müssen vorhanden sein. Wenn ein Zugriff über ODBC erfolgt, muss der entsprechende ODBC-Treiber installiert sein.

MySQL:	User, Passwort, Host, Port
MsSQLServer:	User, Passwort, Server
Oracle:	User, Passwort, Service
IBM DB2:	User, Passwort, ServerIP
MsAccess:	Verzeichnis und Name der MsAccess-Datenbank
SQLite:	Verzeichnis und Name der SQLite-Datenbank

ForeignKeys: Die Foreignkeys sollten in der Datenbank definiert sein, beispielsweise in den Constraints oder in der Tabelle der Relationen.
Berechtigungen: Für die Installation müssen Administrator-Rechte vorhanden sein, da bei der Installation ein Uninstall-Eintrag in der Registry erstellt wird.
Zum Ausführen des Programms muss der Start einer .exe-Datei erlaubt sein.

Hersteller

mediareif Möstl & Reif Kommunikations- und Informationstechnologien OEG

Breitenseer Straße 110/20,
A - 1140 Wien, AUSTRIA
Tel.: + 43 1 971 08 09; www.comfortsql.com
HG. Wien: FN: 215682f; UID-Nr.: ATU 56100203

mediareif
advanced query languages