Л. НАЙЗАБАЕВА

МЕТОДЫ ШИФРОВАНИЯ ДАННЫХ ДЛЯ ПОВЫШЕНИЯ ФУНКЦИОНАЛЬНОСТИ В ОРГАНИЗАЦИИ СПЕЦИАЛЬНЫХ ВИДОВ ЖЕЛЕЗНОДОРОЖНЫХ ПЕРЕВОЗОК

(Представлена академиком НАН РК Ж. Ж. Байгунчековым)

Настоящая работа посвящена созданию комплекса программ для автоматизации управления и организации защиты информационной системы движения железнодорожного транспорта, в частности, занимающихся транспортированием специального груза.

Специальные перевозки — отдельный вид перевозок, обеспечивающий транспортирование специальных грузов для удовлетворения особо важных государственных и оборонных нужд.

К видам специальных перевозок относятся: перевозка опасных, негабаритных, требующих особого температурного режима, особо ценных и хрупких грузов, конвоированные перевозки. Особое внимание требует доставка к месту назначения генеральных грузов; осуществление перевозки негабаритных, наливных, насыпных, режимных грузов, сборных грузов, хранения груза на консолидированных складах. Для перевозки опасных грузов – взрывчатых веществ, ядов, химикатов необходимо согласовывать маршрут следования, режим перевозки, возможно, назначать сопровождение груза и температурный режим, тщательно контролировать погрузку и разгрузку товара. На все согласования уходит дополнительное время, поэтому все участники перевозки должны строго соблюдать и график и маршрут следования груза.

1. Применение Computer-Aided Software/System engineering (CASE)-технологии в логическом проектировании информационной системы

Создание современных информационных систем представляет собой задачу, решение которой требует применения специальных методик и инструментов. Неудивительно, что в последнее время среди системных аналитиков и разработчиков значительно вырос интерес к CASE (Computer-Aided Software/System Engineering) – технологиям и инструментальным CASE-средствам, позволяющим максимально систематизировать и автоматизировать все этапы разработки программного обеспечения.

В данной работе логическое проектирование базы данных создано с помощью CASE средства AllFusion Erwin Data Modeler (Erwin), построена модель «Entity-Relationship» (рис. 1). Эта схема дает интуитивный обзор проекта и особенно полезна для обмена идеями между пользователями.

AllFusion Erwin Data Modeler (ERwin) является ведущим решением для моделирования баз данных для создания и поддержки баз, витрин (data marts) и хранилищ данных, а также моделей ресурсов данных предприятия.

Модели ER*win* визуализируют структуры данных для облегчения организации и управления данными, упрощения сложных взаимосвязей данных, а также технологий создания баз данных и среды развертывания. При этом упрощается и ускоряется процесс разработки базы данных, а ее качество и надежность существенно улучшаются. ER*win* автоматически генерирует таблицы и тысячи строк кода, хранимых процедур и триггеров для баз данных ведущих вендоров. Технология Complete-Compare, используемая в системе, позволяет проводить итеративную разработку таким образом, что модель всегда синхронизируется с базой данных. ER*win* можно использовать для осуществления и обслуживания и всего жизненного цикла базы данных.

ERwin классифицируется по *функциональной полноте* CASE-системы как система, предназначенная для решения частных задач на одном или нескольких этапах жизненного цикла. Также ERwin (Logic Works) относится к числу *независимых* CASE-систем, т.е. в виде автономных систем, не входящих в состав конкретной СУБД. Обычно независимые системы поддерживают несколько форматов баз данных через интерфейс ODBC[1].

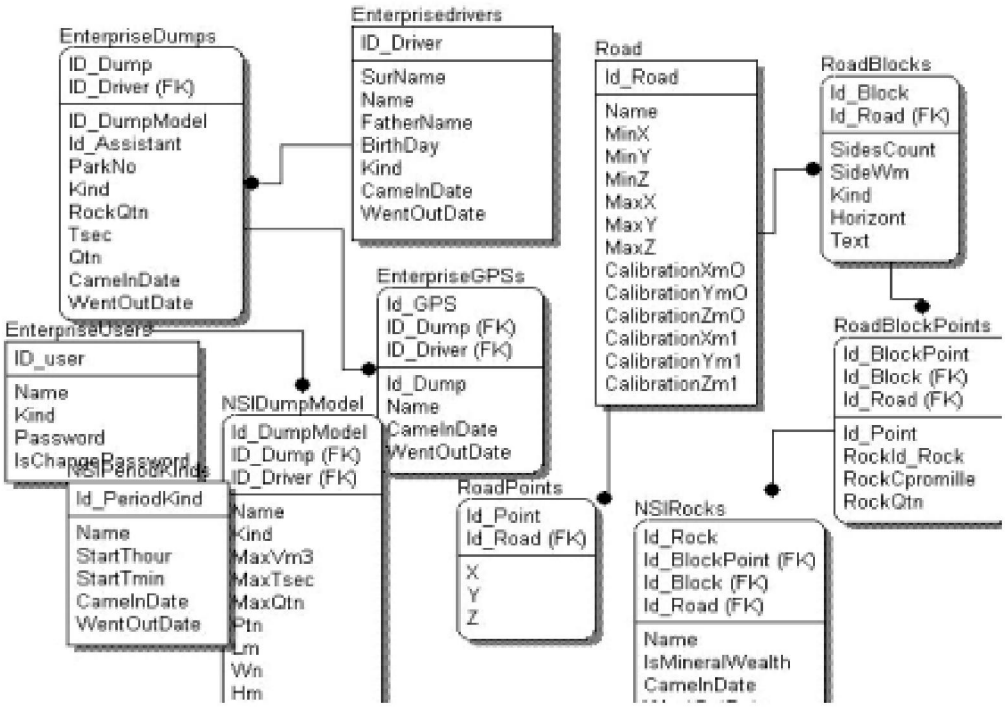

Рис. 1. Фрагмент схемы «Entity-Relationship» для базы данных специализированной транспортной системы в ERwin

2. Создание хранилища данных для управления специальными перевозками в среде MS SQL Server

Следующим шагом явилась проверка всех оперативных применений данных организаций, связанных с их обработкой, и исключение ненужных или повторяющихся данных.

В процессе проектирования БД для решения решать задачи минимизации дублирования данных и упрощения процедур их обработки и обновления данных проведена нормализация отношений. В созданной базе данных таблицы приведены в третью нормальную форму (3NF) по Кодду (Dr.E.F.Codd) [2].

Этап физического проектирования заключается в увязке логической структуры БД и физической среды хранения с целью наиболее эффективного размещения данных, т.е. отображении логической структуры БД в структуру хранения. Решается вопрос размещения хранимых данных в пространстве памяти, выбора эффективных методов доступа к различным компонентам «физической» БД. Результаты этого этапа документируются в форме схемы хранения на языке определения данных (DDL). Принятые на этом этапе решения оказывают определяющее влияние на производительность системы.

В разработанной базе данных использованы гораздо больше возможностей SQL, чем простой инструмент создания запросов. Все ведущие поставщики СУБД используют SQL. Реляционную базу данных и программы, которые с ней работают, можно перенести с одной СУБД на другую с минимальными доработками и переподготовкой персонала. Программные средства, входящие в состав СУБД для персональных компьютеров, такие как, программы для создания запросов, генераторы отчетов и генераторы приложений, работают с реляционными базами данных многих типов. Таким образом, SQL обеспечивает независимость от конкретных СУБД, что является одной из наиболее важных причин его популярности.

3. Применение методов шифрования данных в базе данных средствами СУБД

Известно, что одной из важнейших составляющих проекта базы данных является разработка средств защиты БД. Защита данных имеет два аспекта: защита от сбоев и защита от несанкционированного доступа. Для защиты от сбоев разрабатывается стратегия резервного копирования. Для защиты от несанкционированного доступа каждому пользователю доступ к данным предоставляется только в соответствии с его правами доступа.

В данном проекте применены некоторые способы шифрования встроенными средствами SQL Server при помощи сертификатов [3].

Сертификат – это контейнер для хранения общего ключа, присутствует в виде объекта в базе данных, в SQL Server Management Studio можно просмотреть существующие сертификаты, симметричные и ассиметричные ключи (рис. 2) в контейнере *Databases \имя_базы_данных\ Security\ Certificates:*

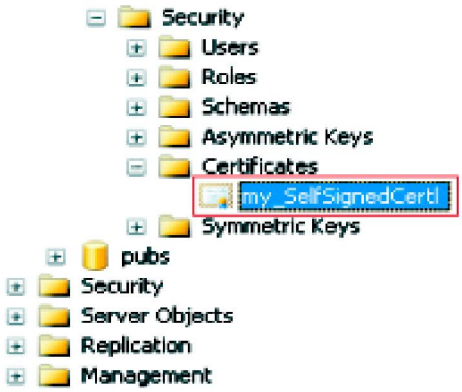

Рис. 2. Просмотр существующих сертификатов

В сертификат помещается информация о версии (в соответствии со стандартом International telecommunication Union X.509), в серийном номере, кому выдан данный сертификат, для каких целей, сколько времени он будет действовать и т.п. Эта информация математически связывается с открытым ключом (при помощи цифрового отпечатка- thumbprint), так что исказить ее будет нельзя.

Обратите внимание, что для создания сертификата вам не требуется никакой центр сертификации – все необходимые средства уже встроены в SQL Server. Однако вы вполне можете загрузить в базу данных сертификат, который был сгенерирован внешним центром сертификации и сохранен в файле (в отдельном файле должен находиться частный ключ), например:

```
USE DBI;
CREATE CERTIFICATE ExternalCertl
FROM FILE - "C:\Certificates\Certl.cer"
WITH PRIVATE KEY
            (FILE – "C:\Certificates\CertlKey.pvk",
DECRYPTION BY PASSWORD = "P@sswOrd); GO
```

Кроме этого, уже готовый сертификат можно также извлечь из подписанной этим сертификатом сборки .NET или из подписанного исполняемого файла. Параметр DECRYPTION BY PASSWORD позволяет указать пароль, который был использован для защиты данного сертификата. Параметр ENCRYPTION BY PASSWORD определяет пароль, который потребуется для расшифровки данных, защищенных сертификатом (для шифрования данных он не нужен). Если этот параметр пропустить, то создаваемый сертификат будет автоматически защищен главным ключом базы данных *(Database Master Key).* Автоматически этот ключ не создается. Чтобы получить возможность работать с ним, нужно предварительно его создать:

```
USE DB1;
CREATE MASTER KEY ENCRYPTION BY
PASSWORD = "P@sswOrd";
```

Кроме пароля, главный ключ базы данных автоматически защищается также главным ключом службы *(Service Master Key).* Этот ключ автоматически генерируется SQL Server в процессе установки. При использовании главного ключа базы данных надо быть очень внимательным: если вы переустановите сервер (а следовательно, изменится главный ключ службы), зашифрованные данные могут быть потеряны. Чтобы этого не случилось, нужно производить регулярное копирование базы данных master или экспортировать главный ключ службы в файл при помощи команды BACKUP SERVICE MASTER KEY.

Обязательный параметр SUBJECT команды CREATE CERTIFICATE определяет цель выдачи сертификата (его значением заполняется соответствующее поле сертификата в соответствии со стандартом X.509vl). После того как сертификат создан, его можно использовать для шифрования данных. Для этой цели применяется специальная функция EncryptByCert:

```
INSERT INTO SecretTable
values(EncryptByCert(Cert_ID("SelfSignedCertl"),N
"Секретные данные"));
```

Если какой-нибудь пользователь после этого произведет запрос к таблице SecretTable, результаты могут его удивить (рис. 3).

Обратите внимание, что функция EncryptByCert принимает в качестве первого параметра не имя сертификата, а его идентификатор. Требуемый идентификатор легко получить при помощи функции Cert_ID.

Рис. 3. Результат запроса к таблице SecretTable

Расшифровка зашифрованных данных производится при помощи функции DecryptByCert. Единственная проблема при работе с этой функцией заключается в том, что она возвращает расшифрованную информацию с использованием типа данных varbinary, поэтому нужно будет произвести преобразование этого типа данных в nvarchar:

SELECT (Convert(Nvarchar(100), DecryptByCert (Cert_ID("SelfSignedCertl"), Secret, N'P@sswOrd'))) FROM SecretTable;

Первый параметр, который принимает функция DecryptByCert, — идентификатор сертификата, возвращаемый той же функцией cert_ID, второй параметр – строковое значение (или переменная, или, как в нашем случае, имя столбца), третий параметр – пароль, которым был защищен сертификат при его создании.

Вывод. В результате разработанной логической информационной модели для оперативного управления и обеспечения безопасности перевозок специального груза решаются следующие задачи: контроль существующей системы управления данных по специальным перевозкам; специальная защита данных; возникают возможности оптимального режима и условий эксплуатации специализированных транспортных средств.

ЛИТЕРАТУРА

1. *Маклаков С.В.* Создание информационных систем с AllFusing Modeling Suite. М.: Диалог-МИФИ, 2003.

2. *Кен Конноли Томас, Бегг Каролин.* Базы данных. Проектирование, реализация и сопровождение. Теория и практика – Database systems. A Practical Approach to Design, Implementation and Management / Пер.с англ. 3-е изд. М.: Вильямс, 2003.

3. *Михеев Р.Н.* MS SQL Server для администраторов. СПб.: БХВ-Петербург, 2006. 544 с.

Резюме

Бағалы жүктерді арнайы тасымалдауын басқарудың логикалық моделі құрастырылған, «Entity-Relationship» сызбасын салуға Computer-Aided Software/System engineering (CASE)- технологиясы қолданылған, MS SQL Server-де физикалық деректер қоры жасалған. Деректер қорын басқару жүйесінің (ДҚБЖ) құралдарының өз мүмкіндіктерімен деректерді SQL-тілінде қорғау ұйымдастырылған.

Summary

Logical model of valuable cargo special transportation management was developed, Computer-Aided Software/System engineering (CASE) – "Entity-Relationship" flowcharting technology – applied, physical database – created in the MS SQL Server, SQL-data security – provided by built-in Database management system (DBMS).

УДК 682.3.07

Казахстанско-Британский технический университет, г. Алматы

Поступила 18.10.10г.