УДК 614:519.6

Лимановская О.В., Смирнов М.Н., Старцев В.С.
suchpublic@gmail.com

КОНВЕРТАЦИЯ БАЗЫ ДАННЫХ BARCLAY В ФОРМАТ, ПРИГОДНЫЙ ДЛЯ ИСПОЛЬЗОВАНИЯ В СОВРЕМЕННЫХ СРЕДСТВАХ КОМПЬЮТЕРНОГО АНАЛИЗА

Введение

В настоящее время, медицинские учреждения повсеместно используют различные системы электронных баз данных пациентов, что в некоторой степени облегчает (или упраздняет) бумажный документооборот и позволяет врачам быстрее получать доступ к данным пациентов, находящихся или ранее находившихся на лечении.

УРНИИФ использует электронную базу данных, созданную в 2004 году и содержащую данные всех когда-либо поступавших на лечение в ПТД пациентов. Хранение, редактирование и доступ к данным пациентов осуществляются при помощи СУБД Barclay 8.4 (другое название – Стационар 3.0). Данная система представляет собой совокупность консольных приложений реализующих основные функции работы с данными и их визуализацию. Формат DBF, используемый данной системой для хранения данных, хоть и признан устаревшим, до сих пор широко используется в финансовых учреждениях. Он подходит для обмена табличными данными, такими, например, как отчеты и реестры платежей. Классический DBF поддерживает основные типы данных – числа, строки, даты и булевы переменные [1].

Тем не менее, в виду некоторых особенностей реализации данной СУБД, помимо прочего, не рассчитанной на дальнейшую автоматизированную обработку данных, и хранящую часть данных в строковой форме, имеющиеся данные в их исходном виде не пригодны для современных средств и методов анализа данных, поэтому необходима конвертация их в новый формат.

Целью данной работы является конвертация обезличенной базы данных пациентов в формат, пригодный для использования современными средствами компьютерного анализа с максимально полным сохранением данных и связей между ними, насколько это предоставляется возможным. В дальнейшем это позволит применение средств компьютерного анализа для возможности поиска необходимых зависимостей и оценки методов лечения.

Образцы и методика эксперимента

Первым делом была проанализирована структура имеющейся базы данных. В таблице 1 изложено описание структурных файлов таблиц, содержащихся в составе предоставленной базы.

Таблица 1 – Описание структурных файлов таблиц СУБД Barclay 8.4

Название таблицы	Описание содержания	Тип таблицы
11NOTE.dbf/12NOTE.dbf	Анкетные данные о пациентах, набранные вручную, связывается через индекс NN (11 – общие вопросы / 12 – хирургическое вмешательство)	dBase 3
STQUES.dbf	Структура для паспортных данных и специальных категорий	dBase 3
11QUES.dbf	Структура для общих вопросов	dBase 3
11SARC.dbf	Данные анамнеза	dBase 3
12QUES.dbf	Структура для вопросов о хирургическом лечении	dBase 3
12SARC.dbf	Данные по операциям	dBase 3
BIGTEXT.dbf	Структура вывода всей информации о пациентах	dBase 3
CONBASE1.dbf	Хранит результаты агрегации для таблиц-отчётов	dBase 3
CONBASE.dbf	Хранит результаты агрегации анкетных данных пациентов	dBase 3
ID.dbf	Данные экрана загрузки программы	dBase 3
INITSTR.dbf	Шаблон конфигурации приложения	dBase 3
INIT.dbf	Данные конфигурации приложения на основе шаблона INITSTR (адреса в системе, отступы, текстовые константы)	dBase 3
NBLOCK.dbf	Данные разделов БЛОК-ФАЙЛА вывода в приложении	dBase 3
SPRAV.dbf	Структура данных с вариантами данных (выпадающие списки) для N-блоков (пунктов анкеты пациента). Используется для выбора вариантов возможных значений в пунктах, не предусматривающих ручной ввод.	dBase 3
STAND.dbf	Записи пользователей (паспортные данные)	dBase 3
TAVIEW.dbf	Параметры визуального вывода таблиц: шрифт, отступы, заголовок	dBase 3
TABL.dbf	Данные запросов (правила выборки) для программы	dBase 3
TXTUSL.dbf	Шаблон таблицы для хранения запросов вида [номер условия, описание, булевый флаг isName, текст условия]	dBase 3
TXTUSL0.dbf	Динамически созданный файл таблицы с составленным запросом	dBase 3
USL0.dbf	Шаблон условия запроса	dBase 3
USLBASE1.dbf	Результат запроса	dBase 3
USLNOW.dbf	Параметры текущей выборки	dBase 3

Приведение графической схемы исходной базы данных не представляется возможным, поскольку связи слишком многочисленны и реализованы среди различных фрагментов программного кода исходной системы. Далее приведено текстовое описание структурной схемы.

Таблицы исходной базы данных можно разделить на четыре вида:

– таблицы-шаблоны для создания новых записей;
– таблицы, хранящие данные о пациентах;
– таблицы, хранящие параметры вариативных полей данных;
– таблицы с внутренней информацией приложения.

Таблицы с внутренней информацией приложения содержат в себе: константы необходимые для работы приложения, элементы графического интерфейса, ссылки на файлы в корневом каталоге СУМБД.

Наличие таблиц для хранения вариативных полей обусловлено отсутствием типа данных для выбора конкретного элемента из списка возможных значений и используемой технологией менеджмента данных в рамках конкретной СУМБД.

Данные разделены на неравные по объёму сегменты, сводящиеся в «блок-файл» для удобства вывода информации об одном пациенте в СУМБД.

Конструкция блок-файла:

– паспортная часть;
– дата выписки, смерти, перевод;
– данные при поступлении;
– данные при выписке;
– хирургическое лечение;
– экспертиза нетрудоспособности;
– оценка приверженности больного к лечению;
– рекомендации по дальнейшему лечению;
– обследование и методики лечения;
– демонстративное наблюдение;
– хирургическое лечение.

Каждый из элементов блок-файла является совокупностью полей соответствующих по тематике названию заголовка.

На основе анализа имеющейся структуры БД, можно выявить следующие трудности:

– поля таблиц базы данных в файлах представлены в виде цифробуквенных сочетаний интерпретируемых внутри СУБД;
– база индексируется при каждом новом включении;
– связи таблиц скрыты в приложении;
– база данных содержит таблицы с параметрами внутреннего функционала приложения и его графического интерфейса.

Результаты анализа

В ходе анализа имеющейся базы данных можно выделить такие сущности как:

– пациент;
– картотека;
– медицинская карта пациента.

Данные из предоставленной базы можно сгруппировать и представить в виде объектно-реляционной системы из трёх сущностей: картотека, медицинская карта пациента, пациент.

Картотека представляет собой хранилище пар целочисленных значений индивидуального номера пациента и номера медицинской карты пациента, связывается по первичному ключу с таблицей «Медицинская карта» и по внешнему ключу с таблицей «Пациент». Данная таблицы должна обеспечивать быстрый доступ к информации о количестве карт заведённых на конкретного пациента.

Таблица «Пациент» хранит данные о пациентах и связывается с таблицей «Медицинская карта» по ключевому полю ID. Для анализа клинических данных пациентов от данной таблицы требуется только уникальный ID, однако в связи с возможным последующим переходом с СУМБД Стационар 3.0 на перспективную разработку, было принято решение о сохранении полей для хранения личных данных пациентов.

Таблица «Медицинская карта» хранит следующие записи: значения номера карты, уникальный номер пациента, а так же тематически разделённые блоки данных с информацией о пациентах, представляющие собой ссылки на таблицы.

Структура базы данных, связи и типы полей представлены на рисунке 1.

Рисунок 1 – Схема сущностей

Алгоритм переноса данных

Перенос данных можно охарактеризовать следующими этапами:

1) Исключение файлов, не содержащих информацию о пациентах

К данной категории можно отнести следующие файлы, содержащие:

– информацию о версии программы;

– переменные: пути к файлам, текстовые и числовые константы;

– параметры окна приложения;

– заголовки графического интерфейса;

– результаты существующих запросов к БД.

В результате данного этапа, на рассмотрении остались файлы, содержащие непосредственную информацию о пациентах и их клинических данных, а именно: 11SARC, 12SARC, 11NOTE, 12NOTE, 11QUES, 12QUES, STAND.

2) Компоновка данных для новых таблиц

Так как в предоставленной базе данных, данные карты пациента были разделены на две категории, это общий анамнез и хирургическое вмешательство, данные в категориях представлены тремя видами таблиц:

– SARC представляет карту пациента заполненную преимущественно числовыми данными типа numeric;

– NOTE хранит набранные в ручную данные соответствующие номерам вопросов в файлах SARC, эти данные хранятся в строковом виде;

– QUES хранит строковые соответствия числовым значениям таблиц SARC и тем самым задаёт максимальное количество возможных значений принимаемых столбцами вопросов.

Надо отметить, что изменение данных в файлах QUES влечёт за собой изменение строковых соответствий к уже существующим данным в картах пациентов, для задания максимального диапазона строковых соответствий требуется изменить структуру таблицы.

Так как все три компоненты в совокупности представляют полные данные карты пациента, а сохранение существующего разделения несёт ряд описанных выше недостатков, для удобства дальнейшей работы с данными их следует объединить в одной таблице.

При объединении следует установить порядок пунктов анкеты и данных ручного ввода в соответствии с их номерами. В таблицах SARC для большинства столбцов в анкете используется тип numeric (число, хранимое в виде строки заданной длины), в новой таблице, их следует заменить строковыми соответствиями из файлов QUES и NOTE. В таблице 2, представлен порядок вопросов «B» и данных ручного ввода «N» с соответствующими типами данных. Это изменение должно сократить количество существующих таблиц, так как данные хранившиеся в таблице NOTE теперь будут использоваться не по ссылке, а напрямую храниться в ячейках с ответами на вопросы анкеты.

Таблица 2 – Описание порядка данных в таблицах базы данных

было				стало			
B1	B2	B3	B4	B1	N1	B2	N2
numeric	numeric	numeric	numeric	text	text	text	text

В таблицах рассматриваемой базы данных, «дата» представлена двумя разными типами: numeric и date. В формате dBase 3 тип данных date хранит строку из 8 чисел формата ГГГГММДД, отличие от типа numeric заключается в длине строки и виде нулевого значения. Для дальнейшей работы мы отдадим предпочтение типу данных numeric, так как числовой формат проще в обработке и более гибок для последующего форматированного вывода или хранения.

На основе данных из таблицы STEND, формируем таблицу пациентов, её схема представлена на рисунке 1, для этого берём только интересующие нас поля: NN, FAMILY, BIRTH, SEXP,

где NN – личный номер пациента,

FAMILY – Фамилия Имя Отчество пациента,

BIRTH – дата рождения,

SEXP – пол пациента.

3) Перенос данных из файлов dbf в таблицы базы данных.

Для переноса данных были использованы скрипты реализованные на языке Python с использованием библиотеки dbf для считывания файлов формата dBase 3.

Данные были выгружены из файлов и перенесены в соответствующие таблицы СУБД SQLite посредством библиотеки sqlite3 и DB-API 2.0 interface for SQLite databases [2].

Заключение

На первом этапе данной работы был выполнен анализ структуры имеющейся в учреждении базы данных, были выделены основные сущности для построения реляционной структуры будущей БД, произведена композиция интересующих нас данных, отделение данные не связанные с пациентами и медицинской информацией в целом, после чего был осуществлён перенос данных пациентов в новую базу данных на базе компактной встраиваемой СУБД SQLite, что позволяет перейти к следующему этапу анализа данных содержащихся в базе данных пациентов, проходивших лечение, с использованием современных компьютерных методов поиска зависимостей и оценки эффективности лечения.

Библиографический список

1. Каррабис Дж.-Д. Программирование в dBASE III PLUS / Дж.-Д. Каррабис. – Москва : Финансы и статистика, 1991. – 240 с.
2. Kreibich J. Using SQLite / J. Kreibich – Sebastopol : O'Reilly Media, 2010. – 530 p.