Для хранения данных разрабатываемой информационной системы предназначена база данных. БД представляет собой хранилище огромных массивов информации в связанном виде для совместного использования. Сама по себе БД не может обслужить запросы пользователей на поиск и обработку информации, так как это просто «склад информации» Чтобы эффективно использовать подобный склад, необходима система управления базой данных.

Лучшим вариантом хранения данных является использование реляционной базы данных. Реляционная база данных – база данных, основывающаяся на реляционной модели. Слово «реляционный» произошло от английского слова «relation» (отношение). Для работы с реляционными БД используют реляционные системы управления базами данных. Использовать реляционные базы данных предложил в 1970 году доктор Кодд, работающий в компании IBM.

Реляционная модель организует данные в виде двумерных таблиц. Каждая реляционная таблица, представляющая собой двумерный массив, должна иметь следующие свойства:

1. каждый табличный элемент является только одним элементом данных;
2. все ячейки в столбце таблицы являются однородными – все элементы столбца должны иметь одинаковый тип (INT, FLOAT, VARCHAR и т. д.);
3. каждый столбец обладает уникальным именем, идентифицирующее данный столбец;
4. в таблице не может быть одинаковых строк;
5. порядок следования строк и столбцов произволен.

Почти каждая система управления реляционными БД обеспечивает работу с языком SQL, с помощью которого возможно определение и модификация структуру сведений, добавление, изменение и удаление данных, а также совершение самых разнообразных выборок данных.

Больше всего распространились серверы реляционных БД, которые базируются на клиент-серверной архитектуре. Данными серверами обеспечивается устойчивая работа с базами данных сразу множества клиентов (ими могут являться десятки, сотни, а также тысячи и млн. клиентов – все зависимо от применяемого оборудования и ПО). Помимо того, реляционная модель данных реализуется так называемыми настольными базами данных, к примеру, dBASE, FoxPro, а также Clarion и Paradox, Access. Почти каждая ведущая настольная БД на данный момент поддерживает возможность работать как клиенты серверов БД с помощью технологий ODBC, а также BDE, ADO и других.

Реляционные БД основаны на строгой теории реляционных БД, основывающейся также на теории множеств, а также теории отношений. Реляционная БД является набором таблиц, между которыми имеются заданные связи. Строки таблиц носят название записей, а элементы, которые включены в запись — полей. В теории реляционных БД таблицы носят название отношений, записи — кортежей, а поля — атрибутов.

В таблице реляционной БД не могут содержаться повторяющиеся записи (строки). Данное требование следует из теории множеств. Наименьший набор полей, дающий возможность отличия записи от любой иной записи — ключ. Все значения ключа в рамках таблицы должны являться уникальными. Каждая таблица должна обладать хотя бы одним ключом, что прямо вытекает из того, что в таблице не могут содержаться повторяющиеся записи. Ключи таблицы могут включать одно поле – эти ключи носят название атомарных или простых ключей. Ключи могут включать несколько полей, то есть составные ключи. Таблица БД может обладать одним ключом или несколькими ключами. Один из ключей назначают как первичный ключ, а остальные — потенциальные (в теории реляционных БД) либо альтернативные (в определенных реализациях отдельных БД).

Нередко применяют суррогатный первичный ключ – ключ, включающий поле (или поля), не несущие сведения из предметной сферы, а выступают как замена (суррогат) для естественных (натуральных) первичных ключей. Как суррогатные ключи зачастую применяются счетчики (генераторы, последовательности) autoincrement либо глобально–уникальные идентификаторы (GUID). В правильности созданная структура БД должна отвечать специальным правилам, основанным на теории отношений и называющимся как нормальные формы.

Понятие нормальной формы было введено Эдгаром Коддом при создании реляционной модели баз данных. Нормализация – это процесс преобразования базы данных, к виду, который отвечает нормальным формам и обеспечивает минимальную избыточность. Цель нормализации – защита базы данных от структурных и логических проблем, являющимися аномалиями данных. К примеру, если есть несколько одинаковых записей в таблице, то в этом случае есть риск нарушения целостности данных при последующем обновлении таблицы. Таблица, прошедшая нормализацию, меньше подвергается этим проблемам, поскольку структура такой таблицы определяет связи между данными, таким образом исключается необходимость существования записей с информацией, которая повторяется. Избыточность можно устранить при помощи разбиения отношений (таблиц) таким образом, чтобы в каждом отношении хранились только первичные факты (факты, которые не выводятся из других хранимых фактов). Таким образом, нормализация не ставит цель уменьшения или увеличения производительности работы или же уменьшения, или увеличения объёма базы данных. Конечная цель нормализации – уменьшение потенциальной противоречивости, хранимой в БД информации. Нормализация применима к таблице, которая представляет собой правильное отношение.

Реляционные СУБД бывают системами, имеющих открытый код (OpenSource) и имеющих закрытый. В соответствии с исследованиями фирмы IDC, предприятия из Европы чаще предпочитают открытое программное обеспечение (ПО). Идут на это они ввиду увеличения качества и гибкости этого ПО.

Система управления БД является программным обеспечением, при помощи которого пользователи имеют возможность создания, определения и поддержания базы данных. Специфика СУБД с открытым кодом состоит в том, что они обладают при себе исходным кодом, который может быть изменен, оптимизирован, ускорен за счет того или иного алгоритма. СУБД представляют собой оболочку, с помощью которой после организации структуры таблиц и заполнения их данными формируется определенная БД. Программные средства включают систему управления вводом-выводом, хранение и обработку информации, создание и тестирование БД, трансляторы [10].

Выбор СУБД для реализации определенной задачи – это довольно сложный процесс, который требует навыков, знаний и внимания профессионала в этой области. Неправильный выбор СУБД может заблокировать процесс реализации данной технологии, а замена СУБД может стоить слишком дорого.

СУБД относится к очень сложной части программного обеспечения, которую трудно понять во всей полноте и является жизненно важной для определения критериев, которые могут быть использованы администратором базы данных в процессе их отбора. Многие практики–разработчики баз данных в своих работах отмечают, что выбор критериев использования СУБД заключается в первичном анализе организационных потребностей и необходимых функций СУБД. Практики–разработчики выделяют ряд факторов выбора СУБД для реализации задачи, в частности:

* требования к применению (это ограничение, которые вводятся в базу данных в приложении);
* функции и инструменты СУБД (встроенный набор инструментов, который помогает облегчить задачу разработки приложений);
* модели СУБД (иерархические, сетевые, реляционные, объектно–ориентированные);
* переносимость СУБД (системы и языки программирования, платформы)
* требования СУБД к оборудованию (минимальная скорость процессора, объем оперативной памяти, дискового пространства и т. д.);
* стоимость СУБД (затраты на приобретение и обслуживание, эксплуатационные расходы, стоимость лицензий, затраты на установку, стоимость обучения и преобразования расходов);
* сохранение целостности данных (защита данных в базе данных от несанкционированного доступа, изменения или уничтожения данных).

В качестве СУБД выбрана реляционная база данных MSSQL. Microsoft SQL Server — система управления реляционными базами данных (СУРБД), разработанная корпорацией Microsoft. Основной используемый язык запросов — Transact-SQL (T-SQL), создан совместно Microsoft и Sybase. Transact-SQL является реализацией стандарта ANSI/ISO по структурированному языку запросов (SQL) с расширениями. Используется для работы с базами данных, размером от персональных до крупных баз данных масштаба предприятия, успешно конкурирует с другими СУБД. Разработчик СУБД - Microsoft - позиционирует этот продукт в качестве «решения для того, чтобы управлять и анализировать корпоративные данные». MS SQL Server уже многие годы выступает как одна из лучших новейших многопользовательских СУБД, с наиболее оптимальным использованием возможностей ОС Windows и обеспечением высокой масштабируемости и надежности.

Языком реляционных баз данных является SQL. SQL – это язык структурированных запросов (Structured Query Language), первый и единственный язык работы с базами данных, который получил широкое распространение и поддерживается всеми производителями коммерческих реляционных СУБД.

SQL является важнейшим компонентом методологии и технологии реляционных баз данных, неотъемлемой частью реляционной СУБД. Без SQL немыслимы разработка и эксплуатация реляционных баз данных.

В составе СУБД SQL выполняет множество функций. Его можно использовать в интерактивном режиме по принципу «запрос-ответ». SQL является подъязыком баз данных в обстановке прикладной программы, составленной на языке программирования. SQL — это язык запросов в приложениях многопользовательской клиент-серверной системы. SQL — это язык администрирования баз данных.

SQL нельзя отнести к конкретному виду языков. Он содержит в себе возможности и языка реляционного исчисления (исчисления кортежей), и алгебраического языка и, несомненно, является реляционно полным.

В силу исторических причин SQL стал стандартным реляционным языком и в настоящее время поддерживается практически всеми системами баз данных. Поэтому каждый специалист по базам данных должен быть знаком с ним.

Язык SQL является языком декларативного типа. В нем отсутствуют какие-либо команды управления ходом вычислительного процесса типа IF-THEN-ELSE, SWITCH, WHILE, DO-WHILE, FOR, GO TO и др. Управление ходом выполнения процесса обработки данных может выполняться интерактивно, в результате действий самого пользователя или при помощи процедурных языков программирования высокого уровня. В связи с этим различают две разновидности SQL — интерактивный и вложенный. По большей части обе формы работают одинаково, но используются различно.

Интерактивный SQL используется для функционирования непосредственно в базе данных и реализуется непосредственно при вводе пользователем отдельных команд. При такой форме SQL при вводе команды она сейчас же выполнится, и пользователь может увидеть результат немедленно.

Вложенный SQL состоит из команд SQL внедренных внутрь программ, написанных на процедурных языках программирования высокого уровня, что делает программы более мощными и эффективными. Однако при таком способе управления процессом обработки данных приходится иметь дело со структурой SQL и стилем управления, который требует некоторых расширений к интерактивному SQL. На практике существуют два основных способа использования SQL в программах:

* + внедрение SQL-операторов в исходный текст программы с последующей ее компиляцией и компоновкой;
  + использование интерфейса прикладного программирования, специализирующегося на работе с базами данных.

База данных в SQL представляет собой набор взаимосвязанных таблиц. При этом различают базовые таблицы и таблицы-представления. Базовая Таблица (TABLE) — это основной структурный элемент базы данных. Ей соответствуют реальные хранимые данные. Концептуальная модель базы данных представляет собой совокупность взаимосвязанных базовых таблиц. Представление (VIEW) является виртуальной таблицей, которая выглядит как реально существующая Таблица. Представление не содержит собственных данных, в нем скомбинированы данные из одной или нескольких связанных базовых таблиц. Средством наполнения таблицы-представления являются запросы, в которых реализуются информационные потребности пользователей базы данных. С помощью представления реализуются локальные представления (внешняя модель данных) базы данных, используемые в прикладных программах или запросах пользователей.

Средством эффективного доступа к хранимым данным являются индексы (INDEX). При помощи индексов осуществляется доступ к данным, упорядоченным по определенным критериям. В качестве критерия упорядочения могут использоваться один или несколько столбцов таблицы. Наглядным образом индекса является Таблица, первый столбец которой содержит значения индексируемых полей, а второй — ссылки на соответствующие строки. При создании базовых таблиц автоматически создаются индексы по первичному ключу, по альтернативным ключам и по внешним ключам.

Как было написано выше, в качестве СУБД для приложения используется Microsoft SQL Server. MS SQL Server выступает как одна из лучших новейших многопользовательских СУБД, с оптимальным использованием возможностей ОС Windows и обеспечением высокой масштабируемости и надежности.