1. Понятия базы данных и системы управления базами данных. Основные свойства баз данных. Трехуровневая архитектура построения баз данных.

Классическое определение БД, данное Дж. Мартином: «БД – это совокупность взаимосвязанных данных при такой их минимальной избыточности, которая допускает их использование оптимальным образом для одного или нескольких приложений в определенной предметной области человеческой деятельности».

Понятие БД

Когда речь идет о БД, то под ней очень часто понимают не сам набор связанных данных, а совокупность, состоящую из набора данных и программы обслуживания (доступа), осуществляющую взаимодействие пользователя с набором данных, а также проведение операций с данными.

СУБД

Полный набор операций с данными предоставляют системы управления базами данных (СУБД). Под СУБД понимают совокупность языковых и программных средств, обеспечивающих создание, поддержание и доступ к данным как со стороны пользователей, так и со стороны приложений.

Кроме средств поддержки структуры данных и операций с данными СУБД также предоставляет развитый пользовательский интерфейс, средства программирования высокого уровня, средства

администрирования, обеспечения секретности и безопасности информации.

Таким образом, под БД будем понимать набор взаимосвязанных данных, хранящихся на физических носителях, а под СУБД – программную среду, позволяющую управлять многими БД и имеющую средства для их разработки.

Основные свойства технологии баз данных

◼ Связность, логическая упорядоченность и структурированность (требует хранения только однородных данных в структурной единице, выделенной для их хранения).

◼ Минимальная избыточность данных (англ. Data Redundancy). Каждый объект должен быть описан только один раз.

◼ Целостность (англ. Data Integrity). Подразумевает правильность данных (англ. Data Validity) в любой момент времени.

◼ Физическая и логическая независимость (англ. Physical and Logical Independence). Физическая независимость данных предполагает независимость данных от их конкретного размещения на физических носителях, их типа, организации и способа доступа. Логическая независимость предполагает независимость приложений (представления, т. е. интерфейса данных) от изменения логической структуры данных.

◼ Безопасность и секретность.

Безопасность и секретность

◼ Под безопасностью данных понимают защиту данных от случайного или преднамеренного доступа к ним лиц, не имеющих на это право, от неавторизированной модификации данных или от их разрушения.

◼ Секретность определяется как право отдельных лиц или организаций решать, как, когда и какое количество соответствующей информации может быть передано другим лицам или организациям.

Так как любая упорядоченность накладывает серьезные ограничения на способ хранения и использования данных, то были предприняты действия, направленные на повышение гибкости доступа к данным. Результатом этих действий стала предложенная в 1978 г. трехуровневая архитектура построения баз данных (рис. 2). Данная схема была разработана как стандарт представления данных (ANSI/SPARC) и в настоящий момент ее поддерживает большинство коммерческих СУБД.

Цель трехуровневой архитектуры заключается в отделении пользовательского представления БД от ее физического представления, т. е. обеспечении независимости от данных.

Первый уровень, внутренний (internal). Определяется физической моделью данных, которая описывает размещение данных на физических носителях и способы доступа к ним, структуру файлов, индексов и отдельных информационных единиц.

Второй уровень, концептуальный (conceptual). Определяется концептуальной моделью данных, которая описывает логическую структуру данных без указания деталей их физического хранения.

Третий уровень, внешний, или уровень представлений (интерфейса). Выводит необходимые данные в требуемом формате, скрывая остальную часть БД.

Внешнее представление – это содержимое БД, каким его видит определенный пользователь. Одному пользователю нужны сведения о товарах и их размещение на складе и он может не иметь понятия, что в БД хранится еще информация о клиентах, поставщиках и т. д. Пользователь может также изменять свое представление, не оказывая влияние на другие представления. Внешний уровень предоставляет также свободу выбора языка общения с БД. Рядовой пользователь может использовать язык интерфейса, т. е. меню и другие запрограммированные действия. Опытный пользователь может воспользоваться языком запросов SQL. Системный программист может использовать расширение SQL и т. д. Все эти уровни связаны между собой программами отображения одного уровня в другой путем трансляции запросов. Запрос от конечного пользователя на требуемые данные должен быть интерпретирован на концептуальном уровне и затем преобразован в конечный запрос на извлечение требуемых данных на физическом уровне. Затем эти данные должны быть преобразованы к виду, запрашиваемому пользователем.