

Тема 7. Программные средства реализации информационных процессов

7.1 Раздел теории

Программное обеспечение (ПО) информационных систем (ИС) – это совокупность программных и документальных средств для создания и эксплуатации систем обработки данных средствами вычислительной техники.

Популярность той или иной ЭВМ, широта и эффективность ее использования зависит не только от надежности и разнообразия ее аппаратного обеспечения (*hardware* компьютера), но и от уровня ее ПО (*software*), который характеризуется разнообразием программ, реализующих алгоритмы решения задач в различных областях науки и техники, производства и управления.

В зависимости от функций, выполняемых программным обеспечением, его можно разделить на 3 уровня:

- базовое программное обеспечение;
- системное программное обеспечение;
- прикладное программное обеспечение.

Каждый следующий уровень опирается на программное обеспечение предшествующих уровней. Каждый вышележащий уровень повышает функциональность всей системы.

Самый низкий (первый) уровень ПО представляет базовое ПО. Оно отвечает за взаимодействие с базовыми аппаратными средствами. Как правило, базовые программные средства непосредственно входят в состав базового оборудования и хранятся в специальных микросхемах – *ПЗВ (ROM – Read Only Memory)*, которые «прошиваются» в процессе производства. Базовое ПО настолько тесно связано с аппаратными средствами, что его иногда считают частью компьютера.

Для ПК совокупность этих программ носит название *BIOS (Base Input Output System – базовая система ввода–вывода)*. В состав *BIOS* входят:

- драйверы стандартных устройств (клавиатура, монитор);
- тестовые программы для контроля работоспособности оборудования;
- программы начальной загрузки;

Эти программы начинают работать при включении ЭВМ. На первом этапе тестируется оперативная память, далее проверяется наличие внешних устройств и только после этого управление ПК передается операционной системе.

Драйвер – это программа, обслуживающая внешнее устройство, которая предоставляет пользователю или программам более высокого уровня набор функций (программные интерфейсы) для управления конкретным внешним устройством, например, драйвер принтера позволяет пользователю задать требуемые параметры печати.

Второй уровень ПО является переходным от первого к третьему. Программы этого уровня обеспечивают взаимодействие других программ компьютера с программами первого уровня и непосредственно с аппаратным обеспечением. Стержнем ПО второго уровня является операционная система (ОС). В состав операционной системы также входят дополнительные драйверы, которые обеспечивают работу с внешними устройствами, для которых не предусмотрены драйверы в базовом ПО. ОС предоставляет программам более высокого уровня набор функций (программный интерфейс), а пользователям – пользовательский интерфейс.

ПО третьего уровня ориентировано на пользователя, предоставляя ему возможность решать конкретные задачи, поэтому ПО этого уровня называется прикладным.

ПО второго и третьего уровней постоянно хранится во внешней памяти и только при необходимости загружается в оперативную память.

Для пользователя наибольший интерес представляет прикладное программное обеспечение, к которому относятся интегрированные пакеты широкого применения, в частности пакет *MSOffice*. В зависимости от версии состав пакета может меняться, но основу его всегда составляют:

- программа *MSWord* (текстовый процессор);
- программа *MSExcels* (табличный процессор);
- программа *MSPowerPoint* (система подготовки презентаций);
- программа *MSAccess* (система управления базами данных).

Программа *MSWord* предназначена для ввода, хранения и форматирования текстовой информации. В *MSWord* автоматизированы функции построения таблиц и формирования изображений.

Программа *MSExcels* предназначена для ввода, хранения и форматирования текстово–символьной (преимущественно символьной) информации. В основе *MSExcels* лежит двумерная электронная таблица. В *MSExcels* автоматизированы функции построения различного рода

графиков, экономических, финансовых, статистических и прочих расчетов.

Программа *MS PowerPoint* предназначена для подготовки презентаций. Основу *MS PowerPoint* составляют шаблоны слайдов и средства мультимедиа (звуковое и видео оформление).

Программа *MS Access* предназначена для разработки, хранения и редактирования баз данных реляционного типа. Основу *MS Access* составляют шаблоны таблиц, форм, запросов и отчетов – режим мастера. Таблицы, формы, запросы и отчеты в *MS Access* можно организовывать и в режиме конструктора – без использования готовых шаблонов.

Основу реляционных таблиц составляют понятия:

Поле – это столбец таблицы.

Запись – это строка таблицы.

Ключевое поле (ключ) – это уникальное поле таблицы, которое не может содержать повторяющихся записей.

Между таблицами в реляционных базах данных можно установить следующие типы связей:

1. Один к одному. Устанавливается между ключевыми полями двух таблиц, в каждой из которых по одному ключевому полю.
2. Один ко многим (многие к одному). Устанавливается двумя способами:
 - а. между ключевым полем одной таблицы и таким же ключевым полем второй таблицы, если в каждой из таблиц по одному ключевому полю
 - б. между ключевым полем одной таблицы и таким же ключевым полем второй таблицы, если в первой таблице одно ключевое поле, а во второй таблице два ключевых поля.
3. Один ко многим. Устанавливается между тремя таблицами. Первая таблицы связывается со второй таблицей через третью промежуточную таблицу. Связь один ко многим можно рассматривать как две связи типа один ко многим, организованным в соответствии со способом б). В этом случае, с одной стороны, ключевое поле первой таблицы связывается с таким же ключевым полем третьей промежуточной таблицы, причем в первой таблице одно ключевое поле, а в третьей таблице – два ключевых поля. С другой стороны ключевое поле второй таблицы связывается с теми же ключевым полем третьей промежуточной таблицы, причем во второй таблице одно ключевое поле, а в третьей промежуточной таблицы (как уже говорилось) два ключевых поля.

При использовании принципов организации указанных типов связей, можно проектировать различные реляционные базы данных различного уровня сложности и из различных предметных областей. Примерами объектов, объединенными различными типами связей могут быть:

1. Один к одному:
 - а. страна – президент (в каждой стране только один действующий президент);
 - б. вуз – ректор;
 - с. кафедра – заведующий кафедрой и т.д.

Связь один к одному самая простая связь. Все поля таблиц, объединенных связью один к одному, могут быть размещены в одной таблице.

2. Один ко многим:
 - а. кафедра – направления (на одной кафедре несколько, т.е. много направлений);
 - б. группа – студенты;
 - с. улица – дома и прочее.

Связь многие к одному это та же связь один ко многим, прочитанная справа налево.

3. Многие ко многим:
 - а. магазины – покупатели (в каждом магазине много покупателей, но и каждый покупатель обслуживается в нескольких магазинах);
 - б. пользователи – сайты;
 - с. преподаватели – студенты и прочее.

Связь многие ко многим читается с двух сторон – слева направо и справа налево. Совокупность таблиц и связей называется схемой данных. Также для базы данных составляется каноническая модель – порядок загрузки таблиц данными.

При построении канонической модели базы данных следует помнить:

1. в первую очередь заполняются таблицы, в которые «не входит» ни одна связь, а выходит связь, начинающаяся с единицы. Если таких таблиц несколько, то безразлично, с какой из таблиц начать загрузку базы данными.
2. во вторую очередь заполняются таблицы, в которые «входит» связь, начинающаяся с бесконечности, а выходит связь, начинающаяся с единицы.

Если такие таблицы связаны с одной таблицей, то их можно заполнять на одном уровне, т.е. безразлично, с какой из них начинать заполнение. Если таких таблиц несколько, и они сгруппированы иерархически, то такие таблицы заполняются в соответствии с уровнем иерархии, т.е. по «старшинству». На этом же уровне заполняются таблицы, в которые «входит» и «выходит» связь, начинающаяся с единицы (с учетом иерархии).

3. в последнюю очередь заполняются таблицы, в которые «входит» и из которых «выходит» связь, начинающаяся с бесконечности (промежуточные таблицы). Если таких таблиц несколько, то они заполняются в соответствии с уровнем иерархии.

