**Содержание:**

1. Состав и способы создания информационного обеспечения

2. Основы организации внутримашинного информационного обеспечения

3. Организация данных во внутримашинной сфере

Список используемой литературы

**1. Состав и способы создания информационного обеспечения**

Информационное обеспечение АСУ - это совокупность единой системы классификации и кодирования технико-экономической информации, унифицированных систем документации и массивов информации, используемых в автоматизированных системах управления. Сущность информационного обеспечения АСУ состоит в информационном отображении условий, состояния и результатов производственного процесса и обмене информацией между органом и объектом управления для регулирования его деятельности.

Основная цель подсистемы информационного обеспечения заключается в том, чтобы представить для решения задач управления необходимые и достоверные сведения в достаточно полном объеме, своевременно и в удобной для использования форме, требующей минимальных затрат машинного времени и труда. Поздно полученная информация часто становится бесполезной, так как решение уже принято.

Информационное обеспечение подразделяют на внемашинное и внутримашинное (рис. 1). К внемашинному информационному обеспечению относят: оперативную документацию, содержащую сведения о состоянии управляемого объекта и среды, нормативно-справочные документы, включающие систематизированную проектно-сметную, техническую, технологическую, организационную и производственную документацию, а также архивную информацию; систему классификации и кодирования информации; инструкции по организации ввода, хранения, внесения изменений в нормативно-справочную документацию, в том числе и в массивы данных о среде.

Внутримашинное информационное обеспечение включает в себя информационную базу на машинных носителях и систему программ ее организации, накопления, ввода и доступа к данным. Источником формирования внутримашинного информационного обеспечения служит внемашинная информационная база.

Основные требования к информационному обеспечению АСУ формулируются на основе данных предпроектного обследования строительной организации. В них обязательно должна быть отмечена необходимость обеспечения адекватности информационной базы предметной области и однозначного трактования модели. Информационная база также должна содержать данные о предметной области, достаточные для программной реализации информационного обеспечения.

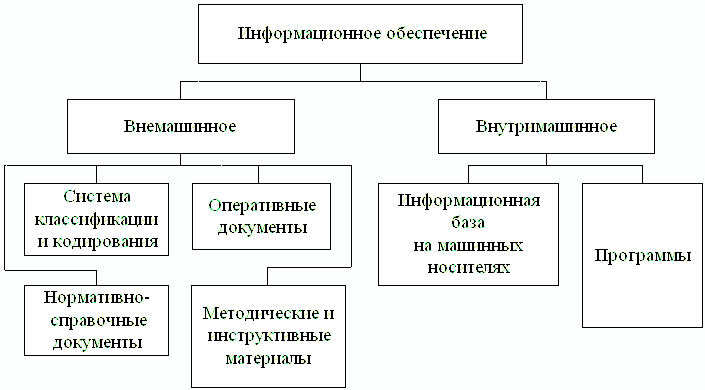


Рис. 1. Структура информационного обеспечения АСУ

В процессе разработки задач АСУ проектирование информационного обеспечения обычно рассматривается как относительно самостоятельная часть общей разработки автоматизированной системы управления. Однако существует и другая методология проектирования с использованием CASE-технологий и CASE-средств, в рамках которой конструирование информационного обеспечения и программных средств решения задач АСУ рассматривается как единый технологический процесс. Ввиду сложности и большой стоимости CASE-технологий и CASE-средств их применяют в настоящее время, как правило только для создания АСУ крупных организаций.

В практике проектирования АСУ чаще всего реализуются следующих два основных подхода к созданию информационного обеспечения: “от предметной области”, “от запроса”. Выбор того или иного способа зависит от содержания исходной информации. Подход “от предметной области”, означает описание объектов управления и связей между ними безотносительно к потребностям пользователей. Иногда этот подход называют объектным или непроцессным. В подходе “от запроса” основным источником информации о предметной области являются запросы пользователей (задачи). Этот подход называется процессным или функциональным.

Преимуществом подхода “от предметной области” является его объективность, системное отображение предметной области и, как следствие, устойчивость информационной модели, возможность реализации большого числа программных приложений по решению задач АСУ, в том числе и заранее незапланированных на созданной информационной базе. Недостатком данного подхода является трудность отбора информации при подготовке информационного обеспечения.

Функциональный подход в большей степени ориентирован на реализацию текущих запросов управленческого персонала строительной организации и не всегда учитывает перспектив развития автоматизи-рованной системы управления. При его использовании могут возникнуть трудности при объединении взглядов различных пользователей. Однако учет запросов руководителей производства позволяет улучшить характеристики функционирования информацион-ного обеспечения. Следует отметить, что подход “от запроса” позволяет руководителям строительных предприятий в наибольшей мере реализовать один из основополагающих принципов создания АСУ - “принцип новых задач”.

Отдельно взятый ни один из указанных подходов не дает достаточной информации для проектирования рационального информационного обеспечения АСУ. Целесообразно совместное применение обоих подходов с ведущим положением объектного подхода. Рекомендуется в качестве первоочередного шага в подготовке информационного обеспечения сделать акцент на подходе “от предметной области”. Однако подготовленную при этом информационную базу следует проанализировать с точки зрения возможности и эффективности выполнения заданных функций.

**2. Основы организации внутримашинного информационного обеспечения**

Внутримашинное информационное обеспечение включает информационную базу на машинных носителях и средства ее ведения. Структура внутримашинной базы определяется моделью логически взаимосвязанных данных конкретной предметной области. В базу данных также входят отдельные невзаимосвязанные массивы входных, выходных и промежуточных данных, хранимых на машинных носителях.

Основными компонентам базы данных являются: нормативно-справочная, плановая, оперативная, учетная информация (рис. 2.). К плановой информации относится та часть нормативных данных, которые непосредственно связаны с организационно-технологическими моделями строительных объектов и плановыми ресурсами по строительным работам. Данные этой группы можно считать условно-постоянными.

Большое внимание в процессе проектирования внутримашинной информационной базы уделяется эффективной организация данных, хранимых на машинных носителях. Выбор тех или иных способов организации данных В ЭВМ во многом определяет в дальнейшем затраты на разработку программных средств обработки информации, на возможность развития базы данных, ее надежность.

Часть сведений в базу данных поступает из внемашинной сферы (документы нормативно-справочной информации, плановые, оперативные документов и др). Некоторые данные информационной базы могут формироваться в процессе решения задач АСУ или поступать по телекоммуникационным каналам связи из других автоматизированных систем управления.

Внутримашинная информационная база характеризуется составом и структурой массивов, способами организации и доступа к данным на машинных носителях. В зависимости от используемых программных средств организация массивов может иметь свои особенности. Существует два основных способа организации информационных массивов: а) в виде отдельных независимых файлов (файловая организация); б) быть в составе базы данных, являющейся интегрированной совокупностью взаимосвязанных массивов.

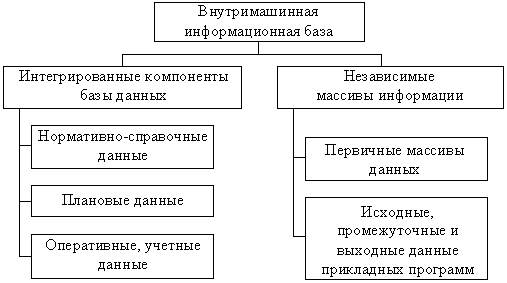


Рис. 2. Состав внутримашинной информационной базы

В качестве независимых массивов с файловой организацией чаще всего выступают первичные массивы, формируемые непосредственно с документов на этапе предбазовой подготовки и файлы, создаваемые в прикладных программах, написанных на алгоритмическом языке (исходные, промежуточные и выходные). Логическая структура файловых массивов и параметры их размещения на машинных носителях содержатся в каждой прикладной программе обработки этих массивов. В этих же программах предусмотрены процедуры их создания и корректировки. Следует отметить, что файловая организация информации не является наглядной и создает предпосылки к дублированию данных. Кроме того, хранение данных в файлах затрудняет актуализацию данных, и не всегда обеспечивает ее достоверность и непротиворечивость.

Более эффективна другая организация технико-экономической информации, заключающаяся в проектировании логически взаимосвязанных массивов в базах данных (рис. 3). Управление такими массивами, включая создание и ведение, выполняется с помощью специализированных программных средств - систем управления базами данных (СУБД).

База данных, по существу, является интегрированной совокупностью недублируемой информации, на основе которой решаются большинство задач АСУ. Логические взаимосвязи в базе данных организуются в соответствии с тем, к какому типу она относится - иерархической, сетевой, реляционной. Существенным преимуществом базы данных является возможность многоаспектного доступа и использования одних и тех же данных различными задачами АСУ.

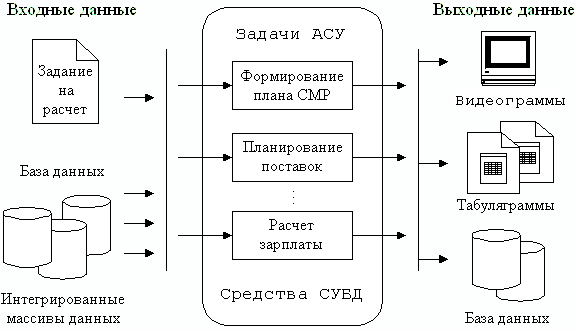


Рис. 3. Схема обработки массивов базы данных в задачах АСУ

Нормативно-справочные данные, как наиболее стабильные, обычно размещают в отдельных массивах базы данных. Технологии формирования и ведения этих массивов имеют свои особенности. Массивы нормативно-справочной информации создаются, как правило, на этапе первоначальной загрузки базы данных. В процессе эксплуатации в эти массивы по мере необходимости дополняются или изменяются, что позволяет поддерживать базу данных в актуальном состоянии.

Плановые данные, характеризующие принятые организационно-технологические решения и включающие сведения о потребности в стоимостных ресурсах, ресурсах типа “мощности”, материалах и изделиях по строительным работам, хранятся в базе данных до окончания строительства соответствующего объекта. Затем они переносятся в архив.

Данные оперативного учета вносятся в базу данных в соответствии с регламентом решения задач, по мере поступления на ввод и обработку документов с оперативной, учетной информацией. Эти данные подлежат накоплению за определенный период (неделя, месяц, квартал), по истечении которого производится их обобщение и обработка. После выполнения очередного расчета (например, формирования календарного графика выполнения строительно-монтажных работ) накопленные данные оперативного учета подлежат удалению или архивированию.

В базе данных также имеются промежуточные (рабочие) и выходные массивы. Их создают (подобно прикладным программам) в процессе решения задач непосредственно сами системы управления базами данных.

Существует два основных режима функционирования базы данных - монопольный и коллективного пользования. В первом случае база данных хранится только на машинных носителях данной ЭВМ и не разрешается одновременный доступ нескольких пользователей.

При наличии средств коммуникации открывается возможность хранить и использовать централизованные базы данных, размещаемые на машине-сервере, в многопользовательском режиме. В этом случае руководители работ со своих рабочих станций, (автоматизированных рабочих мест) получают доступ к общей для всех участников АСУ централизованной информационной базе. Сетевые компьютерные технологии позволяют каждому руководителю работ создавать на своей персональной ЭВМ дополнительную (локальную) базу данных, которая содержит информацию, необходимую только на этом автоматизированном рабочем месте.

В зависимости от конфигурации используемых технических и программных средств при сетевой обработке данных информационной базы может быть осуществлена различная технология работы. Существуют два основных режима сетевой обработки данных - “файл-сервер” и “клиент-сервер”. Технология “файл-сервер” предполагает наличие ЭВМ, выделенной под файловый сервер, на которой находятся ядро сетевой операционной системы и централизованно хранимые файлы. При этой архитектуре обеспечивается коллективный доступ к общей базе данных на файловом сервере. Причем, когда происходит обновление файла одним из пользователей, этот файл блокируется для доступа другим пользователям. Данный режим характерен тем, что запрошенные данные транспортируются с файлового сервера на рабочие станции, где и происходит их обработка средствами системы управления базой данных.

Сетевая компьютерная система “клиент-сервер” предполагает разделение функций обработки данных между клиентом (рабочей станцией) и машиной-сервером баз данных, где обработку осуществляет установленная там система управления базой данных. Запрос на обработку данных выдается клиентом и передается по сети на сервер баз данных, где осуществляется поиск и обработка информации.

Обработанные данные транспортируются по локальной или глобальной сети от сервера к клиенту. Для реализации технологий типа “клиент-сервер” разработан язык структурированных запросов SQL (Structured Queries Language), с помощью которого осуществляются запросы к базе данных АСУ и обеспечивается работа с электронными хранилищами данных других автоматизированных систем управления. Сетевые технологии “файл-сервер”, “клиент-сервер” ориентированны на пользователя-непрограммиста, других диалоговых средств работы пользователей с данными. Следует отметить, что средства управления СУБД позволяют выполнять и ряд других технологий обработки внутримашинной базы данных.

В организации и ведении внутримашинной информационной базы участвуют ряд специальных программных средства ввода и контроля. Эти программы обычно используются для больших информационных баз на этапе предбазовой обработки данных и создания первичных массивов. Средства предбазовой обработки обеспечивают контроль достоверности вводимой в компьютер информации и автоматизацию подготовки больших массивов данных к загрузке и корректировке базы данных.

Эксплуатация информационных баз невозможна без выполнения ряда вспомогательных процедур, например, копирования, архивирования, восстановления, антивирусной защиты и др. Для реализации этих и им подобных задач в составе внутримашинной информационной базы имеются соответствующие базовые программные средства, называемые утилитами.

К средствам ведения внутримашинных баз данных также относятся и прикладные программы, создаваемые на универсальном языке программирования СУБД. Необходимость создания прикладных программ возникает, когда собственные языковые средства системы управления базами данных не позволяют выполнить решение той или иной задачи АСУ.

Прикладные программы АСУ могут разрабатываться и на одном из универсальных алгоритмических языков (Basic, Visual C++, Fortran, Modula и др.). Однако в таких программах не всегда достигнуть независимости программ обработки и самих данных, избежать дублирования данных в информационных массивах разных задач АСУ. Это приводит к многократному вводу одних и тех же данных для разных задач автоматизированной системы управления и вызывает существен-ные проблемы при внесении изменений в исходные данные.

Для обеспечения процессов создания и эксплуатации внутримашин-ной информационной базы необходимы соответствующие технологичес-кие инструкции, в которых должны быть регламентированы процессы ввода, контроля данных и корректировки данных, получения копий файлов, архивирования и восстановления базы данных и др. В инструкциях по вводу и контролю должны быть описаны технологии ввода информации, определена последовательность создания массивов. Здесь же должны быть описаны необходимые проверки достоверности вводимых сведений, используемые методы контроля (на диапазон значений, с помощью контрольных сумм и др.).

Инструкции по загрузке и корректировке базы данных должны определять входные документы (массивы), с которых осуществляется загрузка. В этих инструкциях должны быть описаны экранные формы, которые соответствуют формам входных документов и позволяют одновременно вводить данные в несколько логически взаимосвязанных массивов. При этом должны быть обеспечены требования однократного ввода одной и той же информации в базу данных.

Создание базы данных коллективного пользования, в том числе для работы в компьютерных сетях, также должно сопровождаться необходимыми инструкциями для администрации баз данных. В них определяются функции персонала, по обеспечению доступа пользователей АСУ к общей базе данных с соблюдением требований по защите информации от несанкционированного доступа, определению сферы ответственности за сохранность данных централизованной информационной базы АСУ.

**3. Организация данных во внутримашинной сфере**

Существует два уровня организации данных во внутримашинной сфере - логический и физический. Физическая организация данных определяет способ размещения информации непосредственно на машинных носителях и выполняется программными инструментариями автоматически (без участия человека). Разработчики внутримашинной информационной базы АСУ оперируют в программах только представ-лениями о логической организации данных, которая определяется видом модели данных. Под моделью данных понимается совокупность взаимосвязанных структур данных и операций над этими структурами.

Вид модели и используемые в ней типы структур данных во многом предопределяют выбор системы управления базами данных или языка программирования, на котором создается прикладная программа обработки данных. Следует отметить, что для размещения одной и той же информации во внутримашинной сфере могут быть использованы различные структуры и модели данных. Их выбор возлагается на разработчиков информационной базы АСУ и зависит от многих факторов, в том числе от имеющегося технического и программного обеспечения, объемов информации, сложности задач АСУ.

В ряде случаев при организации данных во внутримашинной сфере применяют файловую модель. При такой модели внутримашинная информационная база представляет собой множество не связанных между собой файлов из однотипных записей с одноуровневой структурой (рис. 4). Запись является основной структурной единицей обработки данных и состоит из фиксированного набора (кортежа) полей, каждое из которых представляет собой элементарную единицу логической организации данных. Структура записи определяется составом и последовательностью входящих в нее полей.

Каждому экземпляру записи, как правило, в соответствие, ставятся один или два ключа записи: первичный (уникальный) и вторичный ключ. Первичный ключ - это одно или несколько полей, однозначно идентифицирующих запись. В случае, если первичный ключ состоит из одного поля, он называется простым, если из нескольких полей - составным ключом. Вторичный ключ, в отличие от первичного, - это такое поле, значение которого может повторяться в нескольких записях файла, то есть он не является уникальным. Если по значению первичного ключа может быть найден один единственный экземпляр записи, то по вторичному - несколько.

Для ускорения доступа к записям файла выполняется процедура индексирования, результатом которой является создание дополнитель-ного индексного файла, содержащего в упорядоченном виде все значения ключей файла данных. Для каждого значения ключа в индексном файле содержится указатель на соответствующую запись файла данных. Наличие индексного файла позволяет по заданному ключу быстро находить запись. Индексирование может производиться не только по первичному, но и по вторичному ключу.

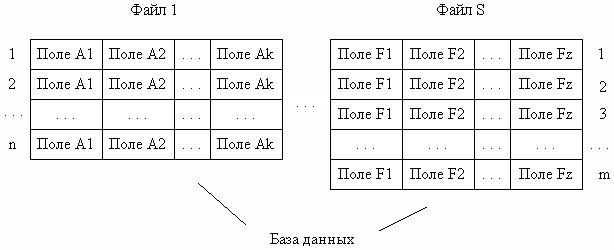


Рис. 4. Файловая организация баз данных (файлы, записи, поля)

Описание логической организации данных файловой модели заключается в присваивании каждому файлу уникального имени, а также в описании структуры его записей. При этом каждому полю задается сокращенное обозначение (имя поля) и указывается формат поля (тип хранимого данного, длина поля и точность числовых данных). Для полей, выполняющих роль уникального (первого) ключа записи, указывается признак ключа. Структура файла обычно описывается таблицей, в которой отмечаются первичные и вторичные ключи.

Файловые информационные базы обрабатываются системами управления файлами (Q&A, Reflex, FFS File и др.), которые не считаются системами управления базами данных. Файловые системы легко осваиваются, достаточно просты и эффективны в использовании. Для работы с ними используются простые языки запросов, либо и вовсе ограничиваются набором программ-утилит. Такие системы обычно поддерживают работу с небольшим числом файлов, содержащих ограниченное число записей с небольшим количеством полей.

Кроме файловых моделей организации данных внутримашинной сферы существуют иерархические, сетевые и реляционные модели. Эти типы моделей являются более сложными и, в отличие от файловой организации данных, поддерживаются СУБД соответствующего типа. Различия между этими классами моделей постепенно стираются. Однако некоторые особенности перечисленных типов моделей следует отметить. Для иерархических и сетевых моделей их структура не может быть изменена после ввода данных, тогда как структура реляционных моделей может изменяться в любое время. С другой стороны, иерархические и сетевые модели обеспечивают более быстрый доступ к информации, чем реляционные модели.

Иерархические модели имеют древовидную структуру, когда каждому узлу структуры соответствует один сегмент, представляющий собой поименованный линейный кортеж данных. Каждому сегменту, кроме корневого, соответствует один входной и несколько выходных сегментов (рис. 5).

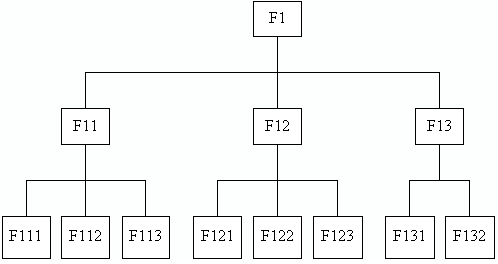


Рис. 5. Структура иерархической базы данных

Каждый сегмент лежит на единственном иерархическом пути, начинающемся с корневого сегмента. При описании такой логической организации данных достаточно для каждого сегмента указать его входной сегмент. Так как в иерархической модели каждому входному сегменту данных соответствует N выходных, то такие модели весьма удобны для представления отношений типа 1:L в предметной области.

Некоторым недостатком иерархических моделей является их неэффективность при реализации отношений типа L:L, медленный доступ к сегментам данных нижних уровней иерархии и четкая ориентация только на определенные типы запросов и др. В связи с этим в настоящее время СУБД, базирующиеся на иерархических моделях, подвергаются существенным модификациям, позволяющим поддерживать более сложные типы структур и, в первую очередь, сетевые их модификации.

Сетевая модель во многом подобна иерархической и отличается от нее только тем, что допускает несколько входных сегментов наряду с возможностью наличия сегментов без входов с точки зрения иерархической структуры. На рис. 6 представлен простой пример сетевой структуры, полученной на основе модификации иерархической топологии (рис. 5).

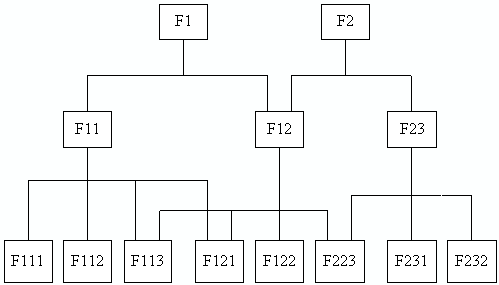


Рис. 6. Структура сетевой базы данных

Графическое отображение структуры связей сегментов в такого типа моделей представляет собой сеть. Сегменты данных в сетевых базах данных могут иметь множественные связи с сегментами старшего уровня. В связи с тем, что в сетевых моделях имена и направление связей не так очевидны, как в иерархических моделях данных, они должны указываться при описания базы данных. В сетевых моделях данных любая запись старшего уровня может содержать данные, относящиеся к набору записей подчиненного уровня. Обращение к набору всех записей реализуется, начиная с записи старшего уровня. При этом нет необходимости, как это выполняется в иерархических моделях, осуществлять доступ к искомому набору записей через корневой сегмент. Обращение к данным возможно с любой точки доступа по связям.

Сетевые модели данных по сравнению с иерархическими являются более универсальным средством отображения во внутримашинной сфере структуры информации для разных предметных областей и это существенно расширяет сферу их применения. Достоинством сетевых моделей является отсутствие дублирования данных в различных элементах модели. Кроме того, технология работы с сетевыми моделями является более удобной, так как доступ к данным практически не имеет ограничений и возможен непосредственно к объекту любого уровня. Допустимы всевозможные запросы. Однако следует отметить, что ввиду сложности сетевых моделей, разработка СУБД на их основе предполагает использование опытных системных аналитиков и программистов. Кроме того, при использовании сетевых моделей более остро стоит проблема обеспечения сохранности информации в базе данных.

Реляционные модели данных отличаются от сетевых и иерархических простотой структур данных, удобным для пользователя табличным представлением и доступом к данным. Большинство современных баз данных в настоящее время разрабатываются на основе моделей подобного типа. Реляционную модель представления информации предложил в 1970 г. сотрудник фирмы IBM Эдгар Кодд. Данная модель позволяет выполнять все необходимые операции по запоминанию и поиску данных и обеспечивает целостность данных.

Модель основана на математическом понятии отношения, расширенном за счет значительного добавления специальной терминологии и развития соответствующей теории. В такой модели общая структура данных (отношений) может быть представлена в виде таблицы, в которой каждая строка значений (кортеж) соответствуют логической записи, а заголовки столбцов являются названиями полей (элементов) записи. Процедуры запоминания и поиска осуществляются с применением операций на множествах (объединение, пересечение, разность, произведение) и реляционных операций (выбрать, спроецировать, соединить, разделить). Отметим, что хотя реляционная модель и выглядит как совокупность связанных таблиц, но на физическом уровне данные хранятся в файлах, содержащих последовательности записей.

В реляционной модели каждому объекту предметной области соответствует одно или более отношений. При необходимости связь между объектами можно указать в явном виде. В такой связи (отношении) в качестве атрибутов указываются идентификаторы взаимосвязанных объектов. В реляционной модели объекты предметной области и связи между ними представляются одинаковыми конструкциями, что существенно упрощает модель.

Суть реляционной модели можно пояснить на следующем примере. Пусть в базе данных строительного предприятия имеются два файла: а) справочник железобетонных изделий; б) отчет о поставках изделий (рис. 7). Каждый из этих файлов содержит определенное число записей, состоящих из фиксированного числа полей (соответственно 4 и 4).

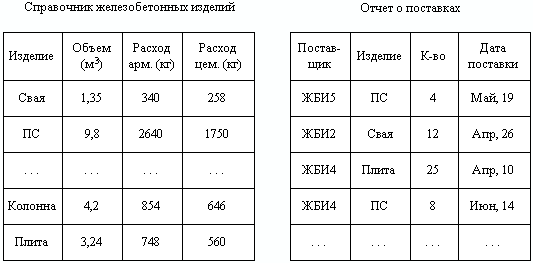


Рис. 7. Фрагмент реляционной модели базы данных

В данном фрагменте базы данных определены два отношения (файла), имеющие общий элемент значения поля Изделие. Операции реляционной алгебры могут объединить два типа записей по этому общему элементу. Например, в результате соединения запись ПС может представиться в следующем виде:

ПС <Объем (м3)><Расход арм. (кг)><Расход цем. (кг)>

<ЖБИ5><К-во><Дата поставки>.....

Иначе говоря, к сведениям о изделии добавляются сведения о всех его поставках, имеющиеся в реляционной базе данных. Связь между записями допускается по нескольким полям, позволяя выполнять достаточно сложные манипуляции с данными. Поля данных, связывающих вместе две записи, могут быть уникальными для данной пары, но могут дублироваться и во многих других записях. Они могут повторяться неоднократно, связывая между собой записи. Аналогичным образом можно проиллюстрировать выполнение в реалиционной модели операций проекции и селекции.

Чтобы не допустить потерь или искажения информации в реляционной базе данных необходим соответствующий контроль всех взаимосвязей записей. Этот контроль выполняется СУБД, которые в процессе работы постоянно пересчитывают число связей для каждой записи базы данных в прямом и обратном направлениях. При больших объемах баз данных осуществление такого контроля может потребовать существенных затрат машинного времени.

**Список используемой литературы:**

1. Автоматизированные информационные технологии в экономике./ Под ред. проф. Г.А.Титоренко. –М.: Компьютер, ЮНИТИ, 2006.-205 с.

2. Компьютерные технологии обработки информации./ Под ред. С.В.Назарова. –М.: Финансы и статистика, 2007. – 487 с.

3. Каpатыгин С. Компьютеp для носоpога. // Кн.З.: Носоpог в моpе данных. // Базы данных: пpостейшие сpедства обpаботки инфоpмации; электpонные таблицы; системы упpавления базами данных. В 2-х томах. - М.: ABF, 20055.

4. Хаселиp Р. Опеpационная система Windows 3.1. - М.: ЭКОМ, 2003. – 156 с.

5. Хаpвей Г. Excel 5.0 для "чайников". - К.: Диалектика, 2001. – 234 с.