Министерство образования Украины

Донбасская Государственная Машиностроительная Академия

Контрольная работа

по дисциплине «Информационные системы»

2008 г.

1. **Стратегии разработки административных информационных систем**

Административные информационные системы (АИС) создаются для совершения управления и обеспечивают неразрывную связь между информацией и управлением. Создание АИС – сложная проблема. Даже для мелких организаций оно предполагает разработку ряда подсистем, которые должны соответствовать принципам интеграции и управляемости.

Существенное влияние на разрабатываемую информационную модель оказывает стратегия (или система взглядов) относительно организации системы. На практике применяют различные сочетания типовых стратегий или подходов

Общесистемный подход.

Общесистемный подход основывается на предположении, что еще до реализации системы мы некоторым обоснованным способом можем распознать взаимосвязи между частями ее базовой информации. Процессы сбора, хранения и обработки данных проектируются и реализуются в рамках всей системы в целом. Хотя этот подход является идеальным, его применение в полном объеме может оказаться весьма трудным из-за практических, политических, социальных проблем. При уже существующей системе проектирование идеальной системы может стать просто академическим упражнением, так как ее реализация вызовет радикальные преобразования. В самом деле, слепое, без определенной технологии, использование проектировщиками этого подхода может привести к крушению иллюзий, что и случается во многих сегодняшних вычислительных системах. Однако в организациях, которые еще не имеют разработанных систем, действующих и считающихся удовлетворительными, рассматриваемый подход может быть успешно применен. Он является идеализированным и не может в полном объеме применяться в реальной организации.

Как можно было увидеть при рассмотрении шести подходов, стратегия выбора подхода должна формироваться с учетом особенностей конкретной системы. Следует принимать в расчет такие факторы, как размер организации, природа ее деловых операций и опыт. Существенно, что выбор стратегии должен быть произведен после тщательной оценки степени риска и преимуществ возможных подходов.

**2. Нормализация и нормальные формы при разработке баз данных**

Центральная задача проектирования базы данных ЭИС -определение количества отношений (или иных составных единиц информации) и их атрибутного состава.

Задача группировки атрибутов в отношения, набор которых заранее не фиксирован, допускает множество различных вариантов решений. Рациональные варианты группировки должны учитывать следующие требования:

1. множество отношений должно обеспечивать минимальную избыточность представления информации,
2. корректировка отношений не должна приводить к двусмысленности или потере информации,
3. перестройка набора отношений при добавлении в базуданных новых атрибутов должна быть минимальной.

Нормализация представляет собой один из наиболее изученных способов преобразования отношений, позволяющих улучшить характеристики БД по перечисленным критериям.

Ограничения на значения, хранимые в реляционной базе данных, достаточно многочисленны. Соблюдение этих ограничений в конкретных отношениях связано с наличием так называемых нормальных форм. Процесс преобразования отношений базы данных к той или иной нормальной форме называется нормализацией отношений. Нормальные формы нумеруются последовательно от 1 по возрастанию, и чем больше номер нормальной формы, тем больше ограничений на хранимые значения должно соблюдаться в соответствующем отношении.

Ограничения, типичные для реляционной модели данных, -это функциональные и многозначные зависимости, а также их обобщения. В принципе, множество дополнительных ограничений может расти и соответственно будет увеличиваться число нормальных форм. Применяемые ограничения ориентированы на сокращение избыточной информации в реляционной базе данных.

Отношение в первой нормальной форме (сокращенно 1НФ) - это обычное отношение с двухуровневой структурой. Недопустимость в структуре отношения третьего и последующих уровней является ограничением, определяющим 1НФ отношения.

Преобразование ненормализованного отношения в представление, соответствующее 1 НФ, - это операция нормализации, рассмотренная выше. Следует отметить определенное терминологическое несоответствие - нормализация СЕЙ при-"водит к 1НФ, а нормализация отношений реляционной БД обычно производится до ЗНФ или 4НФ.

Реляционная база данных в целом характеризуется 1НФ, если все ее отношения соответствуют 1НФ.

Следующие нормальные формы" (вторая и третья) используют ограничения, связанные с понятием функциональной зависимости, которое необходимо предварительно рассмотреть.

Функциональные зависимости и ключи

Функциональные зависимости определяются для атрибутов, находящихся в одном и том же отношении, удовлетворяющем 1НФ.

Простейший случай функциональной зависимости охватывает 2 атрибута. В отношении R(A,B,...) атрибут А функционально определяет атрибут В, если в любой момент времени каждому значению А соответствует единственное значение В (обозначается А -> В).

Иначе говорят, что В функционально зависит от А (обозначается В = f(A)). Первое обозначение оказывается более удобным, когда число функциональных зависимостей растет и их взаимосвязи становятся труднообозримыми; оно и будет использоваться в дальнейшем. Отсутствие функциональной зависимости обозначается А —/-> В.

Можно определить ситуацию А —> В с помощью операции "образ", сказав что множество in B(а) должно содержать один элемент для любого значения а атрибута А.

Рассмотрим несколько примеров:

Пример №1.Фамилия студента, группа. Показывает зависимость А -> В (или В -> А), но не обе зависимости вместе

|  |  |
| --- | --- |
| R1 | |
| ФИО | ГР |
| Иванов Зуев Смирнов Яшина | 1960  1963 1960  1961 |

Пример №2 Магазин и расчетный счет. Наличие взаимно- однозначного соответствия А <-> В.

|  |  |
| --- | --- |
| R2 | |
| Магазин | Расч |
| ММЗ | 704098 |
| Динамо АТЭ | 122095 440162 |

Пример №3 Студент, Дисциплина. Отсутствие функциональной зависимости.

|  |  |
| --- | --- |
| R3 | |
| ФИО | Дисциплина |
| Петров Федин Алешин Петров | Физика Химия Физика Химия |

Понятие функциональной зависимости распространяется на ситуацию с тремя и более атрибутами в следующей форме. Группа атрибутов (для определенности А,В,С) функционально определяет атрибут D в отношении T(A,B,C,D,....), если каждому сочетанию значений <a,b,c> соответствует единственное значение d (а - значение A, b - значение В, с - значение С, d - значение D). Наличие такой функциональной зависимости будем обозначать А,В,С —> D. Случай, когда в правой части функциональной зависимости присутствует несколько атрибутов, не нуждается в специальном рассмотрении. Взаимно-однозначные соответствия для трех и более атрибутов также не имеют самостоятельного значения.

Невнимание к способам кодирования атрибутов может привести к несоответствию функциональных зависимостей и хранящихся данных, что является серьезной проектной ошибкой.

С помощью функциональных зависимостей определяется понятие ключа отношения, точнее ряд разновидностей ключей - вероятные, первичные и вторичные.

Вероятным ключом отношения называется такое множество атрибутов, что каждое сочетание их значений встречается только в одной строке отношения, и никакое подмножество атрибутов этим свойством не обладает. Вероятных ключей в отношении может быть несколько.

Важность вероятных ключей при обработке данных определяется тем, что выборка по известному значению вероятного ключа дает в результате одну строку отношения либо ни одной.

На практике атрибуты вероятного ключа отношения связываются со свойствами тех объектов и событий, информация о которых хранится в отношении. Если в результате корректировки отношения изменились имена атрибутов, образующих ключ, то это свидетельствует о серьезном искажении информации. Следовательно, систематическая проверка свойств вероятного ключа позволяет следить за достоверностью информации в отношении.

Когда в отношении присутствует несколько вероятных ключей, одновременное слежение за ними очень затруднено и целесообразно выбрать один из них в качестве основного (первичного).

Первичным ключом отношения называется такой вероятный ключ, по значениям которого производится контроль достоверности информации в отношении.

Применительно к экономической информации в подавляющем большинстве случаев отношения, полученные из существующих экономических документов, содержат единственный вероятный ключ, который является и первичным ключом. Это объясняется тем, что содержимое экономических документов понимается всеми пользователями одинаково. Далее будем иметь в виду только такие отношения. Наличие двух и более вероятных ключей в отношениях с осмысленной информацией можно объяснить наличием нескольких возможных способов интерпретации одних и те х же данных. Первичный ключ часто называется просто ключ.

Каждое значение первичного ключа встречается только в одной строке отношения. Значение любого атрибута в этой строке также единственное. Если через К обозначить атрибуты первичного ключа в отношении (А,В,С,...,J), то справедливы следующие функциональные зависимости К -> А, К -> В, К -> С,..., К -> J. Набор атрибутов первичного ключа функционально определяет любой атрибут о> ношения. Обратное также верно: если найдена группа атрибутов, которая функционально определяет все атрибуты отношение по отдельности, и эту группу нельзя сократить, то найден первичный ключ отношения.

Для множества функциональных зависимостей существует ряд закономерностей, которые выражаются теоремами. Знание теорем позволяет из исходного множества функциональных зависимостей получать производные зависимости.

Отметим ряд известных теорем о функциональных зависимостях. Атрибуты, фигурирующие в каждой теореме, должны находиться в одном и том же отношении.

Теорема 1

А,В -\* А и А,В -> В.

Доказательство основано на том, что в строке <а,Ь> для атрибутов А и В значение а (как и значение Ь) присутствует один раз.

Теорема 2

А -> В и А ->С тогда и только тогда, когда А -> ВС.

Рассмотрим произвольное значение а атрибута А. Если А ->В и А -\* С, то im В(а) и im C(a) содержат по одному элементу. Предположим, что зависимость А -> ВС неверна и ini ВС(а) состоит из 2 иди более элементов. Тогда либо im B(a), либо im C(a) должны содержать более одного элемента. Полученное противоречие доказывает зависимость А -> ВС.

Обратно, если А -> ВС, то im BC(a) содержит один элемент вида <Ь,с> для любого а. Зафиксируем некоторое значение al. Значение b (как и значение с) встречается в сочетании с д! только один раз, следовательно, справедливо А -> В и А ->С.

Теорема 3

Если А ->" В и В ->С, то А -> С.

Предположим, что зависимость А -> С неверна и множество im С(а) содержит более одного элемента. Каждому значению а соответствует единственное значение b (в силу А -> В), поэтому im C(b) содержит более одного элемента. Получилось противоречие с условием В -> С, что и доказывает теорему.

Примечательно, что доказательства остальных теорем опираются на первые 3 теоремы, а не доказываются от противного.

Теорема 4

Если А—» В, то АС -> В (С произвольно).

Доказательство

АС —> А (теорема 1), А -> В (условие), следовательно, АС -> В по теореме 3.

Теорема 5

Если А -> В, то АС -> ВС (С произвольно).

Доказательство

АС —» В (теорема 4), АС —» С (теорема 1), следовательно, АС->ВС по теореме 2.

Теорема 6

Если А ->В и ВС ->D, то АС -\* D.

Доказательство

Из А -> В следует АС -> ВС (теорема 5). ВС -> D (условие), поэтому АС—> D по теореме 3.

Вторая и третья нормальные формы отношений

Отношение имеет вторую нормальную форму (2НФ), если оно соответствует 1НФ и не содержит неполных функциональных зависимостей.

Неполная функциональная зависимость - это две зависимости:

* вероятный ключ отношения функционально определяет некоторый неключевой атрибут,
* часть вероятного ключа функционально определяет этот же неключевой атрибут.

Отношение, не соответствующее 2НФ, характеризуется избыточностью хранимых данных.

Например:

|  |  |  |  |
| --- | --- | --- | --- |
| Т4 | | | |
| Магазин | Изделие | Цена | План\_1999\_г. |
| Салют | М22 | 50 | 200 |
| Салют | К14 | 40 | 100 |
| АТЭ | М22 | 50 | 300 |
| АТЭ | Т62 | 60 | 100 |

Функциональные зависимости отношения Т4:

Избыточность иллюстрируется тем фактом, что цена изделия указывается столько раз, сколько магазинов продают это изделие (изделие М22 в Т4). Переход к 2НФ и соответственно устранение отмеченной избыточности данных связано с созданием двух отношений вместо исходного отношения Т4.

|  |  |  |
| --- | --- | --- |
| Т41 | | |
| Магазин | Изделие | План\_1999\_г. |
| Салют  Салют  АТЭ  АТЭ | М22  К14  М22  Т62 | 200  100  300  100 |

|  |  |
| --- | --- |
| Т42 | |
| Изделие | Цена |
| М22  К14  Т62 | 50  40  60 |

Отношение соответствует ЗНФ, если оно соответствует 2НФ и среди его атрибутов отсутствуют транзитивные функциональные зависимости (ФЗ).

Алгоритм получения отношений в ЗНФ обладает следующими свойствами:

* сохраняет все первоначальные функциональные зависимости, т.е. зависимость, справедливая в R, справедлива и в одном из производных отношений. Это гарантирует получение осмысленных отношений с легко интерпретируемой структурой;
* обеспечивает соединение без потерь, т.е. значения исходного отношения R могут быть восстановлены из проекций отношения R с помощью операции соединения;
* результат декомпозиции в ЗНФ обычно содержит меньше значений атрибутов, чем исходное отношение R (происходит уменьшение избыточности).

1. **Разработать информационную систему для учета затрат на производство продукции**

Данная задача необходима для планово-экономического отдела предприятия. Ее цель показать затраты на производство с учетом материальных и трудовых ресурсов. Результаты задачи показывают насколько выгодно для предприятия выполнение данного заказа изготовления продукции. Данные о необходимой информации поступают из следующих отделов участвующих в процессе обработки информации:

* + финансовый;
  + отдела материалов;
  + комплектующий отдел;
  + расчетный отдел;
  + др.

Необходимость или периодичность выполнения данной задачи зависит от количества выполняемых заказов. Другими словами, она необходима при выполнении любого заказа, выполняемого на предприятии.

Приведенные входные данные имеют укрупненный вид.

Входные таблицы:

1. Сводная таблица учетов заказов.(Costs)

|  |  |  |
| --- | --- | --- |
| Наименование поля | Идентификация | Тип данных |
| Код записи (ключевое) | ID | Счетчик |
| Код заказа | KodZak | Текстовое |
| Дата | Data | Дата/Время |
| Затраты на материалы | Materials | Денежный |
| Затраты на всп. материалы | SumMaterials | Денежный |
| Накладные расходы | Overheads | Денежный |
| Полуфабрикаты/комплектующие | SemiProducts | Денежный |

1. Таблица учета работы над заказом специалистов (MainManufacture)

|  |  |  |
| --- | --- | --- |
| Наименование поля | Идентификация | Тип данных |
| Код записи (ключевое) | ID | Счетчик |
| Код заказа | ID\_costs | Числовое-целое |
| Код рабочего | WorkerID | Числовое-целое |
| Заработная плата | Zarpl | Денежный |

1. Таблица учета работы над заказом групп работников (SubManufacture)

|  |  |  |
| --- | --- | --- |
| Наименование поля | Идентификация | Тип данных |
| Код записи (ключевое) | ID | Счетчик |
| Код заказа | ID\_Costs | Числовое-целое |
| Количество работников | Kolvo | Числовое-целое |
| Средняя заработная плата | AvrZarpl | Денежный |

1. Таблица работников-специалистов (WorkersList)

|  |  |  |
| --- | --- | --- |
| Наименование поля | Идентификация | Тип данных |
| Код записи (ключевое) | ID | Счетчик |
| Табельный номер | TabNo | Числовое-целое |
| Фамилия, И.О. | FIO | Текстовое |

Построим схему данных.

|  |
| --- |
| **MainManufacture** |
| ID |
| ID\_costs |
| WorkerID |
| Zarpl |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Costs   |  | | --- | | **WorkersList** | | ID | | TabNo | | FIO | |
| ID |
| KodZak |
| Data |
| Materials |
| SumMaterials |
| Overheads   |  |  | | --- | --- | | **SubManufacture** | **SubManufacture** | | ID | ID | | ID\_Costs | ID\_costs | | Kolvo | WorkerID | | AvrZarpl | Zarpl | |
| SemiProducts |

Как видно из таблиц, все они имеют ключевые поля (ID) которые указывают на уникальность записи в любой из таблиц. Так, в таблицах 2 и 3 существуют поля ID\_Costs, которые указывают на определенный заказ. Данное условие было необходимо, поскольку уникальным ключом таблицы COSTS является номер заказа и дата его запуска. Данные поля вместе представляют довольно большое поле-ключ, что будет создавать в других таблицах избыточность информации. Следовательно, довольно таки удобнее ввести одно уникальное поле, которое и будет служить ключом для связки таблиц. Тоже самое касается и Таблицы WorkersList. Как правило, уникальным может быть и табельный номер работника, но есть случаи, когда он не уникален: например, табельные номера каждый раз начинаются с начала в новом отделе, либо это поле имеет буквенно-цифровой вид, что приводит к увеличению обработки информации, поскольку компьютерная техника работает быстрей с цифрами, чем с символами.

Представим данные таблиц на примере.

Costs.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | KodZak | Data | Materials | SubMaterials | overheads | SemiProducts |
| 1 | 11-1 | 01.01.2003 | 10 000,00 грн. | 8 000,00 грн. | 2 000,00 грн. | 3 000,00 грн. |
| 2 | 22-2 | 01.10.2003 | 12 000,00 грн. | 7 500,00 грн. | 2 200,00 грн. | 3 500,00 грн. |
| 3 | 33-3 | 01.11.2003 | 12 250,00 грн. | 8 800,00 грн. | 2 300,00 грн. | 3 300,00 грн. |

MainManufacture

|  |  |  |  |
| --- | --- | --- | --- |
| ID | ID\_Costs | WorkerID | Zarpl |
| 1 | 1 | 1 | 300,00 грн. |
| 2 | 1 | 2 | 310,00 грн. |
| 3 | 2 | 3 | 325,00 грн. |
| 4 | 2 | 2 | 315,00 грн. |
| 5 | 3 | 1 | 300,00 грн. |
| 6 | 3 | 3 | 320,00 грн. |

SubManufacture

|  |  |  |  |
| --- | --- | --- | --- |
| ID | ID\_Costs | Kolvo | AvrZarpl |
| 1 | 1 | 5 | 200,00 грн. |
| 2 | 2 | 7 | 250,00 грн. |
| 3 | 3 | 10 | 180,00 грн. |
| 4 | 1 | 6 | 175,00 грн. |

WorkersList

|  |  |  |
| --- | --- | --- |
| ID | TabNo | FIO |
| 1 | 123 | Иванов И.И. |
| 2 | 124 | Петров П.П. |
| 3 | 125 | Сидоров С.С. |

В ходе работы были созданы запросы, которые, основываясь на данные таблиц рассчитывают промежуточные данные для сводного отчета.

Запрос “Зарплата специалистов” – рассчитывает данные(заработную плату) специалистов по всем выполненным заказам. Это второстепенные данные которые могут быть получены для работников расчетного отдела.

|  |  |  |
| --- | --- | --- |
| TabNo | FIO | Sum\_Zarpl |
| 123 | Иванов И.И. | 600,00 грн. |
| 124 | Петров П.П. | 625,00 грн. |
| 125 | Сидоров С.С. | 645,00 грн. |

Запрос Calc\_Main – рассчитывает данные(заработную плату) по заказам об участии специалистов в процессе выполнения заказа.

|  |  |
| --- | --- |
| toMainBills | ID\_Costs |
| 610,00 грн. | 1 |
| 640,00 грн. | 2 |
| 620,00 грн. | 3 |

Запрос Calc\_Sub – рассчитывает данные(заработную плату) по заказам об участии рабочих в процессе выполнения заказа.

|  |  |
| --- | --- |
| toSubBills | ID\_Costs |
| 2 050,00 грн. | 1 |
| 1 750,00 грн. | 2 |
| 1 800,00 грн. | 3 |

Выходными данными будет сводная таблица/отчет о выполнении всех заказов с учетом затрат на выплату заработной платы, а также итоговыми суммами по заказу.

Отчет о затратах на производство

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Код заказа | Дата заказа | Материалы | | Непредвиденные расходы | Полуфабри-каты | Заработная плата | | Итого |
| Основные | Вспомога-тельные | Специа-листы | Рабочие |
| 11-1 | 01.01.2003 | 10 000,00 грн. | 8 000,00 грн. | 2 000,00 грн. | 3 000,00 грн. | 610,00 грн. | 2 050,00 грн. | 25 660,00 грн. |
| 22-2 | 01.10.2003 | 12 000,00 грн. | 7 500,00 грн. | 2 200,00 грн. | 3 500,00 грн. | 640,00 грн. | 1 750,00 грн. | 27 590,00 грн. |
| 33-3 | 01.11.2003 | 12 250,00 грн. | 8 800,00 грн. | 2 300,00 грн. | 3 300,00 грн. | 620,00 грн. | 1 800,00 грн. | 29 070,00 грн. |