

БАЗЫ ДАННЫХ

часть II

Объектно ориентированные
базы данных

ООБД

В компьютерных технологиях сегодня отчетливо просматривается стремление с минимальными потерями перенести в виртуальный мир объекты мира реального. Объектно-ориентированная СУБД — именно то средство, которое обеспечивает запись объектов в базу данных «как есть». Данное обстоятельство стало решающим аргументом в пользу выбора ООСУБД для переноса семантики объектов и процессов реального мира в сферу информационных систем.

ООБД

Использование объектного подхода к проектированию систем поднимает роль ООБД как средства для наиболее естественного хранения и манипулирования создаваемыми объектами.

ООСУБД находят широкое применение в Internet — текст, картинки, видео и звук, из которых составляется Web-страница, хранятся в ООСУБД как набор объектов, подготовленный к передаче программе-клиенту, что позволяет добиться быстрой реакции сервера на запрос.

ООБД

Все большую популярность получают активные Web-серверы, на лету генерирующие страницы, используя язык Java. Практически все ведущие разработчики ООСУБД выбрали Java одним из основных для себя языков программирования.

ООБД

Единого мнения по поводу того, как конкретно следует организовывать ООСУБД, нет. Тем не менее, ряд свойств продекларированы в «Манифесте систем объектно-ориентированных баз данных», а впоследствии закреплены в документах ODMG, организации, объединяющей ведущих производителей ООСУБД.

Используемая далее терминология отражает требования стандарта ODMG 2.0.

Модель данных

Объектная модель данных характеризуется следующими свойствами:

- Базовыми примитивами являются объекты и литералы. Каждый объект имеет уникальный идентификатор, литерал не имеет идентификатора.
- Объекты и литералы различаются по типу. Все элементы одного типа имеют одинаковый диапазон изменения состояния (множество свойств) и одинаковое поведение (множество определенных операций). Объект, на который можно установить ссылку, называется экземпляром; он хранит определенный набор данных.

ООБД

Модель данных (продолжение)

- Состояние объекта определяется набором значений, реализуемых множеством свойств. Этими свойствами могут быть атрибуты объекта или связи между объектом и одним или несколькими другими объектами.
- Поведение объекта определяется набором операций, которые могут быть выполнены над объектом или самим объектом. Операции могут иметь список входных и выходных параметров строго определенного типа.
- База данных хранит объекты, позволяя совместно использовать их различным пользователям и приложениям. База данных основана на схеме данных, определяемой языком определения данных, и содержит экземпляры типов, определенных схемой.

Модель данных (продолжение)

Каждый тип имеет внешнюю спецификацию и одну или несколько реализаций. Спецификация определяет внешние характеристики типа: пользователю для работы с объектом предоставляется набор операций и набор атрибутов объекта, при помощи которых можно работать с реальными экземплярами. Реализация определяет внутреннее содержание объектов, например операции.

Тип также является объектом. Поддерживается иерархия супертипов и подтипов, реализуя стандартный механизм объектно-ориентированного программирования — наследование.

ООБД

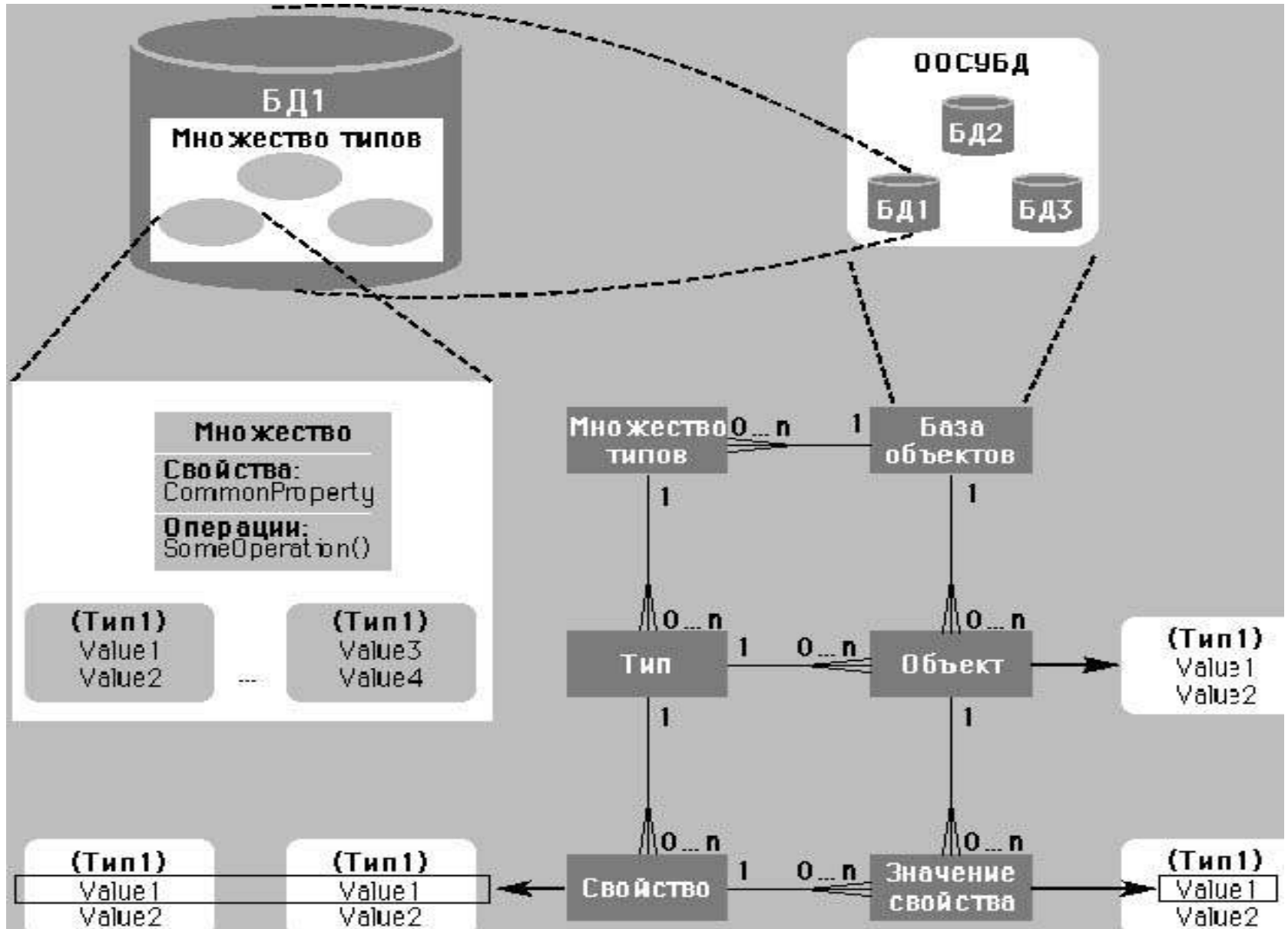


ООБД

Модель данных (продолжение)

ООСУБД обслуживает множество баз данных, каждая из которых содержит определенное множество типов. В базах данных могут содержаться объекты соответствующего типа из этого множества. Тип имеет набор свойств, а объект характеризуется состоянием в зависимости от значения каждого свойства. Операции, определяющие поведение типа, едины для всех объектов одного типа. Свойство едино для всего типа, а все объекты типа также имеют одинаковый набор свойств. Значение свойства относится к конкретному объекту.

ООБД



ООБД

Идентификатор объекта

Каждый объект в базе данных уникален. Существует несколько подходов для идентификации объекта. Самый простой — присвоить ему уникальный номер (OID — object identifier) в базе и никогда больше не повторять этот номер, даже если предыдущий объект с таким номером уже удален.

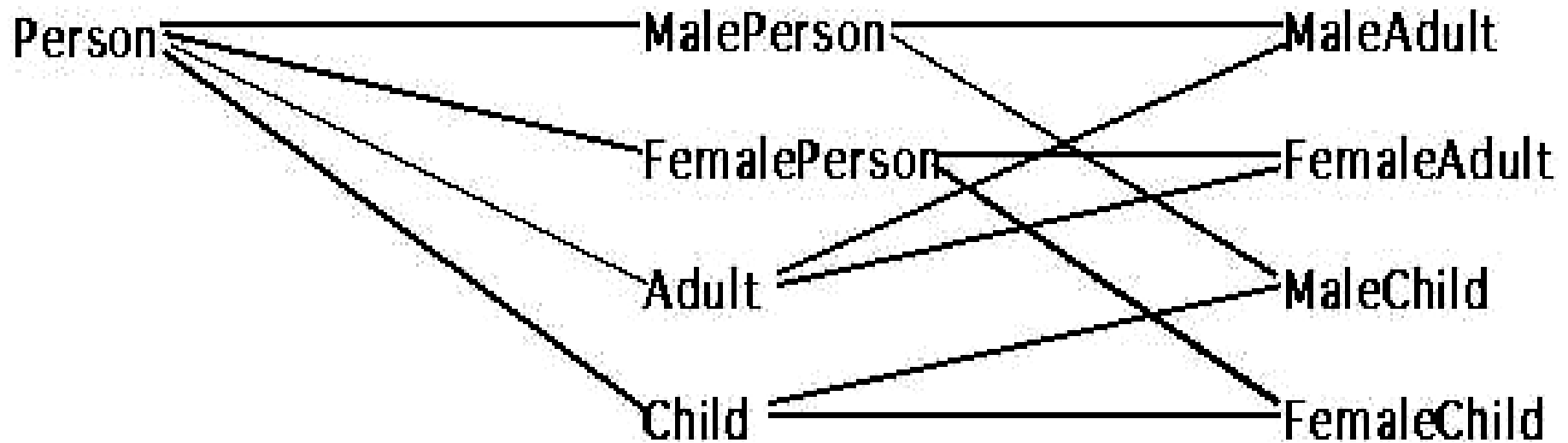
Недостаток — невозможно перенести объекты в другую базу без потери связности между ними. Решение этой проблемы заключается в использовании составного идентификатора. Например, xxxxxxxx:uuuuuuuu, где xxxxxxxx — идентификатор базы данных, uuuuuuuu — идентификатор объекта в базе.

Новые типы данных

Одним из принципиальных отличий объектных баз данных от реляционных является возможность создания и использования новых типов данных. Концептуально объект характеризуется поведением и состоянием. Определение типа заключается в определении поведения, т.е. операций, которые могут быть выполнены объектом или над состоянием объекта — набором атрибутов определенных типов (атрибут может иметь любой объявленный в базе тип).

Создание нового типа не требует модификации ядра базы и основано на принципах объектно-ориентированного программирования: инкапсуляции, наследовании, перегрузке операций и позднем связывании.

ООБД



ООБД

Функционирование базы основано на схеме данных. Любой тип является объектом, следовательно, схемы данных являются уровнем интерпретации специфических служебных объектов, использующих свойства этих объектов как схему для создания новых типов.

Схема данных может быть как первичной для создания классов, так и вторичной, выделяемой из созданных на языке программирования (скажем, на C++) классов и загружаемой в базу. Язык ODL разработан ODMG как универсальный язык описания объектов и не претендует на то, чтобы называться полноценным языком программирования. Для целей разработки предусмотрены элементы расширения классических объектных языков C++, Smalltalk, Java, позволяющих описать структуру объектов, их связи и типы связей.

ООБД

Оптимизация ядра СУБД

Ядро ООСУБД оптимизировано для операций с объектами. Естественными операциями для него являются **кэширование** объектов, **ведение версий** объектов, **разделение прав доступа** к конкретным объектам. Ядро объектно-реляционной СУБД остается реляционным, а «объектность» реализуется в виде специальной надстройки. Как следствие, ООСУБД свойственно более высокое быстродействие на операциях, требующих доступа и получения данных, упакованных в объекты, по сравнению с реляционными СУБД, для которых необходимость выборки связанных данных ведет к выполнению дополнительных внутренних операций.

ООБД

Язык запросов

Общепризнанны две группы вариантов языков запросов:

Первая объединяет языки, унаследованные от SQL и представляют собой разновидность OQL (Object Query Language), языка, стандартизованного ODMG. Объектно-реляционные СУБД используют различные варианты ограниченных объектных расширений SQL.

Вторая группа языков запросов базируется на XML. Собирает название языков этой группы — XML QL (или XQL). Они могут применяться в качестве языков запросов в объектных и объектно-реляционных базах данных.

ООБД

Транзакции

Транзакции в ООСУБД представляют логический блок, гарантирующий атомарность, целостность, изолированность и долговечность.

Атомарность: операции в рамках транзакции либо полностью выполняются, либо полностью не выполняются.

Целостность: транзакция, инициируемая в находящейся во внутренне логически связанном состоянии базе данных, приводит ее в другое логически связанное состояние.

Изолированность: ни один другой пользователь не сможет увидеть изменений, проводимых в рамках транзакции, пока не будет выполнена операция “commit”.

Долговечность: изменения, проведенные в рамках транзакции и сохраненные в базе данных, сохранятся даже в том случае, если произойдет сбой системы.

ООБД

Транзакции (продолжение)

Короткие транзакции характеризуются малым временем выполнения; они могут существовать только в рамках сеанса работы с ООСУБД. Это наиболее простой вид транзакций, реализованный во всех современных СУБД. Все объекты подлежащие изменениям блокируются, а после принятия транзакции разблокируются, изменения же записываются в базу данных.

ООБД

Транзакции (продолжение)

Длинные транзакции предназначены для увеличения производительности при групповой работе. При одновременной работе большого числа пользователей предлагается возможность организации персональной базы данных. Пользователи работают со своей базой, а объекты из нее синхронизируются с групповой базой данных. Пользователь, начав длинную транзакцию, тем самым отмечает объекты, с которыми предстоит работать в групповой базе данных (операция «поставить на контроль» — check out). Эти объекты копируются в его персональную базу, а в групповой базе блокируются.

ООБД

Транзакции (продолжение)

В групповой базе создается объект, содержащий все данные о длинных транзакциях. В случае повреждения групповой базы или физического отключения сервера групповой базы пользователь сможет продолжать работу с объектами в своей персональной базе, а после восстановления групповой базы — синхронизировать объекты. Перед завершением длинной транзакции пользователь должен поместить все измененные объекты обратно в основную базу (операция «зарегистрировать» — check in). После этого объекты копируются в основную базу, а блокировка снимается. В случае аварийного завершения длинной транзакции все изменения будут потеряны.

Транзакции (продолжение)

Вложенные транзакции по принципу функционирования аналогичны коротким. В процессе выполнения одной транзакции формируются другие. Если в текущем сеансе работает один процесс, то создается стек, а если несколько процессов — дерево транзакций.

ООБД

Блокировки

Короткие блокировки (short lock) предназначены для обеспечения последовательного доступа к данным при многопользовательском режиме работы. Они автоматически выполняются во время выполнения коротких транзакций.

Продолжительные блокировки (persistent lock) обеспечивают блокирование объектов на продолжительное время — часы, дни, недели. Применяются совместно с длинными транзакциями.

Объект может быть заблокирован несколькими способами: с исключением снятия другим процессом (hard lock); с возможностью снятия другим процессом (soft lock); по конкретным операциям.

ООБД

Перемещение объектов

Миграция объектов: постоянное их перемещение, например в другую базу данных.

Постановка на контроль (check out): копирование объектов в персональную базу данных при выполнении длинной транзакции.

Регистрация объектов (check in): копирование объектов в групповую базу данных из персональной при выполнении длинной транзакции.

ООБД

Ведение версий

Для всех объектов возможно сохранение всех версий их изменения. Создается граф происхождения версий, поддерживающий ряд свойств:

- 1. Доступ к любой ранее сохраненной версии.** Благодаря этому свойству возможно извлечение из базы данных, например, случайно удаленных данных.
- 2. Установка версии по умолчанию.** Любая порожденная версия создаст ветвь от установленной текущей, не последней версии.
- 3. Удаление версии.** Существует необходимость удаления некоторых версий (например, промежуточных или случайно порожденных) из графа происхождения версий.

ООБД

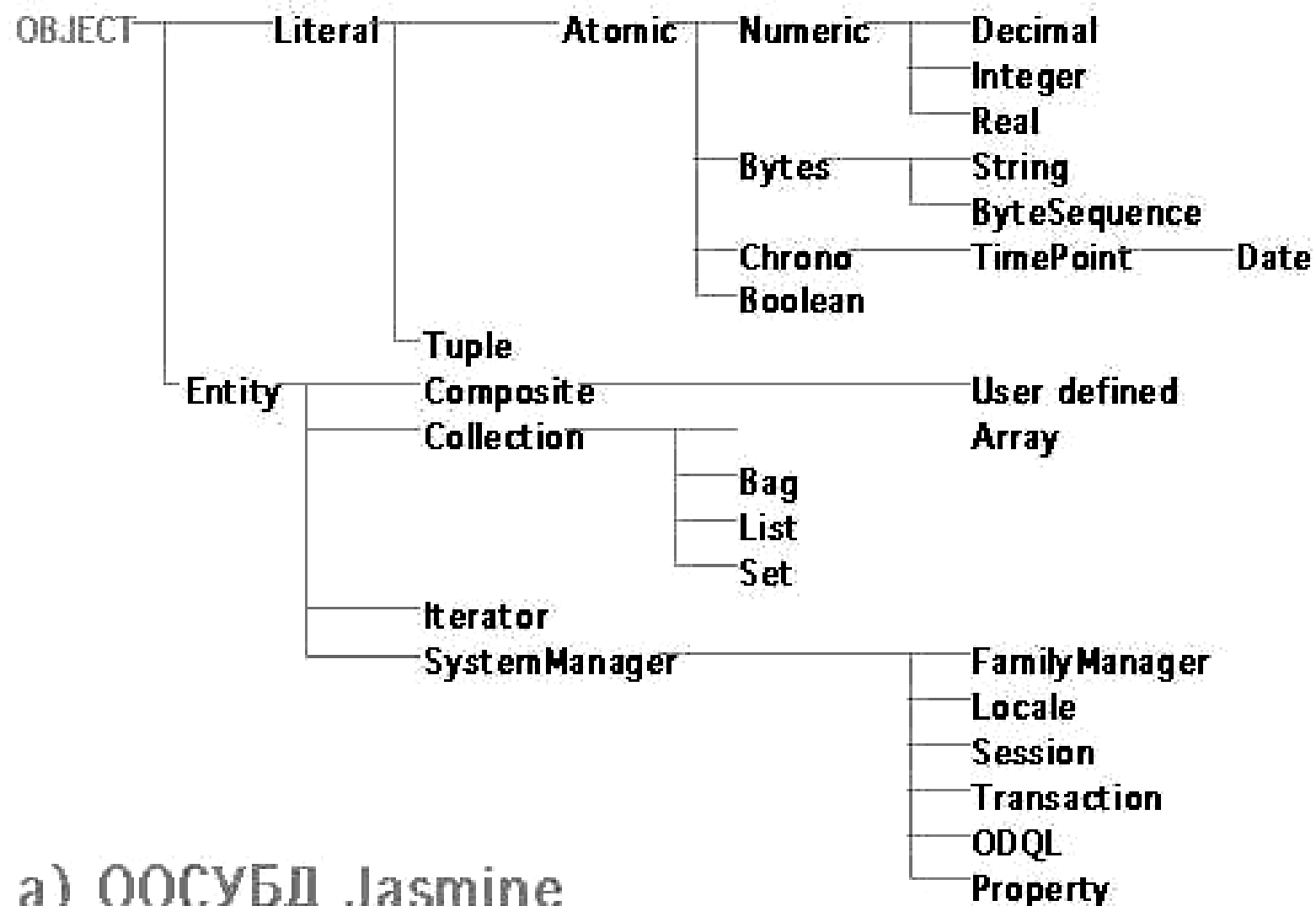
Физические хранилища

Один из ключевых моментов функционирования любой СУБД — хранилище данных. Выделяют три типа хранилищ:

- 1. системное хранилище**, используемое для хранения системы классов, создается на этапе формирования базы данных и содержит информацию о классах, о наличии и месторасположении пользовательских хранилищ;
- 2. пользовательское хранилище** для хранения пользовательских объектов;
- 3. служебное хранилище**, содержащее временную информацию, например сведения о заблокированных объектах, об активных транзакциях, различного вида списки запросов пользователей и т.д.

ООБД

Иерархия типов



a) ООСУБД Jasmine

Особенности реализации некоторых ООСУБД



ООБД

ВОПРОСЫ ?

СОВЕТ ДНЯ:

История:

Дочь вертится перед зеркалом рассматривая новую шубку.

«Мама, какая красивая! Наверное, ты истратила всю папину зарплату?!»

«Доченька! Если бы я только на папу надеялась – вас бы не было...»

**Не надейтесь только на преподавателя!
Ищите материал сами...**