

КОНЦЕПТУАЛЬНОЕ ПРОЕКТИРОВАНИЕ БД

1. НЕОБХОДИМОСТЬ КОНЦЕПТУАЛЬНОГО ПРОЕКТИРОВАНИЯ БД.
2. ОПИСАНИЕ ОБЪЕКТОВ И ИХ СВОЙСТВ.
3. ОПИСАНИЕ СВЯЗЕЙ МЕЖДУ ОБЪЕКТАМИ.

ВВЕДЕНИЕ

Концептуальное проектирование является основой всего процесса проектирования БД. Для того чтобы БД адекватно отражала предметную область, проектировщик БД должен хорошо представлять себе все особенности, присущие данной предметной области, и уметь отображать их в БД. Поэтому перед началом проектирования БД

необходимо детально разобраться, как функционирует предметная область, для отображения которой создаётся БД. На этом этапе важным является участие заказчика, например, менеджера или экономиста.

Целью концептуального проектирования является создание инфологической модели (ИЛМ) предметной области (ПО), которая представляет собой описание ПО, выполненное с помощью специальных языковых средств, не зависящих от используемых в дальнейшем программных средств.

1. НЕОБХОДИМОСТЬ КОНЦЕПТУАЛЬНОГО ПРОЕКТИРОВАНИЯ БД

Желательно, чтобы язык спецификации ИЛМ был одинаково применим как при ручном, так и при автоматизированном проектировании БД. Для этого он должен:

- 1) быть вычисляемым, т.е. восприниматься и обрабатываться компьютером;
- 2) использовать удобный пользователю интерфейс, в частности, графический;

3) быть независимым от оборудования и других ресурсов, подверженных частным изменениям;

4) использовать средства тестирования ИЛМ, а также иметь аппарат для генерации структуры БД после завершения спецификации ИЛМ.

ИЛМ должна легко восприниматься разными категориями специалистов, участвующих в создании информационной системы. Она должна содержать необходимую и достаточную информацию для проектирования базы данных.

ИЛМ включает в себя ряд компонентов, изображённых на рисунке 1.

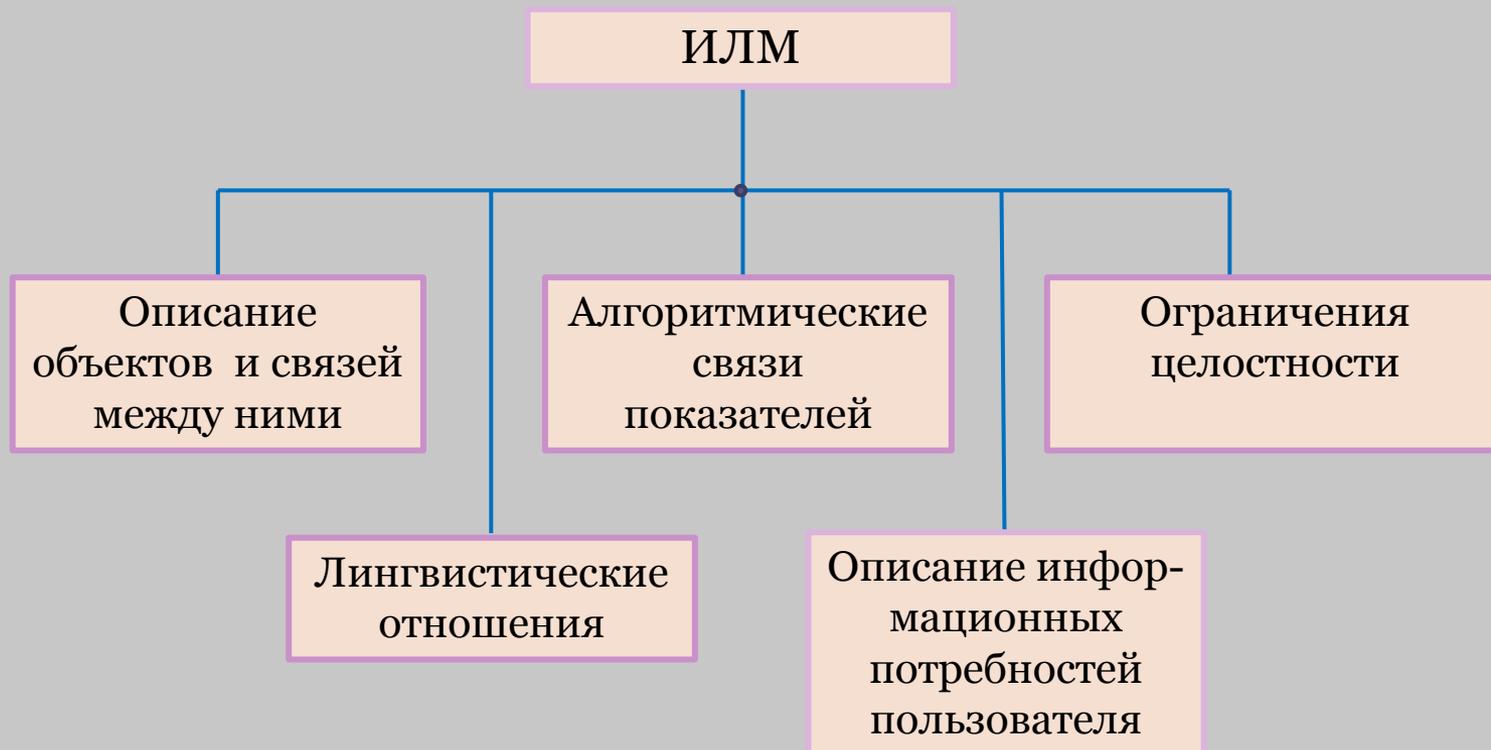


Рисунок 1 – Компоненты ИЛМ

Основным компонентом является **описание объектов предметной области и связей между ними**. Описание предметной области всегда представляется с помощью какой-либо знаковой системы. Поэтому в ИЛМ отражаются не только отношения, присущие предметной области, но и лингвистические отношения, обусловленные особенностями отображения ПО в языковой среде. Поэтому нужно учитывать такие лингвистические категории как «синонимия», «омонимия» и др.

В ИЛМ должны быть отражены и алгоритмические зависимости между показателями. Для этих целей можно использовать графы взаимосвязи показателей (рис. 2), отражающие, какие показатели служат исходными для вычисления других показателей.

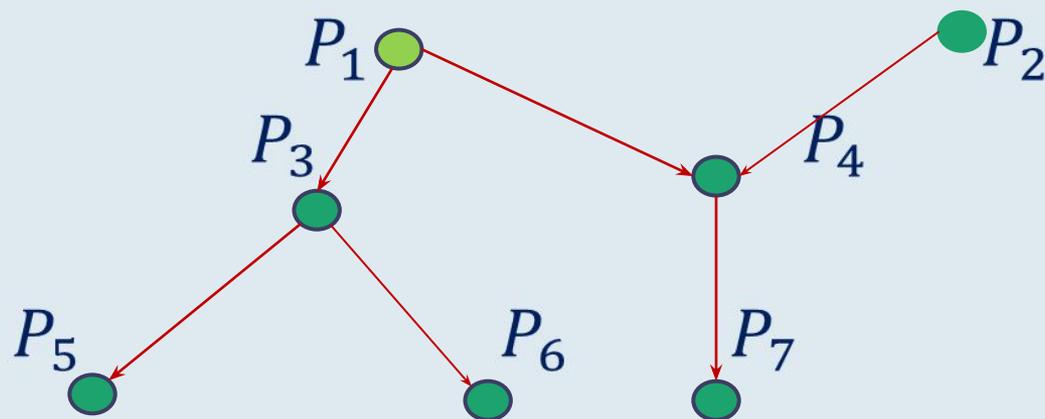


Рисунок 2 – Граф взаимосвязи показателей

Следующим компонентом является описание информационных потребностей пользователей. Это описание должно отображать типы запросов к БД, режим использования данных и т.д.

Важными характеристиками предметной области являются **ограничения целостности**, то есть условия, при которых имеют смысл значения атрибутов, хранящихся в БД, или условия, при которых эти значения могут оказаться записанными в БД.

Проблема целостности данных состоит в обеспечении правильности данных в БД в любой момент времени. Целостность данных обеспечивается набором специальных условий или утверждений, называемых **ограничениями целостности**. Ряд этих ограничений нами были рассмотрены ранее при изучении основных понятий БД, в частности, этой проблеме был посвящён вопрос «**Целостность сущности и ссылок**».

Мы ограничимся рядом дополнений и замечаний, относящихся к проблеме целостности данных. Ограничения целостности могут относиться к разным информационным единицам: *атрибутам, кортежам, отношениям, связям между отношениями* и т.п. Эти ограничения определяются чаще всего особенностями предметной области, хотя могут отображать и чисто информационные характеристики.

Для атрибутов в большинстве случаев используются следующие виды ограничений:

1. Тип и формат атрибута.
2. Задание диапазона значений атрибута.
3. Признак непустого значения атрибута.
4. Задание домена.

Специфическим ограничением на значение атрибута является признак его **уникальности**. Этот признак тесно связан с понятием *первичного ключа*, который, как мы знаем, может быть представлен не только одним атрибутом, но может быть и составным.

Когда речь идёт об ограничениях целостности, относящихся к **кортежу**, то имеется в виду либо ограничение на значения всей строки, рассматриваемой как единое целое, либо ограничения на соотношения значений отдельных полей в пределах одной строки. Часто используются ограничения, затрагивающие несколько взаимосвязанных отношений. Наиболее часто встречающееся из этих ограничений – ограничение целостности **связи** отношений.

Оно выражается в том, что значение атрибута, отражающего связи между объектами и являющегося внешним (вторичным) ключом отношения, **обязательно должно совпадать с одним из значений атрибута, являющегося первичным ключом отношения, описывающего соответствующий объект.** Кроме того, ограничения, отражающие связь отношений, могут представлять собой условия, проверяющие **отсутствие логических противоречий** между данными взаимосвязанных отношений.

Очень важным видом ограничений целостности являются *функциональные зависимости*. Информация об имеющихся в данной ПО функциональных зависимостях фиксируется в ИЛМ и используется при проектировании БД и для контроля целостности при функционировании БД. Для соответствующих полей в БД желательно задать запрет на обновление, который может относиться не только к отдельному полю, но и ко всей строке или отношению.

Разные СУБД обладают набором средств для обеспечения целостности данных. Так, некоторые реляционные СУБД поддерживают концепцию первичного ключа, домена и внешнего ключа. При этом соответствующие проверки ограничений целостности выполняются автоматически. В некоторых системах при описании структуры БД для поля можно задать запрет содержать пустое значение (NOT NULL), диапазон значений поля и т.д.

Нарушение целостности данных может возникнуть в результате ошибок при вводе данных или выполнении иных корректировок, а также в результате сбоев в работе вычислительной системы. Наличие в БД главных и подчинённых таблиц приводит к дублированию данных, а любое явное или неявное дублирование требует специальных мер по обеспечению целостности данных. Реализация контроля за соблюдением ограничений целостности требует в некоторых случаях больших затрат.

2. ОПИСАНИЕ ОБЪЕКТОВ И ИХ СВОЙСТВ

Инфологическая модель предметной области может описываться как аналитическими, так и графическими средствами. Графическое представление является наиболее наглядным и простым для восприятия и анализа. Кроме того, графическое представление поддерживается системами автоматизации проектирования БД.

Для описания объектов и их связей мы воспользуемся графическим способом отображения, позволяющим построить модель, условно называемую **«объект-свойство-связь»** или модель **«сущность-связь»**.

При исследовании и анализе предметной области заказчик или разработчик выделяет **классы объектов**.

Классом объектов называют *совокупность объектов, обладающих одинаковым набором свойств*.

Например, если в качестве предметной области рассмотреть работу приёмной комиссии по приёму ЕГЭ в школе, то в ней можно выделить следующие классы объектов: *учащиеся, экзаменаторы, аудитории* и т.д. Объекты могут быть **материальными**, как перечисленные выше, или **абстрактными**, например, *предметы*, по которым учащиеся сдают экзамены. Каждый объект представляется своим уникальным идентификатором, по значению которого один объект класса отличается от другого объекта этого же класса.

Так, для объектов класса, именуемого «ПРЕДМЕТ», уникальным идентификатором каждого объекта будет название предмета (например, математика). Каждый объект обладает определённым набором свойств. Для объектов одного класса набор этих свойств одинаков, а значения свойств могут отличаться. Например, все объекты класса «УЧАЩИЙСЯ» имеют такие свойства как «Номер паспорта», «ФИО», «Пол», «Год рождения», «Номер школы» и др.

При описании предметной области надо изобразить каждый существенный класс объектов и набор свойств, фиксируемый для объекта данного класса. Абстрактный объект, являющийся обобщённым представителем класса, изображается прямоугольником, в котором записывается имя класса (например, УЧАЩИЙСЯ). Свойство объекта изображается пунктирным прямоугольником, в котором записано свойство (например, номер паспорта).

Каждому классу объектов в ИЛМ присваивается уникальное имя. Именем класса объектов является грамматический оборот *существительного*, у которого могут быть *прилагательные* и *предлоги*. Если имя состоит из нескольких слов, то первым должно стоять существительное, употребляемое в *единственном числе*. Например, правильным именем класса объектов будет «ТОВАР ПРОДОВОЛЬСТВЕННЫЙ», а не «ПРОДОВОЛЬСТВЕННЫЕ ТОВАРЫ».

Если в предметной области традиционно используются разные имена для обозначения какого-либо класса объектов или различные названия свойств объектов (т.е. имеет место синонимия), то все они должны быть зафиксированы в ИЛМ в виде лингвистических отношений. Затем *одно* из этих имён или названий выбирается за *основное*, и только это имя должно в дальнейшем использоваться в ИЛМ. При описании ПО надо *отразить связи между объектом и характеризующими его свойствами в виде линий*, соединяющих обозначение объекта и его свойств (рис.3).

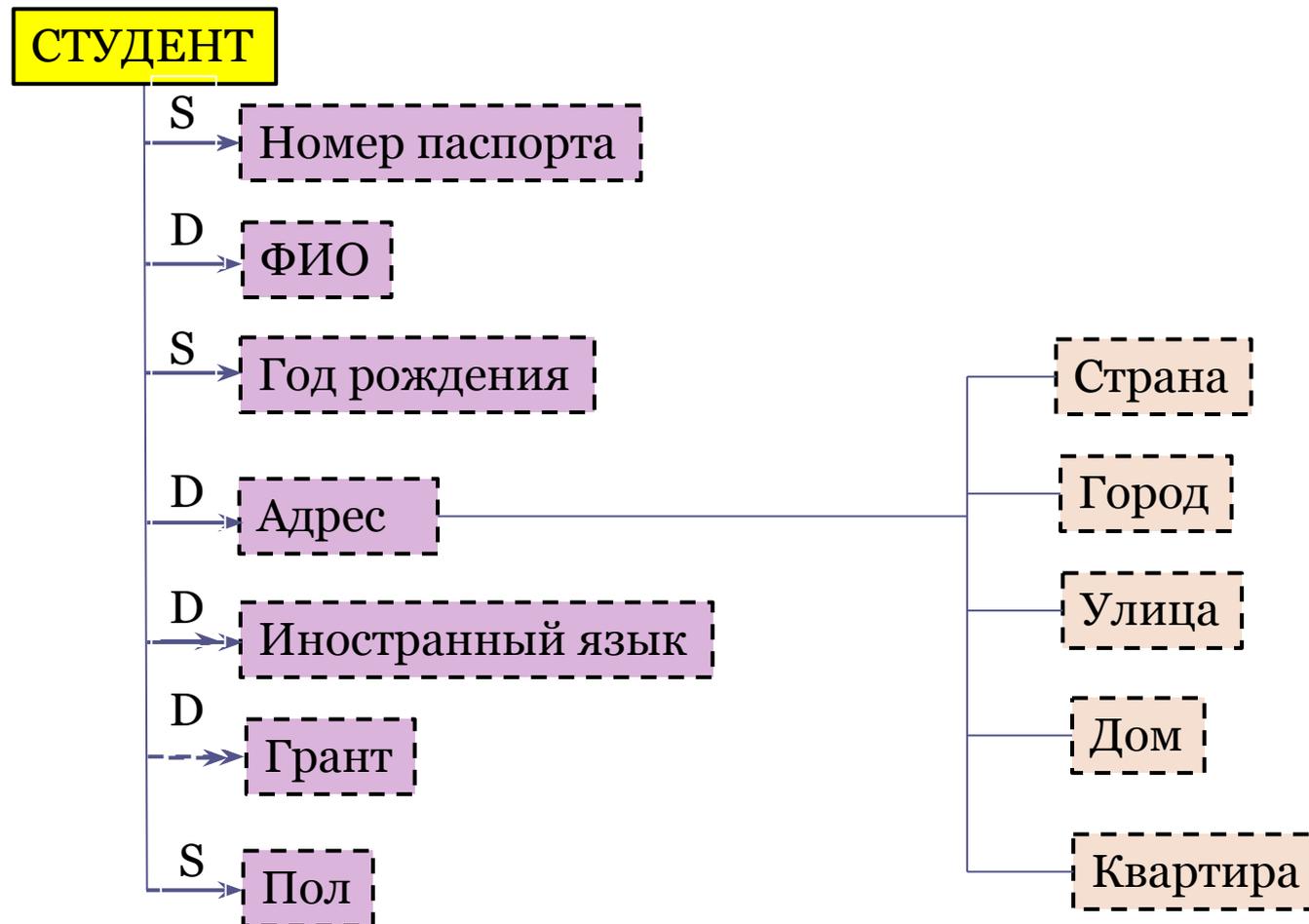


Рисунок 3 – Изображение связи «объект-свойство»

Изображение связи между объектом и его свойством учитывает специфику этого свойства. Так, объект может обладать только одним значением какого-то свойства (например, дата рождения). Такие свойства называют **единичными**. Для других объектов возможно одновременное существование нескольких значений у одного и того же свойства (например, студент может изучать несколько иностранных языков). Такое свойство называют **множественным**.

Связь объекта и единичного свойства обозначается *одинарной стрелкой*, а связь объекта и множественного свойства – *двойной стрелкой*. Кроме того, некоторые свойства являются постоянными, их значение не может измениться с течением времени (например, год рождения). Такие свойства называются *статическими*. Свойства, значения которых могут изменяться со временем (например, адрес), называются *динамическими*. Обозначены они на рис.3 соответственно буквами S и D.

Другой особенностью свойства является присутствие этого свойства у всех объектов данного класса либо отсутствие у некоторых объектов. Напр., отдельным студентам могут предоставляться один или несколько грантов различными организациями, а другие студенты грантов не имеют. Такое свойство называется *условным*. Связь условного свойства с объектом изображается *пунктирной* линией. *Составное* свойство (напр., адрес) в ИЛМ обозначается квадратом, из которого выходят линии, соединяющие его с элементами составного свойства.

3. ОПИСАНИЕ СВЯЗИ МЕЖДУ ОБЪЕКТАМИ

Помимо связи между объектом и его свойствами в ИЛМ фиксируются связи между объектами разных классов. Как известно, различают *типы* связей $1:1$, $1:M$, $M:1$ и $M:M$. Тип связи называют также *степенью* связи. Кроме степени связи в ИЛМ для характеристики связи между объектами разных классов надо указывать так называемый *класс принадлежности*.

Класс принадлежности показывает, должен или не должен объект одного класса участвовать в связи с каким-либо объектом другого класса. Класс принадлежности объекта может быть обязательным или необязательным. Если все объекты данного класса должны участвовать в связи, то класс принадлежности называется **обязательным**. Если некоторые объекты данного класса могут не участвовать в связи, то класс называется **необязательным**.

В ИЛМ связи между объектами с учётом степени связи и класса принадлежности описываются в большинстве случаев с помощью так называемых **ER-диаграмм** (ER – первые буквы английских слов Entity (сущность, объект) и Relationship (связь)). Эти диаграммы впервые были предложены *Ченом* (Chen P.P.S.) в 1979 г. Они называются диаграммами **«сущность-связь»** или **«объект-связь»**. Рассмотрим пример.

Предположим, что предметной областью является кафедра университета, преподаватели которой читают дисциплины, имеющиеся в учебном плане. Двумя главными классами объектов, или сущностями, представляющими интерес, являются **ПРЕПОДАВАТЕЛЬ** и **КУРС**. Эти две сущности соотносятся с помощью связи **ЧИТАЕТ**, что позволяет записать:

ПРЕПОДАВАТЕЛЬ ЧИТАЕТ КУРС

Связь **ЧИТАЕТ** может быть представлена двумя способами (рисунки 4 и 5).

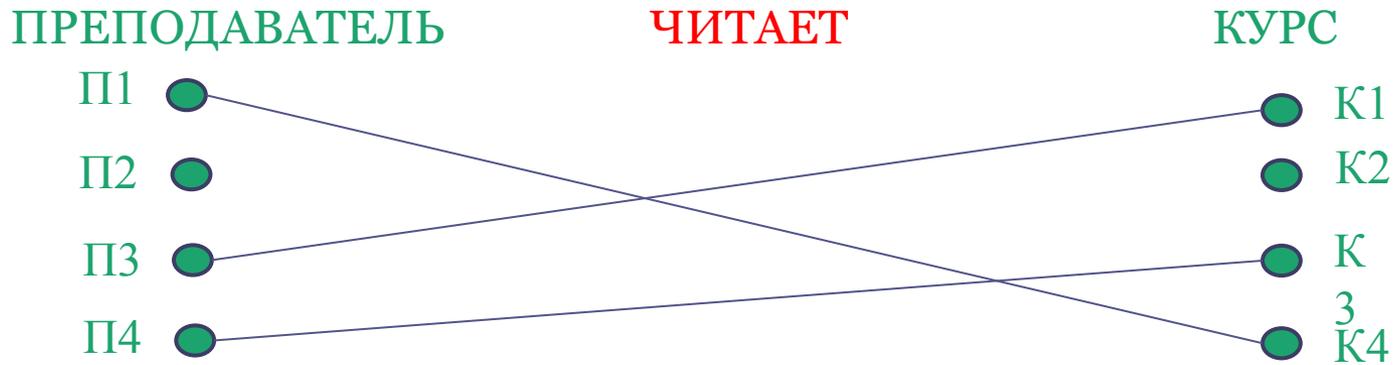


Рисунок 4 – Пример диаграммы ER-экземпляра

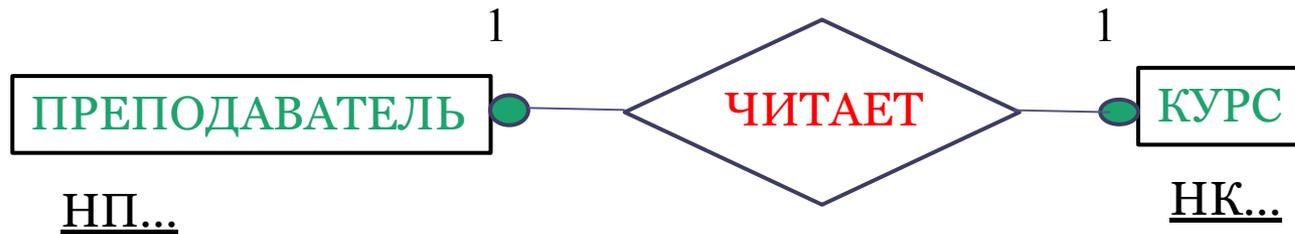


Рисунок 5 – Пример диаграммы ER-типа

На рис.4 связь сущностей представлена с помощью диаграммы ER-экземпляров. На этой диаграмме показано, какой конкретно курс читает каждый преподаватель. На диаграмме ER-экземпляров названия всех сущностей помещены над экземплярами этих сущностей и в этих названиях использованы прописные буквы, в то время как каждый экземпляр сущности идентифицируется значением атрибута. Так, КУРС является сущностью, а К1 – конкретным экземпляром сущности.

Связь также должна иметь имя, и это имя, составленное из прописных букв, размещается над экземплярами связи, при этом экземпляр каждой отдельной связи задаётся линией между теми двумя экземплярами сущностей, которые эта связь соединяет. Например, экземпляр связи между К1 и ПЗ означает, что преподаватель с номером ПЗ читает курс с номером К1. Атрибут или набор атрибутов, используемый для идентификации экземпляра сущности, называется **ключом сущности**.

На диаграмме ER-типа (рис.5) сущности представляются в виде прямоугольников, а связи – в виде ромбов. Ниже каждой сущности размещается подчёркнутый атрибут или набор атрибутов, являющийся ключом сущности для этой сущности. Цифры «1» рядом с сущностями указывают степень связи 1:1, а кружочки на линиях, соединяющих ромб с прямоугольниками, обозначают необязательный класс принадлежности. Иногда используют упрощённый способ изображения связи на диаграмме ER-типа (рис.6):

- 1:1 
- 1:M 
- M:1 
- M:M 

Рисунок 6 – Альтернативные способы изображения связей сущностей с учётом степеней связи

Диаграммы, изображённые на рисунках 4 и 5 соответствуют случаю, когда каждый преподаватель читает не более одного курса, и каждый курс читается не более чем одним преподавателем. Возможные варианты представлены на рисунке 7.

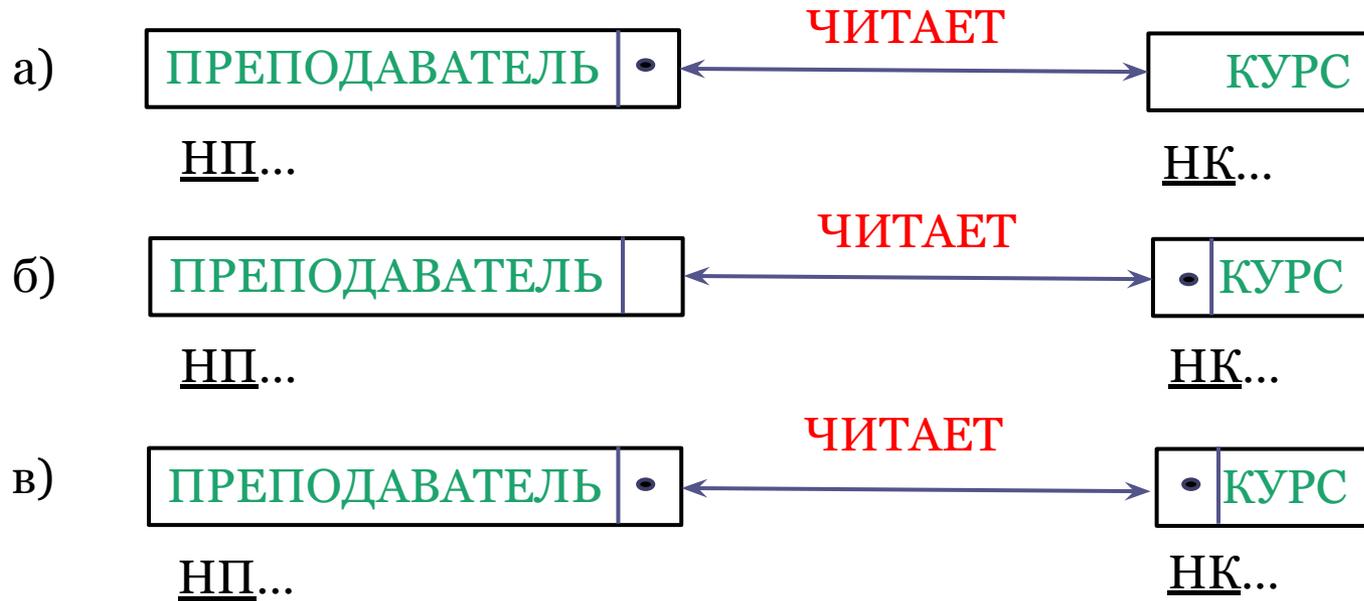


Рисунок 7 – Диаграммы ER-типа для случая степени связи 1:1 и различных классов принадлежности

Диаграмма ER-типа, изображённая на рис.7 а), соответствует случаю, когда каждый преподаватель читает только один курс, а каждый курс читается не более, чем одним преподавателем. Поскольку все преподаватели участвуют в чтении какого-либо курса, то класс принадлежности сущности **ПРЕПОДАВАТЕЛЬ** является обязательным, и этот факт отмечается точкой в блоке, смежном с прямоугольником сущности.

Диаграмма ER-типа, изображённая на рис.7 б), соответствует случаю, когда каждый преподаватель читает не более одного курса (т.е. 1 или 0 курсов), а каждый курс читается только одним преподавателем. Наконец, диаграмма ER-типа, изображённая на рис.7 в), соответствует случаю, когда каждый преподаватель читает только один курс и каждый курс читается только одним преподавателем. Во всех трёх случаях степень связи равна 1:1, а сочетания классов принадлежности сущностей различаются.

Помимо рассмотренных случаев организации чтения преподавателями курсов возможны и другие варианты, например:

- 1) каждый преподаватель может читать в одном семестре несколько курсов, но каждый курс читается не более чем одним преподавателем;
- 2) каждый преподаватель читает не более одного курса в семестре, но каждый курс может читаться сразу несколькими преподавателями;

3) каждый преподаватель может читать в одном семестре несколько курсов (в частности, ни одного курса), и каждый курс может читаться несколькими преподавателями (в частности, вообще не читаться). Этим случаям соответствуют степени связи 1:М, М:1, М:М и различные сочетания классов принадлежности сущностей. На рисунках 8-10 приведены диаграммы ER-типа, отражающие все возможные случаи организации чтения курсов преподавателями.

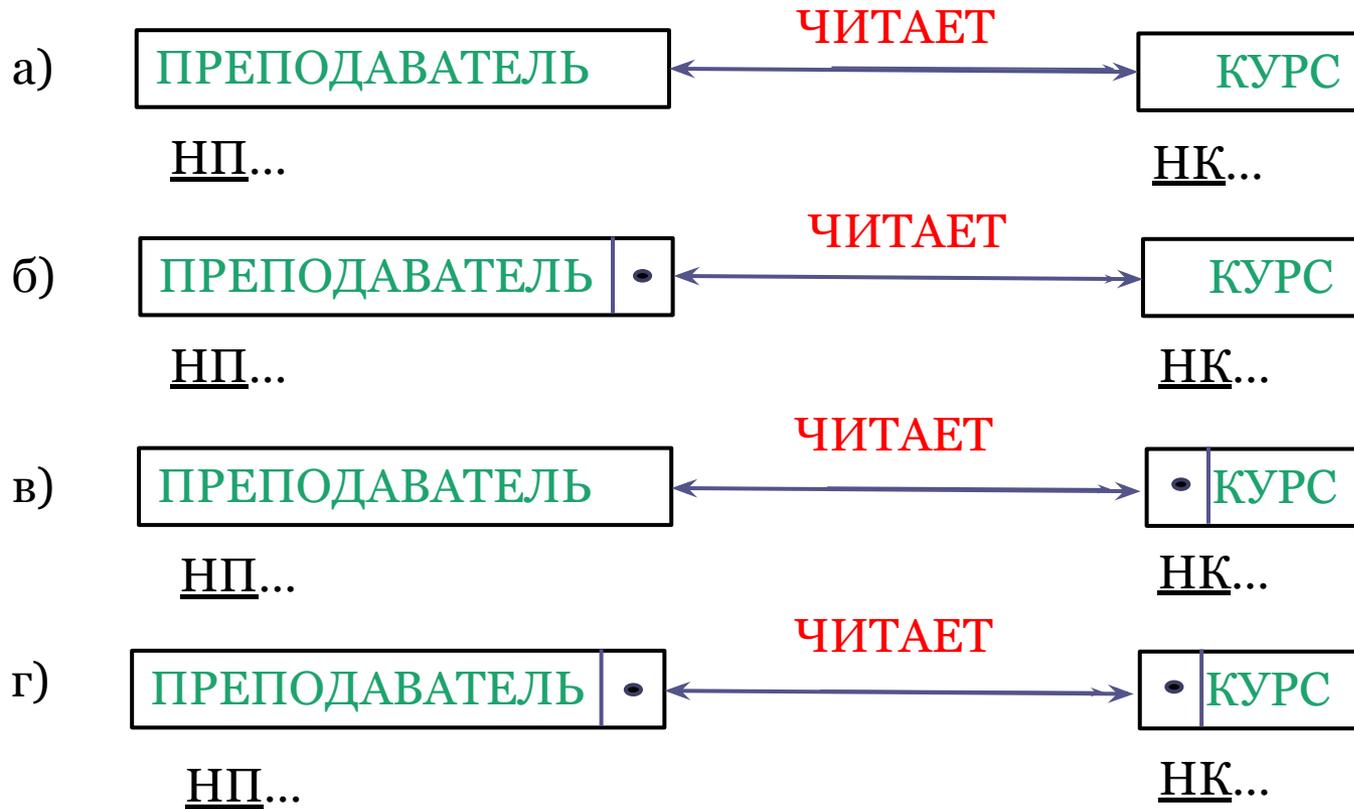


Рисунок 8 – Диаграммы ER-типа для случая связи степени 1:M

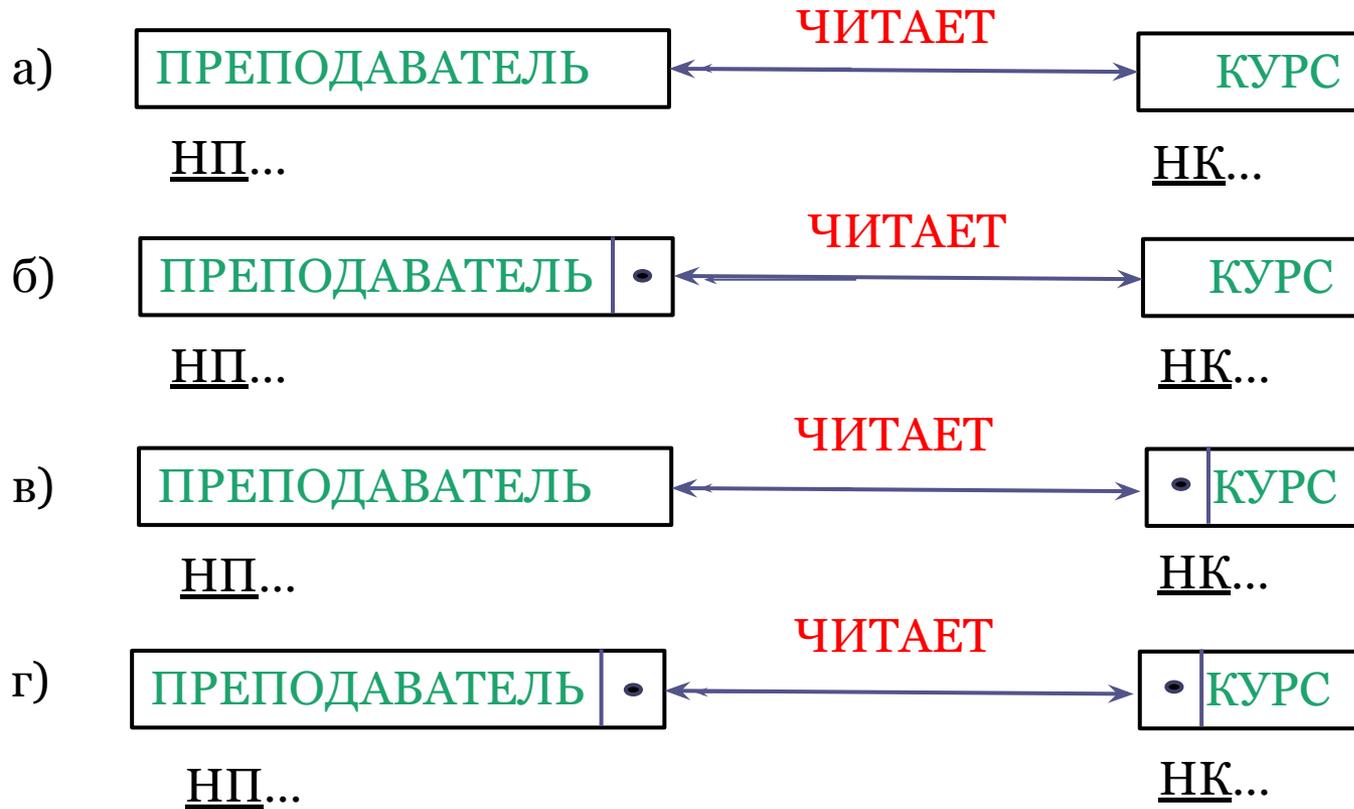


Рисунок 9 – Диаграммы ER-типа для случая связи степени M:1

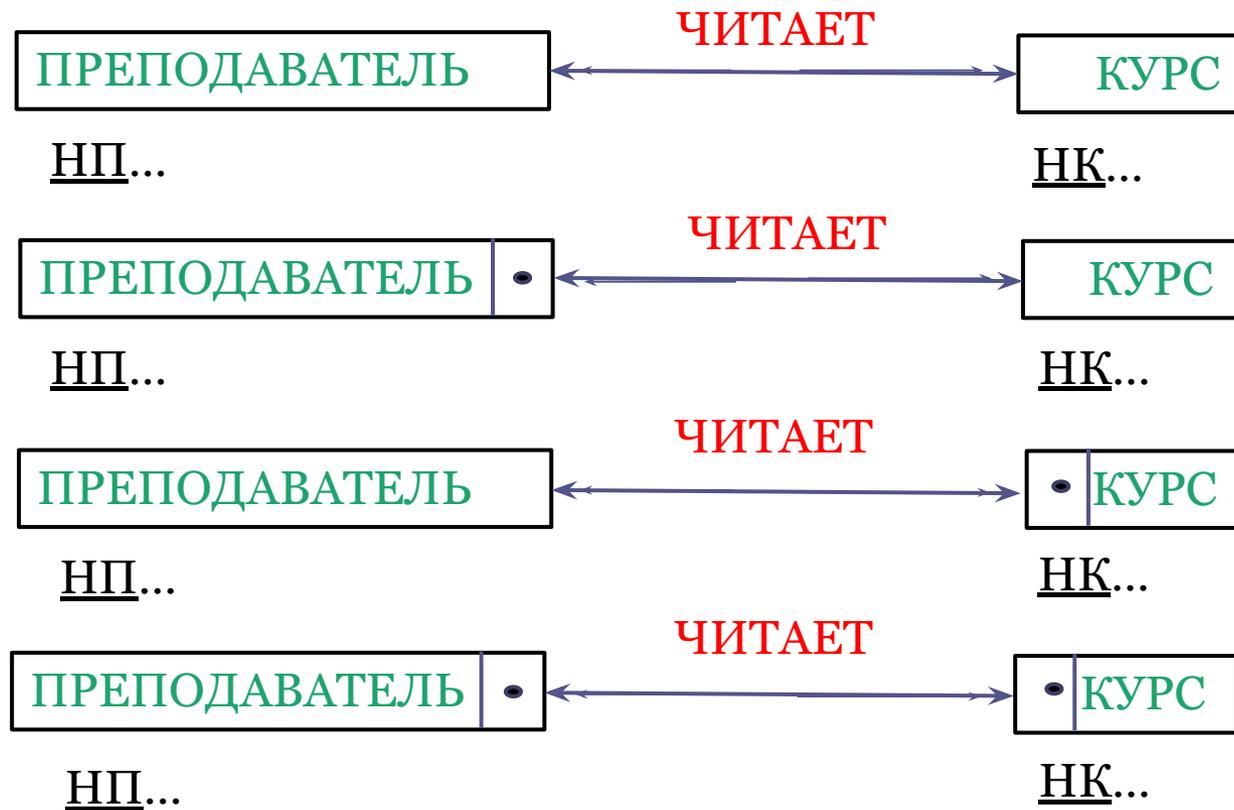
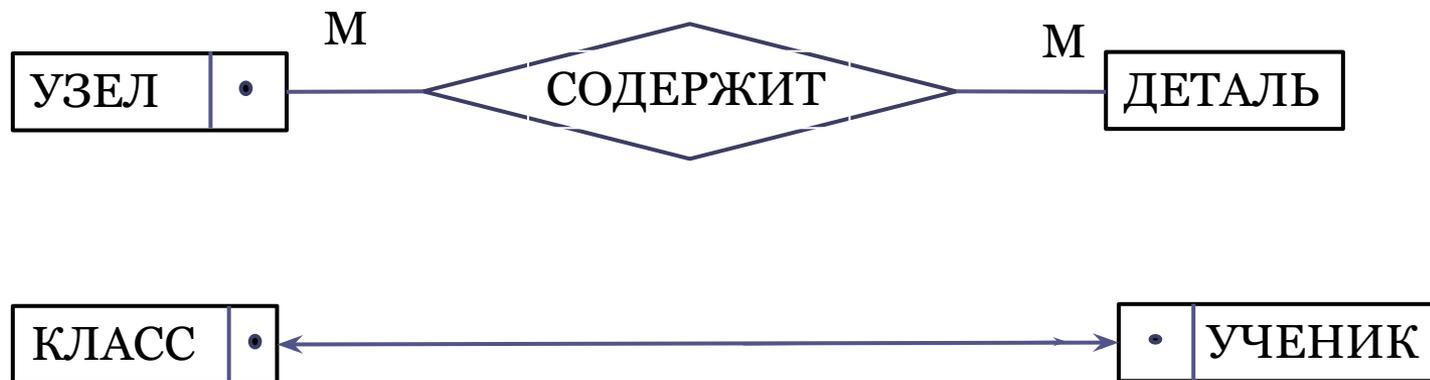


Рисунок 10 – Диаграммы ER-типа для случая связи степени M:M

По степени сложности объекты могут быть **простыми** и **сложными**. Объект считается простым, если он рассматривается как неделимый. Сложный объект представляет собой объединение других объектов, также выделяемых в предметной области. Различают несколько разновидностей сложных объектов: **составные, обобщённые** и **агрегированные**. Составной объект соответствует отображению «целое-часть», например, УЗЛЫ – ДЕТАЛИ, КЛАСС – УЧЕНИКИ и т.п.

Для представления *составных* объектов в ИЛМ используются диаграммы ER-типа, например:



Обобщённый объект отражает наличие связи «род-вид» между объектами предметной области. На рисунке 11 показан фрагмент ИЛМ, представляющий обобщённый объект ЛИЧНОСТЬ для университета.

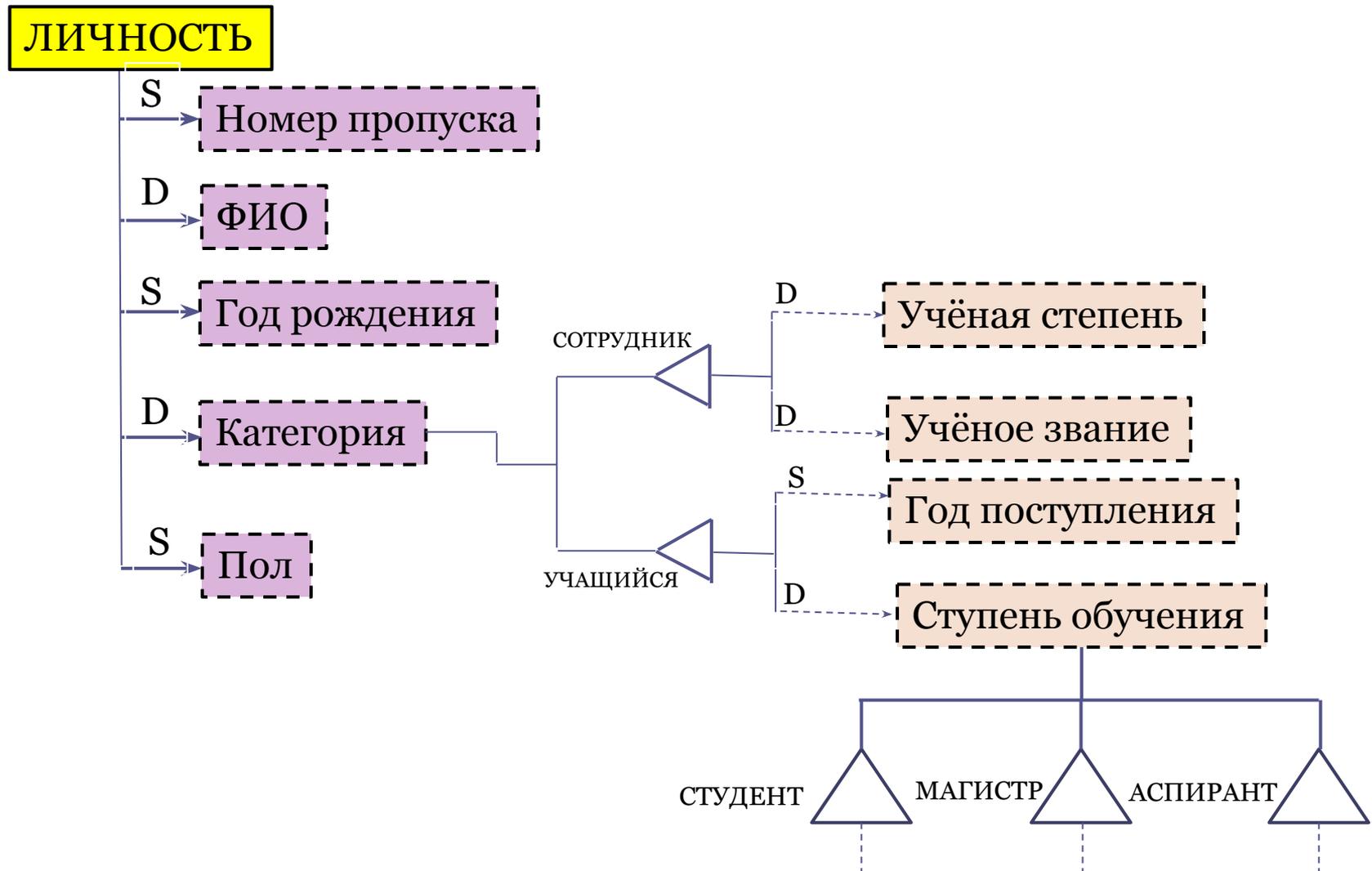


Рисунок 11 – Изображение обобщённого объекта

Агрегированный объект обычно соответствует какому-либо процессу, в который оказываются вовлечёнными другие объекты. Агрегированный объект именуется отглагольным существительным (например, *поставлять* – *поставка*, *выпускать* – *выпуск* и т.д.). В ИЛМ агрегированный объект изображается ромбом, в котором указано имя объекта. Этот ромб соединяется линиями с условными обозначениями объектов, которые образуют агрегированный объект (см. рис. 12).

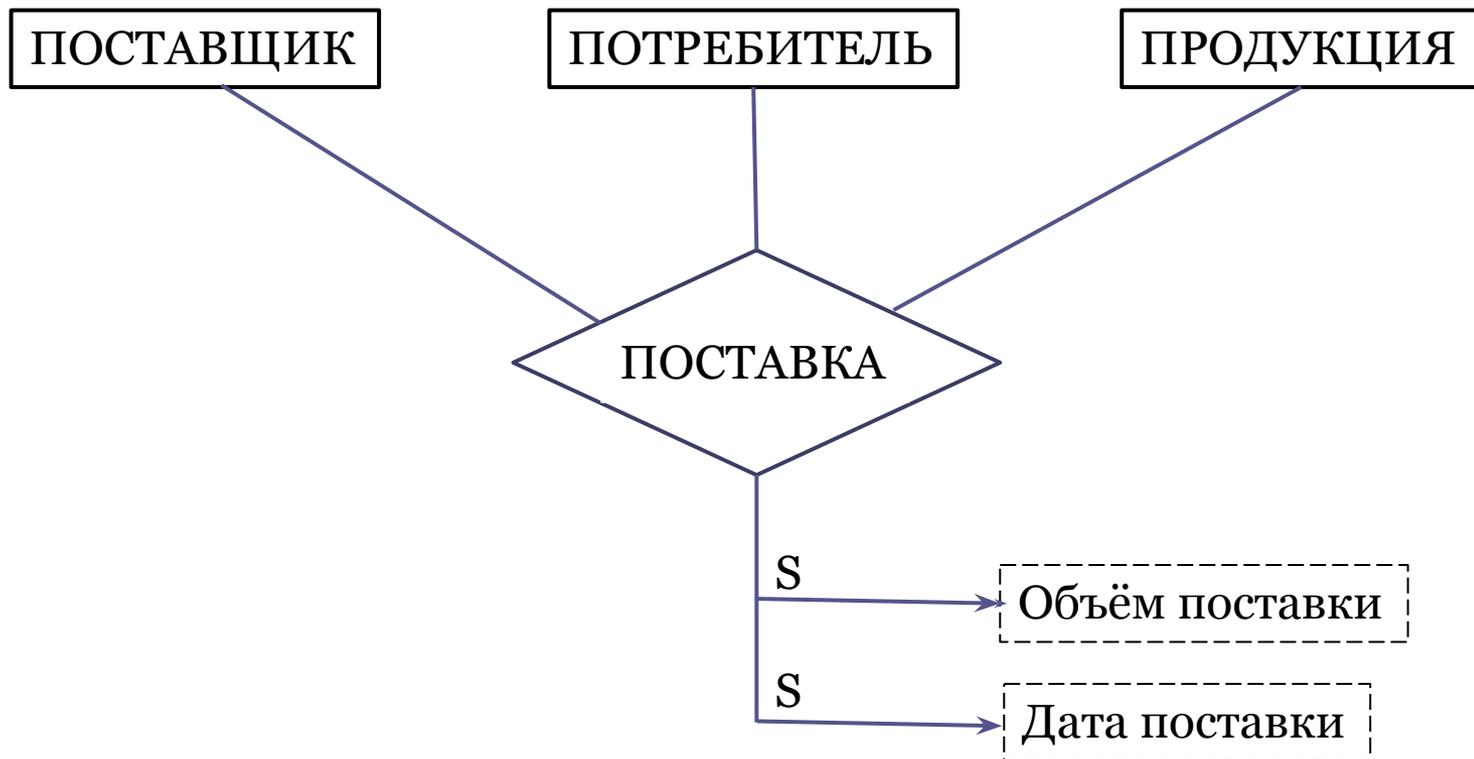


Рисунок 12 – Изображение агрегированного объекта

Агрегированный объект **ПОСТАВКА** объединяет в себе объекты **ПОСТАВЩИК**, **ПОТРЕБИТЕЛЬ**, **ПРОДУКЦИЯ**. Свойства агрегированного объекта изображаются пунктирными прямоугольниками.

ЗАКЛЮЧЕНИЕ

Проектирование БД – это всегда отдельная *творческая* задача, не имеющая полностью формализованного решения. Основная цель семантического моделирования состоит в том, чтобы сделать СУБД более «разумными». А более конкретная цель заключается в предоставлении некоторого систематического подхода к решению проблемы проектирования БД.

На данной лекции мы рассмотрели применение одной из *семантических* моделей, предложенной Ченом, для решения указанной проблемы, а именно модель «*сущность-связь*», иначе называемой *ER-моделью*. Широкую популярность ER-модели можно объяснить наличием в этой технологии использования *диаграмм*, а не какой-либо другой причиной. Однако для использования технологии ER-диаграмм не обязательно принимать все идеи этой модели.

РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА

1. *Дейт, К., Дж.* Введение в системы баз данных, 7-е издание: пер. с англ. – М.: Издательский дом «Вильямс», 2001. – 1072 с.
2. *Илюшечкин, В.М.* Основы использования и проектирования баз данных: учеб. пособие/В. М. Илюшечкин. – М.: Высшее образование, 2009. -213 с. – (Основы наук).
3. *Мирошниченко, Г.А.* Реляционные базы данных: практические приёмы оптимальных решений. – СПб.: БХВ-Петербург, 2005. – 400 с.