

# Тема урока: Диаграммы потоков данных

Цель занятия: освоить назначение и принципы построения DFD - диаграммы потоков данных;

Ключевое слово: (DataFlowDiagramming)-DFD

## План работы:

1. Работы;
2. внешние сущности (ссылки);
3. потоки работ;
4. хранилища данных.

*Цель:* Ознакомиться с технологией построения DFD - диаграмм с помощью CASE – средства VPWIN

*Задачи:*

- получить общие сведения о моделировании потоков данных, о компонентах диаграммы потоков данных DFD;
- ознакомиться со средствами моделирования потоков данных в среде VPwin;
- закрепить полученные знания в ходе самостоятельной работы.

# Задание

- Администрация больницы заказала разработку информационной системы для отдела приема пациентов и медицинского секретариата. Новая система предназначена для обработки данных о врачах, пациентах, приеме пациентов и лечении. Система должна выдавать отчеты по запросу врачей или администрации. В результате предпроектного обследования составлено следующее описание деятельности подразделений.

# Диаграммы потоков данных (Data Flow Diagramming)

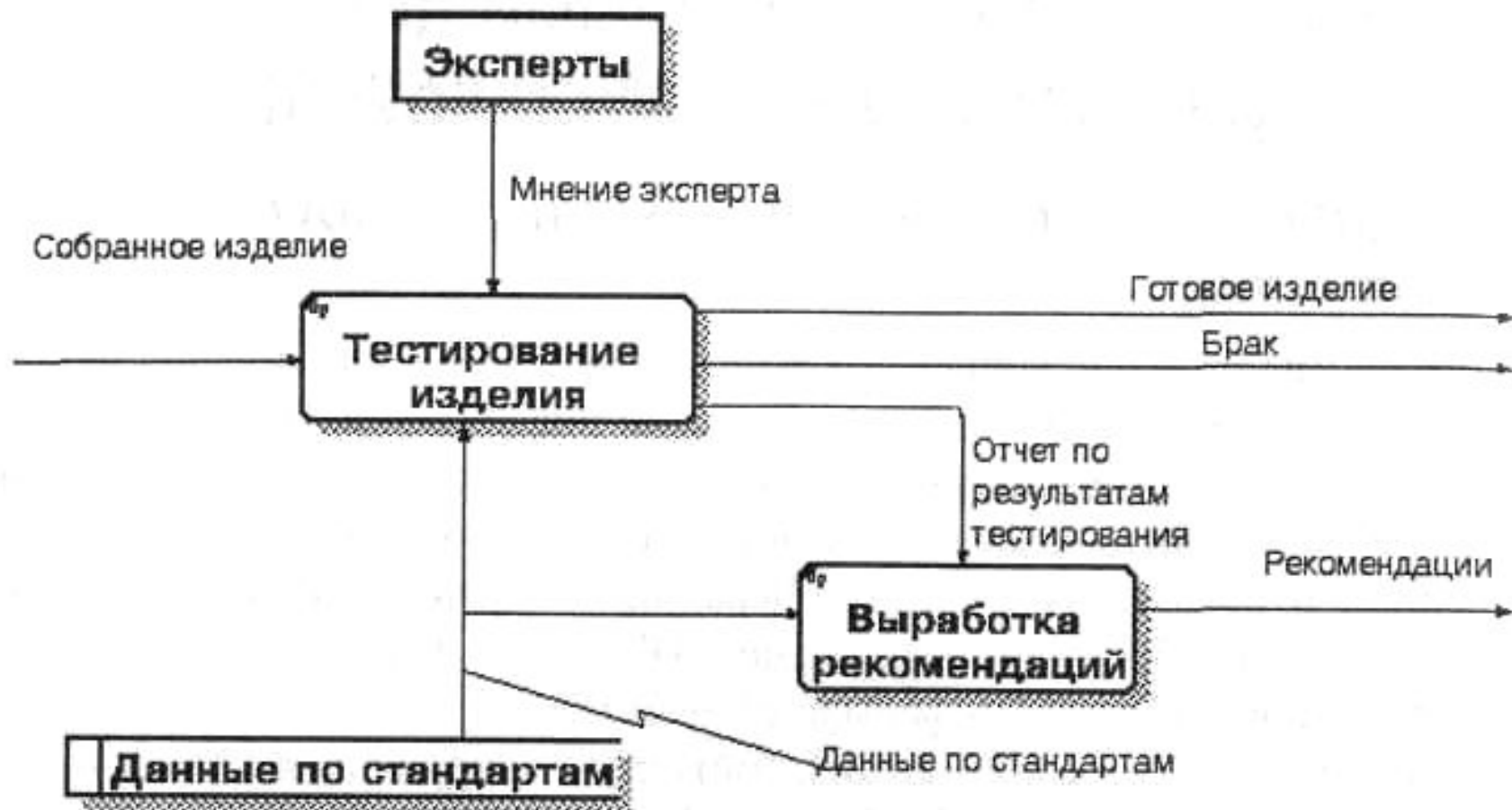
DFD описывает:

- функции обработки информации (работы);
- документы (стрелки, arrow), объекты, сотрудников или отделы, которые участвуют в обработке информации;
- внешние ссылки (external references), которые обеспечивают интерфейс с внешними объектами, находящимися за границами моделируемой системы;
- таблицы для хранения документов (хранилище данных, data store).

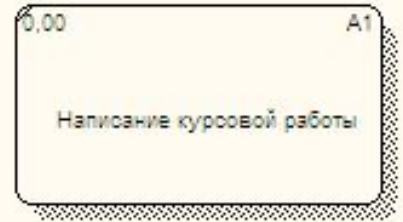
# Элементы диаграммы DFD

- Работы (activity)
- Внешние сущности (external entity)
- Потoki данных (стрелки, arrows)
- Хранилище данных (data store)
- Слияние/разветвление стрелок

# Пример DFD

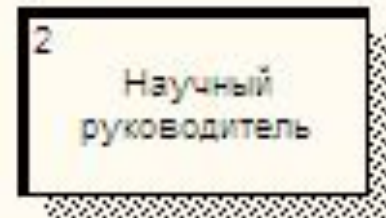


# Работы



- представляют собой функции системы, преобразующие входы в выходы.
- Хотя работы изображаются прямоугольниками со скругленными углами, смысл их совпадает со смыслом работ IDEF0 и IDEF3.
- Так же как работы IDEF3, они имеют входы и выходы, но не поддерживают управления и механизмы, как IDEF0.

# Внешние сущности

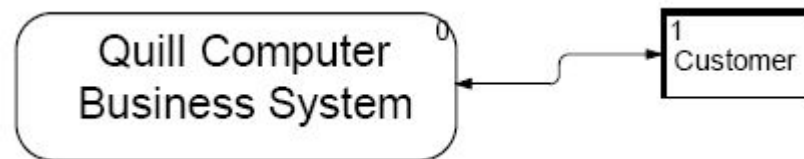


- изображают входы в систему и/или выходы из системы.
- Внешние сущности обычно располагаются по краям диаграммы.
- Одна внешняя сущность может быть использована многократно на одной или нескольких диаграммах.



# Стрелки (Потоки данных)

- Стрелки описывают движение объектов из одной части системы в другую.
- Поскольку в DFD каждая сторона работы не имеет четкого назначения, как в IDEF0, стрелки могут подходить и выходить из любой грани прямоугольника работы.
- В DFD также применяются двунаправленные стрелки для описания диалогов типа "команда-ответ" между работами, между работой и внешней сущностью и между внешними сущностями



# Хранилище данных

1	Customer data
---	---------------

- В отличие от стрелок, описывающих объекты в движении, хранилища данных изображают объекты в покое
- В материальных системах хранилища данных изображаются там, где объекты ожидают обработки, например в очереди.
- В системах обработки информации хранилища данных являются механизмом, который позволяет сохранить данные для последующих процессов

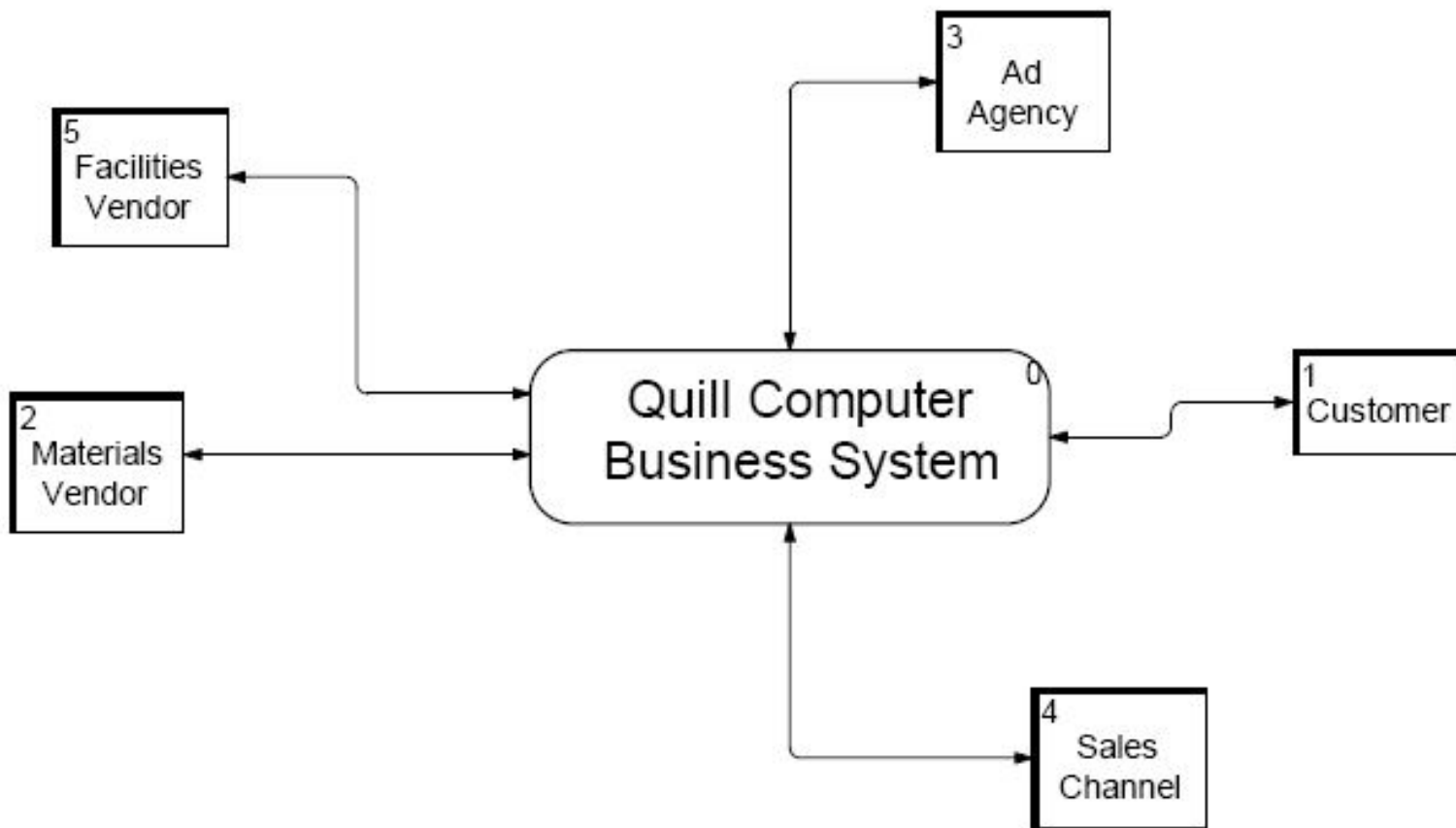
# Построение диаграмм DFD

- Подобно тому, как строится IDEF0:
  - Сначала строится *физическая модель*, отображающая текущее состояние дел.
  - Затем эта модель преобразуется в *логическую модель*, которая отображает требования к существующей системе.
  - После этого строится модель, отображающая *требования к будущей системе*.
  - И наконец, строится *физическая модель*, на основе которой должна быть построена новая система

# Альтернативным подход построения DFD (Event Partitioning)

- событийное разделение: различные диаграммы DFD выстраивают модель системы.
- Во-первых, логическая модель строится как совокупность работ и документирования того, что они (эти работы) должны делать.
- Затем модель окружения (environment model) описывает систему как объект, взаимодействующий с событиями из внешних сущностей. Модель окружения обычно содержит описание цели системы, одну контекстную диаграмму и список событий. Контекстная диаграмма содержит один прямоугольник работы, изображающий систему в целом, и внешние сущности, с которыми система взаимодействует.
- Наконец, модель поведения (behavior model) показывает, как система обрабатывает события. Эта модель состоит из одной диаграммы, в которой каждый прямоугольник изображает каждое событие из модели окружения. Хранилища могут быть добавлены для моделирования данных, которые необходимо запоминать между событиями. Потoki добавляются для связи с другими элементами, и диаграмма проверяется с точки зрения соответствия модели окружения.

# Пример контекстной диаграммы DFD



# Диаграммы потоков данных

Диаграммы потоков данных (DFD) – основное средство моделирования функциональных требований к проектируемой системе.

Требования представляются в виде иерархии *процессов*, связанных потоками данных.

Диаграммы потоков данных показывают, как каждый *процесс* преобразует свои входные данные в выходные, и выявляют отношения между этими *процессами*.

DFD-диаграммы дополняют модель IDEF0 и описывают документооборот и обработку информации.



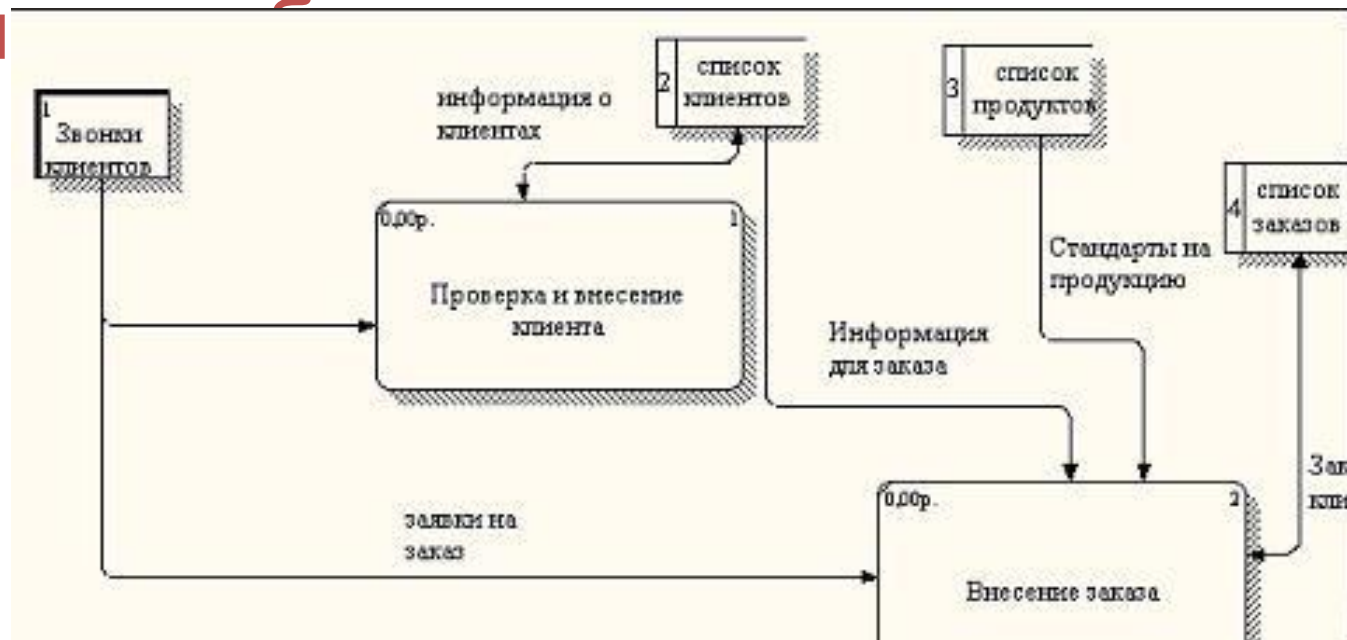
Для того чтобы дополнить модель IDEF0 диаграммой DFD, нужно в *процессе* декомпозиции в диалоге Activity Box Count «кликнуть» по радио-кнопке DFD.

- В палитре инструментов на новой диаграмме DFD появляются новые кнопки:
- *(External Reference)* – добавить в диаграмму внешнюю ссылку;
- *(Data store)* – добавить в диаграмму хранилище данных;
- *Diagram Dictionary Editor* – ссылка на другую страницу. В отличие от IDEF0 этот инструмент позволяет направить стрелку на любую диаграмму (а не только на верхний уровень).



Стрелки IDEF0 представляют собой жесткие взаимосвязи, стрелки DFD показывают, как объекты (включая данные) двигаются от одной работы к другой.

DFD более похожи на физические характеристики системы — движение объектов, хранение объектов, поставка и распростран



В DFD работы (процессы) – функции системы, преобразующие входы в выходы (мысл их совпадает со смыслом работ IDEF0 и IDEF3).

Они имеют входы и выходы, но не поддерживают управление и механизмы.

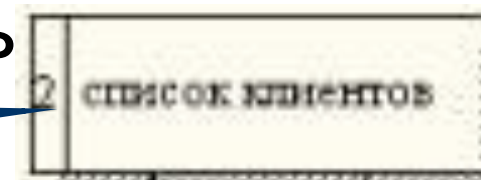
Внешние сущности – входы в систему и/или выходы из системы. *Внешние сущности* изображаются в виде прямоугольника с тенью и обычно располагаются по краям диаграммы.

Одна *внешняя сущность* может быть использована многократно на одной или нескольких диаграммах (чтобы не повторять

Потоки работ – стрелки, которые описывают движение объектов из одной части системы в другую.

Стрелки могут подходить и выходить из любой грани прямоугольника работы. В DFD также применяются двунаправленные стрелки для описания диалогов типа «команда-ответ» между работами, между работой и внешней сущностью и между внешними сущностями.

Хранилища данных изображают объекты в покое (в отличие от стрелок, описывающих движение объектов в динамике).  
Хранилище данных



В материальных системах *хранилища данных* изображаются там, где объекты ожидают обработки, например в *очереди*.

В системах обработки информации *хранилища данных* являются механизмом, который позволяет сохранить данные для последующих *процессов*.

В DFD стрелки могут сливаться и разветвляться, что позволяет описать декомпозицию стрелок.

Каждый новый сегмент сливающейся или разветвляющейся стрелки может иметь собственное имя.

В DFD номер каждой работы может включать префикс, номер родительской работы (A) и номер объекта.

Номер объекта — это уникальный номер работы на диаграмме. Например, работа может иметь номер A.12.4.

Уникальный номер имеют *хранилища данных* и *внешние сущности* независимо от их расположения на диаграмме.

Каждое *хранилище данных* имеет префикс D и уникальный номер, например D5.

Каждая *внешняя сущность* имеет префикс E и уникальный номер, например E5.

# Вернемся к нашему примеру

- Перед приемом в больницу проводится встреча пациента и врача. Врач сообщает в отдел приема пациентов об ожидаемом приеме больного и передает данные о нем. Если пациент принят в больницу впервые, то до его приема в больницу его регистрируют - присваивают регистрационный номер и записывают его данные (фамилия, имя и отчество, адрес и дата рождения).
- Спустя некоторое время врач оформляет в отделе приема пациентов прием больного. При этом определяется порядковый номер приема и запоминаются данные приема пациента. После этого отдел приема посылает сообщение врачу для подтверждения приема больного. В это сообщение включаются регистрационный номер пациента и его фамилия, порядковый номер приема, дата начала лечения и номер палаты.
- В день приема пациент сообщает в отдел приема о своем прибытии и передает данные о себе (или изменения в данных). Отдел приема проверяет и при необходимости корректирует данные о пациенте. Если пациент не помнит свой регистрационный номер, то выполняется соответствующий запрос. После регистрации пациент получает регистрационную карту, содержащую ФИО пациента, адрес, дату рождения, номер телефона, группу крови, название страховой компании и номер страховки.
- Во время пребывания в больнице пациент может лечиться у нескольких врачей; каждый врач назначает один или более курсов лечения, но каждый курс лечения назначается только одним врачом. Данные о курсах лечения передаются в медицинский секретариат, который занимается координацией лечения пациентов, регистрируются и хранятся там. Данные включают номер врача, номер пациента, порядковый номер приема, название курса лечения, дату назначения, время и примечания.
- При необходимости врач запрашивает в медицинском секретариате историю болезни пациента, содержащую данные о курсах лечения, полученных пациентом.

Главная цель контекстной диаграммы потоков данных - продемонстрировать, как каждый процесс преобразует входные данные в выходные, а также выявить отношения между этими процессами.

Диаграммы верхних уровней иерархии (контекстные диаграммы) определяют основные процессы или подсистемы с внешними входами и выходами. Как видно из рис.следующего, происходит обмен данными с внешними сущностями и хранилищами данных.

Отметьте, что стрелки имеют разный цвет. Это для того, чтобы после декомпозиции диаграммы на диаграмме нижнего уровня было понятно, с какой внешней сущностью происходит обмен потоками данных.

Создайте на контекстной диаграмме процесс, выбрав на панели инструментов инструмент Процесс, представленный прямоугольником с закругленными углами.





# Контрольные вопросы

- Для чего служит DFD - диаграмма?
- В чем отличие DFD - диаграммы от IDFE0?
- Какой инструмент используется для построения внешних сущностей?
- Каким инструментом можно построить Хранилище данных?
- Какие графические элементы используются для обозначения на диаграмме Работы, Поточков данных, Хранилищ данных?

# *Тестовое задание*






- **DFD - это:**
- диаграмма бизнес - процесса;
- диаграмма потока данных;
- контекстная диаграмма;
- функциональная диаграмма;
- диаграмма сущность - связь.
- 
- **Какие объекты описываются на диаграмме потоков данных:**
- функции (работы);
- хранилища данных;
- внешние объекты;
- внешние ссылки;
- все ответы правильные.
-

# Тестовое задание

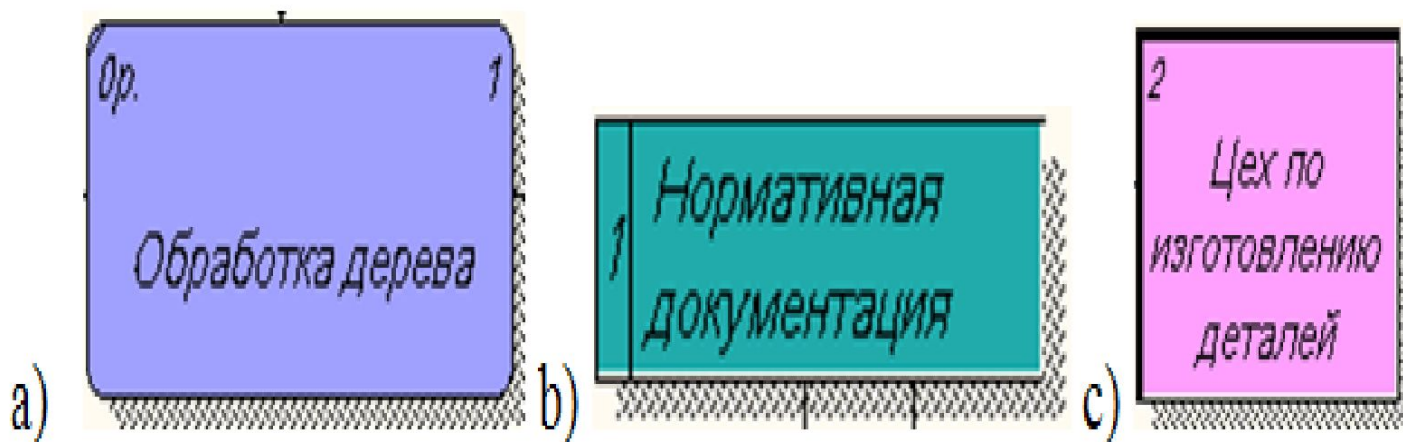
**Что показывают диаграммы потоков данных?**

- внешние источники и приемники данных;
- потоки данных;
- хранилища (накопители) данных;
- все ответы правильные;
- правильного ответа нет.

**Какой инструмент нужно выбрать для создания диаграммы DFD?**

- a)  ;
- b)  ;
- c)  ;
- d)  ;
- e)  .

Укажите номер рисунка, обозначающий на диаграмме Хранилище данных.



# Домашнее задание

**Тема урока:** Метод описания IDEF3

**Ключевое слово:** IDEF3

**План урока:**

1. Работы;
2. Связи.

Интернет ссылки: [Intuit.ru](http://Intuit.ru)