DFD

data flow diagrams

Моделирование

- Эффективный способ документировать систему в процессе анализа - создать **модель системы**.
- Для этой цели используются различные методики: диаграммы потоков данных (*DFD - data flow diagrams, IDEF0*) для описания документооборота в компьютерных системах и в организации, блок-схемы процедур, моделирование данных (*ERD- entity relation diagrams, IDEF1X*), прототипирование и др.

Модели деятельности организации

- Модели создаются в двух видах:
  - ***модель "как есть"***("as-is")- отражает существующие в организации бизнес-процессы;
  - ***модель "как должно быть"***("to-be") - отражает необходимые изменения бизнес-процессов с учетом внедрения ИС.

Модели анализа

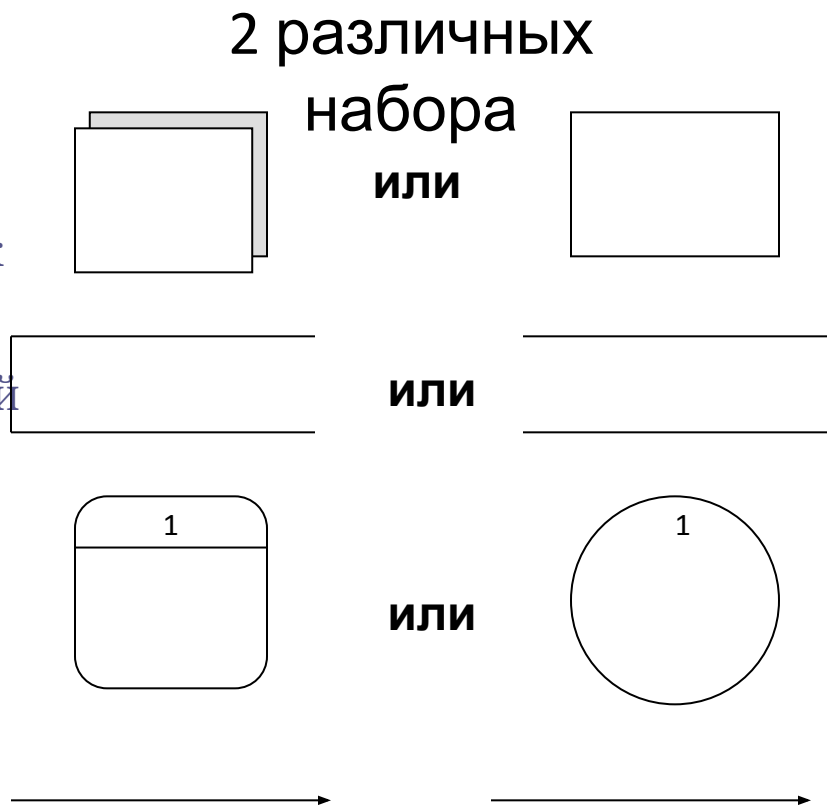
- В соответствии со структурой ИС, соответствующие модели могут включать:
  - DATA – Модель данных в одну страницу очень полезна для установления номенклатуры, правил и направления развития бизнеса.
  - PROCESSES – Применяется одно-двух страничная диаграмма функциональной декомпозиции процессов.
  - INTERFACES – Контекстная диаграмма в одну страницу полезна для прояснения входов и выходов системы с другими системами, организациями и подразделениями.
  - GEOGRAPHY – В одну страницу модель распределения адекватно идентифицирующая текущее распределение операций, относящихся к системе.

Data Flow Diagrams (DFD)

- DFD используется для графического представления системы
- Специальные соглашения используются с тем чтобы смысл DFD был согласован между системными аналитиками
- DFD имеет дело только бизнес-процессами и потоками данных
- Для полного представления системы требуется иерархия диаграмм DFDs

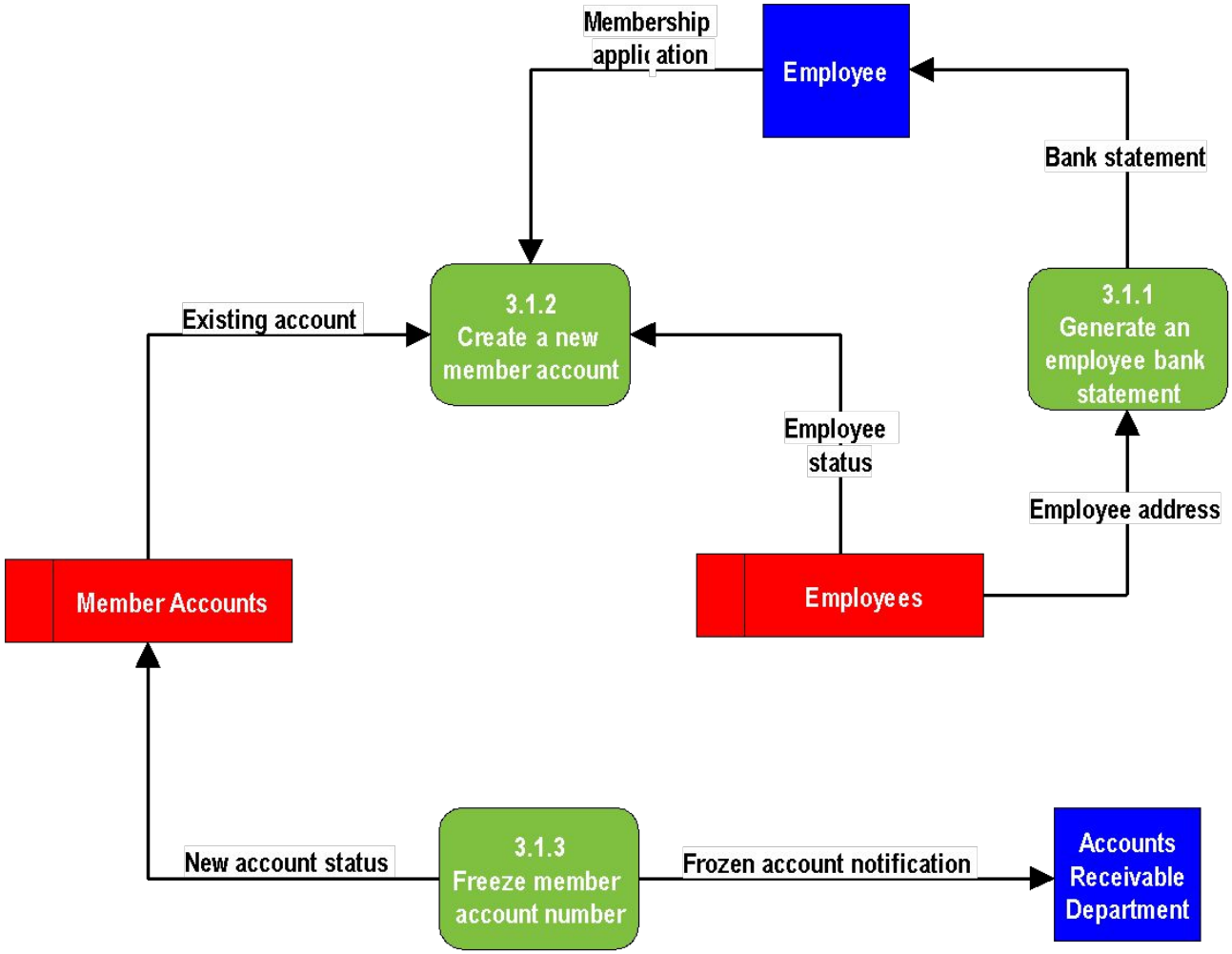
Соглашение о символах DFD

- Используется 4 символа
  - Внешняя сущность
    - Квадрат с тенью или прямоугольник
  - Накопитель данных
    - Прямоугольник с открытой стороной или параллельные линии
  - Процесс
    - Закругленный квадрат или круг
      - **Должен нумероваться**
  - Поток данных
    - Стрелка
  - Все символы подписываются

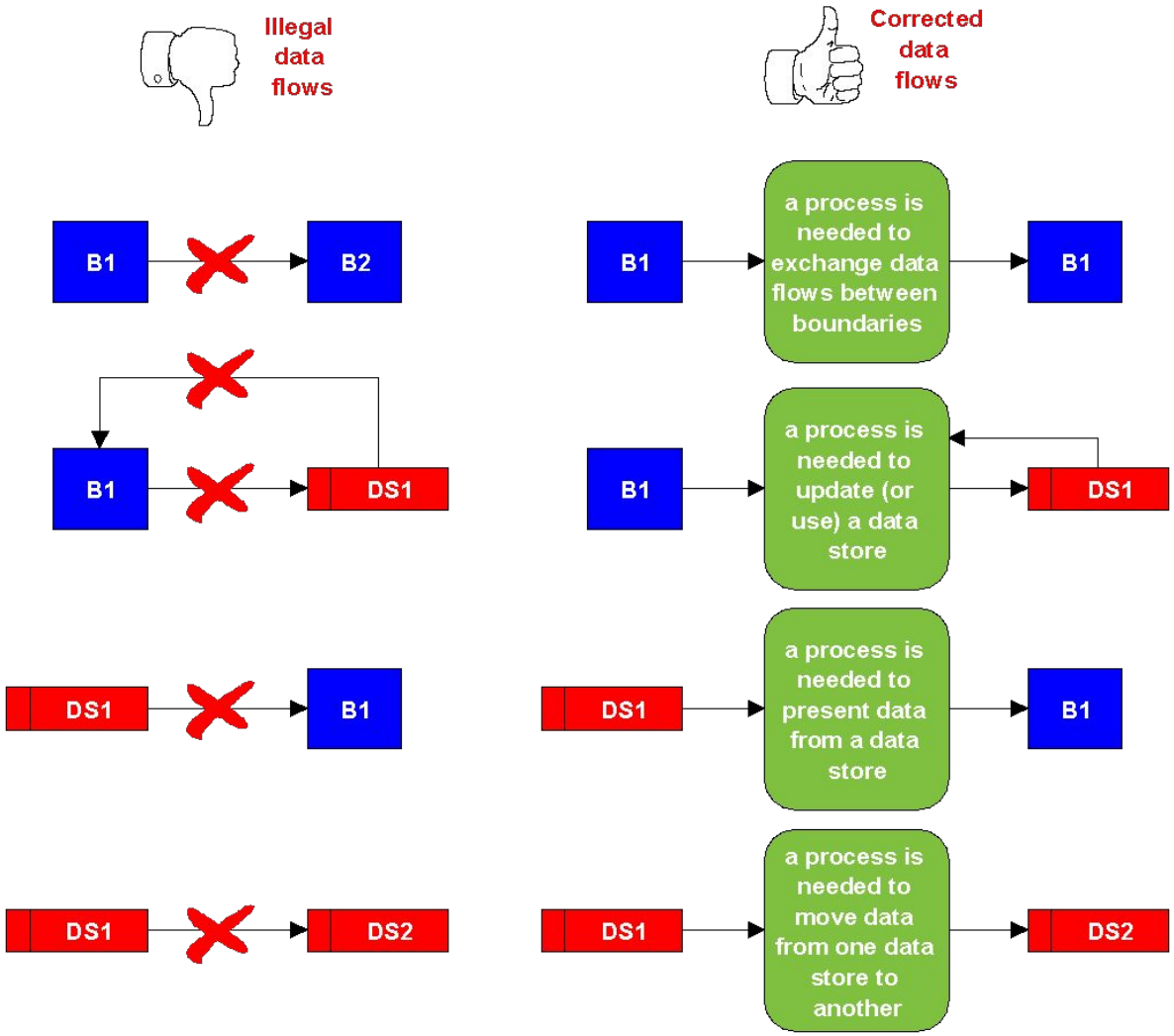


Выберите один набор и следуйте ему!

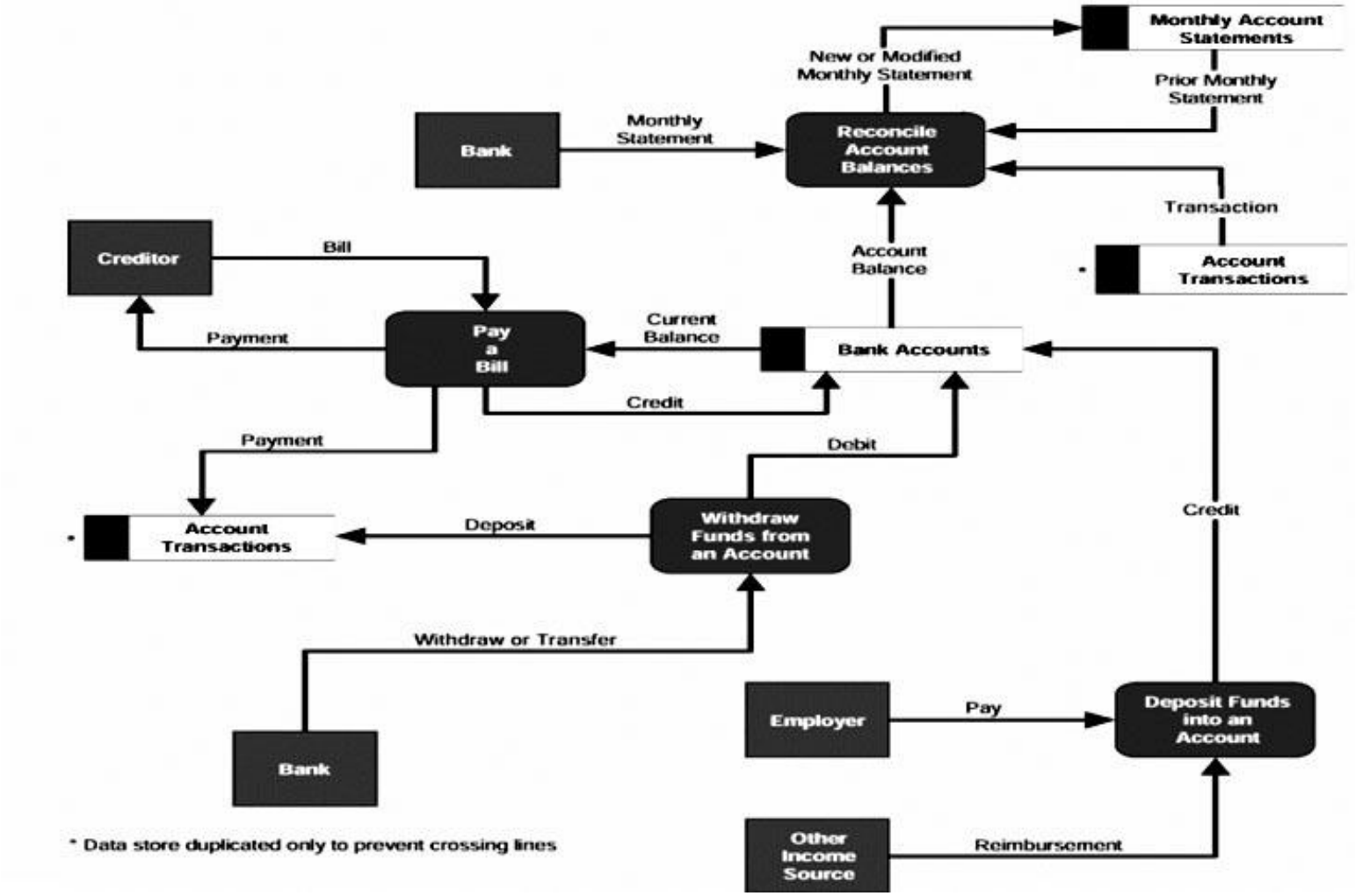
Соглашение о символах DFD



Соглашение о символах DFD

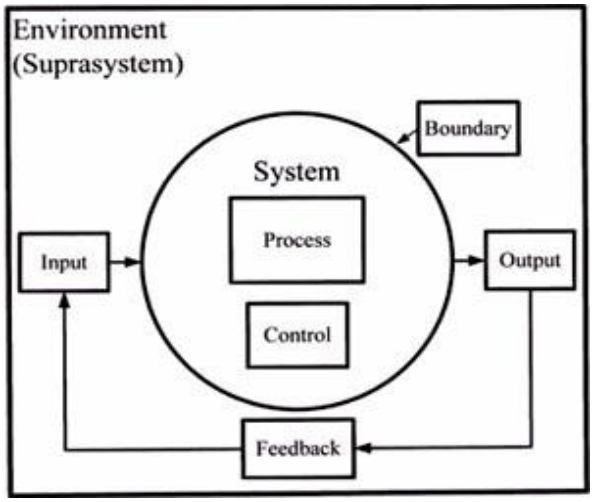


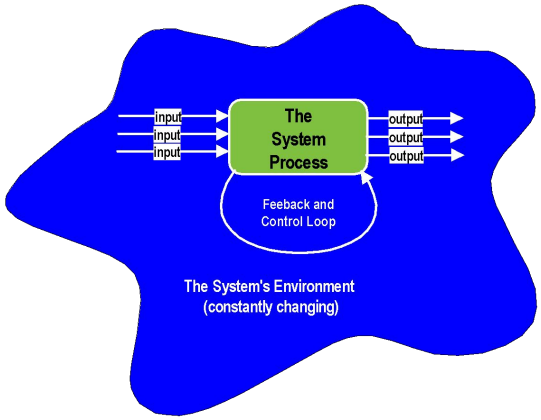
Пример диаграммы



Концепция процессов

- Система представляет собой процесс.
- Простейшая модель системы основана на входе, выходе и самой системе – представляющей процесс.

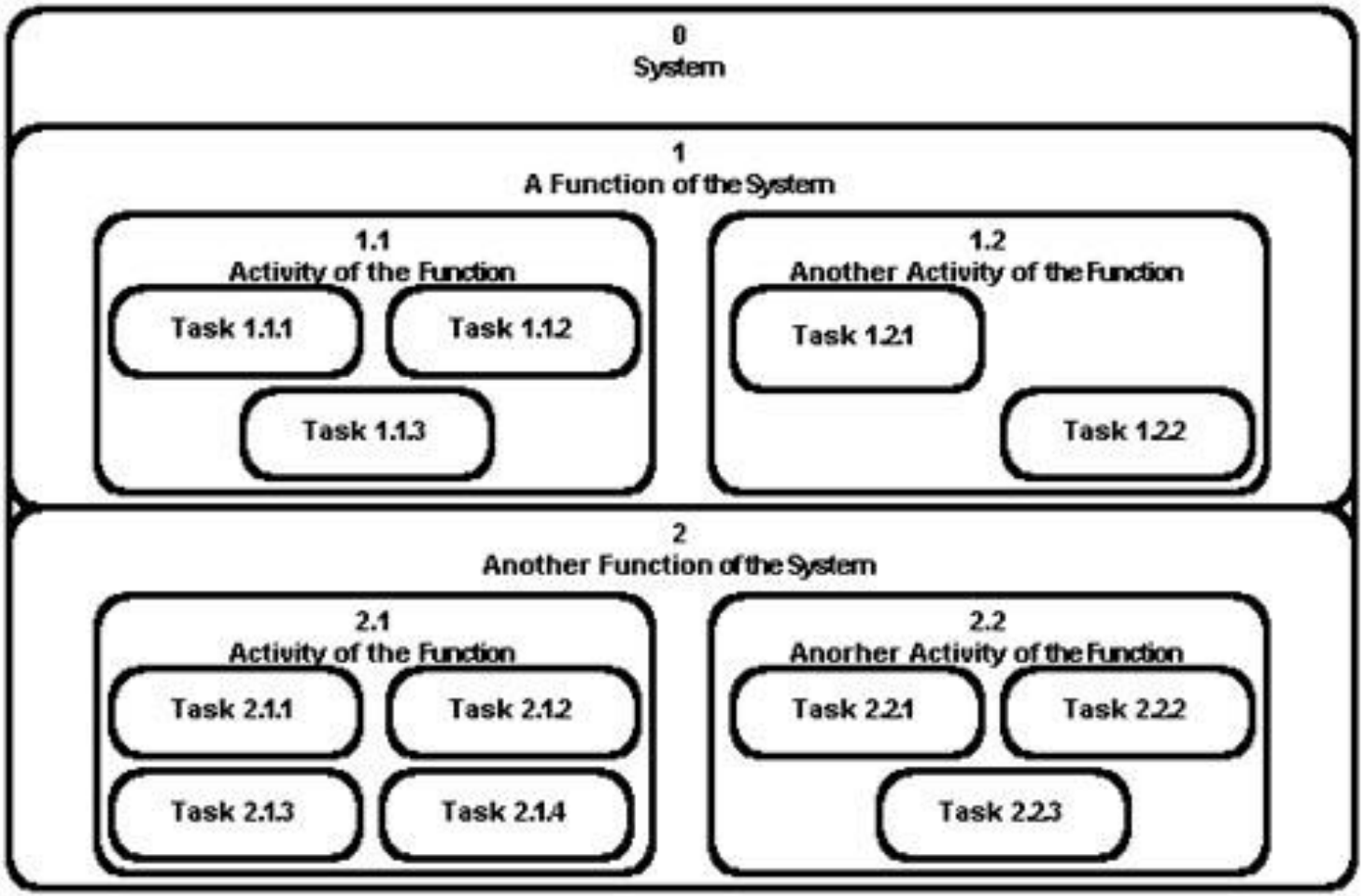




Декомпозиция процессов

- Декомпозиция позволяет разбить систему на подсистемы для улучшения обсуждения, анализа и проектирования.
- Системы разбиваются на подсистемы, которые в свою очередь, также декомпозируются на подсистемы, до тех пор пока не получается управляемый набор системы.

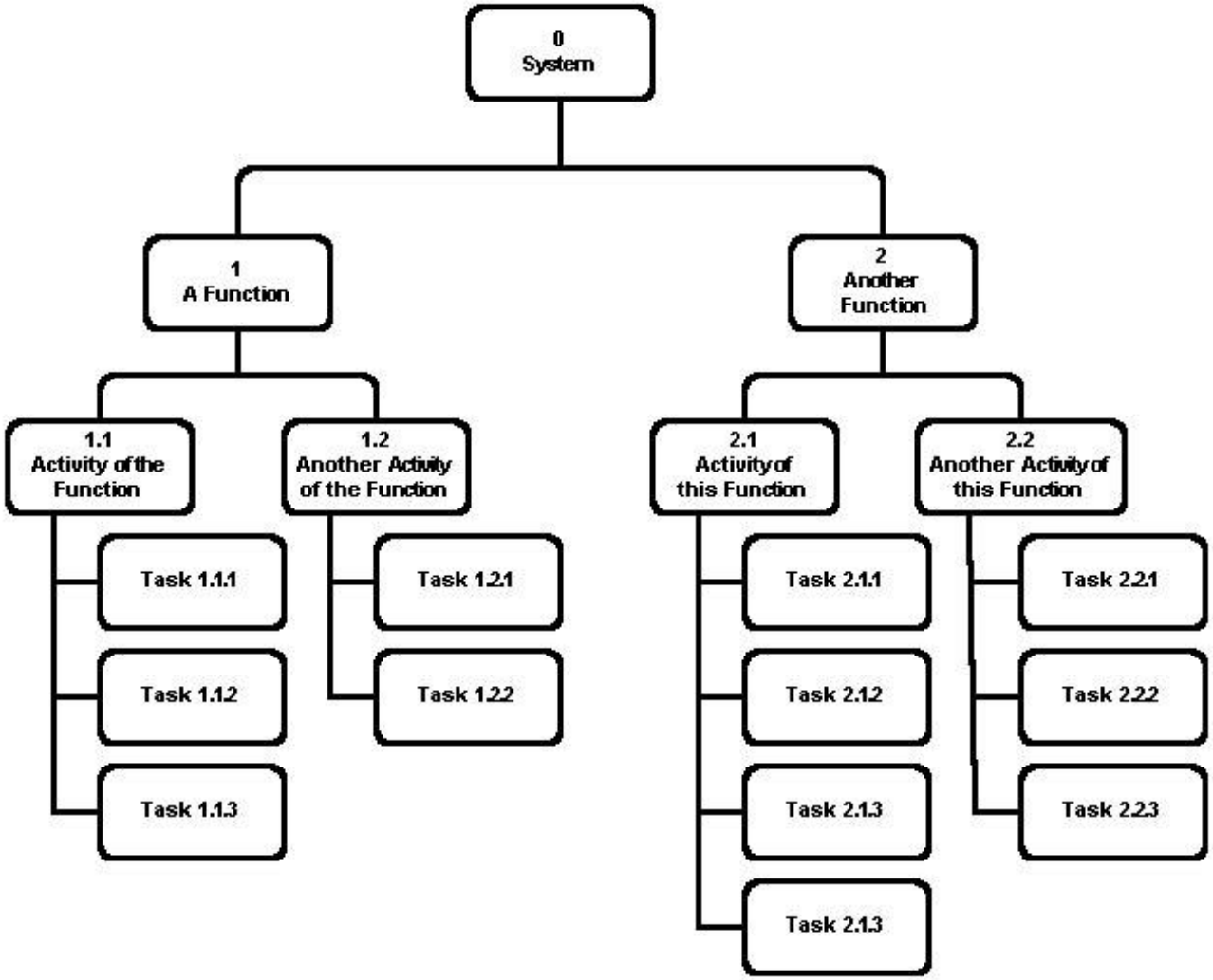
Декомпозиция процессов



Правила диаграммы декомпозиции

- Каждый процесс представляет собой либо родительский, либо порожденный процесс. Либо и тот и другой.
- Родитель должен иметь два и более потомков – один потомок не имеет смысла поскольку он не может представлять дополнительных деталей о системе.
- Соединения в диаграмме декомпозиции не содержат направленных стрелок, поскольку диаграмма представляет структуру, а не потоки.
- Связи не именуются и все имеют одинаковые имена –СОСТОИТ ИЗ - поскольку сумма порожденных процессов эквивалентна родительскому.

Правила диаграммы декомпозиции



Правила диаграммы декомпозиции

- Логический процесс – работа или действие, которые должны быть выполнены. Каждый логический процесс рассматривается как один или более физических процессов, которые могут включать:
  - Работу людей;
  - Работу, выполняемую роботами или машинами;
  - Работу, выполняемую ПО.
- Существует три типа логических процесса:
  - функция,
  - транзакция и
  - элементарный процесс.

Функция

- **Функция** – набор связанных текущих операций в бизнесе.
  - Функции не имеют начала и конца и постоянно выполняются в случае необходимости.
  - Каждая из функций состоит из большого количества отдельных *процессов* для обеспечения конкретных задач и действий.
  - Функции служат для объединения логически связанных действий или задач.
  - Функция обозначается существительным, отражающим ее сущность.

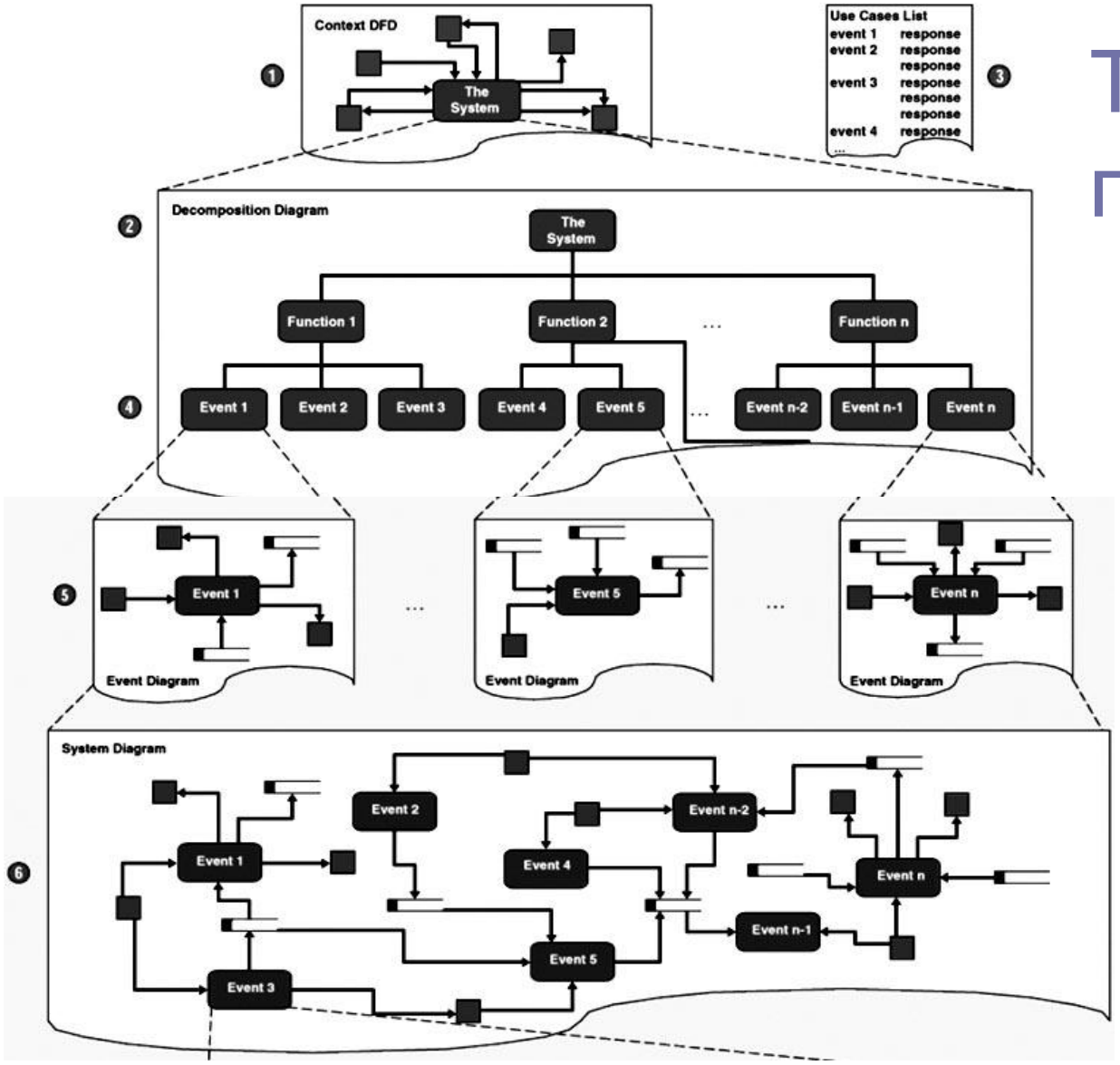
Элементарные процессы

- Функция может быть в дальнейшем декомпозирована на элементарные процессы, которые детально отражают как система должна реагировать на событие.
- Элементарные процессы отдельные, детальные задачи или действия , требуемые для выполнения ответа на событие. Это нижний уровень изображения модели системы. Элементарные процессы следует именовать глаголом и объектом, которые описывает выполняемую работу. Следует оставлять только те логические процессы, которые:
  - **Выполняют вычисления(Например, вычисляют среднее)**
  - **Принимают решения(определяют пригодность заказанной продукции)**
  - **Сортируют, выбирают или обобщают данные(определяют просроченный заказ)**
  - **Преобразуют данные в информацию (генерируют отчет или отвечают на запрос)**
  - **Запускают другие процедуры (включают печь или задают команду роботу)**
  - **Используют сохраненные данные (создают, считывают, модифицируют или удаляют запись)**

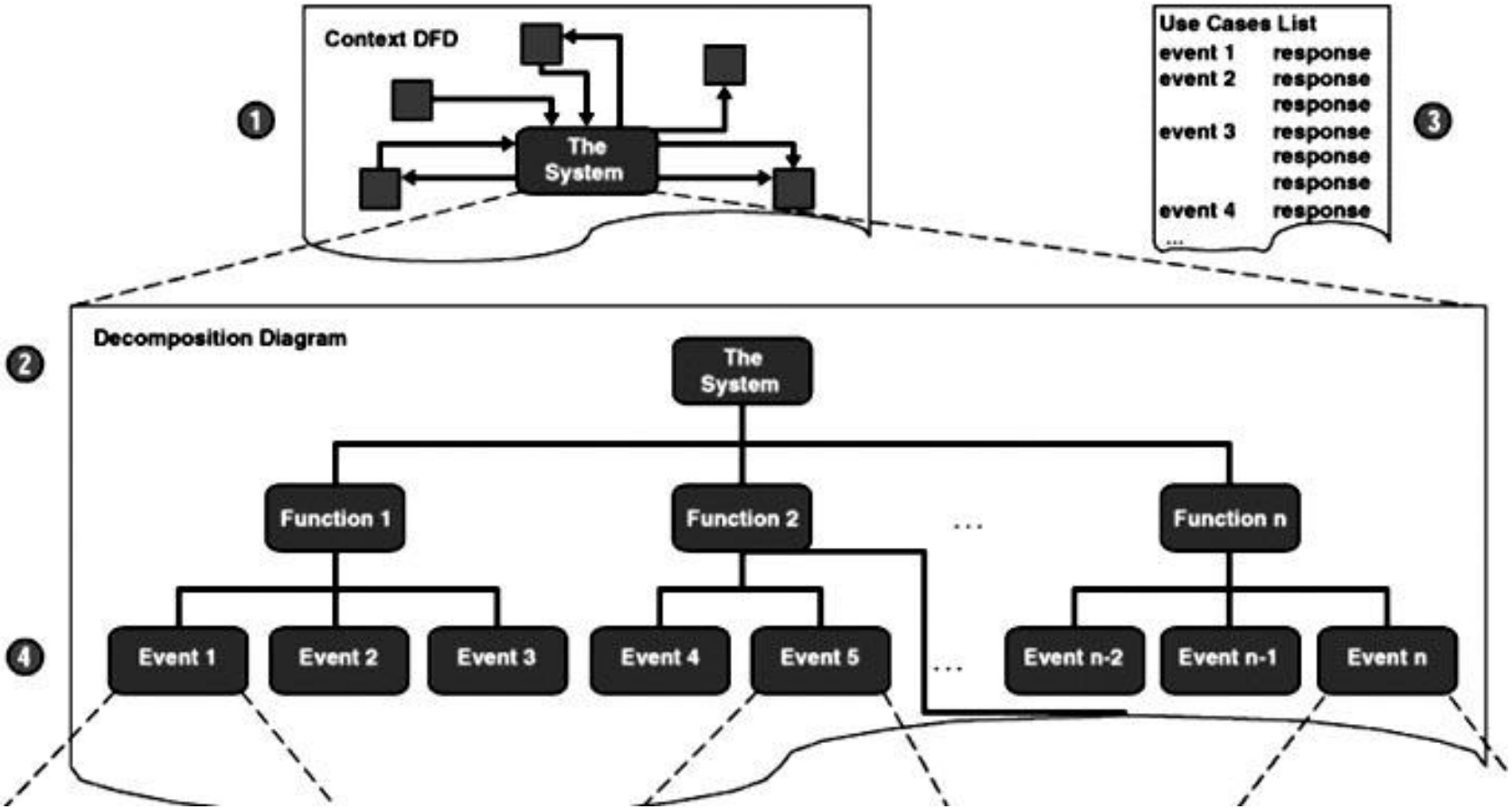
Стратегия моделирования процессов

1. Формируется контекстная диаграмма для установления масштаба проекта.
2. Строится функциональная диаграмма для представления системы логическими подсистемами или функциями
3. Составляется перечень транзакций (событий) для определения и подтверждения событий бизнеса, на которые система должна обеспечивать ответ.
4. Один процесс – обработчик событий - добавляется в диаграмму декомпозиции для каждой транзакции
5. Диаграмма транзакций строится для каждого события.
6. Диаграмма системы строится с помощью объединения диаграмм транзакций
7. Диаграмма элементарных функций строится для каждой транзакции. Эти ДПД показывают все элементарные процессы, хранилища данных и потоки данных для одной транзакции.

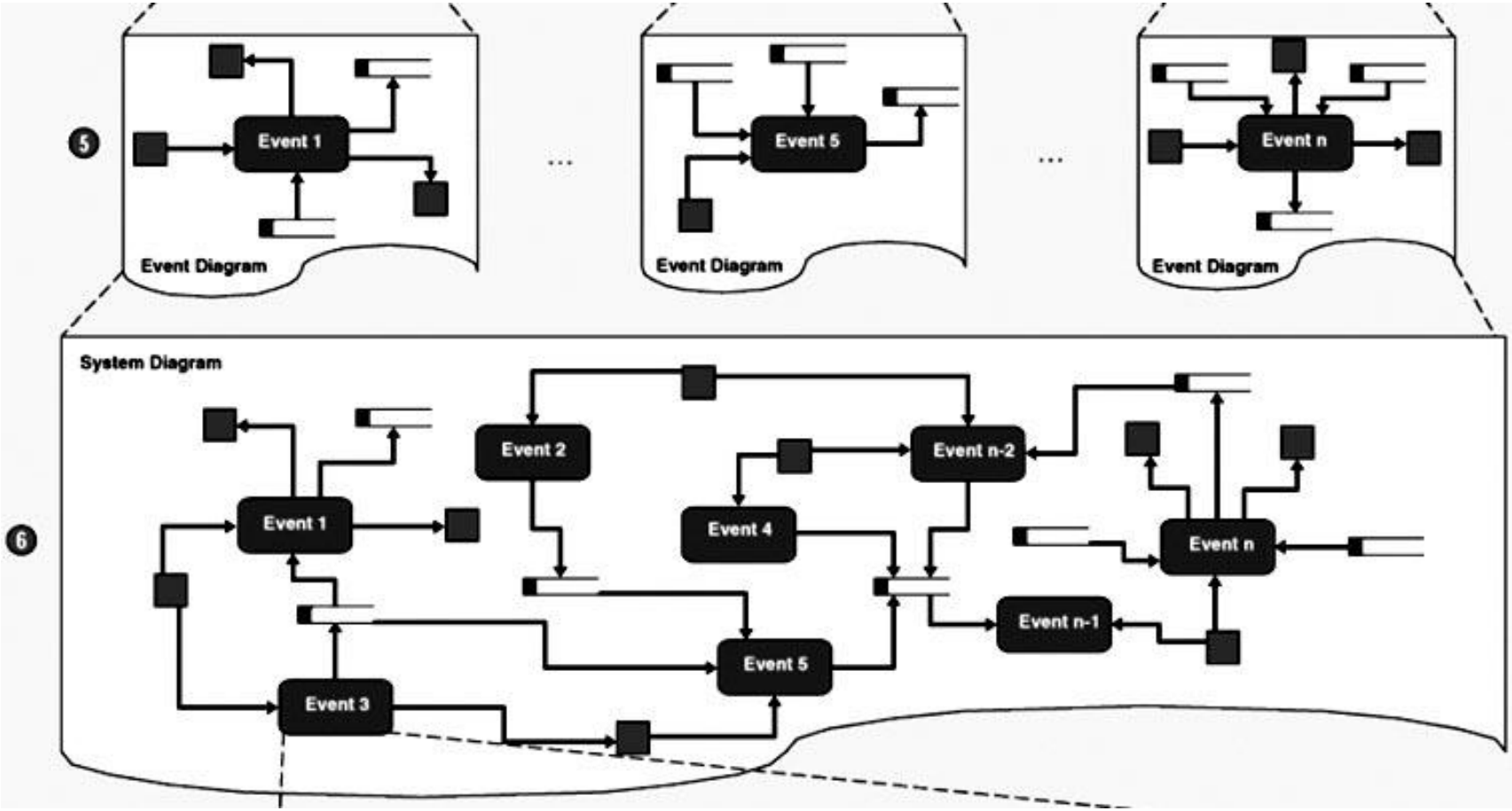
Технология построения DFD



Технология построения DFD

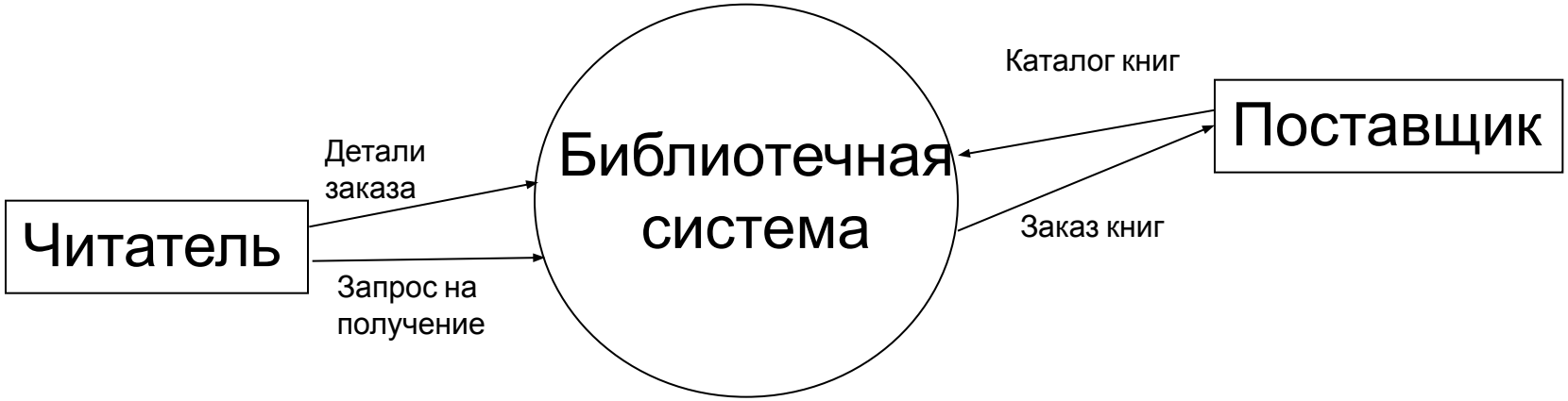


Технология построения DFD

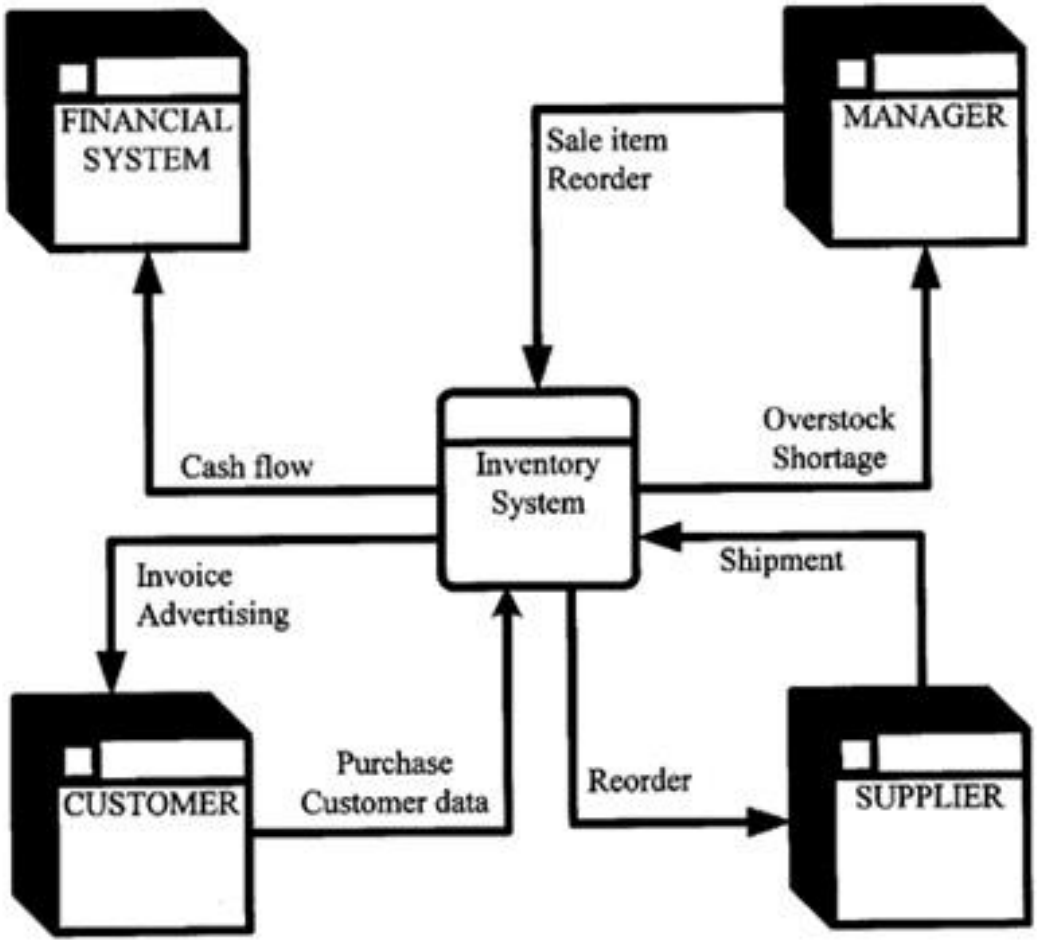


1. Создание контекстной диаграммы

- Для создания DFD и модели системы используется проектирование сверху вниз
- Начинается с уровня контекстной диаграммы, которая определяет масштаб и границы системы
  - Это означает, что вся система представляется как процесс
  - Видимы только внешние сущности с их потоками данных и внешние хранилища данных
  - Должны быть известны границы анализируемой системы



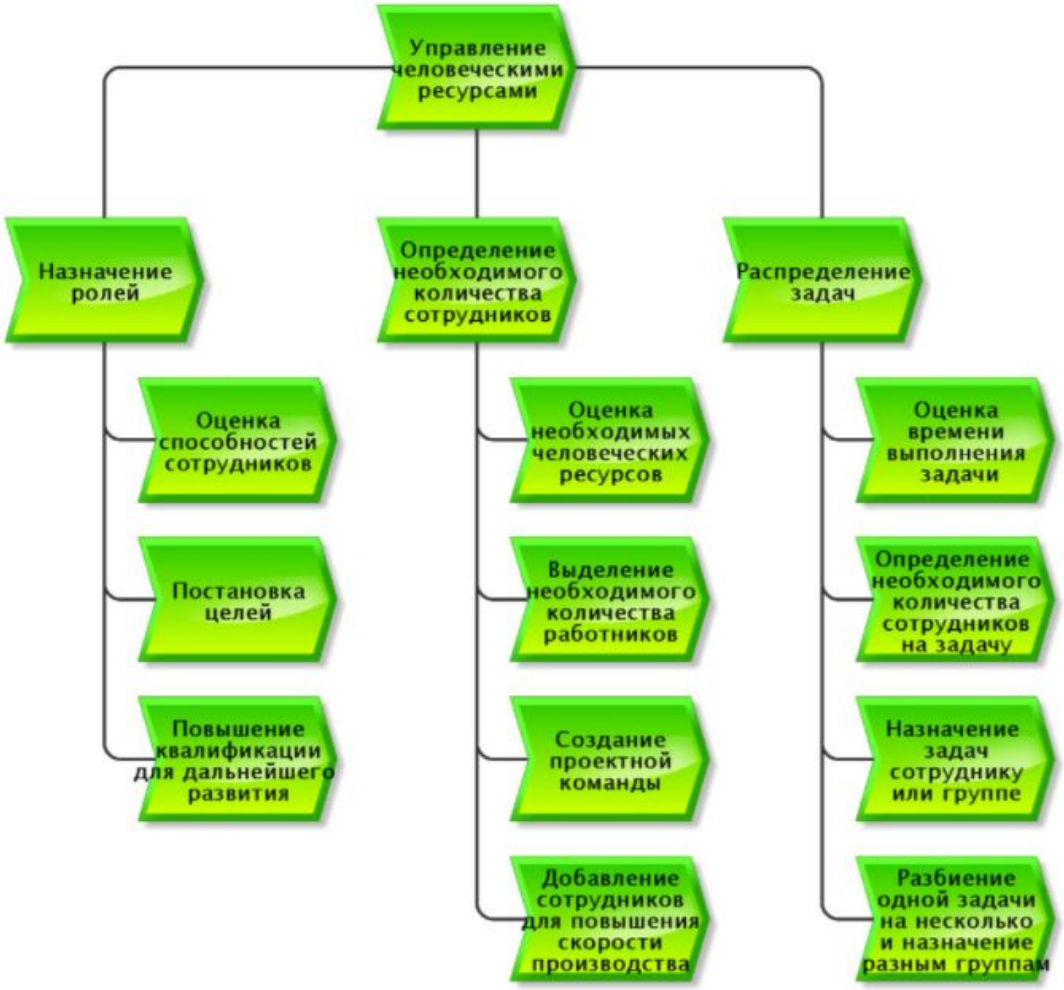
Пример контекстной диаграммы



2. Функциональная декомпозиция

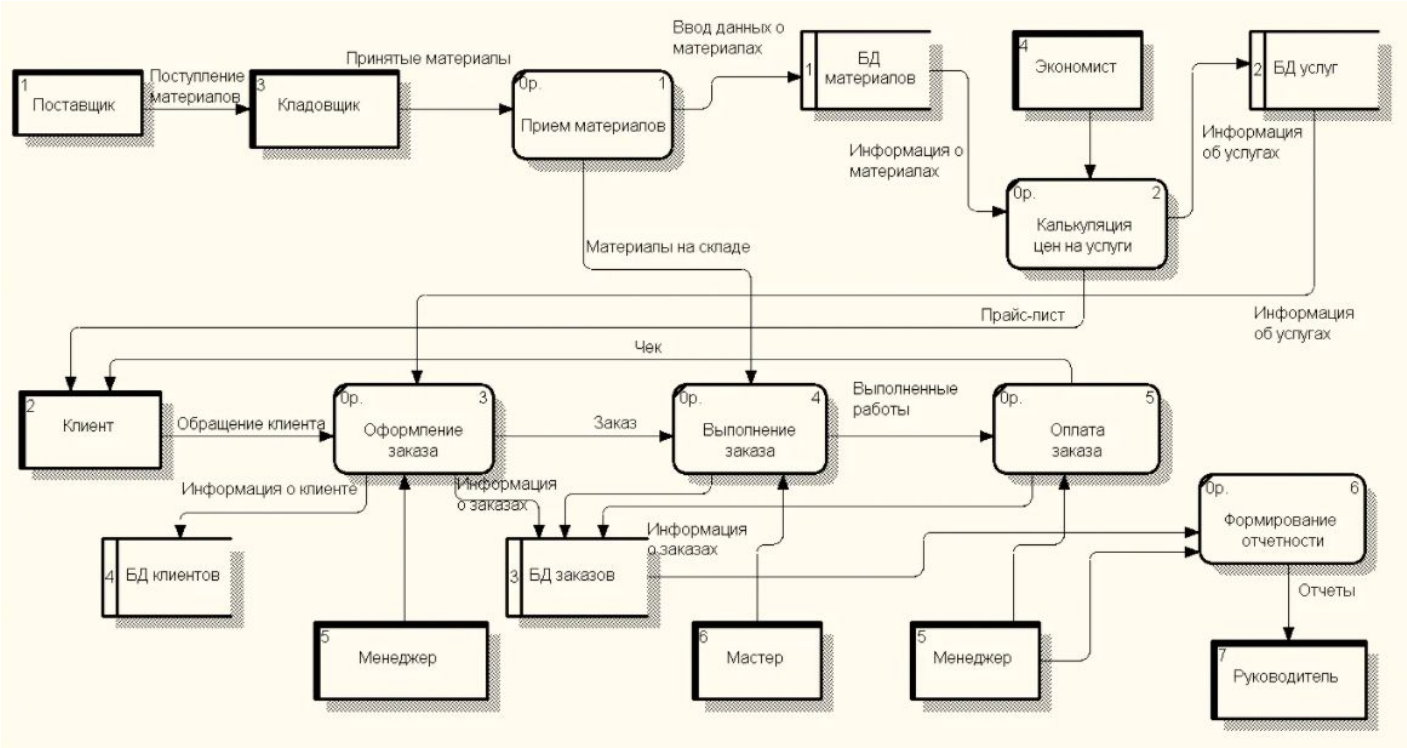
- Корневой процесс представляет всю систему.
- Первоначально система делится на подсистемы и/или функции.
- Подсистемы подразделяются на основе выполнения операций и формирования отчетов.

Функциональная декомпозиция (пример)



Создание DFD

- Разбиение процессов продолжается до тех пор пока не будет достигнут **элементарный уровень**



Критерии элементарного уровня

- наличие у процесса относительно небольшого количества входных и выходных потоков данных (1,2,3 потока);
- возможность описания преобразования данных процессом в виде последовательного алгоритма;
- выполнение процессом единственной логической функции преобразования входной информации в выходную;
- возможность описания логики процесса при помощи миниспецификации небольшого объема (не более 20-30 строк).

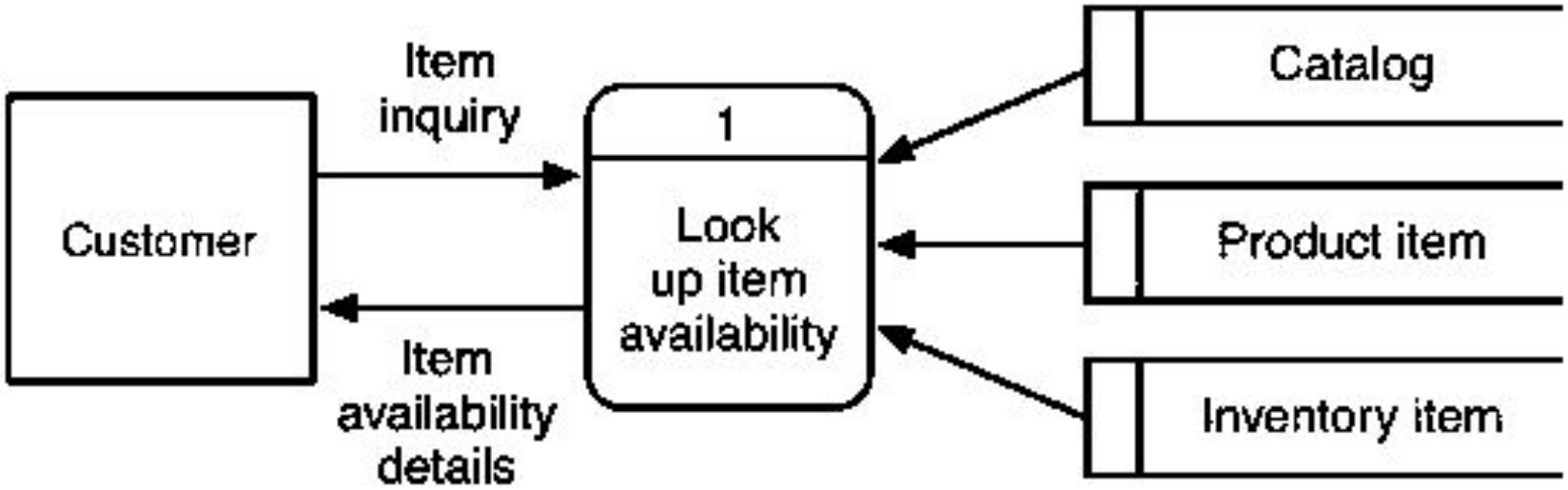
Правила создания DFDs

- **Необходимо выполнять следующие правила для структуры DFD**
  - Диаграмма должна быть сбалансирована.
    - Это означает, что все потоки данных верхнего уровня диаграммы должны появляться на декомпозированном уровне в качестве входного или выходного потока по крайней мере для одного процесса
  - Каждый процесс должен иметь по крайней мере один входной или один выходной поток данных.
    - В противном случае это не действительный процесс.

Правила создания DFD

- Каждый процесс должен быть пронумерован.
  - Не подразумевается никакого порядка нумерации.
- Каждый символ должен быть соответствующим образом поименован.
- Только внешние сущности и накопители данных могут быть
  - **Стоками - только вход данных** или
  - **Источниками – только выход данных**

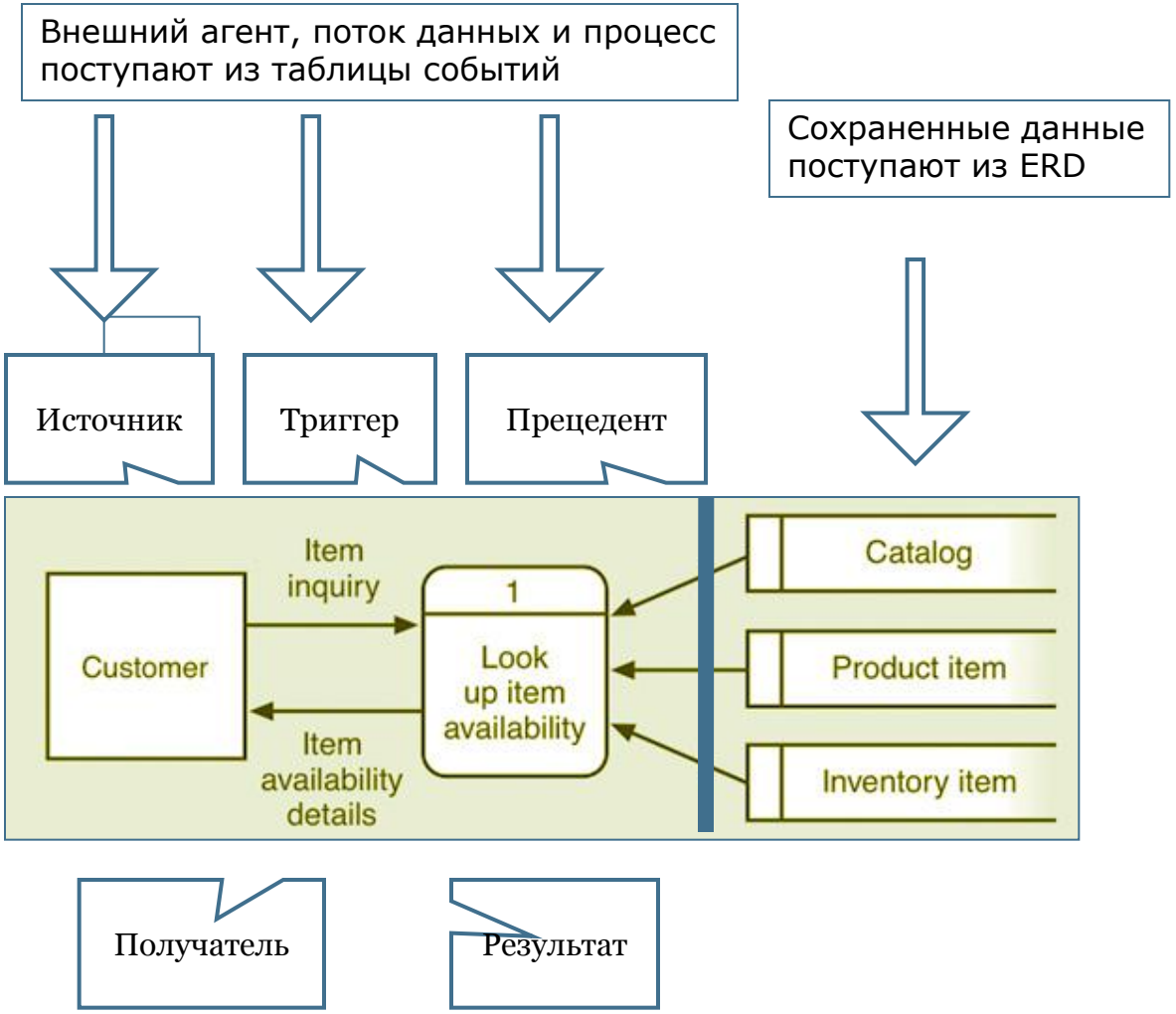
Фрагмент DFD, показывающий прецедент *Просмотр доступных товаров*



Формирование перечня транзакций

- Цель этапа – определить на какие события бизнеса система должна откликаться и какая реакция является соответствующей.
- Существуют три типа транзакций.
  - Внешние события, которые инициируются внешними агентами и отображаемыми как входные потоки данных. (Заказчик запросил баланс).
  - Временные транзакции, которые инициируют процесс на основе времени или временных событий. Временные транзакции иллюстрируются как входные управляющие потоки. (По концу месяца, при выполнении условий, …)
  - Транзакции состояния инициируются на основе изменения состояний или условий в системе.

DFD интегрирует таблицу событий и ERD



Информация о каждом событии хранится в таблице событий: каталог информации о каждом прецеденте

The event that causes the system to do something.

Source: For an external event, the external agent, or actor, is the source of the data entering the system.

Response: What output (if any) is produced by the system?

Event	Trigger	Source	Use case	Response	Destination
Customer wants to check item availability	Item inquiry	Customer	Look up item availability	Item availability details	Customer

Trigger: How does the system know the event occurred? For external events, this is data entering the system. For temporal events, it is a definition of the point in time that triggers the system processing.

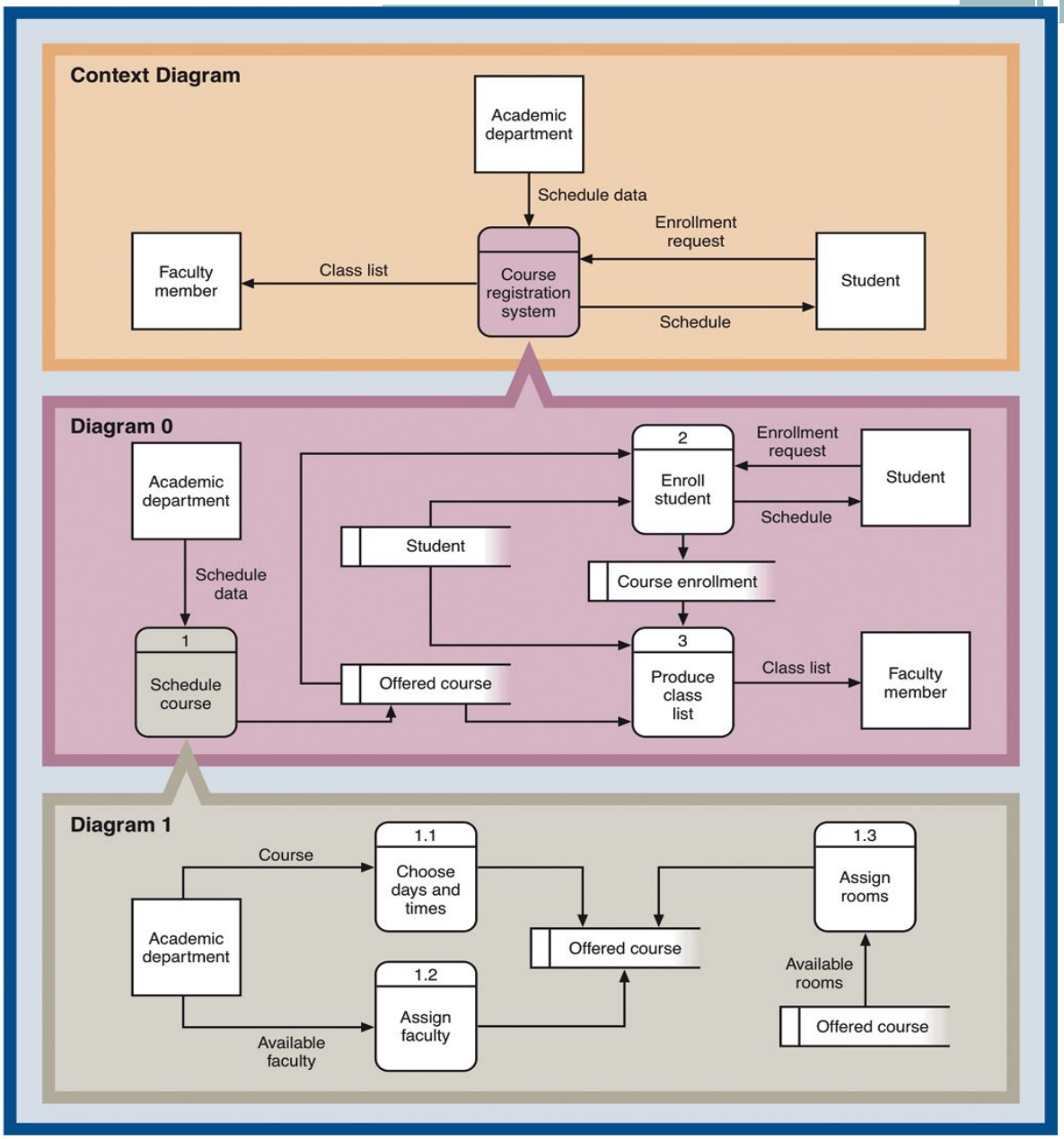
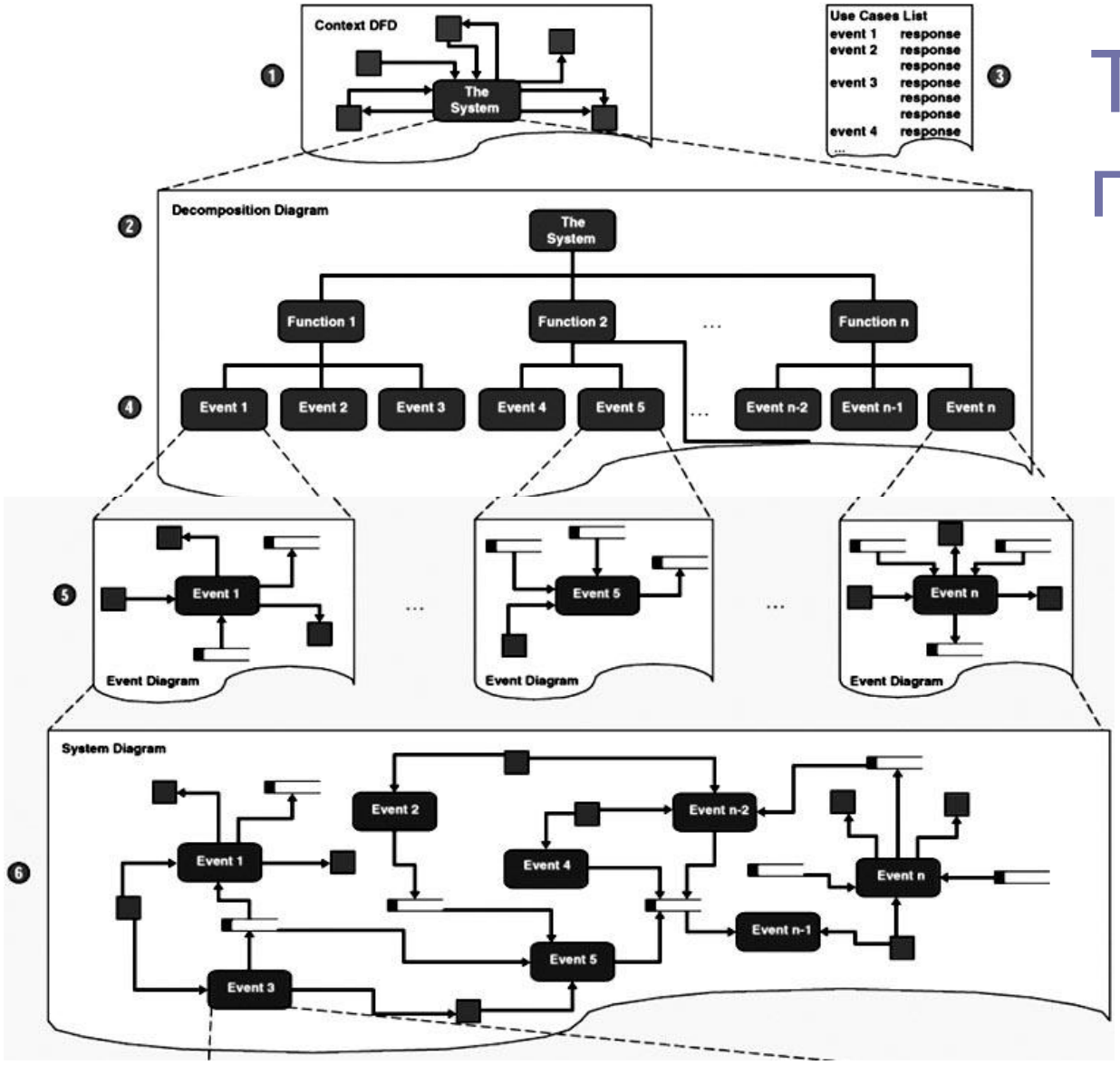
Use case: What does the system do when the event occurs? The use case is what is important to define for functional requirements.

Destination: What external agent gets the output produced?

DFD и уровни абстракции

- DFD – техника моделирования, которая разбивает систему на некоторое количество более подробных моделей
- DFD может отражать либо высокоуровневую обработку (более общий взгляд на систему), либо обработку на низком уровне (более подробный взгляд на процессы)
- Эти различные взгляды на систему (высокоуровневый и низкоуровневый) создают уровни абстракции
- DFD декомпозируется на дополнительные диаграммы для обеспечения многоуровневой структуры моделей

Многоуровневая абстракция DFD Системы регистрации на дисциплины



Технология построения DFD