

CASE технологии

Лекция 4

Методологии проектирования ИС

- Вспомним, что
CASE-технология – это методология проектирования ИС и набор инструментальных средств для наглядного моделирования, анализа модели и разработки приложений
- Изучаем разные методологии проектирования ИС → изучаем разные CASE-технологии

Методологии проектирования ИС

- Структурный анализ и проектирование
- Объектно-ориентированный анализ и проектирование

Структурный анализ и проектирование

- Суть:

система разбивается на функциональные подсистемы, которые в свою очередь делятся на подфункции, подразделяемые на задачи и т.д. Процесс разбиения продолжается вплоть до конкретных процедур

Структурный анализ и проектирование

- Деление на подсистемы соответствует нисходящей методологии проектирования («сверху-вниз»)

Принципы структурного анализа

- **«Разделяй и властвуй»:** решение сложных проблем путем их разбиения на множество меньших независимых задач, легких для понимания и решения
- **Иерархическое упорядочивание:** организация составных частей проблемы в иерархические древовидные структуры с добавлением новых деталей на каждом уровне

Принципы структурного анализа

- **принцип абстрагирования** - определение существенных аспектов системы и отказ от несущественных аспектов
- **принцип формализации** - строгий методический подход к решению проблемы
- **принцип непротиворечивости** - обоснованность и согласованность элементов
- **принцип структурирования данных** - данные должны быть структурированы и иерархически организованы

Структурный анализ

- В структурном анализе используются в основном две группы средств:
 - Средства **описания функций**, выполняемых системой
 - Средства **описания отношений между данными**
- Каждой группе средств соответствуют определенные виды моделей (диаграмм)

Структурный анализ

- наиболее распространенными моделями являются:
 - **SADT** (Structured Analysis and Design Technique) модели и соответствующие функциональные диаграммы
 - **DFD** (Data Flow Diagrams) диаграммы потоков данных
 - **ERD** (Entity-Relationship Diagrams) диаграммы "сущность-связь"

Структурный анализ

- Перечисленные модели (SADT, DFD, ERD) в совокупности дают полное описание ИС независимо от того, является ли она существующей или вновь разрабатываемой.
- Состав диаграмм в конкретном случае зависит от необходимой полноты описания системы

Функциональная модель системы

- Результатом применения методологии SADT является **модель**, которая состоит из диаграмм, фрагментов текстов и глоссария, имеющих ссылки друг на друга.
- **Диаграммы** - главные компоненты модели, все функции ИС и интерфейсы на них представлены как блоки и дуги.
- Место соединения дуги с блоком определяет тип **интерфейса**.
- **Управляющая информация** входит в блок **сверху**
- **Информация**, которая подвергается **обработке**, показана с **левой стороны блока**
- Результаты **выхода** показаны с **правой стороны**
- **Механизм** (человек или автоматизированная система), который осуществляет операцию, представляется **дугой, входящей в блок снизу**

Функциональная модель системы



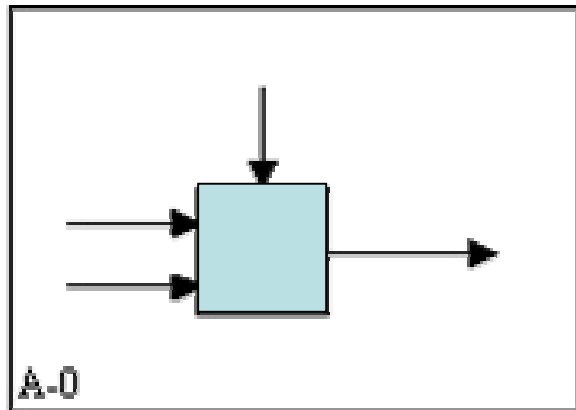
Функциональная модель системы

- Очень простой пример: функциональная модель верхнего уровня для банковской ИС

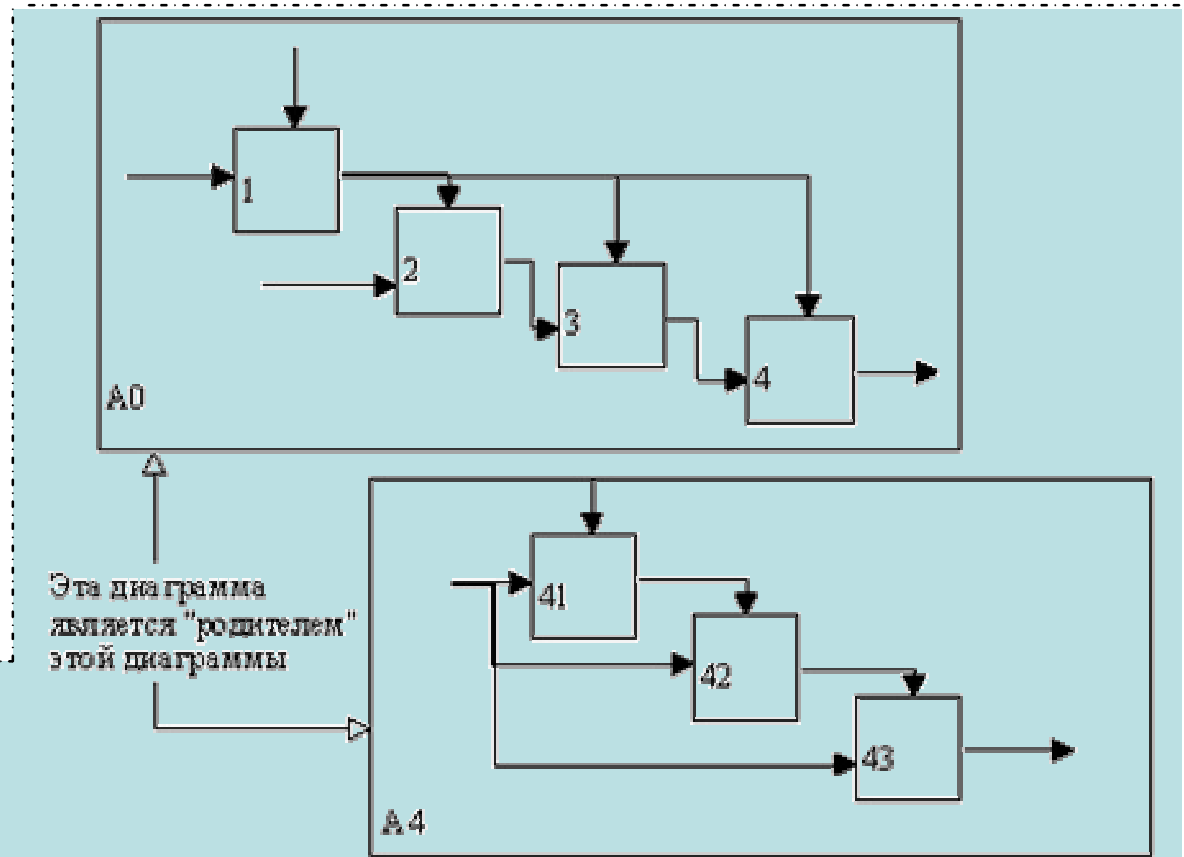


Диаграмма самого высокого уровня (обычно A0) называется еще **контекстной**, или «**взгляд руководителя системы**»

Функциональная модель системы

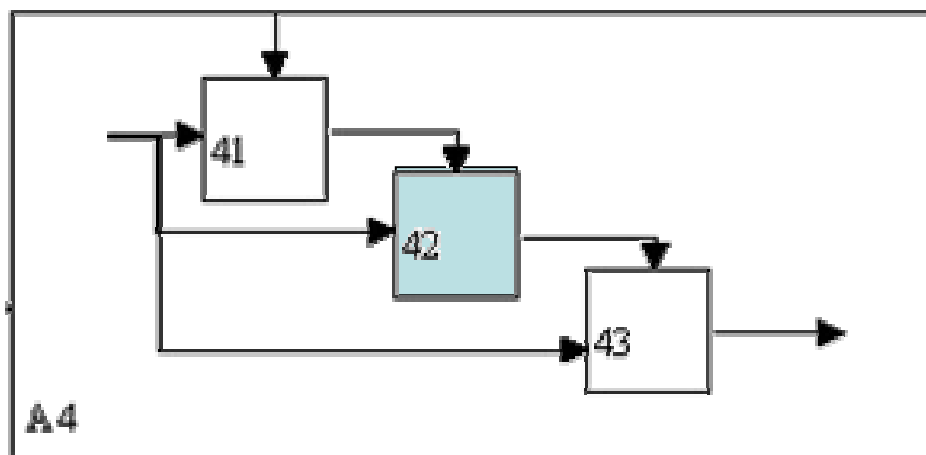


- Более общее представление

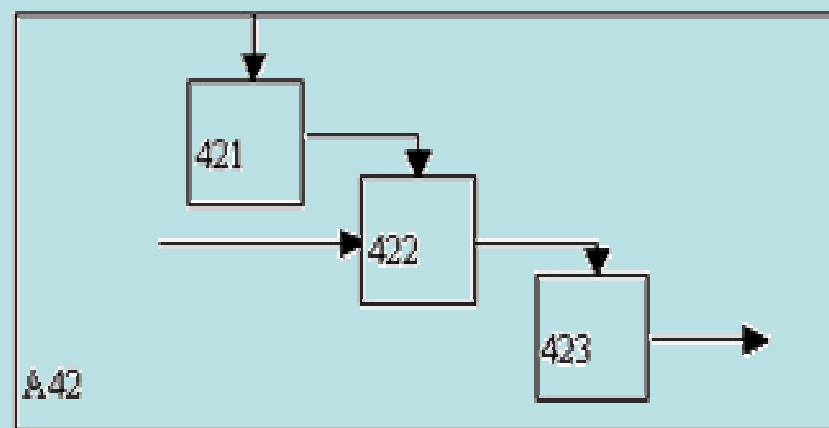


- Более детальное представление

Функциональная модель системы



- Более общее представление



- Более детальное представление

Функциональная модель системы

- Построение SADT-модели начинается с представления всей системы в виде простейшей компоненты – одного блока и дуг, изображающих интерфейсы с функциями вне системы
- Поскольку единственный блок представляет всю систему как единое целое, имя, указанное в блоке, является общим.
- Интерфейсные дуги - также представляют полный набор внешних интерфейсов системы в целом (имена – также общие)

Функциональная модель системы

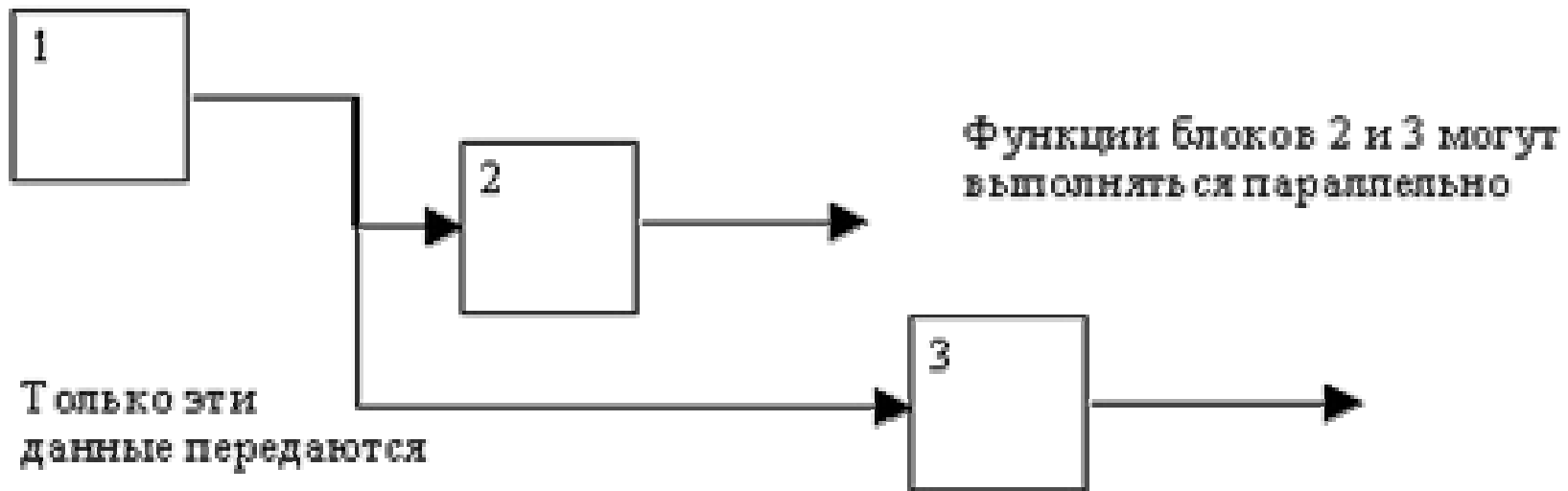
- Блок, который представляет систему в качестве единого модуля, детализируется на другой диаграмме с помощью нескольких блоков, соединенных интерфейсными дугами
- Эти блоки представляют основные подфункции исходной функции
- Данная декомпозиция выявляет **полный** набор подфункций, каждая из которых представлена как блок, границы которого определены интерфейсными дугами

Функциональная модель системы

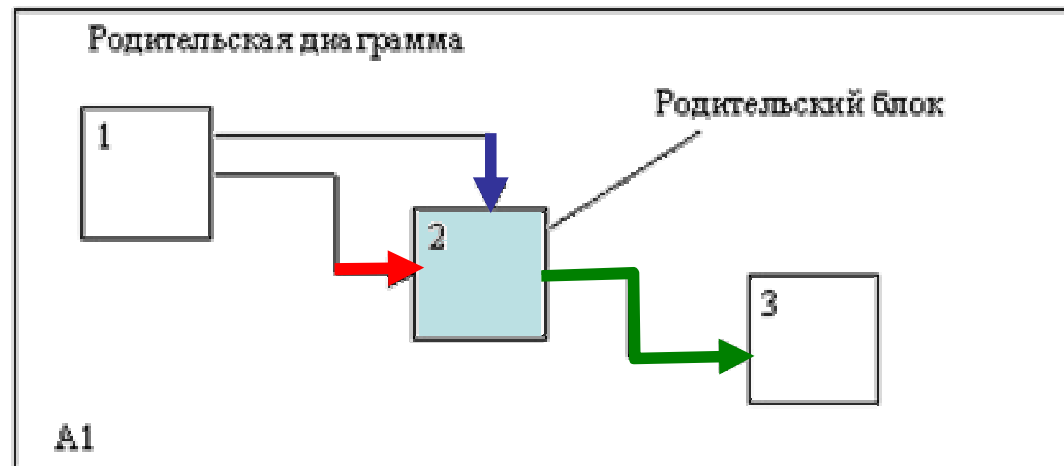
- Во всех случаях каждая подфункция может содержать **только те** элементы, **которые входят в исходную** функцию
- Модель **не может опустить** какие-либо элементы (поскольку контекст уже определен на верхнем уровне)
- К контексту нельзя ничего добавить, и из него не может быть ничего удалено

Функциональная модель системы

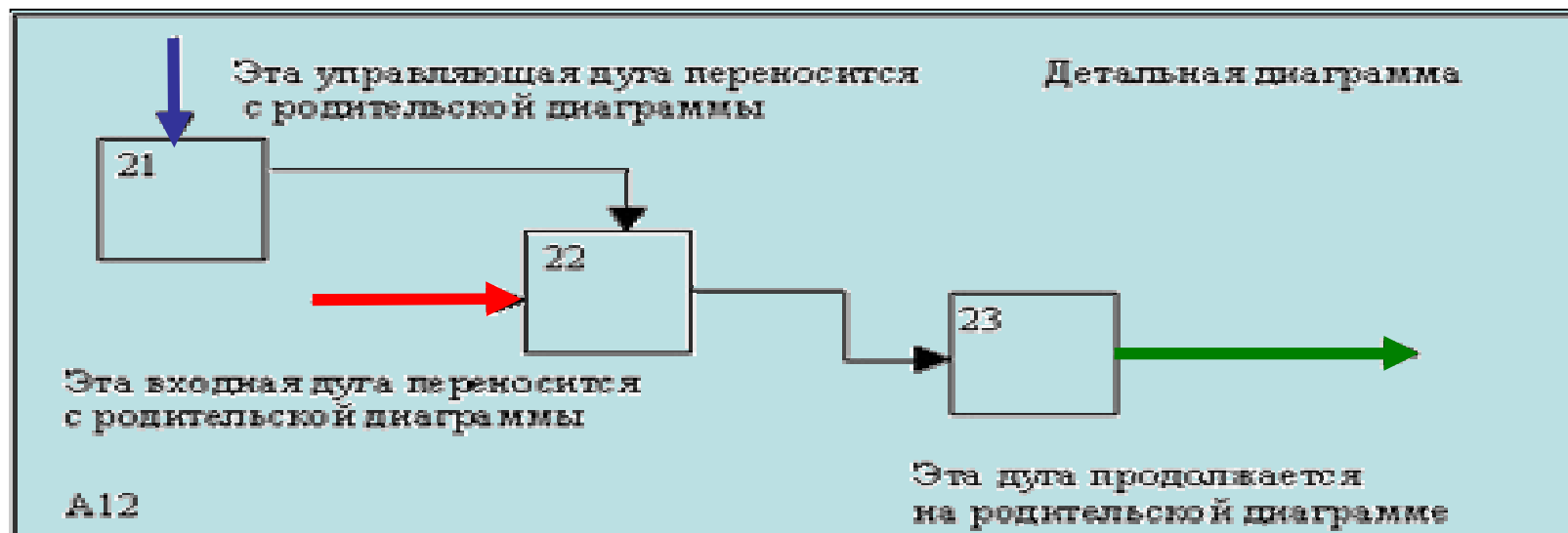
- Одновременное выполнение блоков 2 и 3



Функциональная модель системы

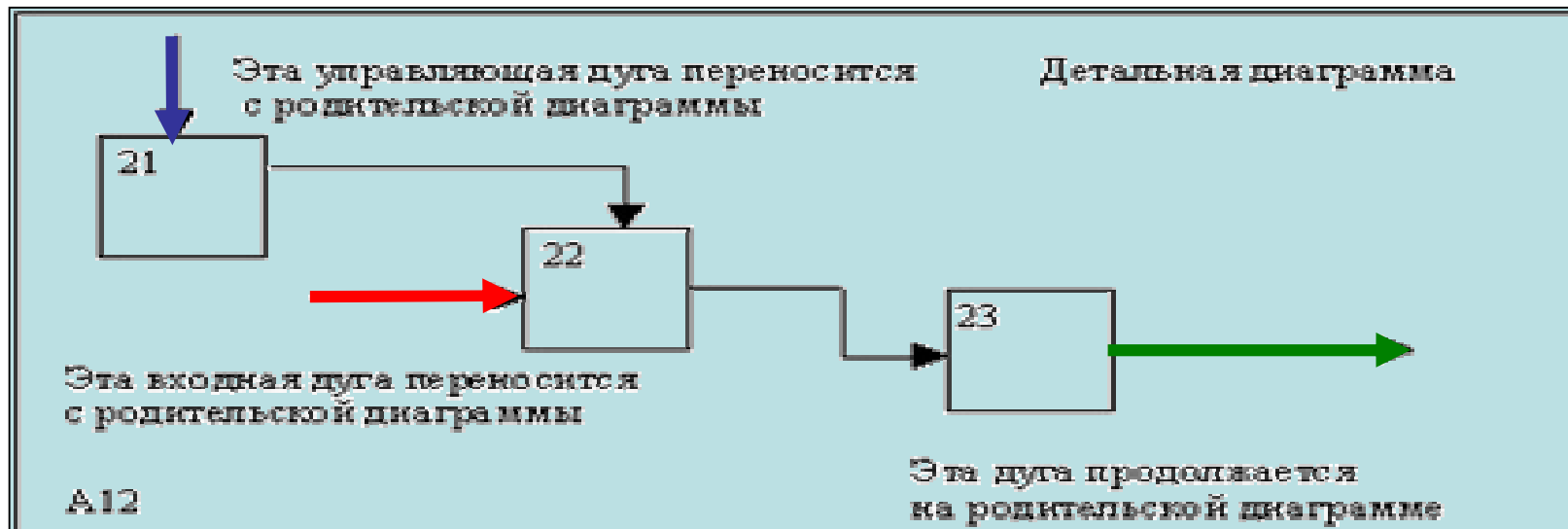


- Соответствие между дугами на родительской и на дочерней диаграммах должно быть полным и непротиворечивым



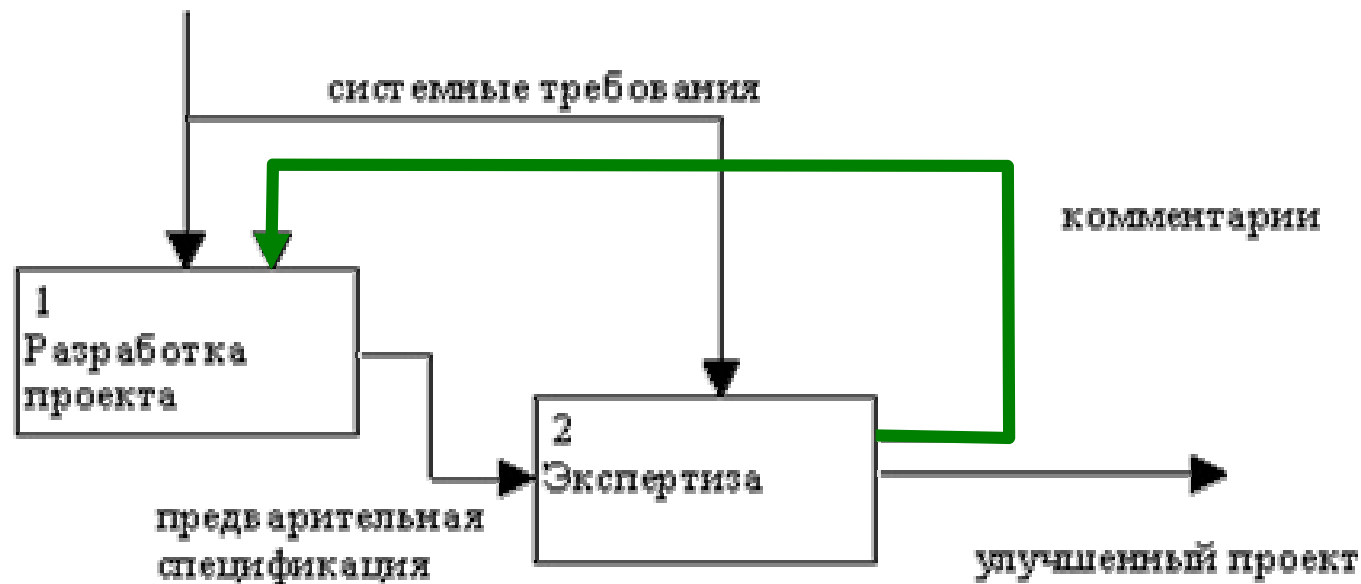
Функциональная модель системы

- Некоторые дуги присоединены к блокам диаграммы обоими концами, у других же один конец остается **неприсоединенным**
- Неприсоединенные дуги соответствуют входам, управлениям и выходам родительского блока
- Источник или получатель этих пограничных дуг может быть обнаружен только на родительской диаграмме.



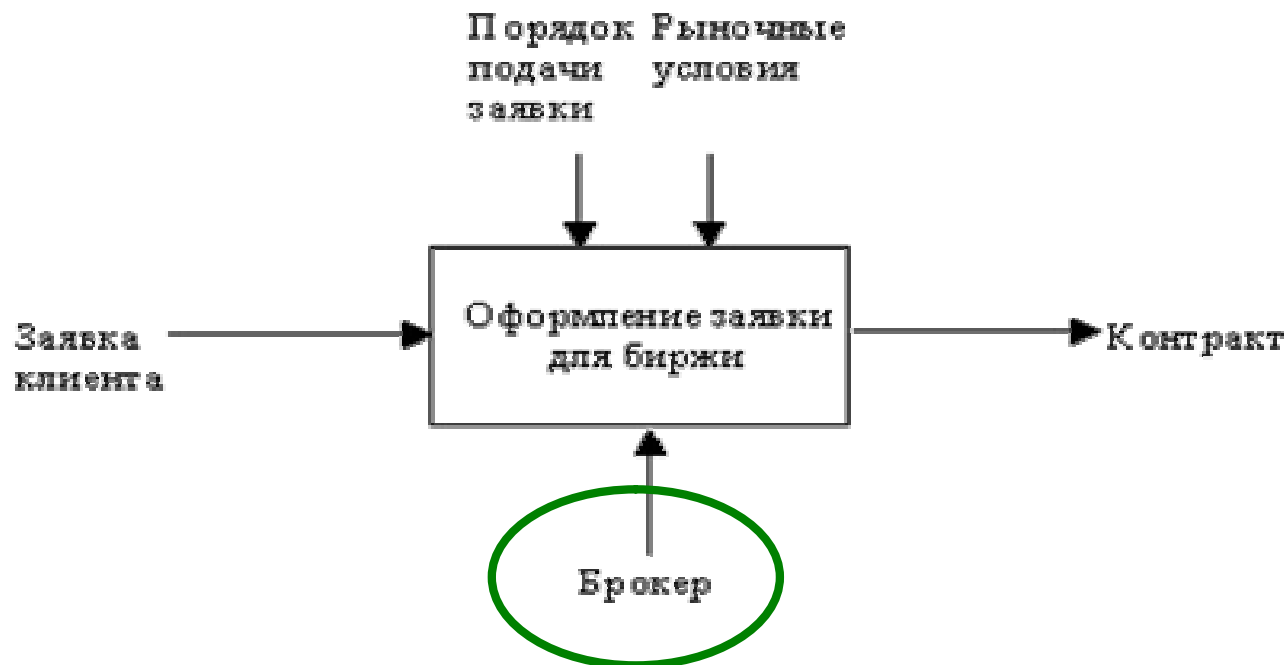
Функциональная модель системы

- На SADT-диаграммах не указаны явно ни последовательность, ни время
- Обратные связи, итерации, продолжающиеся процессы и перекрывающиеся (по времени) функции могут быть изображены с помощью дуг
- **Обратные** связи могут выступать в виде комментариев, замечаний, исправлений и т.д.



Функциональная модель системы

- Механизмы (дуги с нижней стороны) показывают средства, с помощью которых осуществляется выполнение функций
- **Механизм** может быть человеком, компьютером или любым другим устройством, которое помогает выполнять данную функцию



Функциональная модель системы

- Каждый блок на диаграмме имеет свой номер
- Например, A21 является диаграммой, которая детализирует блок 1 на диаграмме A2
- A2 детализирует блок 2 на диаграмме A0, которая является самой верхней диаграммой модели



Типы связей между блоками

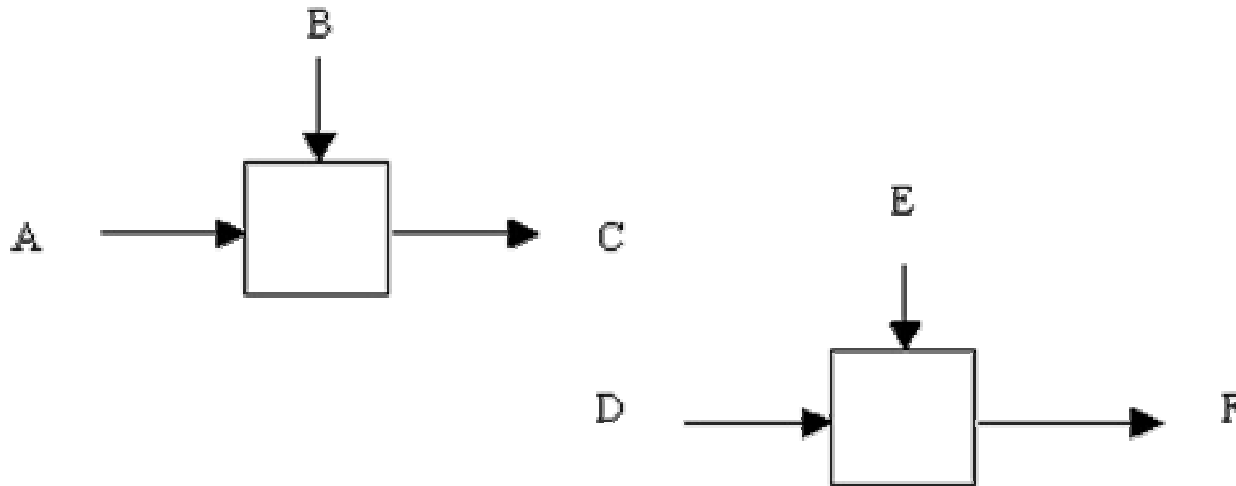
- Различают, по крайней мере, семь типов связывания

| Тип связи | Относительная значимость |
|------------------|-------------------------------------|
| Случайная | 0 |
| Логическая | 1 |
| Временная | 2 |
| Процедурная | 3 |
| Коммуникационная | 4 |
| Последовательная | 5 |
| Функциональная | 6 |

Типы связей между блоками

(0) Тип случайной связности: наименее желательный

- конкретная связь между функциями мала или полностью отсутствует
- Имена данных на SADT-дугах в одной диаграмме имеют малую связь друг с другом



Типы связей между блоками

(1) Тип логической связности

Логическое связывание происходит тогда, когда данные и функции собираются вместе, т.к. они попадают в общий класс или набор элементов, но необходимых функциональных отношений между ними не обнаруживается

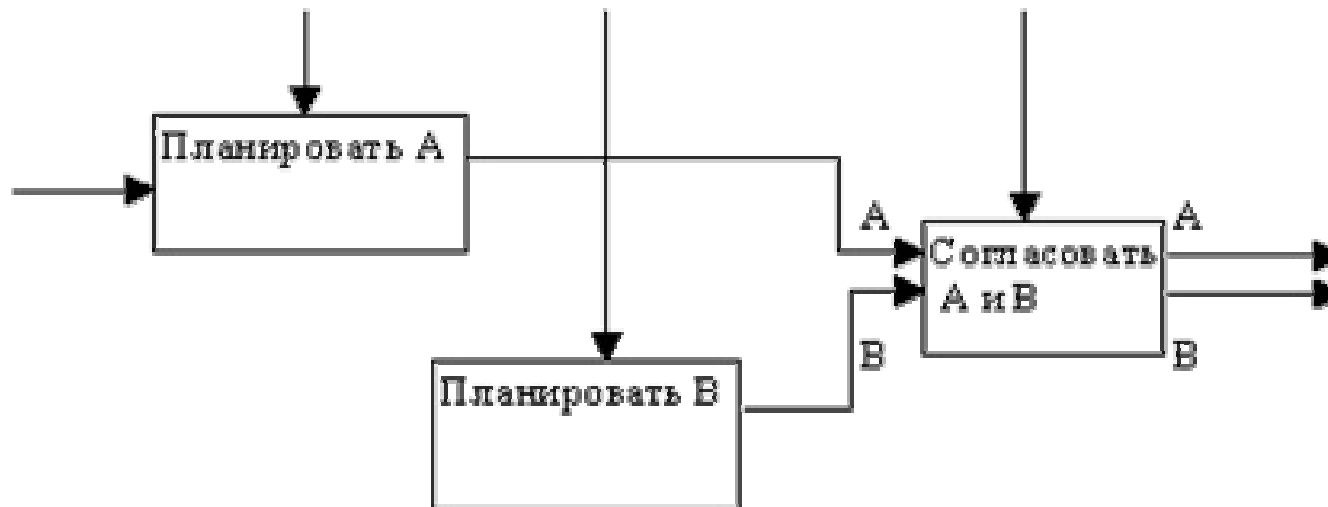
(2) Тип временной связности

Связанные по времени элементы возникают когда они представляют функции, связанные во времени, когда данные используются одновременно или функции включаются параллельно, а не последовательно

Типы связей между блоками

(3) Тип процедурной связности

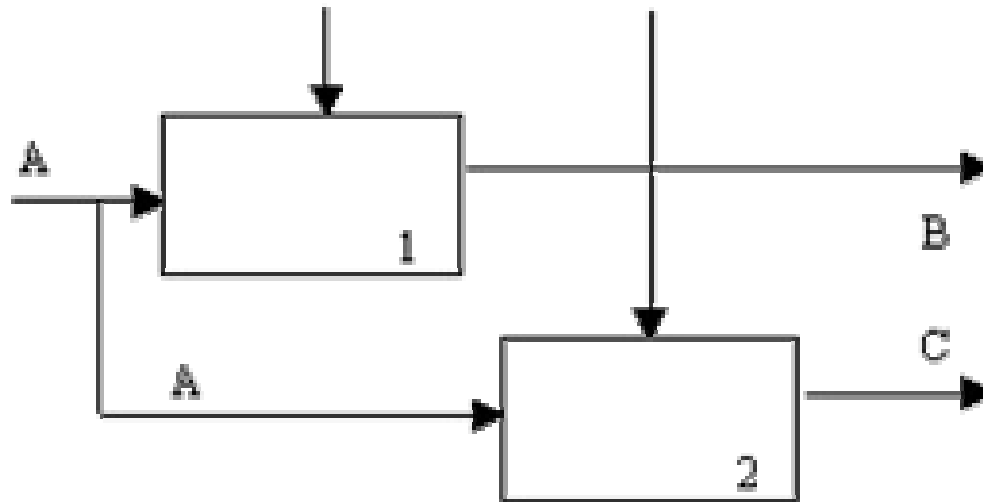
Процедурно-связанные элементы появляются сгруппированными вместе вследствие того, что они выполняются в течение одной и той же части цикла или процесса



Типы связей между блоками

(4) Тип коммуникационной связности

Диаграммы демонстрируют коммуникационные связи, когда блоки группируются вследствие того, что они используют одни и те же входные данные и/или производят одни и те же выходные данные

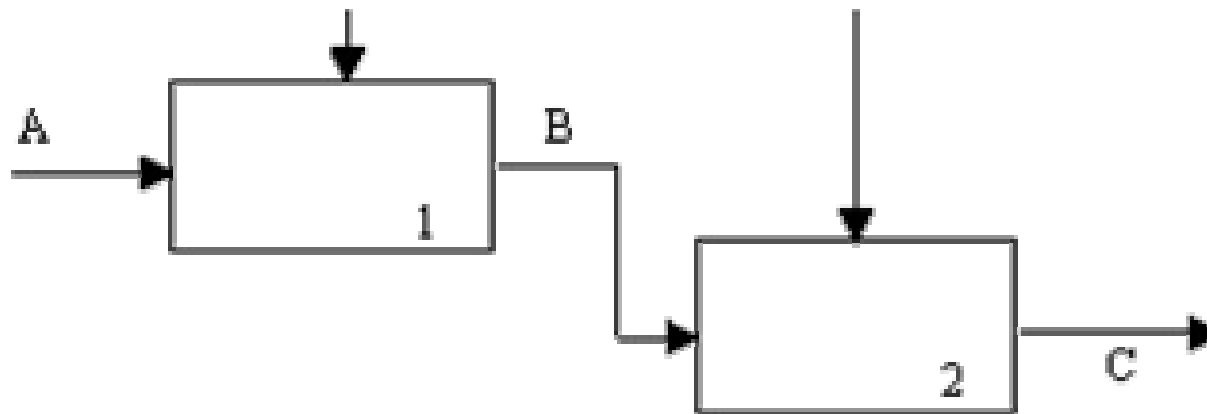


Типы связей между блоками

(5) Тип последовательной связности

На диаграммах, имеющих последовательные связи, выход одной функции служит входными данными для следующей функции

Связь между элементами на диаграмме является более тесной, чем на рассмотренных выше уровнях связей, поскольку моделируются причинно-следственные зависимости

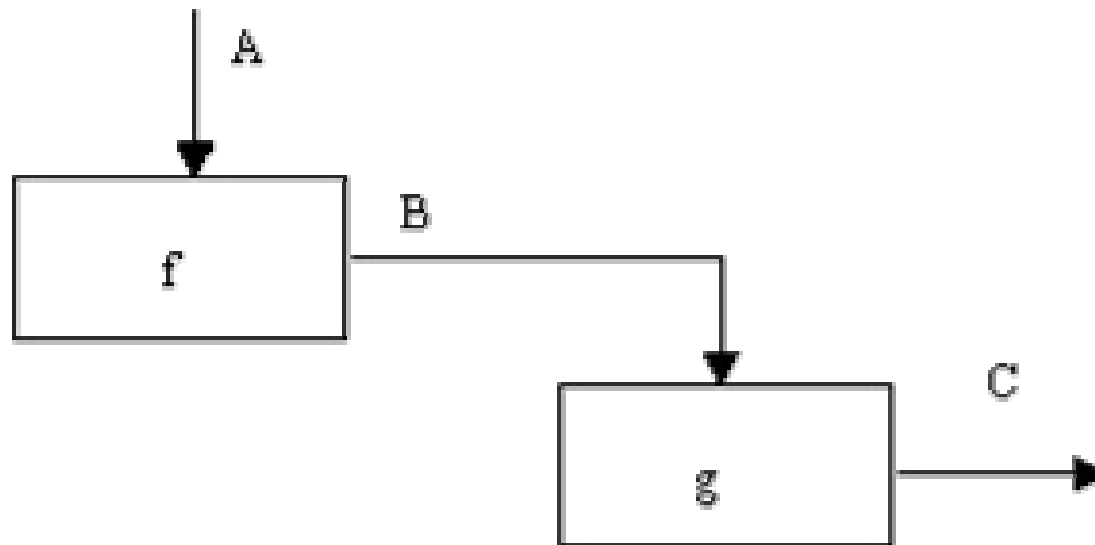


Типы связей между блоками

(6) Тип функциональной связности

Простейший тип функциональной связности имеет следующий вид:

$$C = g(B) = g(f(A))$$



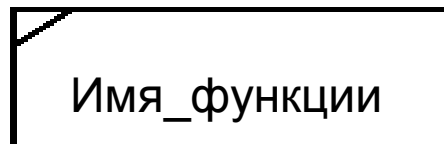
IDEF0: Прикладная часть SADT

- Нотация IDEF0 (**I**ntegration **D**efinition for **F**unction **M**odeling)
 - была разработана на основе методологии структурного анализа и проектирования SADT
 - утверждена в качестве стандарта США
 - успешно эксплуатируется во многих проектах, связанных с описанием деятельности предприятий
- IDEF0 может быть использована для моделирования широкого класса систем

IDEF0: нотация

- Модуль поведения (Unit of Behavior, UOB)

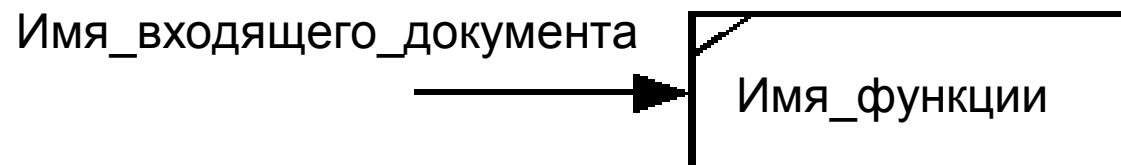
Объект служит для описания функций (процедур, работ), выполняемых подразделениями или сотрудниками предприятия



IDEF0: нотация

- Стрелка слева

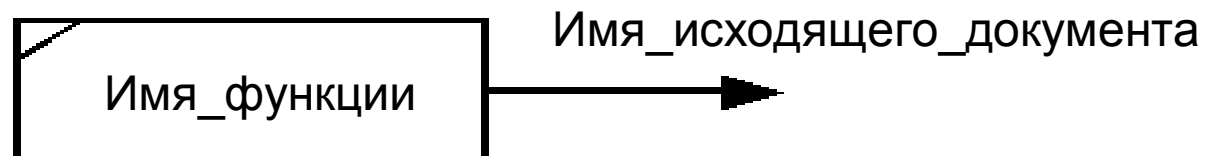
Стрелка описывает **входящие** документы, информацию, материальные ресурсы, необходимые для выполнения функции



IDEF0: нотация

- Стрелка справа

Стрелка описывает **исходящие** документы, информацию, материальные ресурсы, являющиеся результатом выполнения функции



IDEF0: нотация

- Стрелка сверху

Стрелка описывает **управляющее воздействия**, например распоряжение, нормативный документ и т.д.

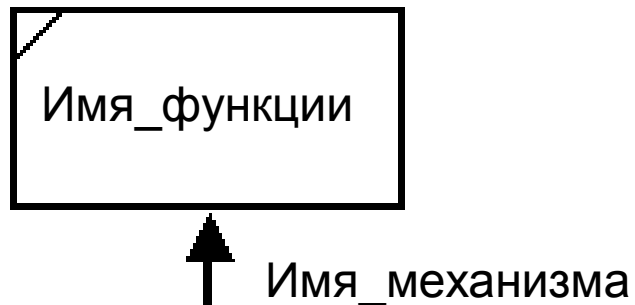
В нотации IDEF0 каждая процедура должна обязательно иметь не менее одной стрелки сверху, отражающей управляющее воздействие



IDEF0: нотация

- Стрелка снизу

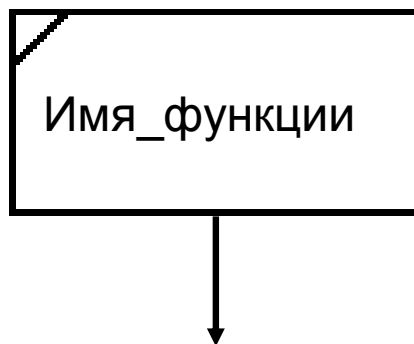
Стрелка снизу описывает т.н. **механизмы**, т.е. ресурсы, необходимые для выполнения процедуры, но не изменяющие в процессе ее выполнения свое состояние



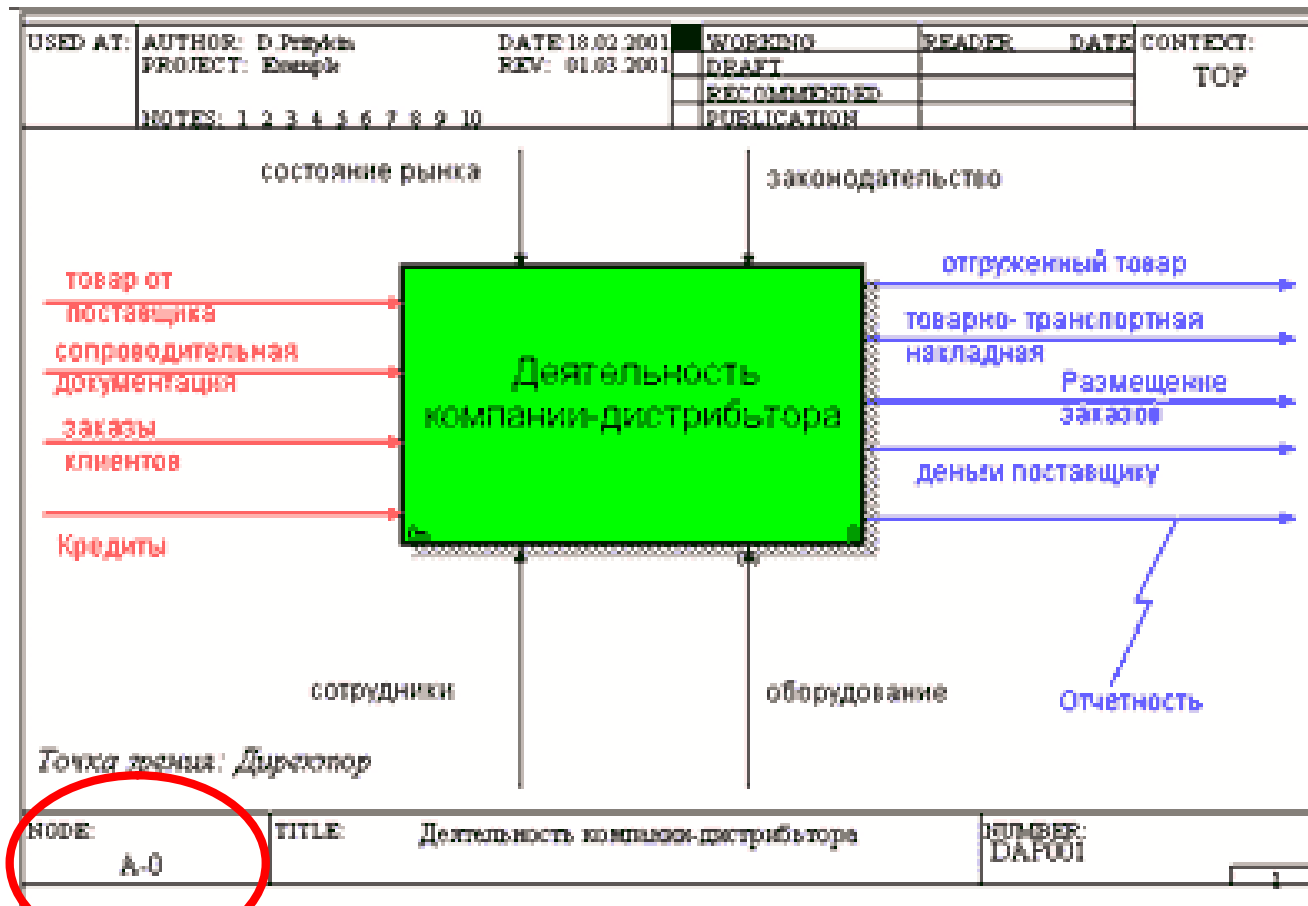
IDEF0: нотация

- Стрелка вниз

Стрелка вниз изображает связь между разными диаграммами или моделями, указывая на некоторую диаграмму, где данная работа рассмотрена более подробно

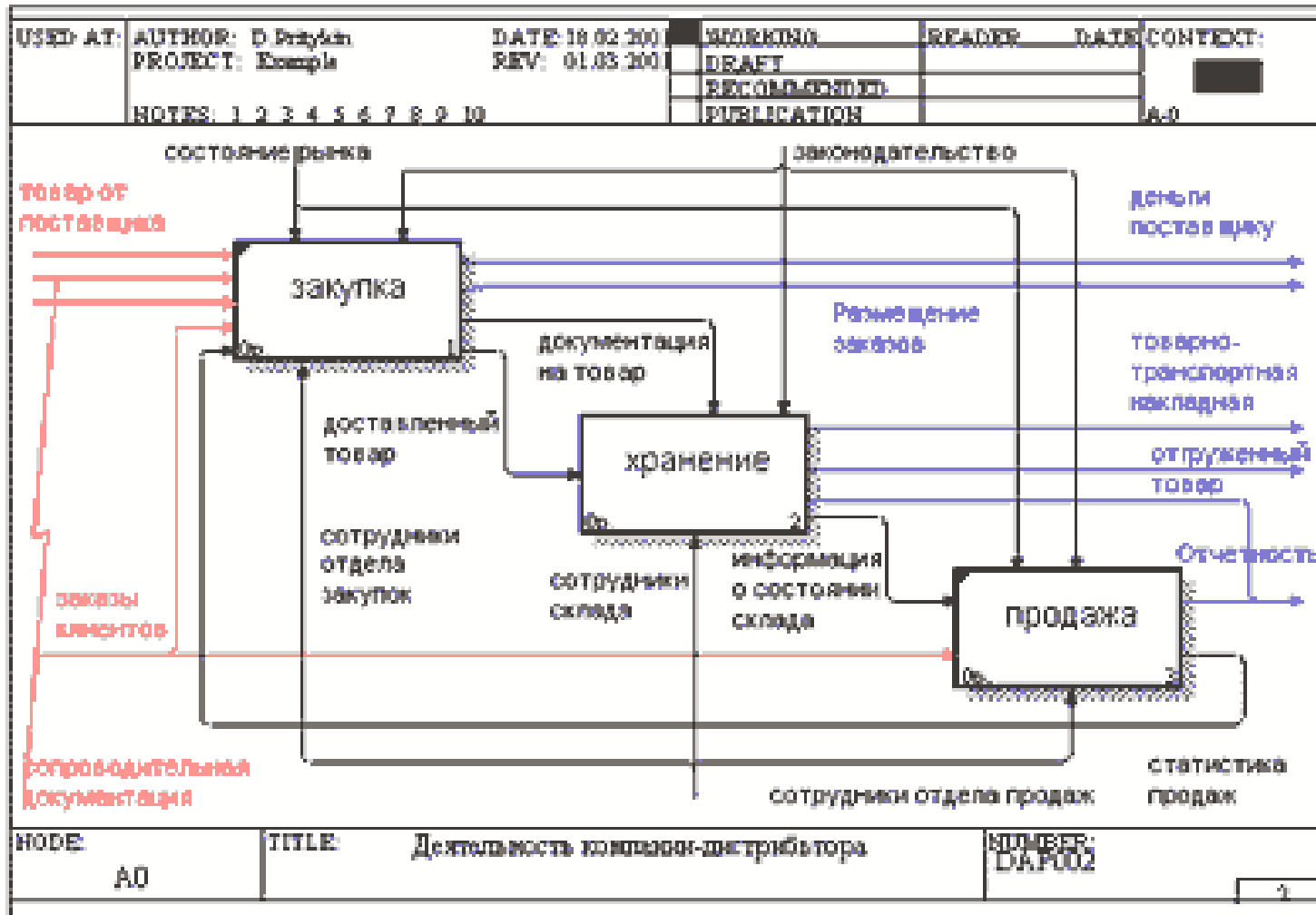


Пример функциональной модели



Контекстная диаграмма (почему?)

Пример функциональной модели



Резюме: что узнали на лекции?

- Какие есть методологии проектирования
- Какие принципы лежат в основе методологии структурного анализа и проектирования
- Какие виды моделей используются в методологии структурного анализа и проектирования
- Познакомились со способом изображения функциональной модели системы – нотацией IDEF0 (часть SADT)