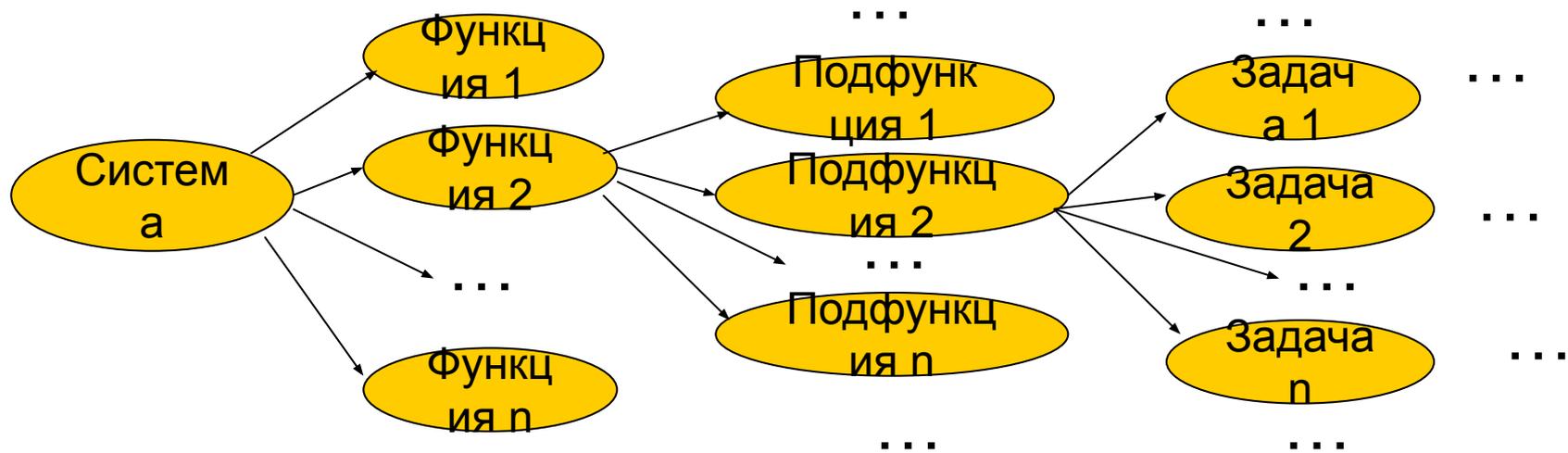


# Структурный подход к моделированию систем

Методология  
функционального  
моделирования IDEF0

# Сущность структурного подхода к моделированию систем

Система разбивается на функциональные подсистемы, которые, в свою очередь, делятся на подфункции, подфункции – на задачи и т.д. до конкретных процедур



# Базовые принципы структурного подхода

- принцип *«разделяй и властвуй»*
- принцип *иерархического упорядочивания*
- принцип *абстрагирования*
- принцип *формализации*
- принцип *непротиворечивости*
- принцип *структурирования данных*

# Методология структурного анализа и проектирования

- 70-е гг. XX века – методология **SADT( Structured Analysis and Design Technique)**
- Предложена *Дугласом Россом* (Douglas Ross)
- Основная идея данной методологии – построение древовидной иерархической модели предприятия.
- В начале 1990-х на основе SADT принят стандарт моделирования бизнес-процессов **IDEF0 (Icam DEFinition)**, являющийся одним из 14 стандартов линейки *IDEF – Integration Definition for Functional Modeling* (в данном курсе будут рассмотрены некоторые из них, в частности, IDEF0, IDEF1X, IDEF3)
- Положения методологии зафиксированы в разработанном в США стандарте IDEF0 (В России – *РД IDEF0 – 2000*)

# Модели структурного подхода,

- **3 типа моделей**, используемых в структурном подходе:
- 1) функциональные модели (ФМ)
- 2) информационные модели (ИМ)
- 3) динамические модели (ДМ)

ФМ	SADT (IDEF0)-модели DFD-модели	Design/IDEF, MS Visio, ERWin Procces Modeler (BPWin)
ИМ	ERD (IDEF1X)	ERWin Data Modeler, MS Visio
ДМ	IDEF3	Пакет ERWin Procces Modeler (BPWin), MS Visio

# Сущность функционального моделирования

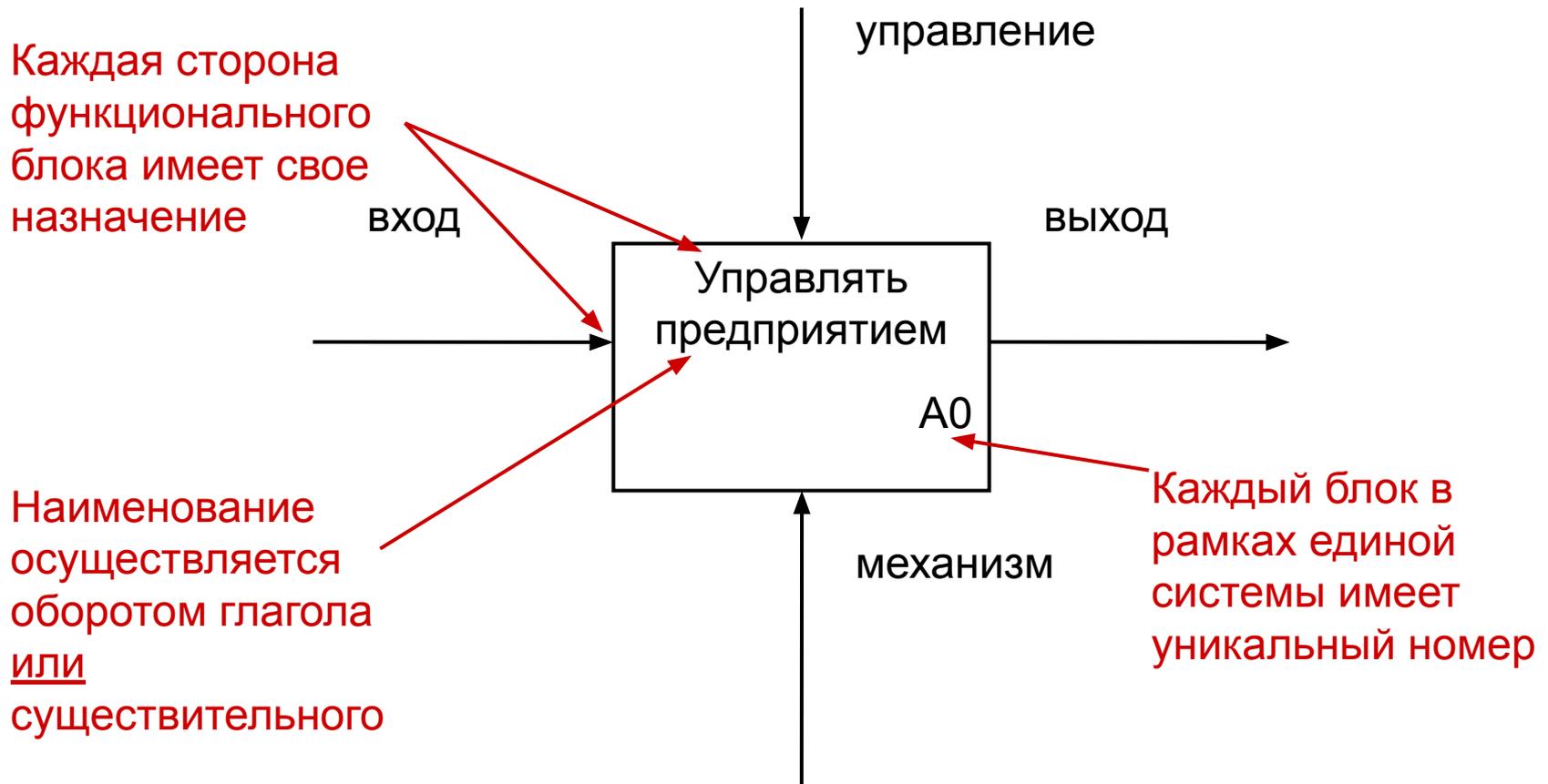
Для любой системы определяющим является ее функциональное содержание, так как оно определяет ее основные свойства. Поэтому в основе функционального моделирования лежит **функциональное содержание системы**, в качестве отношений между функциями рассматривается **информация об объектах**, связывающих эти функции .

# Методология IDEF0

- В основе *IDEF0*-методологии лежат **4 основных понятия**:
  - 1) функциональный блок;
  - 2) интерфейсная дуга (стрелка);
  - 3) декомпозиция;
  - 4) глоссарий.

# Функциональный блок

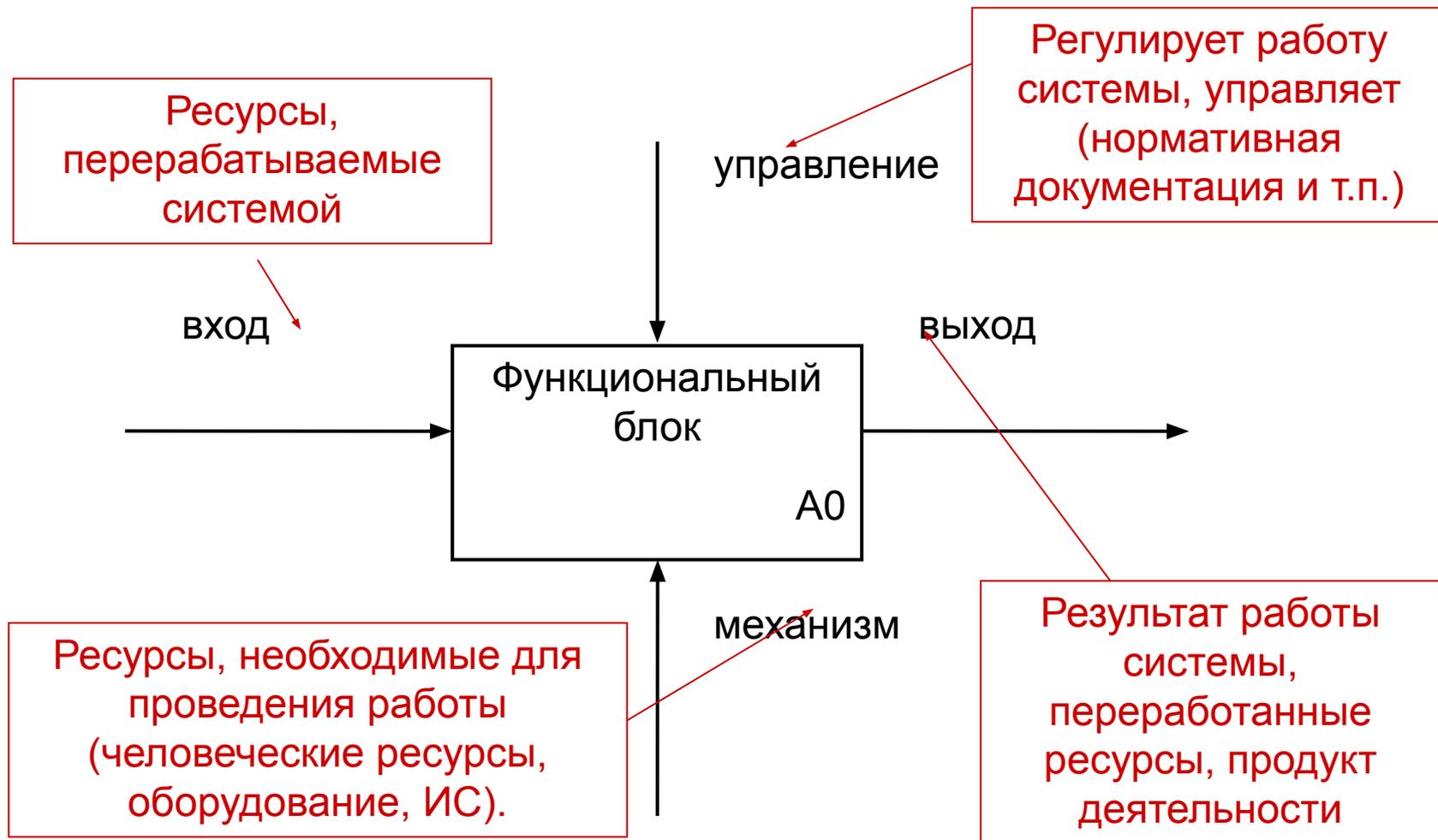
- Олицетворяет некоторую конкретную функцию или работу в рамках рассматриваемой системы
- *РД IDEFO – 2000*: прямоугольник, содержащий имя и номер и используемый для описания функции



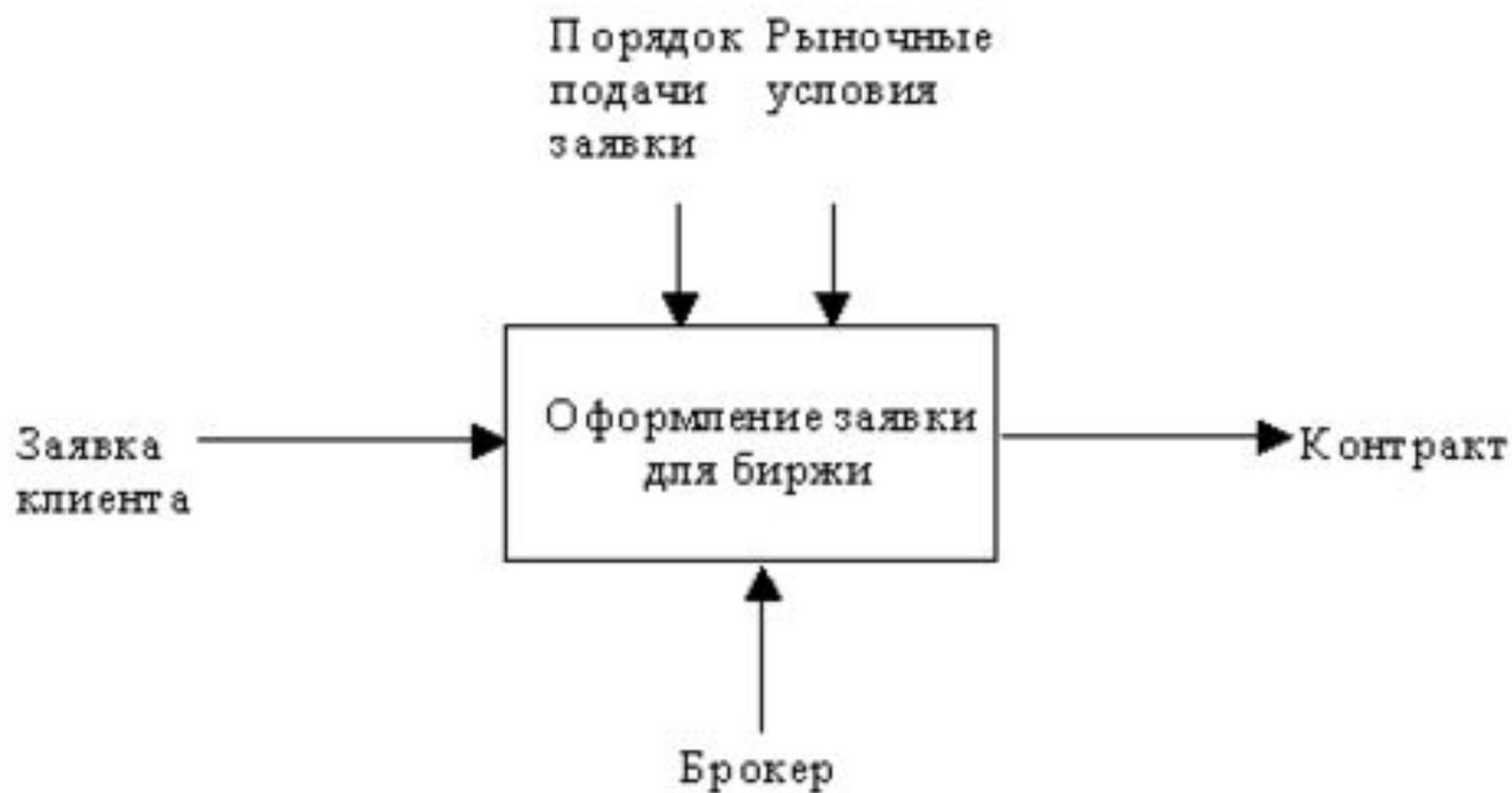
# Интерфейсная дуга

- Интерфейсная дуга отображает элемент системы, который обрабатывается функциональным блоком или оказывает иное влияние на функцию, отображаемую функциональным блоком.
- *Графически* изображается в виде однонаправленной стрелки.
- Каждая дуга должна иметь свое уникальное *название*, сформулированное оборотом существительного (должно отвечать на вопросы кто?, что?). Примеры: информация, разработчик, документ, обработанная заявка.
- В зависимости от того, к какой стороне блока она подходит, интерфейсная дуга будет являться *входящей, выходящей, управления, механизма*.

# Интерфейсная дуга



Стрелки входа может не быть. Остальные интерфейсные дуги обязательны.



# Декомпозиция

- Принцип декомпозиции применяется при разбиении сложных процессов на составляющие его функции. При этом уровень детализации определяется непосредственно разработчиком модели.
- Модель IDEF0 всегда начинается с рассмотрения системы как единого целого, т.е. одного функционального блока с интерфейсными дугами, простирающимися за пределы рассматриваемой области. Такая диаграмма называется *контекстной*, она обозначается идентификатором А-0.
- Для определения границ системы **на контекстной диаграмме обязательно должны быть цель и точка зрения(!!!).**

# Цель моделирования

Цель моделирования должна отвечать на следующие вопросы:

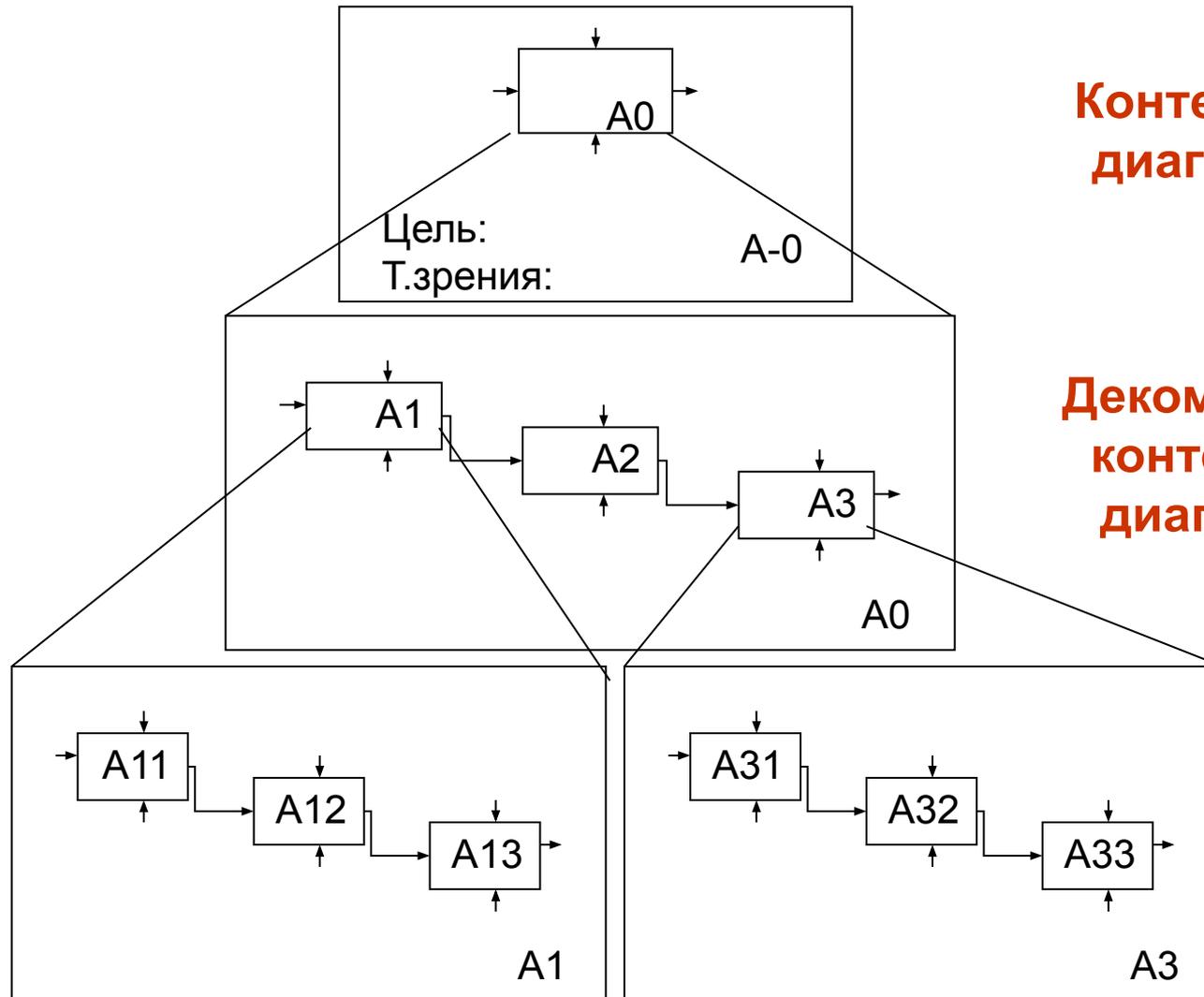
- Почему процесс должен быть замоделирован?
- Что должна показывать модель?
- Что может получить читатель?

Примеры целей: «Идентифицировать слабые стороны процесса сбора данных», «Определить ответственность сотрудников для написания должностных инструкций» и т. п.

# Точка зрения

- Точка зрения – позиция, с которой будет строиться модель. В качестве точки зрения берется взгляд человека, который видит систему в нужном для моделирования аспекте.
- Как правило, выбирается точка зрения человека, **ответственного** за выполнение моделируемой работы.
- Между целью и точкой зрения должно быть жесткое соответствие.

# Декомпозиция



Контекстная  
диаграмма

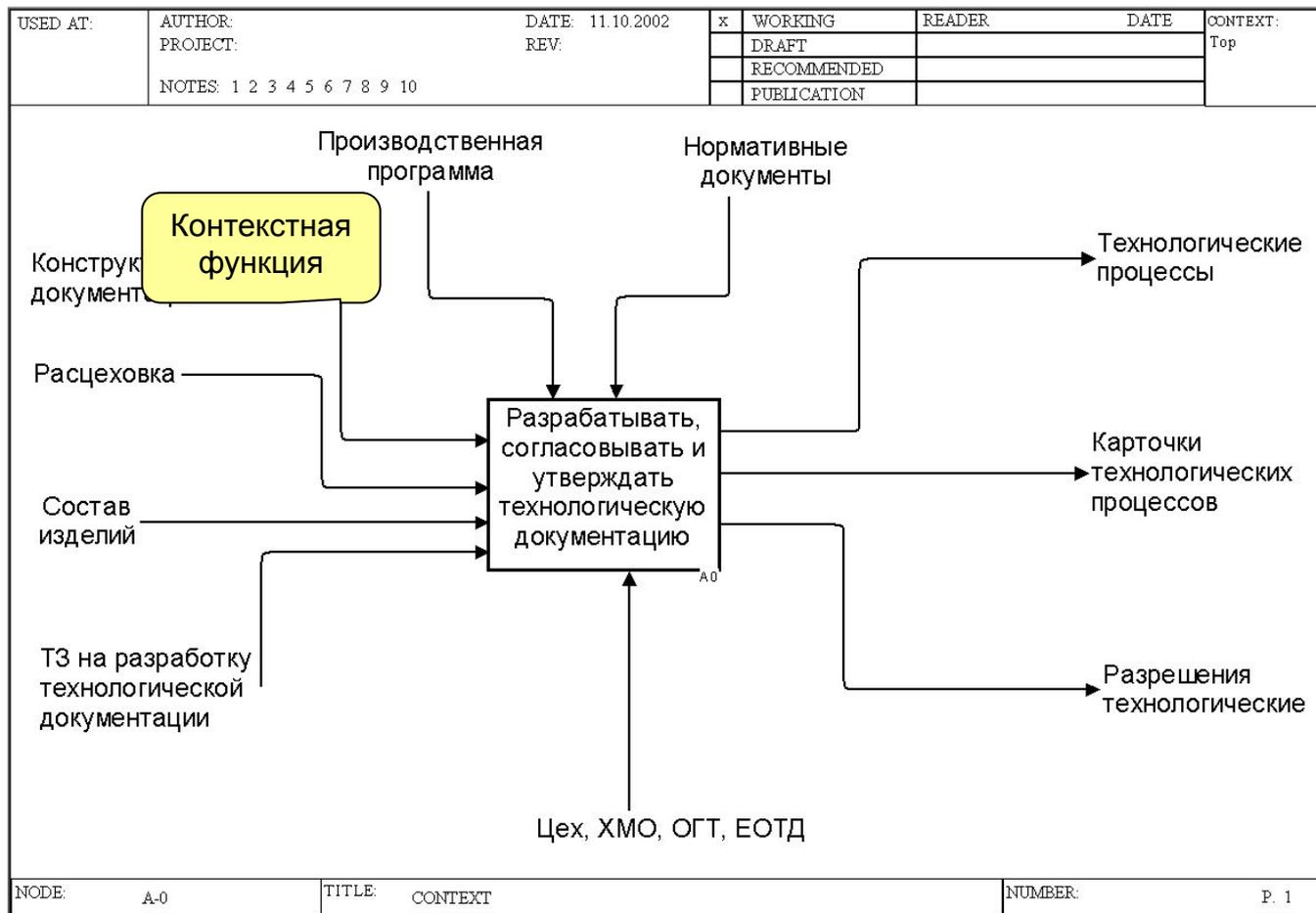
Декомпозиция  
контекстной  
диаграммы

Декомпозиция блока A1

Декомпозиция блока A3

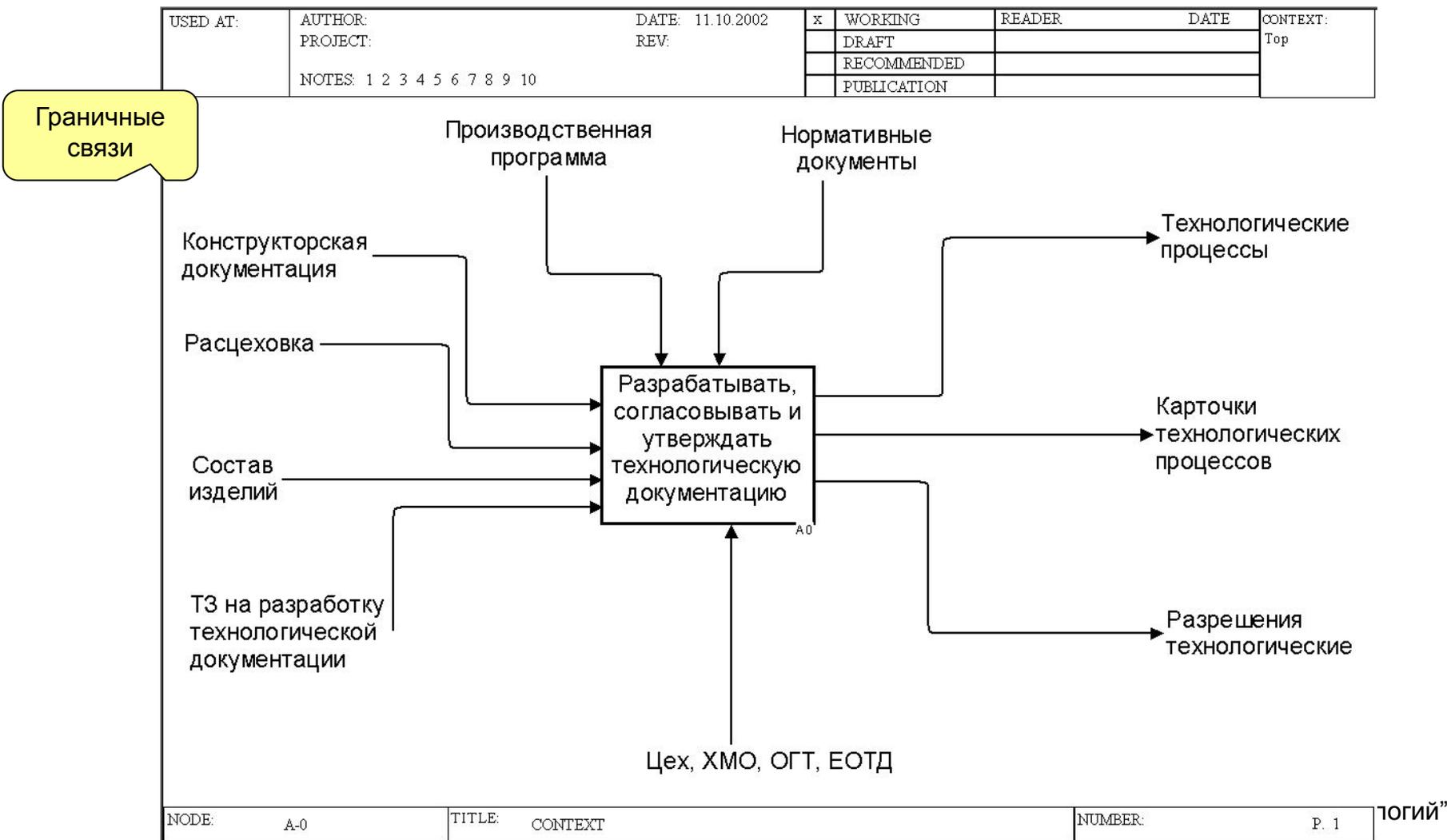
# Контекстная диаграмма(функция)

- Функциональная модель имеет иерархическую структуру.
- Контекстная функция – функция верхнего уровня модели.
- Контекстная функция несет имя основного действия выполняемого системой.
- Изображается на отдельной диаграмме, называемой контекстной.

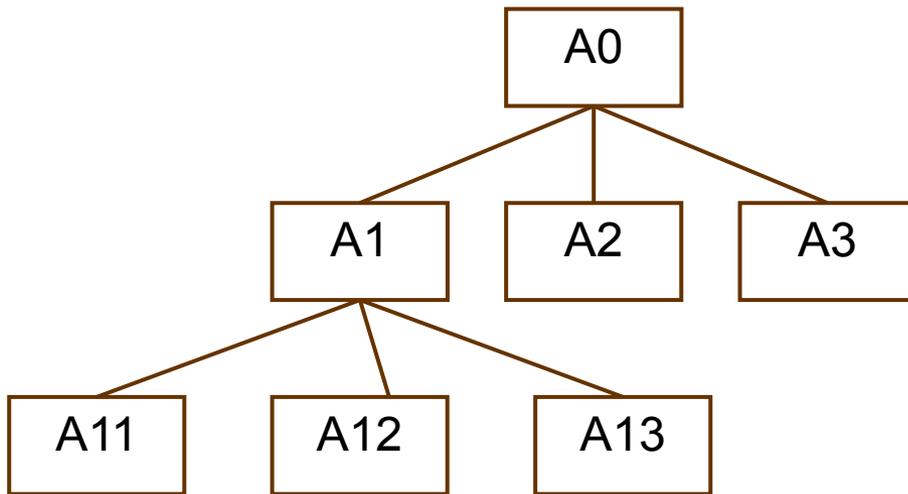


# Контекстная диаграмма

- На контекстной диаграмме указываются связи системы с внешним миром.



# Декомпозиция



Дерево узлов

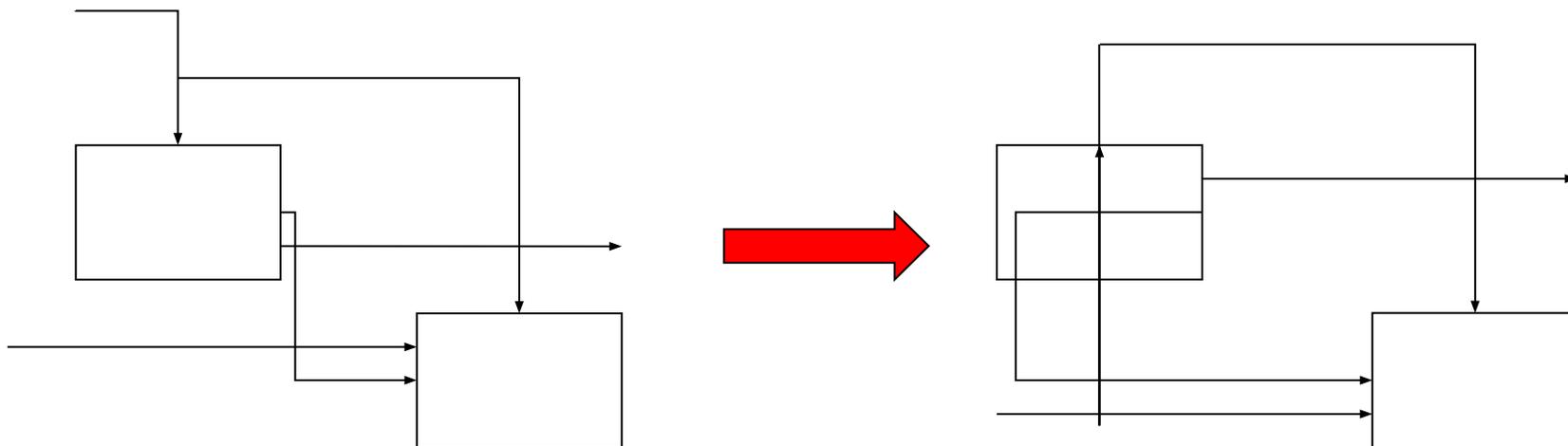
A0 \_\_\_\_\_  
A1 \_\_\_\_\_  
  A11 \_\_\_\_\_  
  A12 \_\_\_\_\_  
  A13 \_\_\_\_\_  
A2 \_\_\_\_\_  
A3 \_\_\_\_\_

Индекс узлов



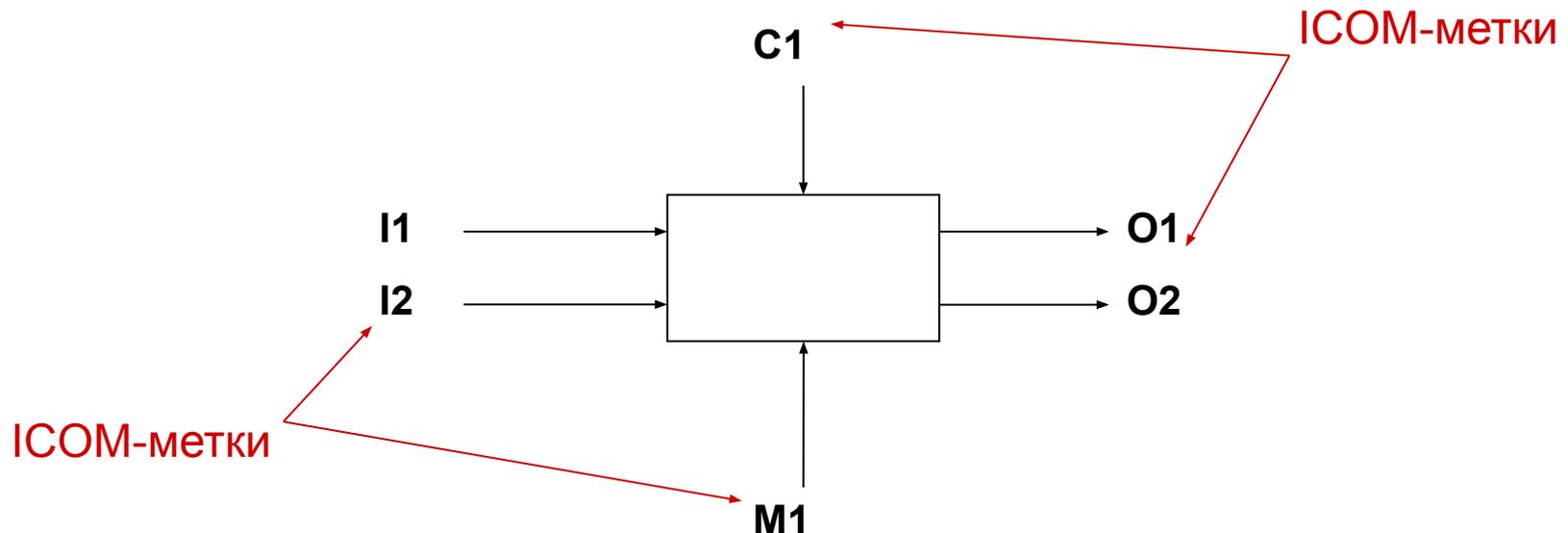
# Основные правила построения диаграмм

1. На одной диаграмме рекомендуется рисовать от 3 до 6 блоков. Иначе диаграмма будет плохо читаемой.
2. Функциональные блоки должны располагаться слева направо сверху вниз в порядке доминирования.
3. Следует избегать излишнего пересечения стрелок.

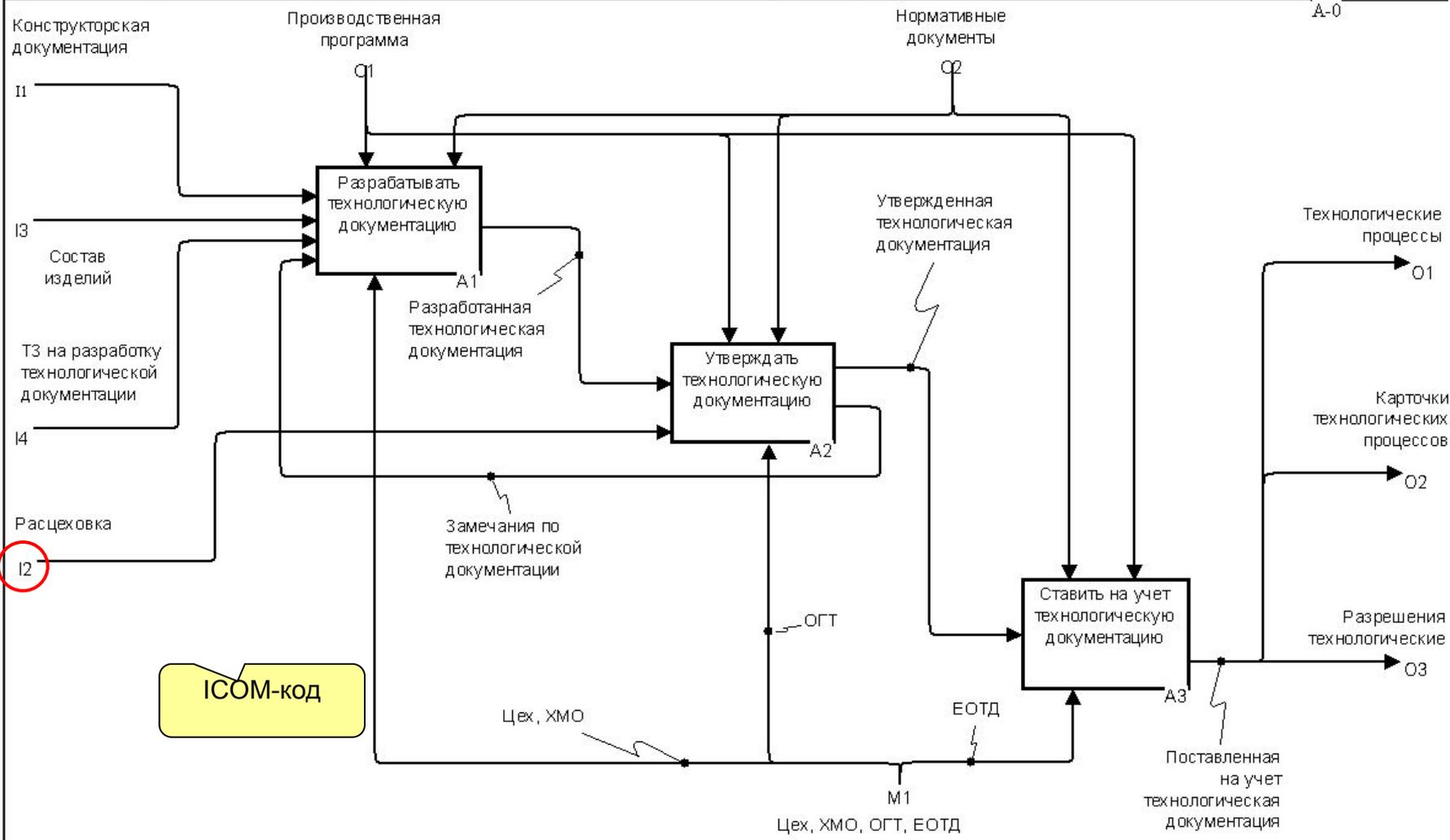


# Граничные стрелки

- **Граничные стрелки** начинаются от границ диаграммы и заканчиваются у функции или наоборот.
- Связывают функции диаграммы с внешним миром.
- Определяются на родительской диаграмме
- Для идентификации граничных стрелок используются ICOM-коды. (Input, Control, Output, Mechanism)



USED AT:	AUTHOR:	DATE: 11.10.2002	x	WORKING	READER	DATE	CONTEXT:
	PROJECT:	REV:		DRAFT			<input type="checkbox"/>
	NOTES: 1 2 3 4 5 6 7 8 9 10			RECOMMENDED			
				PUBLICATION			



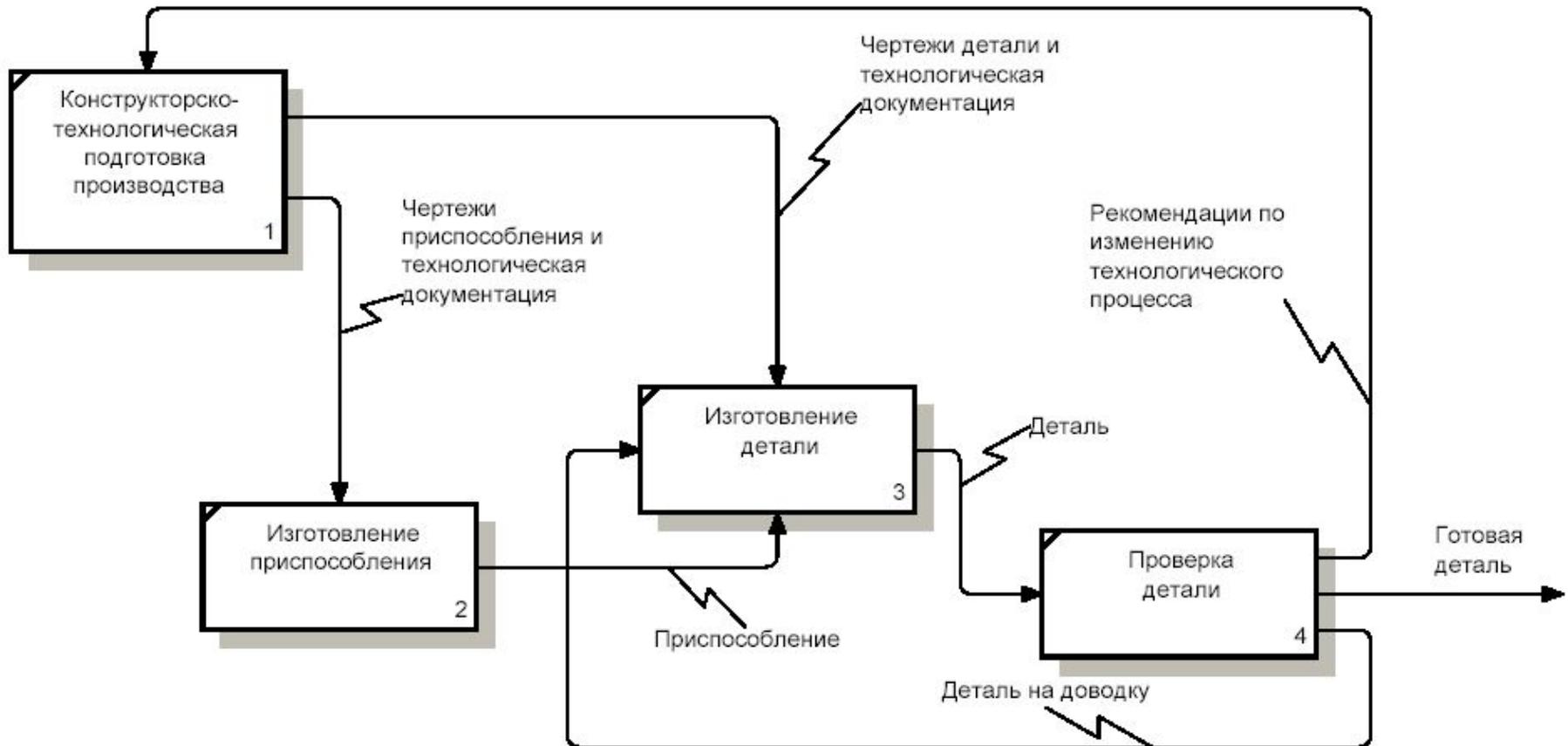
NODE: A0	TITLE: Разрабатывать, согласовывать и утверждать технологическую документацию	NUMBER: P. 2
----------	-------------------------------------------------------------------------------	--------------

# Внутренние связи

- ***Внутренние связи*** не касаются границ диаграммы.
- Разделяются на виды:
  - **Выход-вход.**
  - **Выход-управление.**
  - **Выход-механизм.**
  - **Обратная связь по входу.**
  - **Обратная связь по управлению.**

# ВЫХОД-ВХОД

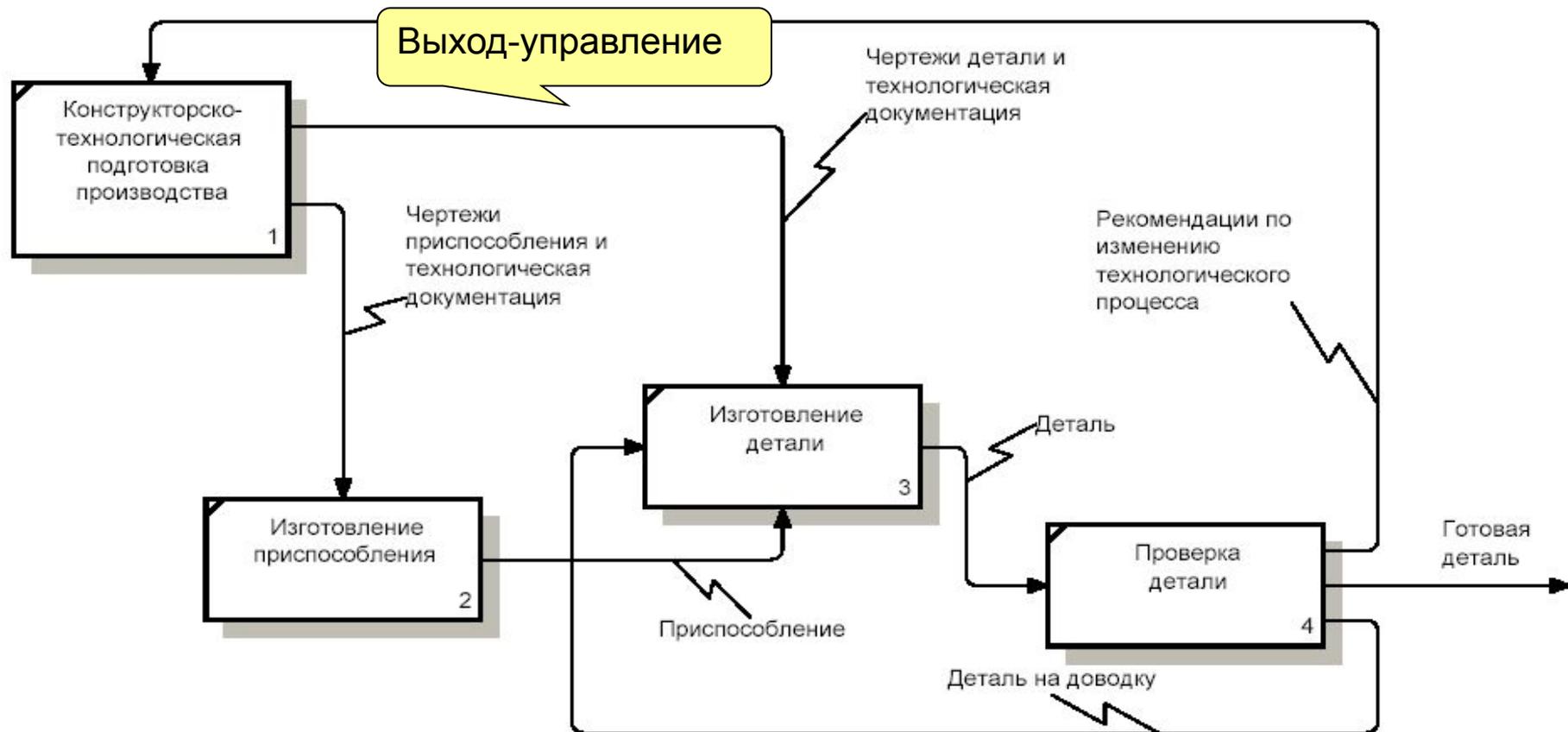
- Описывает последовательность обработки потока объектов.



Выход-вход

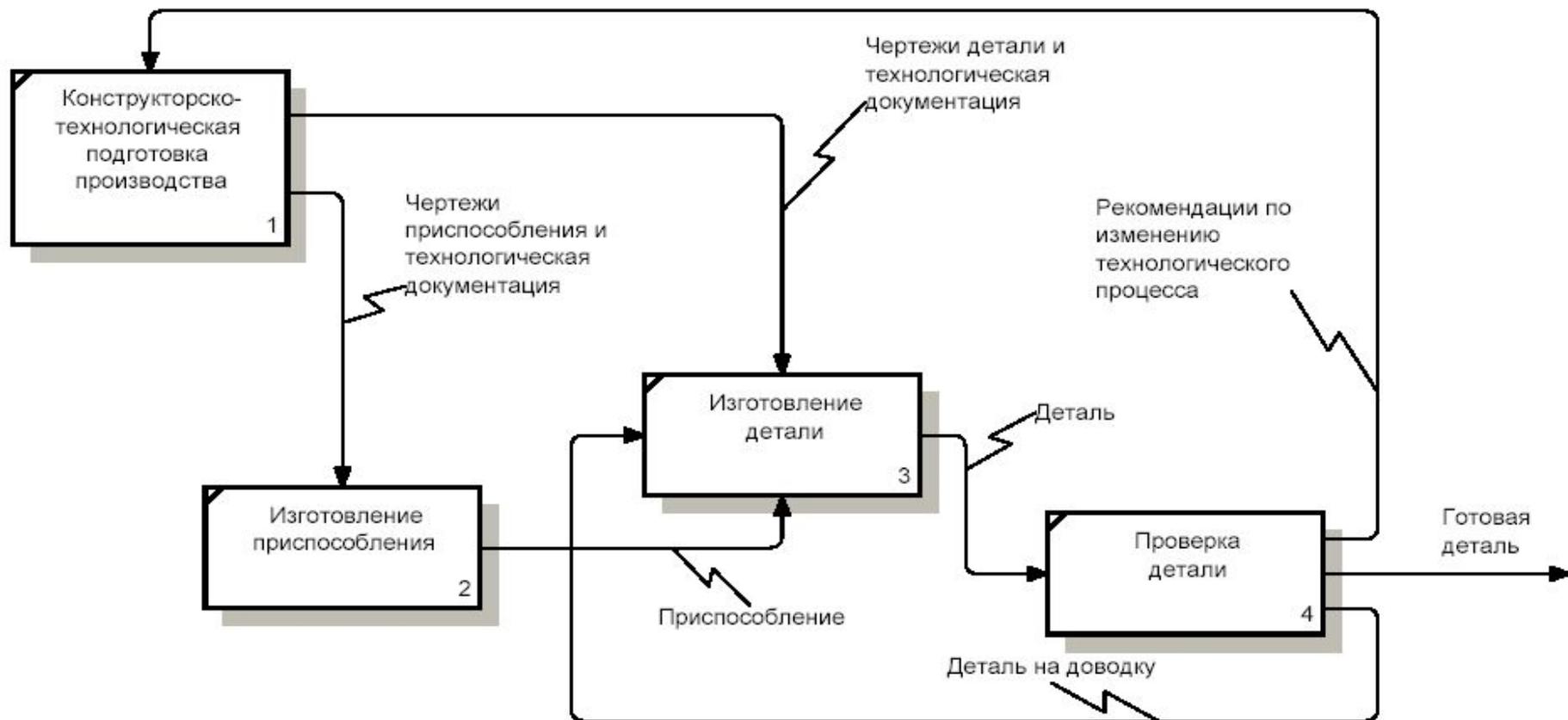
# Выход-управление

- Предшествующая функция управляет выполнением последующей функции.



# Выход-механизм

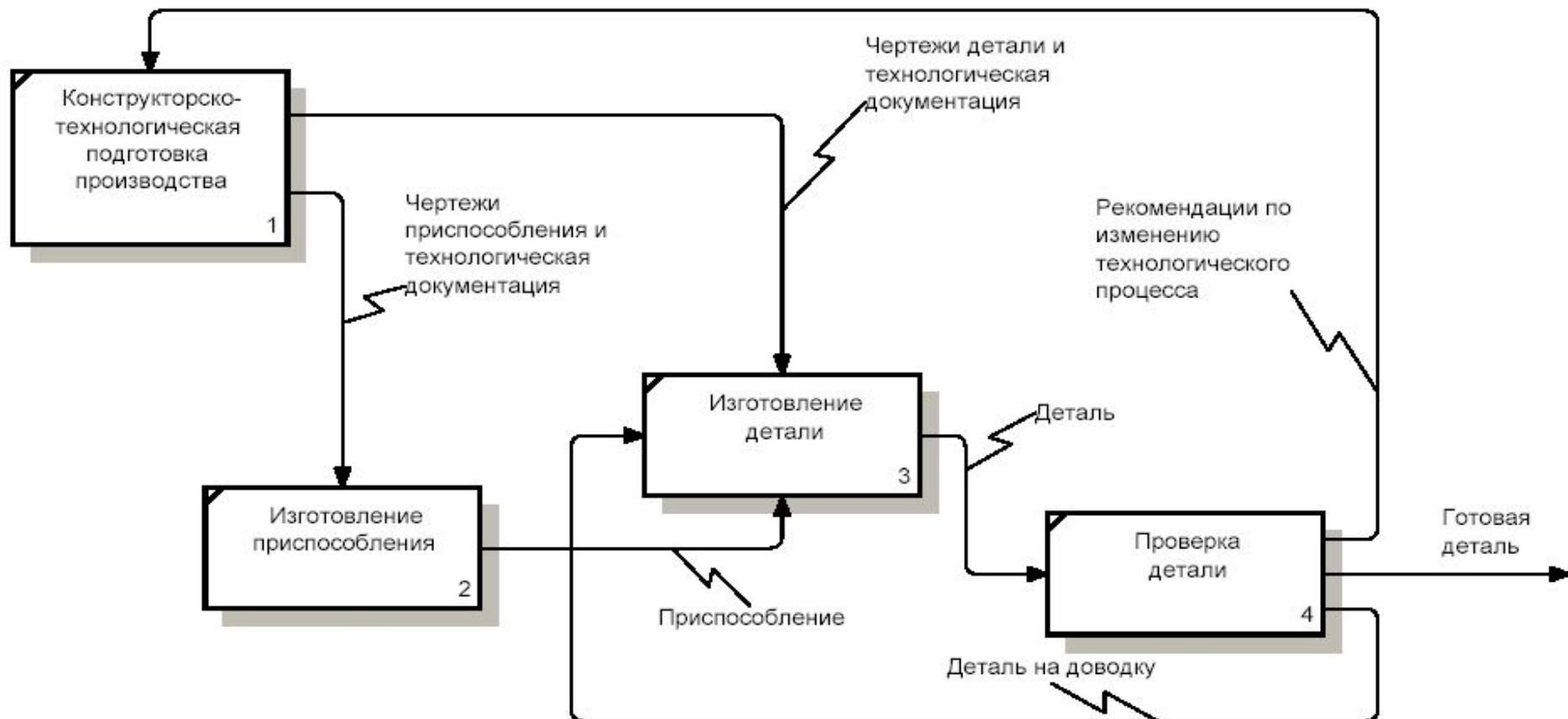
- Одна функция выбирает или создает ресурс, который будет являться исполнительным механизмом другой функции.



Выход-механизм

# Обратная связь по входу

- Выход функции направляется на вход предыдущей.
- Используется для описания возможности повторной обработки потока объектов или для описания циклических действий над потоком.

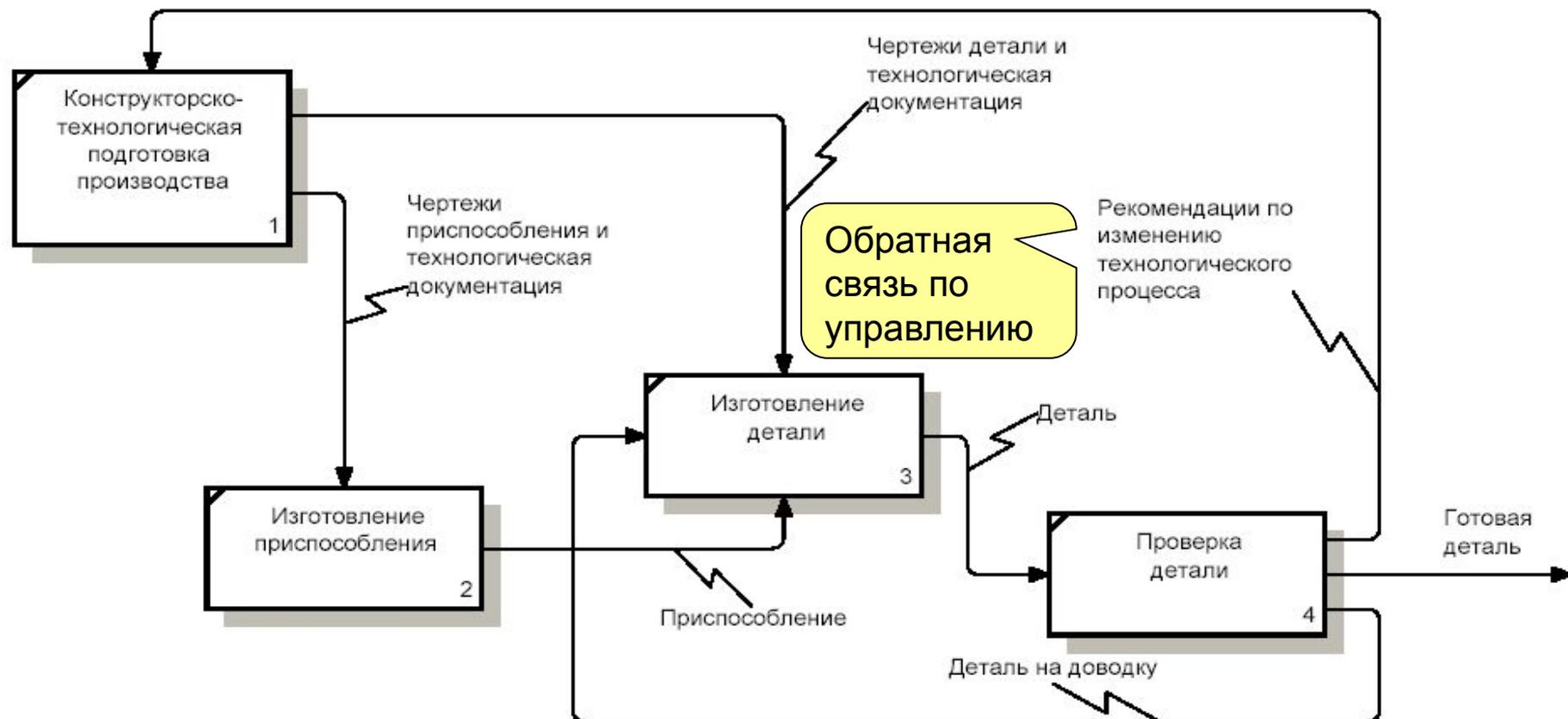


Обратная связь по входу

# Обратная связь по

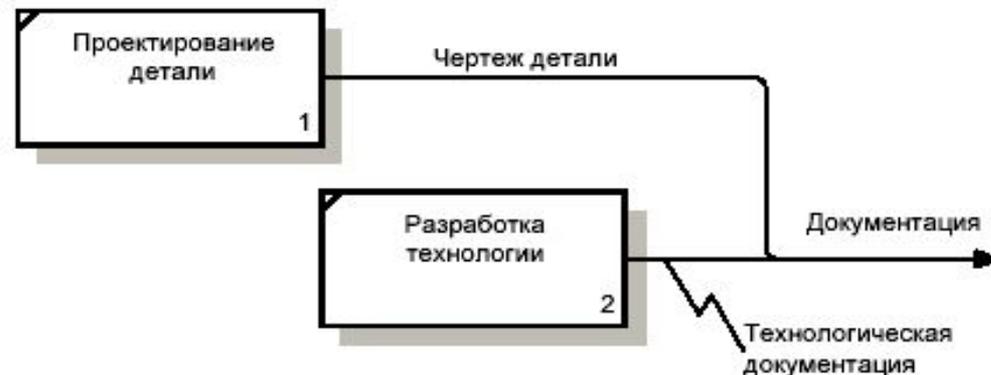
## управлению

Входные функции могут перейти на управление одной из предыдущих функций. Например, она может описывать обратную связь, регулирующую производственные процессы, что может повлиять на качество выпускаемого изделия.



# Слияние стрелок

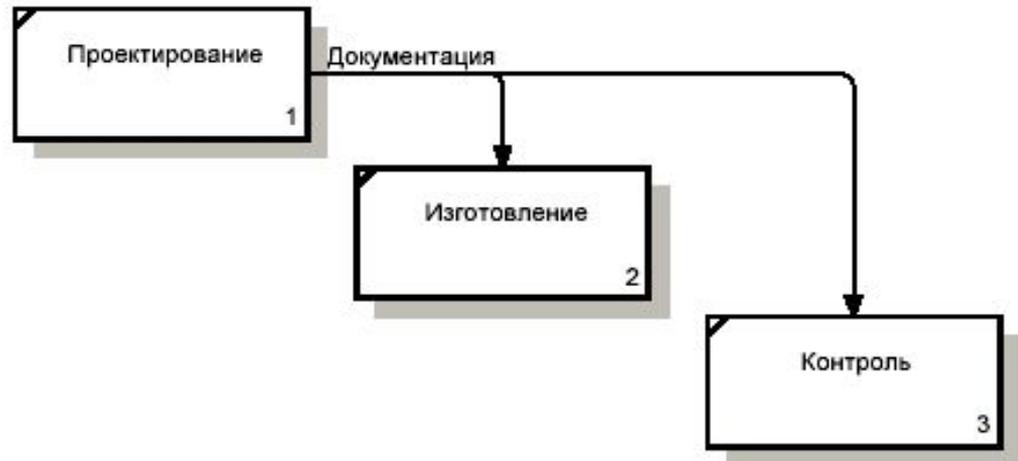
- Функция производит объекты, которые используются в нескольких других функциях.
- Объекты, полученные в результате работы нескольких функций, объединяются в один общий поток.



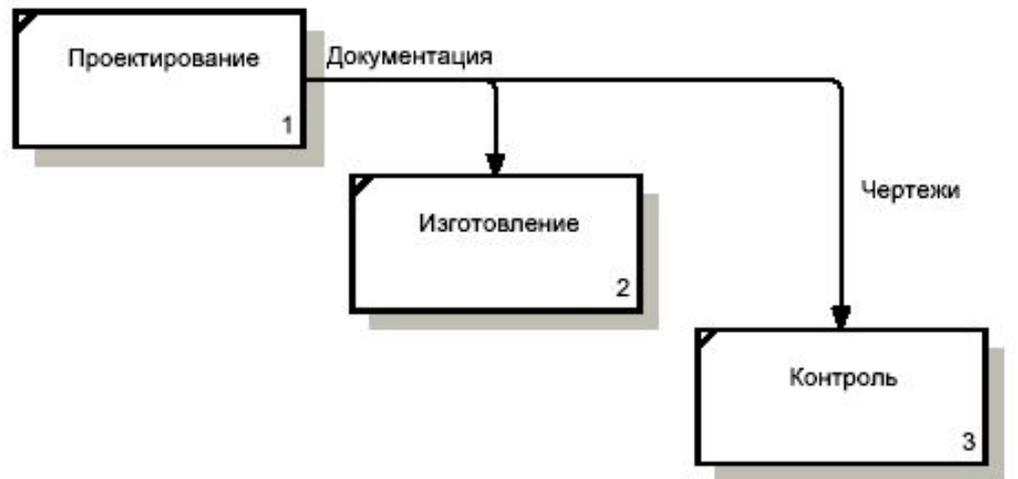
- Случай когда какой либо **однотипный результат** получается от двух различных функций. Достаточно отметить только **общую** часть стрелки.
- Два **различных** выхода сливаются в один общий. Должны быть отмечены **каждая ветвь** и **общий участок** связи.

# Разветвление

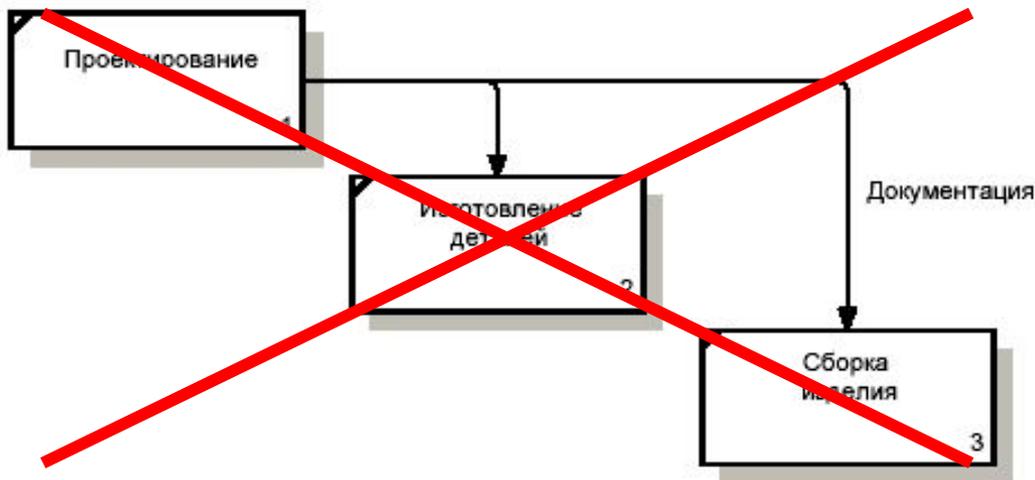
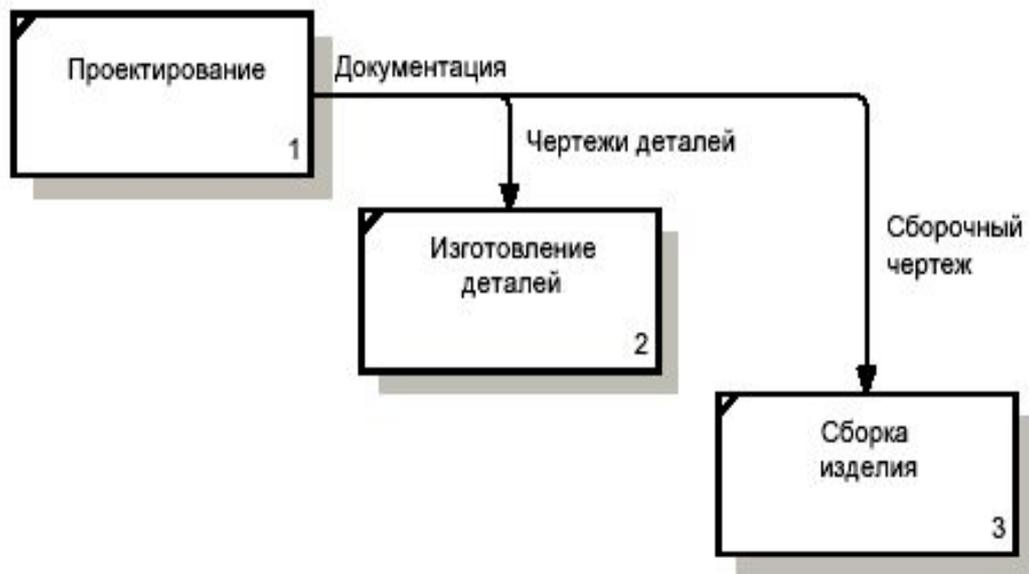
- Поток разветвляясь сохраняет первоначальное содержание.
- Подпись необходима только для общей части стрелки.



- Поток ответвляется от общего потока, неся в себе часть объектов (чертежи).
- Подписываются общая стрелка и ответвления.
- Если ответвление не подписано, то оно несет в себе общий поток объектов.



# Разветвление



- Разделение общего потока на несколько независимых потоков.
- Обозначается общая часть стрелки и каждое ответвление.

- **Ошибка** - не именованы общая часть стрелки и какая либо из ветвей.

# Тоннельные стрелки

Иногда необходимо отобразить граничные стрелки, которые значимы на данном уровне и не значимы на родительской диаграмме. Например, некоторые данные используются только на данном уровне и не используются на других. Без использования механизма тоннелирования малозначимая стрелка появится на всех уровнях модели, что затруднит чтение диаграмм.



# Глоссарий и FEO-страница

- Для каждого из элементов в IDEF0 существует стандарт, подразумевающий создание и поддержку набора соответствующих определений, ключевых слов, повествований, изложений и т.д, которые характеризуют объект, отраженный данным элементом. Этот набор – **глоссарий**, являющийся описанием сущности данного элемента.
- **FEO-диаграмма** (*For Exposition Only*) – это диаграмма, которая поясняет особо интересные и тонкие аспекты диаграмм. Эти диаграммы не ограничены синтаксисом IDEF0. В них может быть текстовая, графическая информация, схемы, альтернативная точка зрения на процесс и т.п.

# Мастерская страница (каркас диаграммы)

- Стандартный бланк для диаграмм (облегчает подшивку и копирование)
- Разделен на 3 основные части:
  - 1) **поле рабочей информации** (для отслеживания диаграммы в процессе моделирования)
  - 2) **поле сообщений** (область рисования диаграммы)
  - 3) **поле идентификации** (идентификация диаграммы и ее позиционирование в иерархии)

# Мастерская страница

USED AT:	AUTHOR: FIO PROJECT: model1	DATE: 27.02.2009 REV: 27.02.2009	WORKING	READER	DATE	CONTEXT:
			DRAFT			TOP
			RECOMMENDED			
			PUBLICATION			

## Поле рабочей информации

### Сведения о модели:

- автор;
- название проекта;
- замечания;
- дата создания и пересмотра.

## Статусы проекта:

- 1) *Рабочая версия* – диаграмма с большим числом изменений на стадии разработки
- 2) *Эскиз* имеет меньше изменений и свидетельствует о достижении некоторого согласия ряда читателей
- 3) *Рекомендовано* – сопутствующие тексты утверждены
- 4) *Публикация* – материал может печататься.

## Поле сообщений

Номер диаграммы

## Поле идентификации

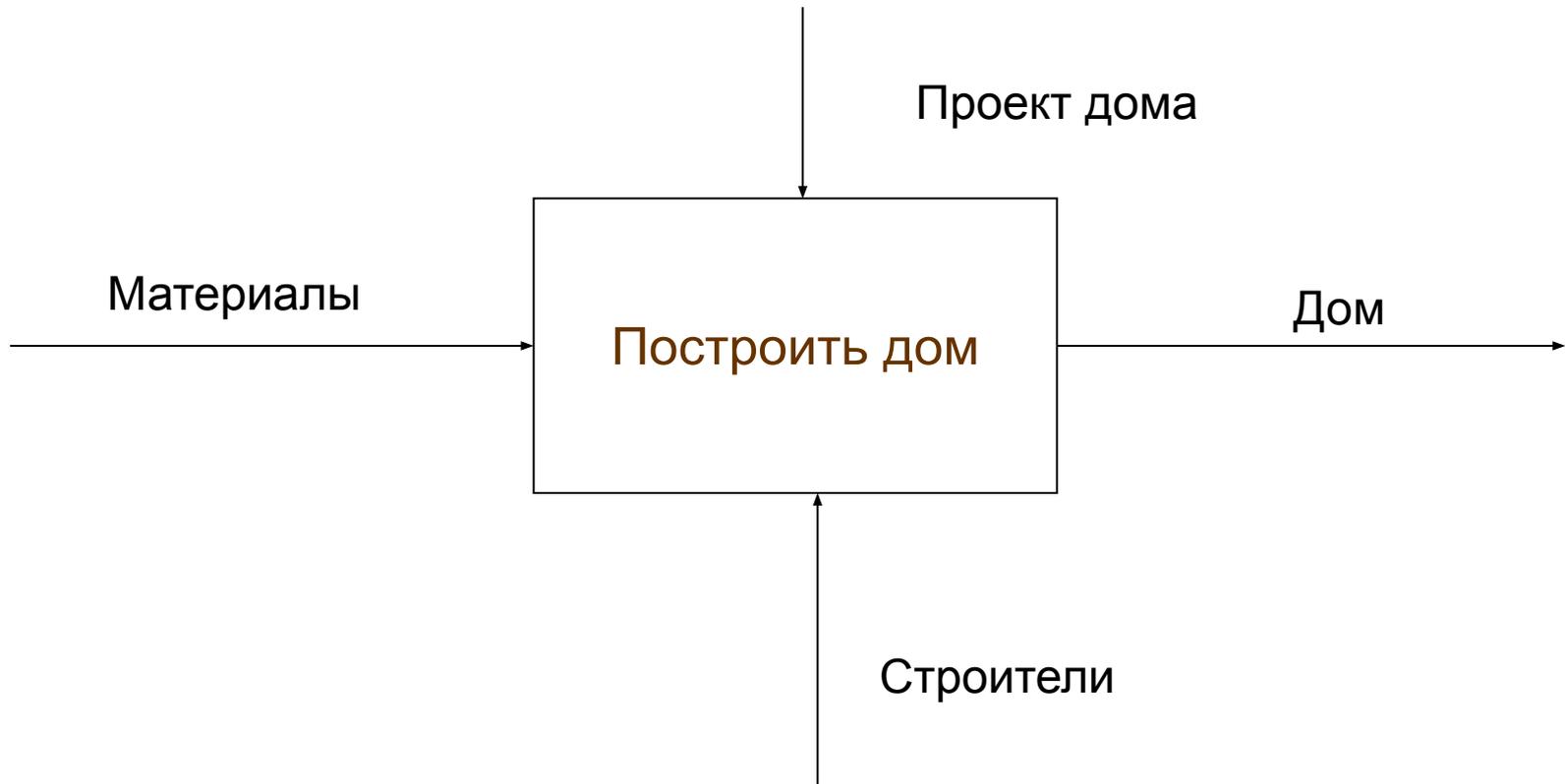
Название диаграммы  
(совпадает с названием  
родительской работы)

Уникальный  
номер версии  
диаграммы

NODE: A-0	TITLE:	NUMBER:
--------------	--------	---------

# Пример модели процесса постройки садового домика

1. Строим контекстную диаграмму.

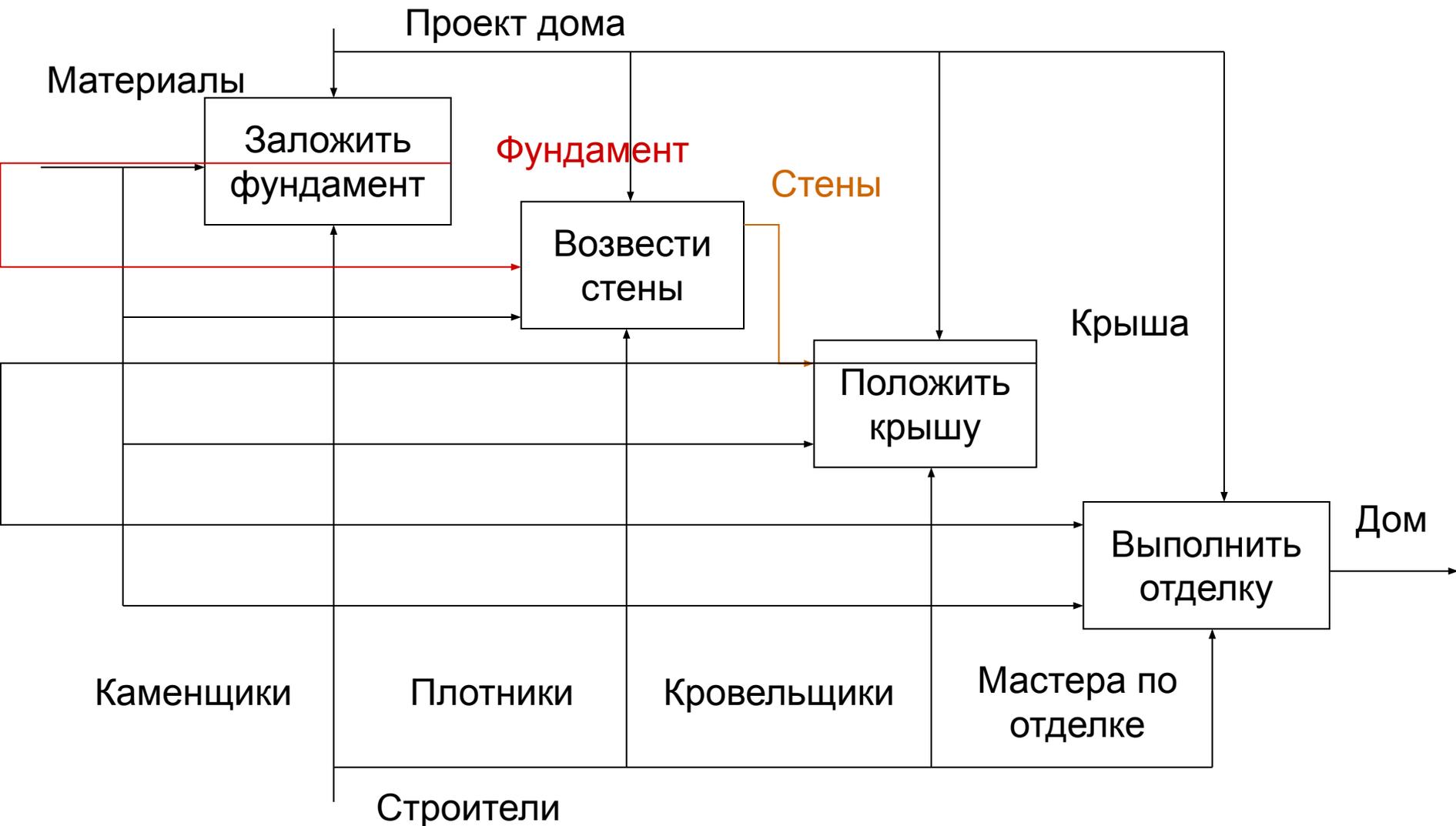


**Цель:** Определить действия, необходимые для постройки дачного домика

**Точка зрения:** владельца дачного участка

# Пример модели процесса постройки садового домика

## 2. Декомпозируем контекстную диаграмму





# Пример модели, построенной с использованием CASE-средства VPWin

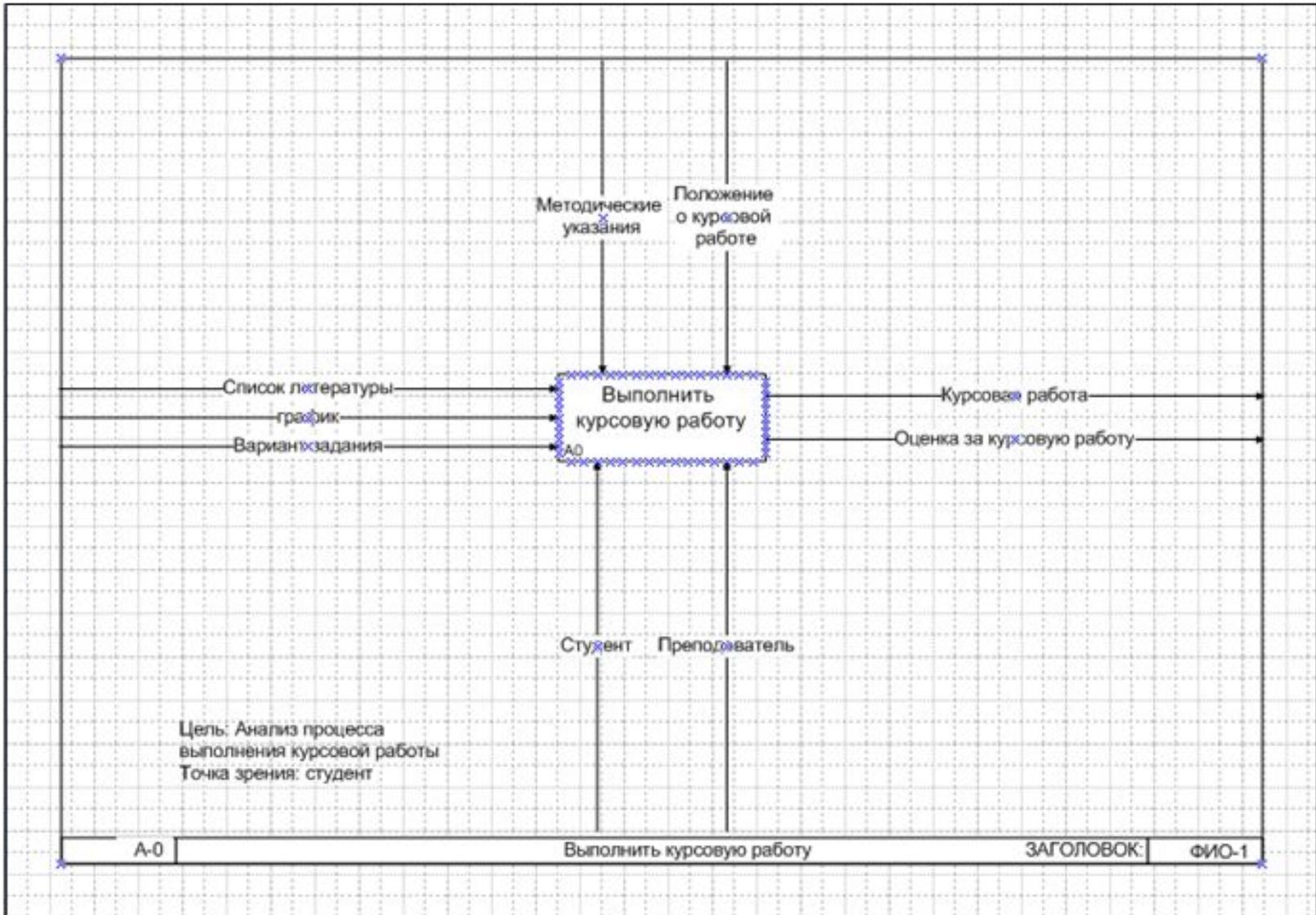


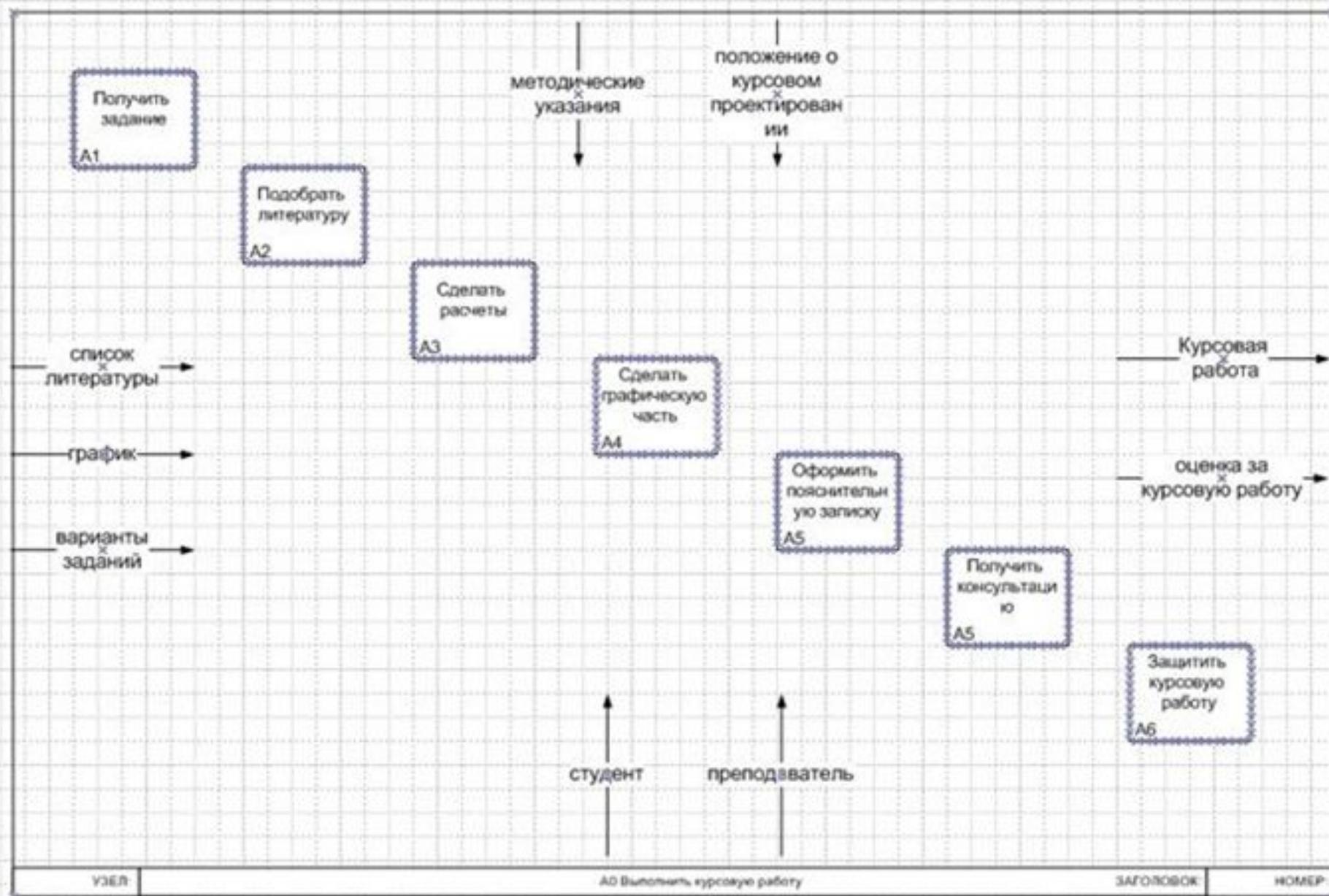
# Пример модели, построенной с использованием CASE-средства VPWin

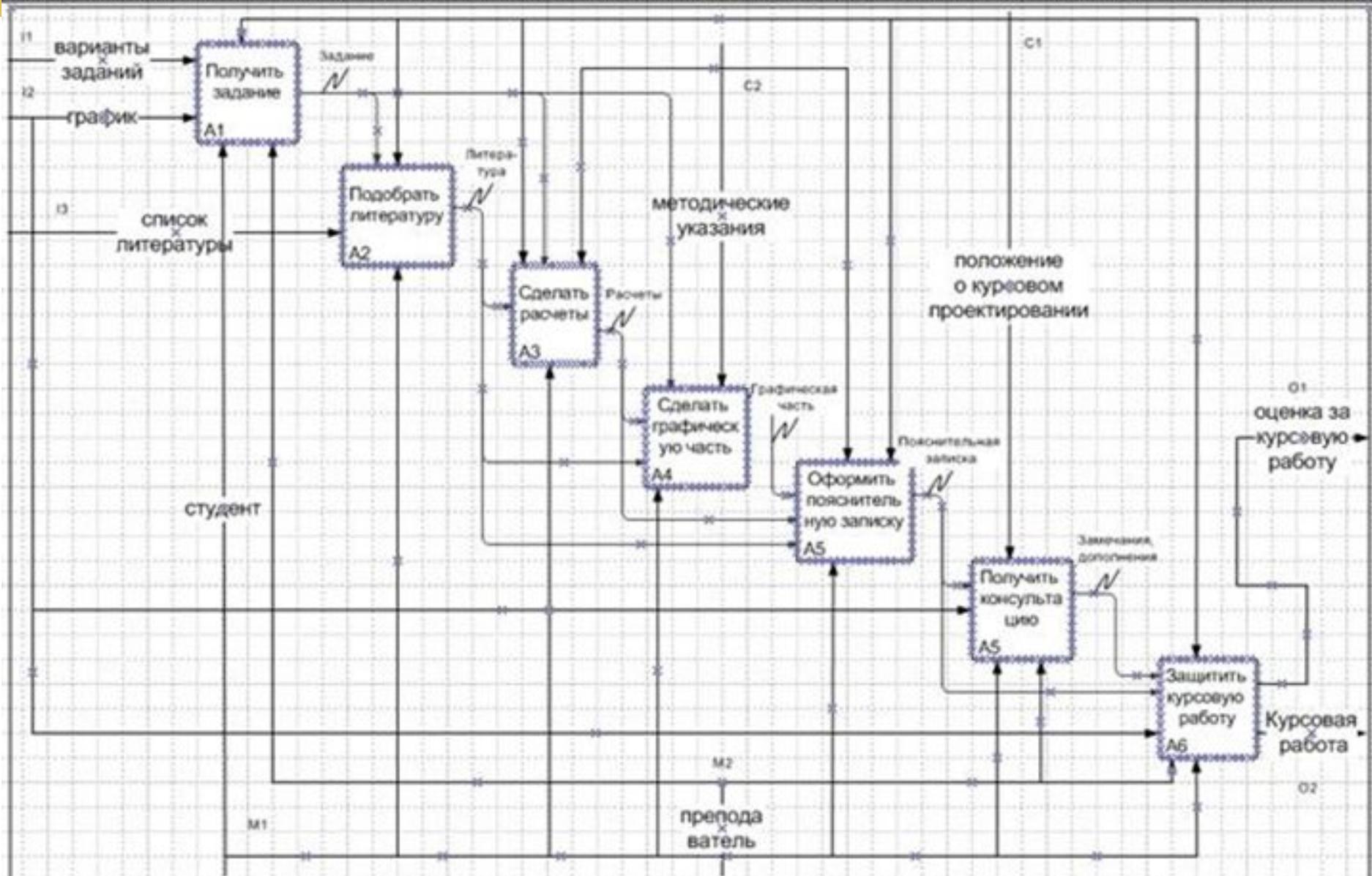


# Дерево узлов









УЗЕЛ	А0 Выполнить курсовую работу	ЗАГОЛОВОК	НОМЕР
------	------------------------------	-----------	-------

# Итоги лекции

Изучены следующие понятия:

- Структурный подход
- Функциональная модель
- Методология SADT/IDEF0
- Функциональный блок
- Интерфейсная дуга
- Декомпозиция
- Глоссарий
- FEO-диаграмма
- Дерево узлов
- Мастерская страница