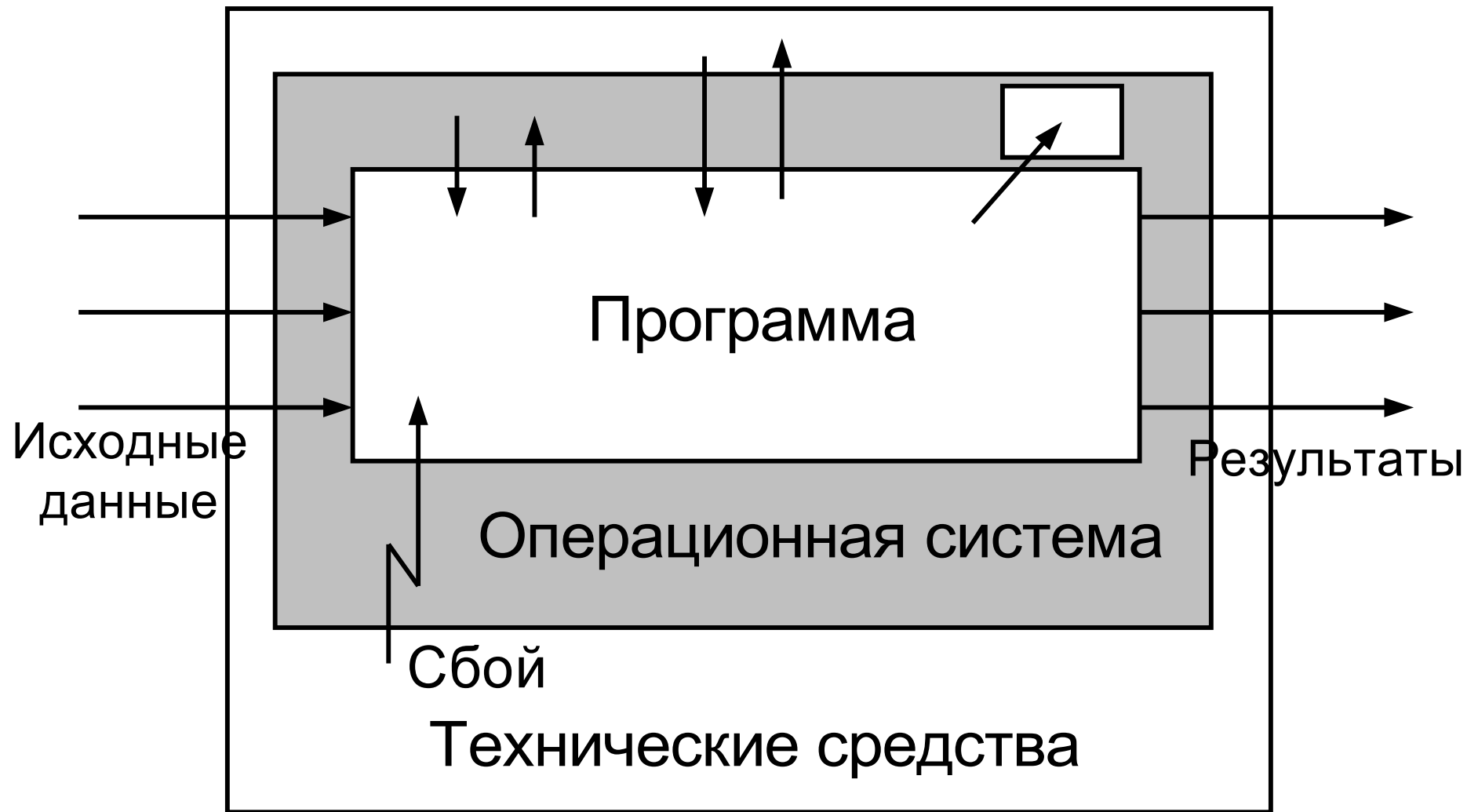


# Технологии программирования

## Лекция 3

**Постановка задачи. Анализ и проектирование при структурном подходе к программированию**

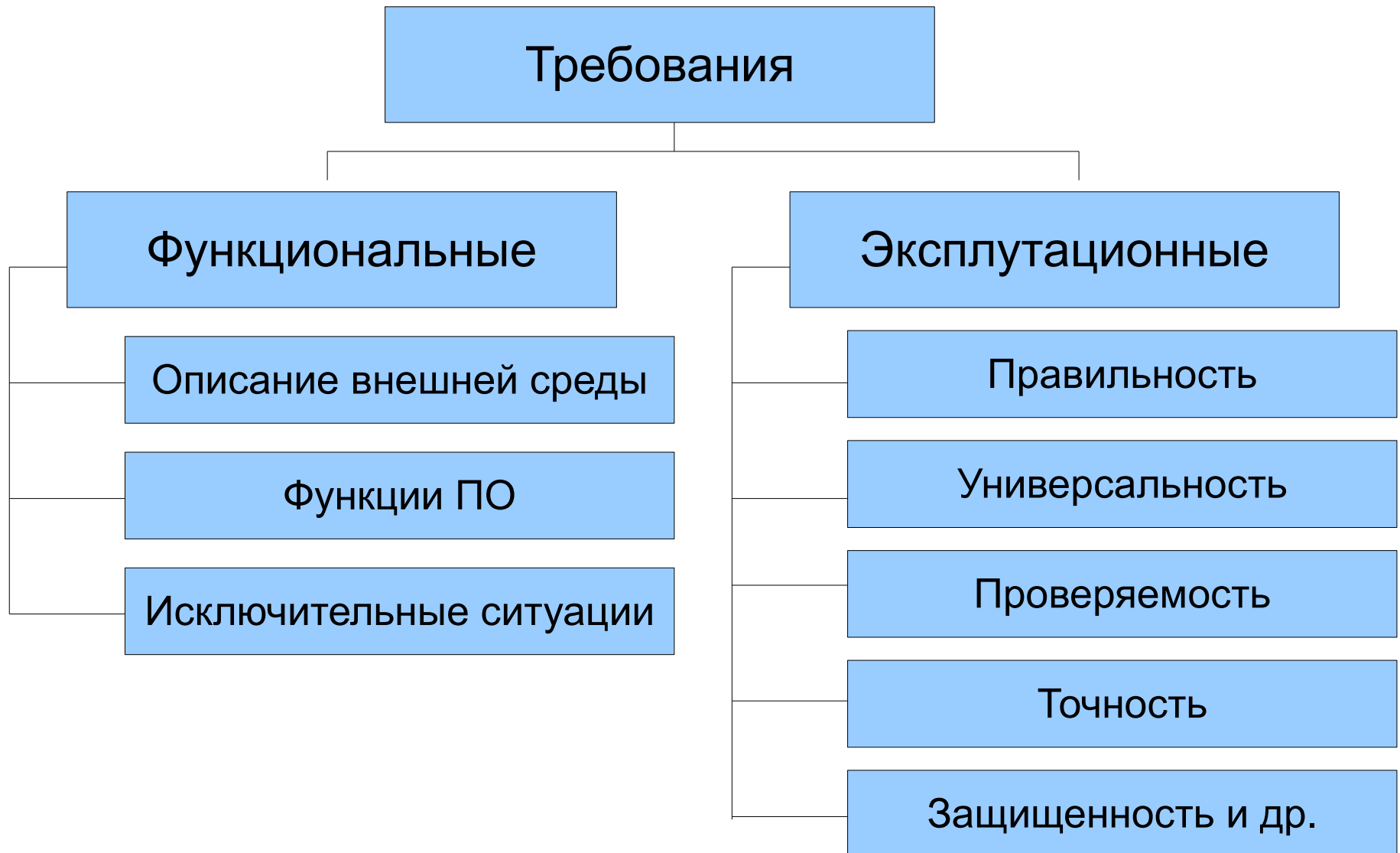
# Факторы, определяющие требования к ПО



# Содержание Технического Задания

- техническое задание должно содержать следующие разделы (ГОСТ 19.201-78):
- введение;
- основания для разработки;
- назначение разработки;
- требования к программе или программному изделию;
- требования к программной документации;
- технико-экономические показатели;
- стадии и этапы разработки;
- порядок контроля и приемки.

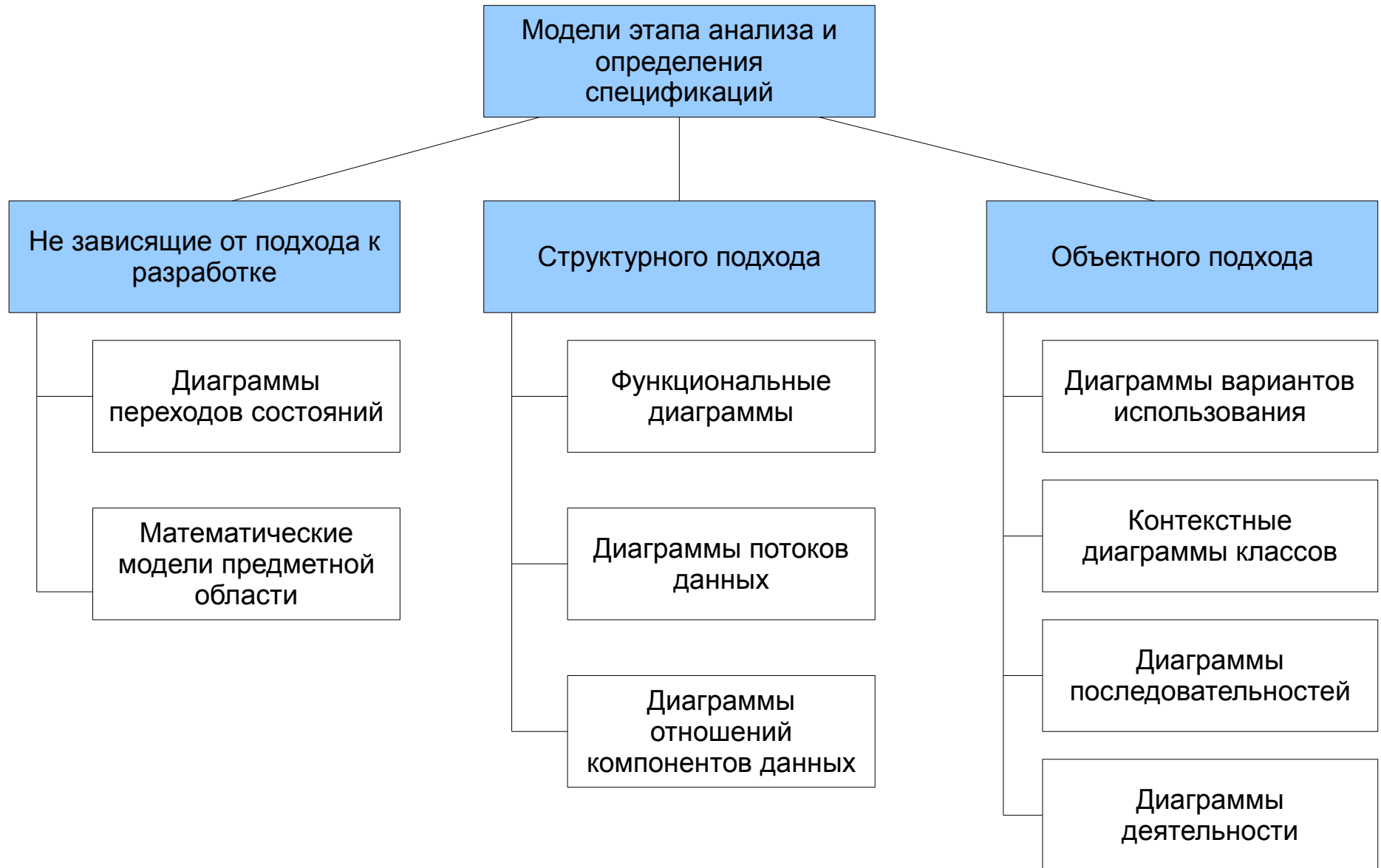
# Требования к программному продукту



# Требования к спецификациям

- требование *полноты* означает, что спецификации должны содержать всю существенную информацию, где ничего важного не было бы упущено, и отсутствует несущественная информация, например детали реализации, чтобы не препятствовать разработчику в выборе наиболее эффективных решений;
- требование *точности* означает, что спецификации должны однозначно восприниматься как заказчиком, так и разработчиком.

# Классификация моделей для разработки спецификаций



# Модели методологии структурного анализа и проектирования

- диаграмм потоков данных (DFD - Data Flow Diagrams), описывающих взаимодействие источников и потребителей информации через процессы, которые должны быть реализованы в системе;
- диаграмм «сущность-связь» (ERD - Entity-Relationship Diagrams), описывающих базы данных разрабатываемой системы;
- диаграмм переходов состояний (STD - State Transition Diagrams), характеризующих поведение системы во времени;
- спецификаций процессов;
- словаря терминов.

## **Спецификации процессов.**

Спецификации процессов обычно представляют в виде краткого текстового описания, схем алгоритмов, псевдокодов, Flow-форм или диаграмм Насси-Шнейдермана. Поскольку описание процесса должно быть кратким и понятным как разработчику, так и заказчику, для их спецификации чаще всего используют псевдокоды.

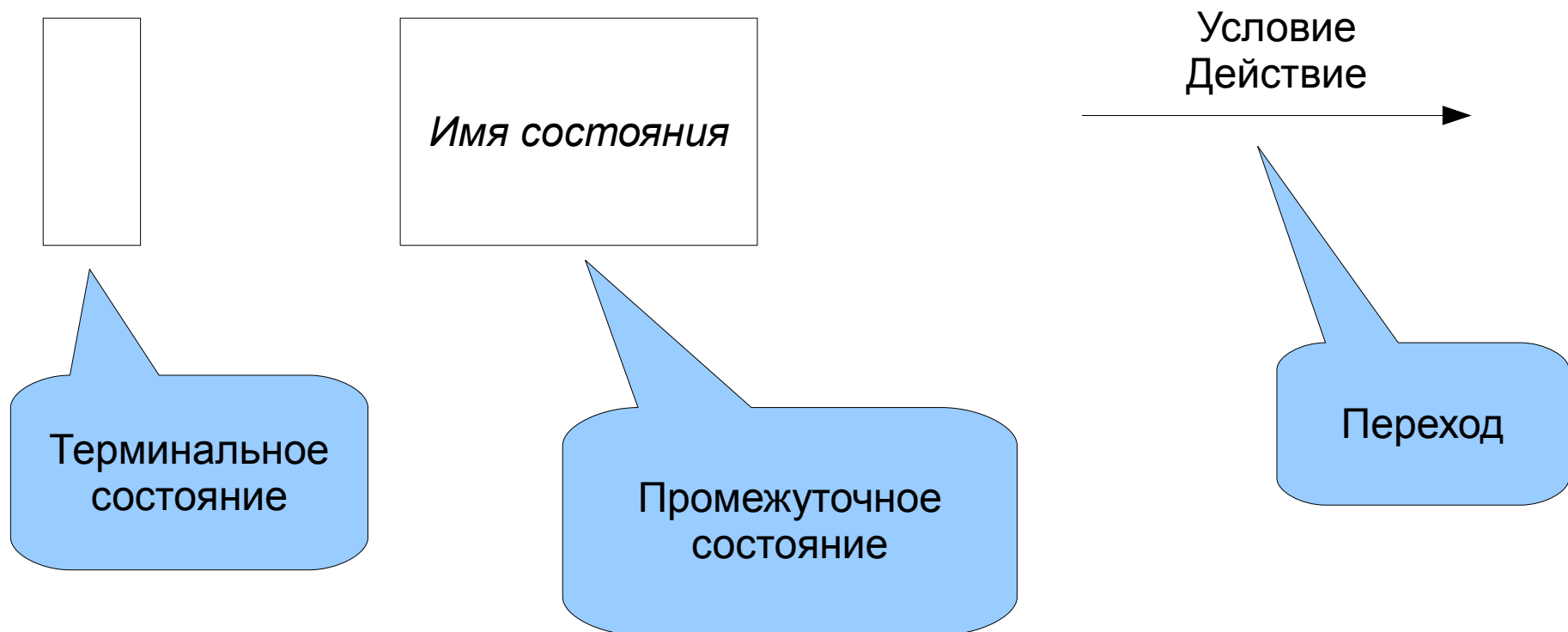
## **Словарь терминов.**

Словарь терминов представляет собой краткое описание основных понятий, используемых при составлении спецификаций. Он должен включать определение основных понятий предметной области, описание структур элементов данных, их типов и форматов, а также всех сокращений и условных обозначений. Он предназначен для повышения степени понимания предметной области и исключения риска возникновения разногласий при обсуждении моделей между заказчиками и разработчиками.

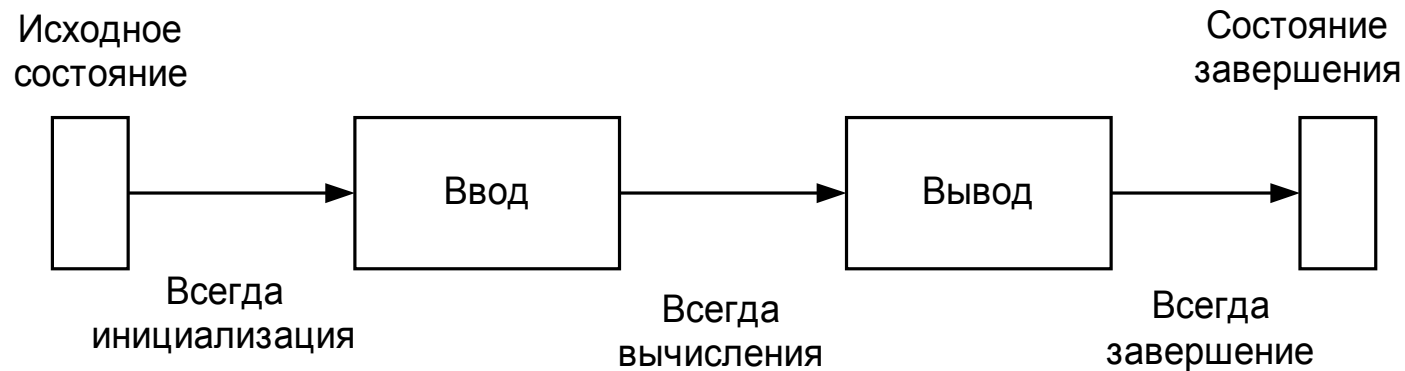


# Диаграммы переходов состояний

• *Диаграмма переходов состояний* является графической формой предоставления *конечного автомата* - математической абстракции, используемой для моделирования детерминированного поведения технических объектов или объектов реального мира.



Пример: Диаграмма переходов состояний программного обеспечения, активно не взаимодействующего с окружающей средой.



Пример: Диаграмма переходов состояний программы построения графиков/таблиц функций



# Диаграммы потоков данных

*Диаграммы потоков данных* позволяют специфицировать как функции разрабатываемого программного обеспечения, так и обрабатываемые им данные.

В основе модели лежат понятия:

- *Внешняя сущность*
- *Процесс*
- *Хранилище данных*
- *Поток данных*

# Нотации диаграммы потоков данных

Понятие

Нотация Йордана

Нотация Гейна-Сарсона

Внешняя сущность

Наименование

Наименование

Система,  
подсистема или  
процесс

Наименование  
Номер

Номер

Наименование

Механизм

Внешняя сущность

Наименование

№

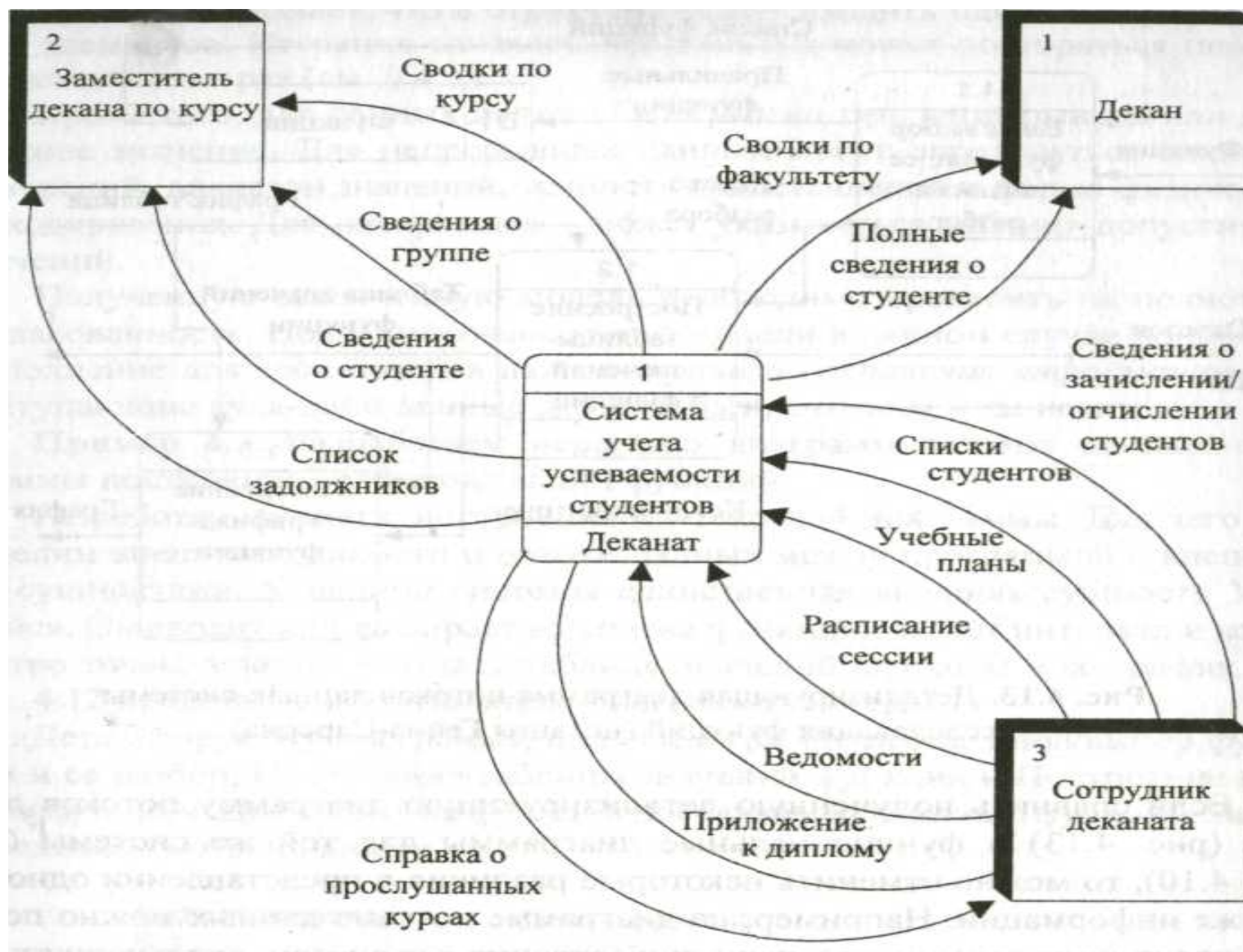
Наименование

Поток

Наименование

Наименование

# Пример контекстной диаграммы



# Детализация

На каждом следующем этапе каждую подсистему контекстной диаграммы детализируют при помощи диаграмм потоков данных.

Решение о завершении детализации процесса принимают в следующих случаях:

- процесс взаимодействует с 2-3-мя потоками данных;
- возможно описание процесса последовательным алгоритмом;
- процесс выполняет единственную логическую функцию преобразования входной информации в выходную.

# Этапы разработки модели

*1 этап* - **построение контекстной диаграммы** - включает выполнение следующих действий:

- классификацию множества требований и организацию их в основные функциональные группы - процессы;
- идентификацию внешних объектов - внешних сущностей, с которыми система должна быть связана;
- идентификацию основных видов информации - потоков данных, циркулирующей между системой и внешними объектами;
- предварительную разработку контекстной диаграммы;
- изучение предварительной контекстной диаграммы и внесение в нее изменений по результатам ответов на возникающие при изучении вопросы по всем ее частям;
- построение контекстной диаграммы путем объединения всех процессов предварительной диаграммы в один процесс, а также группирования потоков.



# Этапы разработки модели

*2 этап* - формирование иерархии диаграмм потоков данных – включает для каждого уровня:

- проверку и изучение основных требований по диаграмме соответствующего уровня (для первого уровня - по контекстной диаграмме);
- декомпозицию каждого процесса текущей диаграммы потоков данных с помощью детализирующей диаграммы или - если некоторую функцию сложно или невозможно выразить комбинацией процессов, построение спецификации процесса;
- добавление определений новых потоков в словарь данных при каждом появлении их на диаграмме;
- проведение ревизии с целью проверки корректности и улучшения наглядности модели после построения двух-трех уровней.

# Пример детализирующей диаграммы

