



# **МОДЕЛИРОВАНИЕ И ПРЕДСТАВЛЕНИЕ СТРУКТУРЫ ПРЕДПРИЯТИЯ.**

**Диаграммы IDEF0.**

## **МОДЕЛЬ ПРЕДМЕТНОЙ ОБЛАСТИ -**

некоторая система, имитирующая структуру или функционирование исследуемой предметной области и отвечающая основному требованию — быть адекватной этой области.



# ТРЕБОВАНИЯ

## К МОДЕЛЯМ ПРЕДМЕТНЫХ ОБЛАСТЕЙ

1) формализация;

2) понятность для заказчиков и разработчиков;

3) реализуемость;

4) обеспечение оценки  
эффективности реализации  
модели.



## ИСТОРИЯ ВОЗНИКНОВЕНИЯ СТАНДАРТА **IDEF0**

**IDEF0**, как стандарт был разработан в 1981 году в рамках программы автоматизации промышленных предприятий, которая носила обозначение **ICAM** (Integrated Computer Aided Manufacturing) и была предложена департаментом Военно-Воздушных Сил США.

Семейство стандартов **IDEF** унаследовало свое обозначение от названия этой программы (**IDEF=ICAM DEFinition**).



## ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ **IDEF**

- Обеспечение групповой работы над созданием модели с участием всех аналитиков и специалистов, занятых в рамках проекта.
- Последняя редакция **IDEF** была выпущена в декабре 1993 года Национальным Институтом По Стандартам и Технологиям США (NIST).



## ОСНОВНЫЕ ЭЛЕМЕНТЫ И ПОНЯТИЯ **IDEFO**

В основе методологии лежат **четыре** основных понятия:

- 1. функциональный блок (Activity Box);**
- 2. интерфейсная дуга (Arrow);**
- 3. декомпозиция (Decomposition);**
- 4. глоссарий (Glossary).**



## ФУНКЦИОНАЛЬНЫЙ БЛОК

- графически изображается в виде прямоугольника;
- олицетворяет собой некоторую конкретную функцию в рамках рассматриваемой системы;
- название каждого функционального блока должно быть сформулировано в глагольном наклонении (например, “производить услуги”, а не “производство услуг”).



КАЖДАЯ ИЗ ЧЕТЫРЕХ СТОРОН ФУНКЦИОНАЛЬНОГО БЛОКА  
ИМЕЕТ СВОЁ ОПРЕДЕЛЕННОЕ ЗНАЧЕНИЕ

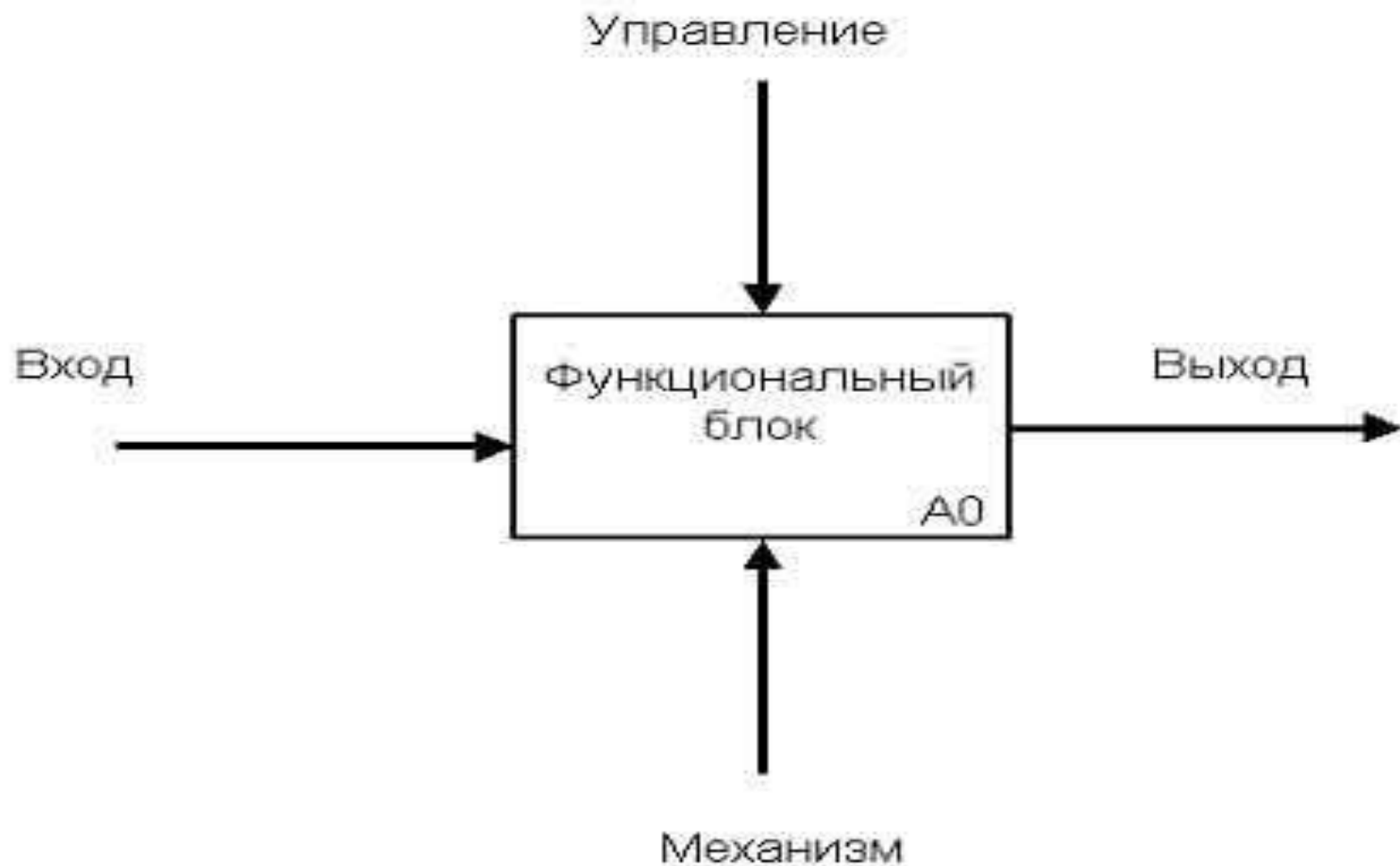
- 1) Верхняя сторона имеет значение “Управление” (*Control*);
- 2) Левая сторона имеет значение “Вход” (*Input*);
- 3) Правая сторона имеет значение “Выход” (*Output*);
- 4) Нижняя сторона имеет значение “Механизм” (*Mechanism*).

Каждый функциональный блок в рамках единой рассматриваемой системы должен иметь свой уникальный идентификационный номер





# ГРАФИЧЕСКОЕ ИЗОБРАЖЕНИЕ ФУНКЦИОНАЛЬНОГО БЛОКА



# ИНТЕРФЕЙСНАЯ ДУГА

- Отображает элемент системы, который обрабатывается функциональным блоком или оказывает иное влияние на функцию.
- Графическим отображением интерфейсной дуги является **однонаправленная стрелка**.
- Каждая интерфейсная дуга должна иметь свое **уникальное наименование**, которое должно быть оборотом существительного.
- С помощью интерфейсных дуг отображают различные объекты, определяющие **процессы**, происходящие в системе.



# ПРАВИЛА ПОСТРОЕНИЯ ИНТЕРФЕЙСНЫХ ДУГ

## Правило 1

- В зависимости от того, к какой из сторон подходит данная интерфейсная дуга, она носит название “входящей”, “исходящей” или “управляющей”.



## Правило 2

- источником” (началом) и “приемником” (концом) каждой дуги могут быть только функциональные блоки, при этом “источник” - только выходная сторона блока, “приемник” - любая из трех оставшихся.



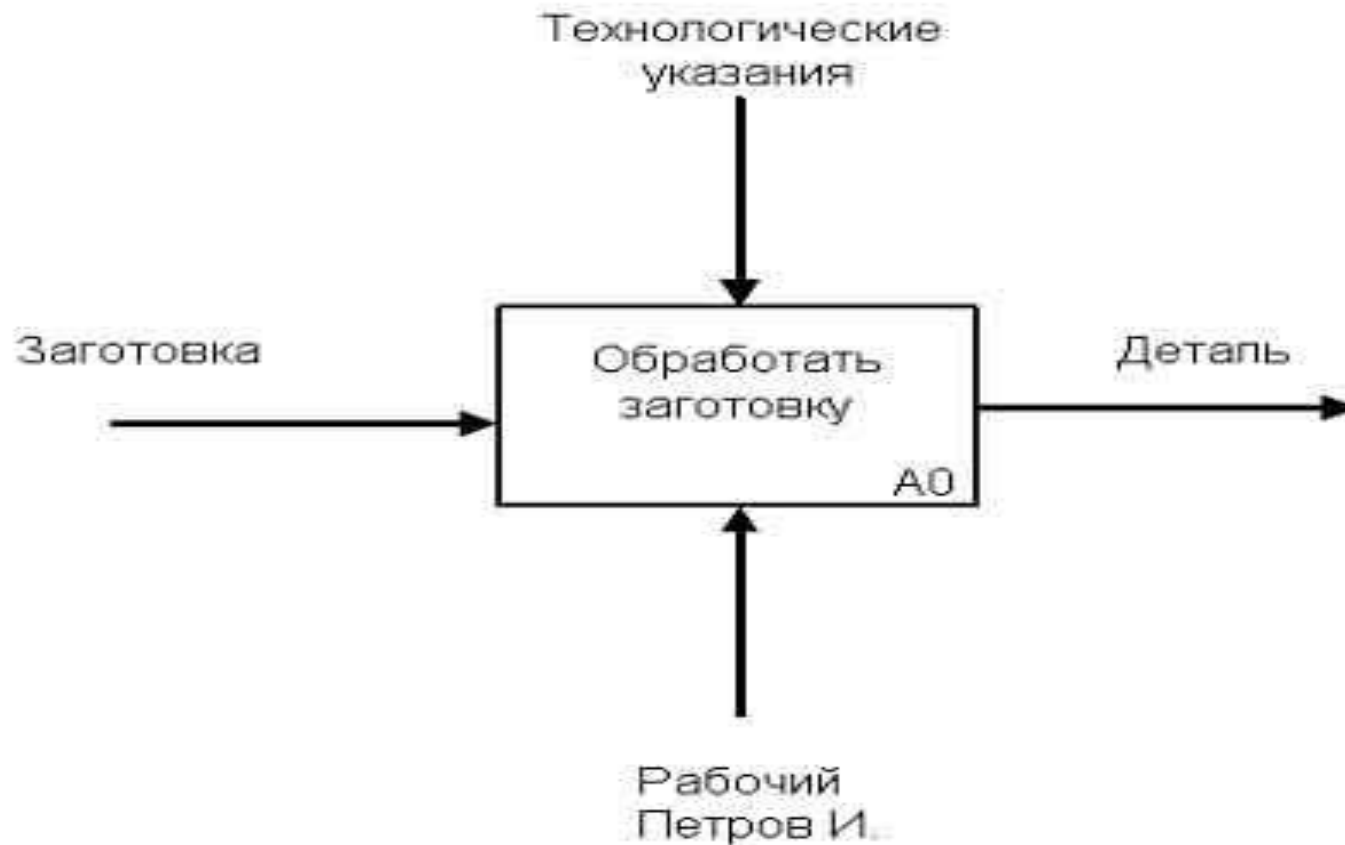
## Правило 3

- любой функциональный блок должен иметь, по крайней мере, одну управляющую интерфейсную дугу и одну исходящую.



## ПРИМЕР 1

При построении IDEF0 – диаграмм важно правильно отделять входящие интерфейсные дуги от управляющих, что часто бывает непросто.



## ПРИМЕР 2



# ДЕКОМПОЗИЦИЯ

- Принцип декомпозиции применяется при разбиении сложного процесса на составляющие его функции.
- Уровень детализации процесса определяется непосредственно разработчиком модели.
- Декомпозиция позволяет постепенно и структурировано представлять модель системы в виде иерархической структуры отдельных диаграмм.





# КОНТЕКСТНАЯ ДИАГРАММА

- Модель IDEF0 всегда начинается с представления системы как единого целого — одного функционального блока с интерфейсными дугами, простирающимися за пределы рассматриваемой области.
- Такая диаграмма с одним функциональным блоком называется контекстной диаграммой, и обозначается идентификатором “А-0”.



## ЦЕЛЬ И ТОЧКА ЗРЕНИЯ

В пояснительном тексте к контекстной диаграмме должна быть указана **цель** (Purpose) построения диаграммы в виде краткого описания и зафиксирована **точка зрения** (Viewpoint).

**Цель** определяет соответствующие области в исследуемой системе, на которых необходимо фокусироваться в первую очередь.

**Точка зрения** определяет основное направление развития модели и уровень необходимой детализации



## ПРОЦЕСС ДЕКОМПОЗИЦИИ (1)

- Функциональный блок, отображающий систему как единое целое, подвергается детализации на другой диаграмме.
- Диаграмма второго уровня содержит функциональные блоки, отображающие подфункции контекстной диаграммы, и называется дочерней.

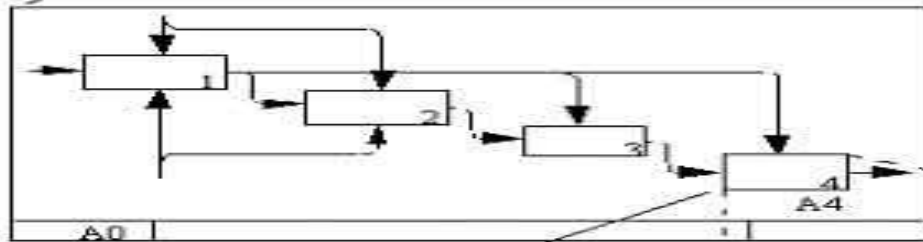
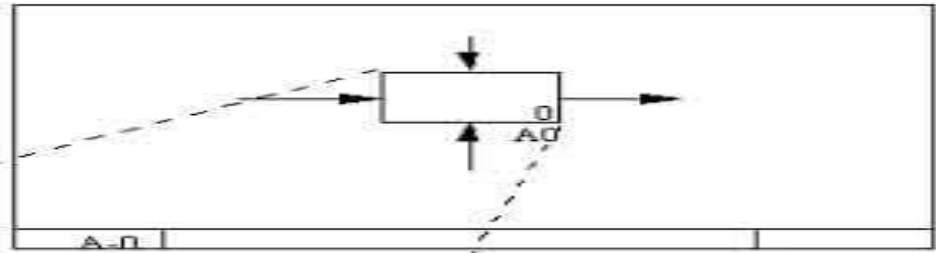


## ПРОЦЕСС ДЕКОМПОЗИЦИИ (2)

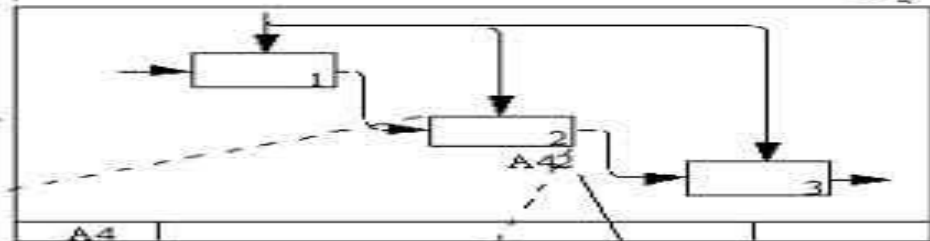
- Каждая из подфункций дочерней диаграммы может быть далее детализирована путем аналогичной декомпозиции соответствующего ей функционального блока.
- В каждом случае декомпозиции функционального блока все интерфейсные дуги, входящие в данный блок, или исходящие из него фиксируются на дочерней диаграмме. Этим достигается структурная целостность IDEF0 – модели



# ПРИМЕР ДЕКОМПОЗИЦИИ



Этот функциональный блок является родительским, по отношению к диаграмме A4

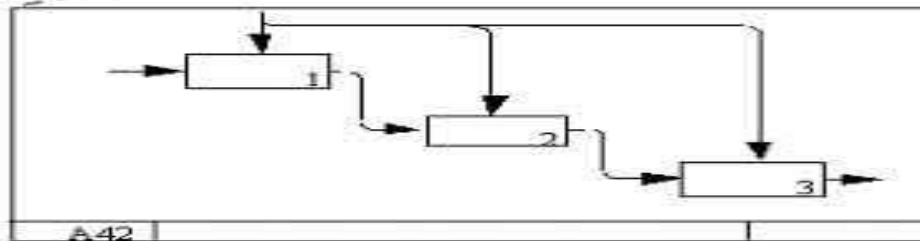


Номер A42 означает, что данный функциональный блок имеет декомпозицию - дочернюю диаграмму с номером A42.

Более обобщенно



Более детально



## ГЛОССАРИЙ -

- набор соответствующих определений, ключевых слов, повествовательных изложений, которые характеризуют объект, отображенный данным элементом.
- Является описанием сущности данного элемента.
- Глоссарий гармонично дополняет наглядный графический язык, снабжая диаграммы необходимой дополнительной информацией.

