

# Лабораторная работа №5

## Архитектура ПС

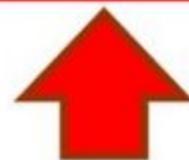
# АРХИТЕКТУРА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

**Архитектура программного обеспечения** — это структура программы или вычислительной системы, которая включает программные компоненты, видимые снаружи свойства этих компонентов, а также отношения между ними.

**Проектирование ПО высокого уровня** (концептуальное проектирование) – **архитектурное проектирование**.  
НЕ РЕШАЕТ ЗАДАЧУ КАКИМ ОБРАЗОМ РЕАЛИЗОВАТЬ ТРЕБУЕМЫЕ ОТ ПО ФУНКЦИИ



**Проектирование ПО низкого уровня** – детальное проектирование. РЕШАЕТ ЗАДАЧУ РЕАЛИЗАЦИИ ФУНКЦИЙ ПО В РАМКАХ СОЗДАННОЙ АРХИТЕКТУРЫ – т.е. в рамках накладываемых ею ограничений.



Архитектура ПО определяет **глобальные ограничения**, накладываемые на проектирование системы, такие как:

- 1) выбор парадигмы программирования
- 2) стандарты разработки ПО,
- 3) принципы проектирования и ограничения, накладываемые государственным законодательством.

# Процессы жизненного цикла ПО

## согласно МС ISO/IEC 12207:95 (ГОСТ Р ИСО/МЭК 12207:99)



# Критерии и примитивы качества ПС согласно МС ISO/IEC 9126

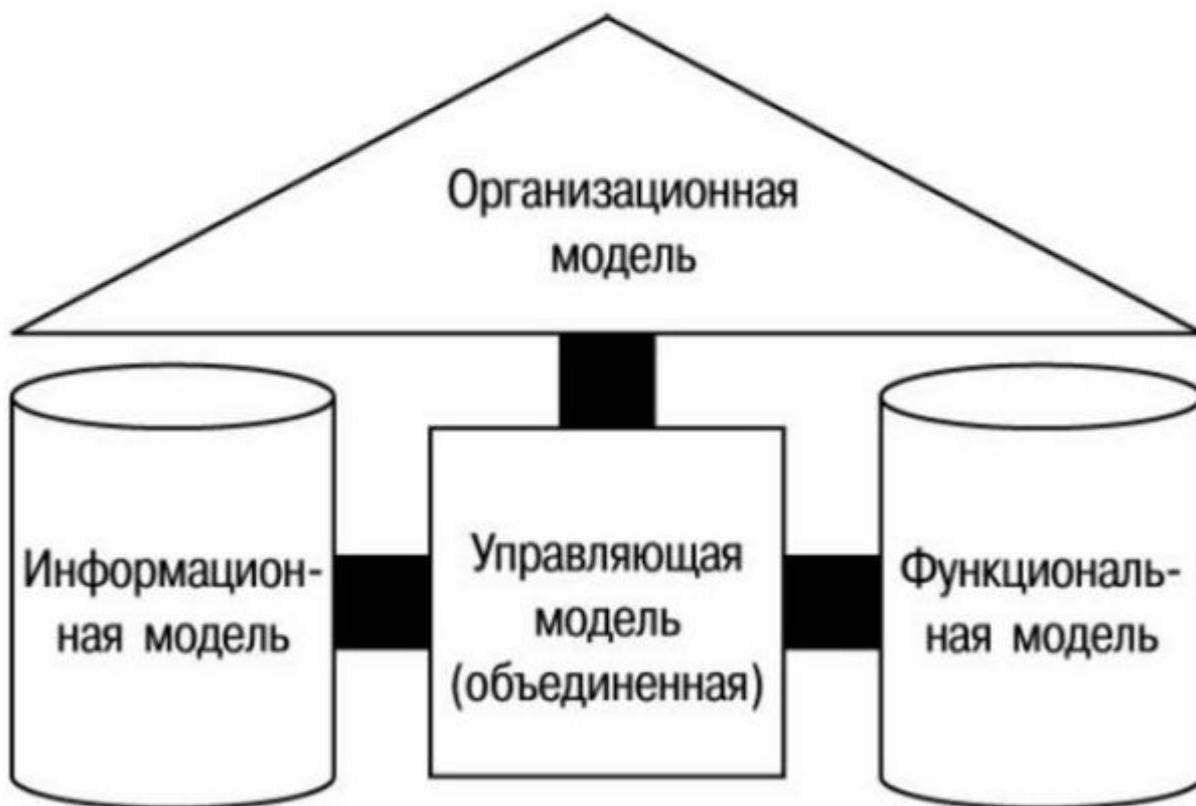


**Разработка архитектуры  
– это способ борьбы со сложностью ПО**

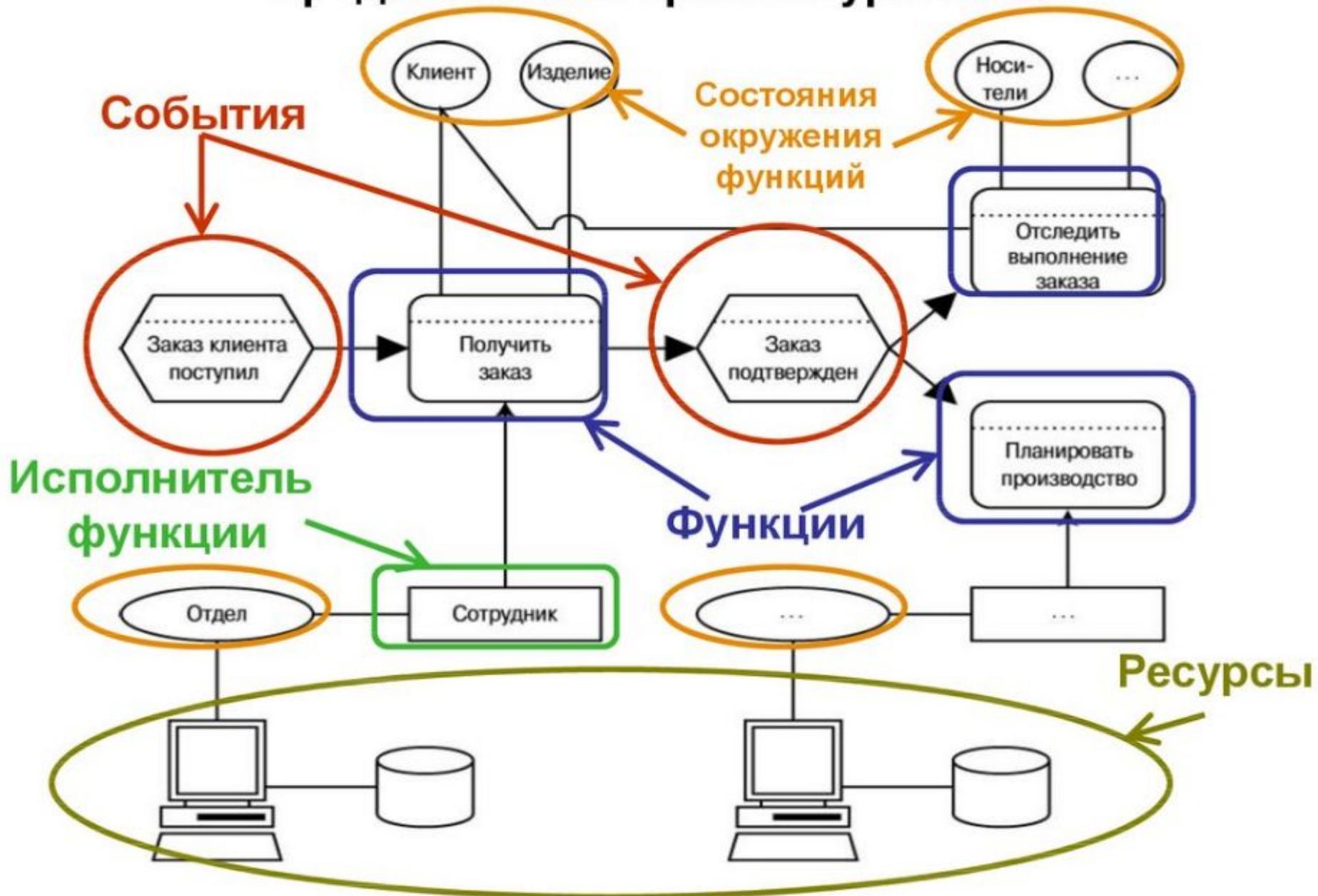
# Представления архитектуры в унифицированном процессе разработки программного обеспечения



**Представления архитектуры интегрированных  
информационных систем  
(ARIS – Architecture of Integrated Information Systems)  
компании IDS Scheer AG**

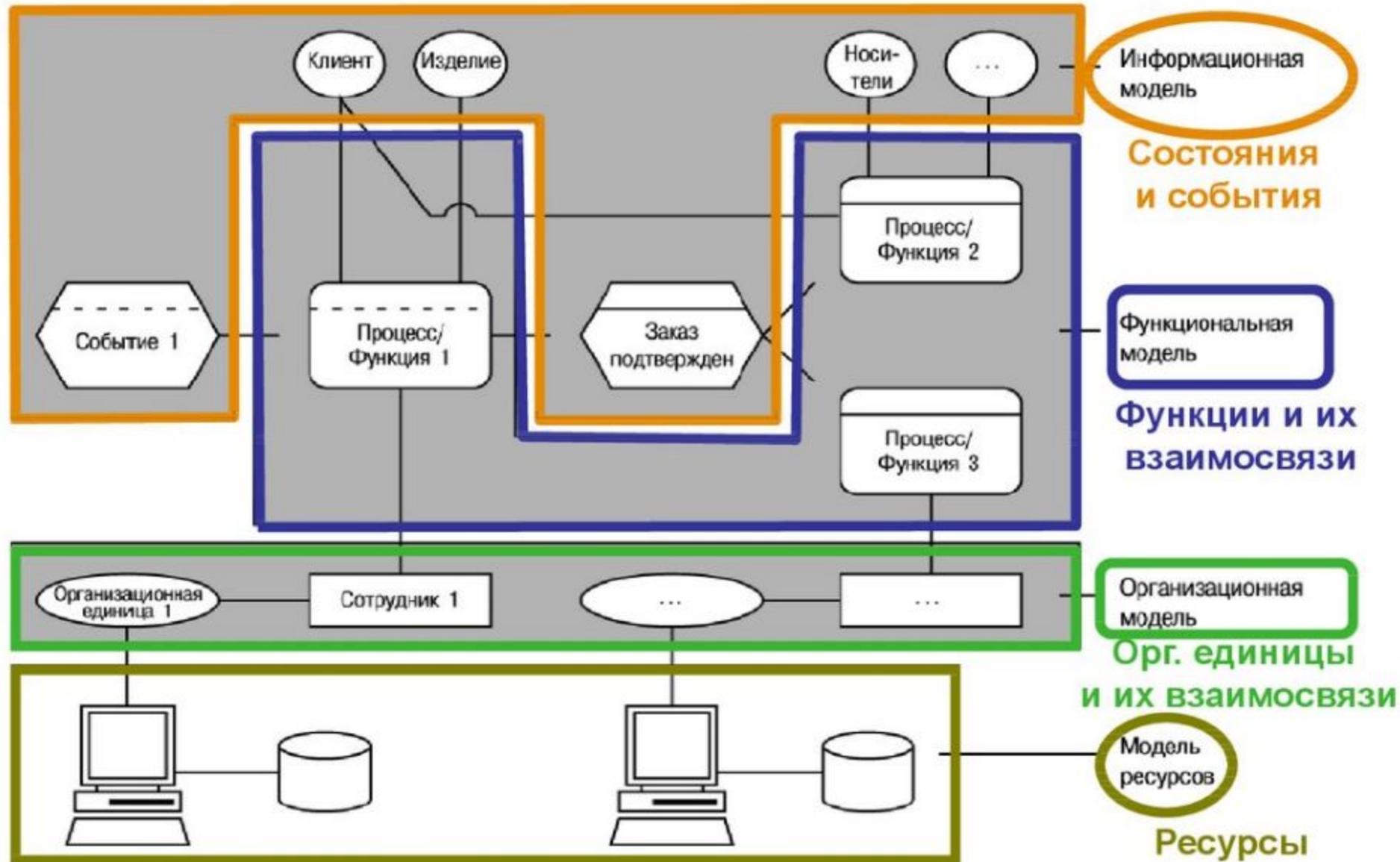


# Представления архитектуры ARIS



Первый шаг при создании архитектуры интегрированных информационных системы (ARIS) - разработка модели бизнес-процесса, для улучшения организации которого разрабатывается ИС

# Представления архитектуры ARIS



Для упрощения модель бизнес-процесса делится на отдельные типы моделей

# Представления архитектуры ARIS

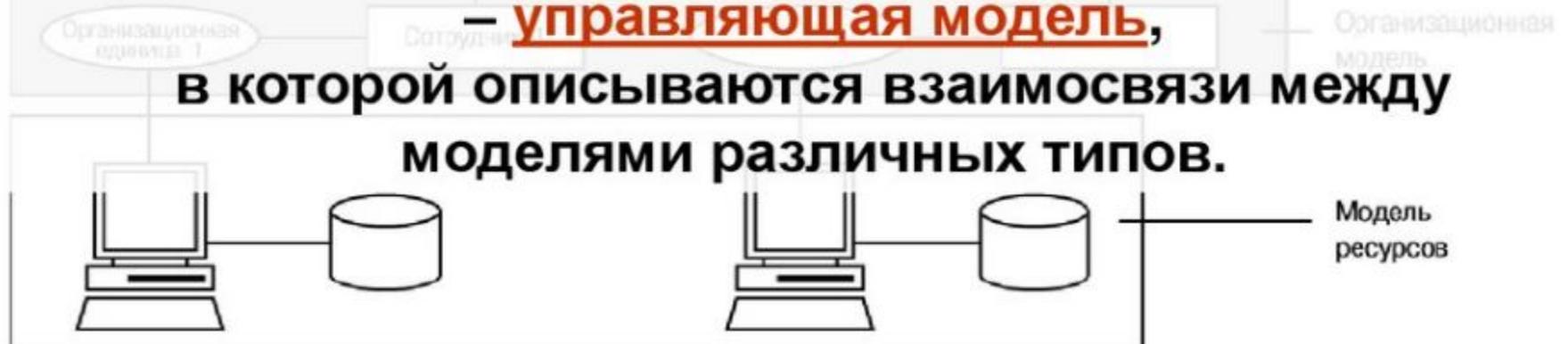


**Декомпозиция бизнес-процесса на отдельные типы моделей уменьшает степень его сложности за счет исключения из рассмотрения взаимосвязей между компонентами процесса, относящихся к другому типу моделей.**

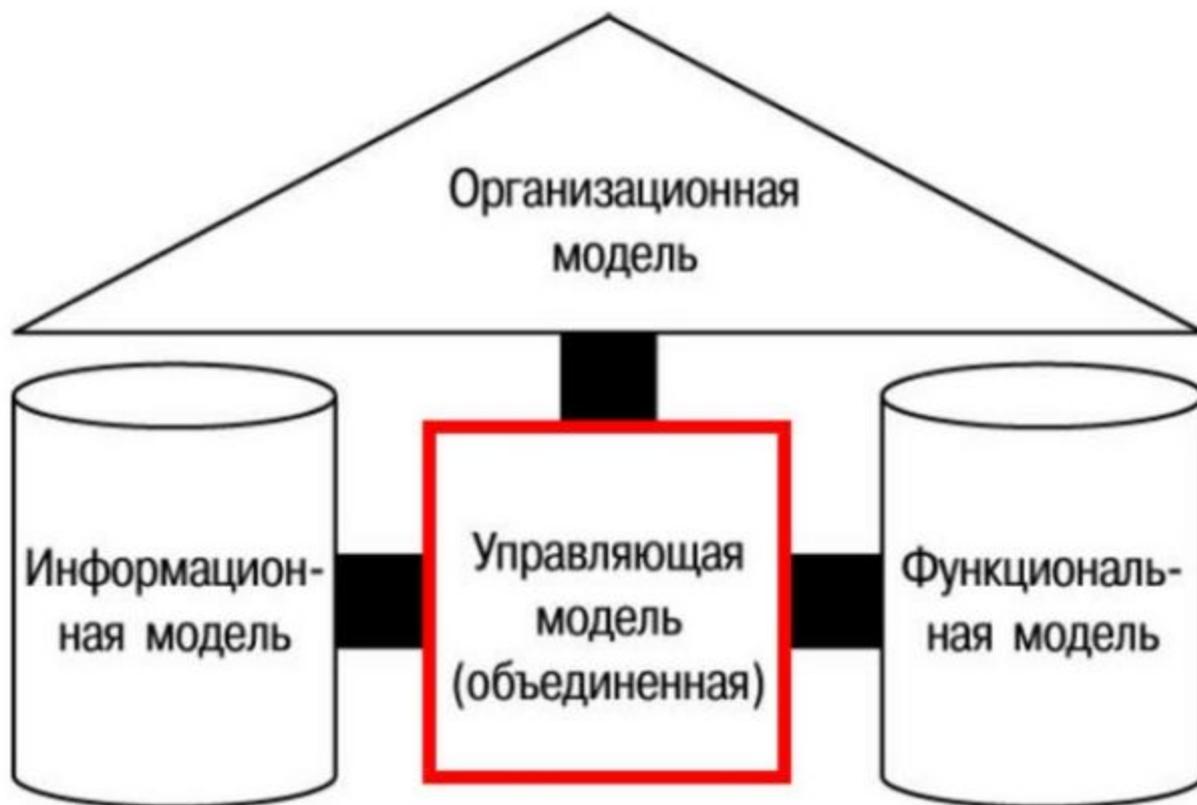
**В связи с этим вводится дополнительный тип модели**

**– управляющая модель,**

**в которой описываются взаимосвязи между моделями различных типов.**



**Представления архитектуры интегрированных  
информационных систем  
(ARIS – Architecture of Integrated Information Systems)  
компании IDS Scheer AG**

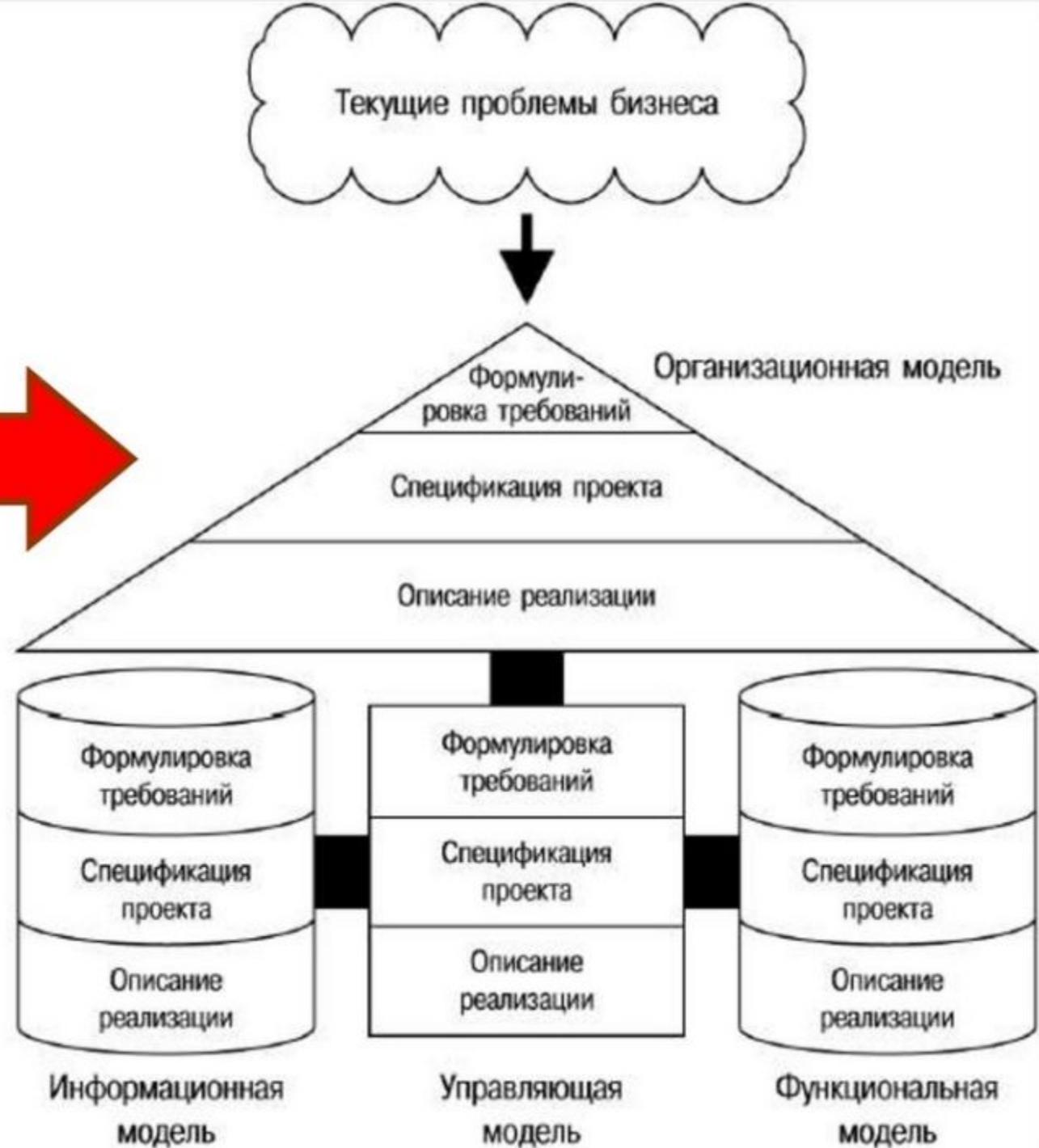
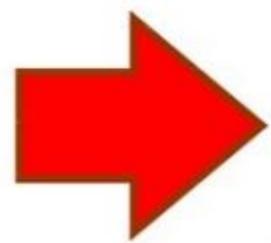


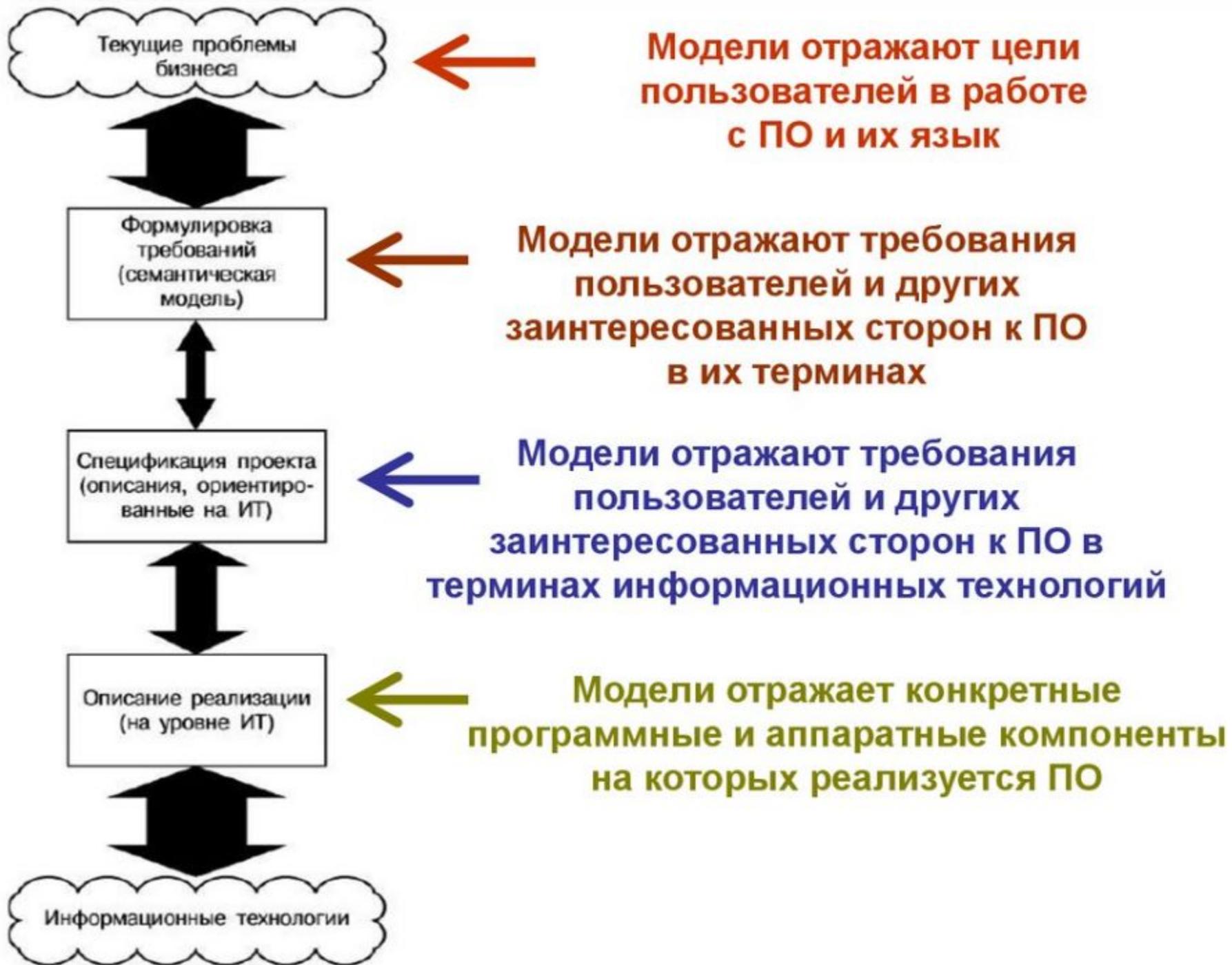
**Управляющая модель – важнейшая компонента архитектуры ARIS, отличающая ее от архитектур, предлагаемых другими авторами.**

# Модель жизненного цикла программного обеспечения ARIS



Каждое представление архитектуры ARIS (каждая модель ARIS) делится на уровни представления, соответствующие каждому из этапов жизненного цикла ПО





# Диаграмма процесса – описание текущих проблем бизнеса

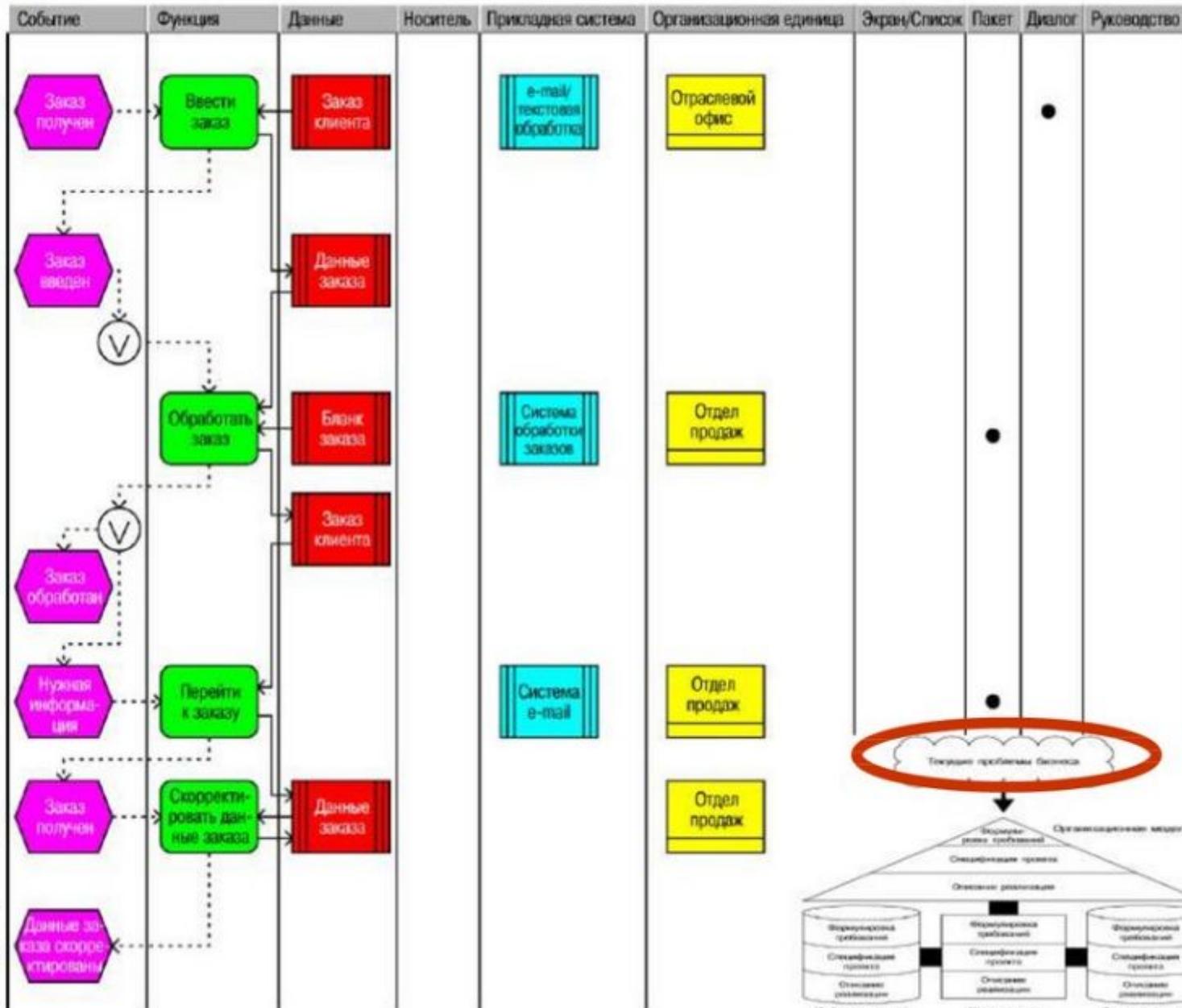
Текущие проблемы бизнеса

Формулировка требований  
(семантическая модель)

Спецификация проекта  
(описания, ориентированные на ИТ)

Описание реализации  
(на уровне ИТ)

Информационные технологии



Текущие проблемы бизнеса

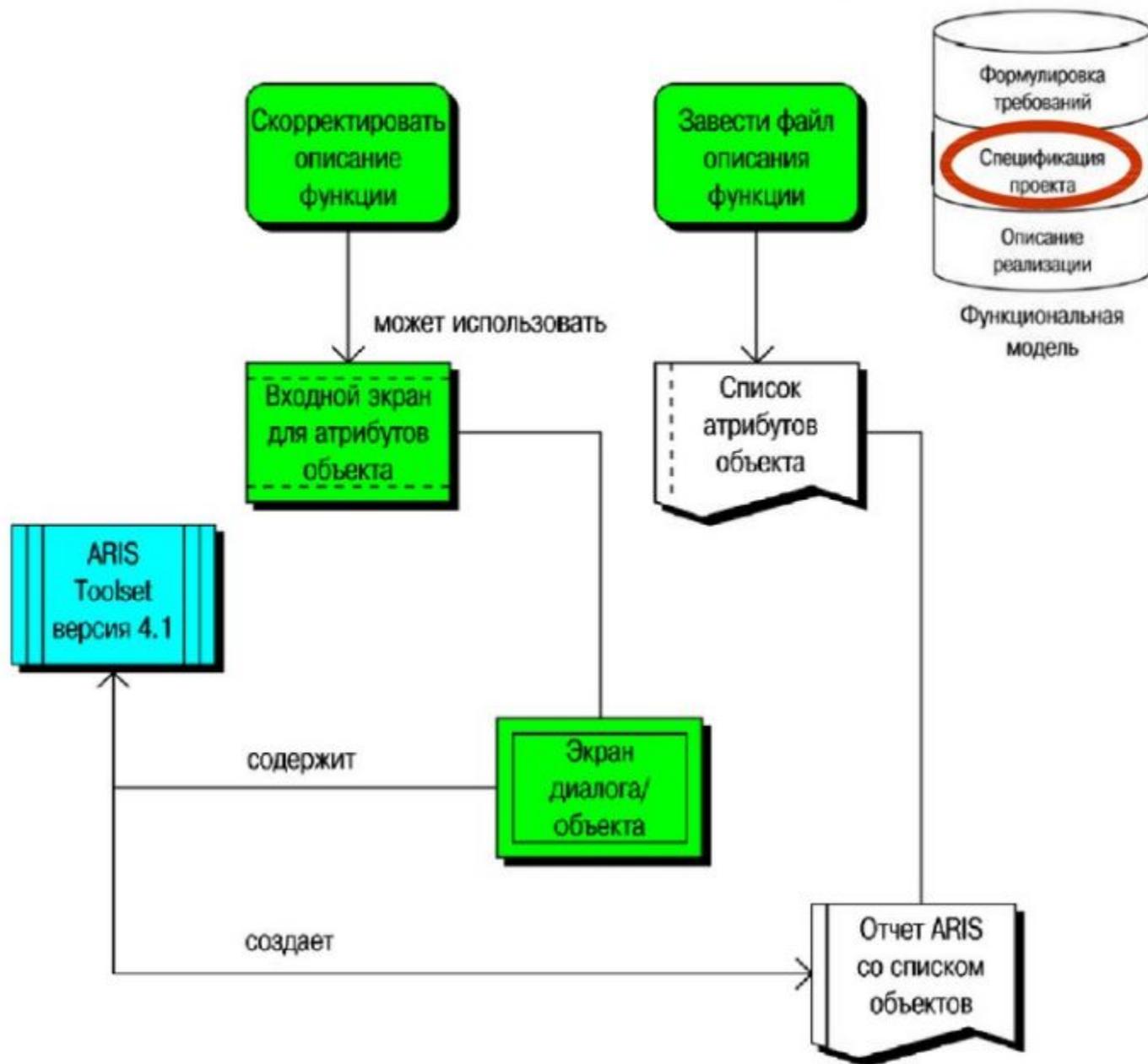




## Функциональное дерево – формулировка требований



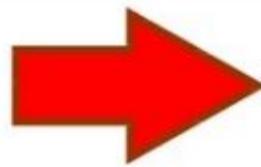
# Привязка экранов и списков – спецификация проекта



# Привязка типов ПС, типов программных элементов и программных элементов – описание реализации



# Представления архитектуры унифицированного процесса разработки ПО (RUP) IBM Rational

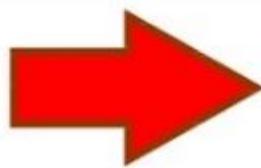


- 1 Бизнес-модель. Определяет абстракцию организации, для которой создается система.
- 2 Модель области определения. Фиксирует контекстное окружение системы.
- 3 Модель Use Case – вариантов использования. Определяет функциональные требования к системе.
- 4 Модель анализа. Интерпретирует требования к системе в терминах проектной модели.
- 5 Проектная модель. Определяет словарь проблемы и ее решение.
- 6 Модель размещения. Определяет аппаратную топологию, в которой исполняется система.
- 7 Модель реализации. Определяет части, которые используются для сборки и реализации физической системы.
- 8 Тестовая модель. Определяет тестовые варианты для проверки системы.
- 9 Модель процессов. Определяет параллелизм в системе и механизмы синхронизации.

## В отличие от ARIS каждое представление архитектуры RUP:

- 1 Ориентировано на определенную группу заинтересованных сторон.
- 2 Описывается с помощью разных моделей предназначенных для определенных групп заинтересованных сторон.
- 3 Каждое представление не делиться на уровни.

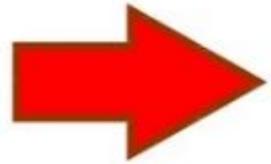
# Представления архитектуры унифицированного процесса разработки ПО (RUP) IBM Rational



# Представления архитектуры интегрированных информационных систем (ARIS) IDS Sheer AG

- В отличие от RUP каждое представление архитектуры ARIS:
- 1 Ориентировано на все группы заинтересованных сторон.
  - 2 Описывается с помощью одной модели.
  - 3 Каждое представление – модель делиться на уровни ориентированные на определенные группы заинтересованных сторон.

# Представления архитектуры унифицированного процесса разработки ПО (RUP) IBM Rational



Для проектировщиков ПС

Логическое представление

Представление процессов

Представление Use Case

Представление реализации

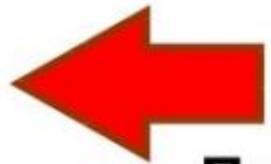
Представление размещения

Для пользователей

Для кодировщиков ПС



Для сетевых администраторов



# Представления архитектуры интегрированных информационных систем (ARIS) IDS Sheer AG

Текущие проблемы бизнеса



Информационная модель

Управляющая модель

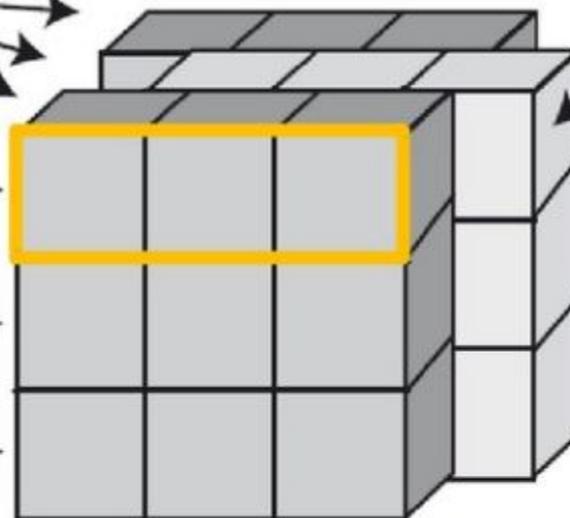
Функциональная модель

**Пронизывают**

Архитектурные аспекты

Архитектурные уровни

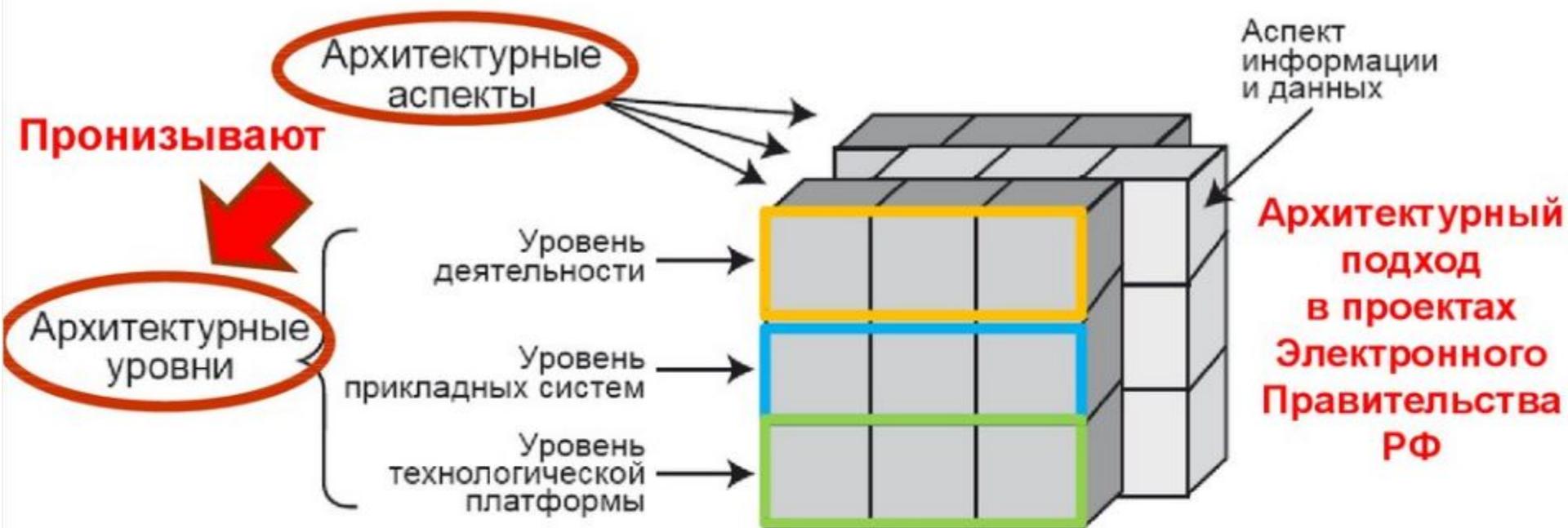
Уровень деятельности  
Уровень прикладных систем  
Уровень технологической платформы



Аспект информации и данных

**Архитектурный подход в проектах Электронного Правительства РФ**

- **аспект информации и данных**, определяющий состав информации, необходимой для поддержания административных процессов и используемой, например, в процессах межведомственного взаимодействия; какие информационные объекты (сущности) нужны при моделировании системной архитектуры; наконец, какие структуры данных и метаданных должны реализовываться на уровне технологической архитектуры для их долговременного использования в прикладных системах;
- **аспект эффективности и результативности**, характеризующий набор ключевых показателей эффективности выполнения государственных функций и предоставления услуг гражданам, бизнесу, другим органам государственной власти на основе ИКТ, а также показатели эффективности использования самих ИКТ;
- **аспект информационной безопасности и доверия**, учитывающий необходимость адекватной защиты информации и данных от попыток искажения, кражи и других способов неавторизованного использования и обеспечивающий доверие пользователей к использованию ИКТ;
- **аспект интеграции и взаимодействия**, обеспечивающий электронное взаимодействие между различными информационными системами и учитывающий организационную, семантическую и технологическую совместимость;
- другие аспекты.



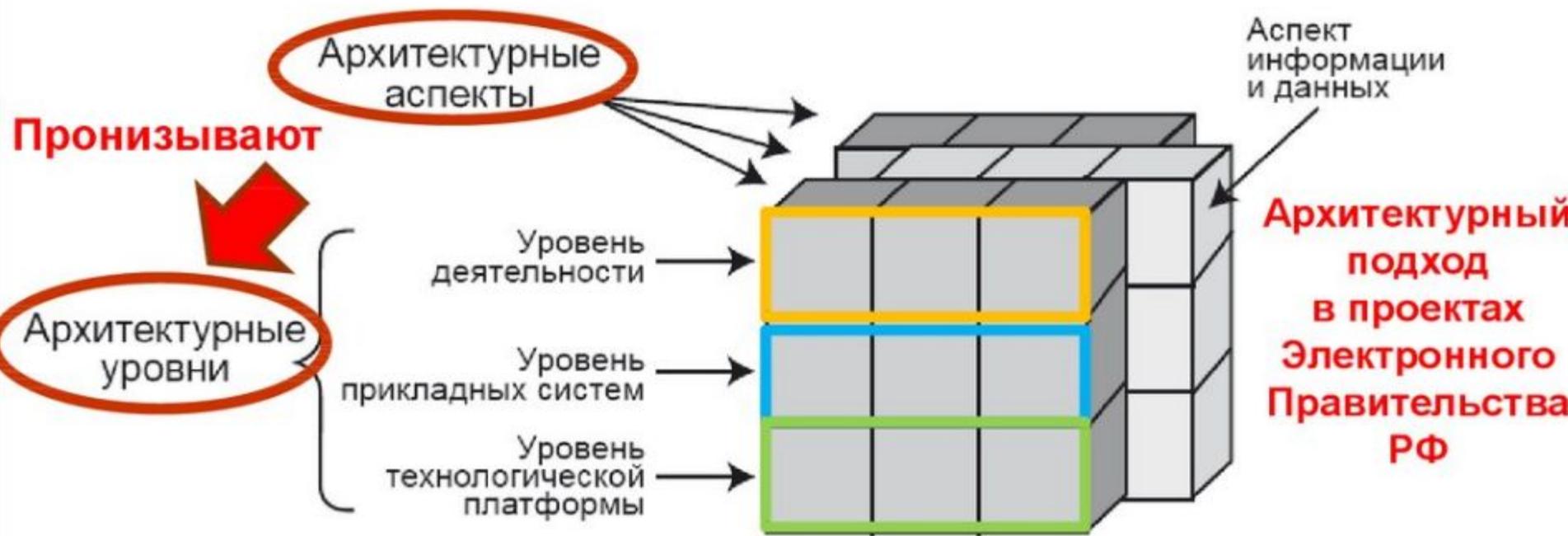
## Аспект: информации и данных

**ОПРЕДЕЛЯЕТ**

**УРОВЕНЬ ДЕЯТЕЛЬНОСТИ:**  
состав информации, необходимой для поддержания административных процессов (например процессов взаимодействия между разными ведомствами)

**УРОВЕНЬ ПРИКЛАДНЫХ СИСТЕМ:**  
информационные объекты (сущности) для реализации на уровне отдельных прикладных систем

**УРОВЕНЬ ТЕХНОЛОГИЧЕСКОЙ ПЛАТФОРМЫ:**  
информационные объекты (сущности) для реализации на уровне информационной системы в целом



## Аспект: эффективности и результативности

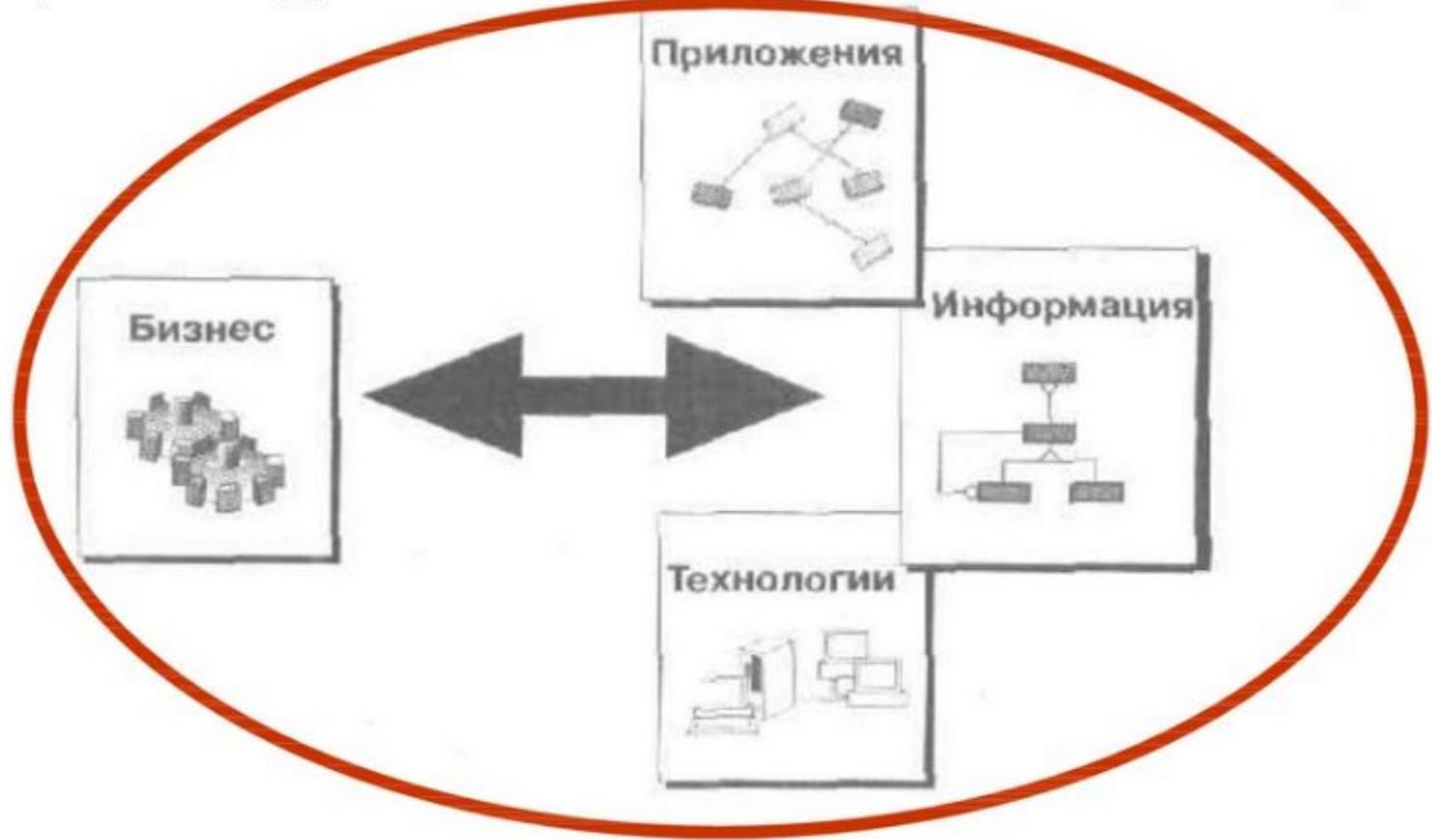


**УРОВЕНЬ ДЕЯТЕЛЬНОСТИ:**  
показатели эффективности работы органов государственной власти в результате внедрения информационной системы

**УРОВЕНЬ ПРИКЛАДНЫХ СИСТЕМ:** показатели эффективности работы отдельных прикладных систем

**УРОВЕНЬ ТЕХНОЛОГИЧЕСКОЙ ПЛАТФОРМЫ:**  
показатели эффективности информационной системы

# Элементы (перспективы) производственной архитектуры Microsoft Solution Framework (MSF)



**Концептуальное, логическое  
и физическое представления**

### Практическое задание

- 1 Определите точки зрения каких заинтересованных сторон и почему должна отражать архитектура Вашего программного средства.
- 2 Какая из рассмотренных моделей архитектур наиболее приемлема для Вашего программного средства.