

# Управление требованиями и проектирование информационных систем

**Лекция 1**

Стариков Н.Ю., ВШЭ ВШБ ДБИ, 1 модуль 2021/2022 уч.года

# ПЛАН ЛЕКЦИИ

- Введение в инженерию требований
- Базовые понятия инженерии требований
  - Требование, описание, модель
  - Системный подход, целевая система
  - Стейкхолдеры, множественность интересов
  - Классификация требований в контексте ИТ-проекта
  - Требования, потребности, ограничения
  - Жизненный цикл, обеспечение, практика
- Методы инженерии требований
  - Фреймворки, стандарты, своды знаний → Essence



# ВВЕДЕНИЕ В ИНЖЕНЕРИЮ ТРЕБОВАНИЙ

# СЛОЖНЫЕ ИНЖЕНЕРНЫЕ ПРОЕКТЫ

- Нанотехнологии
- Квантовые эффекты
- Микрочипы
- Точная механика
- Криптоалгоритмы
- Умные вещи
- Умный транспорт
- Умные города
- Экстремальные условия
- Космические проекты

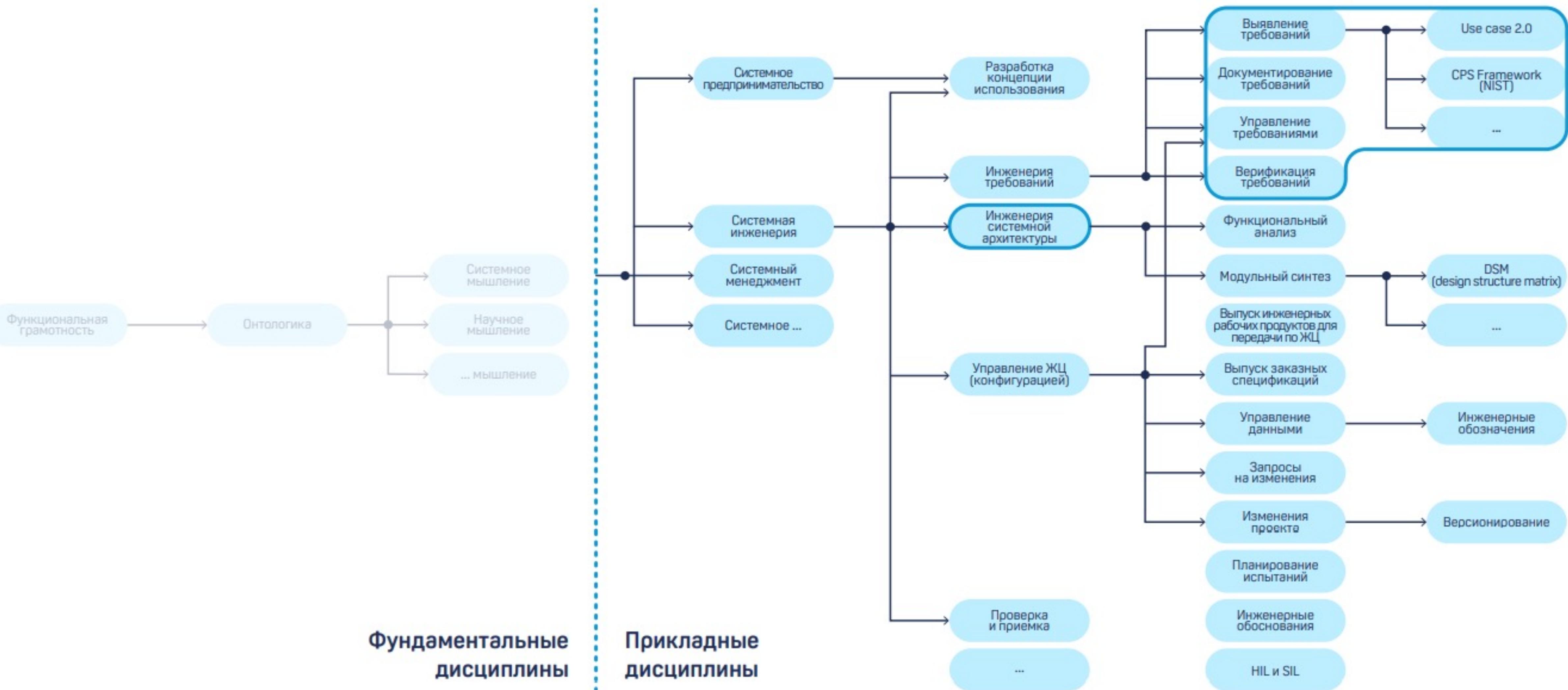


# СОВРЕМЕННАЯ ИНЖЕНЕРИЯ

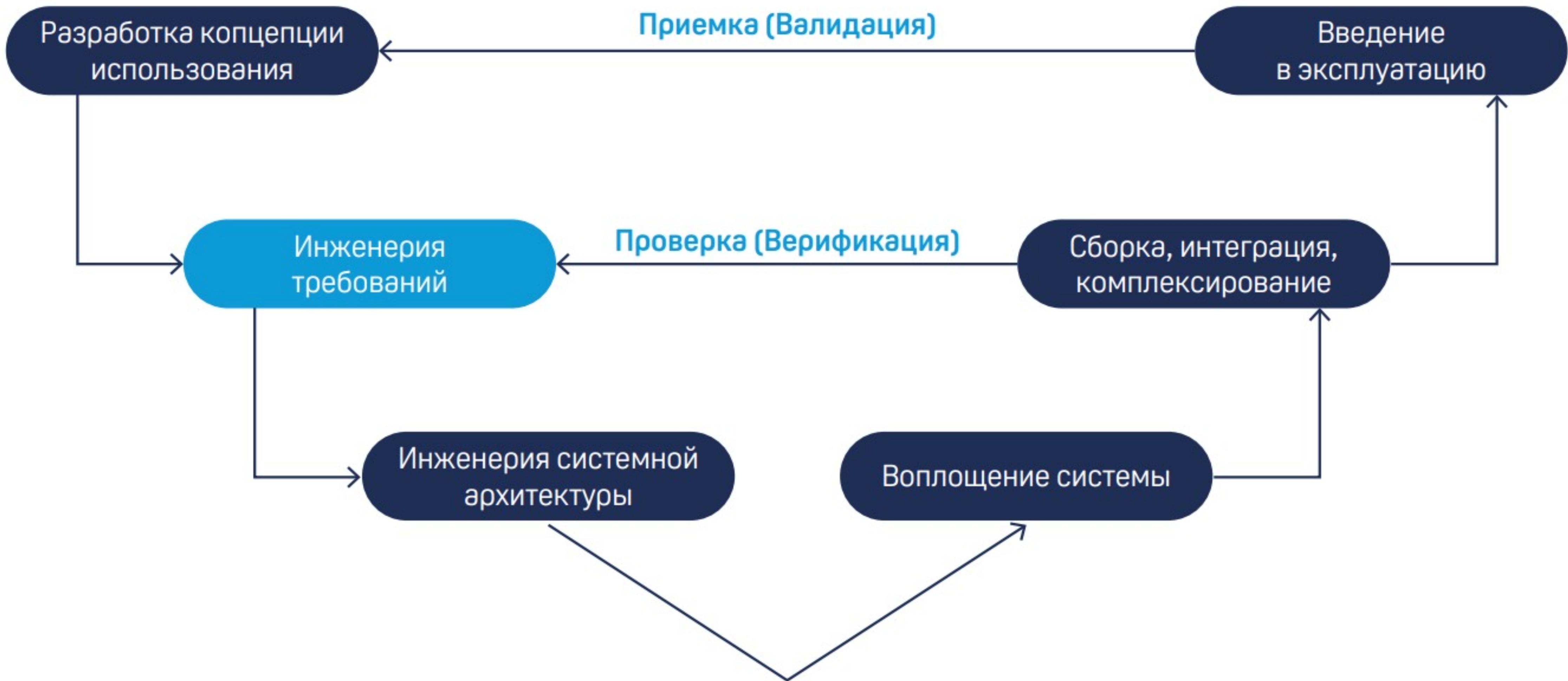
- ISO/IEC/IEEE 15288:2015 Systems engineering
- ISO/IEC/IEEE 12207:2017 Software engineering
- ISO/IEC/IEEE 29148:2018 Requirements engineering
- ISO/IEC/IEEE 42010:2011 (2017) Architecture description
- OMG Essence 1.2 (2018) Method engineering
- OpenGroup ArchiMate 3.1 (2019) Enterprise Architecture Modeling Language
- OpenGroup TOGAF 9.2 (2018) Enterprise Architecture Framework
- NIST PWG CPS Framework 1.0 (2016)



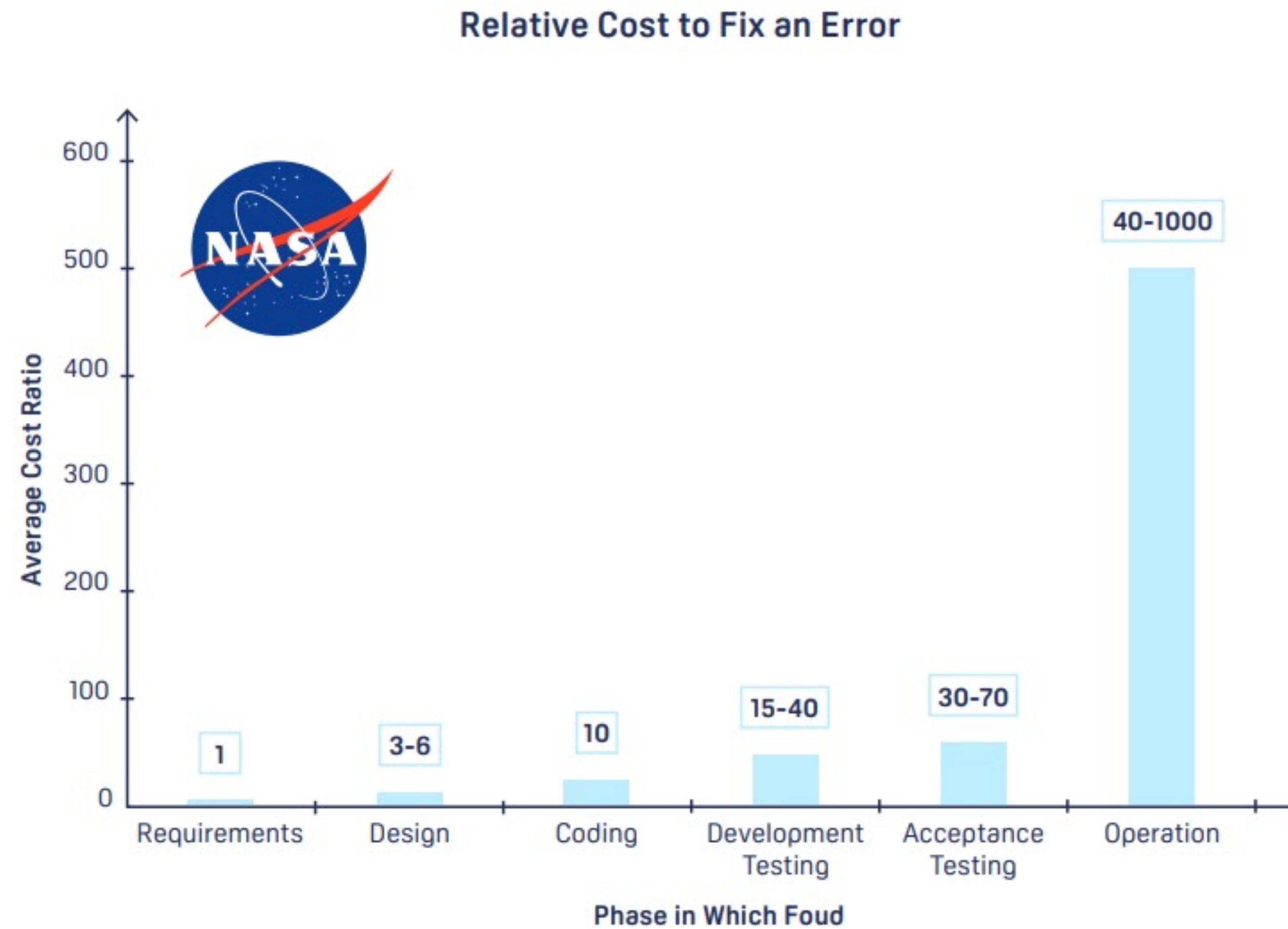
# СТЕК ИНЖЕНЕРНЫХ ДИСЦИПЛИН



# СТЕК ИНЖЕНЕРНЫХ ДИСЦИПЛИН



# ЦЕНА ОШИБКИ



# ЦЕЛЬ ИНЖЕНЕРИИ ТРЕБОВАНИЙ

- Перенести разбирательство с рисками проекта на ранние стадии, за счёт устранения конфликтов, неоднозначностей, неопределённостей
- Обеспечить согласованное понимание проекта среди стейххолдеров системы (покупателей, спонсоров, пользователей, поставщиков, и т.д.)
- Проверить систему на предмет того, удовлетворяет ли она реальным потребностям, может ли быть воплощена в реальности
- Задать основу для проверки архитектуры системы и принятых архитектурных решений, а также определить критерии для выбора из нескольких альтернативных архитектурных решений

# БАЗОВЫЕ ПОНЯТИЯ ИНЖЕНЕРИИ ТРЕБОВАНИЙ

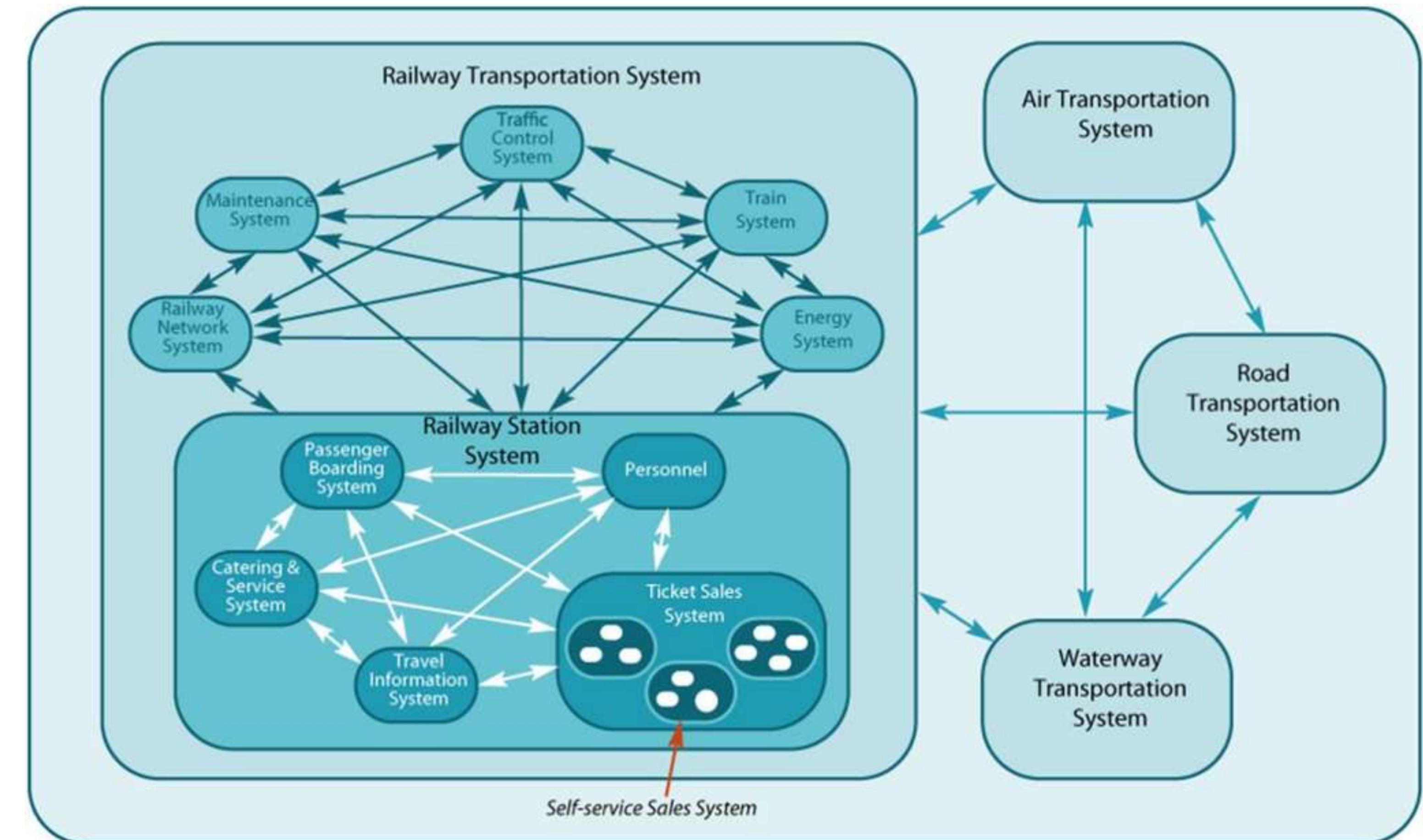
# ТРЕБОВАНИЕ, ОПИСАНИЕ, МОДЕЛЬ

- Требование – это описание объекта в деонтической модальности (долженствование)
- Описание объекта позволяет коммуницировать его модель в терминах выбранного языка
- Модель является упрощённым отражением основных свойств объекта, интересных конкретному лицу

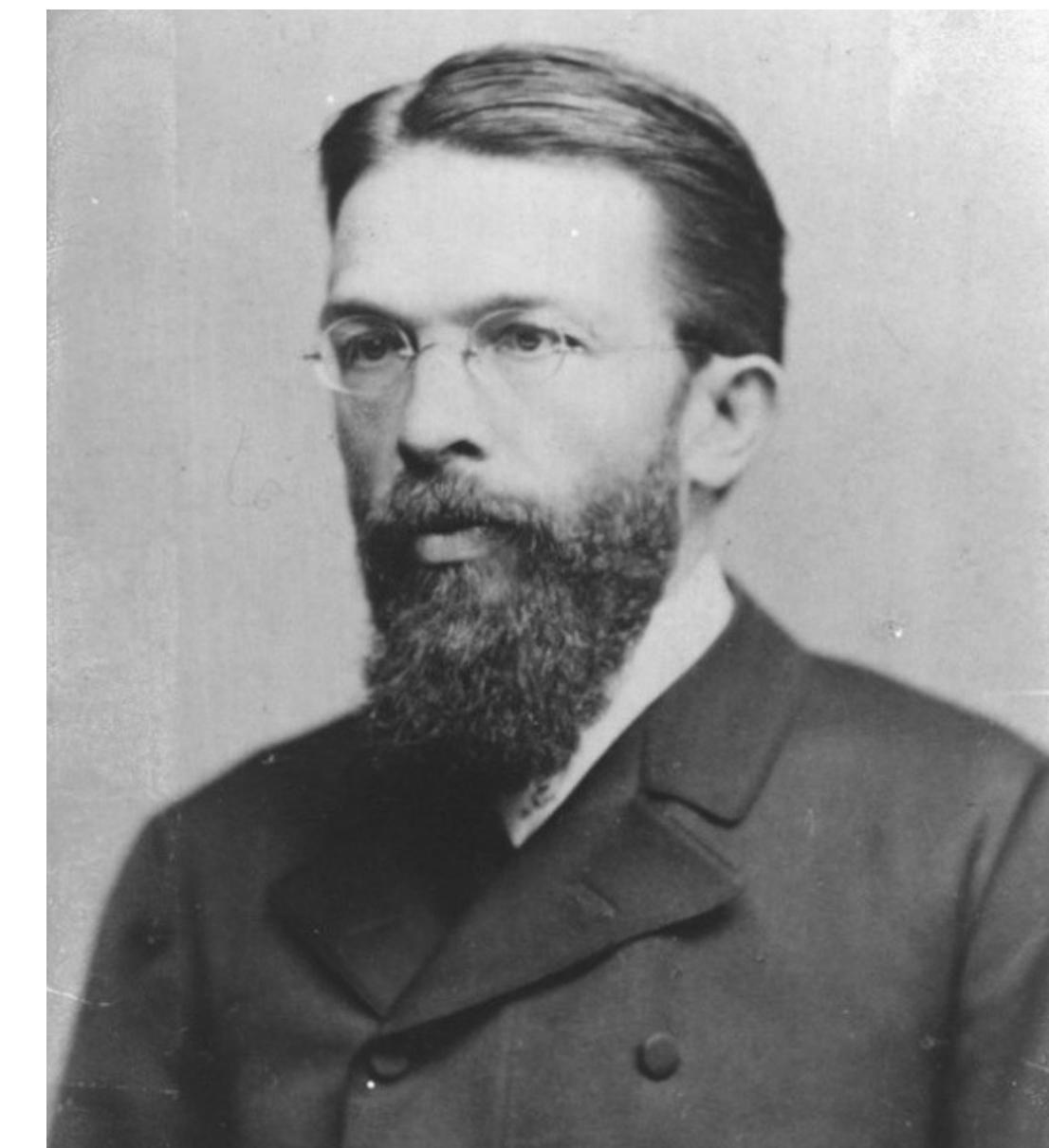
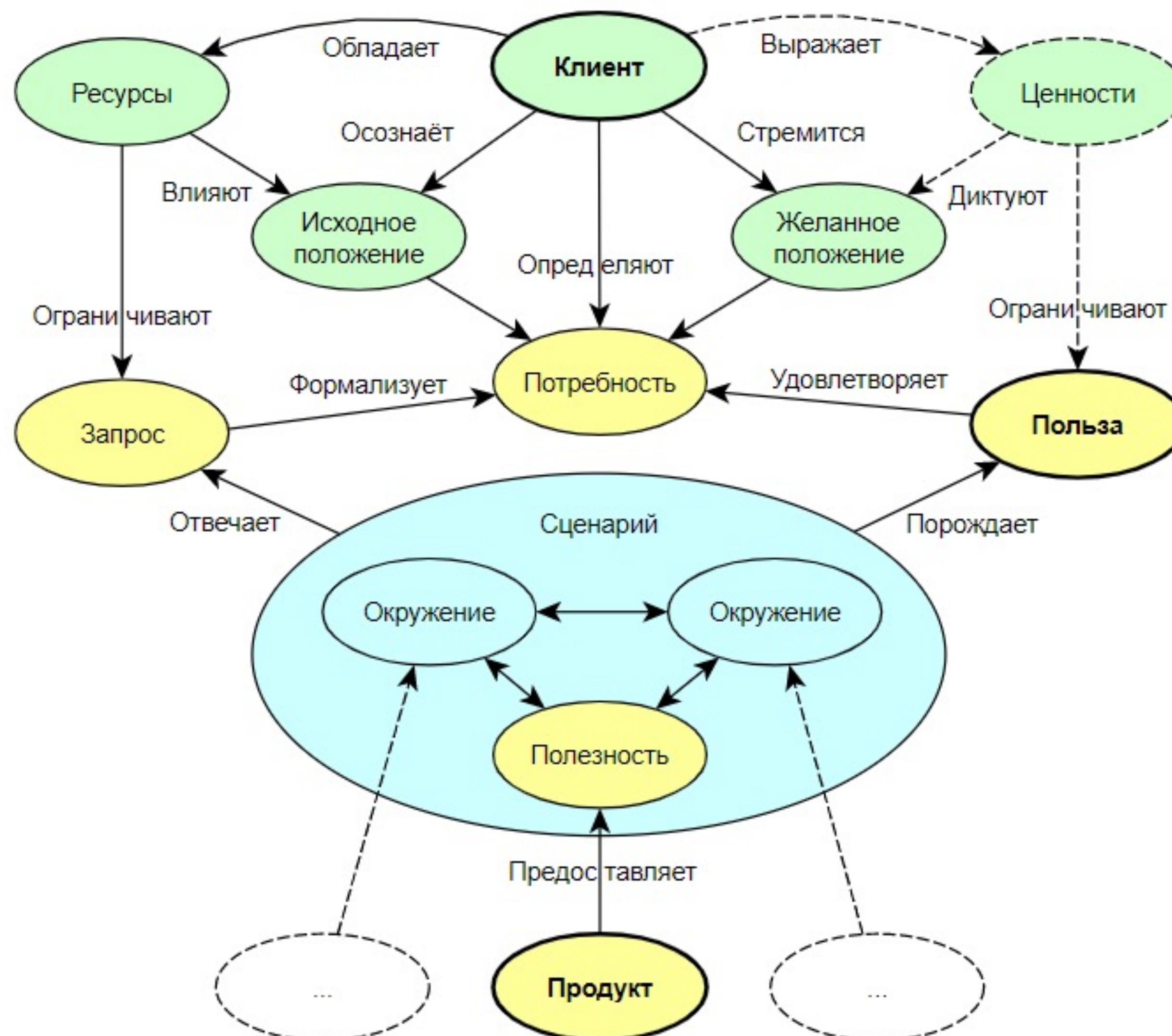


# Системный подход

- Целевая система
- Надсистема
- Системы в окружении
- Обеспечивающая система
- Подсистемы
- Разрабатываемая система (подсистема)



# ГЕНЕАЛОГИЯ ПОЛЕЗНОСТИ

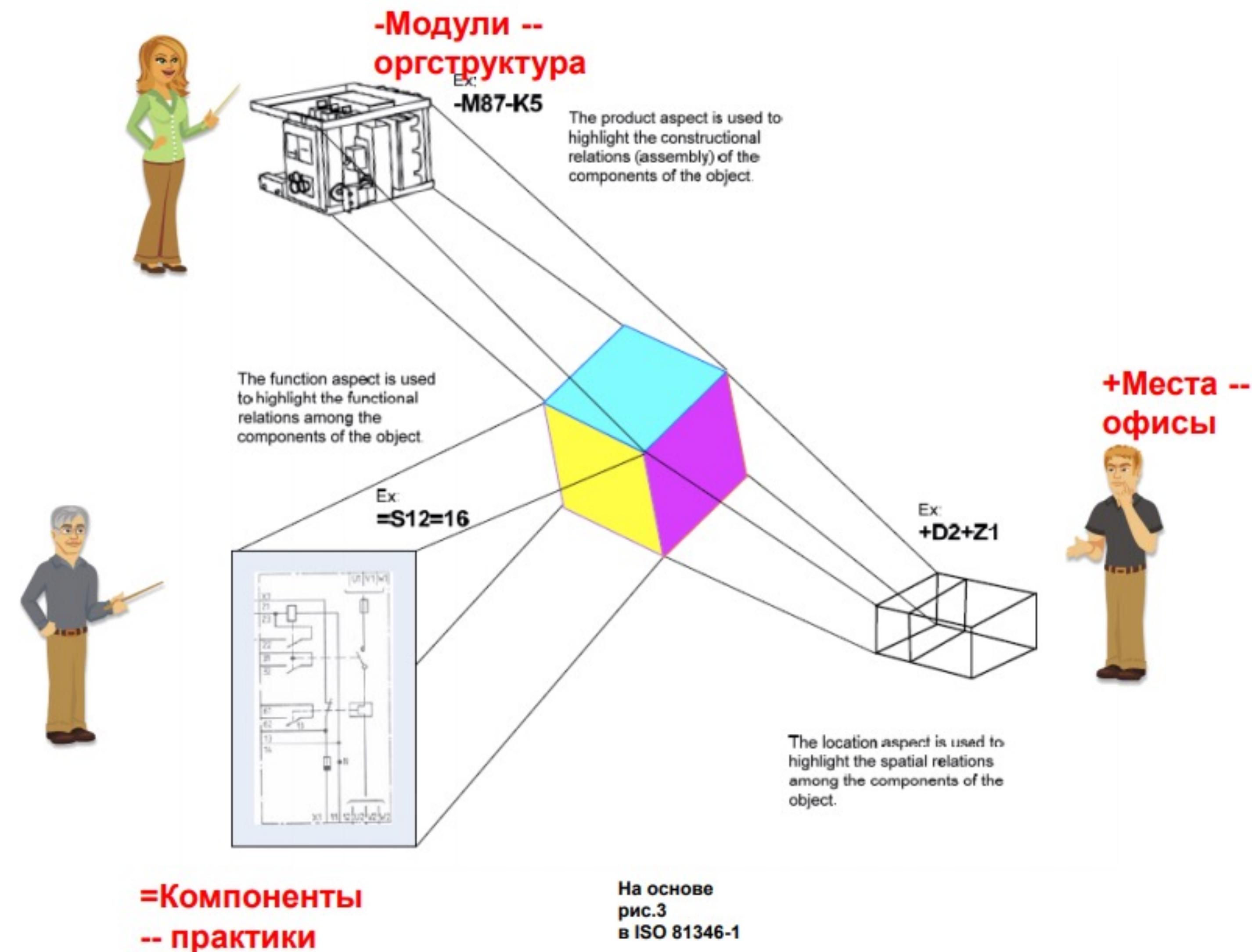


Карл Менгер  
Австрийская  
школа экономики

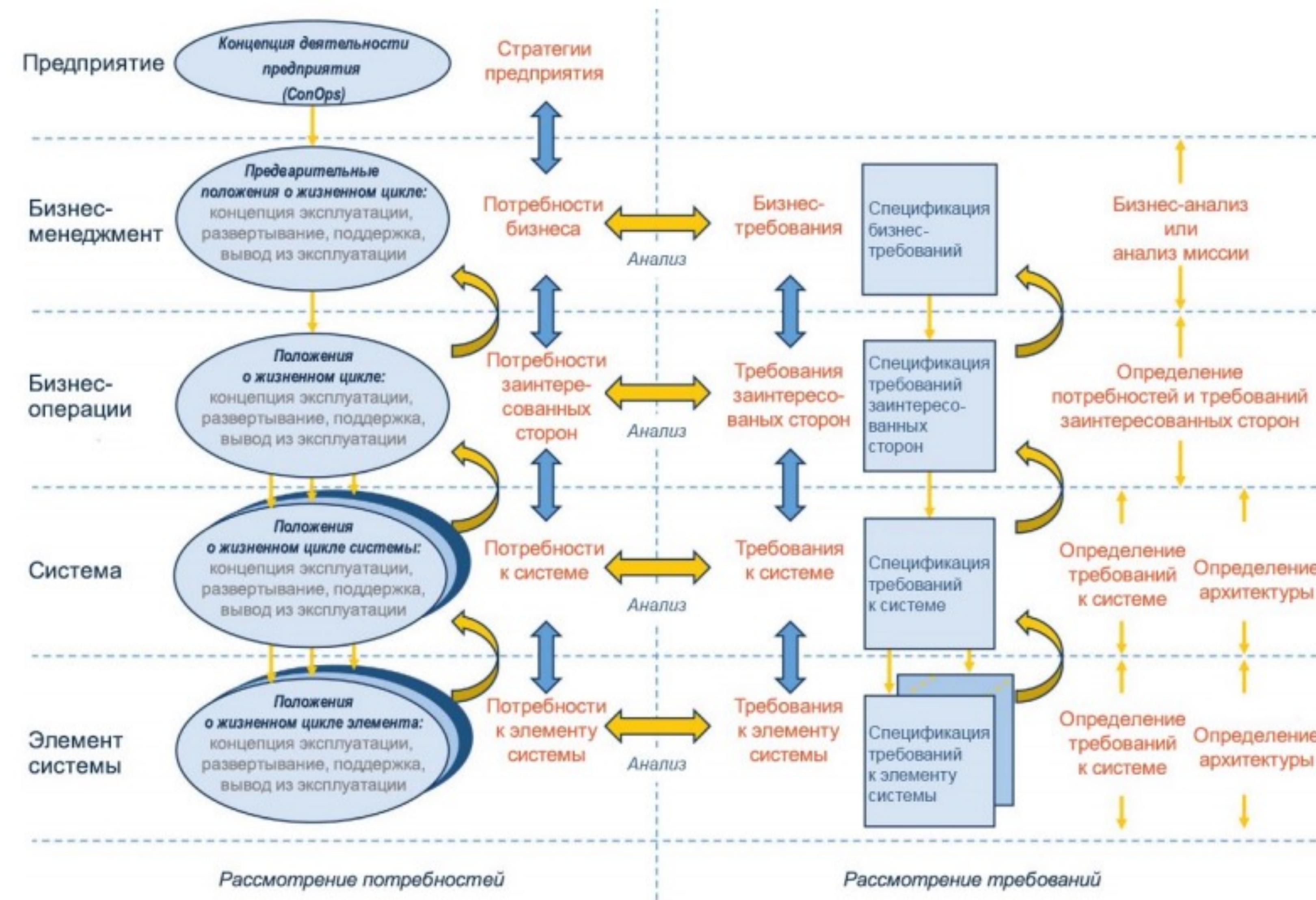
# АНАЛИЗ СТЕЙКХОЛДЕРОВ: РОЛИ



# МНОЖЕСТВЕННОСТЬ ОПИСАНИЙ



# КЛАССИФИКАЦИЯ ТРЕБОВАНИЙ



# КЛАССИФИКАЦИЯ ТРЕБОВАНИЙ

## Пользовательские потребности

Это определение как чёрного ящика использующей системы (using systems), в которой участвует целевая система инженера как «чёрный ящик»

## Требования проектных ролей (стейкхолдеров)

Это требования отдельных проектных ролей (стейкхолдеров), как они их выражают

## Требования к run-time поведению

Это функциональные требования

## Требования к системе

Это согласованные между собой требования к системе, которые отражают требования стейкхолдеров

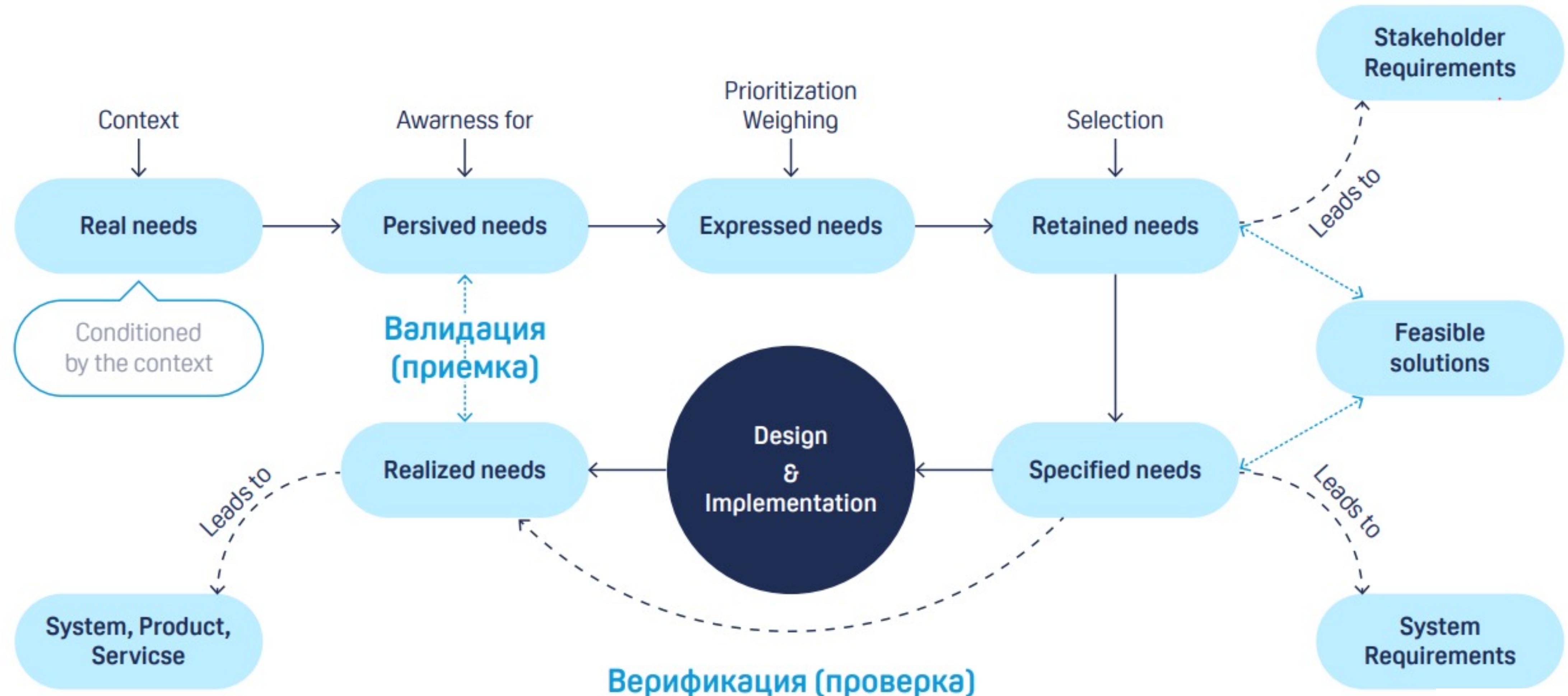
## Требования к качеству

Ожидание поведения системы помимо требований относящихся к run-time

## Ограничения

Конкретные инженерные решения для системы как «прозрачного ящика»

# ЖИЗНЕННЫЙ ЦИКЛ ПОТРЕБНОСТИ

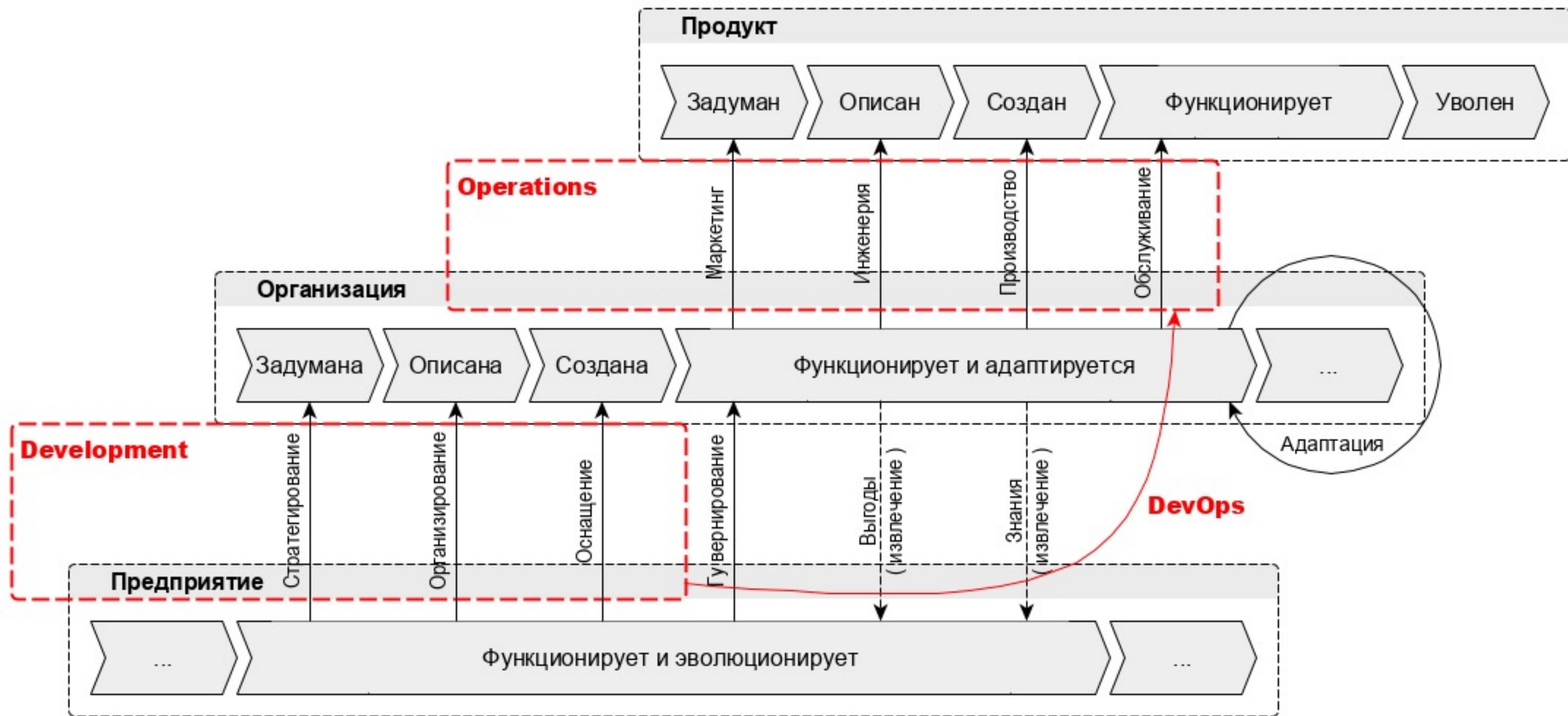


# ЖИЗНЕННЫЙ ЦИКЛ... В ЖИЗНИ ???

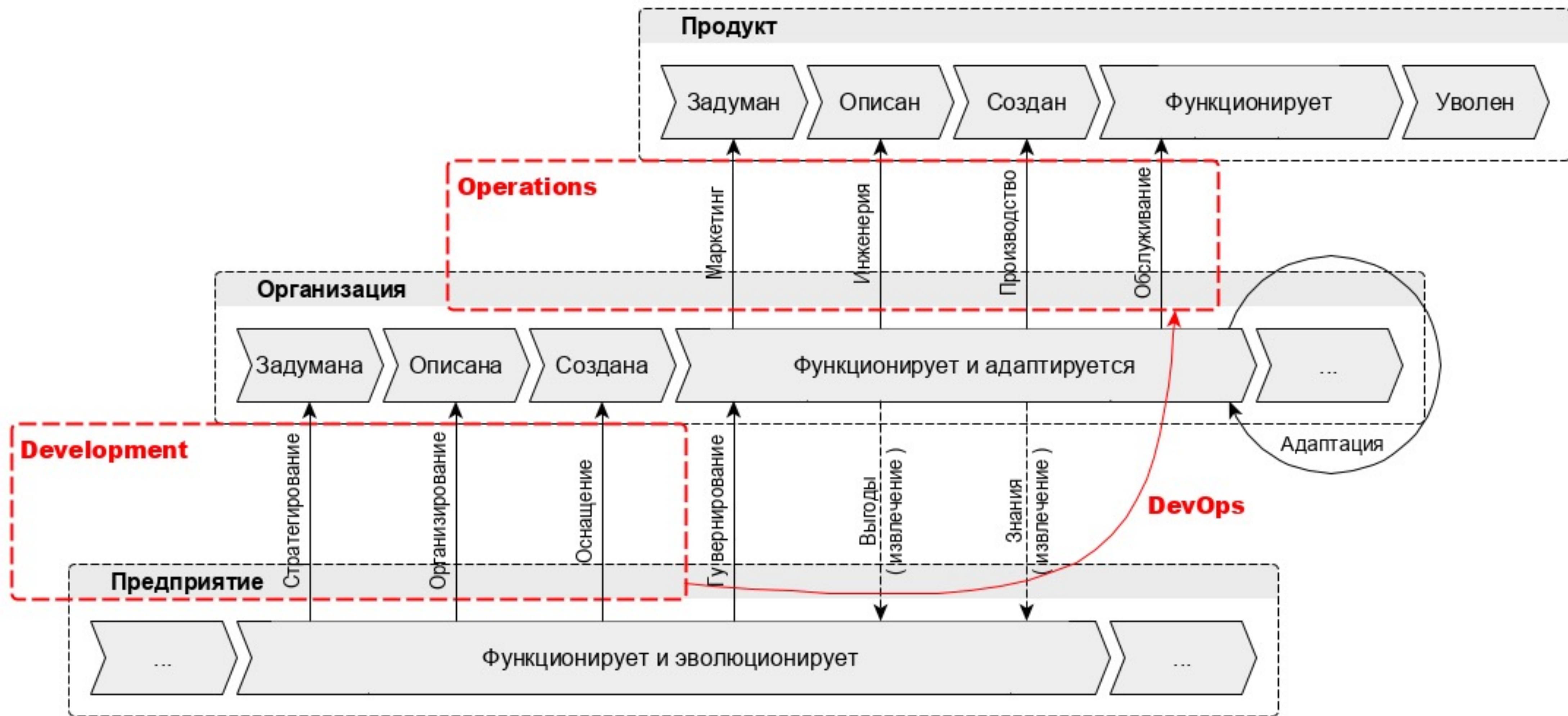
## Жизненный цикл печёночного сосальщика



# ЖИЗНЕННЫЙ ЦИКЛ В ИНЖЕНЕРИИ



# ЖИЗНЕННЫЙ ЦИКЛ В ИНЖЕНЕРИИ



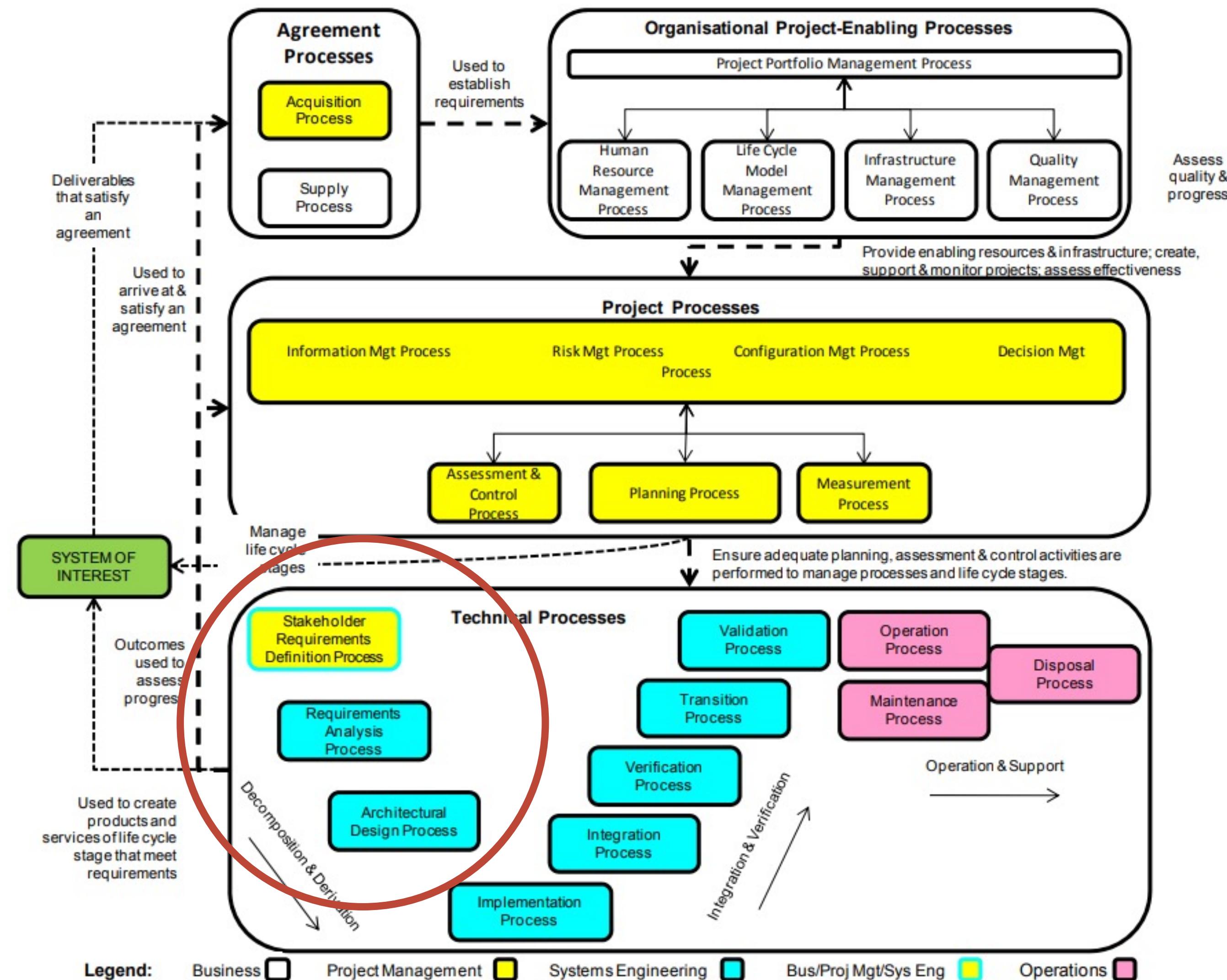
# МЕТОДЫ ИНЖЕНЕРИИ ТРЕБОВАНИЙ

# СТАНДАРТ, ФРЕЙМВОРК... МЕТОД?

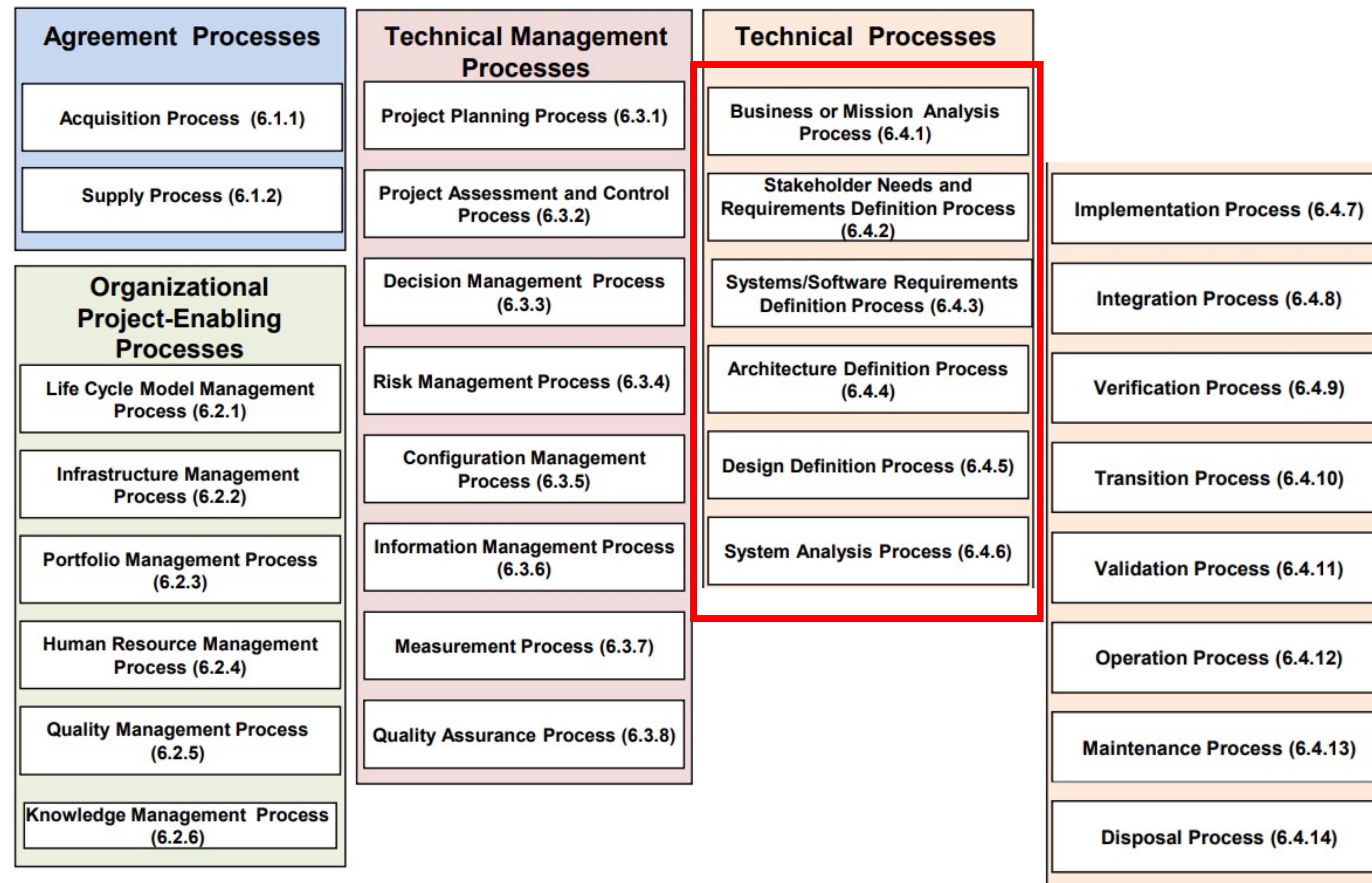
- ISO/IEC/IEEE 15288:2015 Systems engineering
- ISO/IEC/IEEE 12207:2017 Software engineering
- ISO/IEC/IEEE 29148:2018 Requirements engineering
- ISO/IEC/IEEE 42010:2011 (2017) Architecture description
- OMG Essence 1.2 (2018) Method engineering
- OpenGroup ArchiMate 3.1 (2019) Enterprise Architecture Modeling Language
- OpenGroup TOGAF 9.2 (2018) Enterprise Architecture Framework
- NIST PWG CPS Framework 1.0 (2016)



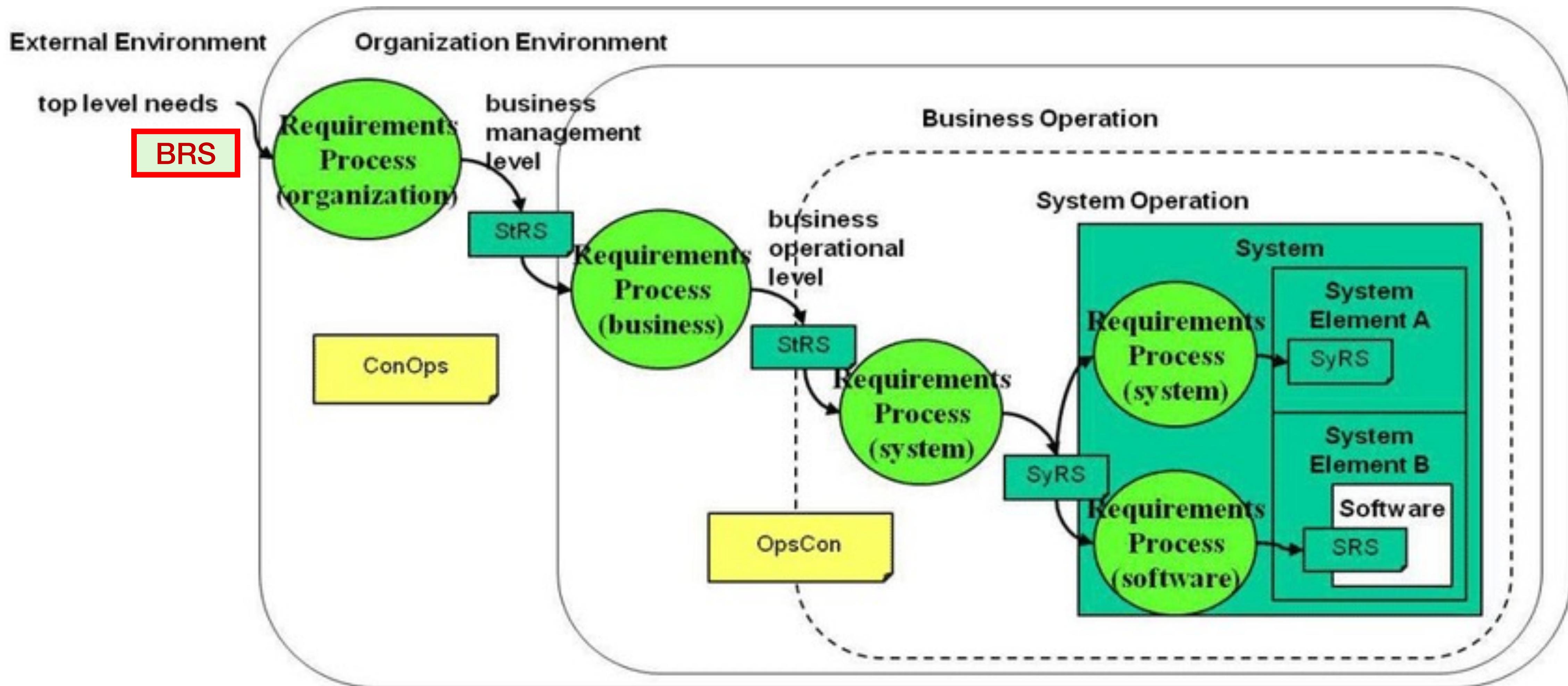
# СТАНДАРТ ISO 15288:2018



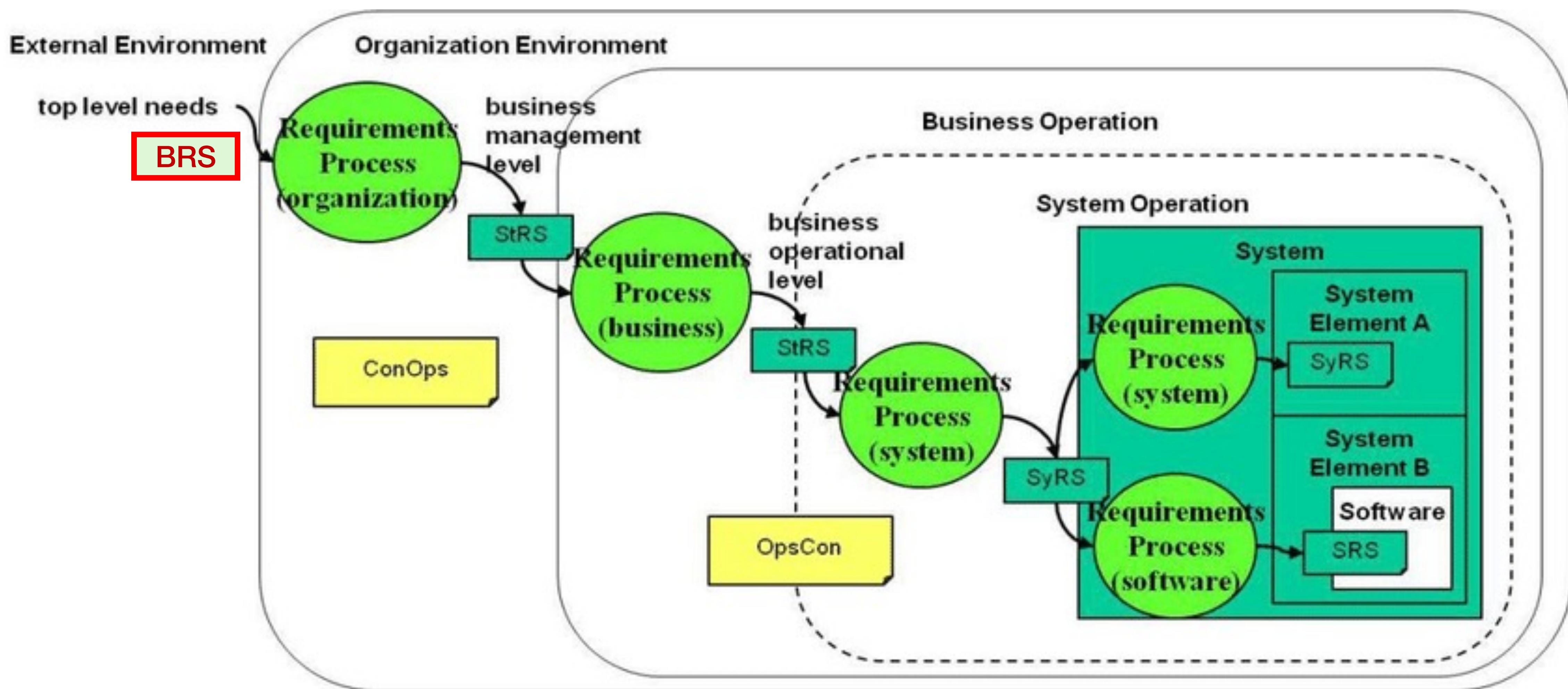
# СТАНДАРТ ISO 12207:2017



# СТАНДАРТ ISO 29148:2018



# СТАНДАРТ ISO 42010:2011



# СВОД ЗНАНИЙ SWEBOK

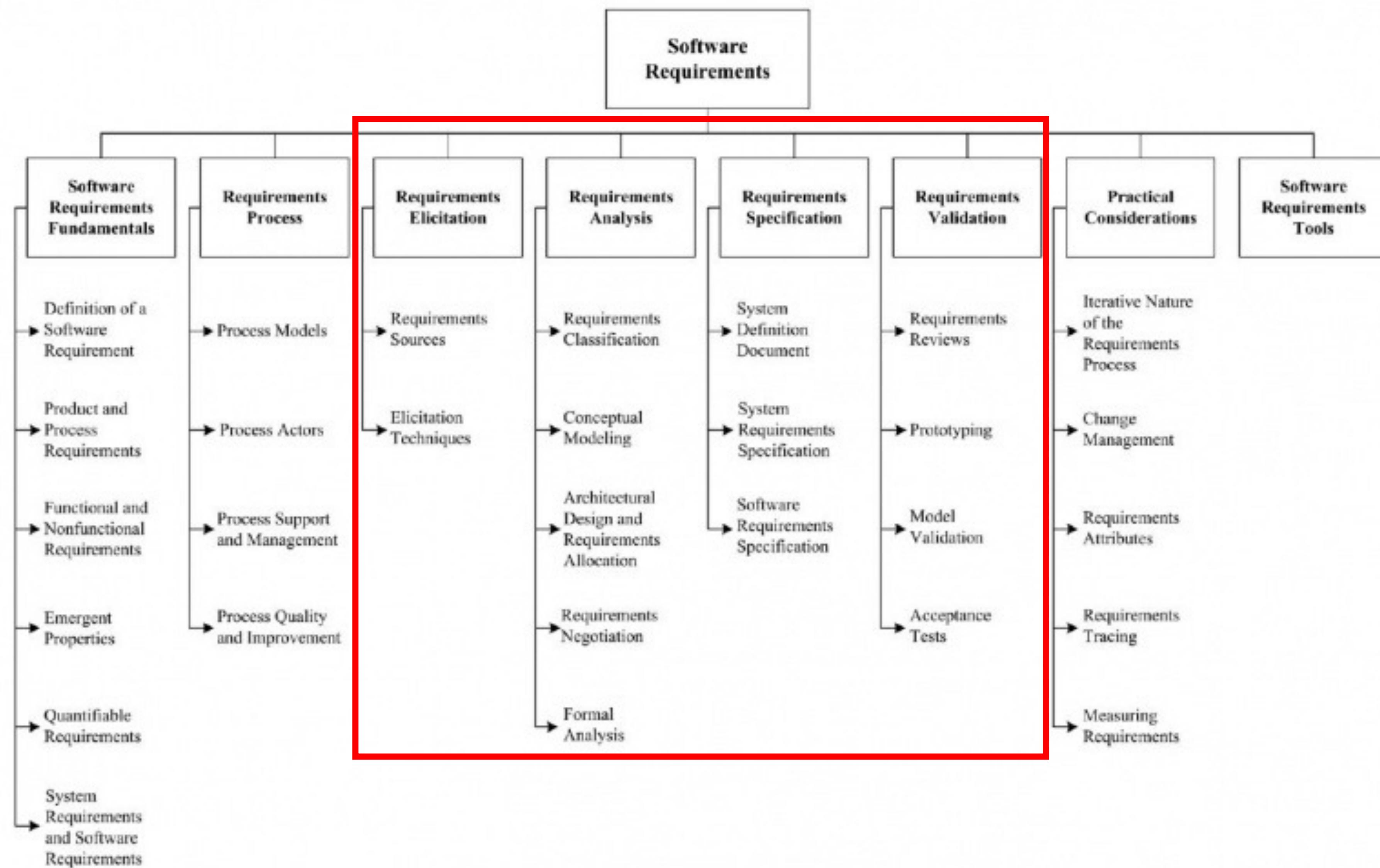
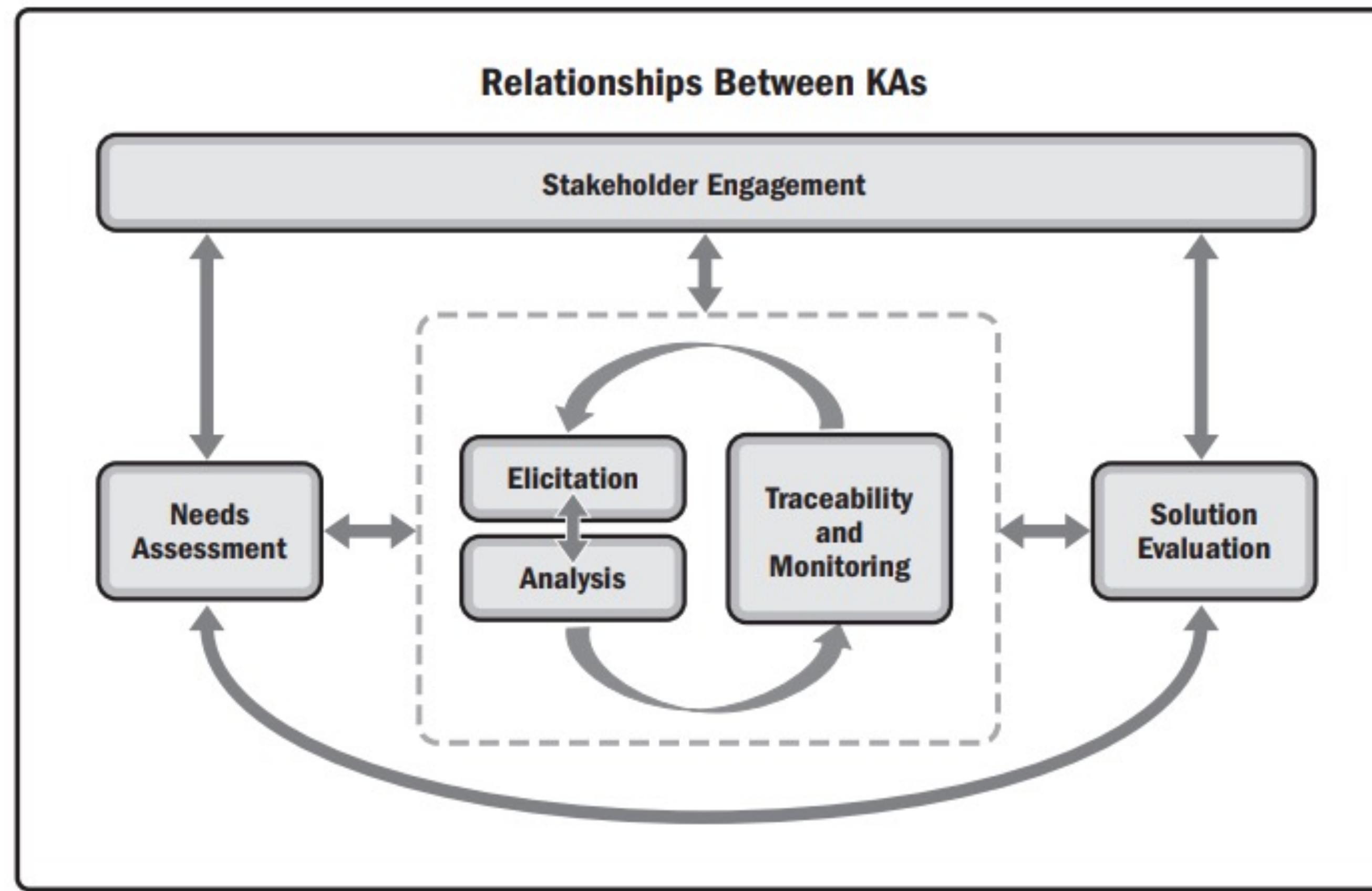


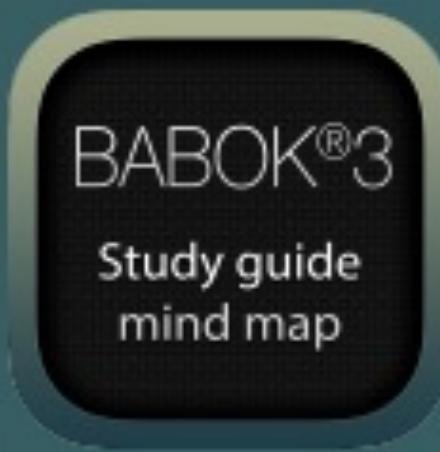
Table X2-1. Categorization and Index of Tools and Techniques

Tool or Technique	Section in <i>Business Analysis for Practitioners: A Practice Guide</i>	Section in <i>The PMI Guide to Business Analysis</i>	Knowledge Areas			
			Needs Assessment	Stakeholder Engagement	Elicitation	Analysis
<b>Stakeholder Maps</b>						
Onion diagram	N/A	5.2.2.4		5.2, 5.3		
Stakeholder matrix	N/A	5.2.2.4		5.2, 5.3		
<b>Ungrouped Tools and Techniques</b>						
Affinity diagram	2.4.5.2	4.3.2.1	4.3			7.9
Backlog management	N/A	7.7.2.1, 7.3.2.4				7.7
Behavior-driven development (BDD)	N/A	7.4.2.1				7.4
Benchmarking	2.4.5.3	4.1.2.1	4.1, 4.3, 4.4			
Burndown charts	N/A	5.4.2.1		5.4, 5.7		7.8
Business architecture techniques	N/A	4.2.2.1	4.2			
Business capability analysis	N/A	4.2.2.2	4.2			
Business rules catalog	4.10.9.1	7.3.2.1				7.3
Capability framework	2.4.6	4.2.2.3	4.2			
Capability table	2.4.5.1	4.2.2.4	4.2, 4.3			

# ПРАКТИКИ PMI-BA



## BABOK®3 Knowledge Areas vs Techniques Matrix



<http://www.mindmeister.com/416093663>

## Business Analysis Planning &amp; Monitoring

- 3.1.6 Plan Business Analysis Approach
- 3.2.6 Plan Stakeholder Engagement
- 3.3.6 Plan Business Analysis Information Management
- 3.4.6 Plan Business Analysis Governance
- 3.5.6 Identify Business Analysis Performance Improvements

## Elicitation and Collaboration

- 4.1.6 Prepare for Elicitation
- 4.2.6 Conduct Elicitation
- 4.3.6 Confirm Elicitation Results
- 4.4.6 Communicate Business Analysis Information
- 4.5.6 Manage Stakeholder Collaboration

## Requirements Life Cycle Management

- 5.1.6 Trace Requirements
- 5.2.6 Maintain Requirements
- 5.3.6 Prioritize Requirements
- 5.4.6 Assess Requirements Changes
- 5.5.6 Gain Consensus

## Strategy Analysis

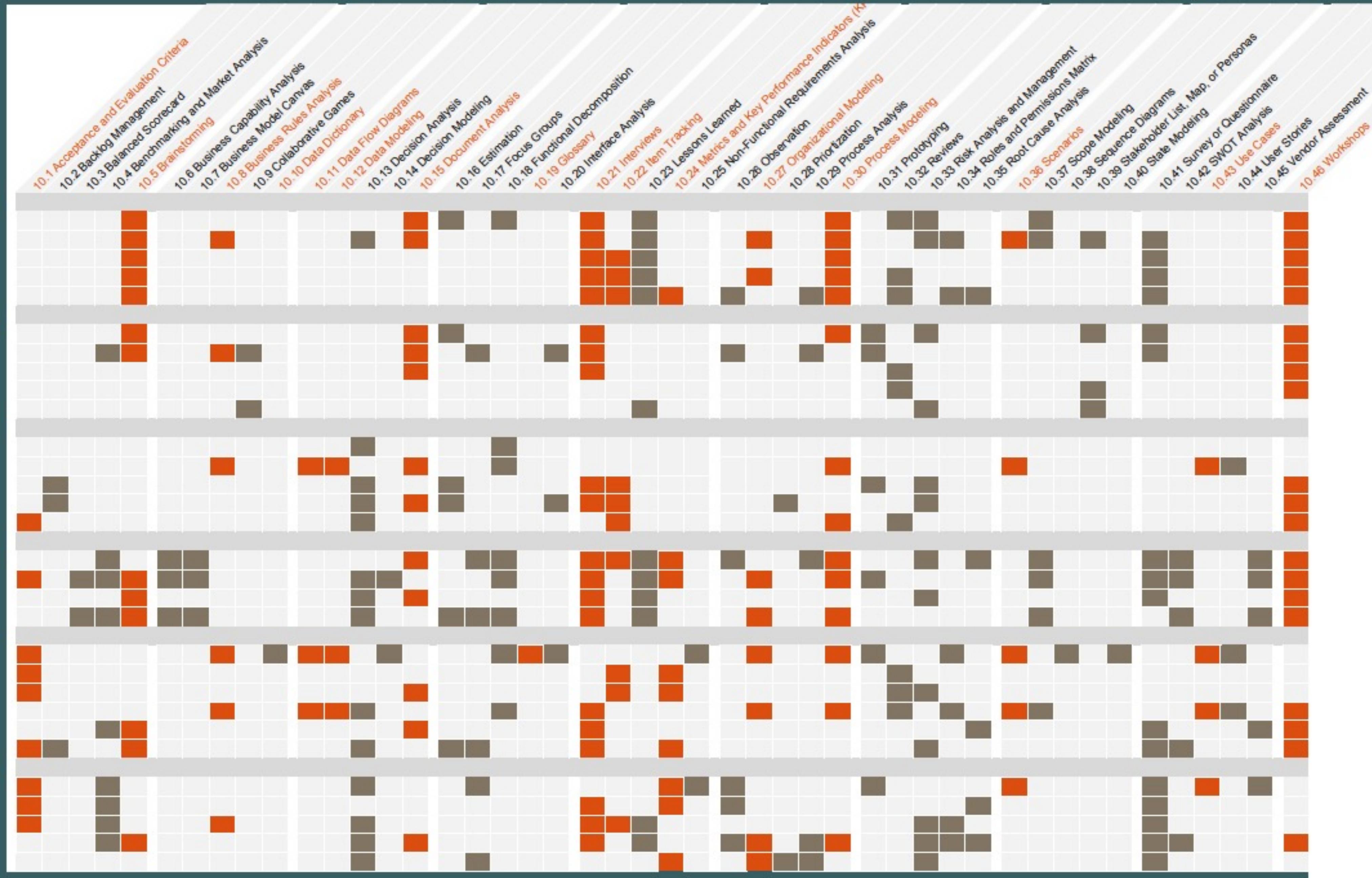
- 6.1.6 Analyze Current State
- 6.2.6 Define Future State
- 6.3.6 Assess Risks
- 6.4.6 Define Change Strategy

## Requirements Analysis and Design Definition

- 7.1.6 Specify and Model Requirements
- 7.2.6 Verify Requirements
- 7.3.6 Validate Requirements
- 7.4.6 Define Requirements Architecture
- 7.5.6 Define Solution Options
- 7.6.6 Analyze Potential Value and Recommend Solution

## Solution Evaluation

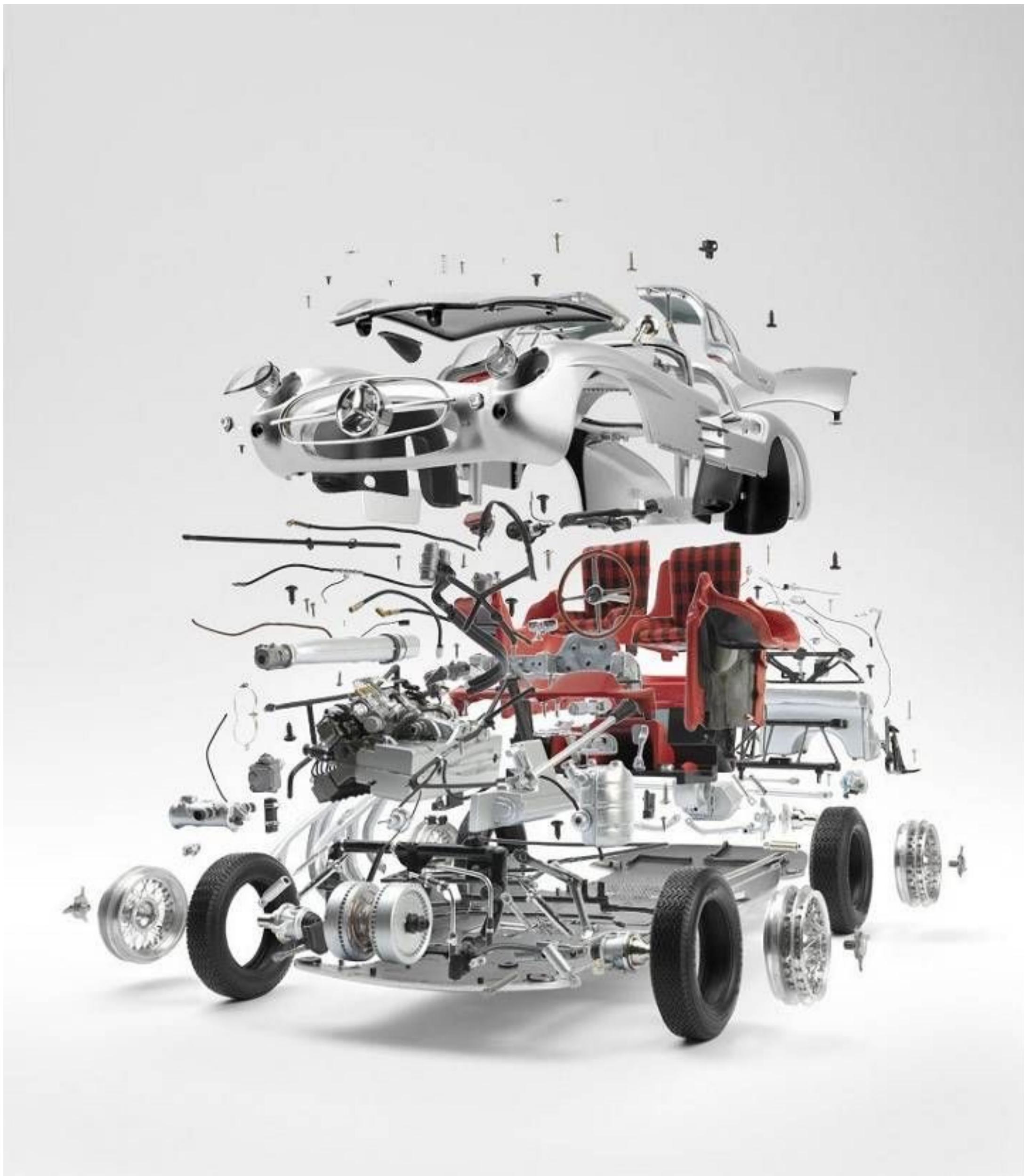
- 8.1.6 Measure Solution Performance
- 8.2.6 Analyze Performance Measures
- 8.3.6 Assess Solution Limitations
- 8.4.6 Assess Enterprise Limitations
- 8.5.6 Recommend Actions to Increase Solution Value



# КАК СОБРАТЬ?

- Много практик
- Некоторые несовместимы
- Разный жизненный цикл
- Высокая сложность

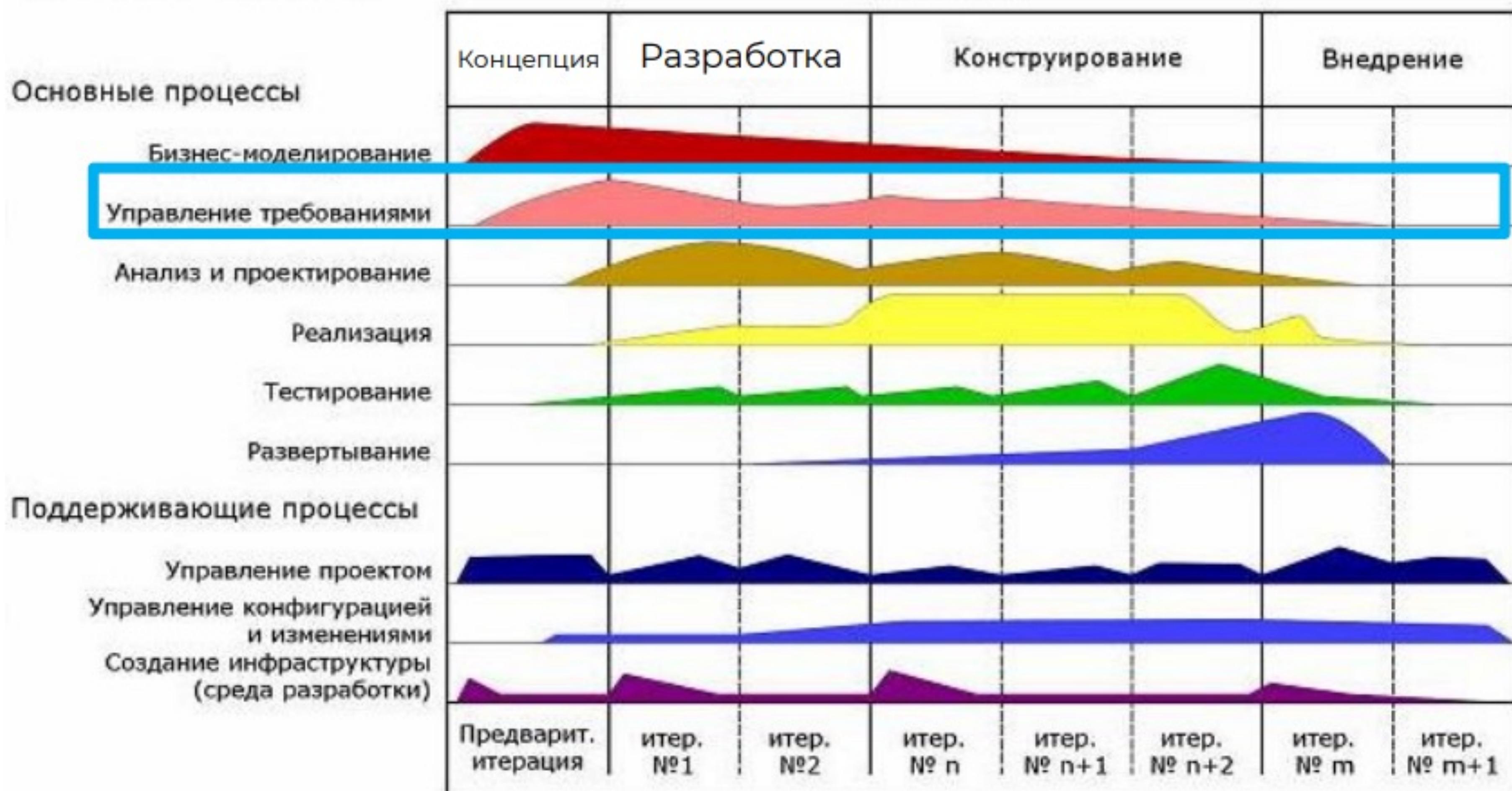
...должен быть метод сборки  
в хорошо увязанный проект



# КАК ВЫГЛЯДИТ В ЖИЗНИ

## Рабочие процессы

## Стадии





Enterprise  
Architecture



People  
Management



Information  
Technology



Asset  
Management



Transformation



Finance



Vendor  
Management



Legal



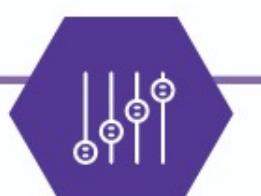
Research &  
Development



Business  
Operations



Strategy



Governance



Marketing



Continuous  
Improvement



Sales



Portfolio  
Management



Product  
Management



Program  
Management



Disciplined  
DevOps



Disciplined  
Agile Delivery  
(DAD)



Security



Data  
Management



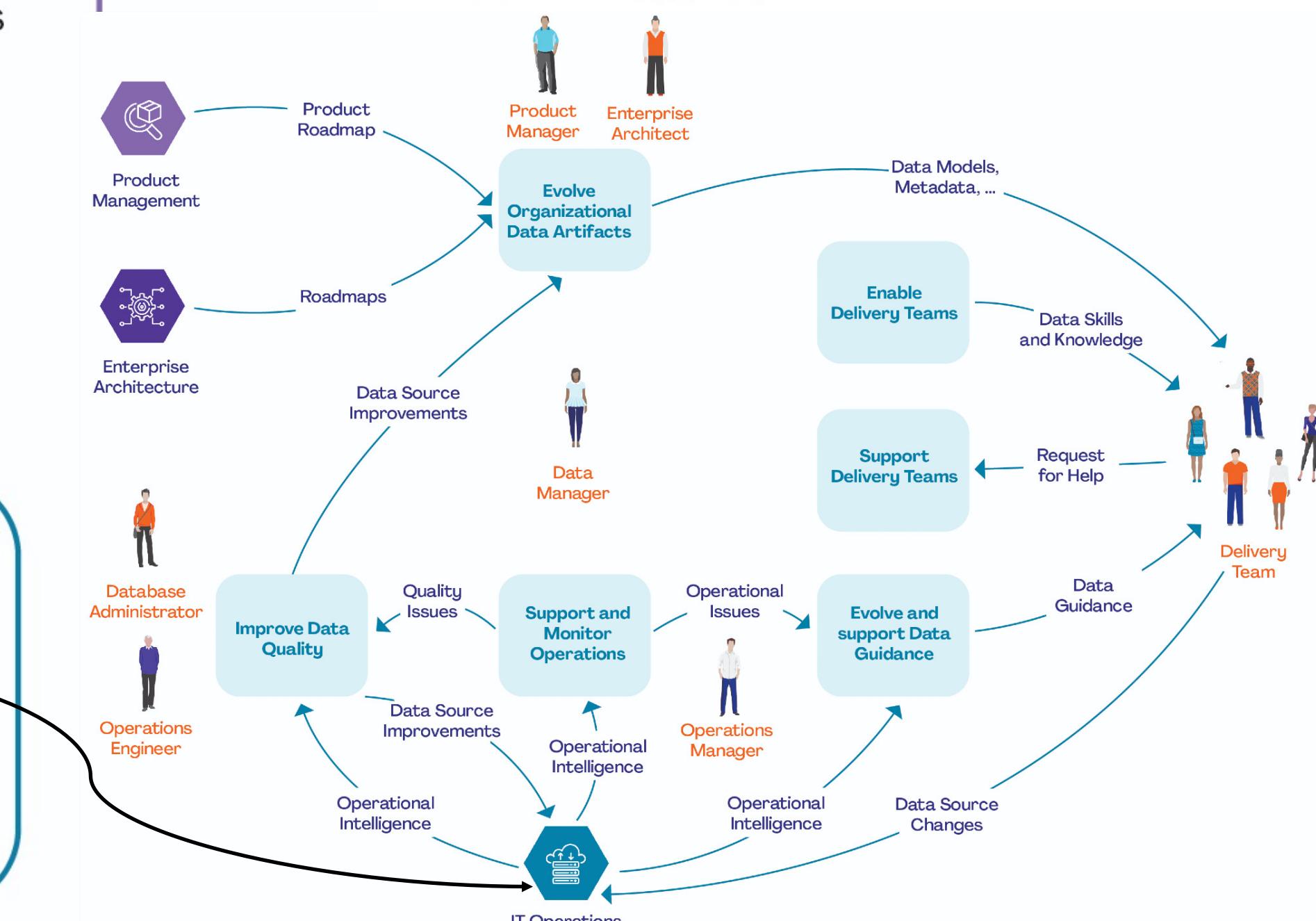
Release  
Management



Support



IT Operations



Disciplined Agile Enterprise



Principles



Promises



Guidelines



Agile



Lean



Serial



Roles

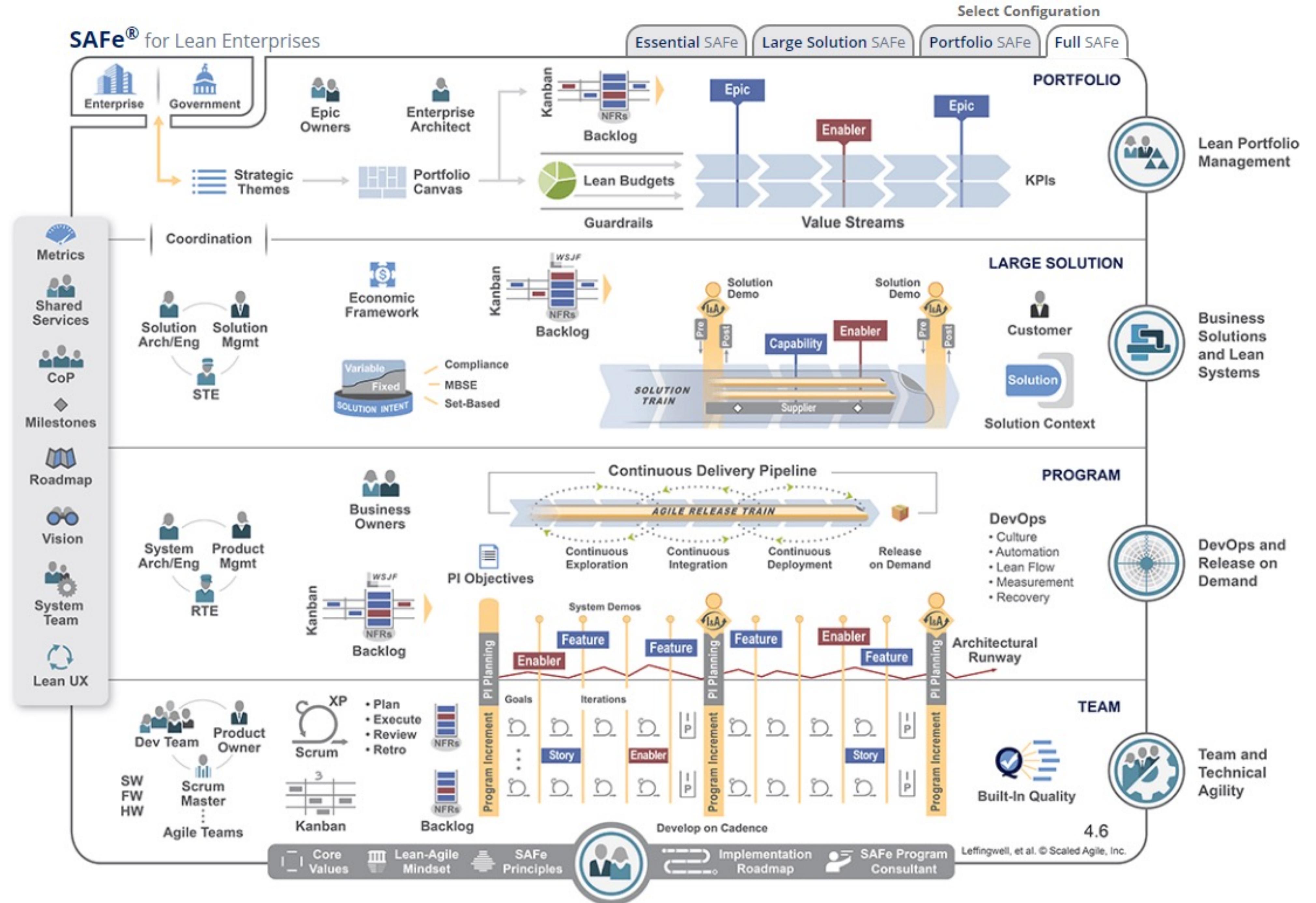


Teams

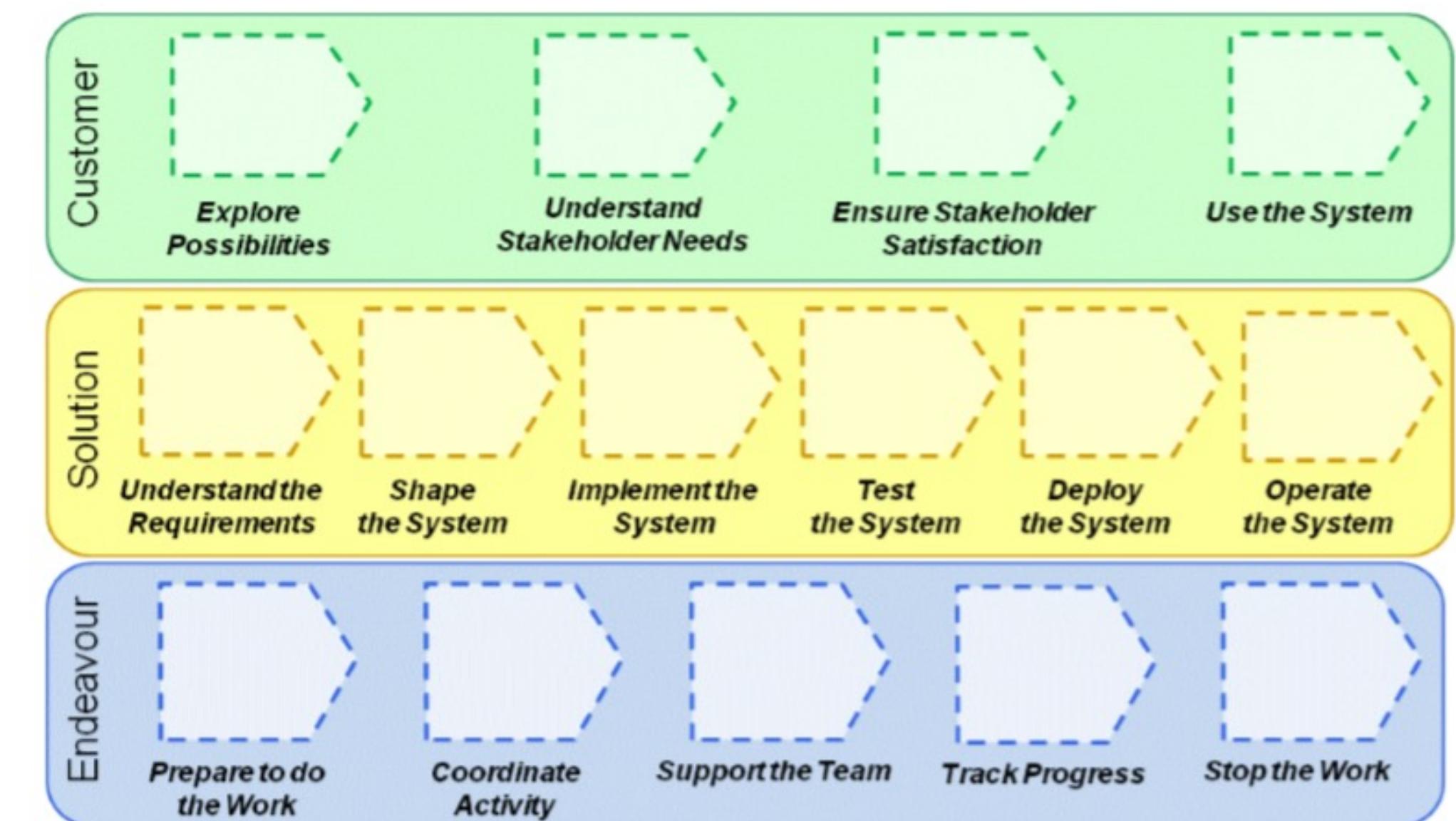
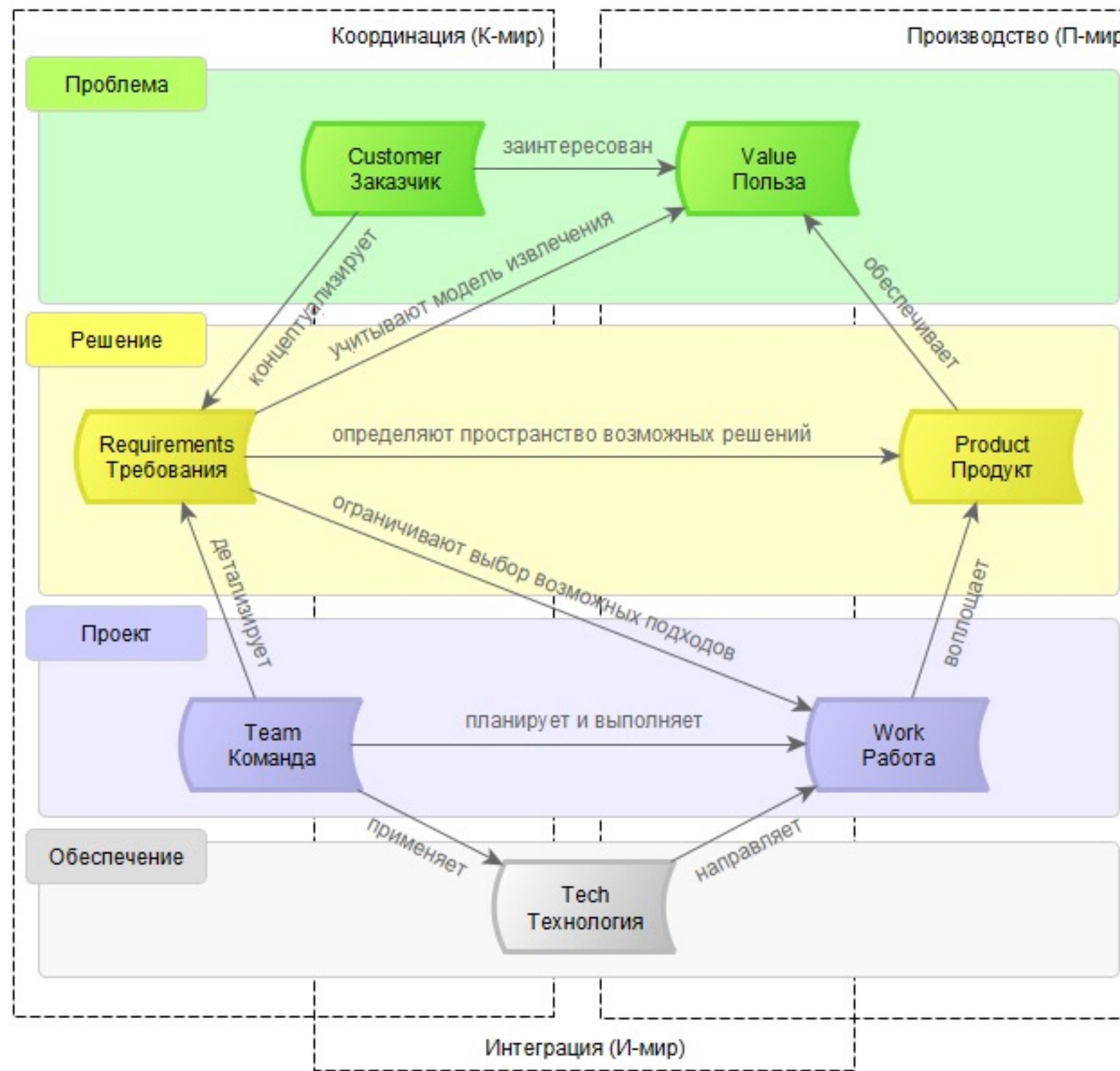


Way of Working  
(WOW)

Foundation



# СОЗДАНИЕ МЕТОДА: OMG ESSENCE



# ОБЩИЕ ПРИНЦИПЫ РАБОТЫ

## Алгоритм работы с требованиями

- 1. Выявление потребностей** стейкхолдеров — определение проблем, которые необходимо решить.
- 2. Разработка требований стейкхолдеров** — анализ, разбор требований, выделение ключевых потребностей, обобщение.
- 3. Разработка требований к системе** — оценка требований в контексте возможностей и существующих бизнес-процессов разработчиков.
- 4. Документирование** — формальное и подробное описание требований, например, согласование ТЗ.
- 5. Трассировка требований на архитектуру** — итеративное взаимодействие с разработчиками по поводу доработок, например, согласно составленному ТЗ.
- 6. Реализация** — работа разработчиков над созданием необходимой функциональности.
- 7. Верификация и валидация** — проверка функциональности внутренними экспертами и конечными пользователями с целью установления соответствия разработки и ТЗ, работоспособности системы внутри использующей системы.

**А ТЕПЕРЬ ПРАКТИКА**

# ТРЕНИНГ: СБОР ТРЕБОВАНИЙ

Сформулировать требования к Умной колонке.

Кидаем формулировки в общий чат, смотрим что получилось здесь:



<https://disk.yandex.ru/i/LaeLCHbX16EDfg>

tbc...