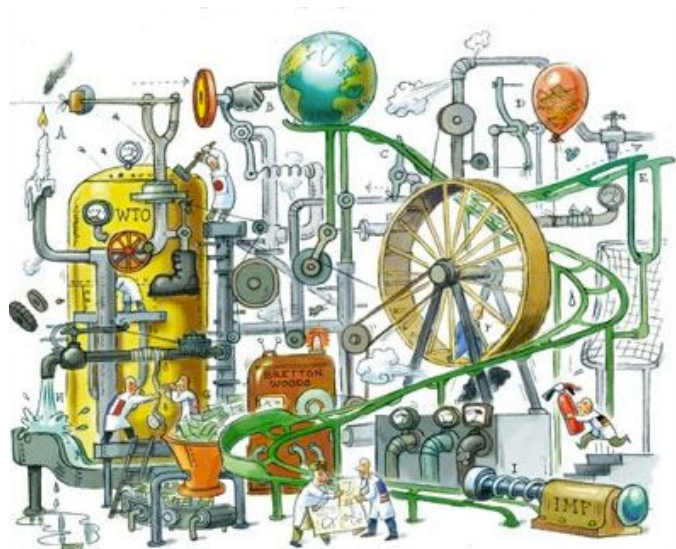




**ЭКСПОНЕНТА**

ЦЕНТР ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ  
И МОДЕЛИРОВАНИЯ

# Системная инженерия и верификация



---

Михаил Песельник  
ЦИТМ Экспонента

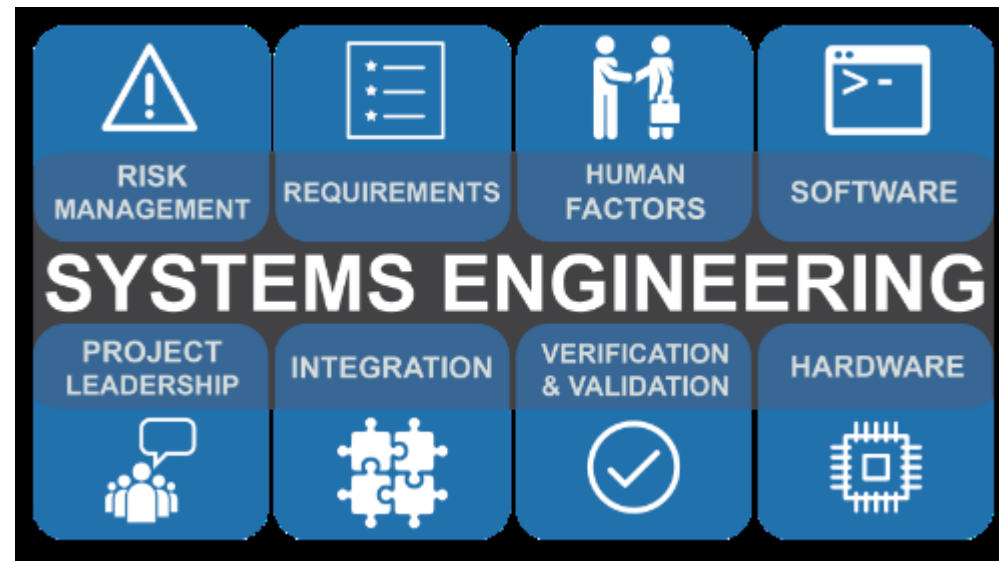
# Цель

- Понять суть системной инженерии
- Подумать о пользе системной инженерии
- Увидеть инструменты MathWorks для системной инженерии

# План

- **Что такое системная инженерия**
- **Инструменты для системной инженерии**

# Что такое системная инженерия?



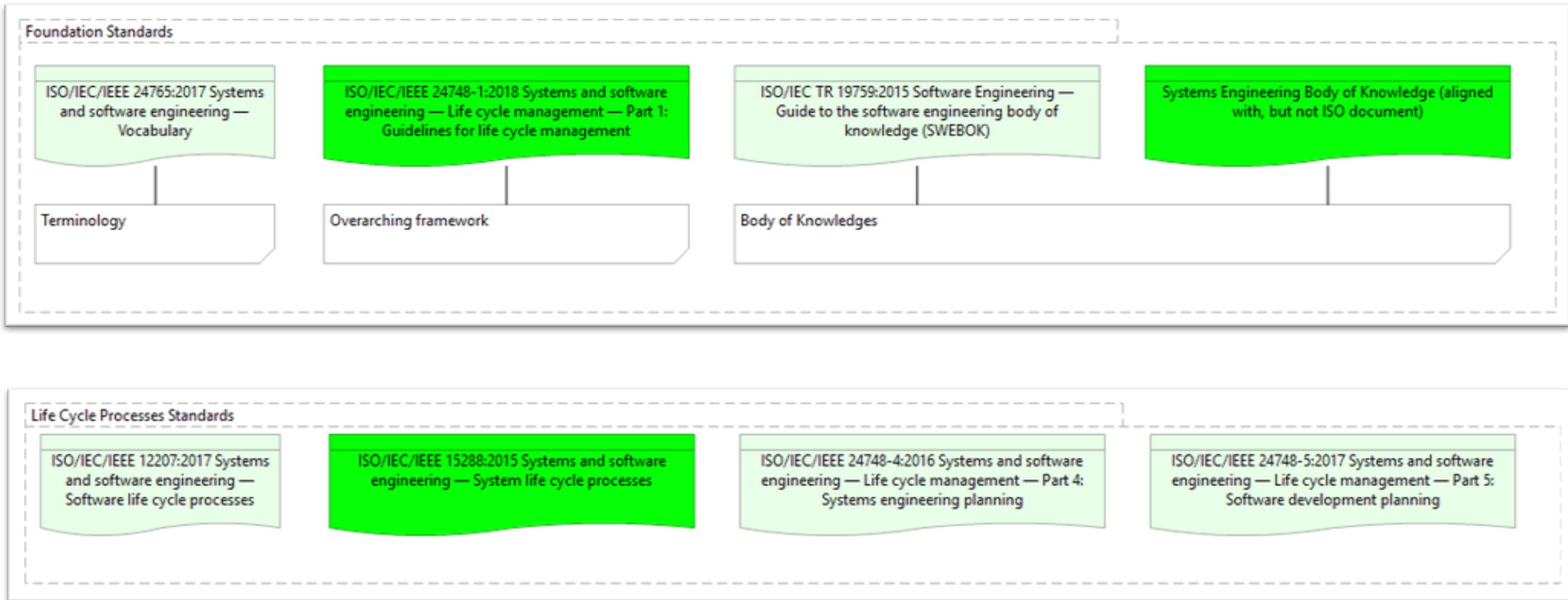
<https://www.incose.org/systems-engineering>

# Зачем нужна системная инженерия?

Системная инженерия - это **управление сложностью**.  
Системная инженерия помогает избежать упущений и неправильных допущений, помогает решать проблемы, возникающие в реальном мире, и дает наиболее эффективное, экономичное и надежное решение.

[https://www.incose.org/docs/default-source/default-document-library/twg-se101-v11-2014-01-20.pdf?sfvrsn=e6c882c6\\_4](https://www.incose.org/docs/default-source/default-document-library/twg-se101-v11-2014-01-20.pdf?sfvrsn=e6c882c6_4)

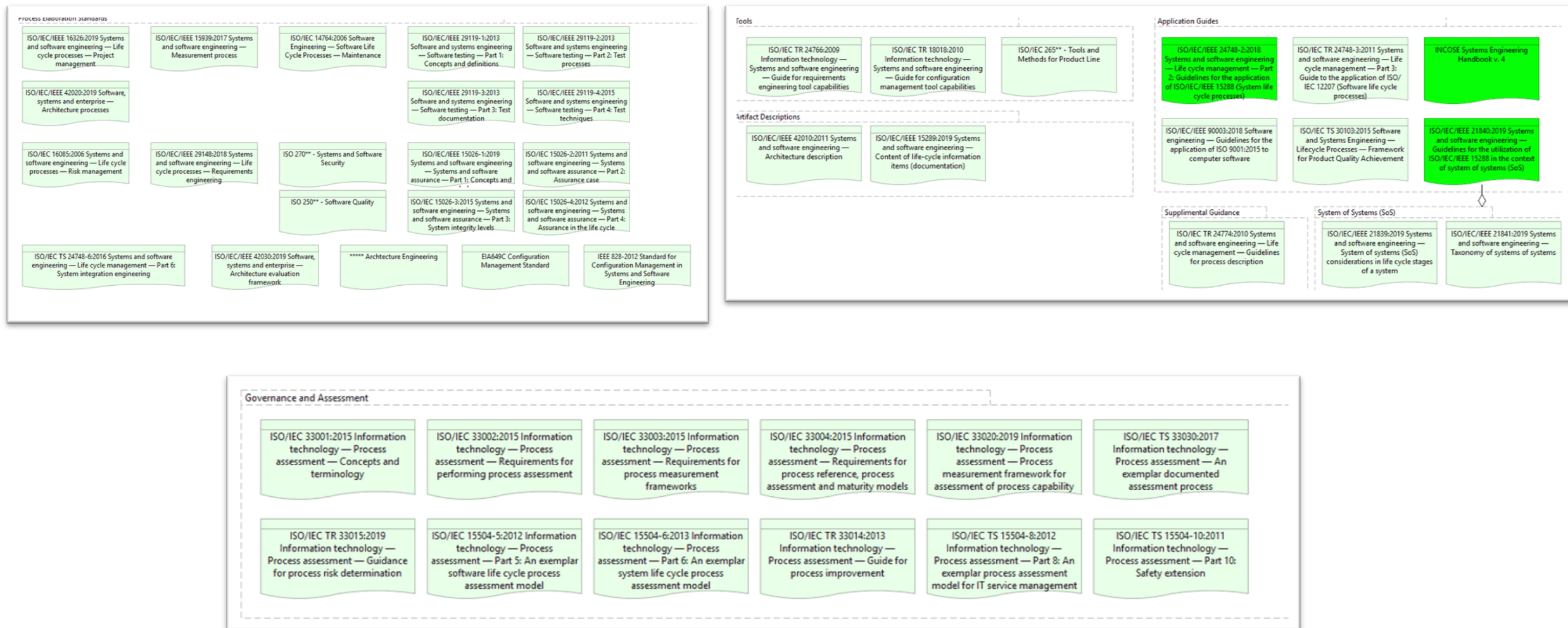
# Семейство стандартов



<https://github.com/alef1986/Systems-Engineering-Standards>

<https://alef1986.github.io/Systems-Engineering-Standards>

# Семейство стандартов



<https://github.com/alef1986/Systems-Engineering-Standards>

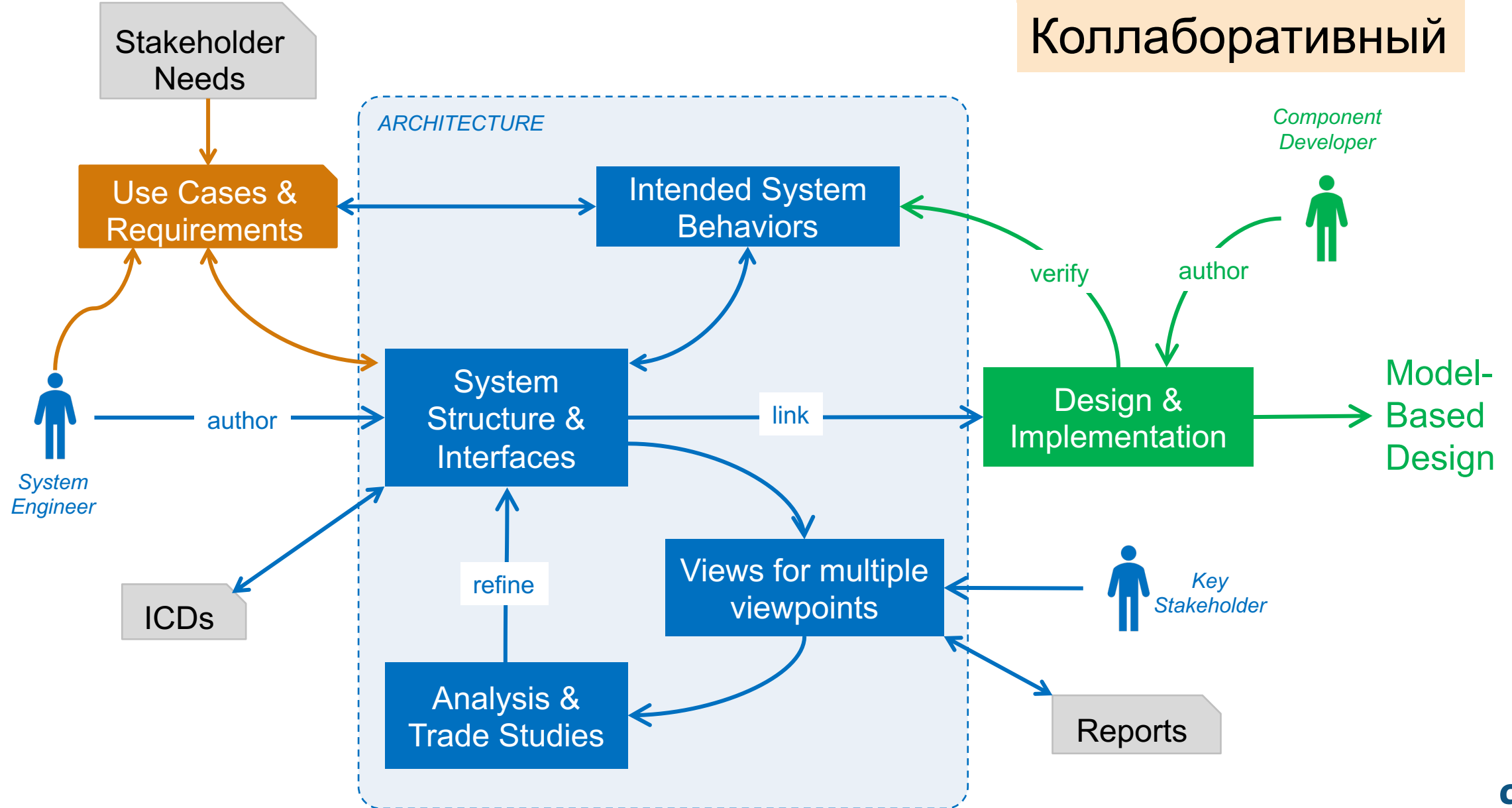
<https://alef1986.github.io/Systems-Engineering-Standards>

# Про системную инженерию

- International Council on Systems Engineering (INCOSE):  
<https://www.incose.org>
- Основные документы:
  - INCOSE Systems Engineering Handbook: <https://www.gfse-handbuch.de/1/incose-systems-engineering-handbuch>
  - Body of Knowledge and Curriculum to Advance Systems Engineering (BKCASE): <https://www.incose.org/products-and-publications/se-body-of-knowledge>
  - ГОСТ Р ИСО/МЭК 15288 – Системная инженерия. Процессы жизненного цикла систем.

# Как выглядит типичный процесс?

Итеративный  
Коллаборативный



Stakeholder  
Needs

Use Cases &  
Requirements

System Structure  
& Interfaces

Intended System  
Behaviors

Views for multiple  
viewpoints

Analysis & Trade  
Studies

Design &  
Implementation

# Как работает системная инженерия и что мы получаем в результате?

Stakeholder  
Needs

Use Cases &  
Requirements

System Structure  
& Interfaces

Intended System  
Behaviors

Views for multiple  
viewpoints

Analysis & Trade  
Studies

Design &  
Implementation

## Система открывания двери гаража



# Начинаем с базового набора потребностей заинтересованной стороны

Stakeholder  
Needs

Use Cases &  
Requirements

System Structure  
& Interfaces

Intended System  
Behaviors

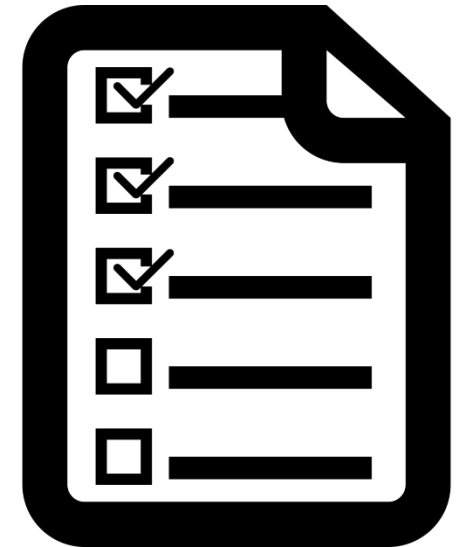
Views for multiple  
viewpoints

Analysis & Trade  
Studies

Design &  
Implementation

Я хочу  
открывать  
дверь из авто

Мне нужен  
пульт  
управления



# Юзкейсы и сценарии определяют, как будет использоваться система

Stakeholder Needs

Use Cases & Requirements

System Structure & Interfaces

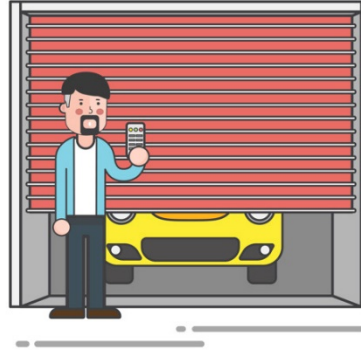
Intended System Behaviors

Views for multiple viewpoints

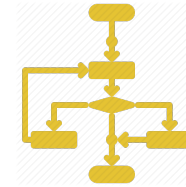
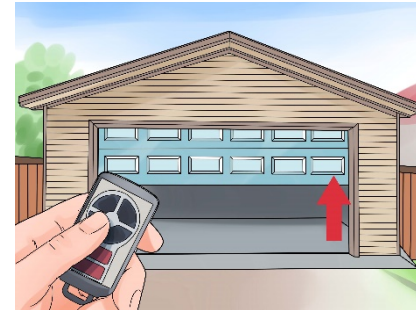
Analysis & Trade Studies

Design & Implementation

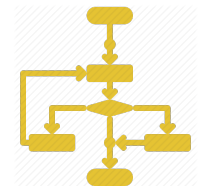
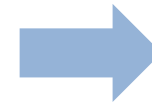
Юзкейс: Человек с Пультom должен управлять дверью



Сценарий работы 1:  
Открыть дверь  
Пультom с  
авторизованным  
пультom



Сценарий работы 2:  
Открыть дверь  
Пультom с  
неавторизованным  
пультom



# ЧТО должна делать система, КТО совершает действие и КАК построена система

Stakeholder  
Needs

Use Cases &  
Requirements

System Structure  
& Interfaces

Intended System  
Behaviors

Views for multiple  
viewpoints

Analysis & Trade  
Studies

Design &  
Implementation

Функциональная  
архитектура  
«ЧТО»

Close

Obstacle  
Detection

Open

Door  
Status

Remote  
Command

Validate  
Remote

Add  
Remote

Reset  
Remotes

# ЧТО должна делать система, КТО совершает действие и КАК построена система

Stakeholder  
Needs

Use Cases &  
Requirements

System Structure  
& Interfaces

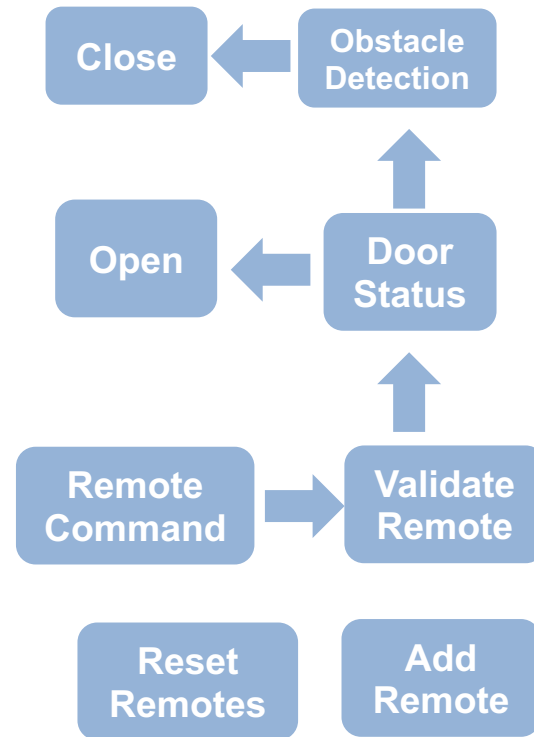
Intended System  
Behaviors

Views for multiple  
viewpoints

Analysis & Trade  
Studies

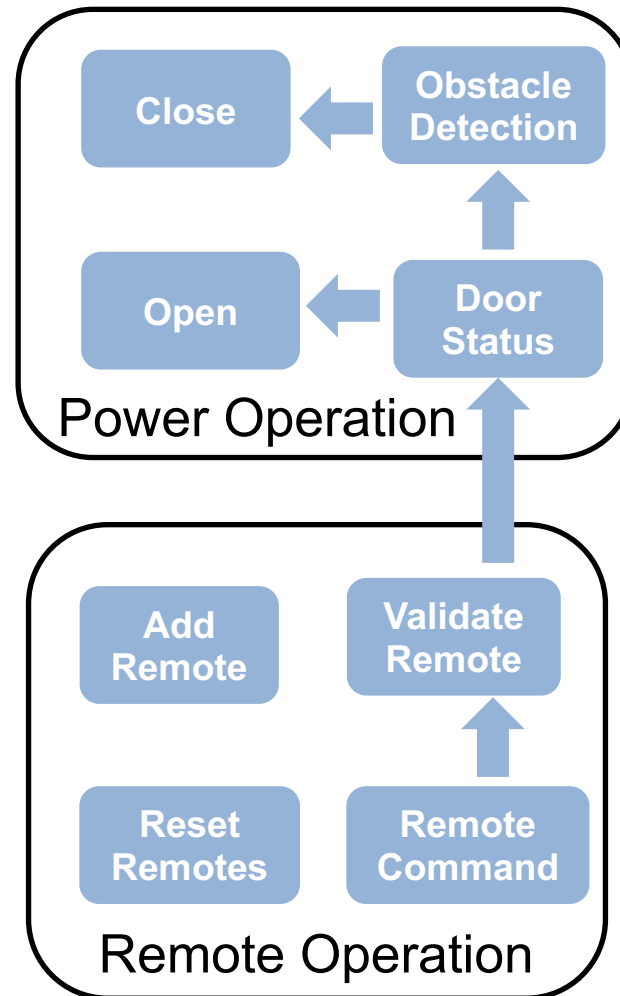
Design &  
Implementation

Функциональная  
архитектура  
«ЧТО»



# ЧТО должна делать система, КТО совершает действие и КАК построена система

Логическая архитектура  
«КТО»



Stakeholder  
Needs

Use Cases &  
Requirements

System Structure  
& Interfaces

Intended System  
Behaviors

Views for multiple  
viewpoints

Analysis & Trade  
Studies

Design &  
Implementation

# ЧТО должна делать система, КТО совершает действие и КАК построена система

Stakeholder  
Needs

Use Cases &  
Requirements

System Structure  
& Interfaces

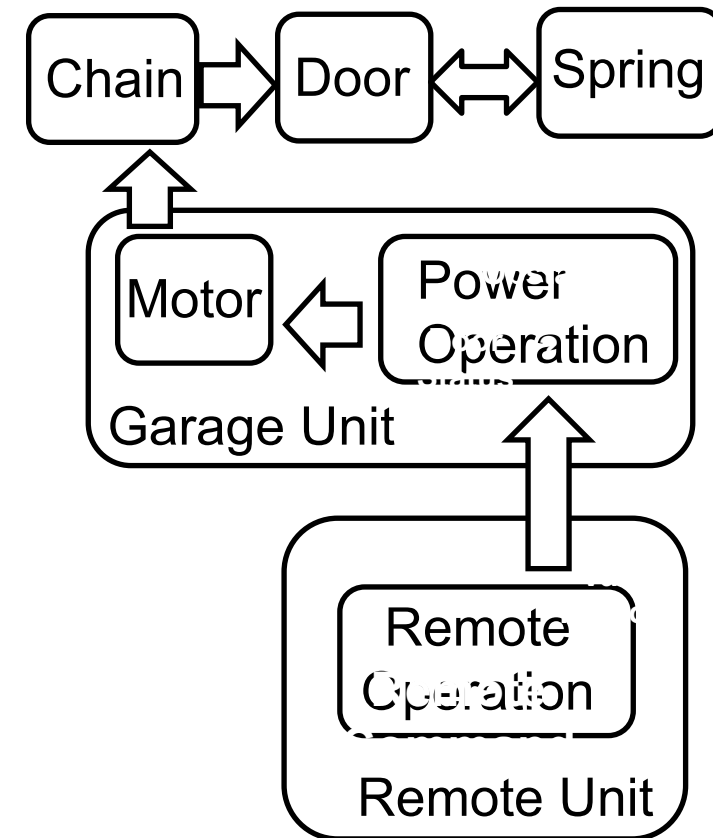
Intended System  
Behaviors

Views for multiple  
viewpoints

Analysis & Trade  
Studies

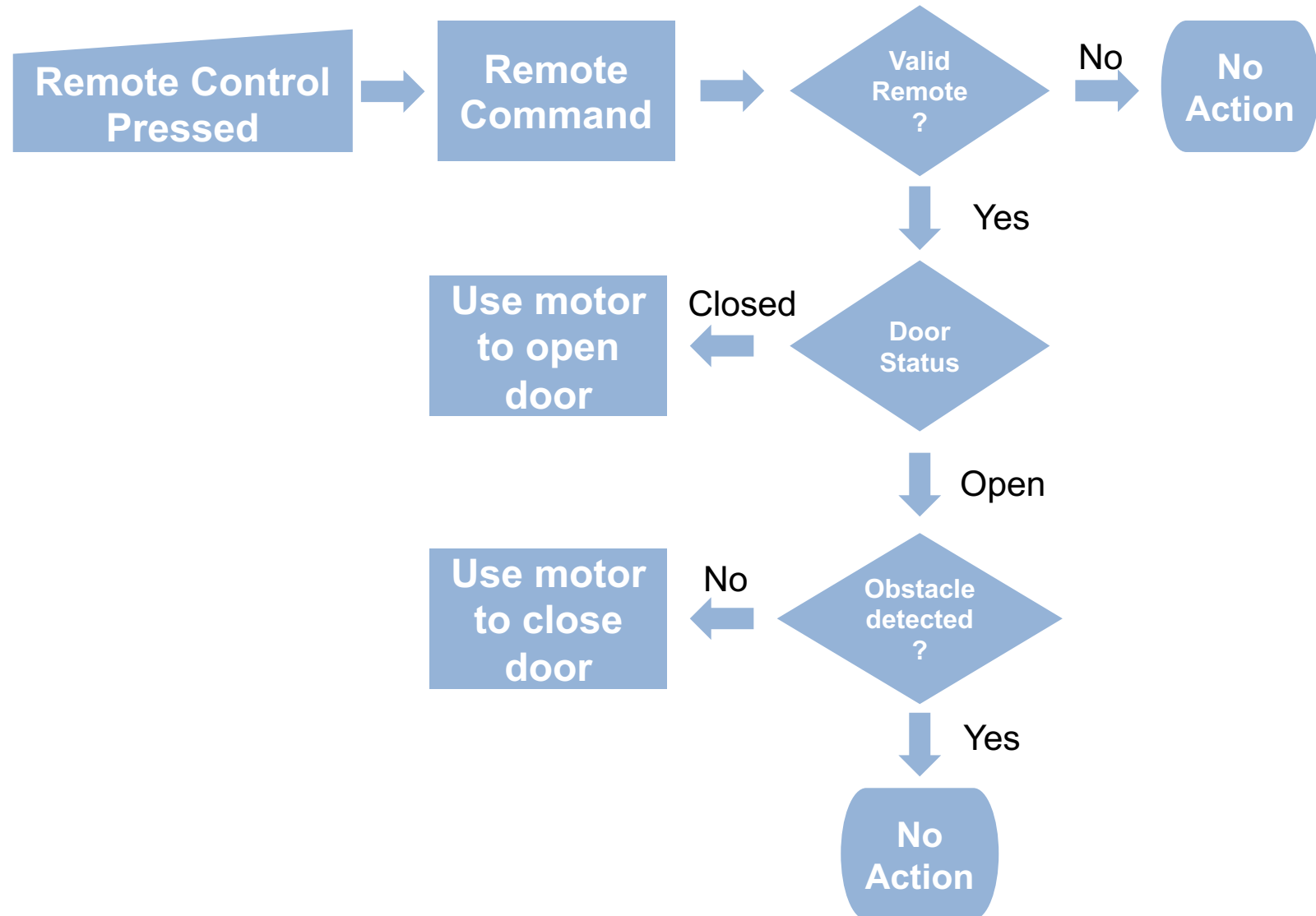
Design &  
Implementation

Физическая  
архитектура  
«КАК»



# Предназначенное поведение системы

Поведение при доступе с пульта



Stakeholder  
Needs

Use Cases &  
Requirements

System Structure  
& Interfaces

Intended System  
Behaviors

Views for multiple  
viewpoints

Analysis & Trade  
Studies

Design &  
Implementation

# Представления используются, чтобы показать систему с определенной точки зрения

Stakeholder  
Needs

Use Cases &  
Requirements

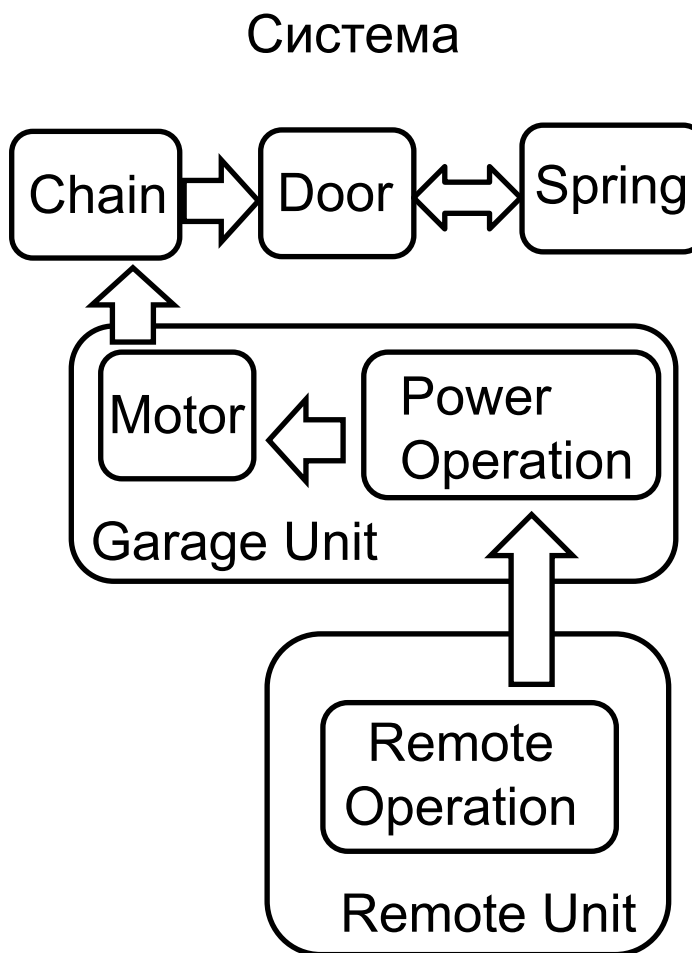
System Structure  
& Interfaces

Intended System  
Behaviors

Views for multiple  
viewpoints

Analysis & Trade  
Studies

Design &  
Implementation



# Представления используются, чтобы показать систему с определенной точки зрения

Stakeholder  
Needs

Use Cases &  
Requirements

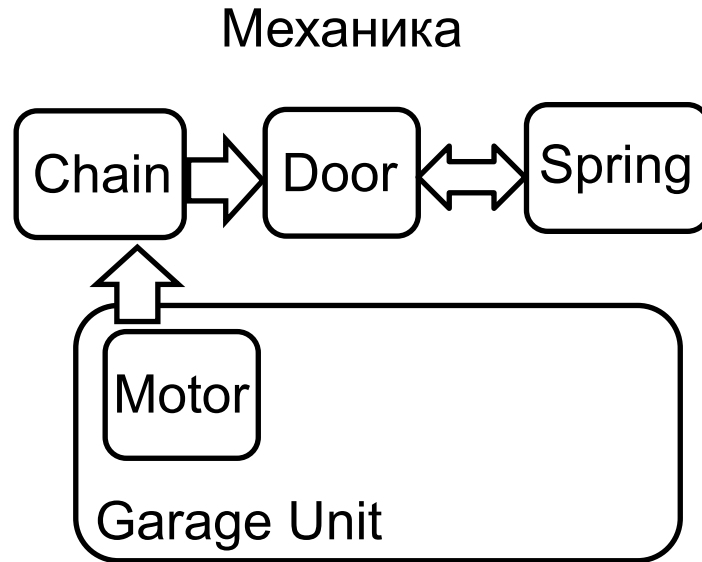
System Structure  
& Interfaces

Intended System  
Behaviors

Views for multiple  
viewpoints

Analysis & Trade  
Studies

Design &  
Implementation



# Представления используются, чтобы показать систему с определенной точки зрения

Stakeholder  
Needs

Use Cases &  
Requirements

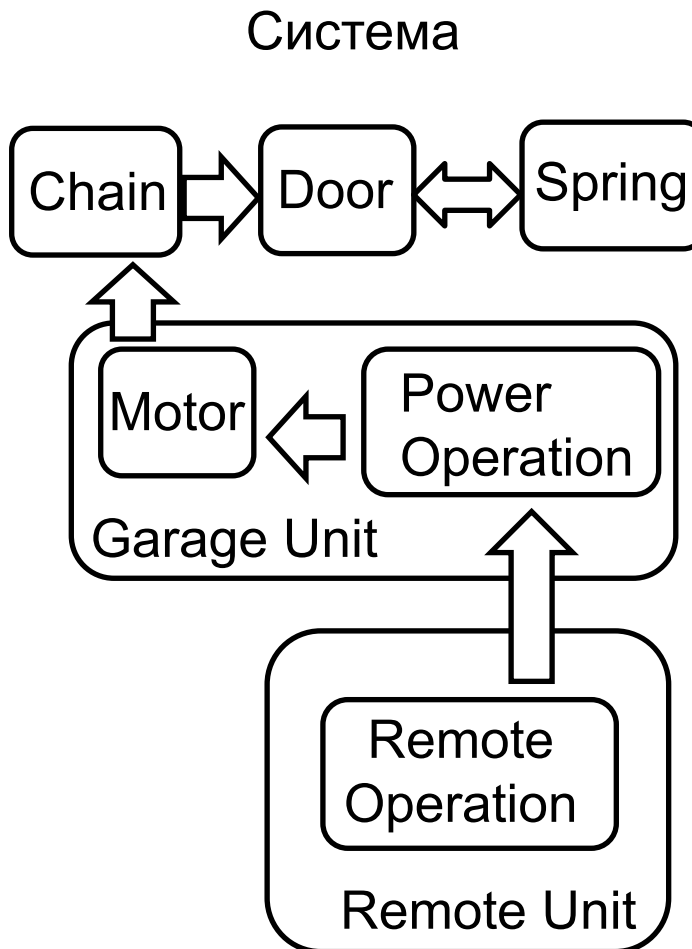
System Structure  
& Interfaces

Intended System  
Behaviors

Views for multiple  
viewpoints

Analysis & Trade  
Studies

Design &  
Implementation



# Представления используются, чтобы показать систему с определенной точки зрения

Stakeholder  
Needs

Use Cases &  
Requirements

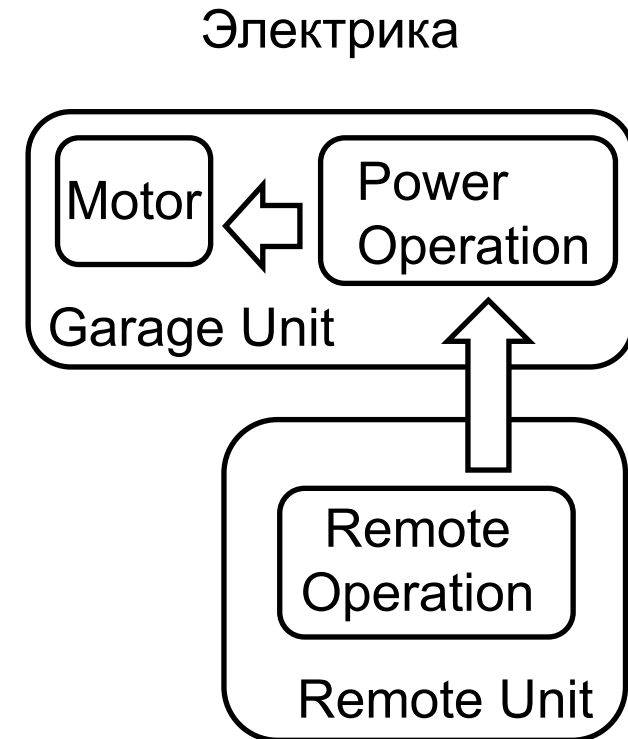
System Structure  
& Interfaces

Intended System  
Behaviors

Views for multiple  
viewpoints

Analysis & Trade  
Studies

Design &  
Implementation



# Представления используются, чтобы показать систему с определенной точки зрения

Stakeholder Needs

Use Cases & Requirements

System Structure & Interfaces

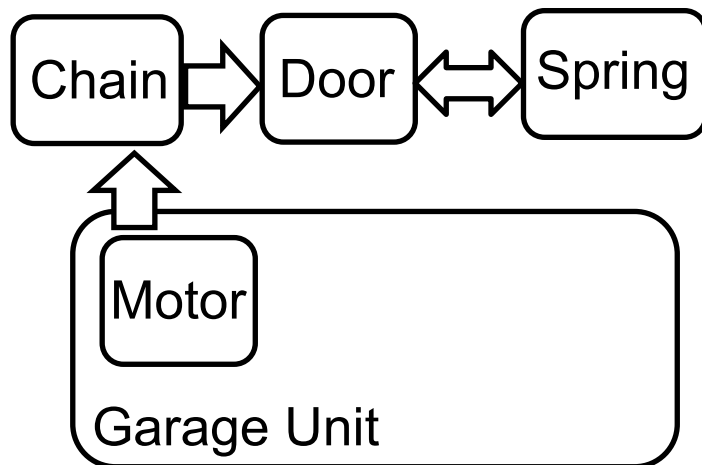
Intended System Behaviors

Views for multiple viewpoints

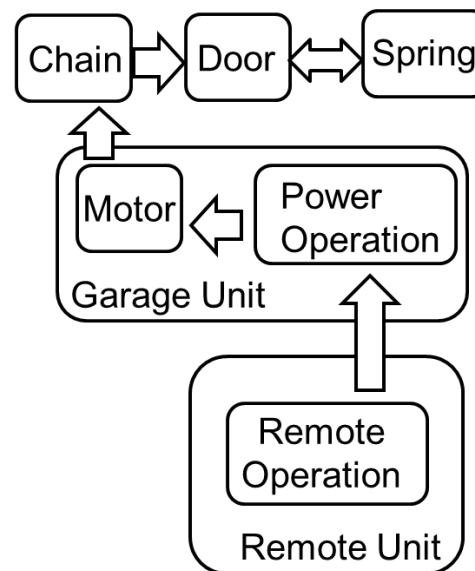
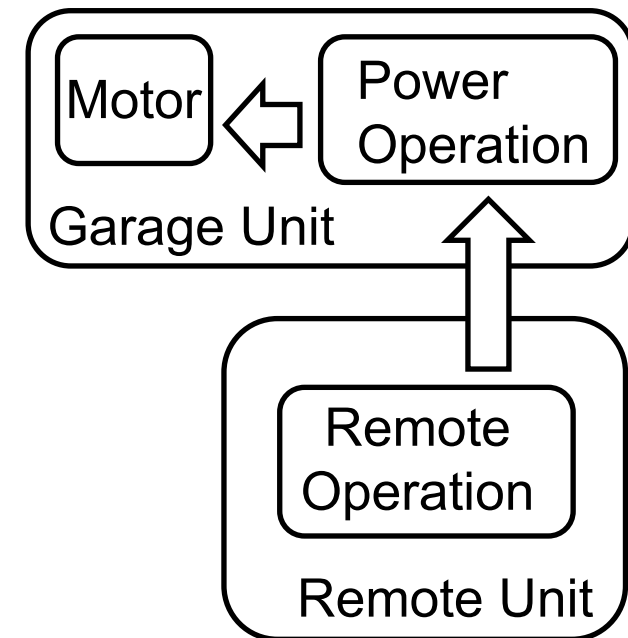
Analysis & Trade Studies

Design & Implementation

Механика



Электрика



# Анализ, поиск компромиссов и сравнение различных системных компонентов и конфигураций

Stakeholder  
Needs

Use Cases &  
Requirements

System Structure  
& Interfaces

Intended System  
Behaviors

Views for multiple  
viewpoints

Analysis & Trade  
Studies

Design &  
Implementation



# Проектирование и реализация системы

Stakeholder  
Needs

Use Cases &  
Requirements

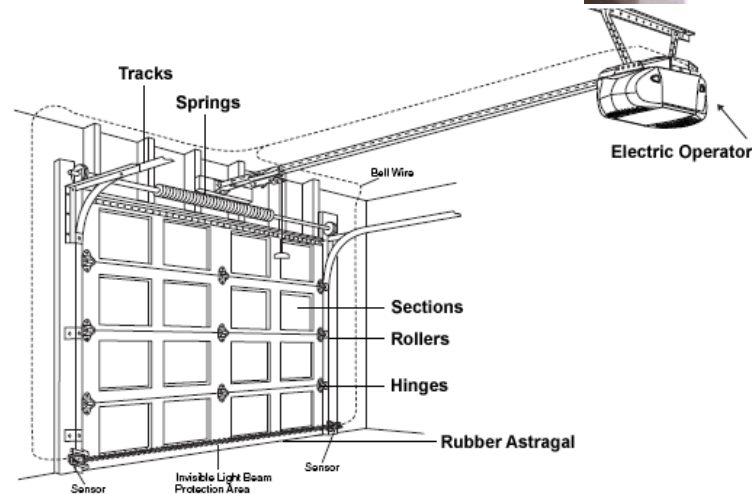
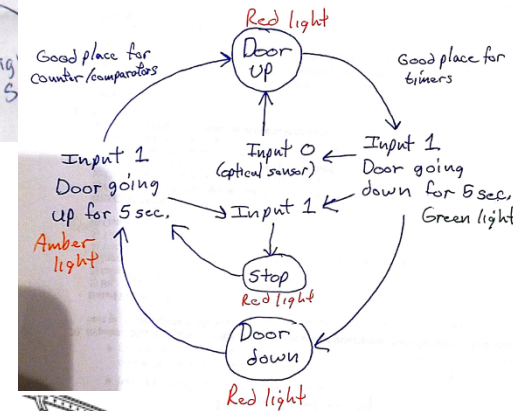
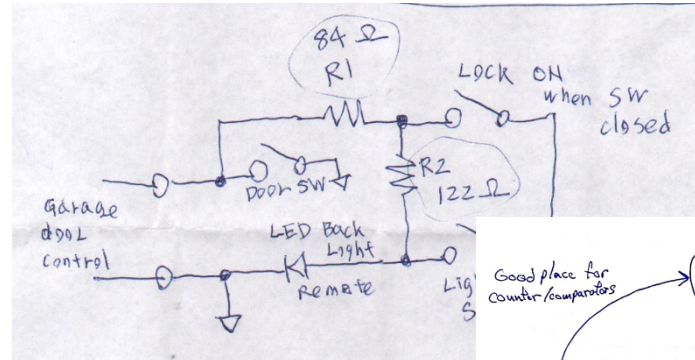
System Structure  
& Interfaces

Intended System  
Behaviors

Views for multiple  
viewpoints

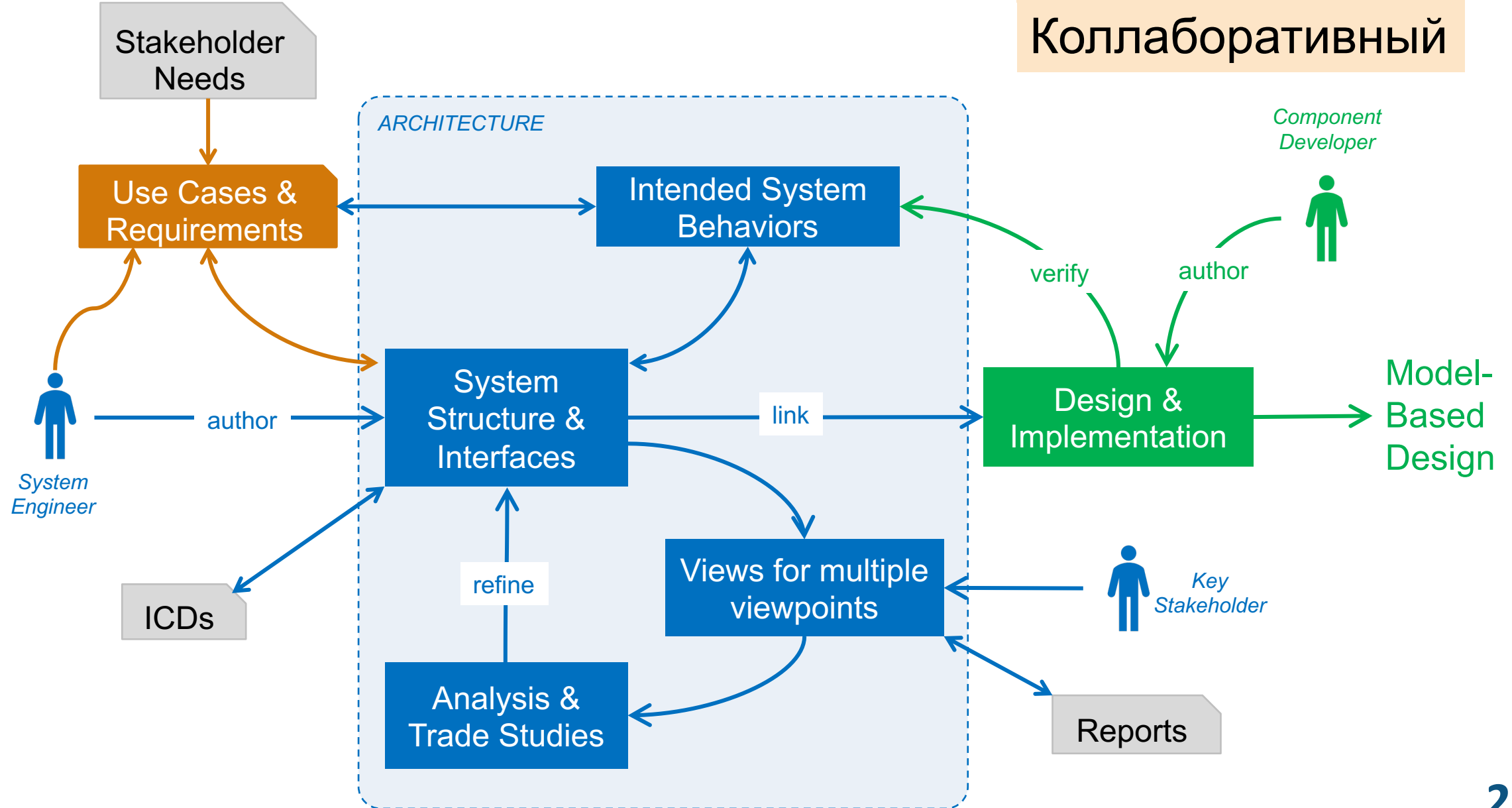
Analysis & Trade  
Studies

Design &  
Implementation

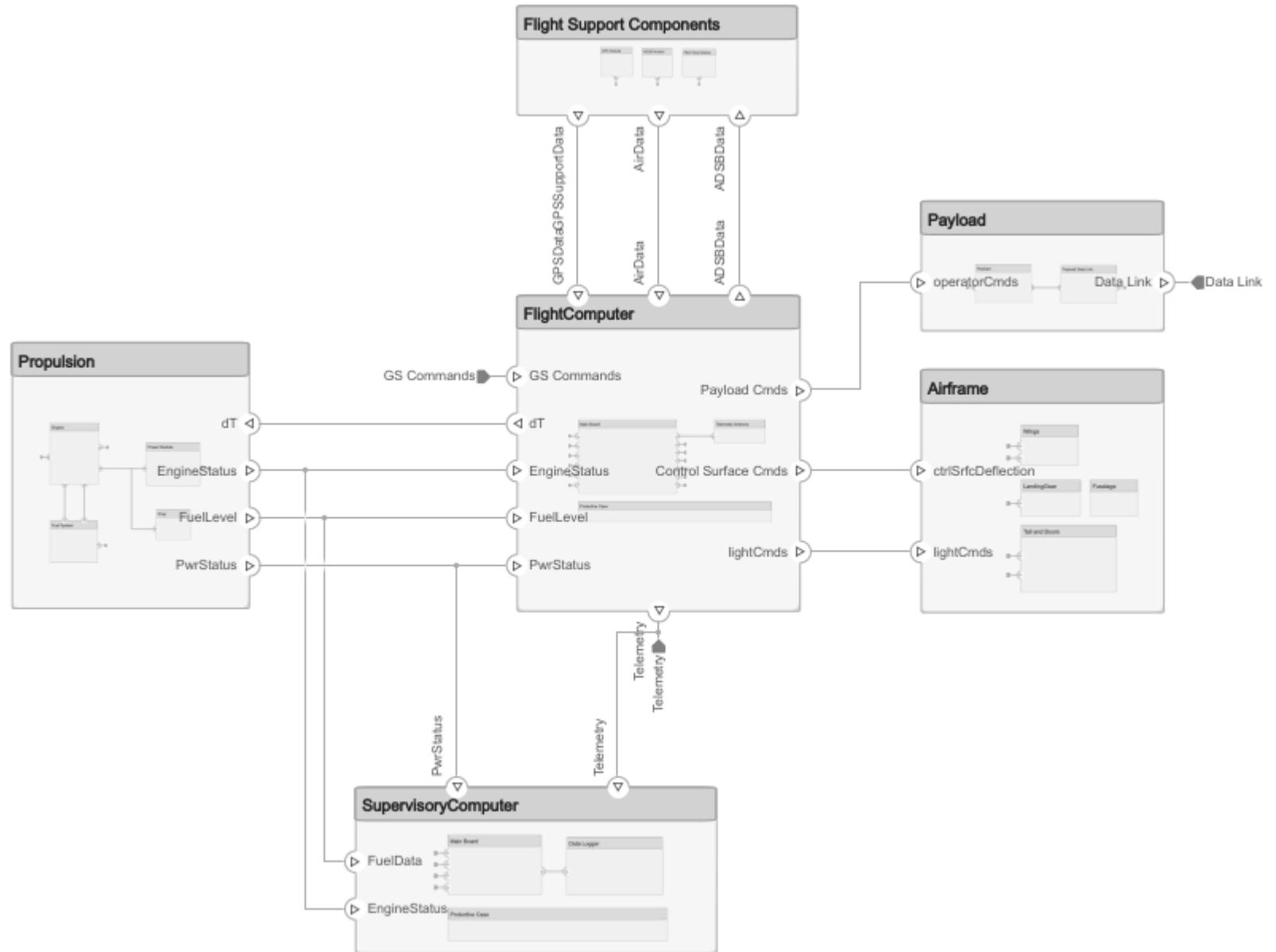


# Как выглядит типичный процесс?

Итеративный  
Коллаборативный



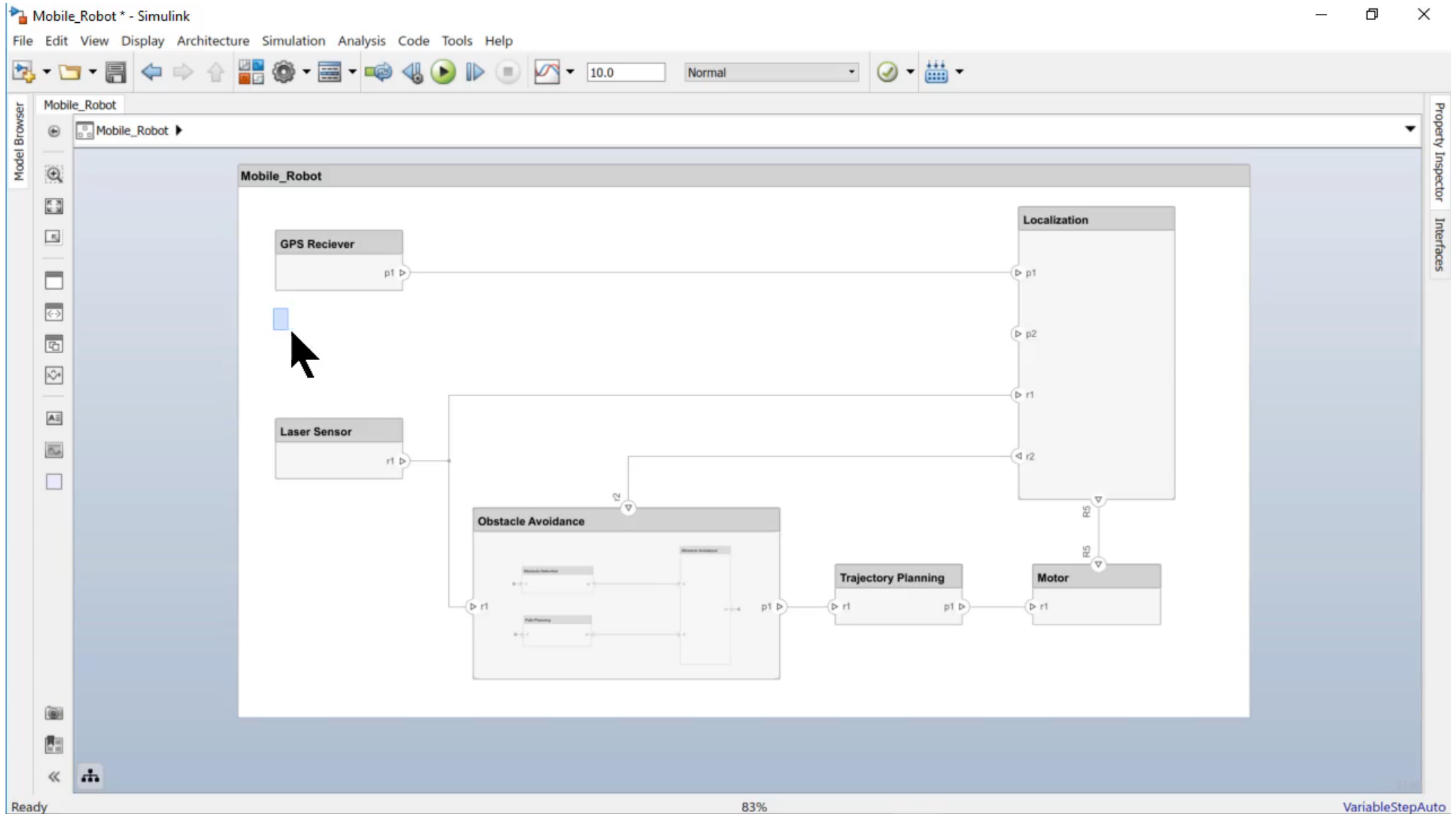
# System Composer



# Введение в System Composer

- Определение и анализ архитектуры и композиции
- Привязка к требованиям и проектирование в Simulink
- Пользовательские представления
- Анализ свойств и требований
- Создание спецификаций и интерфейсов

# Создание иерархической модели системы



# Управление интерфейсами

The screenshot displays the Simulink environment for a model named 'SmallUAV'. The main workspace shows a block diagram with a central 'Flight Computer' block. It is connected to several other blocks: 'Flight Expert Components' above, 'Payload' and 'Antenna' to the right, 'Powertrain' to the left, and 'Supervisory Computer' below. A mouse cursor is pointing at the 'GS Commands' input port of the 'Flight Computer' block.

On the right side, the 'Property Inspector' and 'Interfaces' toolbars are visible.

**Property Inspector**

| Port             |                                      |
|------------------|--------------------------------------|
| NAME             | VALUE                                |
| <b>Main</b>      |                                      |
| Name             | GS Commands                          |
| <b>Interface</b> |                                      |
| Name             | <input type="text" value="REQUEST"/> |
| Action           | REQUEST                              |
| Stereotype       | Add...                               |

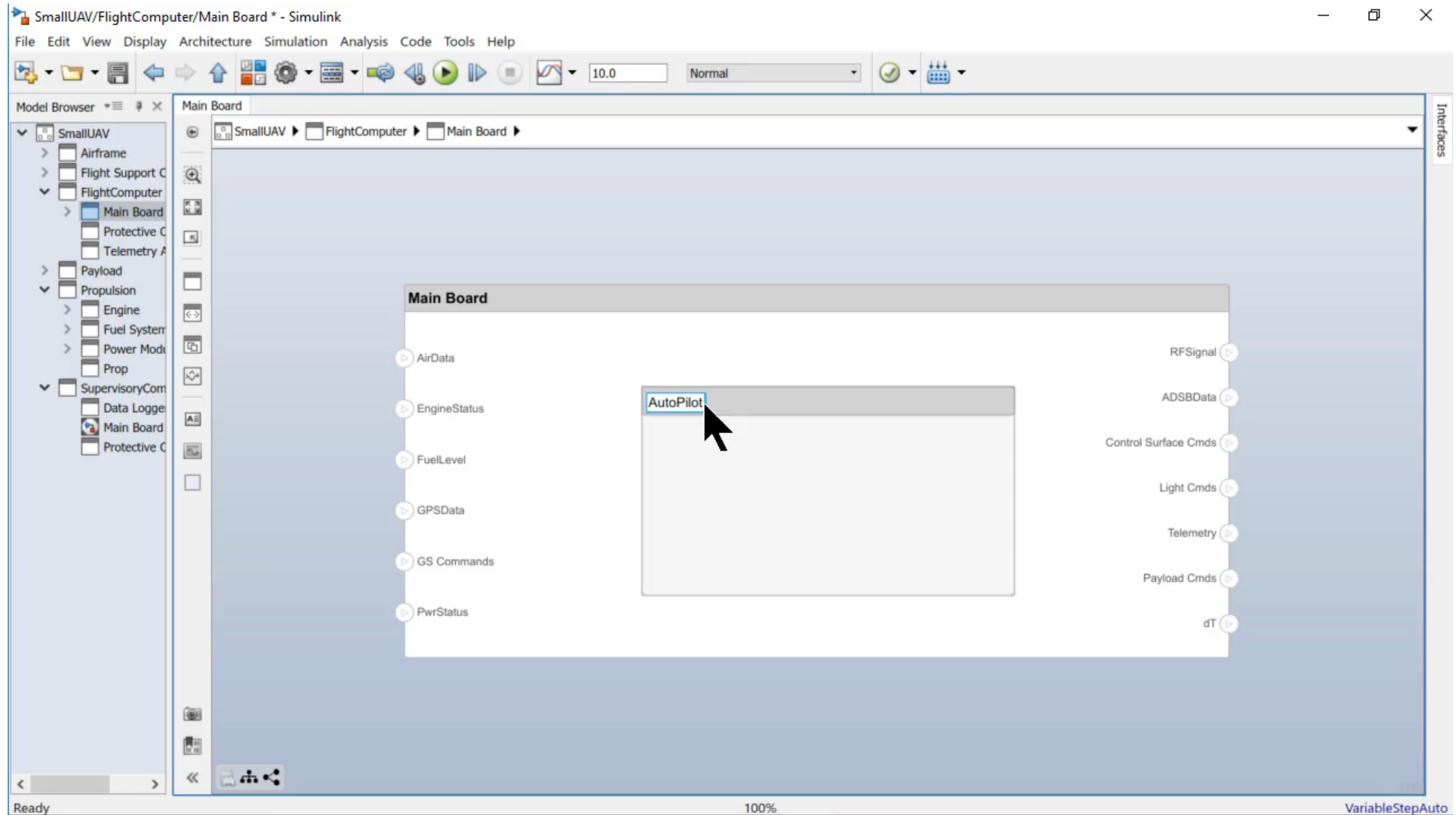
**Interfaces**

Source: SmallUAV.slx

- architecture\_gsCommands
- apConfigParams
- gsCommands
- uavMission
- ctrlSfcdDeflection
- gs\_commands\_bus
- flightCmds
- mission\_item\_bus

At the bottom of the window, the status bar shows 'Ready', '50%', and 'VariableStepAuto'. The Windows taskbar is visible at the very bottom with the time '2:26 PM' and date '2/21/2019'.

# Создание и привязка компонентов Simulink



# Назначение требований и трассируемость

The screenshot displays the Simulink environment for a project named 'SmallUAV'. The main workspace shows a hierarchical diagram of requirements, with a 'Flight Request Components' block at the top. A 'Property Inspector' window is open on the right, showing details for 'Requirement: #18'. The 'Properties' tab is active, displaying the following information:

- Type: Functional
- Index: 1.3.3
- Custom ID: #18
- Summary: Default Payload

The 'Description' tab is also visible, showing the text: 'Aircraft shall provide a pre-installed payload in the form a a gimbal-stabilized visual camera providing Full-Motion video.'

At the bottom of the Simulink window, the 'Requirements - SmallUAV' panel is visible, showing a tree view of requirements. The 'smallUAVReqs' folder is expanded, showing a hierarchy of requirements: 1, 1.1, 1.1.5, 1.1.4, 1.1.12, and 1.1.11. The 'Summary' column for these requirements lists: Aircraft Capabilities, Airworthiness, Control Surface Fault-Tolerance, Emergency Power, Endurance, and Flight Computer.

# Назначение требований и анализ

SmallUAV \* - Simulink

File Edit View Display Architecture Simulation Analysis Code Tools Help

SmallUAV

SmallUAV

Flight Support Components

Flight Computer

Payload

Airframe

Propulsion

Requirements - SmallUAV

View: Requirements

| Index | Summary              | Implemented |
|-------|----------------------|-------------|
| > 1.1 | Airworthiness        |             |
| > 1.2 | Communications       |             |
| ▼ 1.3 | Payload Capabilities |             |
| 1.3.1 | Carrying Capacity    |             |
| 1.3.2 | Payload Bay Capacity |             |
| 1.3.3 | Default Payload      |             |
| 1.3.4 | Payload Protection   |             |

Property Inspector

Requirement: #17

Details

▼ Properties

Type: Functional

Index: 1.3.1

Custom ID: #17

Summary: Carrying Capacity

Description Rationale

Aircraft shall be able to carry up to 4.0 Kg of payload

Keywords:

Links

Implemented by:

- Airframe
- Payload

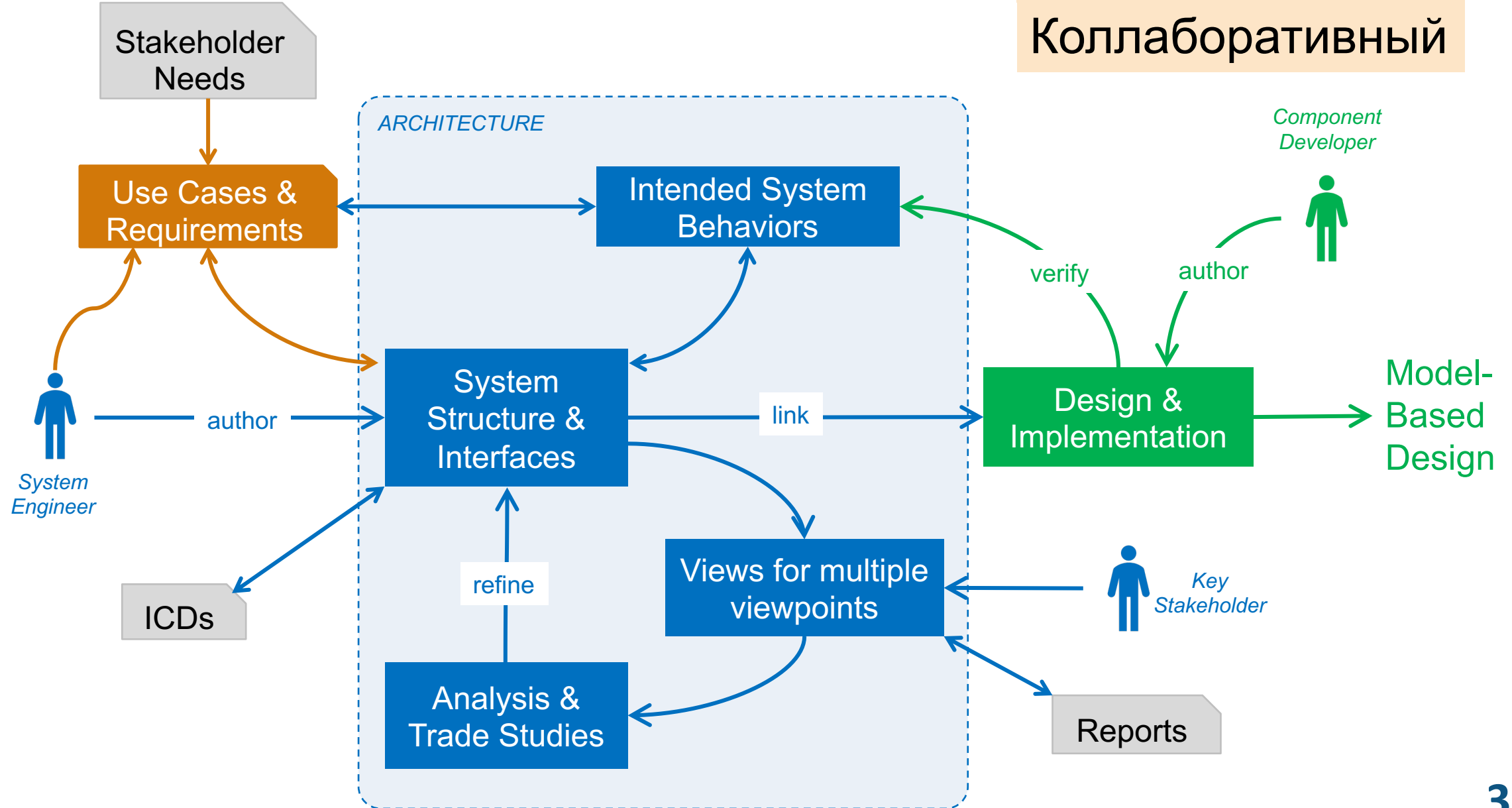
Ready 50% VariableStepAuto

# Дополнительные возможности

- Стереотипы
- Анализ компромиссов с использованием MATLAB
- Повторное использование моделей
- Создание архитектурных проектов
- Архитектурные представления

# Как выглядит типичный процесс?

Итеративный  
Коллаборативный



**tech@exponenta.ru**  
**exponenta.ru**



**ЭКСПОНЕНТА**  
ЦЕНТР ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ  
И МОДЕЛИРОВАНИЯ

- **Технические консультации**
- **Подбор инструментов**
- **Обучение специалистов**
- **Работа на заказ**