Системная инженерия на основе формальных моделей

Дмитрий Рыжов

Начальник отдела
системного моделирования

АО Навигатор
Инженерный центр
г. Зеленоград

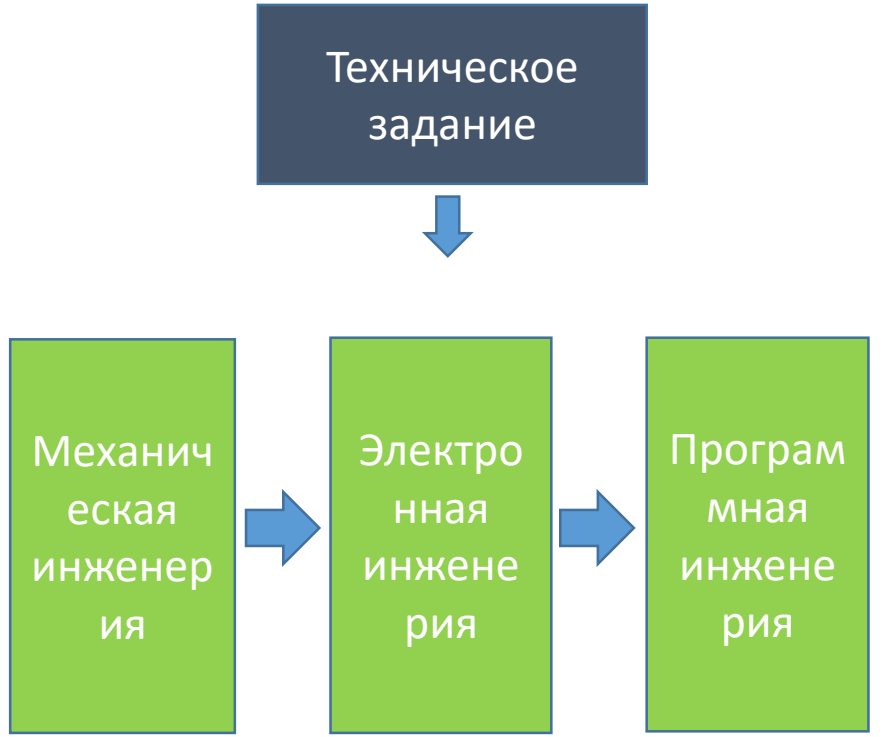
Решаемые задачи

- Разработка программно-аппаратных изделий для авиации
- Сертификация разрабатываемых изделий, аппаратуры, ПО на соответствие требования безопасности
- Постановка процессов системной инженерии, разработки ПО и АО
- Разработка и внедрение ИТ-решений для поддержки процессов системной инженерии

Подход к разработке “на основе ТЗ”

- Разработка компонентов изделия на основе технического задания

- Сначала аппаратура, затем ПО

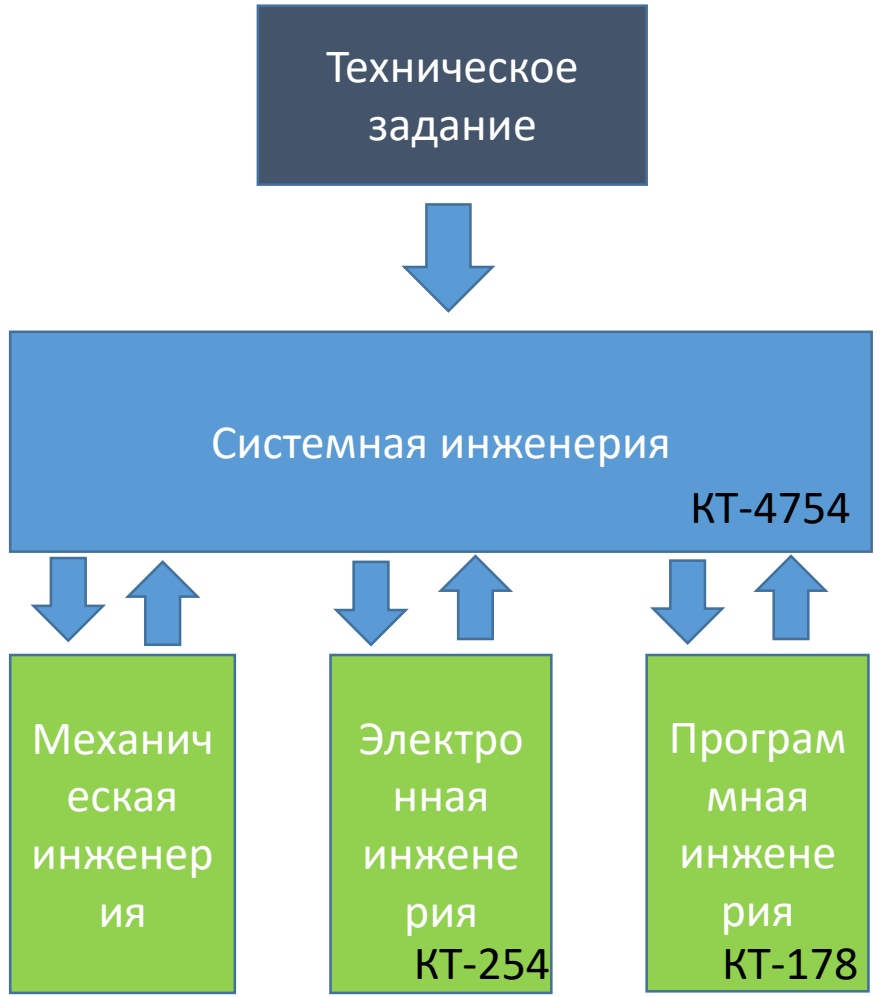


Сертификация

- Невозможность обеспечить безопасность разработанной системы путем испытаний
- Сертификация по безопасности не продукта, а организации, ее процессов разработки
  - Подтверждение, что разработка велась правильно на всем протяжении проекта
- Стандарты безопасности определяют требования для соответствия организации при сертификации
  - 4754, 4761, 254, 178, Р-331
- Чрезвычайно трудоемка
- Требует изменения подходов к работе
  - Постановки процессов
  - Разработки методик
  - Автоматизации процессов и методик

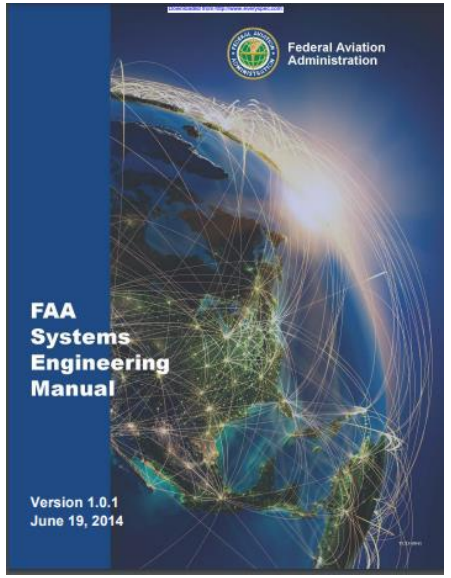
Системная инженерия

- Методология создания успешных систем
  - Соответствие ожиданиям всех заинтересованных сторон
  - Выбор оптимального решения на основе анализа альтернатив
  - Детальные требования на разработку компонентов
  - Успешная интеграция из разработанных компонентов
  - Уменьшение количества переделок на поздних стадиях
- Процессы разработки конструкций, АО, ПО
  - Интегрированные с процессами системной инженерии



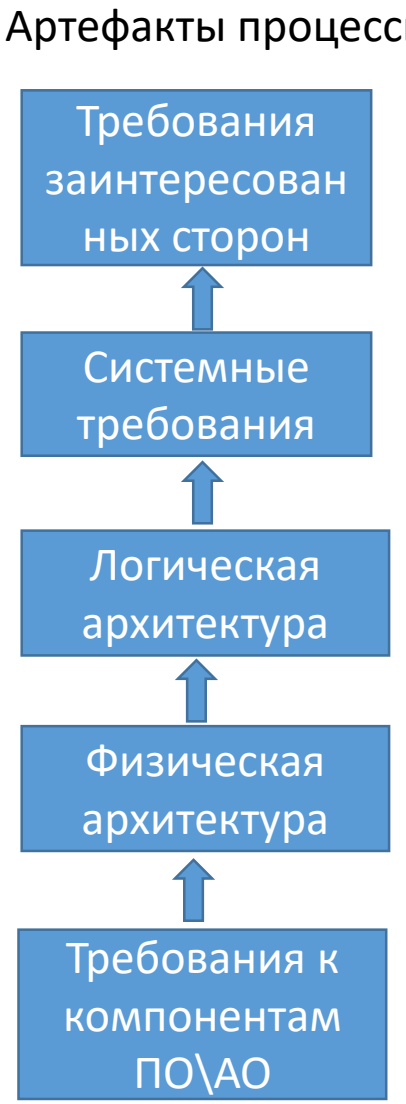
Роль стандарта КТ-4754

- КТ 4754
  - Содержит требования к процессам системной инженерии в разрезе безопасности
  - Не является полной спецификации процессов системной инженерии и методик их выполнения
- Материалы по процессам системной инженерии
  - IEEE Std 1220-2005, ANSI/EIA-632, ГОСТ Р ИСО/МЭК 15288
  - FAA System Engineering Manual
  - FAA Requirements Engineering Management Handbook
    - По сути является описанием моделе-ориентированного подхода



Процесс системного анализа и проектирования

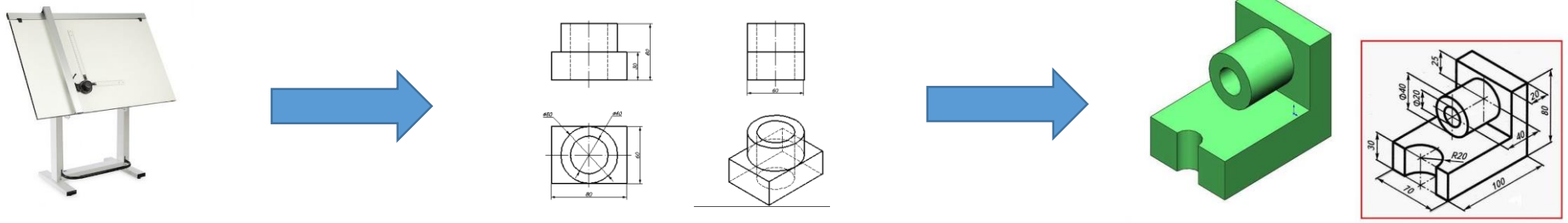
- Собрать требования заинтересованных сторон
- Определить системные требования
- Определить логическую архитектуру системы
- Определить физическую архитектуру системы
- Распределить системные требования по компонентам системы\изделия
  - Выдать требования к компонентам (подсистеме, ПО, АО)



Направление стрелок отображает соответствие предыдущим уровням

Системная инженерия: от “Кульмана” к “3D моделям”

- Кульман
  - Текстовые документы с требованиями в MS Word, схемы в MS Visio
- “2D чертежи”
  - Системы управления требованиями (DOORS, Polarion, …)
- “3D модели”
  - Формальные (информационные) модели системы



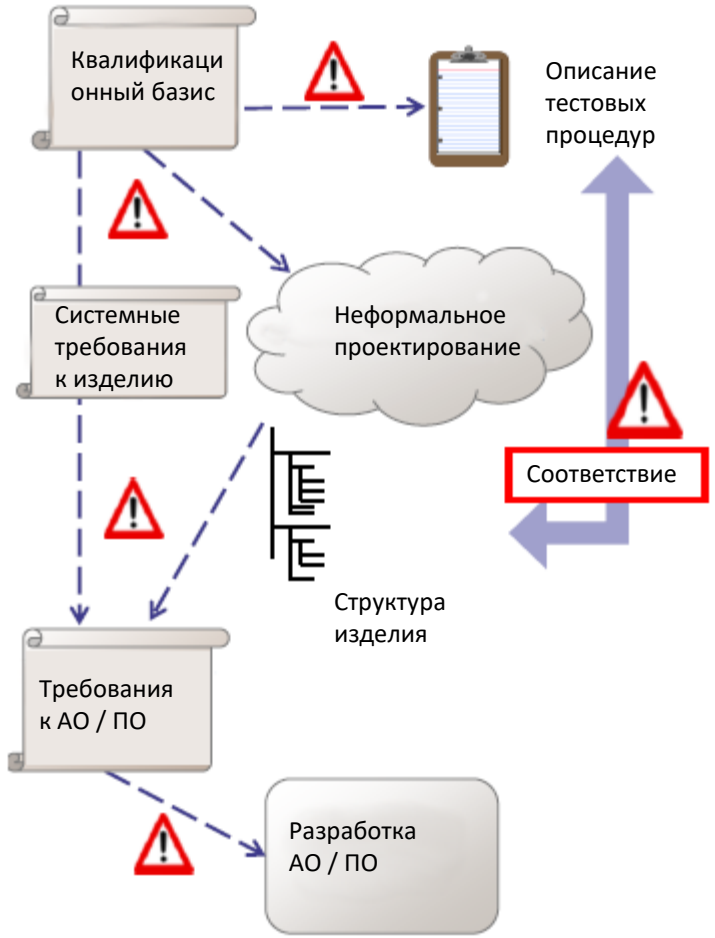
Ручные чертежи

Не связанные электронные чертежи

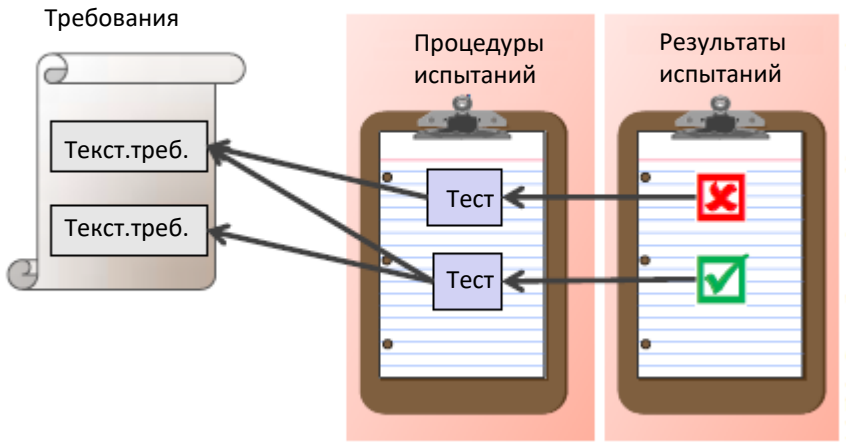
Создание чертежей на основе 3D моделей

Подход на основе текстовых требований

Спецификация системных требований и требований к АО \ ПО

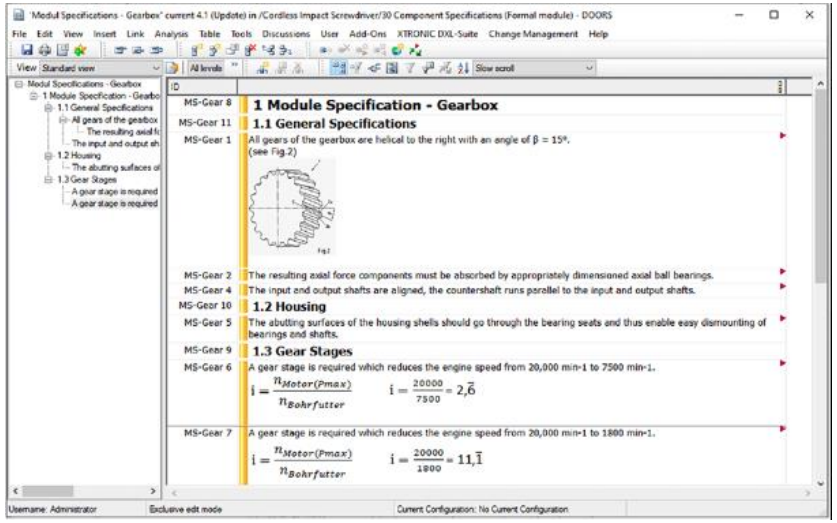


Валидация и верификация текстовых требований



Управление требованиями (2D)

- Основные артефакты
  - Документы\Модули с требованиями
- Текстовые формулировки требований
- Атомарность требований
- Классификация требований с помощью атрибутов
- Трассировка требований
- Настраиваемые табличные представления требований
- Включение в документы графических схем, выполненных в других инструментах



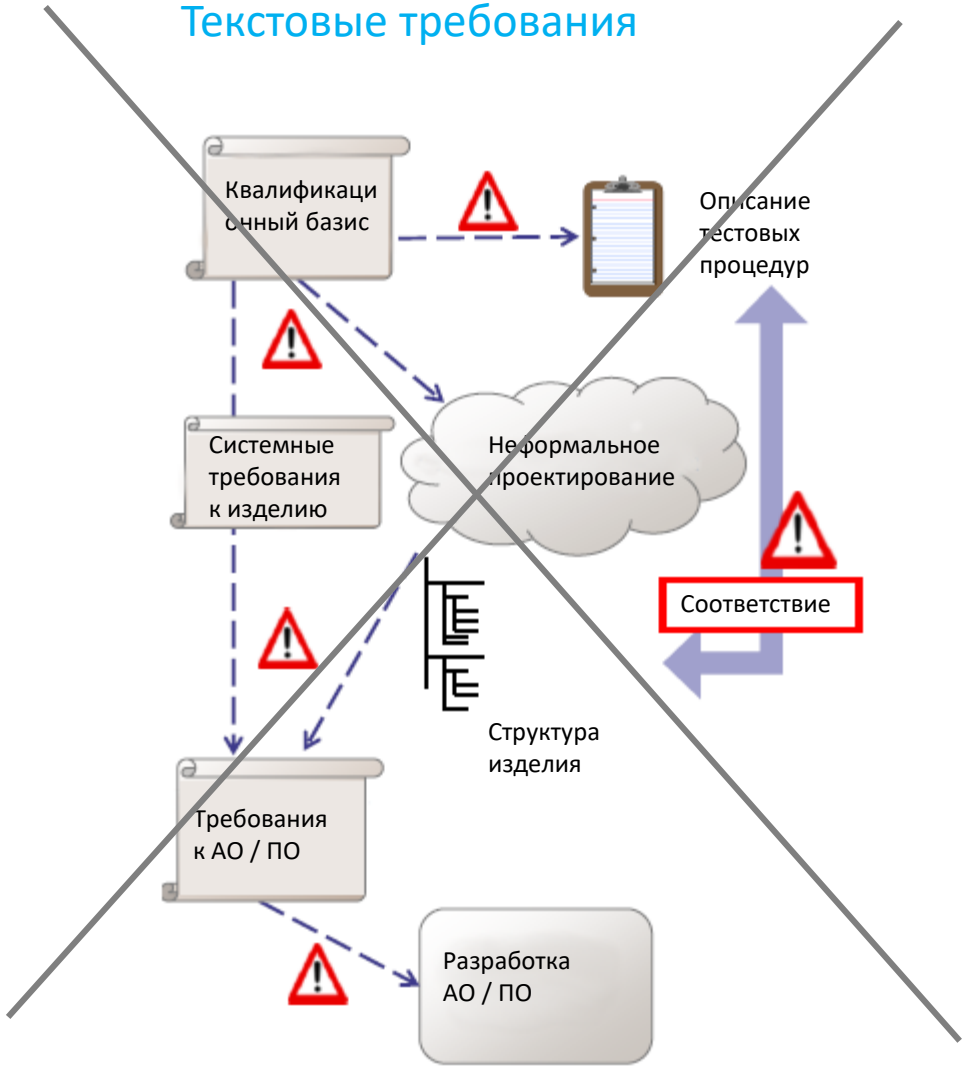
DOORS

Пример текстового требования:

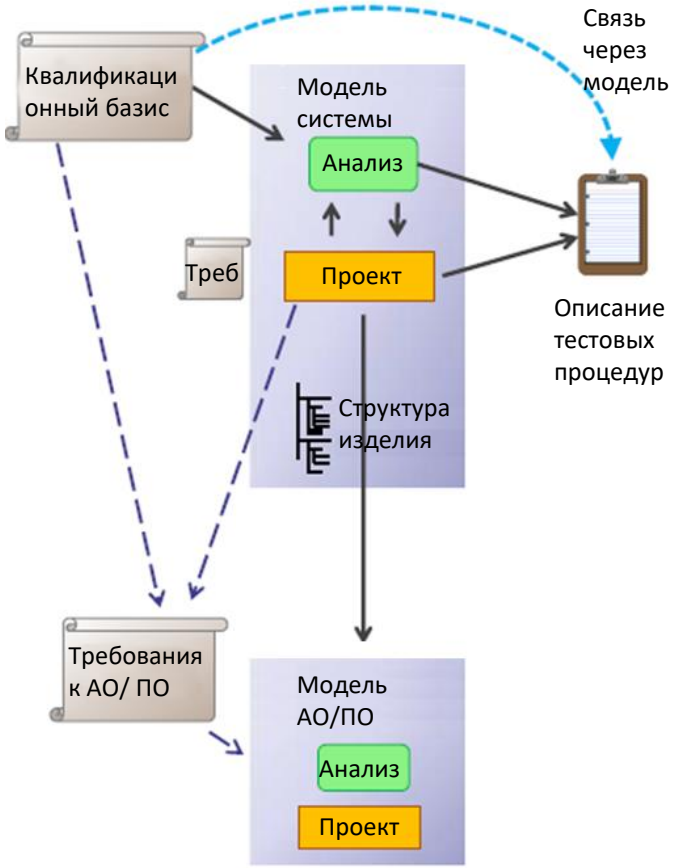
Система должна принимать команду РУЧНОЕ.

Отличие подхода на основе моделей

Текстовые требования



Текстовые требования + модели



Формальные (информационные) модели (3D)

- Способ описания системы с целью анализа, проектирования, анализа безопасности, контролепригодности, спецификации, валидации, верификации
- Мета-модель – определяет допустимые типы элементов и связей в формальной модели
  - Атомарность “на уровне сущностей мета-модели”
  - Основа языка моделирования
- Не правильная интерпретация формальных (информационных) моделей:
  - - это не диаграммы
  - - не являются исполняемыми
  - - не могут выступать в виде требований

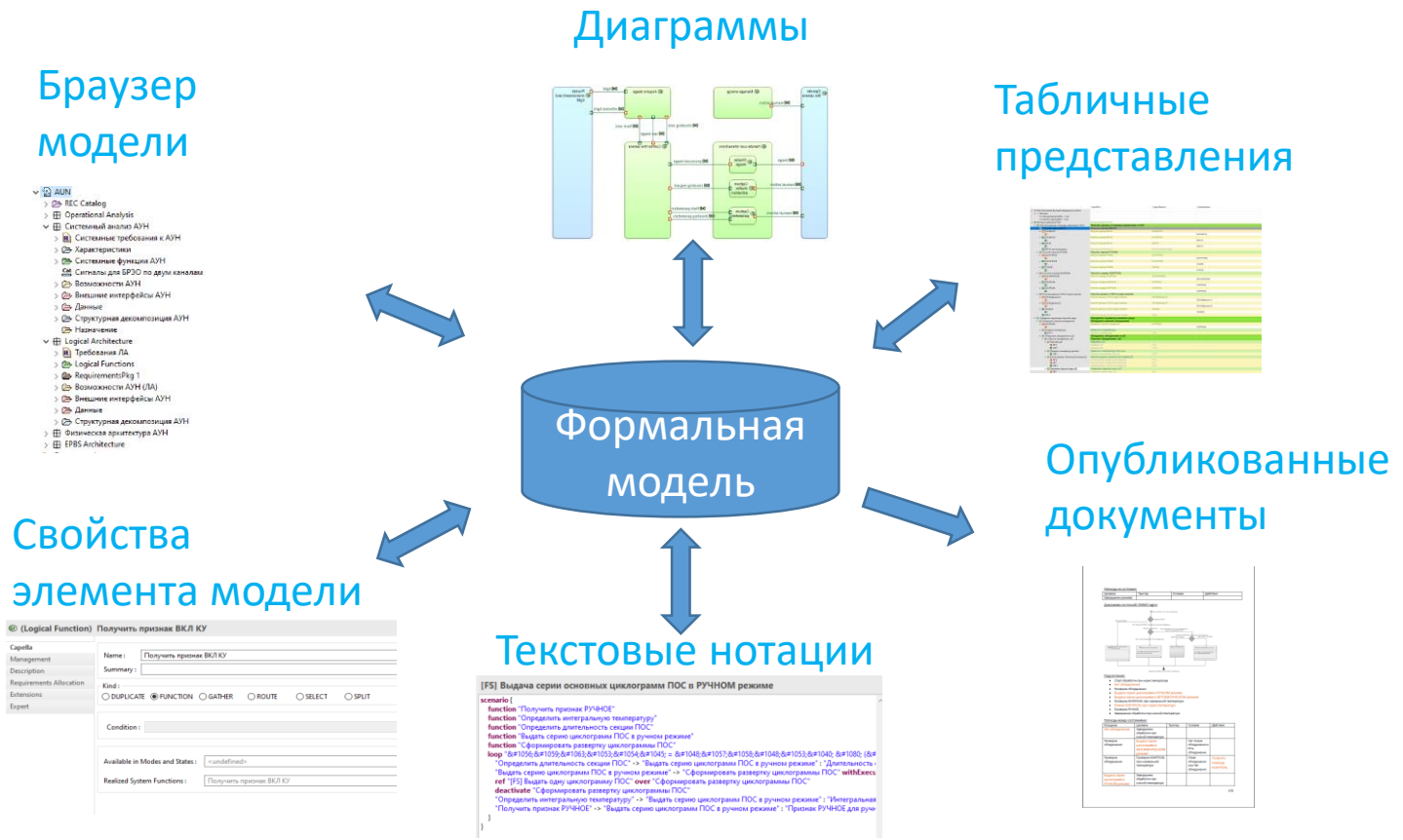
- Требование -> аллоцировано на -> Функция
- Функция -> взаимодействует с -> Функция
- Функция -> аллоцирована на -> Компонент
- Компонент -> размещен на -> Узел

Пример формального требования:

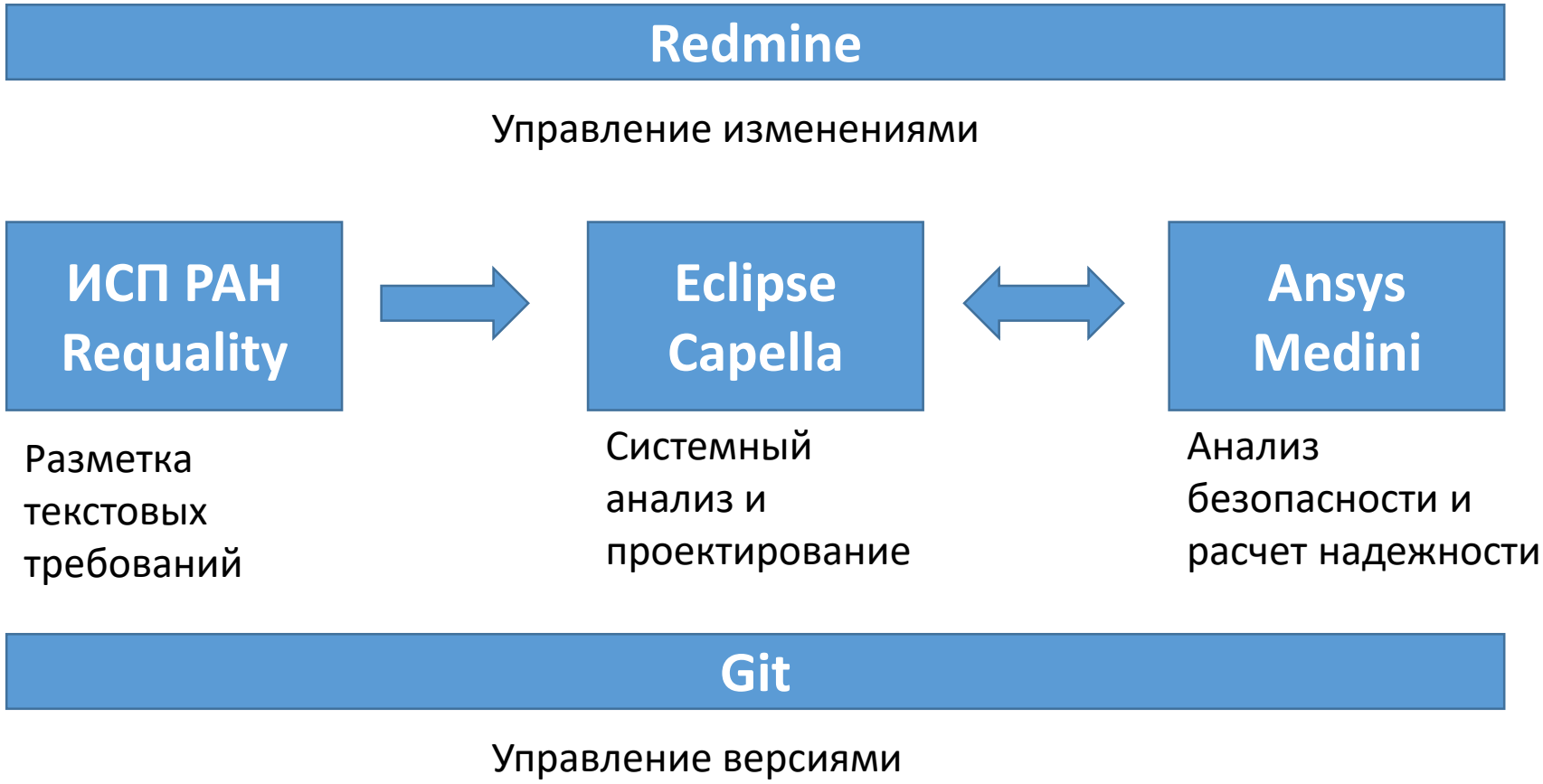
Функция Получить команду РУЧНОЕ аллоцирована на Система АУН.

Представления формальной модели

- Предназначены для анализа и редактирования формальных моделей
- Различные представления для решения различных задач
- Синхронизованы между собой и формальной моделью



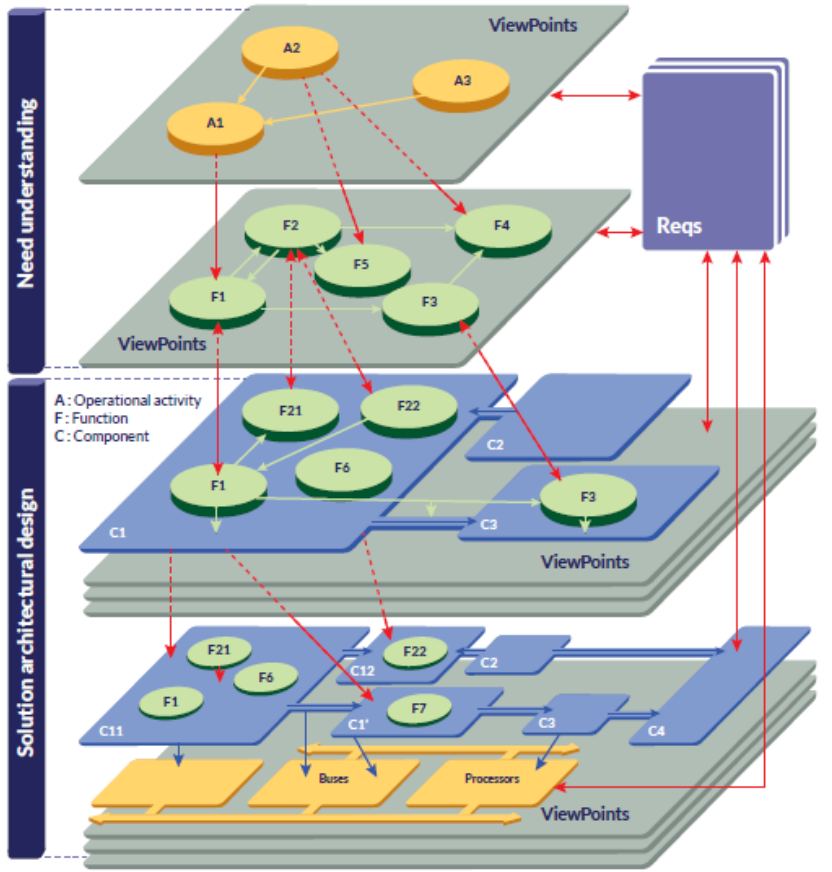
Решение для системной инженерии на основе формальных моделей



https://www.requality.ru/ru/ https://www.eclipse.org/capella/

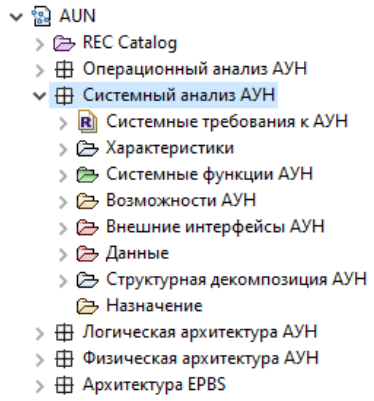
Eclipse (Thales) Capella

- Capella - инструмент для инженерии требований, системного анализа и проектирования на основе формальных моделей
- Arcadia – процесс системного анализа и проектирования на основе формальных моделей в Capella
- Внутренняя разработка компании Thales
- Open Source с 2015 года
- Следствие безуспешных попыток внедрения SysML и IBM Rational Rhapsody в течении 8 лет в Thales
- Более 1000 системных инженеров в Thales обучены использованию Capella
- Активно развивается в концепции Industrial Source в рамках Eclipse
- Seimens System Engineering Workbench – Capella, интегрированная в TeamCenter PLM
- Многоплатформенность: Windows \ Linux \ Mac



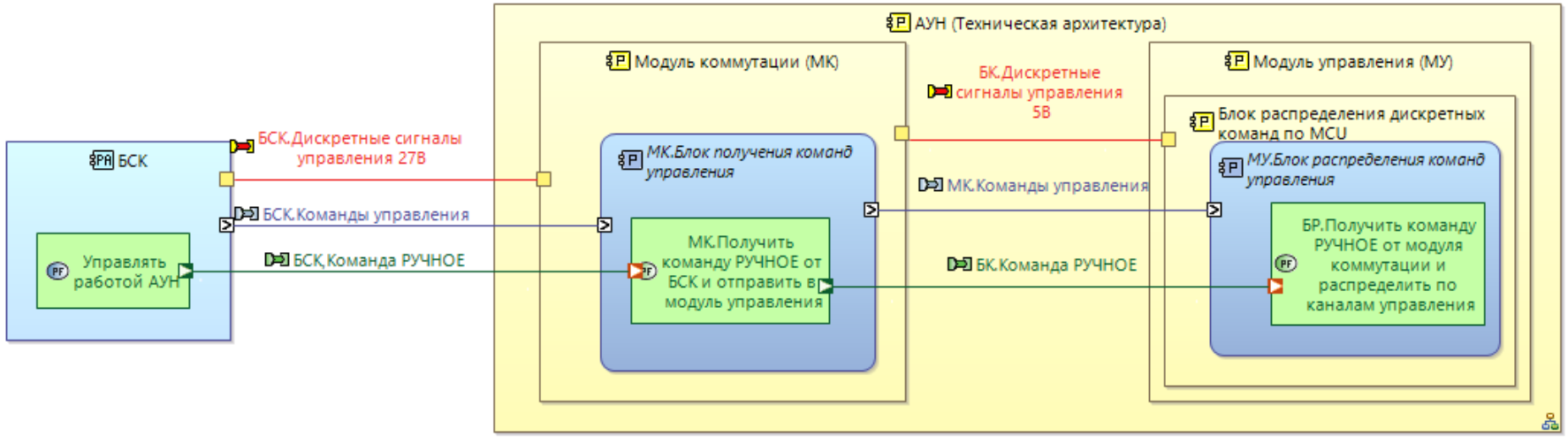
Язык моделирования Capella

Интегрированная модель для всех уровней анализа и проектирования систем, ПО, АО



Элементы модели на каждом уровне:

- Требования
- Возможности
- Сценарии и функциональные цепочки
- Функции и функциональны взаимодействия
- Структура: компоненты и узлы
- Режимы, состояния, переходы
- Данные
- Интерфейсы

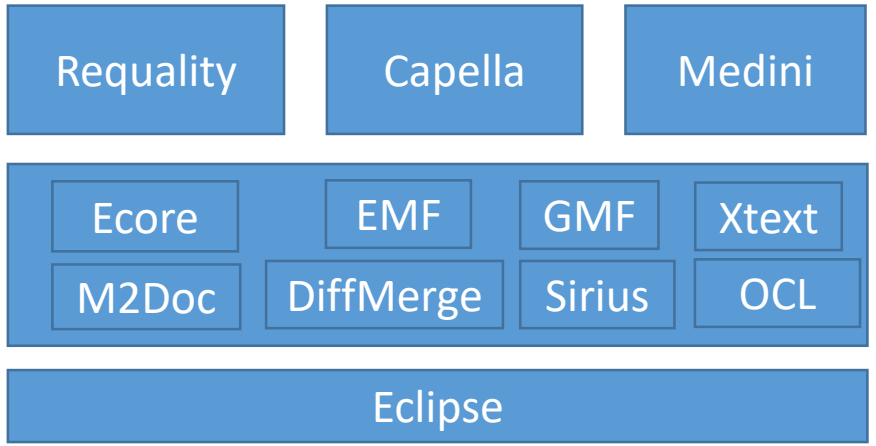


Интеграция и кастомизация инструментов моделирования

- Eclipse – платформа для создания интегрированных средств разработки

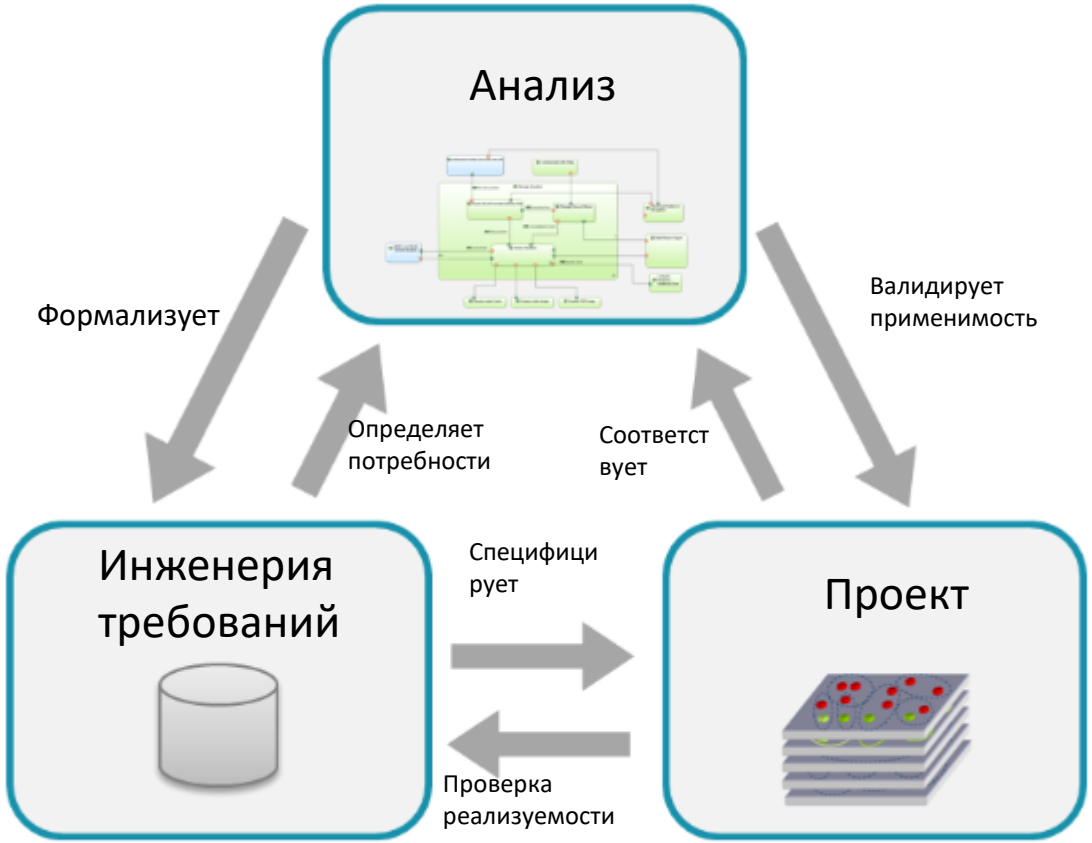
- Eclipse Modeling Framework
  - Основа большого количества современных инструментов моделирования (коммерческих и open source)

- Единый технологический стек для расширения инструментов моделирования
  - Ecore - Определение мета-моделей
  - EMF – Генерация Java API и иерархических редакторов
  - Sirius - новые графические и табличные представления
  - Xtext - новые текстовые нотации
  - M2Doc - генерация документов
  - DiffMerge - сравнение и объединения моделей
  - OCL - запросы к данным моделей
  - …



Инженерия требований, интегрированная с проектированием

- Разметка текстовых требований квалификационного базиса(КБ) в Requality
- Импорт\обновление требований КБ из Requality в Capella
- Аллокация требований КБ на элементы модели в Capella
- Представления для анализа покрытия требований

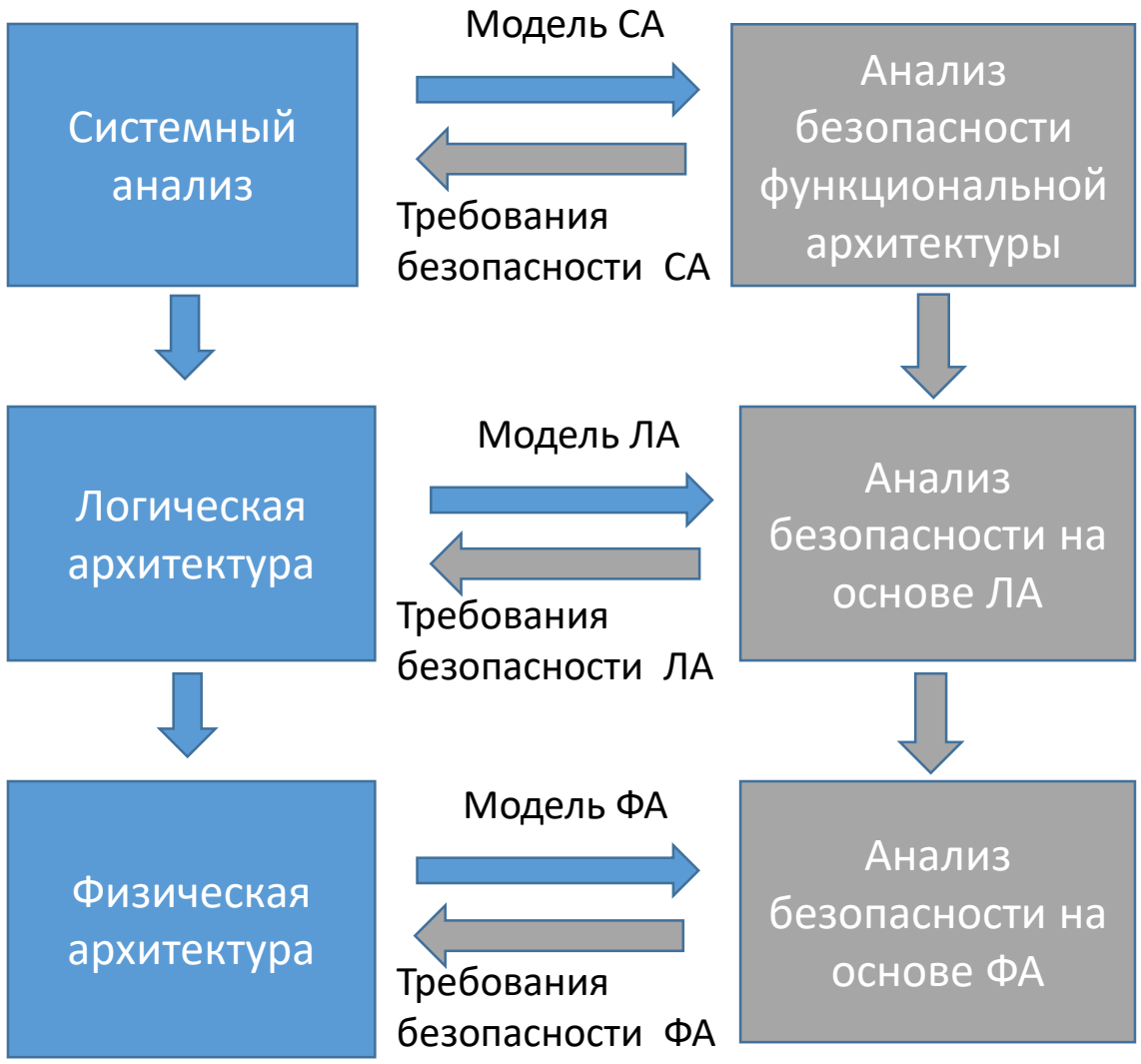


Проектирование с учетом специальных требований

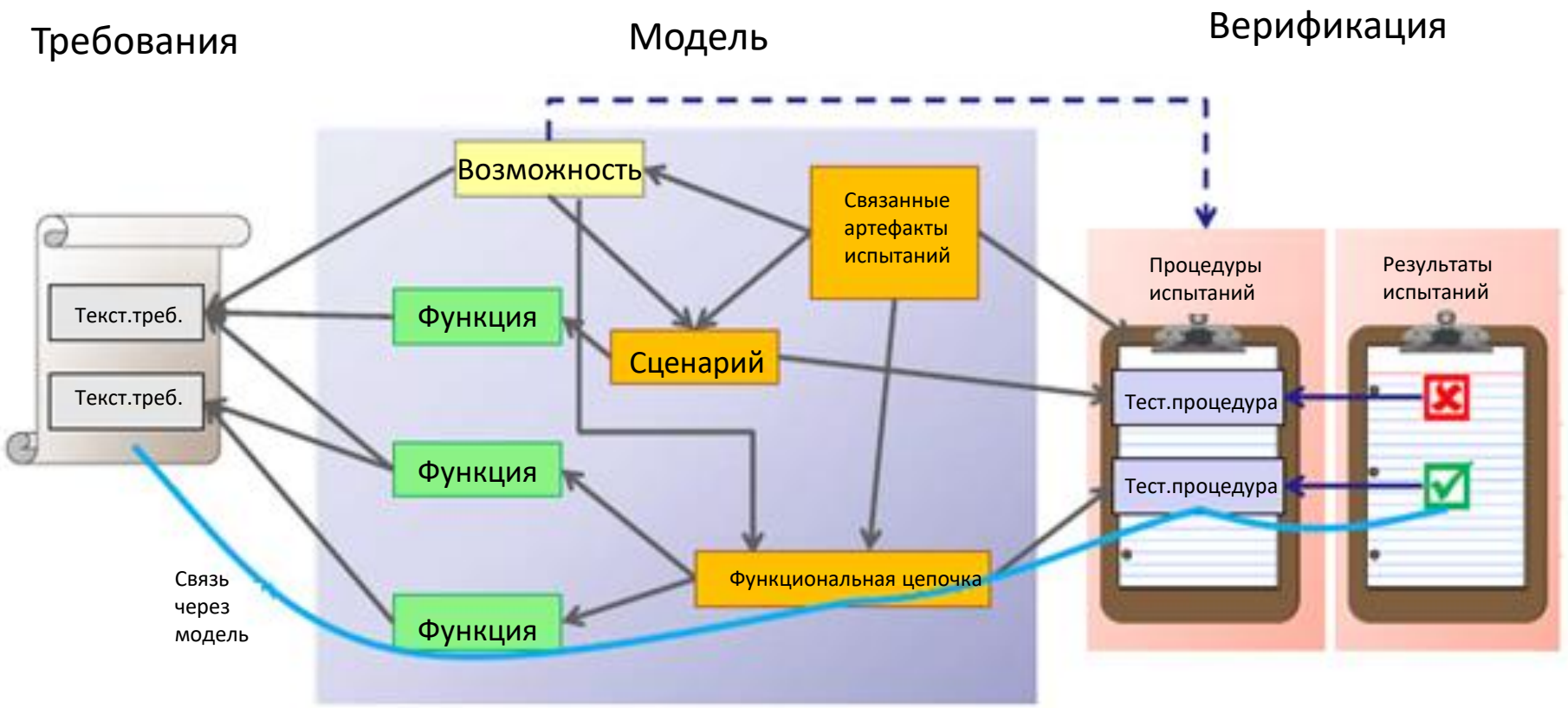
Анализ альтернативных решений
в соответствии с ISO 42010

Предварительный анализ безопасности на основе формальных моделей

- Определение требований по безопасности на ранних этапах проекта
- Импорт\обновление моделей из Capella в Medini
- Адаптировали AADL error model для использования в Медини
  - Локальные модели распространения ошибок
  - Объединение локальных моделей ошибок на основе архитектуры системы
  - Генерация глобальных FTA, FMEA
- Переиспользование результатов анализа безопасности с предыдущих уровней



Верификация на основе формальных моделей



Анализ и проектирование ПО и АО на основе формальных моделей системы

- Генерация формальных моделей ПО и АО в Capella на основе модели системы
  - Требования верхнего уровня к ПО/АО
  - Логическая архитектура ПО/АО
- Автоматическая трассировка от моделей ПО и АО к модели системы
- Обновление формальных моделей ПО и АО при внесении изменений в модель системы
- Определение требований низкого уровня к ПО на основе моделей

От информационных моделей к имитационным моделям и коду

- Формальные (информационные) модели являются спецификацией требований различного уровня абстракции
- Имитационные модели на различных уровнях проектирования
- TASTE (https://taste.tools/)
  - Генерация моделей AADL, ASN.1 на основе моделей Capella
  - Детальное описание поведения функций с использованием различных нотаций\языков программирования
    - SDL, MSC, Simulink, VHDL, C, C++
  - Генерация кода для имитационных моделей\целевой системы

Спасибо за внимание!

Дмитрий Рыжов

+7 (909) 156-69-95

dryzhov@navigat.ru