

Программная инженерия

Анализ и проектирование
программного обеспечения – 4

Вопросы

- Описание архитектуры
 - Повторение: стандарты
 - Состав
 - Зависимости проектных решений
- Разработка архитектуры
 - Цели проектирования
 - «Плохие» и «хорошие» парадигмы проектирования
- Практический пример: РИС ППУ
 - Архитектура
 - Интерфейсы

Стандарты

Состав

Зависимость проектных решений

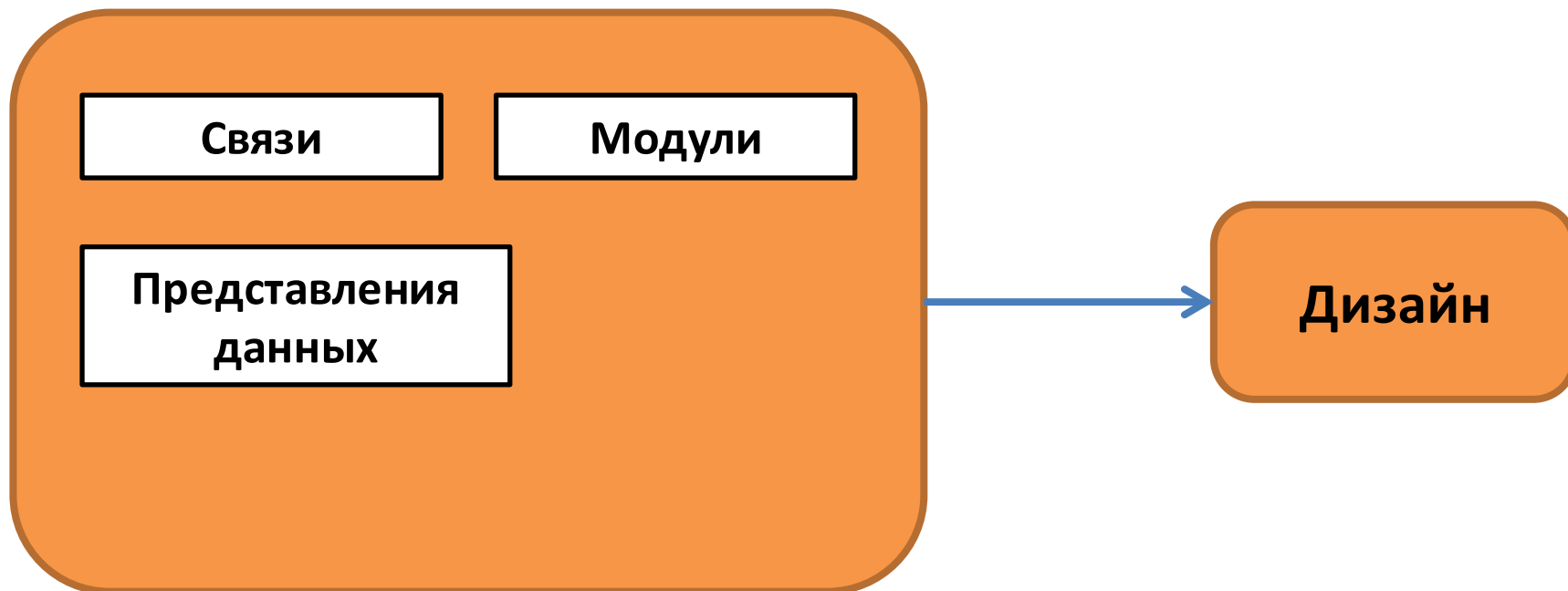
ОПИСАНИЕ АРХИТЕКТУРЫ

Стандарты

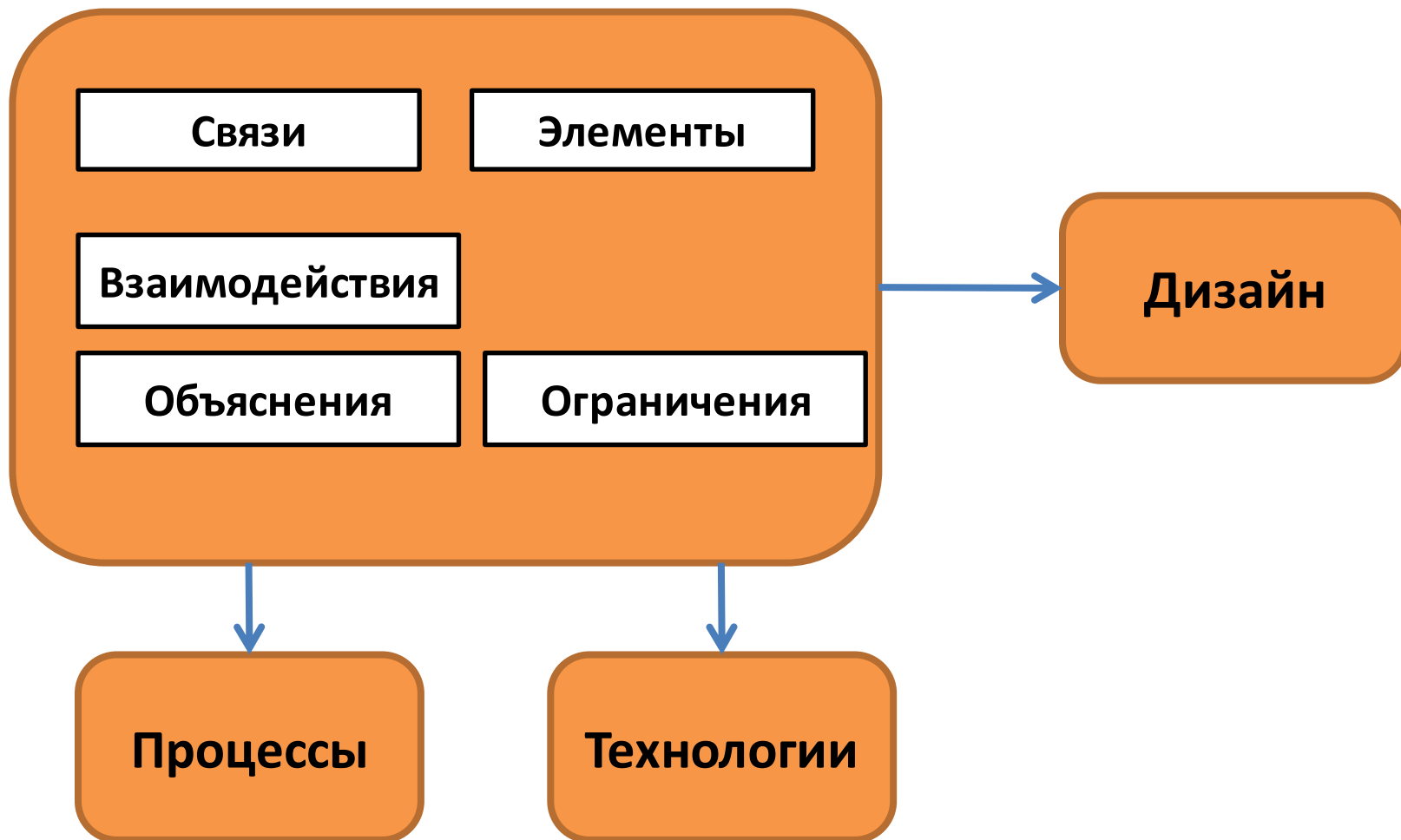
- ISO/IEC/IEEE 42010:2011 Systems and software engineering. Architecture description (Системная и программная инженерия. Описание архитектуры)
- ГОСТ 2.103-68 ЕСКД. Стадии разработки
- ГОСТ 2.119-73 ЕСКД. Эскизный проект
- ГОСТ 2.120-73 ЕСКД. Технический проект

Что входит в архитектуру – 1/4

До 1990-х



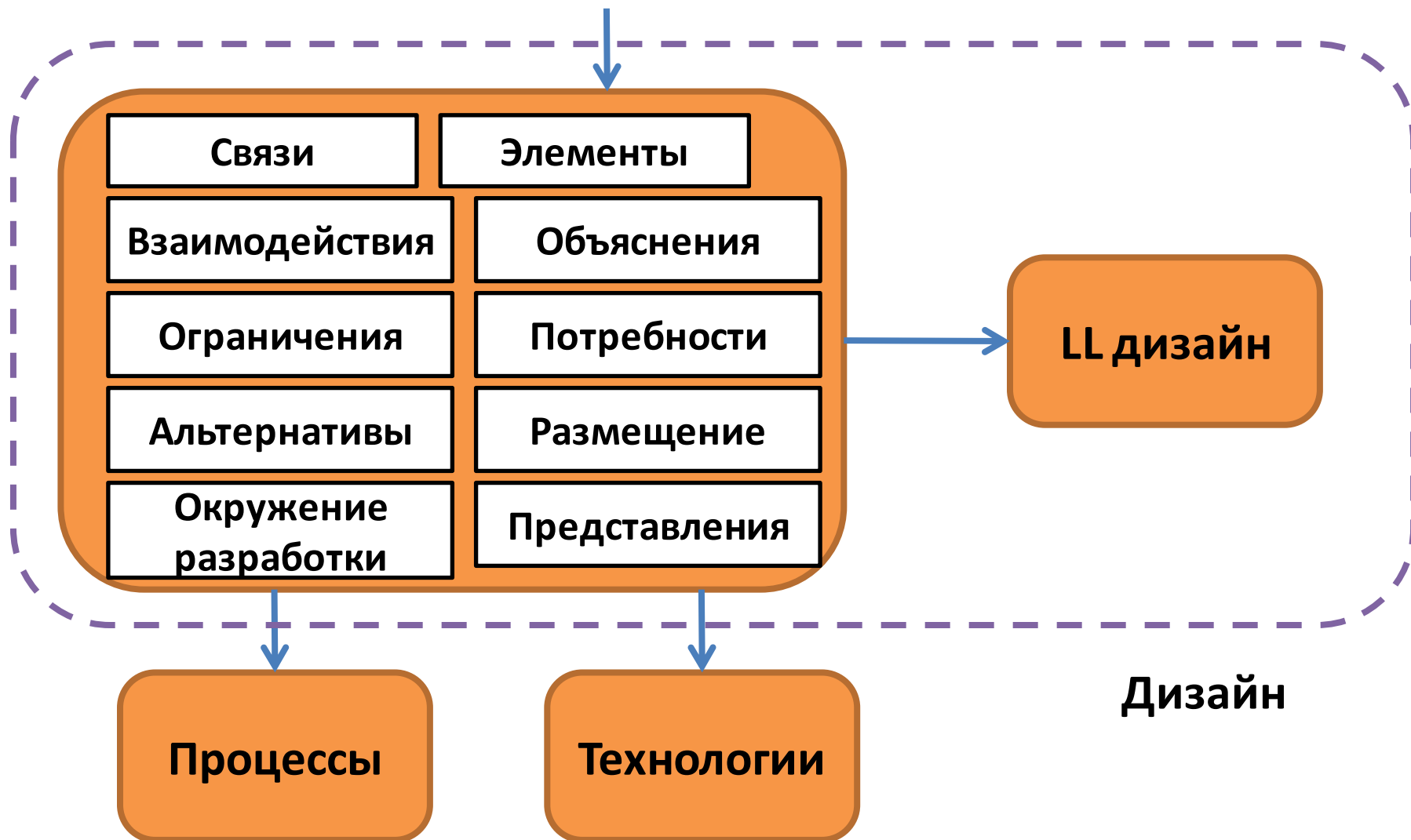
Что входит в архитектуру – 2/4



Что входит в архитектуру – 3/4

Потребности

Целевая система

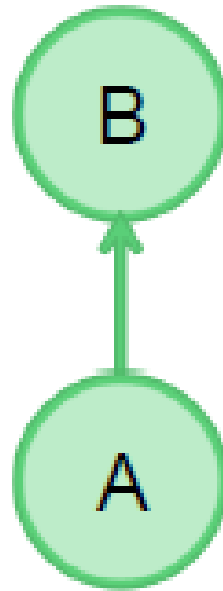


Что входит в архитектуру – 4/4

- Элементы, модули, связи, структуры, виды, формы, каркасы, разделения, ограничения, интересанты, потребности, окружение, размещение, управление, интерфейсы, слои, артефакты, технологии, статика и динамика, поведение, свойства, отношения, принципы, стили...
- → То есть самые разнообразные **проектные решения**

Зависимость проектных решений – 1/7

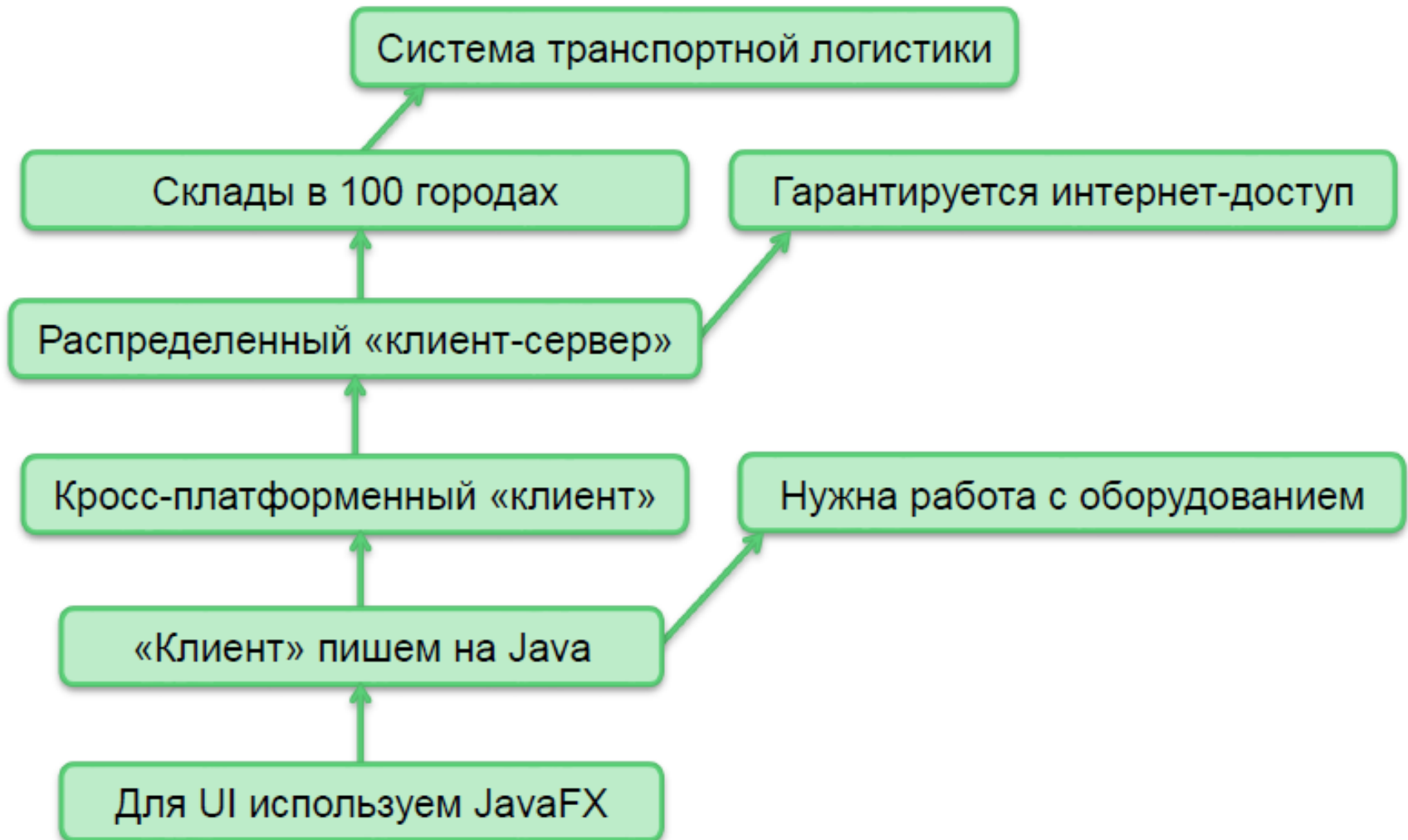
- Решение А зависит от решения В, если А имеет смысл или целесообразно только в том случае, когда принято и актуально решение В



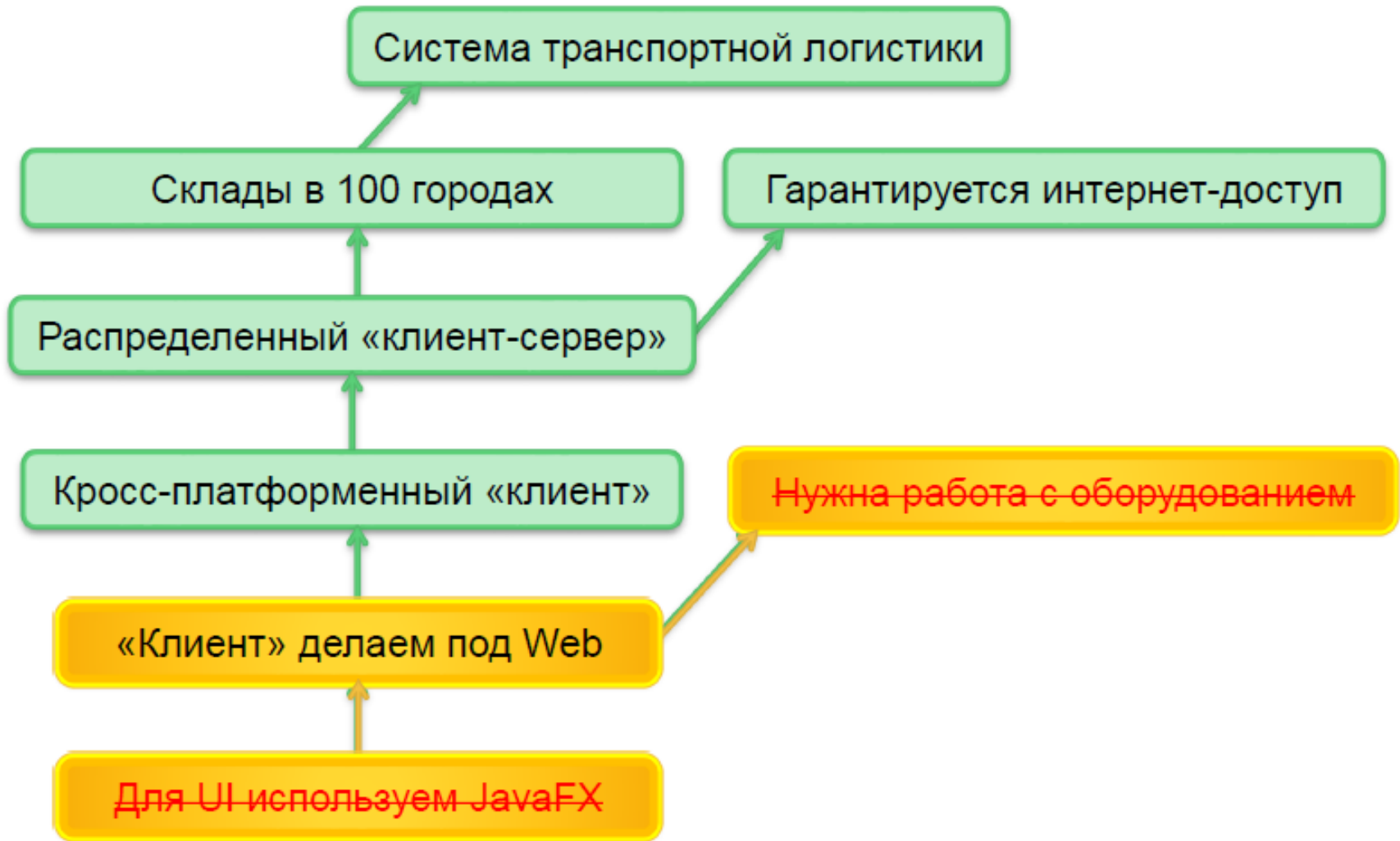
Зависимость проектных решений – 2/7

- Зависимости не бывают цикличны, решения образуют направленный граф («дерево решений»)
- Если изменяется некоторое решение, то придется пересмотреть все решения, которые прямо или косвенно зависят от него
- Фундаментальные решения (содержание архитектуры) – решения, от которых зависит слишком много и изменение которых обойдется слишком дорого

Зависимость проектных решений – 3/7



Зависимость проектных решений – 4/7

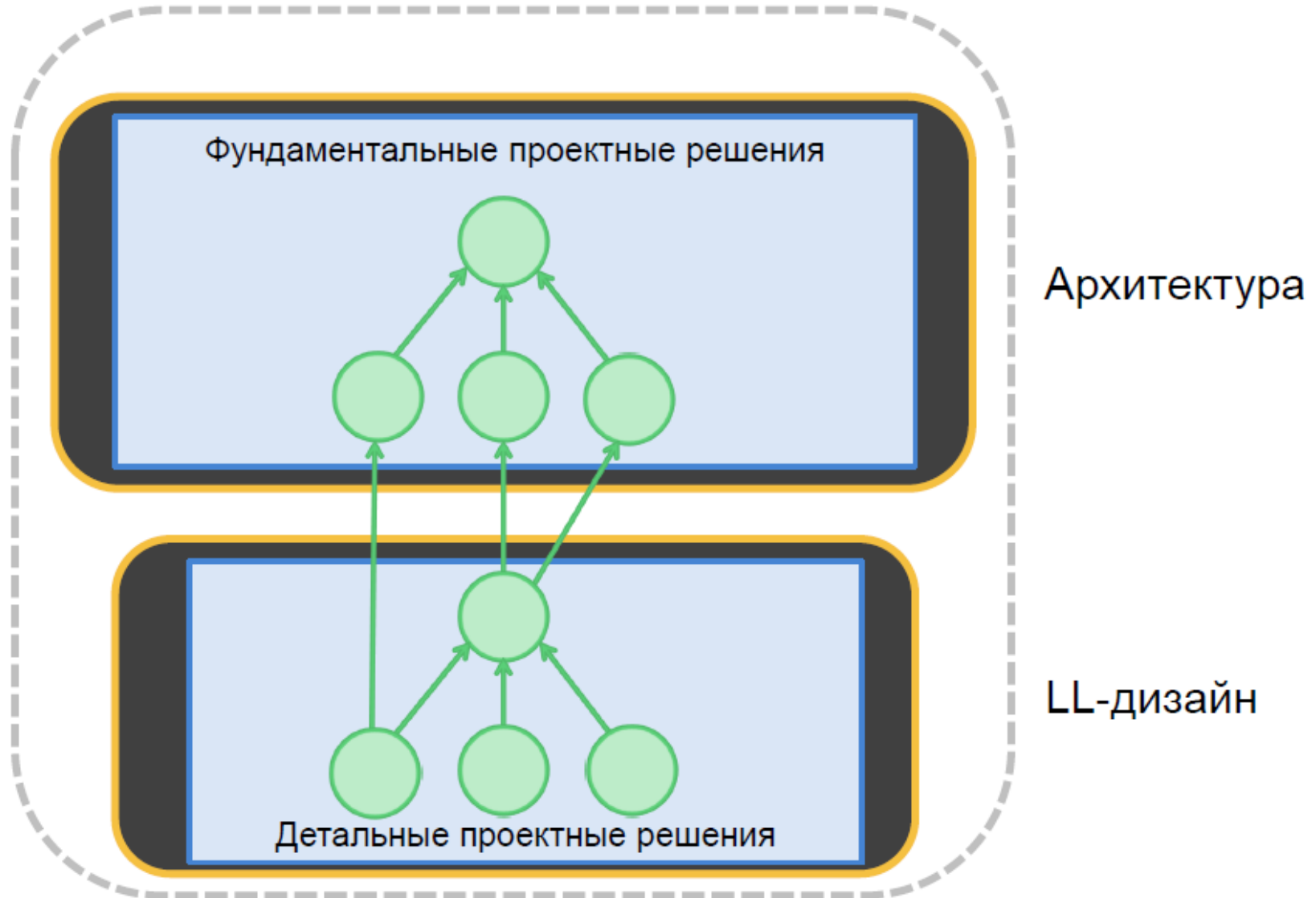


Зависимость проектных решений – 5/7



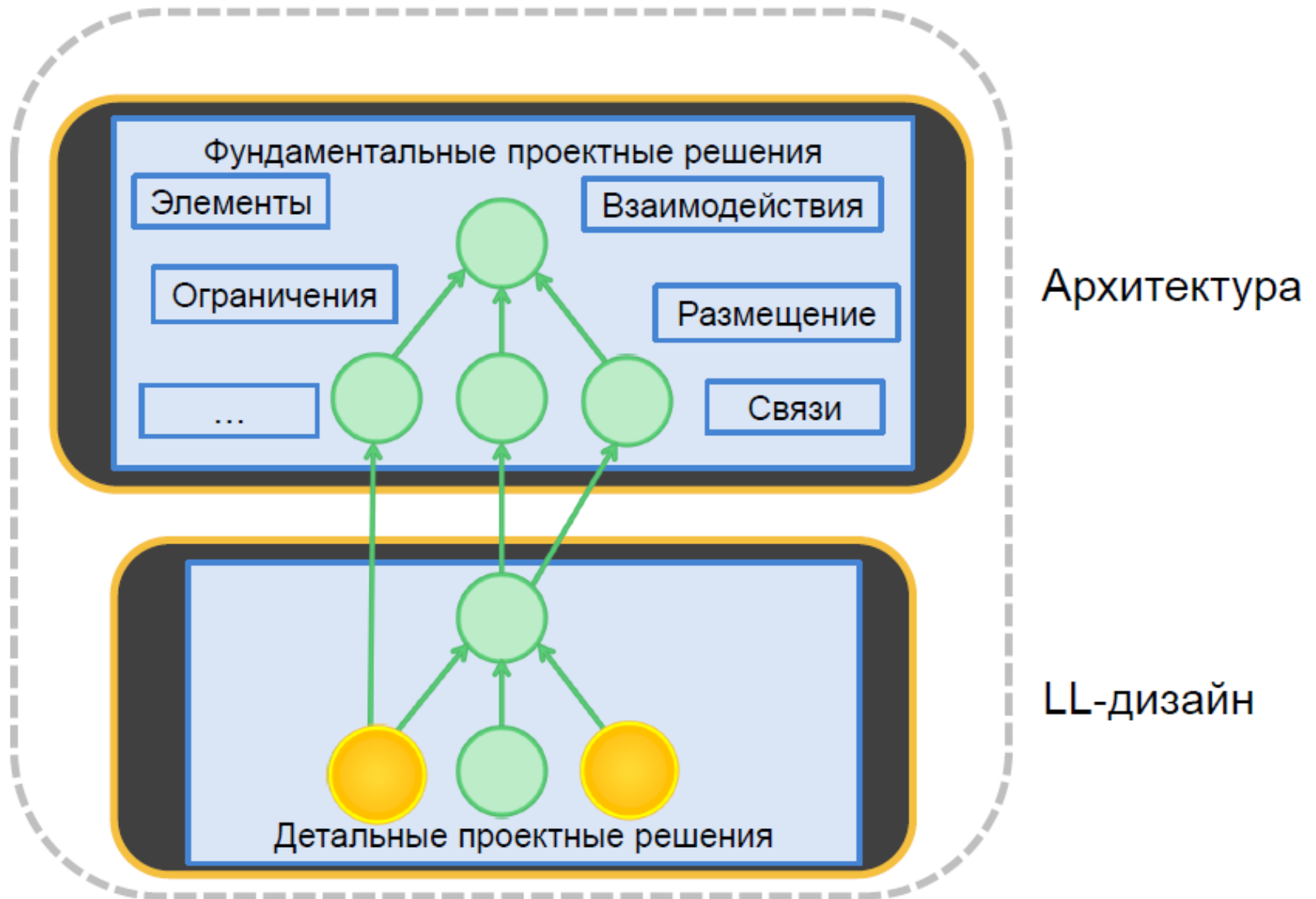
Зависимость проектных решений – 6/7

Дизайн



Зависимость проектных решений – 7/7

Дизайн



Цели проектирования
«Плохое» и «хорошее»
проектирование

РАЗРАБОТКА АРХИТЕКТУРЫ

Как ставить цель архитектору?

- Назначение системы
- Атрибуты качества
- Ограничения (включая \$ и t)
- Свойства окружения
 - Designtime
 - Runtime
- Не любые требования и ограничения!
 - Только фундаментальные (нужен анализ)

«Плохое» проектирование – 1/12а

- Лима
 - Архитектуры нет (нет целого)
 - Про целое ничего нельзя сказать, кроме объема

«Плохое» проектирование – 2/12а

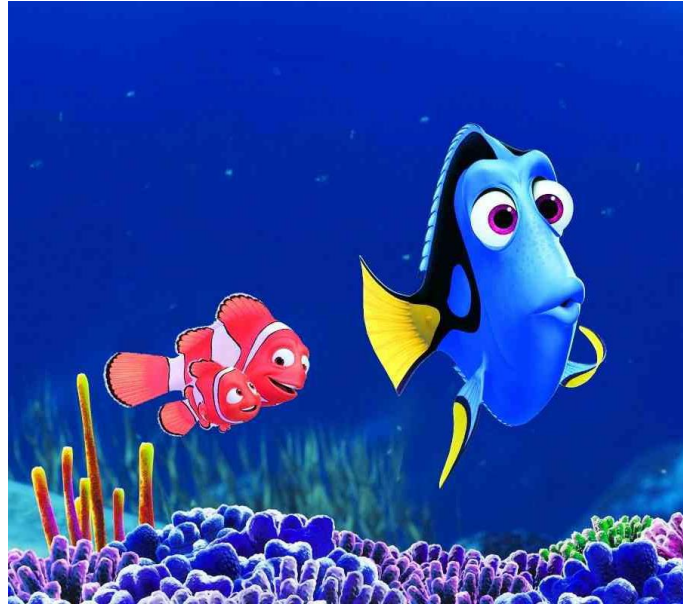
- Скрыта за кодом

- Архитектура есть, но «скрыта за кодом» → не проявлена
- Кто-то придумал, остальные постоянно «реверсируют» (с переменным успехом), ответственность потеряна
- Обсуждение предмета невозможно



«Плохое» проектирование – 3/12а

- Memento
 - Решения не фиксируются, забываются
 - Принимаются заново
 - Или искажаются из-за потери оснований

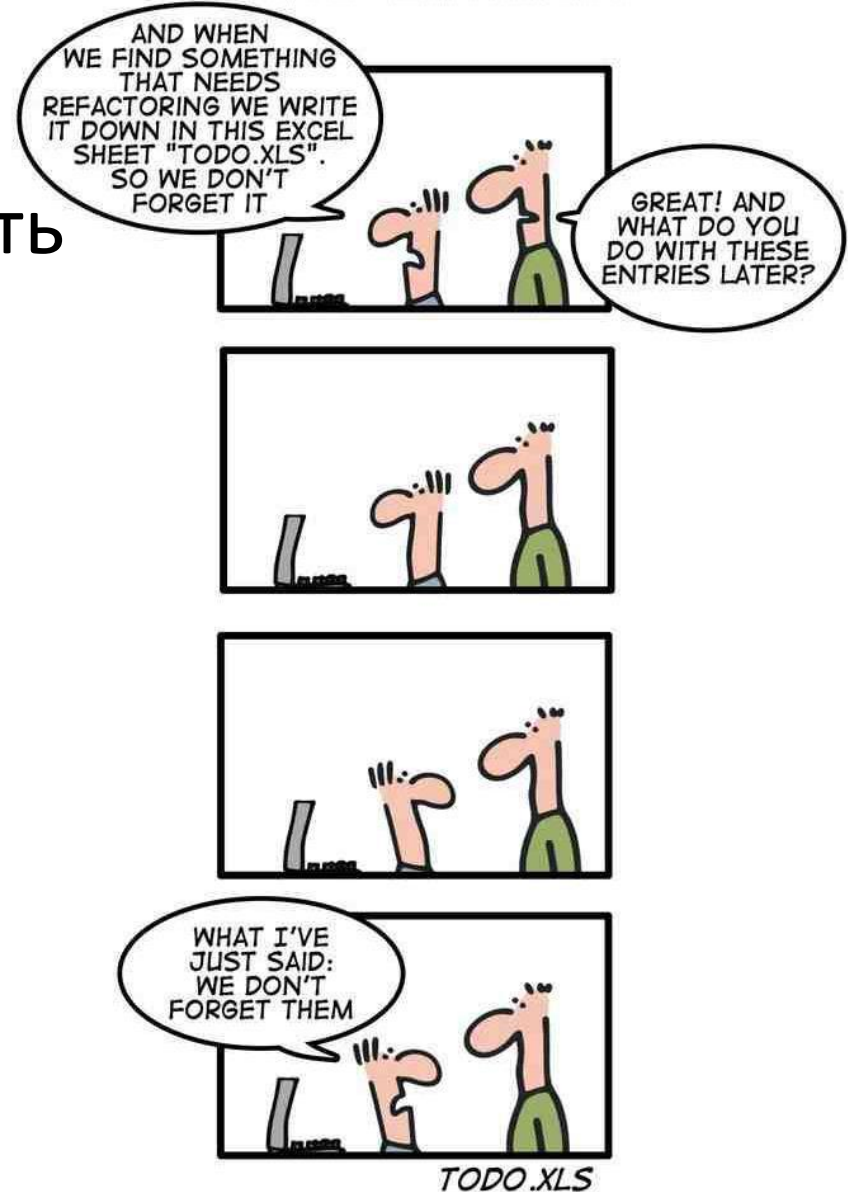


«Плохое» проектирование – 4/12а

- Big Design Up Front
 - Попытка спроектировать все до мелочей и сразу
 - Все проектирование произвести до кодирования
 - Весь дизайн запихивается в архитектуру

«Плохое» проектирование – 5/12а

- Технический долг
 - Осознание долга есть
 - Но с ним ничего не делается



«Плохое» проектирование – 6/12а

- Тайное качество
 - Ненаправленность на требуемое качество или работа на «внутреннее» качество
 - → Не бывает «внутреннего качества»! то есть такого, которое не обеспечивает внешнее, может быть, на долгом периоде
 - Само требуемое качество (как краткосрочное, так и долгосрочное) тоже нуждается в анализе и фиксации
 - Здесь же: мода, авангардизм

«Плохое» проектирование – 7/12а

- Башня из слоновой кости
 - Архитектор творит нечто, что ему кажется прекрасным
 - Всем остальным это непонятно и неудобно
 - Сам архитектор при этом «код не пишет» –он выше этого

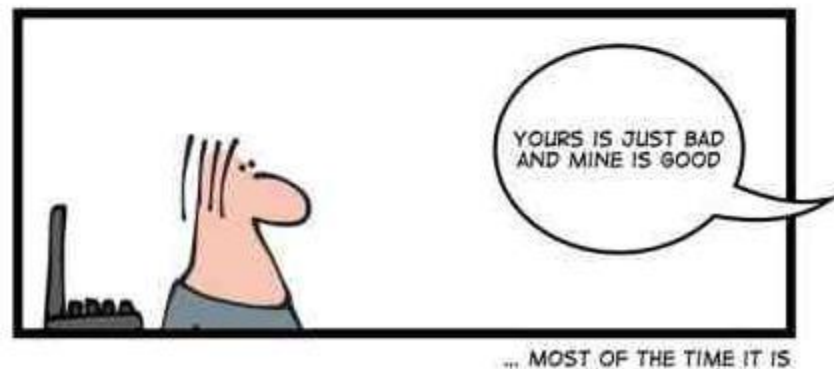
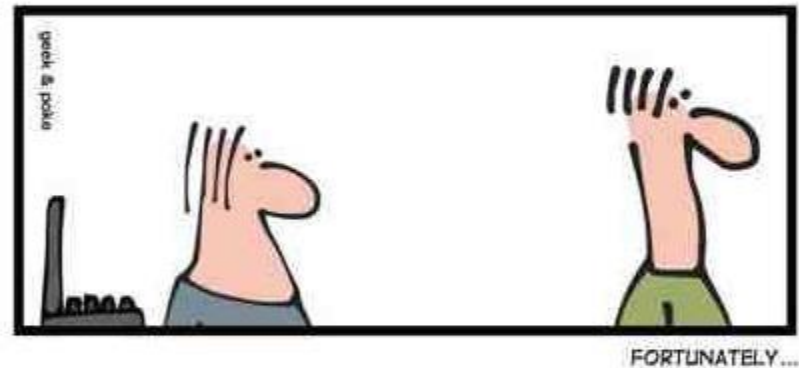
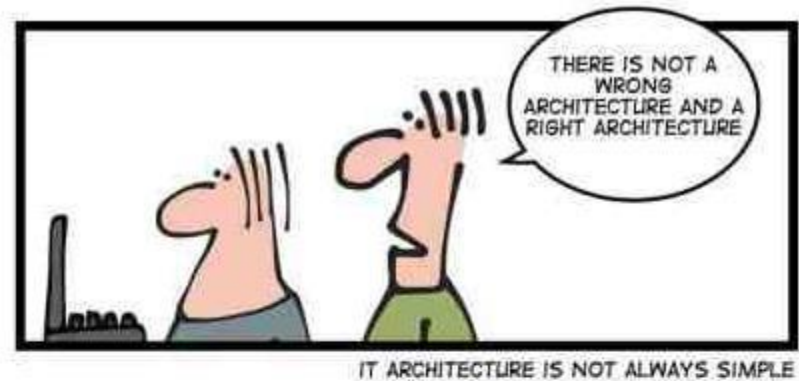


«Плохое» проектирование – 8/12а

- Анархический Agile
 - Отрицается любое главенство
 - Страшилки Ivory Tower и BDUF как оружие
 - «Мы все равны»
 - И поэтому никто не отвечает
 - Принцип «равенства» важнее результата

«Плохое» проектирование – 9/12а

- Главенство авторитета
 - Главенство авторитета над целями
 - Переход на личности
 - сравниваются не решения, а регалии архитекторов
 - Статус важнее результата



«Плохое» проектирование – 10/12а

- Лучший программист
 - Архитектор = самый сильный программист?
 - Специфический предмет. Специфический материал и свойства
 - Специфические компетенции



«Плохое» проектирование – 11/12а

- После нас – хоть потоп
 - Успех оценивается по короткому результату (сдача системы)

«Плохое» проектирование – 12/12а

- Отлито из бронзы
 - «Архитектура от старой системы проверена, возьмем для новой системы ее»
 - Фрэнк Ллойд Райт: «Забудьте обо всех архитектурах мира, если не понимаете того, что они были хороши в своём роде и в своё время»

«Плохое» проектирование – 12а/12а

- Швейцарский нож
 - Универсальная архитектура – одна на сильно непохожие (но кажущиеся похожими) проекты
 - «Там же везде БД, документы, трехзвенка и гриды»
 - Неверно выбран масштаб повторного использования, получается сложно и дорого
 - Software Product Lines



«Хорошее» проектирование

- Экспертная оценка архитектуры
- Архитектура для клиента
 - Обсуждение архитектуры с представителями заказчика и другими интересантами
 - Архитектура представляется в адекватных для них представлениях, отражающих их (различные!) интересы

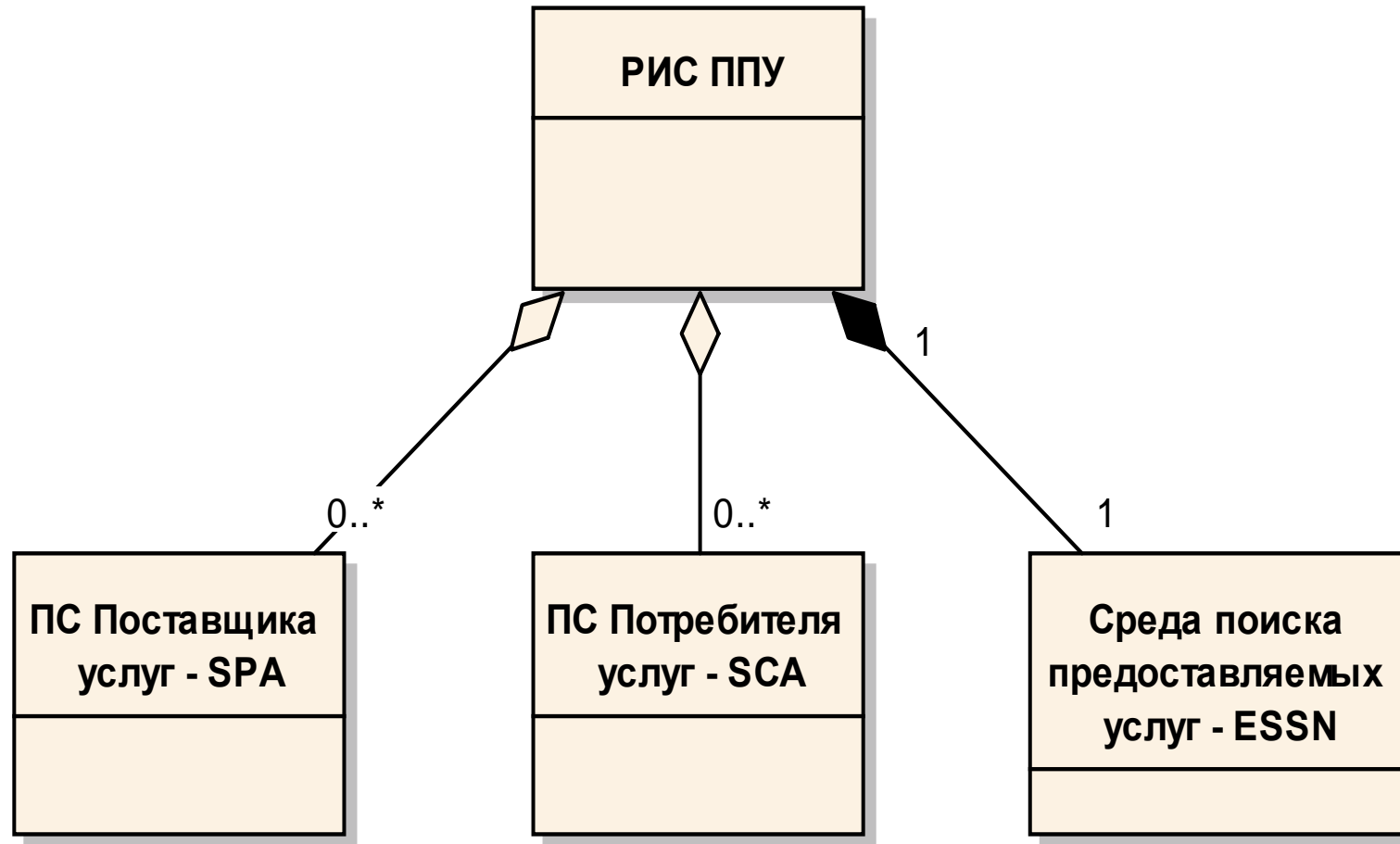
Общая архитектура

Интерфейсы

ПРАКТИЧЕСКИЙ ПРИМЕР: РИС ППУ

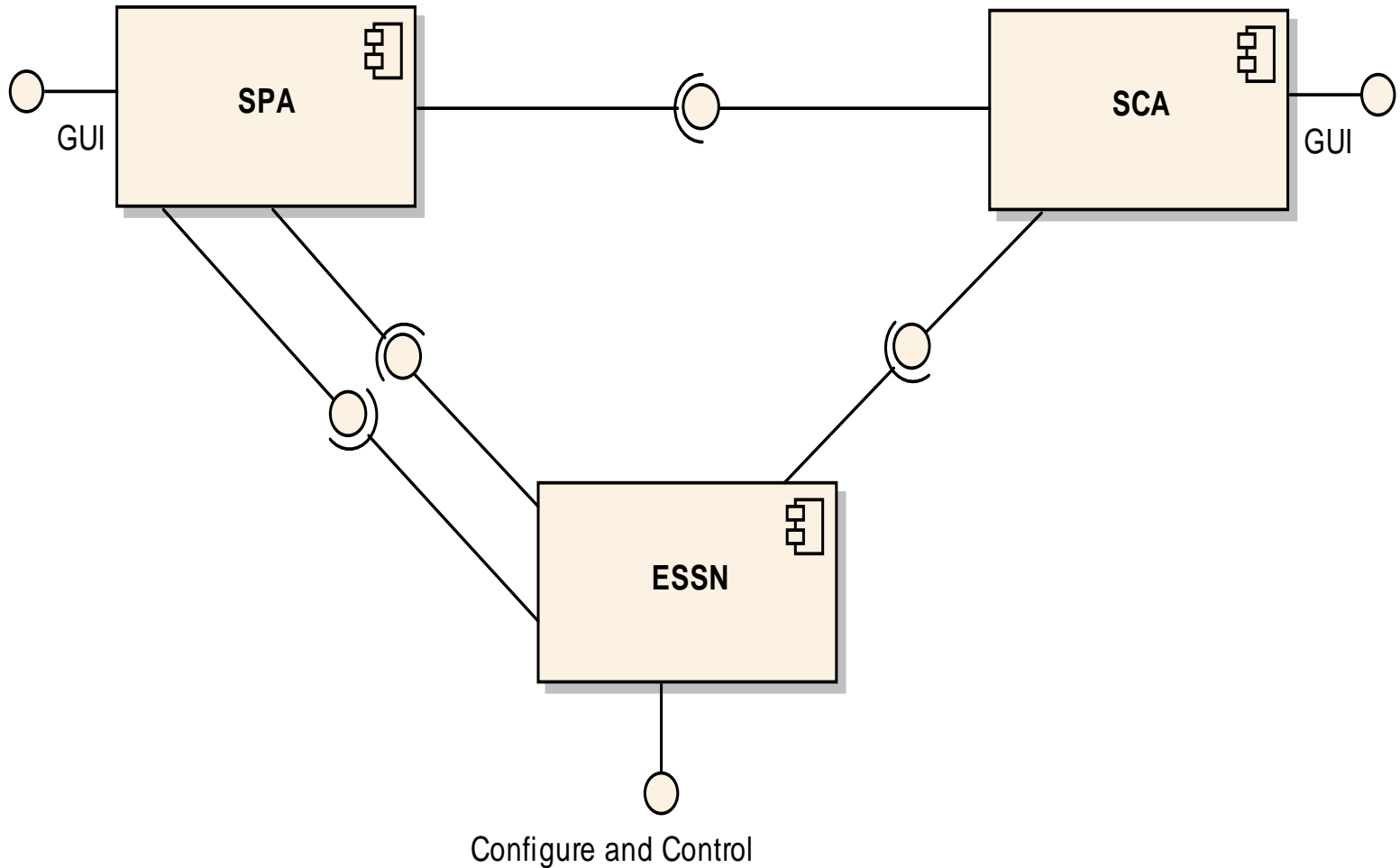
Общая архитектура РИС ППУ

class РИС ППУ

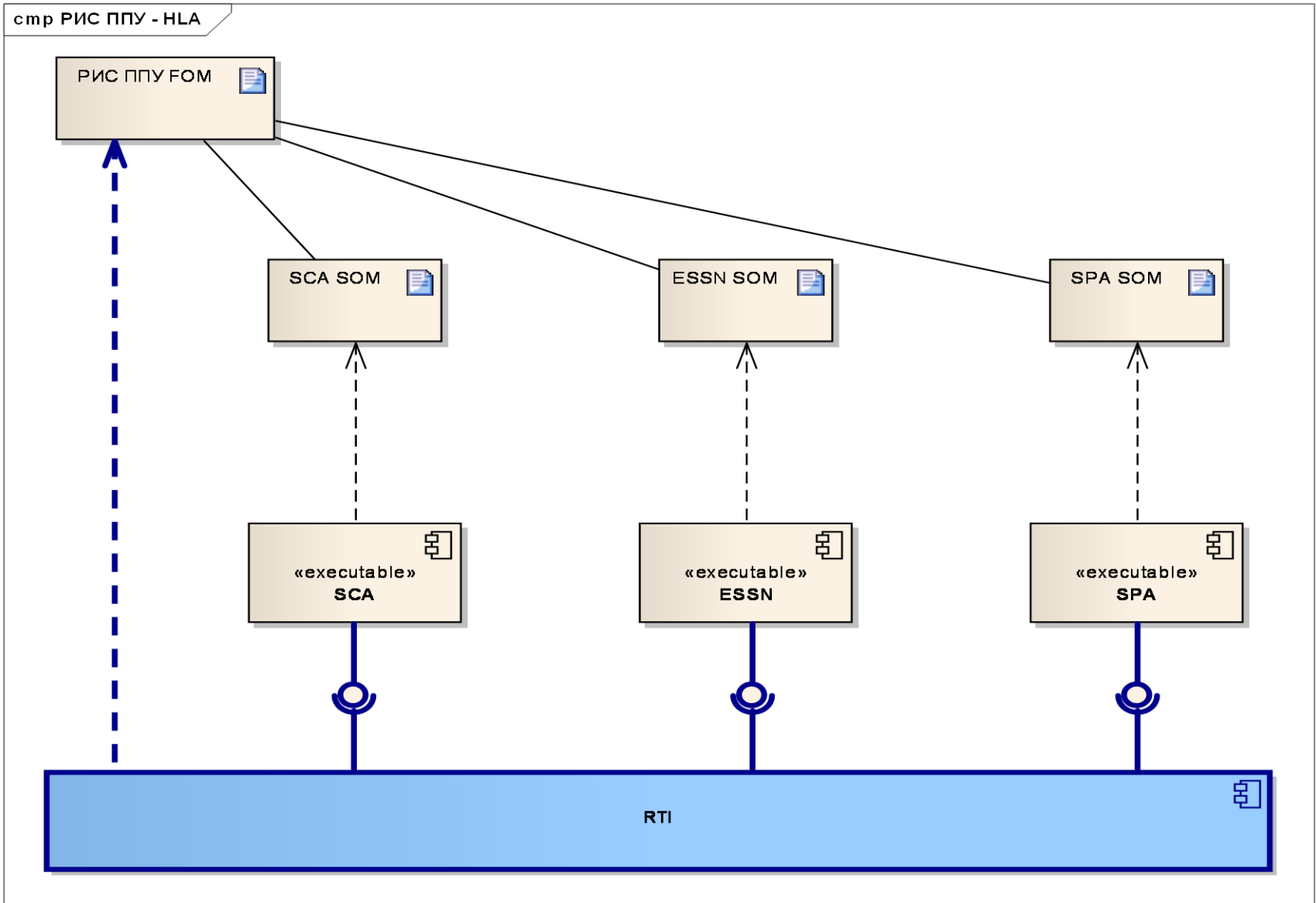


Интерфейсы РИС ППУ

стр Интерфейсы РИС ППУ



ПЗ-2: РИС ППУ на базе HLA RTI



Все пропало

Ужас-ужас

ПРАКТИЧЕСКИЕ ЗАДАНИЯ