

## Лекция 2

**Контекст задачи анализа требований. Выявление требований  
. Формирование *Видения***

- ✓ **Проект** – это уникальная (в отличие от операций) деятельность, имеющая **начало и конец во времени**, направленная на достижение **заранее определенного результата/цели**, создание определенного, уникального продукта или услуги, **при заданных ограничениях по ресурсам и срокам**, а также **требованиям к качеству и допустимому уровню риска**.
  - ✓ Под *управлением проектом* понимают применение к работам по проекту знаний, навыков, средств и технологий в целях удовлетворения потребностей всех заинтересованных сторон или для получения превосходящих их ожидания результатов. Чтобы удовлетворить этим требованиям и ожиданиям, необходимо найти оптимальное сочетание между целями, сроками, затратами, качеством и другими характеристиками проекта.
  - ✓ Управление проектами подчиняется четкой логике, которая связывает между собой различные области знаний и процессы управления проектами.
- 
- 

# Разработка требований

---

## Один из первых и фундаментальных этапов разработке любого ПО

Разработка требований к программному обеспечению – процесс выявления, формулирования, анализа, документирования и верификации требований, подлежащих выполнению в продукте.

В его ходе системный аналитик формирует **реестр требований**, являющийся основой для документа или системы управления требованиями.

Полнота и качество проведения разработки требований играют ключевую роль в успехе всего проекта.

---

# Анализ требований, бизнес-анализ, анализ проблемной области

---

- ✓ Сотни методик, методологий, процессов, стандартов, регламентирующих потоки работ при разработке автоматизированных информационных систем.
- ✓ АТ – начало цепочки работ, результаты определяют успех проекта.
- ✓ Работы, связанные с бизнес-анализом и бизнес-моделированием. Их роль не столь очевидна и принимается далеко не всеми методологиями.

*Стоит ли собирать информацию о предприятии, для которого разрабатывается (выбирается) АИС в виде бизнес-моделей или пропустить этот этап и сразу формировать артефакты АТ?*

---



# Задача анализа бизнес-процессов (деловое моделирование)

---

- Часть более общей задачи анализа проблемной области.
- С позиций моделирования, анализ требований (АТ) и анализ проблемной области (АПО) - принципиально разные процессы.
- АПО преследует классические цели создания модели: налицо объект (автоматизируемое предприятие или организационная система, ОС), задача аналитика - отразить этот объект в создаваемой модели с требуемой степенью точности.
- Анализ требований, напротив, направлен на моделирование воображаемого, еще не существующего объекта (АИС). Т.е. сначала создается модель, а затем, на ее основании, синтезируется объект

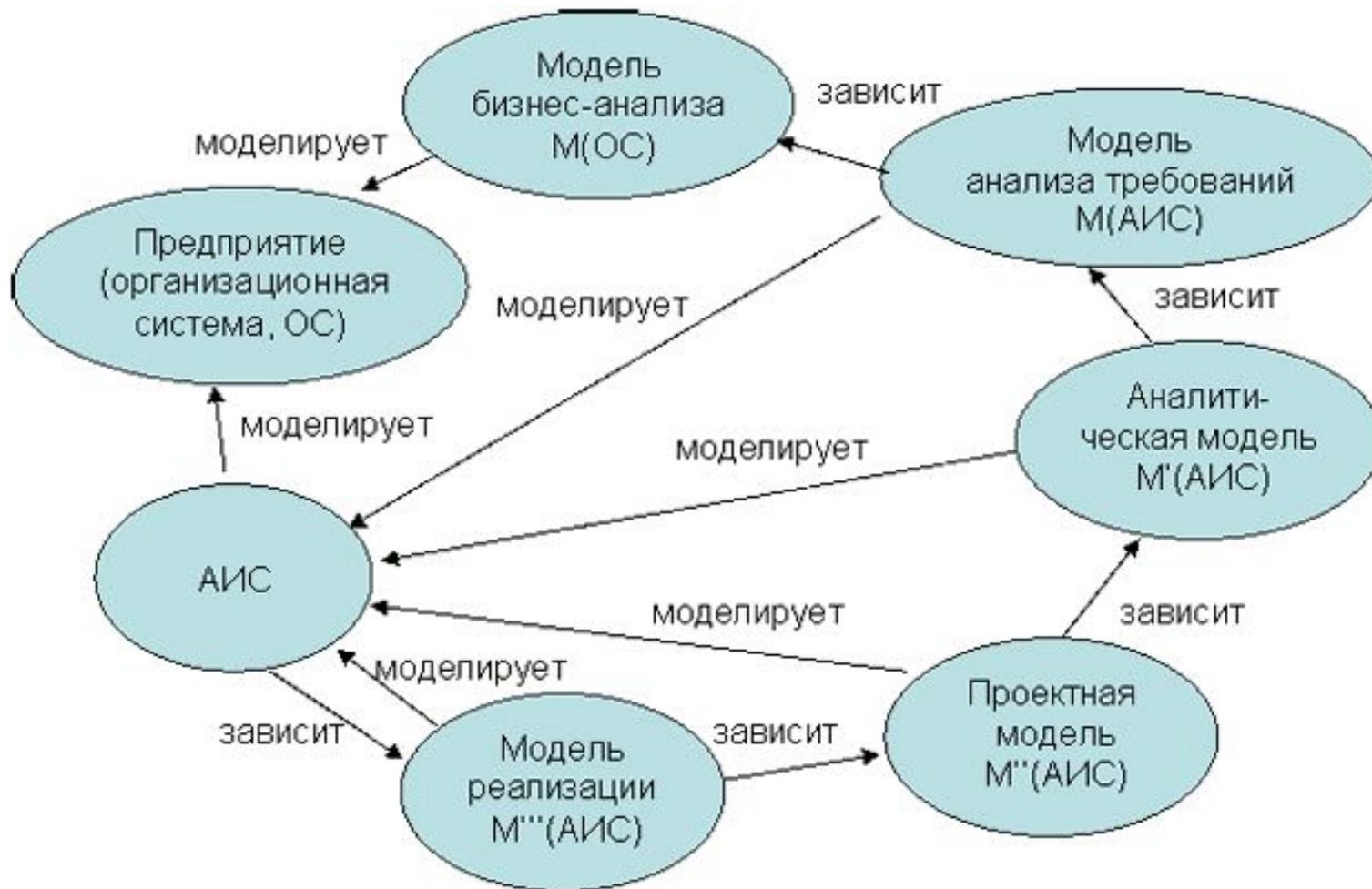
---

Если моделирующий субъект обладает неявными знаниями об ОС в достаточном объеме - АПО можно исключить. На практике это возможно в следующих случаях:

- Разработчик является частью (структурным подразделением, дочерним предприятием и т.д.) ОС, в коллектив Разработчика входят эксперты, хорошо знающие предметную область;
- Заказчик наравне с Разработчиком участвует в создании документа АТ и разделяет с ним ответственность за принятие решений. Это - путь "agile методологий".



# Обобщенная "формула" создания АИС: ОС->М(ОС)->М(АИС)->М'(АИС)->М''(АИС)->М'''(АИС)->АИС



□ Анализируя модель проблемной области, в ней можно вычлениТЬ:

---

1. задачи и функции, реализуемые внутри ОС и функции коммуникации ОС и среды,
2. устройство предметной области (в начале - на уровне *концептуальной модели*),
3. требования к информации и ее обработке.

Выделив среди функций те, которые подлежат автоматизации, получаем основу для выявления функциональных требований к системе.

Остальная собранная на этапе АПО информация служит для поиска нефункциональных требований. В результате получаем модель АТ, как первое приближение модели АИС,  $M(\text{АИС})$ .

Затем, путем углубленного анализа и проектирования, формируются аналитическая модель  $M'(\text{АИС})$ , проектная модель  $M''(\text{АИС})$  и модель реализации  $M'''(\text{АИС})$ .

Модель уровня реализации позволяет синтезировать АИС как совокупность программных, информационных, организационных и др. артефактов.

АИС в свою очередь представляет собой модель организационной системы  $M'(\text{ОС})$ , замыкая цикл моделирования.

---



# Методологии бизнес-анализа

---

по *типам моделей*:

1. модели, преследующие цель анализа и улучшения организационной системы (например, *SWOT* , *VCM*, *BPR*, *CPI/TQM/ISO9000*, *BSC*),
2. модели общего назначения, такие, как *SADT*, *DFD*, *IDEF1*, *IDEF3*, *IDEF5* и др.,
3. модели, специально разработанные для использования при автоматизации (например, *ISA*, *BSP*, *ARIS*, *RUP*).

Наиболее развитая модель описания проблемной области предлагается в методологии *ARIS*.

---



# Архитектура ARIS (*Architecture of Integrated Information Systems*) — архитектура интегрированных информационных систем

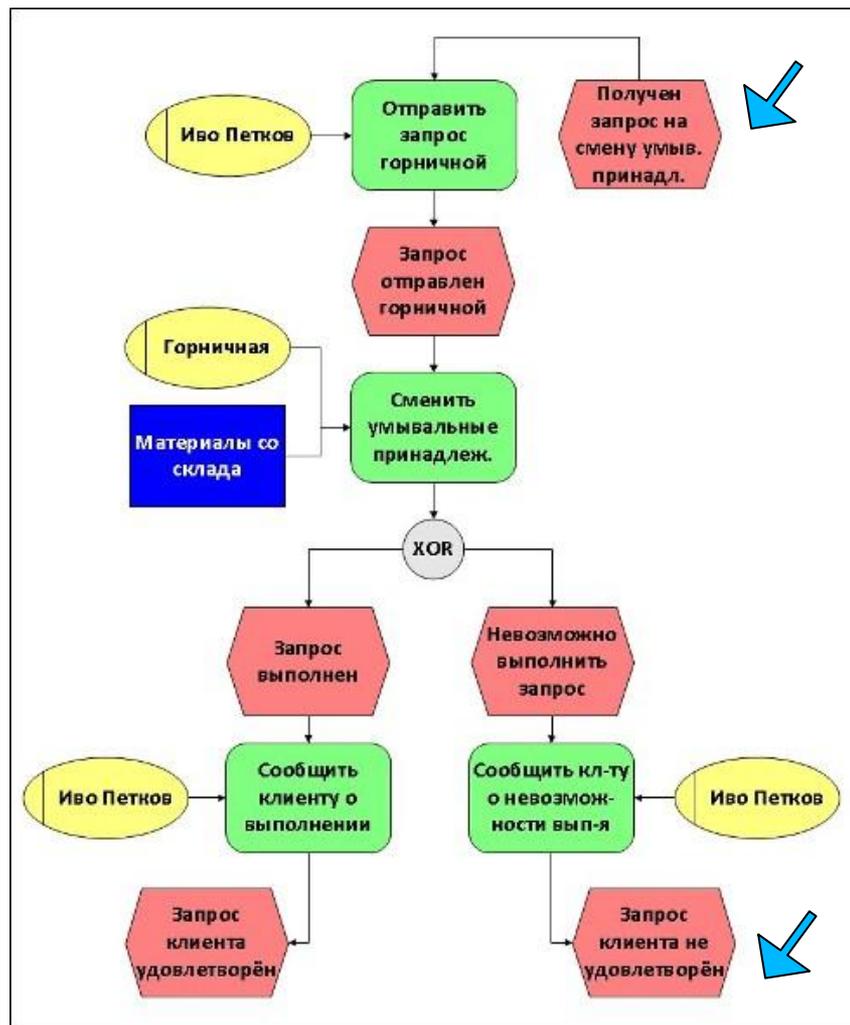
---

- Методология и CASE-средство для моделирования бизнес-процессов организаций с целью их автоматизации от компании Softwareag  
<http://www.softwareag.com/corporate/products/aris>.
- Основные черты методологии – поддержка 5 различных, но взаимосвязанных между собой точек зрения на предприятие; наличие сверхмощного, не имеющего аналогов графического языка, насчитывающего десятки различных диаграмм и сотни специальных символов для всевозможных аспектов деятельности предприятий и его автоматизации. Включает многие наработки из мирового арсенала бизнес-моделирования, инкапсулирован UML. Наиболее известная диаграмма ARIS – EPC (event-driven process chain - событийная цепочка процессов).



# ЕРС-диаграмма процесса обработки запроса клиента в отеле

Привести систему из состояния «Получен запрос от клиента на смену умывальных принадлежностей» в состояние «Запрос клиента удовлетворён».



## Архитектура ARIS. Подсистемы

---

- ▣ *Организационная. Определяет структуру организации - иерархию подразделений, должностей и конкретных лиц, многообразие связей между ними, а также территориальную привязку структурных подразделений.*
  - ▣ *Функциональная. Определяет функции, выполняемые в организации.*
  - ▣ *Подсистемы входов/выходов. Определяют потоки используемых и производимых продуктов и услуг.*
  - ▣ *Информационная (подсистема данных). Описывает получение, распространение и доступ к информации (данным).*
  - ▣ *Подсистема процессов управления. Определяет логическую последовательность выполнения функций посредством событий и сообщений. Можно сказать, что подсистема управления - это совокупность разнесенных во времени сообщений разного рода.*
  - ▣ *Подсистема целей организации. Описывает иерархию целей, достигаемых в ходе выполнения того или иного процесса.*
  - ▣ *Подсистема средств производства. Описывает жизненный цикл основных и вспомогательных средств производства.*
  - ▣ *Подсистема человеческих ресурсов. Описывает прием на работу, обучение и продвижение по службе персонала организации.*
  - ▣ *Подсистема расположения организационных структур. Описывает территориальное расположение организационных единиц.*
- 



# Требования и архитектура АИС

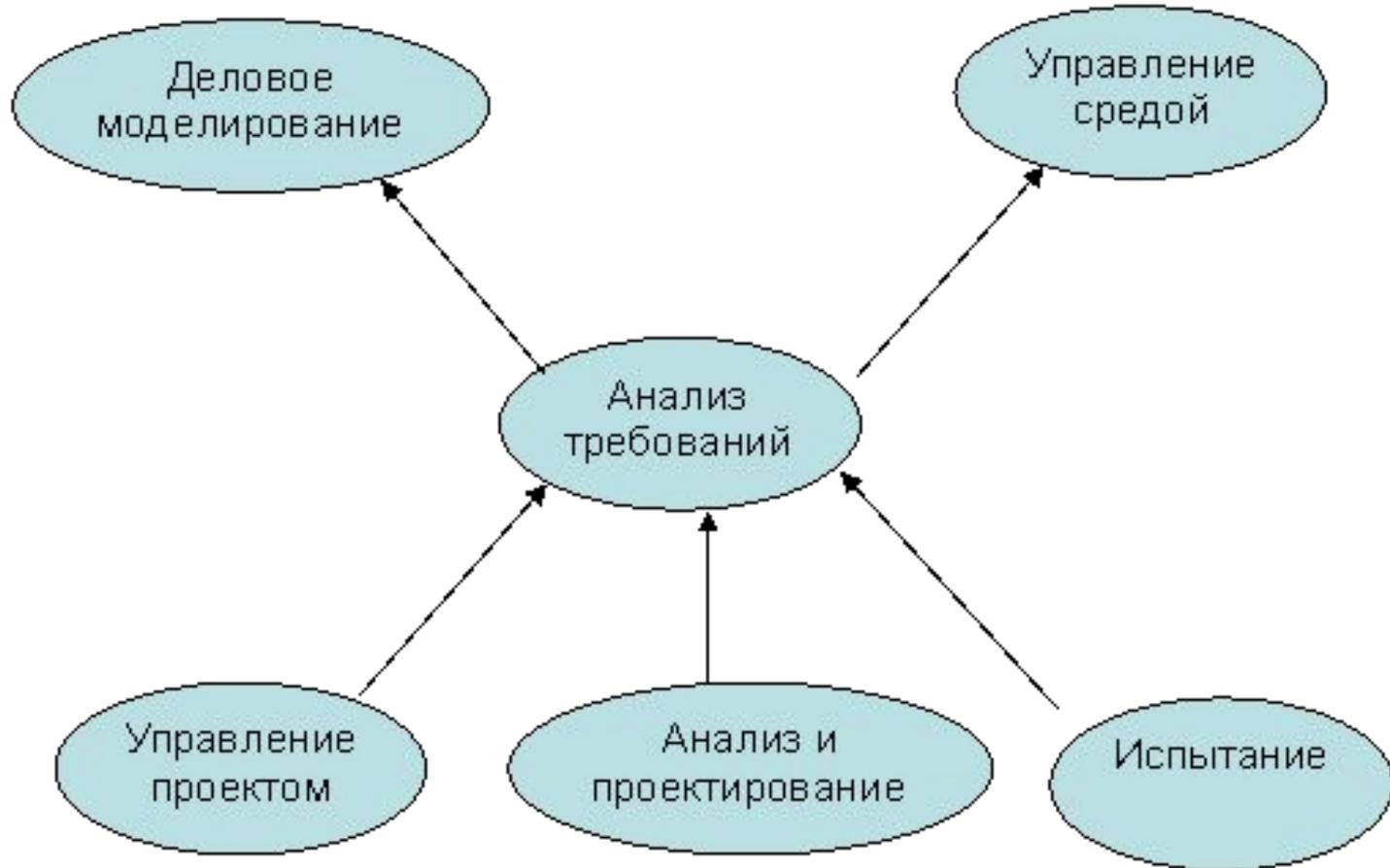
Метафора архитектуры *RUP* — 4+1 представлений: логическое, представление процессов, представление реализации и физическое представление связываются между собой представлением вариантов использования (use case), которое играет центральную роль в выработке архитектуры системы.

Требования первичны по отношению к архитектуре.



# Анализ требований и другие рабочие потоки программной инженерии

---



# Источники требований (1)

---

2 уровня:

- бизнес-требования и
- требования пользователей.

Проблема — требования формулируются к создаваемой, еще не существующей системе (решается начальная подзадача задачи проектирования АИС), а представители Заказчика далеко не всегда бывают компетентны в данном вопросе.

Поэтому, наряду с требованиями, высказанными Заказчиком, целесообразно собирать и требования от других совладельцев системы: сотрудников аналитической группы исполнителя, внешних экспертов и т.д.

- Результирующий, часто достаточно сырой материал рассматривается, как документ "Требования совладельцев"



## Источники требований (2)

---

Модель создаваемой информационной системы в определенной мере должна отражать модель ОС.

- важным источником информации являются артефакты, описывающие предметную область. Это могут быть документы с описанием бизнес-процессов предприятия, документы (должностные инструкции, распоряжения, своды бизнес-правил), принятые на предприятии.
- "лучшие практики" — описания моделей деятельности успешных компаний отрасли, используемые длительное время в сотнях и тысячах компаний по всему миру.

Основные источники, образующие "вход" процесса выявления требований, — требования, высказанные совладельцами, как таковые или (и) артефакты, описывающие объект исследования.

Чтобы данные поступили "на вход", аналитики требований должны проделать большую работу, связанную с подбором респондентов и информационных материалов, организацией интервью и т.д.

---



# Стратегии выявления требований

---

- **Интервью**
- **Анкетирование**
- **Наблюдение**
- **Самостоятельное описание требований**
- **Совместные семинары**
- **Прототипирование**



# Интервью

---

## Ключевая стратегия выявления требований

- подготовка,
- проведение интервью (опроса),
- завершение.



# Подготовка

---

Позволяет спланировать процесс опроса и выработать стратегию управления им.

Важность подготовительного этапа возрастает, если респондент является "дефицитным" полезным ресурсом (президентом крупной компании).

Рекомендации:

- выберите нужного собеседника;
- договоритесь о встрече;
- установите предварительную программу встречи;
- изучите сопутствующую информацию;
- согласуйте свои действия с группой проектирования .

При выборе собеседника для целей сбора требований определяющими являются:

- 1) Он действительно является экспертом по данному вопросу;
  - 2) Его мнение действительно является ценным при формировании целевого набора требований.
- Заранее оговорить цель встречи и ограничить беседу в пределах часа или менее.
  - Полезными приемами являются формирование программы беседы и ознакомление с ней респондента, подробное планирование беседы вплоть до записи подготовленных вопросов. Такое интервью называют структурированным . В дополнение к нему – неструктурированное интервью (неформальная встреча). Цель такого интервью – побудить респондента к

# Проведение опроса

---

- Правильно организовать и поддерживать поток информации от эксперта к аналитику.
- Начало разговора: представиться и сформулировать цель встречи, обговорить возможность ведения записей.
- Первый вопрос. Задаёт тон всему разговору, поэтому хорошо продумайте его.
- Делайте записи обо всем (о специальных терминах, взаимосвязях между частями системы и т.п.) и ограничивая время беседы. Запишите *SADT*-функции и данные, попытайтесь набросать диаграмму. Задавайте вопросы, которые уточняют и подтверждают ответы.
- Не возражайте.
- Никогда не задавайте наводящих вопросов или да/нет вопросов. Записывайте то, что вам говорят, и просите подвести итог или дать пояснения.



# Завершение

---

Достаточное основание для завершения беседы – одна из следующих причин:

- вы уже получили достаточно информации;
- вы получаете большой объем неподходящей информации;
- обилие информации вас подавляет;
- эксперт начинает уставать;
- у вас с экспертом часто возникают конфликты.

- 
- Кратко подытожьте основные пункты и сделайте обзор полученных сведений, которые могут быть опущены или неверно истолкованы. Договоритесь о времени следующей встречи, получите рекомендации для ближайших опросов. Поставьте эксперта в известность, когда и как вы собираетесь использовать полученную информацию и когда вы пришлете ему материал на рецензирование.
  - Всегда оформляйте материалы опроса сразу же после встречи с экспертом. В этом случае немедленно возникает обратная связь, и вы минимизируете возможность потери важной информации.



## Рекомендации при опросе

---

- ▣ *делайте паузы, пока эксперт думает. Никогда не перебивайте, подсказывая ответ или задавая другой вопрос;*
  - ▣ *старайтесь не задавать наводящих вопросов, вопросов-подсказок, вопросов, содержащих ответ, потому что это не позволяет эксперту делиться своими знаниями;*
  - ▣ *делайте записи, чтобы сосредоточиться на предмете разговора и чтобы подготовиться к следующему вопросу, но не стенографируйте, иначе вы можете потерять контроль над опросом.*
- 



# Анкетирование

---

Самый малозатратный для аналитика и наименее эффективный способ извлечения информации. Обычно применяется как дополнение к другим стратегиям выявления требований.

Недостатки: респонденты часто бывают неспособны, либо слабо мотивированы, чтобы хорошо и информативно заполнить анкету. Велика вероятность получить неполную или ложную информацию.

Преимущество: подготовка и анализ анкет требуют небольшой ресурс.

Известна рекомендация формулировать в анкетах вопросы с замкнутым циклом ответов в одной из следующих трех форм.

- Многоальтернативные вопросы. Может расширяться комментариями респондента в свободной форме.
  - Рейтинговые вопросы. Представляют predetermined набор ответов на сформулированные вопросы. Используются такие значения, как "абсолютно согласен", "согласен", "отношусь нейтрально", "не согласен", "абсолютно не согласен", "не знаю".
  - Вопросы с ранжированием. Предусматривает ранжирование (упорядочивание) ответов путем присваивания им порядковых номеров, процентных значений и т.п.
- 



# Наблюдение

---

Полезная стратегия получения информации (строго говоря, по результатам наблюдения можно получить модель ОС, а не модель АТ).

Пассивное и активное наблюдение. При активном наблюдении аналитик работает, как участник команды, что позволяет улучшить понимание процессов.

Через наблюдение аналитики получают информацию о происходящих день за днем операциях из первых рук. Во время наблюдения за работой системы часто возникают вопросы, которые никогда бы не появились, если бы аналитик только читал документы или разговаривал с экспертами.

Недостаток:

наблюдатель, как и всякий "измерительный прибор", вносит помехи в результаты измерений: сотрудники организации, находясь "под колпаком" могут начать вести себя по-другому, чем обычно.

---



# Самостоятельное описание требований

---

Документы – хороший источник информации. Чтение документов – прекрасный способ получить первоначальное представление о системе и сформулировать вопросы к экспертам.

- По результатам анализа документов и собственных знаний аналитик может составить описание требований и предложить его представителям Заказчика в качестве информации к размышлению, либо основы для формирования технического задания.
- Недостаток этой стратегии – опасность пропуска знаний, специфичных для объекта исследования, либо неформализованных знаний, эмпирических правил и процедур, широко используемых на практике, но не вошедших в документы.



# Совместные семинары

---

Правила мозгового штурма предполагают полную раскрепощенность и свободу мнений. Первое правило мозгового штурма – полный запрет на любую критику – стимулирует творческую фантазию.

На втором этапе, все высказанные мнения тщательным образом обсуждаются, заведомо неприемлемые варианты отсеиваются, формируются коллективные предложения.

Правила JAD-метода(joint application design ), считающегося одним из современных способов извлечения требований, были впервые сформулированы в конце 1970-х годов компанией IBM. Участники JAD-совещания:

- ▣ **Ведущий** - специалист в области межличностных коммуникаций. Должен ориентироваться в предметной области, но не обязательно хорошо ориентироваться в проблемах IT.
- ▣ **Секретарь**. Фиксирует ее результаты на компьютере. Возможно применение CASE-средств.
- ▣ **Заказчики** - пользователи или руководители, основные участники, формирующие, обсуждающие требования и принимающие решения.
- ▣ **Разработчики** - аналитики и другие участники проектной команды. Работают в большей части в пассивном режиме с целью наилучшего понимания проблемной области.

Преимущество: работа в группе более продуктивна, группы быстрее обучаются, более склонны к квалифицированным заключениям, позволяют исключить многие ошибки.

Одна из самых затратных стратегий, однако окупается за счет меньшего количества ошибок и отказе от формализации в пользу живого общения, выработке общего

# Прототипирование

---

Ключевая стратегия выявления требований в большинстве современных методологий

Программный прототип - "зеркало", отражающее, как понял Исполнитель требования Заказчика.

Метод RAD (*rapid application development*) – один из наиболее известных способов быстро создавать прототипы.

RAD базируется на следующих базовых принципах:

- Эволюционное прототипирование;
  - CASE-средства, как основной инструмент, включая возможности прямого и обратного проектирования и автоматической генерации кода;
  - Высококвалифицированные специалисты, хорошо владеющие развитыми инструментальными средствами;
  - Интерактивный JAD-метод (Joint Application Development or Design) - совместная разработка (или проектирование) приложений, в котором общение совмещается с разработкой в режиме on-line;
  - Жесткие временные рамки (как противовоидие от "расползания границ" проекта): если команда не укладывается в срок - функционал сужается.
- 



# Формирование видения

---

Концепция служит для того, чтобы помочь Заказчику выявить именно те требования к системе, которые помогут ему оптимизировать работу своего предприятия в долгосрочной перспективе.



# Концепция в ГОСТ РФ (1)

---

В соответствии с ГОСТ 34.601-90 "Автоматизированные системы. Стадии создания" после этапа формирования (выявления) требований к системе выполняется этап разработки концепции системы.

Основные работы этапа:

- Изучение объекта;
- Проведение научно-исследовательских работ (НИР);
- Разработка вариантов концепции АС;
- Оформление отчета о выполненной работе.

Так как данный этап хронологически стоит на втором месте, к его началу у Разработчика на руках уже имеется документ, в котором собраны основные требования пользователей.

- Работы над концепцией начинаются с обследования объекта автоматизации. Выполняются НИР, направленные на исследование принципиальной реализуемости требований и возможных вариантов реализации.
- 
- 

## Концепция в ГОСТ РФ (2)

---

ГОСТ, в отличие от большинства современных методологий, в общем случае закладывает *многоальтернативность* вариантов концепции системы и планов их реализации.

Каждый из проработанных вариантов оценивается с позиций требуемых ресурсов и функциональности.

Для вариантов представляются оценки преимуществ и недостатков. Полезность проработки нескольких вариантов концепции заключается в том, что Заказчику трудно сформулировать самостоятельно видение системы, в то время, как выбор из набора вариантов, представленных Разработчиком – вполне посильная задача.

Концепция должна отражать оценки качества, условия приемки системы, оценку эффекта, ожидаемого от реализации.

При оформлении отчета необходимо привести обоснование предлагаемого варианта.

---



# Принципы RUP

---

- Ранняя идентификация и непрерывное (до окончания проекта) устранение основных рисков.
- Концентрация на выполнении требований заказчиков к исполняемой программе (анализ и построение модели прецедентов (вариантов использования)).
- Ожидание изменений в требованиях, проектных решениях и реализации в процессе разработки.
- Компонентная архитектура, реализуемая и тестируемая на ранних стадиях проекта.
- Постоянное обеспечение качества на всех этапах разработки проекта (продукта).
- Работа над проектом в сплоченной команде, ключевая роль в которой принадлежит архитекторам.



# Шаблоны RUP

---

Видение

Варианты использования

Детализированное описание ключевых вариантов  
использования

Анализ и спецификация специальных требований

**Спецификация требований**



# Видение в RUP

---

*[методология IBM Rational Unified Process]*

*Документ «Видение проекта» (Vision) — краткое описание сути будущего продукта — разрабатывается на основе анализа потребностей бизнеса.*

**Главная функция документа** – подтверждение и согласование единого видения целей, задач и результатов всеми участниками проекта.

Видение определяет что и зачем делается в проекте.

Видение проекта - это ключевой документ, который используется для принятия решений в ходе всего проекта, а также на фазе приемки — для подтверждения результата.

---

# Видение в RUP. Этапы

---

- Формулировка проблем.
- Идентификация совладельцев
- Определение границ системы
- Идентификация ограничений
- Формулировка постановки задач
- Определение возможностей системы
- Оценка результатов



# Глоссарий

---

## Роль глоссария при АТ

Глоссарий рассматривается как документ, удостоверяющий общее понимание основной терминологии Заказчиком и Разработчиком.

Помимо формирования требований совладельцев другим результатом начальной фазы выявления требований является концептуальный анализ проблемной области. Самым первым результатом его является формирование глоссария (словаря) основных используемых терминов. Значение глоссария трудно переоценить: он является основой, ключом для единообразного понимания описаний требований Заказчиком и Разработчиком.

- Глоссарий является отправной точкой для построения более развернутых моделей проблемной области, которые, на стадии реализации информационной системы, ложатся в основу объектной модели (для объектно-ориентированных приложений) и модели данных (для генерации схемы базы данных).
  - Глоссарий оформляется, как текст, состоящий из абзацев, каждый из которых определяет значение одного из терминов проблемной области. Термин обычно выделяют полужирным кеглем. Иногда проблемную область целесообразно сегментировать на ряд "подобластей" (subject areas). Тогда каждой из них в глоссарии выделяется отдельный параграф.
-

# Пример

---

## Глоссарий терминов

**Актор** - actor – лицо, играющее особую роль, система ПО или аппаратное устройство, которое взаимодействует с системой для достижения полезных целей.

**Анализ требований** - analysis, requirements – процесс классификации информации, касающейся требований, по различным категориям, оценка требований для определения желаемого качества, представление требований в различных формах, выделение детальных требований из требований более высокого уровня, обсуждение приоритетов требований и т.д.

**Аналитик требований** - requirements analyst – роль члена команды по разработке требований. Аналитик отвечает за работу с заинтересованными лицами за извлечение, анализ, определение, утверждение требований и за их управление.

**Архитектура** - architecture — структура системы, включающая ПС и оборудование (они, собственно говоря, и составляют систему), взаимосвязи между этими компонентами и поведение компонентов, видимое другим компонентам.



# Видение / рамки в MSF

---

**Основными задачами** фазы выработки концепции являются создание ядра проектной группы и подготовка **документа общего описания и рамок проекта** (vision/scope document).

Формирование видения проекта и специфицирование его рамок – не одно и то же.

Видение (vision) – ничем не ограничиваемое представление о том, каким должно быть решение .

Рамки (scope) дают четкие границы того, что из предложенного этим видением будет реализовано в условиях существующих проектных ограничений.

**Управление рисками** представляет собой итеративный процесс, осуществляемый на протяжении всего жизненного цикла проекта. Во время фазы выработки концепции проектная группа готовит документ оценки рисков и представляет главные риски проекта вместе с общим описанием и рамками проекта.

Во время фазы выработки концепции производится выявление и анализ **бизнес-требований**. Более детально эти требования рассматриваются во время фазы планирования.

Ведущим *ролевым кластером* на фазе выработки концепции является "Управление продуктом".

---



# Ролевые кластеры методологии MSF

---



# *Ролевые кластеры методологии MSF*

---

- Управление продуктом ( Product Management) . Ключевая цель данного ролевого кластера – удовлетворение заказчика.
  - Управление программой ( Program Management ). Основная задача этого ролевого кластера – обеспечить реализацию решения в рамках ограничений проекта. Для этого контролируются календарный график проекта, объем работы и отведенный на проект бюджет. Рассматриваемый кластер обеспечивает своевременное достижение требуемых результатов и удовлетворение ожиданий спонсора на протяжении проекта.
  - Разработка ( Development) . Первостепенной задачей ролевого кластера "Разработка" является построение решения в соответствии со спецификацией.
  - Тестирование ( Test) . Задача данного ролевого кластера – одобрение выпуска продукта только лишь после того, как все дефекты выявлены и улажены.
  - Удовлетворение потребителя ( User Experience) . Цель этого ролевого кластера – повышение эффективности использования продукта.
  - Управление выпуском ( Release Management) . Цель этого ролевого кластера – беспрепятственное внедрение и сопровождение продукта. Эта роль служит связующим звеном между проектной группой и группами процессов сопровождения.
- 



# Разделы шаблона MSF

---

## **Бизнес-преимущества**

- Описание преимуществ
- Формулировка видения
- Анализ выгод

## **Концепция решения**

- Цели, задачи, предположения и ограничения
- Анализ применимости
- Требования

## **Рамки**

- Список характеристик/функций
- Вне рамок
- Стратегия подготовки релизов
- Критерии применимости
- Эксплуатационные критерии

## **Стратегии проектирования решения**

- Стратегия проектирования архитектуры
- Стратегия технического проектирования

