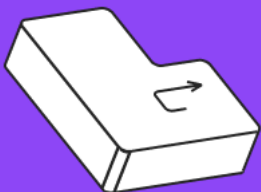


Концептуальное проектирование

Технологическая
специализация системный
аналитик



Оглавление

Введение	2
Термины, используемые в лекции	2
Роль концептуального проектирования и его место в технологическом прогрессе	3
Определение концепции системы	5
Проведение концептуального исследования	7
Уточнение и учет требований	10
Семантическое моделирование	12
Апробация концепции	15
Что можно почитать еще?	18
Используемая литература	18

Введение

На предыдущем курсе мы изучили основы системного анализа и научились управлять требованиями к создаваемым системам. В курсе мы детально рассмотрим вопросы создания сложных информационных систем. Вы изучите основы проектирования и получите навыки описания систем с нулевого уровня зрелости.

Цель данного курса, подготовить специалиста способного максимально быстро влиться в рабочий процесс создания и развития программных продуктов, который сможет принести пользу команде, а продукт уберечь от типичных ошибок проектирования.

Данная лекция открывает курс проектирования и задает основы подхода к проектированию отвечая на вопросы о начальной стадии проектирования сложных и запутанных систем.

Термины, используемые в лекции

Система – совокупность взаимодействующих или взаимосвязанных элементов.

Модель системы – формализованное описание системы.

Системный анализ – совокупность приемов научного познания представляющий собой последовательность действий по установлению структурных связей между переменными или элементами исследуемой системы.

Наблюдатель – лицо обозначающее точку зрения, относительно которой происходит описание системы.

Концепция – комплекс взглядов на что-либо, связанных между собой и образующих единую систему, являющуюся генеральной линией для анализа и проектирования системы

Концепт – воплощение идеи

User Story – пользовательская история служащая для выявления требований пользователя системы

MVP – минимально жизнеспособный продукт, являющийся концептом системы

Роль концептуального проектирования и его место в технологическом прогрессе

Вспомним основные понятия, без которых сложно будет изучать данный курс. Прежде всего это понятие **системы**.

Систему будем определять определить как совокупность взаимодействующих или взаимосвязанных элементов. Система обладает целостностью, то есть свойства элементов системы не тождественны свойствам систем. Также система однозначно может быть выделена из среды.

Вторым важным понятием для целей проектирования систем является **модель системы**, представляющее абстрактное математическое описание системы.

Выделение концептуальных исследований в самостоятельный этап исследований объектов исходит из реалий практики создания информационных систем. По своей природе большинство информационных систем являются сложными и состоят из большого числа элементов, связи между которыми не вполне очевидны на первый взгляд.

Ошибка при начальном этапе проектирования системы является самой дорогой для исправления и сравнима с ценой ошибки при извлечении требований заинтересованных сторон.

В случае, если задача является новой (новаторской, инновационной), то результат реализации системы выполненной без полноценного проведения исследования описываемого объекта будет очевидно непредсказуем.


К счастью сказать, большинство практических задач ранее были теоретически описаны и решены, а значит существует возможность переиспользовать шаблон решения. Но есть два обстоятельства, которое позволяют говорить об актуальности концептуального исследования при старте проектирования системы.

Прежде всего это большая вариативность синтеза связей объектов исследования, ведь при N элементах в системе возникает факториал N состояний связей. В простых очевидных системах из двух-трех элементов эти взаимодействия могут быть вполне очевидны. Итоговая оценка в 2-6 варианта внутренних взаимодействий выглядит вполне доступным для анализа без проведения отдельного исследования. Но как быть если в системе существует порядка 10 элементов?


Эти элементы могут взаимодействовать, при этом итоговая вариативность упорядоченных взаимодействий элементов системы превышает три миллиона. Системы из 100 элементов не могут быть рассчитаны без применения квантовых вычислительных мощностей. Таким образом нахождение оптимальной конфигурации сложных систем является актуальной и трудно решаемой задачей.

Второй особенностью является необходимость выбора одного из общепринятых шаблонов проектирования, и настройка его под свои нужды.

Для иллюстрации этого тезиса давайте вспомним о выдающихся победах одних концепций, которые привели к величайшим провалам других.



Вспомните самые выдающиеся технологические сдвиги, которые привели к потере актуальности множеством систем



Ответ: на предыдущих лекциях мы вспомнили катастрофу американского шаттла «Колумбия». Несомненно, на этапе детального описания системы было упущено нештатное взаимодействие одной из термозащитных пластин с корпусом корабля. Но только случай выявил концептуальную проблему возможности взаимодействия случайных элементов термозащиты со случайными элементами корпуса. Именно ошибка в концепции явилась основанием прекращения использования аппаратов. Им даже не помогла модернизация.

На смену пилотируемым кораблям пришли беспилотные многоразовые челноки, которые обнулили концепцию одноразовых ракет.

Но не будем долго говорить о том как «космические корабли бороздят просторы мирового океана», хотя о творении Элона Маска можно сказать именно так!

В качестве другого примера можно привести появление смартфонов. Лишь появившись в 1998, первый полноценный смартфон Alcatel OneTouch COM не завершил эпоху кнопочных телефонов, потому что сам по себе ввод информации через экран не решает пользовательской проблемы скорости и удобства использования, но со временем концепция использования физической клавиатуры для ввода информации в телефоне себя полностью изжила. И теперь на смену сенсорному дисплею приходит уже голосовой ввод информации, распознаваемой с использованием глубоких нейронных сетей и предиктивные модели, способные предугадывать желания потребителя.

Еще один актуальный пример касается сельского хозяйства, где сетевая концепция мониторинга состояния и управления режимами работы комбайнов полностью замещает концепцию управления специалистом. Современный комбайн

представляет собой сложный роботизированный комплекс с внешним контролем, в то время как комбайнёр стал пилотом, способным отреагировать на нештатную ситуацию, но от которого не требуется непрерывное управление процессом.

Но говоря о роли и месте концептуального моделирования, нельзя не отметить смену концепции Web 1.0 наступившую с появлением социальных сетей и развитие концепции Web 2.0. При этом технологического изменения не происходило, происходило лишь переосмысление места и роли пользователя в процессе использования глобальной сети. Так изначальный концепт предполагал лишь восприятие информации пользователями глобальной сети Интернет, а роль поставщика информации отводилась специализированным сайтам.

С пересмотром концепции основным поставщиком информации в сети стали её пользователи. Это потребовало решения проблемы их авторизации, так как изначальный интернет предполагал анонимную культуру потребления информации. Теперь же для все стало нормой что сайты распознают социальный профиль, а резонансная новость опубликованная на личной странице суперсайта, становится достоянием общественности.

Это глобальное перестроение концепции не потребовало никаких технических изменений, но привела к расцвету рынка Интернет решений и породила массу процессных и технологических новшеств.

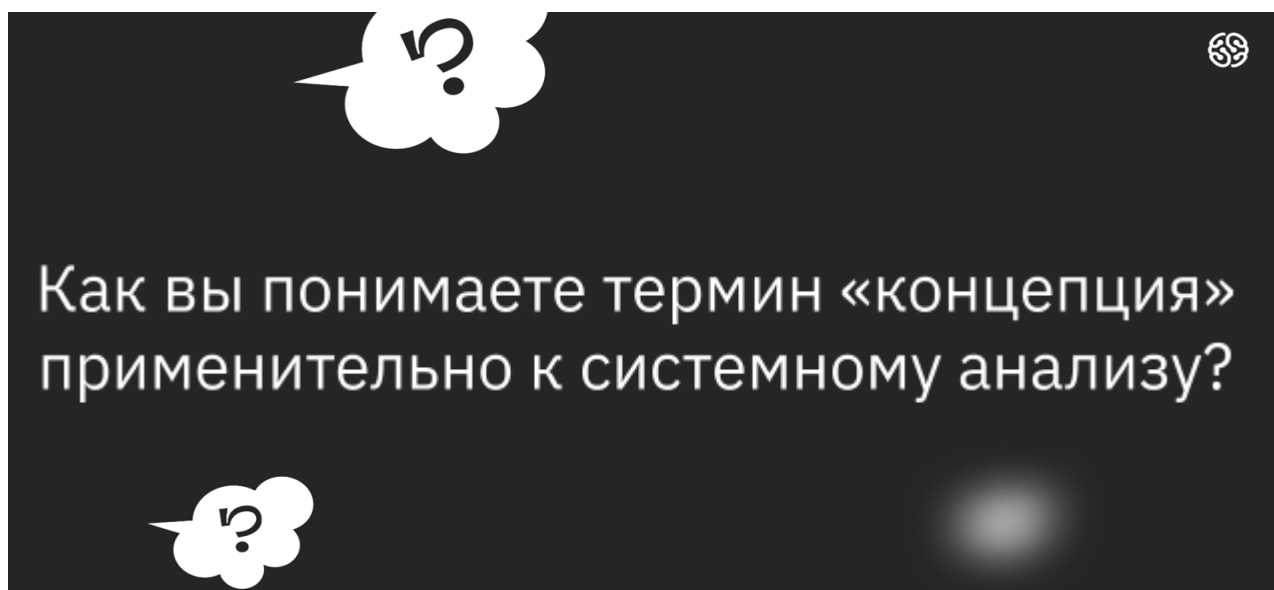
Из рассмотренных примеров можно заметить, что концептуальное моделирование не строго следует за техническим прогрессом. А наоборот, технологический прогресс достигается за счет смены концептуальных шаблонов использующих инструментарий актуальных технических решений.

В программных приложениях ситуация обстоит соответствующим образом. Прогресс развития информационных технологий происходит стремительно. Например, общепринятые концепции передачи сообщений подвергаются тщательной проверке в множестве систем и постепенно появляются технические решения способные видоизменить до неузнаваемости казалось бы незыблемые концепты. Так в последнее время во многих прикладных информационных системах технологии распределенного журналирования приходят на смену технологиям передачи сообщений, что выражается в вытеснении технологии асинхронной коммуникации между сервисами с использованием сообщений (Message Queue – MQ), технологиями на основе Apache Kafka.

Стадия концептуального проектирования начинается с детального анализа первичных данных и уточнения концептуальной модели данных, после чего проектируется архитектура системы. При этом оценивается возможность использования существующих ИС и выбирается соответствующий метод их преобразования. После построения проекта уточняется исходный бизнес-план. Выходными компонентами этой стадии являются концептуальная модель данных, модель архитектуры системы и уточненный бизнес-план.

Определение концепции системы

Для начала давайте вспомним определение, что же такое концепция. Общее понимание концепции позволит нам точнее определить и выделить концептуальное исследование как важный класс системно значимых задач.



Ответ: Концепция — от латинского conceptio «система понимания». Иными словами концепция может быть определена как:

- **комплекс взглядов** на что-либо, связанных между собой и образующих единую систему,
- **основная точка зрения**, то есть определённый способ понимания, трактовки явлений или процессов,
- **основной замысел** целесообразный цели создания системы

Концепт это больше субъектная сторона вопроса, конкретное воплощение идеи. Концепция это больше идеологическая сторона вопроса, сама идея или целесообразную систему идей.

В озвученном смысле концепция отражает стратегическое видение системы как продукта имеющего определенную структуру и свойства, обеспечивающие удовлетворение потребности в решении конкретной проблемы её заказчика с учетом конечных ресурсов, требуемых для её проектирования и реализации.

Концепция, определяющая структуру и взаимосвязь компонентов сложного объекта называется её архитектурой («Международный центр научной и технической информации» - <http://www.icsti.su>)

Концепция велосипеда служит для стабильного преобразования энергии вращения педалей в равномерное и управляемое движение велосипедиста, что достигается за счет сочетания механики передачи усилия при вращении педалей на колеса, воздействии закона сохранения импульса на вращающиеся колеса, предотвращающего падение и органа управления, позволяющего изменять направление движения по требованию велосипедиста.

Предложенная ещё в 15 веке концепция транспортного средства в трудах Леонардо да Винчи была воплощена только в 19 веке. Но стоит ли нам изобретать велосипед?

Оказывается, с тех пор как велосипед появился его концепция лишь уточнялась, становясь отражением меняющихся требований. Но принципиально, изобретение колеса или велосипеда более не требуется с тех пор как они были изобретены.



Центральным вопросом концептуального анализа является переиспользование накопленного опыта, посредством выбора готового шаблона проектирования позволяющего построить систему с ожидаемыми свойствами

Искомый шаблон должен удовлетворять следующим критериям:

1. Решать существующую базовую проблему, являющуюся целью создания системы,
2. Соответствовать предметной области,
3. Быть стабильным и не восприимчивым к рискам,
4. Быть реализуемым,
5. Быть оптимальным.

Выбор готового шаблона проектирования можно соотнести с краткой формой концептуального исследования.

Для его выбора требуется выполнить последовательно ряд действий:

1. Извлечь и проанализировать требования, раскрывающие суть взаимного отношения системы и решаемой проблемы,
2. Изучить и описать предметную область для которой решается проблема, определить точное место проектируемой системы в предметной области,
3. Определить ключевые риски их влияние на систему для поддержания её стабильности,
4. Определить реализуемость системы с учетом существующих ограничений,
5. Выбрать оптимальное архитектурное решение.

В случае если подобный концептуальный шаблон не будет определен для системы в целом или для отдельных её элементов, то для определения концептуального проекта необходимо провести полноценное качественное концептуальное исследование.

Проведение концептуального исследования

Давайте разберемся с предпосылками концептуального исследования.

1. С позиции широкого подхода к определению системы ее рассматривают как целостное образование, обладающее свойствами, которые нельзя свести к сумме свойств элементов.
2. Наличие качественной определенности исследуемого объекта, выражающейся в том что объект подобен своей модели, по которой возможно всегда воссоздать объект.
3. Существование определенной исследовательской позиции, от которой зависит представление модели системы – точки зрения наблюдателя.

При проведении концептуального исследования предполагается, что система – это не сам объект, а лишь отражение некоторой его грани, причем той грани, которая характеризует его целостность посредством выделения в объекте совокупностей элементов и отношений между элементами.

Прежде всего, проведение качественного концептуального анализа предполагает разрешение базовой проблемы существования системы.



С позиций системного анализа под проблемой понимается ситуация, характеризующаяся различием между необходимым результатом действия системы и существующим состоянием результата или его отсутствием, если система не создана

В ходе концептуального анализа дается качественный ответ на вопрос как достичь требуемого результата при функционировании системы. Сами по себе проблемы целесообразности систем являются слабоструктурированными ввиду того, что зависят от множества факторов включая зависимости от внутренней структуры системы, отдельных её элементов или непредсказуемости внешней среды.

Таким образом, предметом концептуальных исследований являются слабоструктурированные проблемы на этапе их локализации в системе.

Решение проблемы в завершенном виде должно представлять собой такую систему, при которой различие между фактическим результатом и желаемым исчезает. В зависимости от степени исходного различия (глубины проблемы) и сложности системы, решающей проблему, характер процесса нахождения решения может быть различным.

В этом смысле полезно различать *количественные* и *качественные* проблемы. К первому типу относятся те проблемы, решение которых заключается в поиске и изменении количественных характеристик существующей системы и не требует ее существенного перестроения, то есть изменения концепции.



Решение качественной проблемы представимо за счет перестроения и изменения состава компонент системы, а также изменения их внешних связей

Существенным здесь является то, что до решения качественной проблемы ее основные компоненты не выявлены до такого уровня, при котором решение может заключаться в количественном анализе свойств существующей системы.

Эта особенность качественных проблем предполагает проведение исследований, при которых бы синтезировалась структура системы, устанавливалась бы граница системы в виде отношений со средой, выделялись бы и взаимоувязывались подсистемы, непосредственно влияющие на результат решения, формулировались бы системные задачи, то есть задачи, связанные с отношениями в системах. В отсутствие этих компонентов проблема считается нерешенной и слабоструктурированной.

Отличительной чертой подобных исследований, связанных с решением слабоструктурированных проблем, является работа с неколичественными объектами. Количества, количественный анализ приобретают смысл только после качественного сопоставления компонентов проблем и на этом этапе исследований являются вторичными. Именно этап решения качественных проблем связан с концептуальными исследованиями.



Предметом концептуальных исследований являются слабоструктурированные проблемы на этапе формирования системных представлений о компонентах этих проблем и объектов, с которыми эти проблемы связаны.

Вопросы, на которые отвечают концептуальные исследования:

- *Что нужно для того, чтобы...?*
- *Что такое...?*
- *При каких условиях...?*
- *Чем это отличается от...?*
- *Как...?*
- *Каким образом...?*
- *Какого типа...?*
- *Какие варианты...?*

Завершением концептуального исследования является исчерпание качественных вопросов и переход к количественным определениям:

- *Сколько...?*
- *На сколько...?*

Концептуальный взгляд на систему специфичен тем, что качественная определенность объектов и явлений в виде их существенных и отличительных признаков представляется в форме понятий. Поскольку понятия между собой связываются в различные содержательные отношения, то на понятиях как на элементах могут быть построены системы. Такие системы и представляют собой концептуальные модели объектов исследования.

💡 Система в виде совокупности понятий, определяющих моделируемый объект, и отношений между понятиями называется концептуальной моделью.

💡 В свою очередь любая целостная взаимосвязанная понятийная модель называется онтологией.

Построение адекватной онтологии позволяет анализировать влияние архитектуры системы на результат решения проблемы, но не сводится к её однозначному выбору.

Выбор оптимального варианта архитектуры системы определяется с использованием методов принятия решений.

Основные методы принятия решений:

- в определенных условиях — предельный и линейный методы,
- в условиях риска — матрица решений и дерево решений,
- в условиях неопределенности – критерий Вальда, Невежа, Гурвица, Лапласа.

В условиях большой вариативности факторов влияющих на архитектуру проектируемой системы анализ производится с использованием специализированных автоматизированных систем поддержки концептуального проектирования.

Основные подходы, которые легли в основу автоматизированных систем поддерживающих концептуальное проектирование:

- формализованное описание естественнонаучных и научно-технических эффектов на основе онтологии научно-технических характеристик,
- Теория решения изобретательских задач (ТРИЗ),
- Энергоинформационная модель цепей и метод структурных параметрических схем (ЭИМЦ),

- Система структурирования физических знаний и поискового конструирования.

Данные методы и подходы лежат за рамками настоящего курса и могут быть изучены отдельно. Результатом их применения можно считать построение концептуальных схем, то есть понятийных конструкций, раскрывающих качественную сторону влияния структуры системы на решаемую проблему и определение оптимальной архитектуры системы целесообразной для решения проблемы с учетом затрачиваемых ресурсов.

Концептуальные схемы достаточно автономны и могут быть использованы для построения различных моделей. Кроме того завершенная концептуальная модель может быть использована как схема в новом исследовательском процессе.

Переиспользование готовых и апробированных концептуальных схем является не чем иным, как использованием готовых шаблонов проектирования. А последовательное развертывание концептуальных моделей информационных систем обеспечивает обоснованную и контролируемую конкретизацию проекта информационных систем от расплывчатого представления о ее задачах к практической реализации.

В качестве основных техник концептуального моделирования применяется множество техник включая наиболее релевантные построению информационных систем:

- Контекстная модель (CM) – представление окружения системы.
- Модель композитной структуры (CSM) – структурное представление системы, наиболее ассоциируемое с концептуальным описанием,
- Модель потоков данных (DFM) – раскрывает структуру информационных потоков,
- Модель данных (ERM) – раскрывает объектную структуру обрабатываемых данных,
- Модель событий (EPC) – раскрывает событийную структуру причинно-следственных связей в системе,
- Модель переходов состояний (STM) – раскрывает исторический контекст обработки данных.

В качестве основного стандарта в области проектирования информационных систем выступает UML – язык унифицированного графического описания.



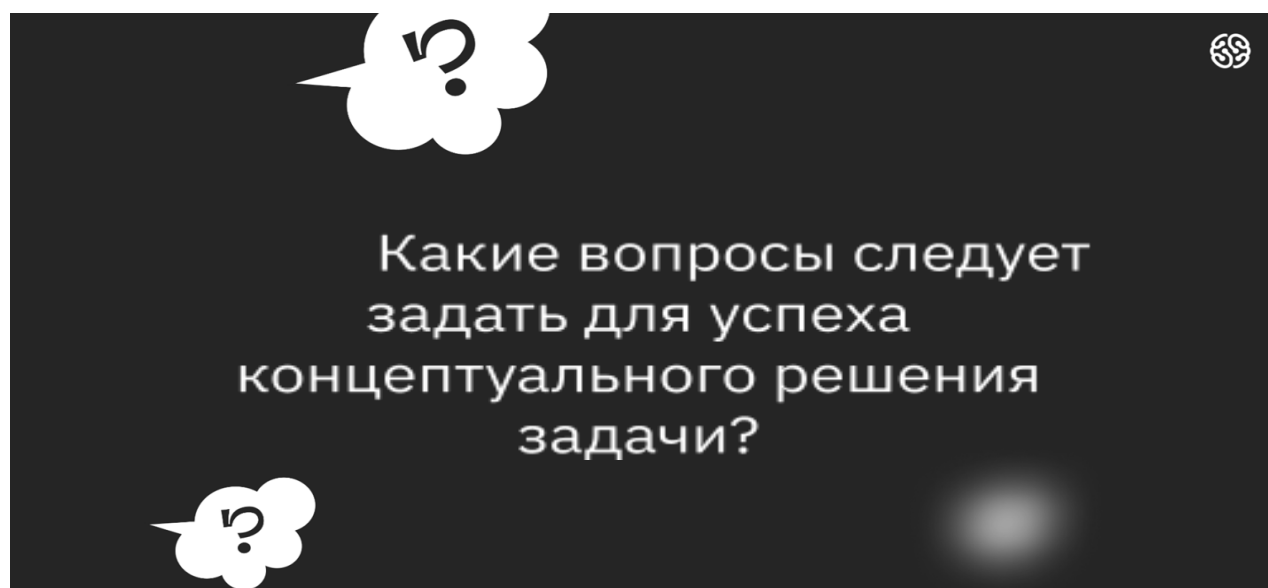
Конкретный набор моделей для построения концептуальной схемы выбирается исходя из специфики задачи.

Построение концептуальной схемы решения проблемы осуществляется как целостный процесс при условии, что этапы и средства перехода между моделями и между уровнями абстракции системы в ходе синтеза подсистем обеспечивают фиксацию общих системных свойств в соответствии с требованиями. На решение этой задачи и ориентированы методология и инструментарий концептуального синтеза информационных систем.

В качестве примера простейшего концептуального исследования рассмотрим ставшую классической задачу автоматизации службы приема запросов.

К примеру, существует организация, которой требуется снизить издержки горячей линии — службы приема и обработки запросов. Эти излишние издержки возникают из-за относительно низкого уровня качества обслуживания клиентов. Персонал горячей линии не обладает полнотой знаний и опытом для того чтобы быстро и безошибочно классифицировать обращения клиентов. В результате работы клиентской службы часть обращений получают необоснованный отказ, а часть обращений направляется в нецелевые подразделения, не имеющие отношения к данным обращениям.

Эта вполне обычная ситуация имеет решение. Но для его поиска необходимо сперва задать несколько фундаментальных вопросов. Придумайте их на основе лекционного материала.



Ответ:

- *Что нужно для того, чтобы операторы не ошибались?* Сразу приходит на ум два варианта: либо убрать операторов, либо обучить их. В первом случае придется автоматизировать процесс приема обращений, что обычно делается за счёт выделения нескольких уровней обслуживания. И первый уровень на котором производится классификация запросов, хорошо поддается автоматизации, но дальнейшие уровни требуют качественного перенаправления запроса со стороны службы горячей линии.
- *Что такое верная качественное решение запроса?* Со стороны службы горячей линии качественно решенным считается запрос на который был дан быстрый ответ либо направленный в правильное (целевое) подразделение поддержки.
- *При каких условиях запрос разрешается специалистами горячей линии?* Если он является типовым, имеющим стандартное решение, и при этом запрос не является повторным и массовым.
- *Чем такой запрос отличается от не типового запроса?* Нетиповые запросы не имеют однозначных исторических параллелей с успешно решенными запросами. Обычно такие запросы могут содержать уникальные случаи, а также недостаток информации или её избыток, что затрудняет классификацию.
- *Как специалист горячей линии принимает решение о том что запрос типовой?* Специалист по своему опыту принимает решение, в случае необходимости проводит интервьюирование.
- *Каким образом этот процесс может быть автоматизирован?* Процесс может быть автоматизирован за счет обучения алгоритма на успешных примерах классификации. Но живой речи достаточно высокая вариативность использования слов, поэтому применение алгоритмов рассчитанных, или как говорят “заточенных”, на обработку условий представляется затруднительным.
- *Какого типа алгоритмы стоит применять в данной задаче?* Для решения задачи лучше всего подходят алгоритмы машинного обучения использующие предварительно натренированные на положительном и отрицательном опыте модели.
- *Какие варианты подходят для решения данной задачи?* Для решения нашей задачи возможно использовать модель BERT (Bidirectional Encoder Representations from Transformer) — это языковая модель, в основе которой лежит мультислойный двунаправленный кодирующий Трансформер.

Трансформер получает на вход последовательность фонов и после внутренних преобразований выдает закодированное представление этой последовательности, который далее используется для оценки соответствия входящего запроса одному из типовых запросов имеющих стандартное решение или маршрут перенаправления.

Следующим этапом, обуславливающим завершение концептуального исследования и начало концептуального проектирования является этап уточнения требований. Этот этап является важнейшим этапом концептуального моделирования.

Уточнение и учет требований

Давайте поговорим о первом шаге концептуального проектирования, а именно об уточнении и учете требований к системе.

Требования к системе могут быть обусловлены предметной областью, включающей существующие нормы и правила, так и требованиями заинтересованных сторон проекта.

Учет требований к создаваемой или развиваемой системе является важным в силу того, что ошибки в требованиях допущенные на начальном этапе сказываются самым большим итогом на реализации при их исправлении.

Задача этого этапа определить минимальный набор требований в максимальной степени удовлетворяющих потребителя системы в лице заказчика.

Учет требований предметной области состоит в процессе её познания, но по большей части он статичен и хорошо изучен в сфере ИТ-технологий для понятийного домена бизнес среды.

Учет требований заинтересованных сторон представляет намного больший интерес при концептуальном проектировании информационных систем. Это наиболее подвижное и неопределенное поле открывающее возможности работы с требованиями системному аналитику, который занимая активную позицию может открыто дискутировать с заинтересованными сторонами обсуждая реализуемость

требований с учетом критерия конечной целостности и качества проектируемой системы.

Изначально требования фиксируются в виде пользовательских историй (US – User Story) с максимально простым описанием:

- Идентификатор истории
- Название
- Роль использования
- Описание использования
- Предварительные условия
- Результат использования

Например: для меня как для Пользователя сайта важно видеть информацию о названии организации при нахождении на странице официального сайта организации.



На этапе концептуального проектирования крайне важно не упустить ключевые требования воплощенные в пользовательские истории.

Итог концептуального проектирования в части учета и уточнения требований разумно представить в виде актуальной карты пользовательских историй (USM - User Story Map). Этот ныне популярный в продуктовой разработке подход был предложен Джефом Паттоном, как способ визуализации пользовательских историй. Графически, это представление ролей пользователей и действий, которые те совершают с ИТ-продуктом. Если мы делаем приложение интернет-магазина, то в USM владелец продукта описывает то, как клиент будет его использовать.

Для чего это нужно?

- Для проектирования пользовательского опыта в продукте,
- Для определения границ MVP (минимальной работоспособной версии продукта) и планирования релизов на базе пользовательского сценария,
- Но прежде всего для формирования единого понимания назначения системы у команды разработки и заинтересованных лиц

Как это работает?

Для создания карты пользовательских историй все пользовательские истории группируются по элементам создаваемой системы, после чего приоритизируются по

важности реализации. Распространенный и рассматриваемый нами метод приоритизации MoSCoW получил свое название от акронима, образованного следующими классификациями приоритета: Must have (Должен иметь), Should have (Следовало бы иметь), Could have (Может иметь) и Would like (Хотелось бы иметь). Буква «о» делает акроним произносимым.

- Must have (обязательные истории, обозначаемые зеленым) - это истории, которые должны быть поставлены для решения текущих проблем бизнеса. В данном случае мы рассматриваем истории с приоритетом Must have как критически необходимые для создания системы на уровне MVP. Must have понимается как минимальное подмножество историй для начала использования системы.
- Should have (требуемые истории, оранжевым) столь же важны, как и истории Must have, но они могут не быть срочным или для их реализации можно использовать обходной путь (workaround).
- Could have (возможные истории, синим) – это менее критичные истории. Можно сделать после Should have, если будет возможность
- Would like (желательные истории, серым) – можно сделать после Could have или обойтись без этих историй. Это своего рода "Бантики", которые пользователь хотел бы получить в дополнение к основному функционалу системы.

Исходя из актуализированной и приоритизированной карты пользовательских историй определяется круг требований входящих в границы проекта на начальном и дальнейших этапах, что качественно, то есть концептуально, влияет на архитектуру и состав проектируемой системы.

В дальнейшем при функциональном проектировании карта пользовательских историй уточняется с построением схемы вариантов использования (UC - Use Case) и карте пути клиента (CJM – Customer Journey Map).

Семантическое моделирование

Предметная область — раздел науки, изучающий предметные аспекты системных процессов и системные аспекты предметных процессов и явлений. Это определение можно считать системным определением предметной области.

Для концептуального проектирования системы крайне важно определить её роль и место в существующем контексте предметной области, наполненном информацией о различных процессах и явлениях.

При построении концепции системы в информационном смысле важно определить её точное место таким образом, чтобы минимизировать энтропию, то есть вариативность, используемой информации в модели системы, при этом предоставить системе полноту данных для алгоритмических инструментов позволяющих решить проблему.



Суть семантического моделирования представляет собой понимание смысла предметной области, а в практическом плане представляет инструментарий для выявления структурных закономерностей предметной области для органичного встраивания в неё проектируемой системы.

Ключевыми понятиями семантического моделирования предметной области являются понятия сущности и связи, а также информационных потоков. Эти понятия являются важными для анализа отношений как внутри системы, так и вне её. Семантическое моделирование ведется с использованием представлений отражающих контекстное представление системы и структуру связанных сущностей, фиксируемых в виде графических диаграмм.

Контекстная диаграмма системы (SCD) в инженерии — это диаграмма, которая определяет границу между системой или частью системы и ее средой, показывая объекты, которые взаимодействуют с ней. Эта диаграмма представляет собой высокоуровневое представление системы.

Контекстная диаграмма — это первое, что вы создаете при работе с системой.

Как составляются контекстные диаграммы?

На диаграмме выделяется система как единое целое – центральный элемент. Пользователи и другие внешние системы записываются по кругу.

Добавляются информационные связи между проектируемой системой, пользователями и внешними системами.



Важно понимать и отразить в качестве комментария контекст в котором происходит взаимодействие.

Следующий инструмент концептуального проектирования является абсолютно значимым для системы, собственно любая информационная система должна быть понятийно определена. Любое понятие обозначает понятийную категорию, именуемую сущностью. То есть другими словами сущность (Entity) — реальный либо воображаемый объект, имеющий существенное значение для рассматриваемой предметной области.

Наиболее распространенным средством сематического моделирования является модель «сущность-связь» (ERM). Она была впервые введена Питером Ченом в 1976 году, а затем развита в работах других авторов и воплощена в стандартах UML. Изначально речь шла про моделирование данных хранимых в базах данных и используемых системами, но затем понятие было расширено до основного инструмента семантического анализа, имеющего необходимую мощность для моделирования структур со сложными связями.

Базовыми понятиями ERM являются сущность, связь и атрибут.



Каждая сущность должна иметь наименование, выраженное существительным в единственном числе. Примерами сущностей могут быть такие понятийные классы – отражения группы однотипных объектов имеющих общие черты, как например «Сайт», «Страница», «Заказ», «Клиент». При графическом представлении каждая сущность в модели изображается в виде прямоугольника с наименованием.

Основной (неформальный) способ идентификации сущностей — это поиск абстракций, описывающих физические или материальные объекты, процессы и явления, роли людей, организации и другие понятия. Единственным формальным способом идентификации сущностей является анализ текстовых описаний предметной области, выделение из описаний имен существительных и выбор их в качестве «кандидатов» на роль моделируемых абстракций.

Атрибутом выступает любая значимая и определенная характеристика сущности, которая указывается в том же квадрате под наименованием сущности с указанием их типа. Например, как атрибуты простых (скалярных) типов «Цена», «Фамилия» и «Вес», так и составные комплексные атрибуты, например «Паспортные данные» или «Содержание заказа».

Уникальным идентификатором называется не избыточный набор атрибутов, значения которых в совокупности являются уникальными для каждого экземпляра сущности.

Связь — это некоторая ассоциация между двумя сущностями. Одна сущность может быть связана с другой сущностью или сама с собою. Связи позволяют по одной сущности находить другие сущности, связанные с ней.

💡 Каждая сущность может обладать любым количеством связей с другими сущностями модели

💡 Степенью связи называется количество сущностей, участвующих в связи. Связь степени 2 называется бинарной, степени N—N-арной. Связь, в которой одна и та же сущность участвует в разных ролях, называется рекурсивной, или унарной.

В стандарте UML, как наиболее подходящем для семантического моделирования, дополнительно определены четыре типа связей:

- Зависимость
- Ассоциация
- Обобщение
- Реализация

Эти связи представляют собой базовые строительные блоки для описания отношений в UML, используемые для разработки хорошо согласованных моделей. Первая из них – зависимость – семантически представляет собой связь между двумя элементами модели, в которой изменение одного элемента (независимого) может привести к изменению семантики другого элемента (зависимого). Графически представлена пунктирной линией, иногда со стрелкой, направленной к той сущности, от которой зависит еще одна; может быть снабжена меткой.

Зависимость — — — — — ➔

Зависимость – это связь использования, указывающая, что изменение спецификаций одной сущности может повлиять на другие сущности, которые используют ее.

Ассоциация —————➔

Ассоциация – это структурная связь между элементами модели, которая описывает набор связей, существующих между объектами. Ассоциация показывает, что объекты одной сущности (класса) связаны с объектами другой сущности таким образом, что можно перемещаться от объектов одного класса к другому.

Например, класс Человек и класс Школа имеют ассоциацию, так как человек может учиться в школе. Ассоциации можно присвоить имя «учится в». В представлении однонаправленной ассоциации добавляется стрелка, указывающая на направление ассоциации.

Двойные ассоциации представляются линией без стрелок на концах, соединяющей два классовых блока.

Ассоциация может быть именованной, и тогда на концах представляющей её линии будут подписаны роли, принадлежности, индикаторы, мультипликаторы, видимости или другие свойства.

Множественность ассоциации представляет собой диапазон целых чисел, указывающий возможное количество связанных объектов. Он записывается в виде выражения с минимальным и максимальным значением; для их разделения используются две точки.




Устанавливая множественность дальнего конца ассоциации, вы указываете, сколько объектов может существовать на дальнем конце ассоциации для каждого объекта класса, находящегося на ближнем ее конце. Количество объектов должно находиться в пределах заданного диапазона. Множественность может быть определена как единица 1, ноль или один 0..1, любое значение 0..* или *, один или несколько 1..*. Можно также задавать диапазон целых значений, например 2..5, или устанавливать точное число, например 3.

Агрегация

Агрегация – особая разновидность ассоциации, представляющая структурную связь целого с его частями. Как тип ассоциации, агрегация может быть именованной. Одно отношение агрегации не может включать более двух классов (контейнер и содержимое).

Агрегация встречается, когда один класс является коллекцией или контейнером других. Причём, по умолчанию агрегацией называют агрегацию по ссылке, то есть когда время существования содержащихся классов не зависит от времени существования содержащего их класса. Если контейнер будет уничтожен, то его содержимое — нет.


Графически агрегация представляется пустым ромбом на блоке класса «целое», и линией, идущей от этого ромба к классу «часть».

Композиция 


Композиция — более строгий вариант агрегации. Известна также как агрегация по значению.

Композиция — это форма агрегации с четко выраженными отношениями владения и совпадением времени жизни частей и целого. Композиция имеет жёсткую зависимость времени существования экземпляров класса контейнера и экземпляров содержащихся классов. Если контейнер будет уничтожен, то всё его содержимое будет также уничтожено.

Графически представляется, как и агрегация, но с закрашенным ромбом.

Обобщение 

Третья связь — обобщение — выражает специализацию или наследование, в котором специализированный элемент (потомок) строится по спецификациям обобщенного элемента (родителя). Потомок разделяет структуру и поведение родителя. Графически обобщение представлено в виде сплошной линии с пустой стрелкой, указывающей на родителя.

Реализация 

Четвертая — реализация — это семантическая связь между классами, когда один из них (поставщик) определяет соглашение, которого второй (клиент) обязан придерживаться. Это связи между интерфейсами и классами, которые реализуют эти интерфейсы. Это, своего рода, отношение «целое-часть». Поставщик, как правило, представлен абстрактным классом. В графическом исполнении связь реализации — это гибрид связей обобщения и зависимости: треугольник указывает на поставщика, а второй конец пунктирной линии — на клиента.



Крайне важно определить позицию наблюдателя и выбрать степень абстракции описания модели системы таким образом, чтобы семантическая часть модели была наиболее содержательной и наглядной.

Апробация концепции

После того как проработаны вопросы связанные с требованиями к системе и проанализирована предметная область в которой проектируется система, на повестку концептуального проектирования выходят вопросы обеспечения реализуемости, стабильности и оптимальности концепции системы.

При проведении полного концептуального исследования возникает возможность всестороннего рассмотрения исследуемых концепций. Но полноценное концептуальное исследование требует определенной подготовки, так как является строго научной деятельностью и значительных трудозатрат, что целесообразно при проектировании принципиально нового решения. Даже если речь идет о новом взгляде на привычную концепцию велосипеда.



При проектировании информационных систем связанных со сложными человеко-машинными взаимодействиями зачастую вместо полноценного концептуального исследования целесообразно провести краткий SWOT анализ, раскрывающий основные риски и преимущества концепции, а затем, если требуется ранжировать и сравнить проектируемые концепции или варианты реализации.

SWOT-анализ – это метод комплексной оценки факторов, напрямую и косвенно влияющих на бизнес результат.

SWOT расшифровывается так:

Strengths – сильные стороны,

Weakness – слабые стороны,

Opportunities – возможности (внешней среды),

Threats – угрозы (внешней среды).

	Сильные	Слабые
Угрозы	Фактор 1, действия и решения Фактор 2, действия и решения	Фактор 1, действия и решения Фактор 2, действия и решения
Возможности	Фактор 1, действия и решения Фактор 2, действия и решения	Фактор 1, действия и решения Фактор 2, действия и решения

Далее концепции дорабатываются, после чего сравниваются параметрически. Сильные стороны одной концепции переносятся в слабые стороны другой концепции и наоборот.

Факторы	Вес	Концепция 1	Концепция 2
Угроза 1	0.00 – 1.00	Оценка 0-10	Оценка 1-10
Угроза 1	0.00 – 1.00	Оценка 0-10	Оценка 1-10
Возможность 2	0.00 – 1.00	Оценка 0-10	Оценка 1-10
Возможность 2	0.00 – 1.00	Оценка 0-10	Оценка 1-10
	$\Sigma = 1$	$\Sigma = B1*O3 + B2*O4 - (Y1*O1 + Y2*O2)$	$\Sigma = B1*O3 + B2*O4 - (Y1*O1 + Y2*O2)$
Итоги		0-10	0-10

Выбирается концепция с наибольшей итоговой суммой результата.



Как известно, практика лучший критерий истины

Подтверждение результата оценки осуществляется посредством создания действующего прототипа – MVP версии продукта, смысл реализации которой подтвердить или опровергнуть результаты концептуального проектирования.

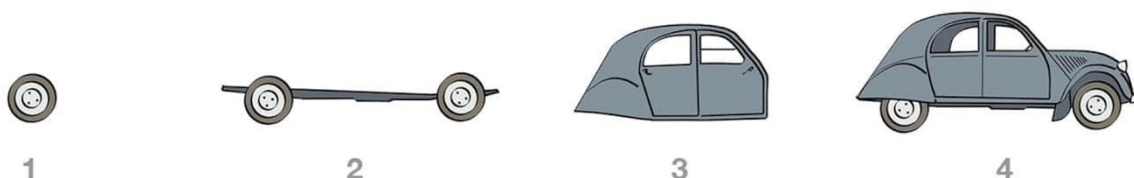
Важно отметить что системе должна быть присуща концептуальная, то есть качественная целостность. Другими словами, с первой итерации разработки

ограниченного решения центральной проблемы система должна быть сообразна её полному концептуальному воплощению.

Для иллюстрации можно обратиться к гипотетическому процессу создания автомобиля.

Например, при каскадном покомпонентном изготовлении мы получим вместо прототипа готовое изделие на последнем этапе, соответственно данный подход не может быть назван целесообразным с учетом апробации концепции проекта. Сначала будет спроектировано и изготовлено колесо, затем границы системы будет расширены, в неё войдет рама, затем возможно, фокус сместится на кузов, а в заключении мы столкнемся со сборкой автомобиля из разрозненных компонент. И если даже кузов подойдет к шасси, совершенно не очевидно что автомобиль понравится заказчику, который ни разу его ни видел. Тем менее очевидным то, что данный автомобиль решит ключевую проблему потребителя наилучшим образом.

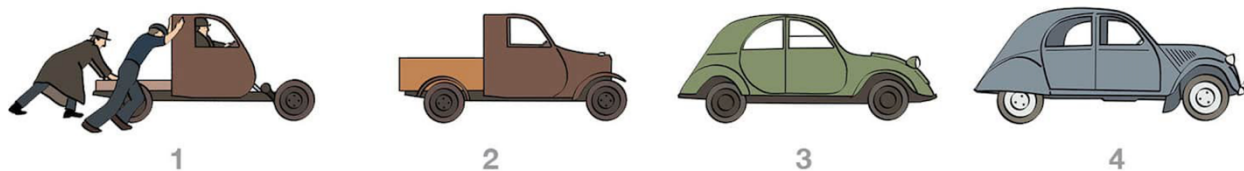
Опыт реализации многих проектов показывает существенный риск изменения фокуса концепции при реализации системы, что отражается в количестве неуспешных проектов, конечный продукт которых не соответствовал требованиям заказчика и не в должной мере решал ключевую проблему потребителя системы.



Следующий отрицательный пример относится к полному изменению концепции по мере реализации системы, подобный подход так же не сможет способствовать апробации концепции проекта. Так на концепцию велосипеда воздействуют одни факторы внешней среды, а на автомобиль совершенно другие. Но и внутренняя структура и отношение к потребителю у этих систем различно.



По настоящему концепция MVP воплощается тогда и только тогда, когда первичная версия продукта являющегося физическим воплощением концептуальной модели системы сообразно желаемому результату, наглядно и очевидно для всех участников процесса без лишних пояснений.



В ходе апробации системы попутно решается вопрос её реализуемости или осуществимости. Чем точнее мы измеряем затраченные ресурсы на изготовление MVP, тем ценнее становится результат апробации концепции системы для ответа на вопрос о её реализуемости.

Встреченные препятствия на пути апробации позволяют с минимальными потерями модифицировать концепцию системы, либо поняв что система в соответствии с принятой концепцией не реализуема с учетом ресурсов и требований заказчика, полностью поменять концепцию.

Таким образом мы с вами рассмотрели методику как обоснованно и надежно спроектировать концепцию системы решающую ключевую проблему её потребителя. Подготовленная концепция воплощается в концептуальных артефактах стратегии проекта, включая устав проекта, заявление о концепции, философию проекта, дорожные карты проекта, иерархические диаграммы и прочие документы раскрывающие концептуальную архитектуру создаваемой или изменяемой системы.

Затем на поздних этапах проектирования разработанная концепция детализируется функционально и технологически вплоть до всех значимых нюансов физического воплощения. Переработка концепции требуется лишь в случае обнаружения принципиального разночтения с требованиями заказчика или по причине возникновения новой технологической возможности для совершенствования системы.

Что можно почитать еще?

1. Для более глубокого изучения концептуального проектирования сложных инновационных систем [Концептуальное проектирование: «Теория изобретательства»](#)
2. Для развития кругозора по методам работы с пользовательскими историями [«User story mapping: как посмотреть на проект глазами пользователя»](#)
3. А также [Инструменты Impact Mapping \(IM\) и Story Mapping \(SM\) применяются в сфере планирования, разработки и внедрения проектов](#)
4. Узнать больше про то, каким образом возможно изобрести велосипед [«Development of a Student Bicycle»](#)
5. Изучить UML на официальном сайте <http://uml.org>

Используемая литература

1. Д. Р. Трутнев, «Архитектуры информационных систем основы проектирования», учебное пособие
2. А. А. Голованов, «Проектирование информационных систем», учебное пособие
3. И.А. Спицина, К.А. Аксенов, «Системный анализ и моделирование информационных систем»
4. Новиков Ф.А., «Учебно-методическое пособие по дисциплине «Анализ и проектирование на UML»