

Лекция 1. Основы проектирования ПС

Основные понятия методологии проектирования

1. Цели и содержание методологии проектирования ПС
2. Моделирование бизнес-процессов
3. Сравнение методик (функциональный и объектный подходы)

Методология проектирования

В реальных условиях проектирование — это поиск способа, который удовлетворяет требованиям функциональности системы средствами имеющихся технологий с учетом заданных ограничений.


Метод проектирования:

- организованная совокупность процессов создания ряда моделей, которые описывают различные аспекты создаваемой системы с использованием четко определенной нотации.

Технология проектирования:

- совокупность технологических операций в их последовательности и взаимосвязи, приводящая к разработке проекта системы.

Области проектирования



проектирование объектов данных, которые будут реализованы в базе данных;

проектирование программ, экранных форм, отчетов, которые будут обеспечивать выполнение запросов к данным;

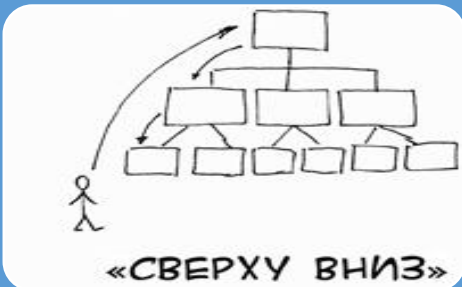
учет конкретной среды или технологии, а именно: топологии сети, конфигурации аппаратных средств, используемой архитектуры (файл-сервер или клиент-сервер), параллельной обработки, распределенной обработки данных и т.п.

Этапы развития технологий проектирования



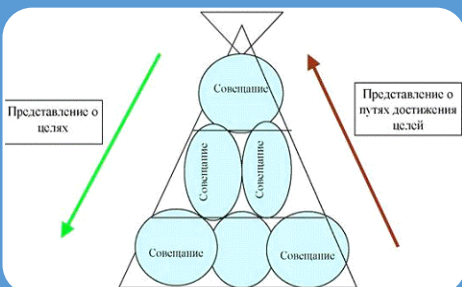
Метод "снизу-вверх"

- НЕ создание тиражируемых продуктов, а обслуживание сотрудников конкретного учреждения.
- Успешно автоматизируются отдельные, важные с точки зрения руководства рабочие места.
- Общая же картина "автоматизированного предприятия" просматривается недостаточно хорошо, особенно в перспективе («Лоскутная автоматизация»).



Метод "сверху-вниз"

- ОДНА программа должна удовлетворять потребности всех пользователей:
- резко ограничены возможности разработчиков в структуре информационных множеств базы данных, использовании вариантов экранных форм, алгоритмов расчета и, следовательно, НЕТ возможности по ведению глубокого, часто специфического аналитического и производственно-технологического учета.



Метод многокомпонентности

- адаптация подсистемы ПО к принятым в организации условиям работы.
- Проведение модернизации одного из компонентов не затрагивает центральную часть (ядро) и другие ее компоненты, что значительно повышает надежность, продолжительность жизни автоматизированной системы и обеспечивает наиболее полное выполнение требуемых функций.

Цель проекта

Цель проекта
можно
определить как
решение ряда
взаимосвязанных
задач,
включающих в
себя:

требуемой функциональности системы и уровня ее
адаптивности к изменяющимся условиям функционирования;

требуемой пропускной способности системы;

требуемого времени реакции системы на запрос;

безотказной работы системы;

необходимого уровня безопасности;

простоты эксплуатации и поддержки системы.

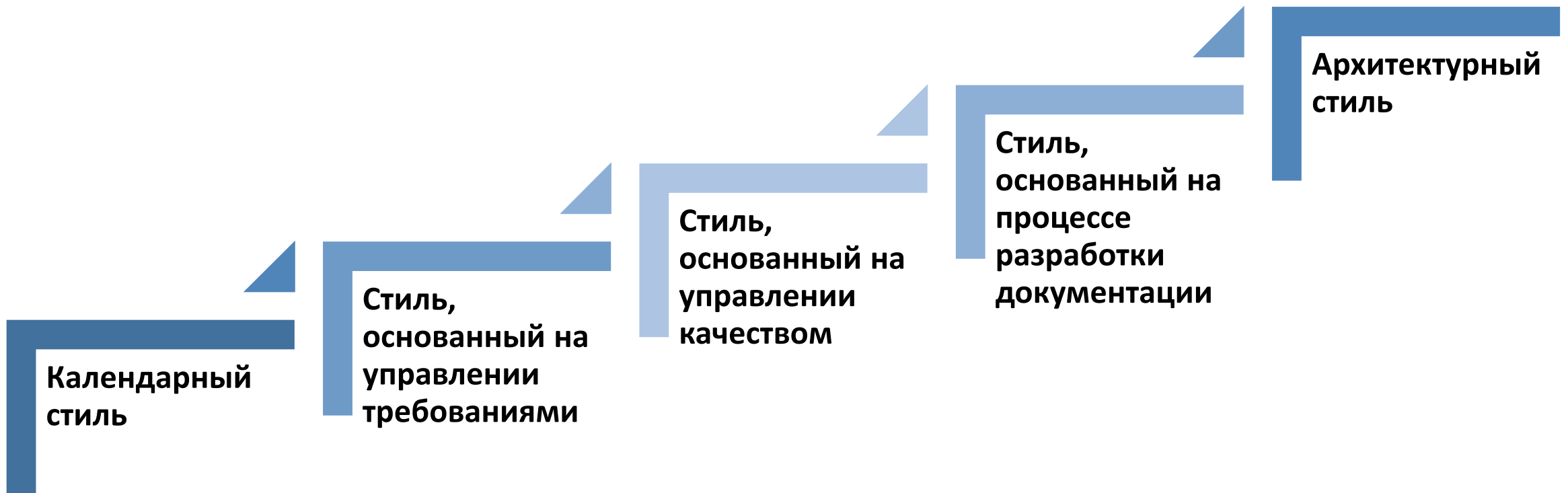
Характеристики идеальной системы



РЕАЛЬНЫЕ ФАКТЫ

- Ни одна сложная система не идеальна
- Архитектура системы является искусством компромисса – решение, усиливающее одну из этих характеристик, часто ослабляет другие

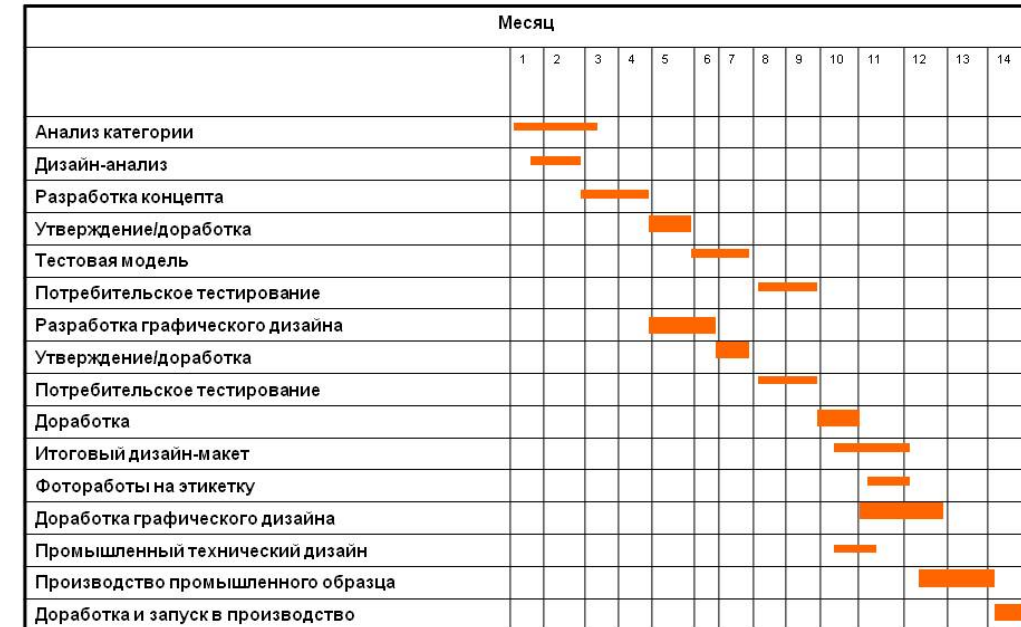
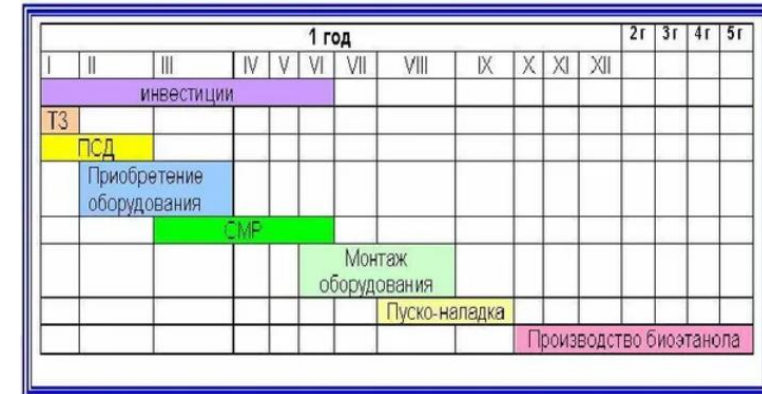
5 подходов к проектированию



Календарный стиль

- Calendar-driven
- Основан на графике работ
- Проектные решения принимаются из целей и задач конкретного этапа
- Мало внимания уделяется процессу разработки, документации, созданию стабильных архитектур и внесению изменений
- Высокая суммарная стоимость владения в долгосрочной перспективе

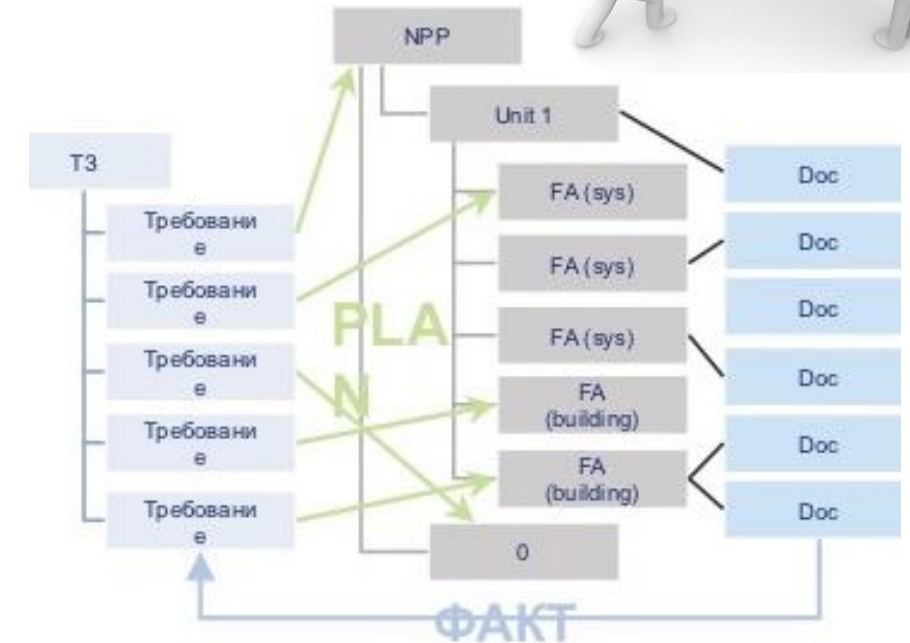
Календарный план проекта



Управление требованиями



- Requirements-driven
- Основное внимание уделяется функциональным характеристикам системы
- Проектные решения принимаются исходя из локальных целей, связанных с реализацией определенных функций
- Эффективен при устойчивых требованиях
- Недостаточное внимание требованиям стандарта ISO 9126 (управление качеством)
- Нестабильность разрабатываемых архитектур



Процесс разработки документации

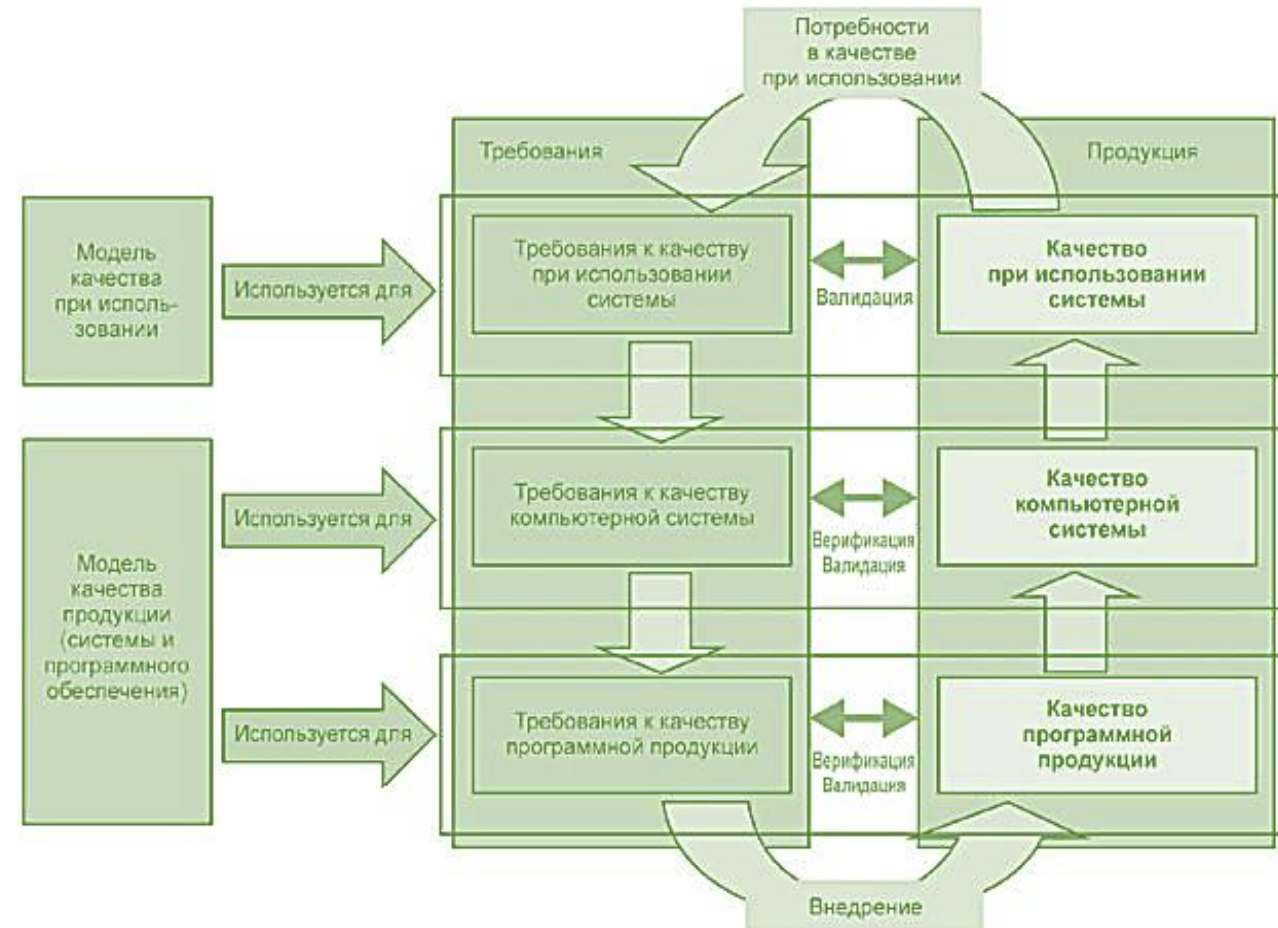
- Documentation-driven
- Вырожденный вариант стиля, основанного на управлении качеством
- Ориентирован на разработку документации
- Применяется в государственных и крупных компаниях
- Большие затраты времени и сил на разработку документации в ущерб качеству кода
- Создаваемая документация часто не используется ни пользователем, ни заказчиком



Управление качеством

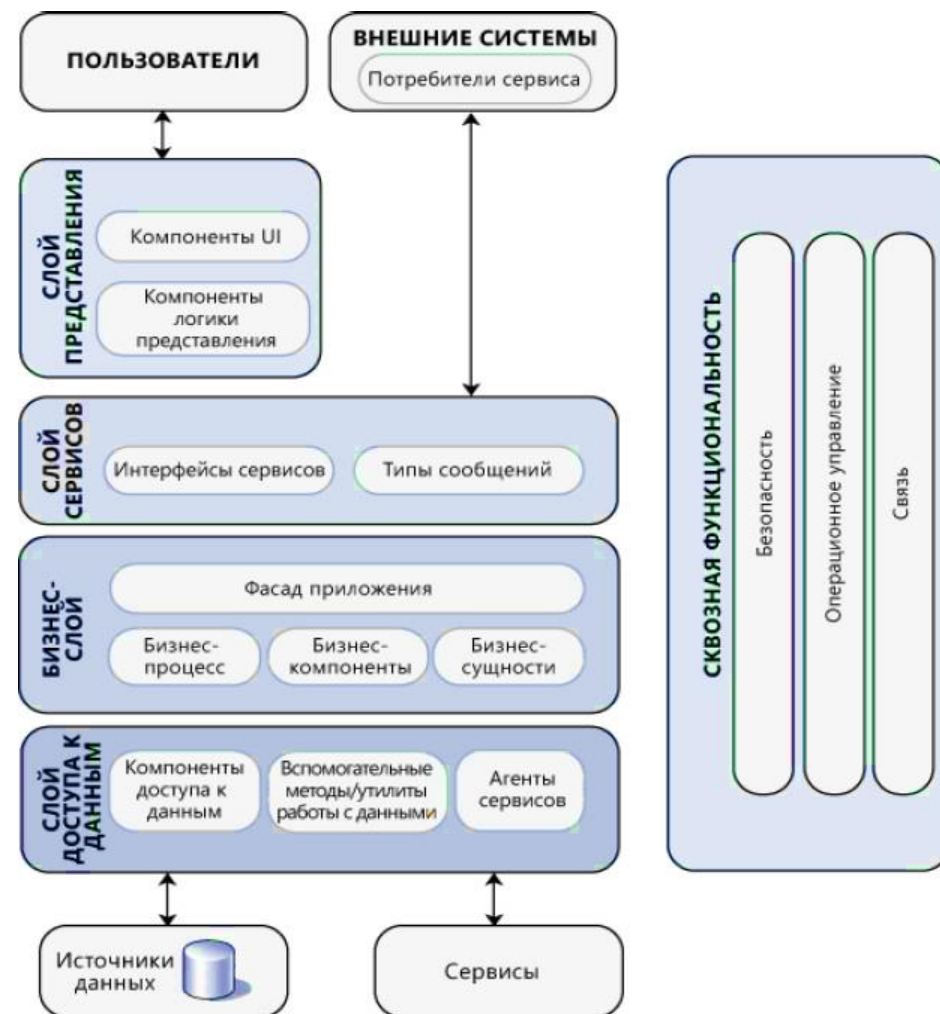


- Quality-driven
- Консервативный стиль
- Для разработки систем с экстремальными характеристиками
- Определенные качества системы оптимизируются в ущерб другим
- Нестабильность архитектурных решений
- Очень трудно вносить изменения



Архитектурный стиль

- Architecture-driven
- Создание фреймворков, легко адаптируемых ко всем потенциальным требованиям
- Проектирование включает две задачи: создания фреймворка и создание конкретной системы
- Устраняет недостатки стиля, основанного на управлении требованиями
- Обеспечивает оперативное изменение существующей и добавление новой функциональности



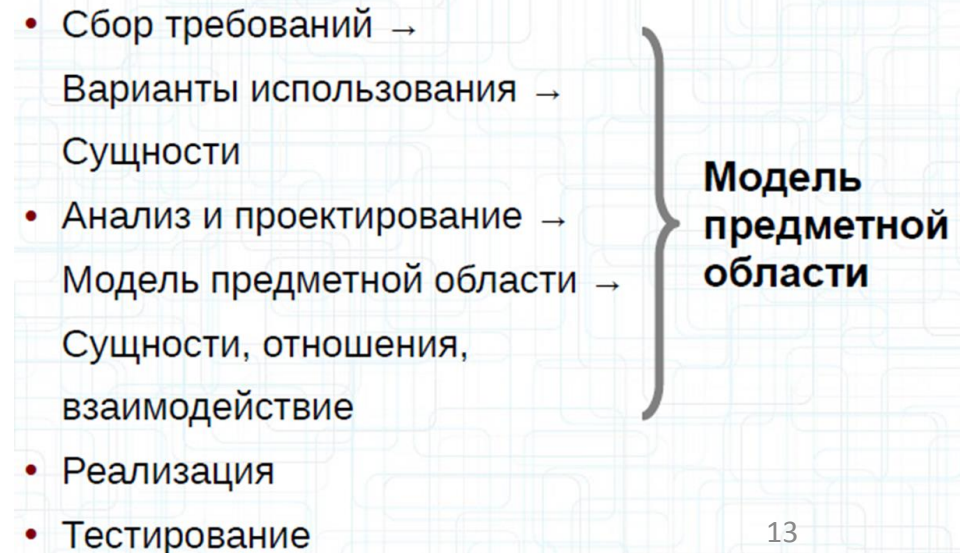
Процесс разработки

Процесс разработки представляет собой процесс построения и последовательного преобразования ряда согласованных моделей на всех этапах жизненного цикла (ЖЦ) системы.

На каждом этапе ЖЦ создаются специфичные для него модели – *деятельности организации, требований, проекта, требований к приложениям и т.д.*

Модели формируются рабочими группами команды проекта, сохраняются и накапливаются в проекте.

Создание моделей, их контроль, преобразование осуществляется с использованием специальных программных инструментов - CASE-средств.



Жизненный цикл ИС

Стратегия проектирования ИС

- определяется использованием соответствующей модели жизненного цикла, определяющей последовательность стадий проектирования и выполняемых в них процессов.

Жизненный цикл ИС

- ряд событий, происходящих с системой в процессе ее создания и использования.

Модель жизненного цикла

- структура, содержащая стадии, процессы (действия и задачи), которые осуществляются в ходе разработки, функционирования и сопровождения программного продукта в течение всей жизни системы, от определения требований до завершения ее использования.

Стандарты, регламентирующие ЖЦ ПО

ГОСТ 34.601-90 Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания.

ISO/IEC 12207:1995 Information technology - Software life cycle processes (Информационные технологии. Процессы жизненного цикла программного обеспечения)

ISO/IEC 15288 Systems engineering. System life cycle processes (Системотехника. Процессы жизненного цикла системы)

ISO - International Organization of Standardization - Международная организация по стандартизации, IEC - International Electrotechnical Commission - Международная комиссия по электротехнике

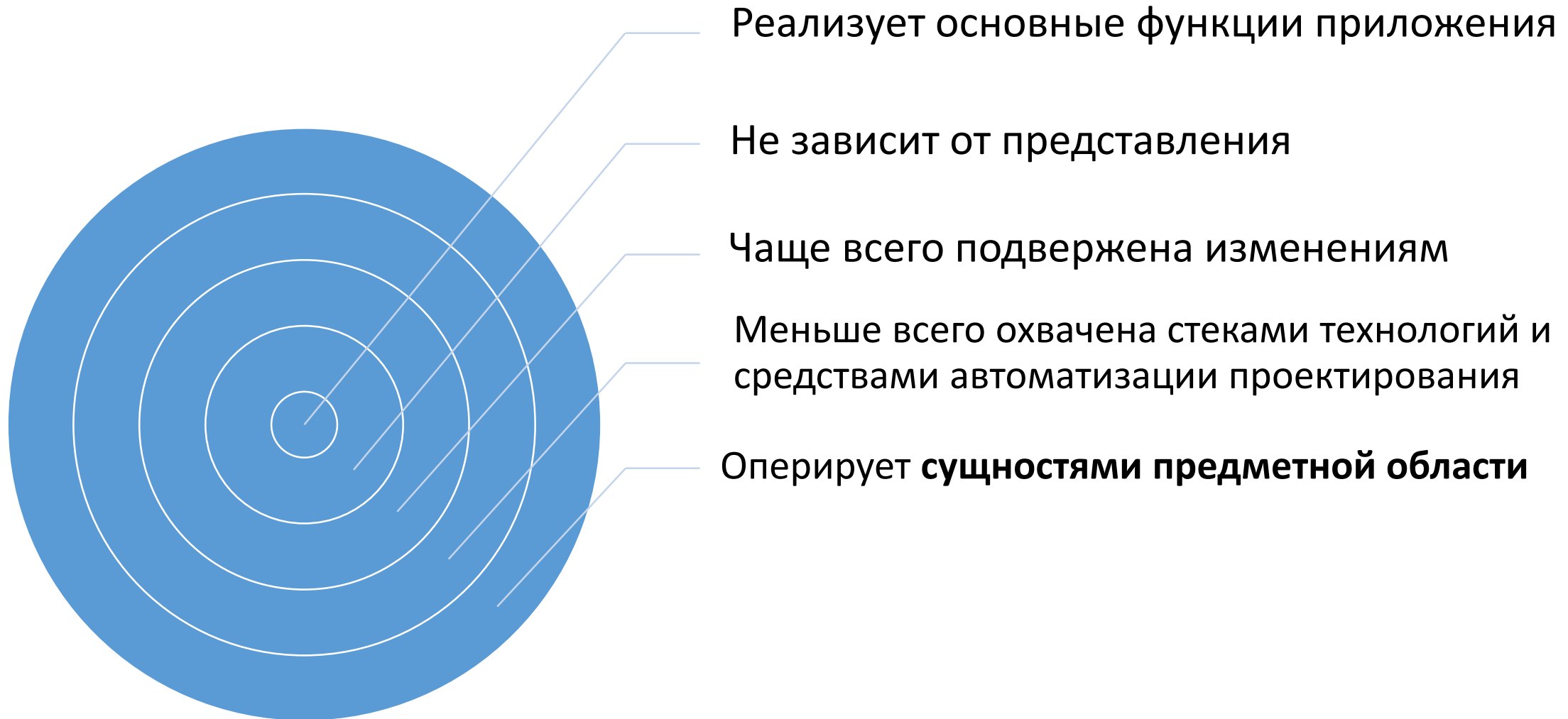
Custom Development Method (методика Oracle) по разработке прикладных информационных систем

Rational Unified Process (RUP)

Microsoft Solution Framework (MSF)

**Корпоративные
методики**

Бизнес-логика



Модель предметной области

Модель

- строго организованная выборка знаний предметной области и способ их структуризации

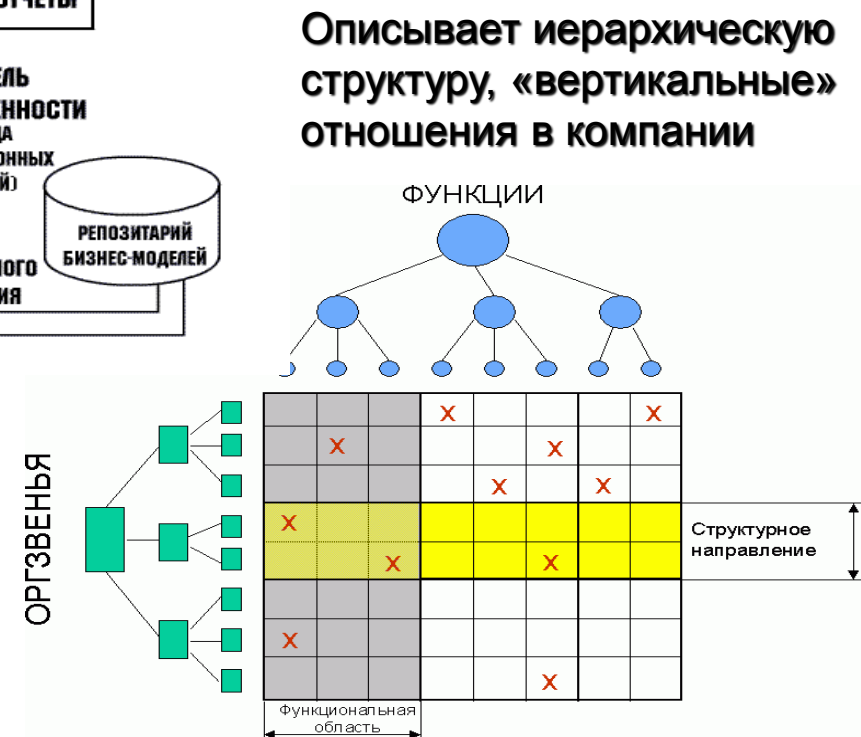
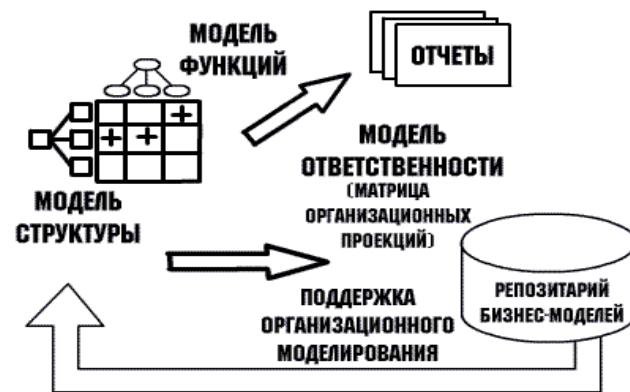
UML

- один из языков описания модели.



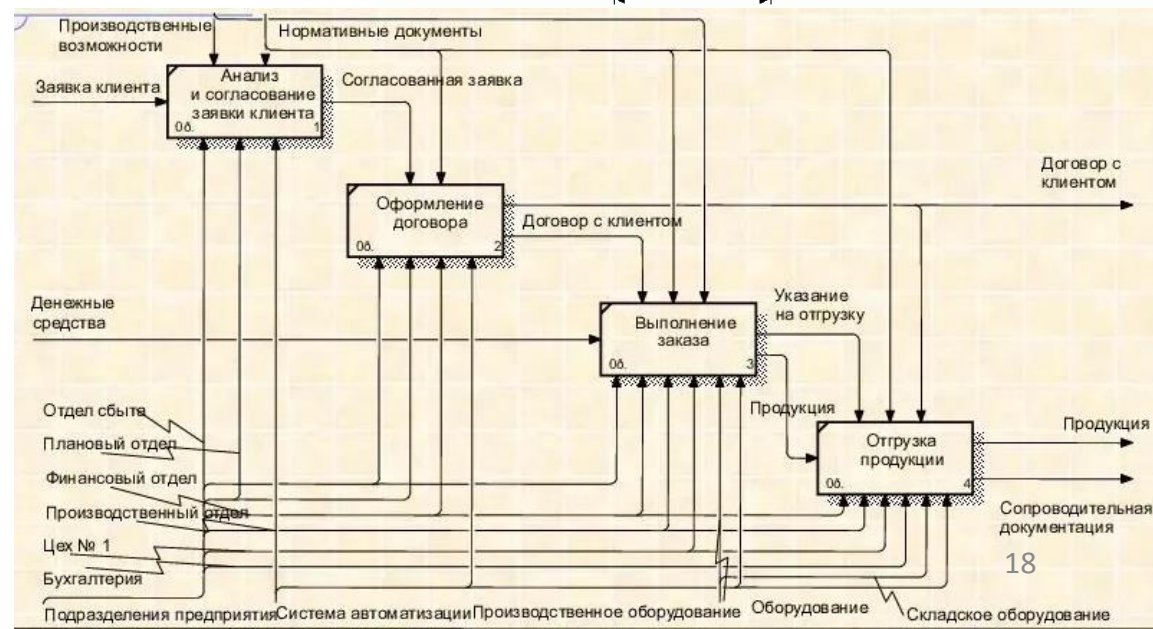
Виды моделей

- Организационно-функциональная модель компании (описывает распределение функций и задач между подразделениями, сферы ответственности за реализацию бизнес-стратегии, организацию документооборота)



- Бизнес-процессная модель компании (описывает выполнение бизнес-процессов, информационные входы-выходы операций, взаимодействие между подразделениями и исполнителями)

Описывает «горизонтальные» отношения



Основные подходы к разработке моделей

Структурный анализ

- метод исследования системы, которое начинается с общего обзора и затем детализируется, приобретая иерархическую структуру со все большим числом уровней.

Объектно-ориентированное моделирование

- подразумевает описание статической структуры системы в терминах объектов и связей между ними, а поведение системы описывается в терминах обмена сообщениями между объектами. Каждый объект обладает своим собственным поведением, моделирующим поведение объекта реального мира.

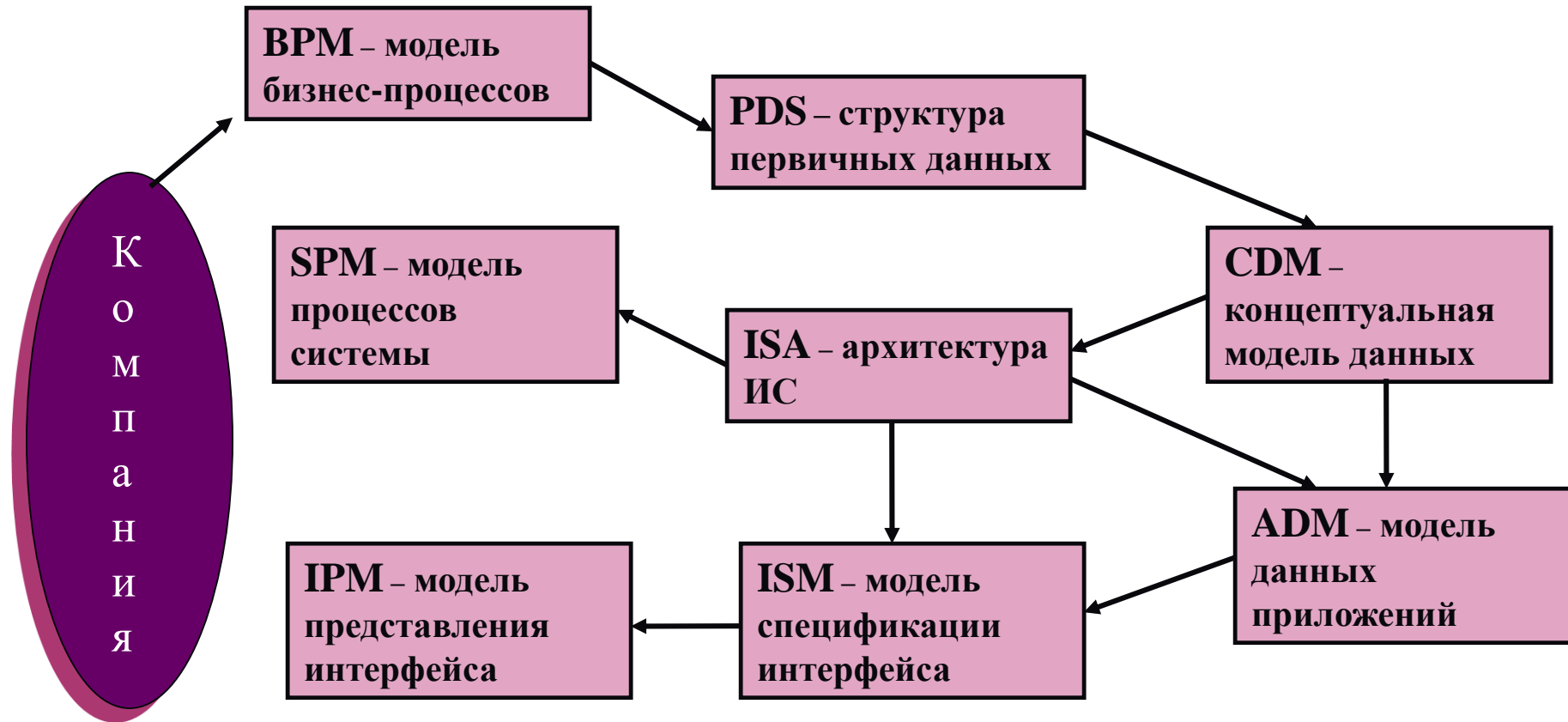
Технология Aris

- управляемые событиями модели.

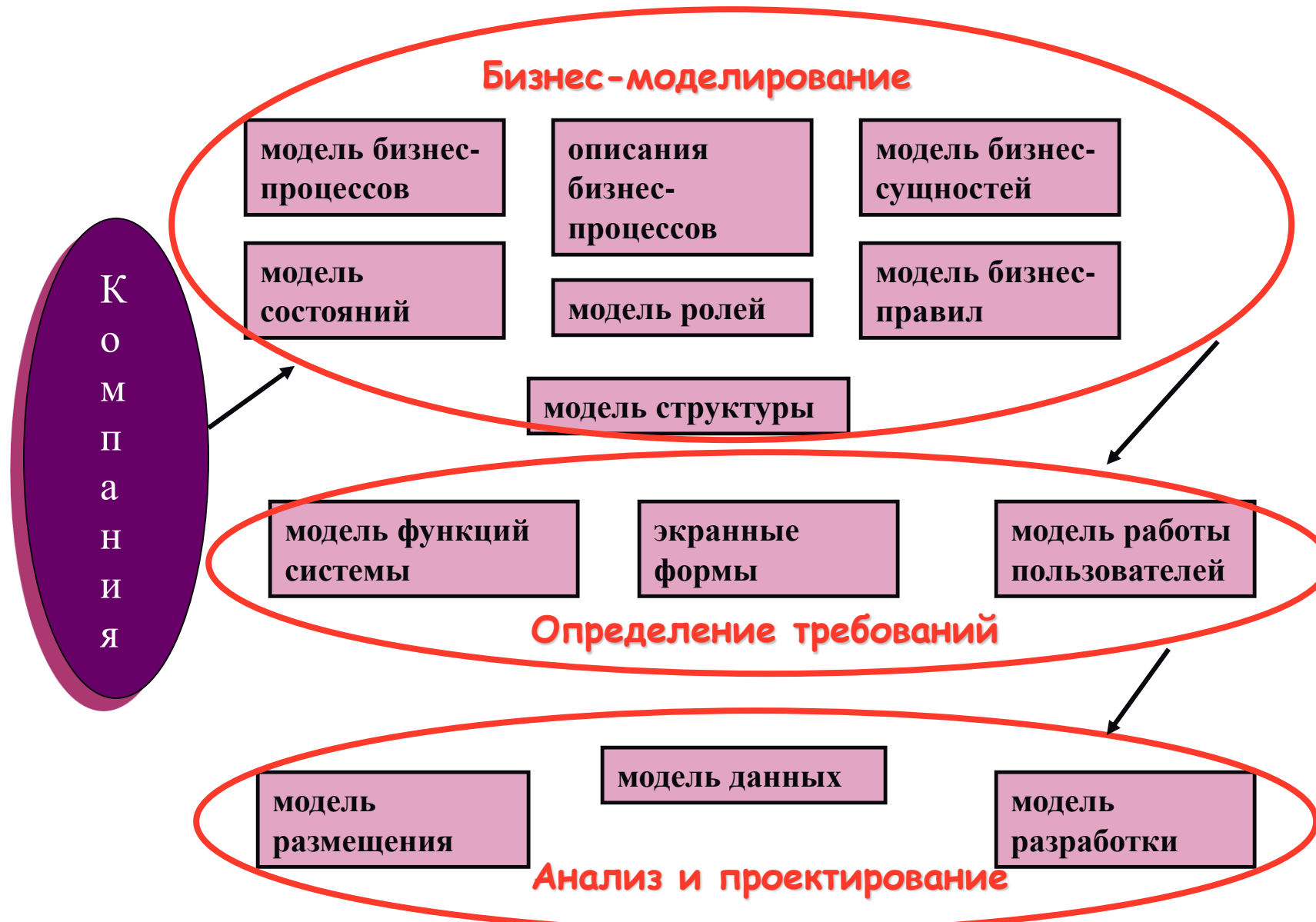
Программные средства проектирования

- IDEF Designer, ERwin\BPwin, Oracl Designer, BPM Workbench, Aris, Rational Rose

Технология проектирования DATARUN

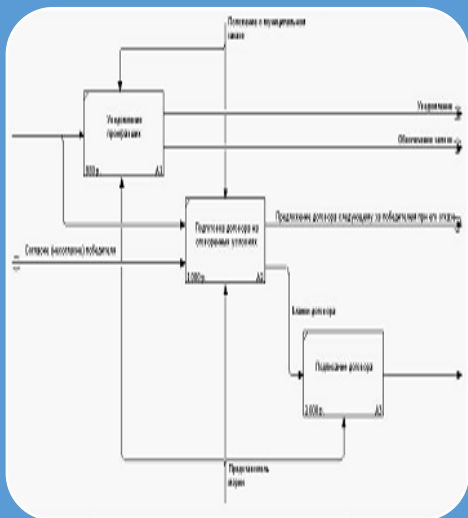


Технология проектирования RUP



Структурная методология проектирования

Сложность больших систем преодолевается расчленением их на части («черные ящики») и иерархической организацией этих «черных ящиков» в модели. На каждом уровне модели пользователю нет необходимости знать внутреннее устройство «черного ящика», рассматриваются только входы\выходы и реализуемая функция.



Критерии разбиения системы на «черные ящики»:

- каждый «черный ящик» реализует единственную функцию системы;
- функция каждого «черного ящика» должна быть легко понимаема независимо от сложности ее реализации;
- связи между «черными ящиками» вводятся только при наличии связи между соответствующими функциями системы;
- связи между «черными ящиками» должны быть максимально простыми

Стандарты IDEF

(Integrated Computer Aided Manufacturing DEFinition) (1981г)

- IDEF0 - методология функционального моделирования. Система отображается в виде набора взаимосвязанных функциональных блоков.
- IDEF1 – методология моделирования информационных потоков внутри системы, позволяющая отображать и анализировать их структуру и взаимосвязи;
- IDEF1X (IDEF1 eXtended) – методология построения реляционных структур, относится к типу методологий “Сущность-взаимосвязь” (ER – Entity-Relationship) и используется для моделирования реляционных БД в системе;
- IDEF3 – методология документирования процессов. С помощью IDEF3 описываются сценарий и последовательность операций для каждого процесса.
- IDEF4 – методология построения объектно-ориентированных систем.



Функциональный подход

В **функциональных моделях** главными структурными компонентами являются функции (операции, действия, работы), которые на диаграммах связываются между собой потоками объектов.

Для проверки корректности моделирования предметной области между функциональными и объектными моделями устанавливаются взаимно однозначные связи.

Модели данных

- При функциональном подходе объектные модели данных в виде ER-диаграмм "объект — свойство — связь" разрабатываются отдельно.

Функциональный подход



- реализация структурного подхода к проектированию по принципу "сверху-вниз", когда каждый функциональный блок может быть декомпозирован на множество подфункций и т.д.,
- модульное проектирование ИС.



- процессы и данные существуют отдельно друг от друга — помимо функциональной декомпозиции существует структура данных, находящаяся на втором плане,
- не ясны условия выполнения процессов обработки информации, которые динамически могут изменяться.

Объектно-ориентированная методология

Основными понятиями ОО подхода являются объект и класс.

Объект — предмет или явление, имеющее четко определенное поведение и обладающие состоянием, поведением и индивидуальностью. Структура и поведение схожих объектов определяют общий для них класс.

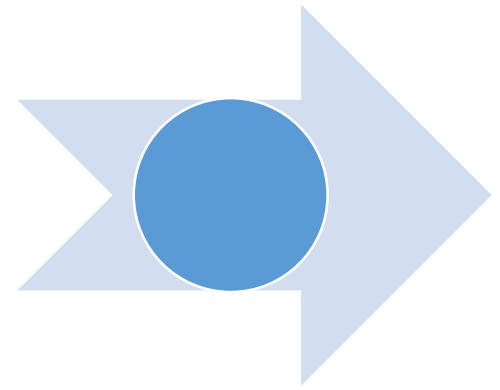
Класс – это множество объектов, связанных общностью структуры и поведения.

Полиморфизм может быть интерпретирован как способность класса принадлежать более чем одному типу.

Наследование означает построение новых классов на основе существующих с возможностью добавления или переопределения данных и методов.

Концептуальной основой является объектная модель, которая строится на принципах:

- абстрагирование;
- инкапсуляция;
- модульность;
- иерархия;
- типизация;
- параллелизм;
- устойчивость.



Язык и процесс моделирования

Большинство существующих методов объектно-ориентированного подхода включают **язык моделирования UML** и описание **процесса моделирования RUP**.

Процесс

- это описание шагов, которые необходимо выполнить при разработке проекта.

UML

- В качестве языка моделирования объектного подхода используется унифицированный язык моделирования **UML**, который содержит стандартный набор диаграмм для моделирования.

Объектно-ориентированный подход



- Объектная декомпозиция дает возможность создавать модели меньшего размера путем использования общих механизмов.
- Использование объектного подхода существенно повышает уровень унификации разработки и пригодность для повторного использования.
- Объектная декомпозиция позволяет избежать создания сложных моделей, так как она предполагает эволюционный путь развития модели.
- Объектная модель естественна, поскольку ориентированна на человеческое восприятие мира.



Высокие начальные затраты.

Этот подход не дает немедленной отдачи. Эффект от его применения сказывается после разработки двух–трех проектов и накопления повторно используемых компонентов.

Диаграммы, отражающие специфику объектного подхода, менее наглядны.

Сравнение методик

- ✓ Недостатки функциональных моделей снимаются в **объектно-ориентированных моделях**, в которых главным структурообразующим компонентом выступает класс объектов с набором функций, которые могут обращаться к атрибутам этого класса.
- ✓ Для классов объектов характерна иерархия обобщения, позволяющая осуществлять **наследование** не только атрибутов (свойств) объектов от вышестоящего класса объектов к нижестоящему классу, но и функций (методов).
- ✓ В случае наследования функций можно абстрагироваться от конкретной реализации процедур (**абстрактные типы данных**), которые отличаются для определенных подклассов ситуаций.
- ✓ Это дает возможность осуществлять повторное использование программного кода при модификации программного обеспечения.
- ✓ Таким образом, адаптивность объектно-ориентированных систем к изменению предметной области по сравнению с функциональным подходом значительно выше.

Сравнение методик

- ✓ При объектно-ориентированном подходе изменяется и принцип проектирования.
- ✓ Сначала выделяются классы объектов, а далее в зависимости от возможных состояний объектов (жизненного цикла объектов) определяются методы обработки (функциональные процедуры).
- ✓ Это обеспечивает наилучшую реализацию динамического поведения информационной системы.
- ✓ Для объектно-ориентированного подхода разработаны графические методы моделирования предметной области, обобщенные в языке унифицированного моделирования **UML**.
- ✓ Однако по наглядности представления модели пользователю-заказчику объектно-ориентированные модели явно уступают функциональным моделям.
- ✓ Каждая из рассмотренных методик позволяет решить задачу построения формального описания рабочих процедур исследуемой системы.
- ✓ Все методики позволяют построить модель **"как есть" (AS-IS)** и **"как должно быть" (TO-BE)**.

Критерий выбора методики моделирования

При выборе **методики моделирования** предметной области обычно в качестве критерия выступает **степень ее динамичности**.

В рамках одной и той же ИС для различных классов задач могут требоваться различные виды моделей, описывающих одну и ту же проблемную область, должны использоваться **комбинированные модели предметной области**.

Для более
регламентированных
задач

- больше подходят функциональные модели.

Для более адаптивных бизнес-
процессов

- управления рабочими потоками, реализации динамических запросов к информационным хранилищам — объектно-ориентированные модели.

Критерий выбора методики моделирования

Функциональные методики

- Лучше дают представление о существующих функциях в организации, о методах их реализации, чем выше степень детализации исследуемого процесса, тем лучше они позволяют описать систему.
- Под лучшим описанием понимается наименьшая ошибка при попытке по полученной модели предсказать поведение реальной системы.
- Но отсутствие выделенных управляющих процессов, потоков и событийной ориентированности не позволяет предложить эту методику в качестве единственной.

Объектный подход

- На уровне общего описания системы функциональные методики допускают значительную степень произвола в выборе общих интерфейсов системы, ее механизмов и т.д., то есть в определении границ системы.
- Хорошо описать систему на этом уровне позволяет объектный подход, основанный на понятии сценария использования.
- Сценарий использования - сеанс взаимодействия действующего лица с системой, в результате которого действующее лицо получает нечто, имеющее для него ценность.

Формирование комбинированной модели

*Недостатки применения отдельной методики лежат не в области описания реальных процессов, а в неполноте **методического подхода**.*

Наилучшим способом преодоления недостатков рассмотренных методик является формирование **синергетической методики**, которая позволит комбинировать различные модели отдельных методик.

При этом из каждой методики берется часть методологии, наиболее полно и формально изложенная, и обеспечивается возможность обмена результатами на различных этапах применения синергетической методики.

*Идея **синергетической методики (комбинированной модели)** заключается в последовательном применении функционального и объектного подхода с учетом возможности реинжиниринга существующей ситуации.*

Пример синтетической методики – построение комбинированной модели

1. Определение границ системы. На этой стадии при помощи **анализа потоков данных выделяют внешние сущности** и собственно моделируемую систему.
2. Выделение сценариев использования системы. На этой стадии **при помощи критерия полезности строят** для каждой внешней сущности **набор сценариев использования системы**.
3. Добавление системных сценариев использования. На этой стадии **определяют сценарии, необходимые для реализации целей системы**, отличных от целей пользователей.
4. Построение диаграммы активностей по сценариям использования. На этой стадии строят **набор действий системы**, приводящих к реализации сценариев использования;
5. Функциональная **декомпозиция диаграмм активностей** как контекстных диаграмм методики IDEF0.
6. Формальное описание отдельных функциональных активностей (с применением различных нотаций).