**Лекция № 5 (22)** – 29.09.23 г. МООАиП

### **8.1.4.4. Методология объектно-ориентированного анализа и**

### **проектирования**

Методология структурного анализа и проектирования, использующая в ка­честве центрального понятия - понятие “функция”, позволяет адекватно, т.е. до­статочно точно для практики, описать предметную область (объект автоматизации) – построить функциональную модель. Однако при этом функциональная модель оказывается критичной к точечным внесениям изменений, которые требуют переформатирования всей модели или значительной её части. Это – принципиальный недостаток данной методологии, осложняющий эффективную коллективную работу над проектом будущей системы.

Нечувствительной к корректировкам и допускающей масштабирование разрабатываемой модели является методология объектно-ориентированного анализа и проектирования (ООАиП). В основании этой методологии находится понятие “объект”, определение которого приведено ранее. Любой предмет, вещь, устройство, механизм и т.п., т.е. всё, что можно увидеть и потрогать (реальная действительность) – это объект. Всё, что можно познать, представить или вообразить, например, алгоритм, программа, файл, функция (виртуальная действительность) – это тоже является объектом. Каждый объект обладает свойствами и предназначен для выполнения какого-то действия, а также существует не в вакууме, а в окружении множества объектов, себе подобных, с некоторыми из которых находится в физических связях или логических отношениях. Так устроен мир и любая выделенная из него конкретная ситуация, которая по-разному может быть представлена на естественном языке, и каждая вариация будет своеобразной вербальной моделью данной ситуации.

Любой естественный язык состоит из слов и правил. Слова образуют словарь, а правила определяют порядок следования слов, образуя череду словосочетаний, предложений и состоящих из них текстов произвольной длины. Слова, словосочетания, предложения и порядок следования слов, словосочетаний и предложений с учётом взаимной соподчинённости или согласованности являются строительными блоками языка, и они являются уникальными для конкретного языка.

Люди общаются между собой на том или ином естественном языке. Естественный язык также используется человеком для описания окружающей его действительности – картины мира реальной или виртуальной. Естественный язык обладает исключительной выразительной мощностью. Но общения и картины мира на естественном языке сопровождаются существенной долей неоднозначности или неопределённости, которая в большинстве своём не критична для людей, однако для компьютеров она пока не допустима. Поэтому для вычислительной техники и средств телекоммуникаций приходится создавать искусственные языки.

Для методологии объектно-ориентированного анализа и проектирования искусственным языком является язык UML (англ. Unified Modeling Language - универсальный язык моделирования). Основные понятия этого языка, и в целом, объектно-ориентированного подхода кратко и понятно изложены в работе [1].

Словарь UML содержит три типа строительных блоков:

**сущности** – соответствуют реальным или виртуальным объектам действительности,

**отношения** – связывают различные сущности (объекты),

**диаграммы** – представляют собой упорядоченные множества сущностей и отношений, таким образом, описывая формально и однозначно некоторую фиксированную область действительности.

В UML предполагается, что объекты должны обладать свойствами:

**абстракции –** предполагает выделение важных и нужных свойств и операций объекта, позволяющих точно описать объект при решении конкретной задачи,

**наследования –** это общие свойства и операции, свойственные классам и их подклассам,

**полиморфизма –** предполагает применение операции с одинаковым названием, но различным содержанием для различных классов,

**инкапсуляции** – скрытие объектом выполнения внутренних операций от внешнего мира и других объектов [1, с. 5-6].

Например, в процессе функционирования хлебопекарни обязательно участвуют работники. Если сказать, что любой работник имеет ФИО, оклад, метод без реализации с названием «работать ()», то в процессе работы хлебопекарни будет более интересна и важна его конкретная должностная обязанность, мало будет важен работник как таковой, т.е. понятие «работник хлебопекарни» будет иметь абстрактное значение. В этом случае может быть применена абстракция и, следовательно, для того чтобы задать работнику его функции, которые определяются должностной обязанностью, могут быть применены, во-первых, наследование от «работника хлебопекарни», во-вторых, полиморфизм – определение соответствующей реализации метода «работать ()». Кроме того, такие данные как, например, оклад работника, обычно не публично известны, поэтому следует ограничить к ним доступ со стороны внешнего мира, что позволяет реализовать одно из свойств объектов – инкапсуляция.

**8.1.4.4.1. Объектно-ориентированный анализ**

Объектно-ориентированный анализ предметной области осуществляется с использованием языка UML, сведения о котором приведены в [2]. В общем случае – это язык диаграмм.

Каждая диаграмма UML используется для конкретной цели, обычно для визуализации определенного аспекта системы. В каждой диаграмме для достижения ее цели используются определенные символы UML.Этот язык моделирования содержит два различных типа базовых диаграмм: **диаграммы структуры** и **диаграммы поведения**. Диаграммы структуры отображают статическую структуру элементов системы [2, с. 18-19]. Из всех видов диаграмм структуры можно выделить следующие:

диаграммы классов**,**

диаграммы объектов.

Диаграммы поведения изображают динамическое поведение элементов системы [2, с. 19]. Из всех видов диаграмм поведения можно выделить следующие:

диаграммы прецедентов,

диаграммы сотрудничества,

диаграммы деятельностей.

Более детально и наглядно каждая диаграмма UML разобрана в[3].

Прежде чем приступать к реализации любого проекта, необходимо иметь ясные представления о предметной области. Проще говоря, необходимо определить назначение проекта: какие цели преследует будущая система, какие функции и возможности она должна предоставить конечному пользователю. В связи с этим разработку «чертежа» системы следует начинать с построения **диаграммы прецедентов**. Диаграмма прецедентов описывает то, что должна выполнять проектируемая система, не вдаваясь в подробности, как она это выполняет [3, с. 9-10].

Главная цель прецедентного моделирования – описание функциональных требований к системе.

Ключевыми элементами диаграммы прецедентов являются:

**актер –** представляет собой связное множество ролей, которые пользователи исполняют во время взаимодействия с проектируемой системой,

**прецедент** – описание множества последовательностей действий (включая их варианты), которые выполняются системой для того, чтобы актер получил результат, имеющий для него определенное значение,

**связь –** отношение ассоциации, которое устанавливает, какую конкретную роль играет актер при взаимодействии с прецедентом [3, с. 12-13].

Диаграммы прецедентов в нотации UML и на примере хлебопекарни представлены на рис. 8.4.4.1 [2, с. 263] и рис. 8.4.4.2.

Актер

Однонаправленное соединение

Прецедент

Рис. 8.4.4.1. Пример диаграммы прецедентов в нотации UML

Персонал

Выпечка хлеба

Подготовка сырья

Подготовка рецепта

Проверка хлебопекарного оборудования

“включает”

“включает”

“включает”

Рис. 8.4.4.2. Диаграмма прецедентов на примере хлебопекарни

После того как будет получено четкое представление о том, кто и как будет пользоваться системой, необходимо перейти к составлению словаря понятий предметной области, т.е. определению объектов, с которыми работает исследуемая система, и связей между ними. Группы объектов с общими свойствами объединяются в классы. На этом этапе важно грамотно построить **диаграмму классов**, так как в результате этого будет получена объектно-ориентированная картина будущей системы [3, с. 10].

Диаграммы классов используются для моделирования статического вида системы с точки зрения проектирования. На диаграмме классов показано множество классов, интерфейсов и отношений между ними.

**Классом** называют описание совокупности объектов с общими атрибутами, операциями, отношениями и семантикой.

**Атрибутом** называется именованное свойство класса, включающее описание множества значений.

**Операцией** называется реализация услуги, которую можно запросить у любого объекта класса для воздействия на поведение.

**Интерфейсом** называется набор операций, используемый для специфицирования услуг, предоставляемых классом [3, 29-31].

Диаграммы классов в нотации UML и на примере хлебопекарни приведены на рис. 8.4.4.3 [2, с. 266] и рис. 8.4.4.4.

Для иллюстрации конкретных примеров и отдельных частей сложных систем дополнительно может быть построена **диаграмма объектов** [3, с. 10].

Диаграммы объектов позволяют моделировать экземпляры сущностей, которые содержатся в диаграммах классов. На диаграмме объектов показано множество объектов и отношений между ними в некоторый конкретный момент времени.

**Экземпляром** называется конкретная материализация абстракции, к которой могут быть применены операции и которая может сохранять их результаты.

**Объект** является отдельным экземпляром класса, который создается на этапе выполнения программы [3, с. 49].

Диаграмма объектов применяется обычно для описания небольших частей системы со сложными отношениями или для иллюстрации конкретных примеров (рис. 8.4.4.5 и рис. 8.4.4.6).

# **8.1.4.4.2. Объектно-ориентированное проектирование**

После того как получено функционально-ориентированное и объектно-ориентированное представление системы, необходимо объединить их и получить спецификацию системы, которую уже можно выдавать программистам на

Класс

Реализация

Соединение

Интерфейс

Зависимость

Обобщение

Агрегация

Рис. 8.4.4.3. Пример диаграммы классов в нотации UML

+ФИО

#Оклад

Работник хлебопекарни

Смеситель

+Замесить тесто()

Рабочий В

+Утилизировать отходы()

Эксперт

+Проверить качество()

Рабочий Б

+Перегрузить на поддон()

Рабочий А

+Загрузить хлебопечку()

Рис. 8.4.4.4. Диаграмма классов на примере персонала хлебопекарни

имяАтрибута = значение

имяОбъекта: имяКласса

имяАтрибута = значение

имяОбъекта: имяКласса

имяАтрибута = значение

имяОбъекта: имяКласса

Рис. 8.4.4.5. Пример диаграммы объектов в нотации UML

ФИО = “Иванов Иван Иванович”

Оклад = 20000

Иван: Смеситель

Вид = “Мука”

Мука: Сырье

Вид = “Сырье”

Вода: Сырье

Вид = “Дрожжи”

Дрожжи: Сырье

Вид = “Соль”

Соль: Сырье

Рис. 8.4.4.6. Диаграмма объектов на примере отдельной части хлебопекарни (конкретные виды сырья, которые использует для приготовления теста конкретный смеситель)

реализацию. А для этого необходимо построить диаграмму взаимодействия для соответствующих прецедентов [3, с.10].

Одной из диаграмм взаимодействия UML является **диаграмма сотрудничества**. Диаграмма сотрудничества акцентирует внимание на организации

объектов, принимающих участие во взаимодействии.

Ключевыми элементами диаграммы сотрудничества являются:

**объект**,

**связь**,

**сообщение** [3, с. 63].

На рис. 8.4.4.7 [2, с. 265] и рис. 8.4.4.8 приведены диаграммы сотрудничества в нотации UML и на примере реализации прецедента «Подготовка сырья».

Следующим этапом разработки системы может быть построение динамической модели. Динамическая модель может быть представлена **диаграммой деятельностей**, которая иллюстрирует логику и последовательность переходов от одной деятельности к другой [3, с. 10].

Ключевыми элементами диаграммы деятельностей являются:

**состояние деятельности** – специальный случай состояния с некоторым входным действием и, по крайней мере, одним выходящим из состояния переходом,

**переход** – показывает путь из одного состояния деятельности в другое,

**точка принятия решений** – описывает различные пути выполнения в зависимости от значения некоторого булевого выражения,

**полоса синхронизации** – используется для обозначения разделения и слияния параллельных потоков (стадий процесса функционирования системы, которые могут выполняться параллельно) [3, с. 85-87].

Диаграммы деятельностей в нотации UML и на примере хлебопекарни приведены на рис. 8.4.4.9 [2, с. 264] и рис. 8.4.4.10.

В итоге изложены основные идеи методологии объектно-ориентированного анализа и проектирования, которые подробно рассмотрели в [1], [2], [3].

Следует сказать, что при работе над любым серьезным проектом нужно еще на стадии анализа и проектирования системы, а не на этапе программирования или стадии эксплуатации понять то, что разрабатываемая система может и насколько это соответствует ожиданиям и потребностям пользователя. Язык UML представляет собой инструмент, с помощью которого можно построить «чертеж» будущей системы, понятный не только всем тем, кто принимает участие в разработке, но и конечному пользователю системы [3, с. 113].

1: Сообщение

Объект

Связь

Сообщения

Рис. 8.4.4.7. Пример диаграммы сотрудничества в нотации UML

Поставщик

Контроль

Хранение

:Смеситель

1: принять товар

2: загрузить на склад

3: предоставить сырье

Рис. 8.4.4.8. Диаграмма сотрудничества на примере реализации прецедента «Подготовка сырья»

Название 1-ой дорожки

Название 2-ой дорожки

Переход

Действие

Нет

Да

Решение

Горизонтальная синхронизация

Начальное состояние

Конечное состояние

Рис. 8.4.4.9. Пример диаграммы деятельностей в нотации UML

Смеситель

Рабочий А

Рабочий Б

Эксперт

Рабочий В

Приготовление теста

Загрузка хлебопечки

Перегрузка на поддоны

Проверка качества

Утилизация брака

Отходы

Брак

[Да]

[Нет]

Хлеб

Рис. 8.4.4.10. Диаграмма деятельностей на примере хлебопекарни

Литература

1. Коуров В.Г. Проектирование информационных систем на базе UML и концепции ECO. М.: МГГУ им М.А. Шолохова. 2007. – 95 с.

2. Максимчук Р.А. и др. UML для простых смертных. М.: Лори. 2008. – 268 с.

3. Файзрахманов Р.А. и др. Проектирование автоматизированных информационных систем на основе объектно-ориентированного подхода. Пермь: Изд-во Пермского гос. технического ун-та. 2011. – 222 с.

**Контрольные вопросы для самопроверки знания**

**лекционного материала**

**КВ №316.** Дать лекционные определения понятий “методология”, “способ”, “метод” и довести воспроизведение понятий до автоматизма, т.е. быстро и правильно по памяти. Методология объектно-ориентированного анализа и проектирования: краткая характеристика. В методологии что является способом и что методом.

**КВ №317.** Дать лекционное определение понятия “черный ящик” и из лекции привести графическую модель “черный ящик”. Довести воспроизведение определения и модели до автоматизма, т.е. быстро и правильно по памяти. Представить ГАС “Выборы” моделью “черный ящик”.

**КВ №318.** Дать лекционные определения понятий “контроль”, “измерение” и привести из лекции графическую модель системы контроля. Довести воспроизведение определений до автоматизма, т.е. быстро и правильно по памяти. Представить ГАС “Выборы” в виде СК.

**КВ №319.** Дать лекционные определения понятий “система целенаправленная”, “система целеустремлённая”, “система человеко-машинная”, “система самоорганизующаяся”, “система автоматического управления”, “ автоматизированная система управления”, “ручная система управления” и довести воспроизведение определений до автоматизма, т.е. быстро и правильно по памяти. По способности к целеполаганию к каким классам (целеустремлённым, целенаправленным, человеко-машинным или самоорганизующимся) следует отнести САУ, АСУ, РСУ. К какой из перечисленных систем следует отнести пассажирский самолет и ракету, летящие в атмосфере.

**КВ №320.** Дать лекционные определения понятий “система управления”, “система автоматического управления”, “ автоматизированная система управления”, “ручная система управления” и привести из лекции графическую модель системы управления. Довести воспроизведение определений и модели до автоматизма, т.е. быстро и правильно по памяти. По способности к целеполаганию к каким классам (целеустремлённым, целенаправленным, человеко-машинным или самоорганизующимся) следует отнести САУ, АСУ, РСУ. Оценить по шкале “потенциальный – умеренный – чрезвычайный” удельный вес человеческого фактора в САУ, АСУ, РСУ. Предложить альтернативную шкалу для оценки человеческого фактора в автоматизированных системах.

**КВ №321.** Дать лекционные определения понятий “объект автоматизации (предметная область)”, ” система управления”, “система автоматического управления”, “ автоматизированная система управления”, “ручная система управления” и привести из лекции графическую модель системы управления. Довести воспроизведение определений и модели до автоматизма, т.е. быстро и правильно по памяти. Сформулировать предметные области для САУ, АСУ, РСУ.

**КВ №322.** Дать лекционные определения понятий “система”, “объект”, “управление”, ““объект управления”, “система управления” и привести графическую модель системы управления. Довести воспроизведение определений и модели до автоматизма, т.е. быстро и правильно по памяти. Построить графическую модель САУ.

**КВ №323.** Дать лекционные определения понятий “система”, “объект”, “управление”, ““объект управления”, “система управления” и привести графическую модель системы управления. Довести воспроизведение определений и модели до автоматизма, т.е. быстро и правильно по памяти. Построить графическую модель АСУ.

**КВ №324.** Дать лекционные определения понятий “система”, “объект”, “управление”, ““объект управления”, “система управления” и привести графическую модель системы управления. Довести воспроизведение определений и модели до автоматизма, т.е. быстро и правильно по памяти. Построить графическую модель РСУ.

**КВ №325.** Дать лекционные определения понятий “целеустремлённость” и “архитектура АСОИУ” как среды жизнедеятельности будущего, зарождающейся сегодня. Довести воспроизведение определений до автоматизма, т.е. быстро и правильно по памяти. Охарактеризовать информацию, интеллект и технологию как триединый базис архитектуры АСОИУ. Технологическая целеустремлённость ведущих мировых держав (Китай, России, США) и судьба глобального миропорядка.