

**Шарук Алексей Андреевич**

Магистрант НАЧОУ ВПО СГА

**Направление:** Информатика и вычислительная техника

**Магистерская программа:** Распределенные автоматизированные системы

**Сопоставление и взаимосвязь структурного и объектно-ориентированного подходов в проектировании ИС**

**Аннотация.** Применение информационных технологий немислимо без создания программного обеспечения, разработка которого зачастую является сложным многоступенчатым процессом. В ходе этого процесса осуществляются ручные трансформации неформальных моделей, определяющих объекты решаемой задачи, в формализованное представление предметной области. Полученные модели описываются в виде программ на языках программирования, после чего происходит их преобразование в машинные коды. Во многом специфика разрабатываемых приложений определяется предметной областью, характеризующей состав и особенности моделей, описывающих решаемую задачу.

**Ключевые слова:** структурный подход, объектно-ориентированный подход.

Разработка информационных систем – достаточно сложная, трудоемкая работа, занимающая долгий период времени и требующая привлечения специалистов высокой квалификации. Несмотря на это разработка информационных систем нередко выполняется на интуитивном неформальном уровне, включающим в себя практический опыт разработчика, экспертные оценки и частые экспериментальные проверки качества функционирования разрабатываемой системы. В процессе создания и функционирования информационной системы потребности пользователей постоянно изменяются

или уточняются, усложняя проектирование и техническую поддержку таких систем.

У большинства людей понятие «программирование» создает ассоциации программирования структурным методом как например в языке Basic или Pascal, т. е. поэтапное исполнение команд, «сверху-вниз» на основе декомпозиции по функциональности, в которой вся информационная система в целом представляется как единая функция раздробленная на более мелкие подфункции, которые также разбиваются еще на множество более мелких подфункций. Процесс останавливается, когда подзадачу невозможно разбить на более мелкую подзадачу или подобное разделение является явно нецелесообразным.

В подобном случае информационная система конструируется иерархически – сверху вниз: от основной программы к подпрограммам более нижнего уровня, причем на каждом уровне допускается использование только простых последовательностей инструкций, циклов и условных разветвлений.

Разбиение программы на более мелкие подпрограммы представляет одну из наиболее ответственных и трудных задач, так как это разбиение влияет на дальнейший ход разработки всей системы, его надёжность, эффективность, продолжительность процесса тестирования и процесса отладки системы, техническое сопровождение и возможность использование комплектационных программных процедур.

Итак, сущность структурного подхода к разработке программного обеспечения заключается в его разбиении на автоматизируемые функции: система разбивается на функциональные подсистемы, которые, в свою очередь, делятся на подфункции, те – на задачи итак далее до конкретных процедур [1]. При этом автоматизируемая система сохраняет целостное представление, в котором все составляющие компоненты взаимосвязаны. При разработке системы «снизу-вверх», от отдельных задач ко всей системе, целостность теряется, возникают проблемы при описании информационного взаимодействия отдельных компонентов.

К сожалению, структурного подхода бывает недостаточно для разработки более сложной системы, необходим более новый подход к методам проектирования информационных систем – технология объектно-ориентированного программирования.

Объектно-ориентированное программирование включает лучшие идеи, воплощённые как в структурном программировании, так и в модульном [2].

Объектно-ориентированное программирование позволяет разложить проблему на составные части. Каждая составляющая становится самостоятельным объектом, содержащим свои собственные коды и данные, которые относятся к этому объекту. В этом случае программирование в целом упрощается, и программист получает возможность оперировать гораздо большими по объёму программами.

Объектно-ориентированный подход основан на категории объектной модели – классе, он соединяет в себе данные на элементарном уровне, так и методы, которые над ними выполняются. Как раз эти различия, связанные с переходом от структурного к объектно-ориентированному подходу, являются более критично заметными. При преодолении разделения процессов от данных необходимо решить проблему сложности системы, которая решается методом использования механизма компонентов. Синтаксис объектно-ориентированных языков описывает всего одну операцию: послать сообщение А объекту В с параметрами С1СВn. Параметры сообщения – это объекты, которые сами могут быть результатами обработки других сообщений.

К недостаткам объектно-ориентированного программирования относится некоторая потеря производительности в работе разработанного приложения, которое в современных компьютерах, по мере роста их производительности, становится менее заметным. Переход на технологию объектно-ориентированного проектирования – это смена мировоззрения проектировщика, а не просто изучение новых инструментов и языков программирования. Кроме того, при переходе от структурного подхода к

объектному, необходимы финансовые затраты на приобретение новых средств разработки, дополнительных библиотек и компонентов.

Однако, комбинируя методы проектирования возможно добиться большего результата. Взаимосвязь структурного и объектно-ориентированного подхода выявлена в общности понятий обоих подходов и может проявляться в различных формах. Так, одно из вариантов является использование структурного анализа и составления модели как основы для объектно-ориентированного проектирования. Структурный и объектно-ориентированный подходы к программированию различаются по своей сути и обычно ведут к совершенно разным решениям одной задачи. Этот вывод верен как для стадии реализации, так и для стадии проектирования системы.

Конечно, объектно-ориентированная модель более понятно отражает реальный процесс проектирования, представляющий собой взаимосвязь взаимодействующих объектов системы. На практике же, при проектировании и составлении мануалов продолжается формирование стандарта языка поддерживающими структурный подход, диаграммы, отражающие особенности объектного подхода (диаграммы классов и т. п.), гораздо более наглядны и легко понимаются непрофессионалами. Именно поэтому задача внедрения взаимосвязанных технологий, а именно обеспечение всех участников проекта, в том числе и будущих пользователей системы языком понимания, обеспечивается на сегодняшний день только структурными методами.

Сравнивая объектно-ориентированный и структурный стиль программирования, является важным выбрать критерии для сравнения.

Основной критерий в оценке программных продуктов – это сложность, а основными требованиями к технологии разработки являются: удобство сопровождения, возможность наращивания уже существующей программы новыми модулями, способность разработанных программных объектов к использованию в похожих проектах [3].

При этом на второй план отступает такое требование, как время, затраченное на проектирование первоначальной версии программы, потому что

его воплощение обычно не позволяет соблюсти все остальные условия потому, что процесс разработки программного обеспечения не заканчивается первой версией, а чаще всего, дает новый толчок к анализу и пересмотру методов и технологий программирования.

Основой взаимосвязи между структурным и объектно-ориентированным подходами является общность ряда категорий и понятий обоих подходов (процесс и вариант использования, сущность и класс и др.). Эта взаимосвязь может проявляться в различных формах. Так, одним из возможных подходов является использование структурного анализа как основы для объектно-ориентированного проектирования. При этом структурный анализ следует прекращать, как только диаграммы потоков данных начнут отражать не только предметную область, но и область программного обеспечения.

После выполнения структурного анализа и построения диаграмм потоков данных вместе со структурами данных и другими продуктами анализа можно различными способами приступить к определению классов и объектов. Так, если взять какую-либо отдельную диаграмму, то кандидатами в объекты могут быть внешние сущности и накопители данных, а кандидатами в классы – потоки данных.

При проектировании достаточно больших систем происходит сбор информации, ее анализ и анализ сложности разработки на начальных этапах (анализ требований к разрабатываемой системе и разработка спецификаций системы для более качественного ее восприятия), в то время как сложность и трудоемкость последующих этапов остается сравнительно низкой. Как раз, для преодоления трудностей проектирования системы на начальном этапе разработки рекомендовано использовать структурный метод проектирования – метод исследований, который начинается с общего планирования системы и по ходу проектирования детализуется, создается иерархическую структуру будущей системы с большим числом уровней. Каждый уровень рассматривает ограниченное число элементов, каждый из которых, в свою очередь, может быть разделен на составляющие детали на следующем уровне декомпозиции.

При этом соблюдаются формальные правила ведения информации, обычно используются диаграммы различных типов или структурные схемы.

### Литература

1. Алешин Л.И. Руководство по изучению дисциплины «Автоматизированные информационные системы. М., 2006.
2. Объективно-ориентированное программирование [Электронный ресурс] // Режим доступа: [http://cyberfac.ru/publ/informatika/algorithmizacija\\_i\\_programmirovanie/obektno\\_orientirovannoe\\_programmirovanie\\_oop/43-1-0-1378](http://cyberfac.ru/publ/informatika/algorithmizacija_i_programmirovanie/obektno_orientirovannoe_programmirovanie_oop/43-1-0-1378)
3. Буч Г. Объектно-ориентированный анализ и проектирование с примерами приложений на C++. 2-е изд.: Пер. с англ. М.: Бином; СПб.: Невский диалект, 1999.

© Бюллетень магистранта 2014 год №6