# 1. Методологические основы проектирования информационных систем

## 1.1 Технология проектирования информационных систем

Под проектом ИС будет понимать проектно-конструкторскую и технологическую документацию, в которой представлено описание проектных решений по созданию и эксплуатации ИС в конкретной программно-технической среде.

Под проектированием ИС понимается процесс преобразования входной информации об объекте проектирования, о методах проектирования и об опыте проектирования объектов аналогичного назначения в соответствии со стандартами в проект ИС.

Объектами проектирования ИС являются отдельные элементы или их компоненты функциональных и обеспечивающих частей. Так, функциональными элементами в соответствии с традиционной декомпозицией выступают задачи, комплексы задач и функции управления. В составе обеспечивающей части ИС объектами проектирования служат элементы и их компоненты информационного, программного и технического обеспечения системы.

В качестве субъекта проектирования ИС выступают коллективы специалистов, которые осуществляют проектную деятельность, как правило, в составе специализированной проектной организации, и организация-заказчик, для которой необходимо разработать ИС. При большом объеме и жестких сроках выполнения проектных работ в разработке системы может принимать участие несколько проектных коллективов. В этом случае выделяется головная организация, которая координирует деятельность всех организаций-соисполнителей.

Осуществление проектирования ИС предполагает использование проектировщиками определенной технологии проектирования, соответствующей масштабу и особенностям разрабатываемого проекта. Технология проектирования – это совокупность методологии и средств проектирования ИС, а также методов и средств организации проектирования.

В основе технологии проектирования лежит технологический процесс, который определяет действия, их последовательность, состав исполнителей, средства и ресурсы, требуемые для выполнения этих действий. Так, технологический процесс проектирования ИС в целом делится на совокупность последовательно-параллельных, связанных и соподчиненных цепочек действий. Действия, которые выполняются при проектировании ИС, могут быть определены как неделимые технологические операции или как подпроцессы технологических операций. Все действия могут быть собственно проектировочными, которые формируют или модифицируют результаты проектирования, и оценочными действиями, которые вырабатывают по установленным критериям оценки результатов проектирования.

Таким образом, технология проектирования задается регламентированной последовательностью технологических операций, выполняемых в процессе создания проекта на основе того или иного метода, в результате чего стало бы ясно, не только ЧТО должно быть сделано для создания проекта, но и КАК, КОМУ и в КАКОЙ ПОСЛЕДОВАТЕЛЬНОСТИ это должно быть сделано.

К основным требованиям, предъявляемым к выбираемой технологии проектирования, относятся следующие:

– созданный с помощью этой технологии проект должен отвечать требованиям заказчика;

– выбранная технология должна максимально отражать все этапы цикла жизни проекта;

– выбираемая технология должна обеспечивать минимальные трудовые и стоимостные затраты на проектирование и сопровождение проекта;

– технология должна быть основой связи между проектированием и сопровождением проекта;

– технология должна способствовать росту производительности труда проектировщика;

– технология должна обеспечивать надежность процесса проектирования и эксплуатации проекта;

– технология должна способствовать простому ведению проектной документации.

## 1.2 Принципы проектирования сложных объектов

При проектировании сложных объектов используются следующие принципы:

– декомпозиция и иерархичность построения описаний объектов проектирования;

– многоэтапность и итерационность процесса проектирования;

– типизация и унификация проектных решений.

Описания технических объектов должны быть по сложности согласованы: 1) с возможностями восприятия человеком; 2) с возможностями оперирования описаниями в процессе их преобразования с помощью имеющихся средств проектирования.

Выполнить эти требования в рамках единого описания удается лишь для простых изделий. Как правило, требуется структурирование описаний и соответствующее разбиение представлений об объекте на иерархические уровни и аспекты. Это позволяет распределить работы по проектированию сложных объектов между подразделениями проектировщиков, что способствует повышению эффективности и производительности труда.

Разделение описаний по степени детализации отображаемых свойств и характеристик объекта лежит в основе блочно-иерархического подхода к проектированию и приводит к появлению иерархических уровней (уровней абстрагирования) в представлениях об объекте.

Рисунок2_2

Рис. 1.1 – Блочно-иерархическое разделение сложного объекта

На уровне 0 (верхнем уровне) сложный объект  рассматривается как система  из  взаимно связанных и взаимодействующих элементов  на уровне 1 (см. рис. 1.1).

Каждый из элементов в описании уровня 1 представляет собой также довольно сложный объект, который, в свою очередь, рассматривается как описание системы  на уровне 2. Элементами системы  являются объекты , , где  – количество элементов в описании системы . Как правило, выделение элементов  происходит по функциональному признаку.

Подобное разбиение продолжается вплоть до получения на некотором уровне элементов, описания которых дальнейшему делению не подлежат, то есть до элементов, описание которых уже известно. Такие элементы по отношению к объекту  называются ***базовыми*** элементами.

***Принцип иерархичности*** означает структурирование представлений об объекте проектирования по степени детальности описаний.

***Принцип декомпозиции*** означает разбиение представлений каждого уровня на ряд составных частей (блоков) с возможностью раздельного (поблочного) проектирования объектов на уровне 1, объектов  на уровне 2 и т.д., т.е. каждый уровень разбивается на блоки.

Если решение задач высоких иерархических уровней предшествует решению задач более низких иерархических уровней, то проектирование называют *нисходящим*. Если раньше выполняются этапы, связанные с низшими иерархическими уровнями, то проектирование называют *восходящим*. У каждого из этих двух видов проектирования имеются преимущества и недостатки.

При нисходящем проектировании система разрабатывается в условиях, когда ее элементы еще не определены и, следовательно, сведения о их возможностях и свойствах носят предположительный характер.

При восходящем проектировании, наоборот, элементы проектируются раньше системы, и, следовательно, предположительный характер имеют требования к системе. В обоих случаях из-за отсутствия исчерпывающей исходной информации имеют место отклонения от возможных оптимальных технических результатов.

Поскольку принимаемые предположения могут не оправдаться, часто требуется повторное выполнение проектных процедур предыдущих этапов после выполнения проектных процедур последующих этапов. Такие повторения обеспечивают последовательное приближение к оптимальным результатам и обуславливают *итерационный* характер проектирования.

На практике обычно сочетают восходящее и нисходящее проектирование. Например, восходящее проектирование имеет место на всех тех иерархических уровнях, на которых используются *унифицированные* (стандартные) элементы. Очевидно, что унифицированные элементы, ориентированные на применение в ряде различных систем определенного класса, разрабатываются раньше, чем та или иная конкретная система из этого класса.

Обычно унификация объектов имеет целью улучшение технико-экономических показателей производства и эксплуатации изделий. Использование типовых и унифицированных проектных решений приводит так же к упрощению и ускорению проектирования.

Однако, унификация целесообразна только в таких классах объектов, в которых из сравнительно небольшого числа разновидностей элементов предстоит проектирование и изготовление большого числа систем. Именно эти разновидности элементов подлежат унификации.

Для сложных систем и для элементов, реализующих новые физические принципы или технологические возможности, в каждом конкретном случае приходится заново выполнять многоуровневое иерархическое проектирование.

В этих условиях целесообразно ставить вопрос не об унификации изделий, а об унификации средств их проектирования и изготовления, в частности об унификации проектных процедур в рамках систем автоматизированного проектирования.

Окончательное описание проектируемого объекта представляет собой полный комплект схемной, конструкторской и технологической документации, оформленной в соответствии с требованиями ГОСТов: ЕСКД (единая система конструкторской документации), ЕСТД (единая система технологической документации), ЕСПД (единая система программной документации). Этот комплект документации предназначен для использования в процессе изготовления и эксплуатации объекта проектирования.

Важное значение в этих описаниях имеют математические модели объектов проектирования, так как выполнение проектных процедур при автоматизированном проектировании основано на оперировании математическими моделями.

Математическая модель (ММ) технического объекта – система математических объектов (чисел, переменных, матриц, множеств и т.п.) и отношений между ними, отражающих некоторые свойства технического объекта.

При проектировании используют математические модели, отражающие свойства объекта, существенные с позиции проектировщика.

Среди свойств объекта, отражаемых в описаниях на определенном иерархическом уровне, в том числе в ММ, различают свойства: систем; элементов систем и внешней среды, в которой должен функционировать объект. Количественное выражение этих свойств осуществляется с помощью величин, называемых параметрами. Величины, характеризующие свойства системы, элементов системы и внешней среды, называют соответственно входными, внутренними и внешними параметрами.

Однако существование ММ не означает, что она известна разработчику и может быть представлена в явном функциональном виде. Типичной является ситуация, когда математическое описание процессов в проектируемом объекте задается моделью в форме системы уравнений, в которой фигурирует вектор фазовых переменных. Фазовые переменные характеризуют физическое или информационное состояние объекта, а их изменения во времени выражают переходные процессы в объекте. Например, состояние некоторой фирмы можно определить такими фазовыми переменными: сырье, материалы, финансовые и трудовые ресурсы.

Выделим следующие особенности параметров в моделях проектируемых объектов:

1. Внутренние параметры в моделях -го иерархического уровня становятся выходными параметрами в моделях более низкого ()-го иерархического уровня. Так, например, трудовые ресурсы являются внутренними при проектировании производственной фирмы и в то же время выходными при проектировании отдела кадров этой фирмы.

2. Выходные параметры или фазовые переменные, фигурирующие в модели одной из подсистем, часто оказываются внешними параметрами в описании других подсистем. Так, например, выходные параметры подсистемы планирования выпуска продукции некоторой компании являются внешними параметрами подсистемы материально-технического снабжения этой компании.

3. Большинство выходных параметров объекта являются функционалами.

4. В техническом задании на проектирование должны фигурировать величины, называемые техническими требованиями к выходным параметрам (нормами выходных параметров). Данные нормы представляют собой границы допустимых диапазонов изменения выходных параметров.

## 1.3 Классификация типовых проектных процедур

Проектная процедура называется *типовой*, если она предназначена для многократного применения при проектировании многих типов объектов.

Различают проектные процедуры анализа и синтеза. Синтез заключается в создании описания объекта, а анализ – в определении свойств и исследовании работоспособности объекта по его описанию, т.е. при синтезе создаются, а при анализе оцениваются проекты объектов.

Процедуры анализа делятся на процедуры одно- и многовариантного анализа. При одновариантном анализе заданы значения внутренних и внешних параметров, требуется определить значения выходных параметров объекта. Подобная задача обычно сводится к однократному решению уравнений, составляющих математическую модель, что и обуславливает название этого вида анализа. Многовариантный анализ заключается в исследовании свойств объекта в некоторой области пространства внутренних переменных. Такой анализ требует многократного решения систем уравнений (многократного выполнения одновариантного анализа).

Процедуры синтеза делятся на процедуры структурного и параметрического синтеза. Целью структурного синтеза является определение структуры объекта – перечня типов элементов, составляющих объект, и способа связи элементов между собой в составе объекта. Параметрический синтез заключается в определении числовых значений параметров элементов при заданных структуре и условиях работоспособности на выходные параметры объекта, т.е. при параметрическом синтезе нужно найти точку или область в пространстве внутренних параметров, в которых выполняются те или иные условия (обычно условия работоспособности).

На рис. 1.2 представлена типичная последовательность проектных процедур на одном из этапов нисходящего проектирования. На предыдущем этапе решались задачи ()-го иерархического уровня.

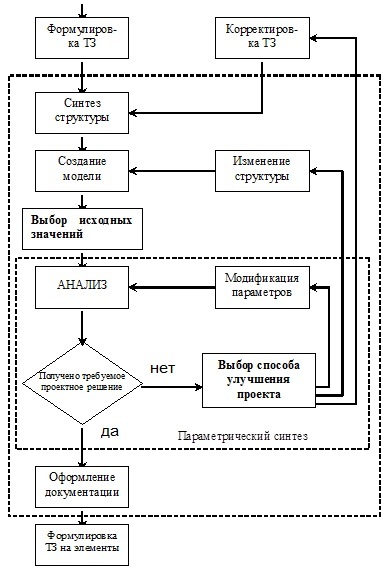


Рис. 1.2 – Схема процесса проектирования

Проектирование системы начинается с синтеза исходного варианта ее структуры. После выбора исходных значений параметров элементов выполняется анализ варианта, по результатам которого становится возможной его оценка. Обычно оценка заключается в проверке выполнения условий работоспособности, сформулированных в техническом задании (ТЗ).

Если условия работоспособности выполняются в должной мере, то полученное проектное решение принимается, затем описывается система ()-го уровня в принятой форме и формулируется ТЗ на проектирование элементов данного уровня. Если же полученное проектное решение неудовлетворительно, то выбирается один из возможных путей улучшения проекта.

Обычно проще всего изменить числовые значения параметров элементов. Совокупность процедур модификации параметров элементов, анализа и оценки результатов анализа представляет собой процедуру *параметрического синтеза*. Если модификации параметров целенаправленны и подчинены стратегии поиска наилучшего значения некоторого показателя качества, то процедура параметрического синтеза является *процедурой оптимизации*.

Возможно, что путем параметрического синтеза не удастся добиться приемлемой степени выполнения условий работоспособности. Тогда используют другой путь, связанный с модификациями структуры. Новый вариант структуры синтезируется, для него повторяются процедуры формирования модели и параметрического синтеза.

Если не удается получить приемлемое решение и на этом пути, то ставится вопрос о корректировке ТЗ, сформулированного на предыдущем этапе проектирования. Такая корректировка может потребовать повторного выполнения ряда процедур -го иерархического уровня, что и обуславливает итерационный характер проектирования.

Существует характерная особенность взаимосвязи проектных процедур анализа и синтеза. Эта взаимосвязь имеет характер вложенности процедуры анализа в процедуру оптимизации (параметрического синтеза) и процедуры оптимизации в процедуру синтеза (структурного и параметрического).

Вложенность означает, что:

– анализ входит как составная часть в оптимизацию, а оптимизация – в синтез;

– однократное выполнение процедуры оптимизации требует многократного выполнения процедуры анализа;

– однократное решение задачи синтеза требует многократного решения задачи оптимизации.

Очевидно, такой же характер взаимодействия имеют процедуры анализа: однократный многовариантный анализ основан на многократном одновариантном анализе.

Нетрудно подсчитать, что синтез проектного решения может потребовать чрезмерно большого количества вариантов анализа. Один из путей решения этой проблемы – применение достаточно точных и сложных математических моделей и алгоритмов анализа только на завершающих итерациях синтеза. Для большинства просматриваемых вариантов структуры при этом выполняется лишь ориентировочная оценка на основе косвенных критериев, упрощенных моделей и алгоритмов. Такая оценка позволит без существенных затрат вычислительных ресурсов отсеять большинство неперспективных вариантов и оставить для тщательного анализа малое число вариантов.

# 2 Структурный подход к проектированию информационных систем

## 2.1 Сущность структурного подхода

Сущность структурного подхода к разработке ИС заключается в ее декомпозиции на автоматизируемые функции: т.е. система разбивается на функциональные подсистемы, которые в свою очередь делятся на подфункции, эти подфункции подразделяются на задачи и так далее. Процесс разбиения продолжается вплоть до конкретных процедур. При этом автоматизируемая система сохраняет целостное представление, в котором все составляющие компоненты взаимоувязаны.

Все наиболее распространенные методологии структурного подхода базируются на ряде общих принципов:

– принцип решения сложных проблем путем их разбиения на множество меньших независимых задач, легких для понимания и решения;

– принцип иерархического упорядочивания, т.е. принцип организации составных частей проблемы в иерархические древовидные структуры с добавлением новых деталей на каждом уровне.

– принцип абстрагирования: заключается в выделении существенных аспектов системы и отвлечения от несущественных;

– принцип формализации: заключается в необходимости строгого методического подхода к решению проблемы;

– принцип непротиворечивости: заключается в обоснованности и согласованности элементов;

– принцип структурирования данных: заключается в том, что данные должны быть структурированы и иерархически организованы.

На стадии проектирования ИС модели расширяются, уточняются и дополняются диаграммами, отражающими структуру программного обеспечения: архитектуру ПО, структурные схемы программ и диаграммы экранных форм.

## 2.2 Методология функционального моделирования SADT

Методология SADT (Structured Analysis and Design Technique) представляет собой совокупность методов, правил и процедур, предназначенных для построения функциональной модели объекта какой-либо предметной области. Функциональная модель SADT отображает функциональную структуру объекта, т.е. производимые им действия и связи между этими действиями.

Дальнейшим развитием данной методологии стало появление стандарта IDEF0. IDEF (Integration DEFinition) – семейство совместно используемых методов для процесса моделирования.

Результатом применения методологии SADT является модель, которая состоит из диаграмм, фрагментов текстов и глоссария, имеющих ссылки друг на друга. Диаграммы - главные компоненты модели, все функции ИС и интерфейсы на них представлены как блоки и дуги. Место соединения дуги с блоком определяет тип интерфейса. Управляющая информация входит в блок сверху, в то время как информация, которая подвергается обработке, показывается с левой стороны блока, а результаты выхода – с правой стороны. Механизм (человек или автоматизированная система), который осуществляет операцию, представляется дугой, входящей в блок снизу (рис. 2.1).

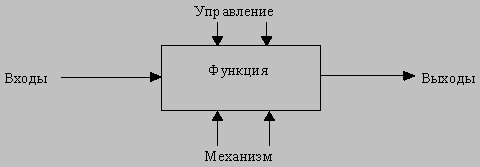


Рис. 2.1 – Функциональный блок и интерфейсные дуги

SADT-модель дает полное, точное и адекватное описание системы, имеющее конкретное назначение. Целью модели является получение ответов на некоторую совокупность вопросов. Эти вопросы неявно присутствуют (подразумеваются) в процессе анализа и, следовательно, они руководят созданием модели и направляют его. Это означает, что сама модель должна будет дать ответы на эти вопросы с заданной степенью точности. Если модель отвечает не на все вопросы или ее ответы недостаточно точны, то мы говорим, что модель не достигла своей цели. Определяя модель таким образом, SADT закладывает основы практического моделирования. Только поняв, насколько хорошо нужно ответить на поставленные вопросы, можно определить, когда процесс моделирования можно считать завершенным (т.е. когда модель будет соответствовать поставленной цели).

Модель является некоторым толкованием системы. Поэтому субъектом моделирования служит сама система. Однако моделируемая система никогда не существует изолированно: она всегда связана с окружающей средой. Причем зачастую трудно сказать, где кончается система и начинается среда. По этой причине в методологии SADT подчеркивается необходимость точного определения границ системы. SADT-модель всегда ограничивает свой субъект, т.е. модель устанавливает точно, что является и что не является субъектом моделирования, описывая то, что входит в систему, и подразумевая то, что лежит за ее пределами. Ограничивая субъект, SADT-модель помогает сконцентрировать внимание именно на описываемой системе и позволяет избежать включения посторонних субъектов. Вот почему утверждается, что SADT-модель должна иметь единственный субъект.

С определением модели тесно связана позиция, с которой наблюдается система и создается ее модель. Поскольку качество описания системы резко снижается, если оно не сфокусировано ни на чем, SADT требует, чтобы модель рассматривалась все время с одной и той же позиции. Эта позиция называется "точкой зрения" данной модели. "Точку зрения" лучше всего представлять себе как место (позицию) человека или объекта, в которое надо встать, чтобы увидеть систему в действии. С этой фиксированной точки зрения можно создать согласованное описание системы так, чтобы в модели не смешивались бы несвязанные описания.

После того как определены субъект, цель и точка зрения модели, начинается первая интеграция процесса моделирования по методологии SADT. Субъект определяет, что включить в модель, а что исключить из нее. Точка зрения диктует автору модели выбор нужной информации о субъекте и форму ее подачи. Цель становится критерием окончания моделирования. Конечным результатом этого процесса является набор тщательно взаимоувязанных описаний, начиная с описания самого верхнего уровня всей системы и кончая подробным описанием деталей или операций системы.

Одной из наиболее важных особенностей методологии SADT является постепенное введение все больших уровней детализации по мере создания диаграмм, отображающих модель.

На рис. 2.2 приведены четыре диаграммы и их взаимосвязи, показана структура SADT-модели. Каждый компонент модели может быть декомпозирован на другой диаграмме. Каждая диаграмма иллюстрирует "внутреннее строение" блока на родительской диаграмме. Дуги, входящие в блок и выходящие из него на диаграмме верхнего уровня, являются точно теми же самыми, что и дуги, входящие в диаграмму нижнего уровня и выходящие из нее, потому что блок и диаграмма представляют одну и ту же часть системы.

Построение SADT-модели начинается с представления всей системы в виде простейшей компоненты - одного блока и дуг, изображающих интерфейсы с функциями вне системы. Поскольку единственный блок представляет всю систему как единое целое, имя, указанное в блоке, является общим. Это верно и для интерфейсных дуг – они также представляют полный набор внешних интерфейсов системы в целом.

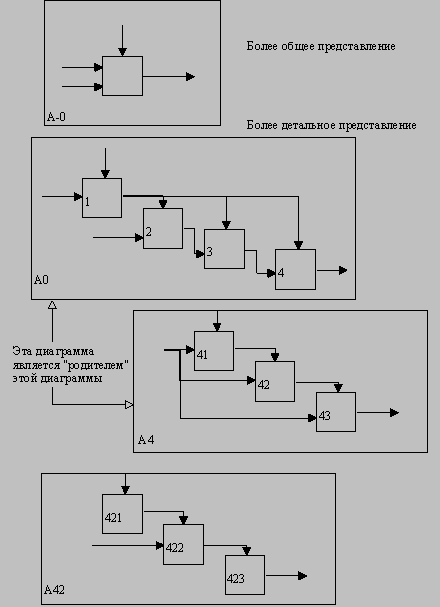


Рис. 2.2 – Структура SADT-модели. Декомпозиция диаграмм

Затем блок, который представляет систему в качестве единого модуля, детализируется на другой диаграмме с помощью нескольких блоков, соединенных интерфейсными дугами. Эти блоки представляют основные подфункции исходной функции. Данная декомпозиция выявляет полный набор подфункций, каждая из которых представлена как блок, границы которого определены интерфейсными дугами. Каждая из этих подфункций может быть декомпозирована подобным образом для более детального представления. Во всех случаях каждая подфункция может содержать только те элементы, которые входят в исходную функцию.

Модель SADT представляет собой серию диаграмм с сопроводительной документацией, разбивающих сложный объект на составные части, которые представлены в виде блоков. Детали каждого из основных блоков показаны в виде блоков на других диаграммах. Каждая детальная диаграмма является декомпозицией блока из более общей диаграммы. На каждом шаге декомпозиции более общая диаграмма называется *родительской* для более детальной диаграммы.

Некоторые дуги присоединены к блокам диаграммы обоими концами, у других же один конец остается не присоединенным. Не присоединенные дуги соответствуют входам, управлениям, механизмам и выходам родительского блока. Источник или получатель этих пограничных дуг может быть обнаружен только на родительской диаграмме.

Каждый блок на диаграмме имеет свой номер. Блок любой диаграммы может быть далее описан диаграммой нижнего уровня, которая, в свою очередь, может быть далее детализирована с помощью необходимого числа диаграмм. Таким образом, формируется иерархия диаграмм.

Для того, чтобы указать положение любой диаграммы или блока в иерархии, используются иерархические номера диаграмм. Например, А21 является диаграммой, которая детализирует блок 1 на диаграмме А2. Аналогично, А2 детализирует блок 2 на диаграмме А0. Номером самого верхнего блока является А-0.

Блоки SADT никогда не размещаются на диаграмме случайным образом. Они размещаются по степени важности, как ее понимает автор диаграммы. В SADT этот относительный порядок называется доминированием. Доминирование понимается как влияние, которое один блок оказывает на другие блоки диаграммы. Например, самым доминирующим блоком диаграммы может быть либо первый из требуемой последовательности функций, либо планирующая или контролирующая функция, влияющая на все другие функции. Наиболее доминирующий блок обычно размещается в верхнем левом углу диаграммы, а наименее доминирующий - в правом нижнем углу. В результате получается "ступенчатая" схема, подобная представленной на рисунке 2.2. Расположение блоков на странице отражает авторское определение доминирования. Таким образом, топология диаграммы показывает, какие функции оказывают большее влияние на остальные. Чтобы подчеркнуть это, SADT-аналитик нумерует блоки в соответствии с порядком их доминирования.

В методологии SADT требуется только пять типов взаимосвязей между блоками для описания их отношений: управление, вход, обратная связь по управлению, обратная связь по входу, выход-механизм. Связи по управлению и входу являются простейшими, поскольку они отражают прямые воздействия, которые интуитивно понятны и очень просты.

Отношение управления возникает тогда, когда выход одного блока непосредственно влияет на блок с меньшим доминированием. Например, блок А1 влияет на блоки А2 и А3. Отношение входа возникает тогда, когда выход одного блока становится входом для блока с меньшим доминированием, например, выход блока А2 становится входом функции А3.

Обратная связь по управлению и обратная связь по входу являются более сложными, поскольку они представляют итерацию или рекурсию, т.е. выходы из одной функции влияют на будущее выполнение других функций, что впоследствии влияет на исходную функцию. Обратная связь по управлению возникает тогда, когда выход некоторого блока влияет на блок с большим доминированием. Связь по входной обратной связи имеет место тогда, когда выход одного блока становится входом другого блока с большим доминированием.

Связи "выход-механизм" встречаются нечасто и представляют особый интерес. Они отражают ситуацию, при которой выход одной функции становится средством достижения цели для другой. Связи "выход-механизм" характерны при распределении источников ресурсов (например, требуемые инструменты, обученный персонал, физическое пространство, оборудование, финансирование, материалы).

Разветвления дуг, изображаемые в виде расходящихся линий, означают, что все содержимое дуг или его часть может появиться в каждом ответвлении дуги. Дуга всегда помечается до разветвления, чтобы дать название всему набору. Кроме того, каждая ветвь дуги может быть помечена или не помечена в соответствии со следующими правилами:

– непомеченные ветви содержат все объекты, указанные в метке дуги перед разветвлением (т.е. все объекты принадлежат этим ветвям);

– ветви, помеченные после точки разветвления, содержат все объекты или их часть, указные в метке дуги перед разветвлением (т.е. каждая метка ветви уточняет, что именно содержит ветвь).

Слияние дуг в SADT, изображаемое как сходящиеся вместе линии, указывает, что содержимое каждой ветви идет на формирование метки для дуги, являющейся результатом слияния исходных дуг. После слияния результирующая дуга всегда помечается для указания нового набора объектов, возникшего после объединения. Кроме того, каждая ветвь перед слиянием может помечаться или не помечаться в соответствии со следующими правилами:

– непомеченные ветви содержат все объекты, указанные в общей метке дуги после слияния (т.е. все объекты исходят из всех ветвей);

– помеченные перед слиянием ветви содержат все или некоторые объекты из перечисленных в общей метке после слияния (т.е. метка ветви ясно указывает, что содержит ветвь).

Хорошая методология структурного анализа, позволяющая создавать отдельные диаграммы, должна гарантировать правильное соединение всех диаграмм для образования согласованной модели. SADT-диаграммы имеют внешние дуги – дуги, как бы выходящие наружу и ведущие к краю страницы. Эти дуги являются интерфейсом между диаграммой и остальной частью модели. SADT требует, чтобы все внешние дуги диаграммы были согласованы с дугами, образующими границу этой диаграммы. Другими словами, диаграмма должна быть "состыкована" со своей родительской диаграммой. Обычно это означает, что внешние дуги согласованы по числу и наименованию (но не обязательно по расположению) с дугами, касающимися декомпозированного блока родительской диаграммы.

В SADT принята система обозначений, позволяющая аналитику точно идентифицировать и проверять связи по дугам между диаграммами. Эта схема кодирования дуг –"ICOM" - получила название по первым буквам английских эквивалентов слов вход (Input), управление (Control), выход (Output), механизм (Mechanism). Коды ICOM чрезвычайно эффективны, поскольку они позволяют аналитику быстро проверять согласованность внешних дуг диаграммы с граничными дугами соответствующего блока родительской диаграммы.

Они также обеспечивают согласованность декомпозиции, поскольку все дуги, входящие в диаграмму и выходящие из нее, должны быть учтены. Одним из способов такой стыковки может служить присваивание кодов ICOM внешним дугам новой диаграммы согласно следующим правилам:

– представьте себе рисунок новой диаграммы внутри разлагаемого блока. Продлите внешние дуги почти до края диаграммы. Зрительно соедините каждую внешнюю дугу диаграммы с соответствующей граничной дугой декомпозируемого блока.

– присвойте код каждой зрительной связи. Используйте I для входных дуг, С - для связей между дугами управления, О - для связей между выходными дугами, М - для связей между дугами механизма.

– добавьте после каждой буквы цифру, соответствующую положению данной дуги среди других дуг того же типа, касающихся родительского блока. Причем входные и выходные дуги пересчитываются сверху вниз, а дуги управлений и механизмов пересчитываются слева направо. Теперь запишите каждый код около окончания каждой внешней дуги.

**Пример**

Книжный магазин занимается продажей художественной литературы. В торговом зале книги располагаются на стеллажах, каждый из которых имеет свой номер и тематику. Любой клиент может обратиться к продавцу-консультанту и получить подробную информацию о книгах, имеющихся в магазине, их цене и заказать интересующие книги, которых нет в наличии. В торговом зале имеется касса, где осуществляется непосредственная продажа книги. Книги, которые не выставляются в торговый зал, хранятся на складе. Исполнительный директора работает с книжными издательствами по поставкам партий книг, которые водитель доставляет на склад, а также занимается координацией действий между торговым залом и складом. Отдел бухгалтерии осуществляет бухгалтерский и кадровый учет. За чистотой на складе и в торговом зале следит уборщик. Директор, с помощью секретаря, руководит всей работой в магазине.

Цель информационной системы книжного магазина – автоматизация деятельности книжного магазина.

Задачи данной информационной системы:

1. Автоматизация регистрации данных о клиенте. Необходимо, чтобы данные о клиенте сохранялись в нашей ИС. Эти данные используются для идентификации клиента и для связи сотрудниками магазина с ним.

2. Автоматизация регистрации заявки клиента на книгу. Необходимо хранить все сведения по заявкам клиентов на поставку книг. Эти сведения используются как для оформления общего заказа партии книг издательству-поставщику, так и для дальнейшего анализа текущего спроса на книги.

3. Автоматизация складского учета. Необходимо в ИС заносить факты закупок и продаж книг, составлять общий заказ на поставку партии книг издательству-поставщику.

4. Автоматизация бухгалтерского учета. Необходимо в ИС вести бухгалтерский учет согласно действующему законодательству.

Информационная система книжного магазина состоит из четырех подсистем (см. приложения А–В).

Подсистема А1 «Подсистема регистрации данных о клиенте» предназначена для ввода, хранения и выдачи сведений о клиенте. Персональные данные клиента вводятся администратором в ИС с помощью клавиатуры. В ИС можно поставить ограничения на регистрацию нового клиента (например, по месту жительства). Введенные данные клиента используются в дальнейшем для оформления заявки клиента на книгу.

Подсистема А2 «Подсистема регистрации заявки клиента на книгу» предназначена для оформления заявки клиента на книги, которые имеются в каталоге книг, но отсутствуют в нашем магазине. Для полноценного оформления заявки нужно получить данные о клиенте. Оформленная заявка клиента используется для составления общего заказа издательству на поставку партии книг.

Подсистема А3 «Подсистема автоматизированного складского учета» состоит из трех модулей. В модуле А31 «Модуль оформления общего заказа на поставку партии книг» исполнительный директор, анализируя заказы клиентов и текущий спрос по продажам, оформляет заказ поставщику на поставку партии книг. Естественным ограничителем выступает сумма оборотных средств магазина, которую можно потратить на данный заказ. Результатом данного заказа является обновленный каталог книг.

С помощью модуля А32 «Модуль оприходование партии книг» заведующий складом должен зарегистрировать в ИС доставленные в магазин книги. Факт регистрации книги должен быть учтен при дальнейшем ее списании. Само списание книги можно с помощью модуля А33 «Модуль списания книги». Для списания книги есть два основания: непосредственная продажа и порча книги. Информация по текущему наличию книг учитывается как при оформлении следующего заказа на партию книг, так и при оформлении заявки клиента.

Бухгалтерия, работая с подсистемой А4 «Подсистема автоматизированного бухгалтерского учета», ведет автоматизированный бухгалтерский учет книжного магазина. Так как самостоятельная разработка этой подсистемы нецелесообразна, то можно воспользоваться готовой системой от компании 1С «1С: Предприятие» с конфигурацией «1С: Бухгалтерия».

Список пользователей ИС книжного магазина:

– администратор;

– исполнительный директор;

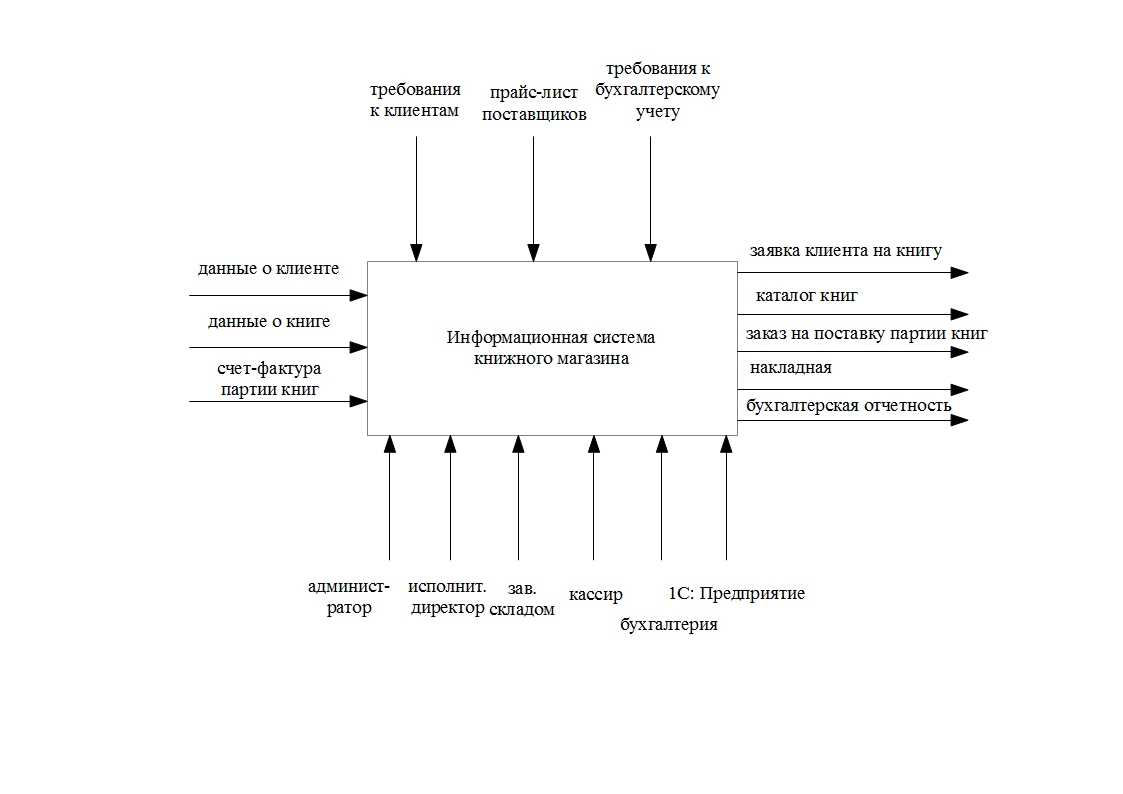
– заведующий складом;

– кассир;

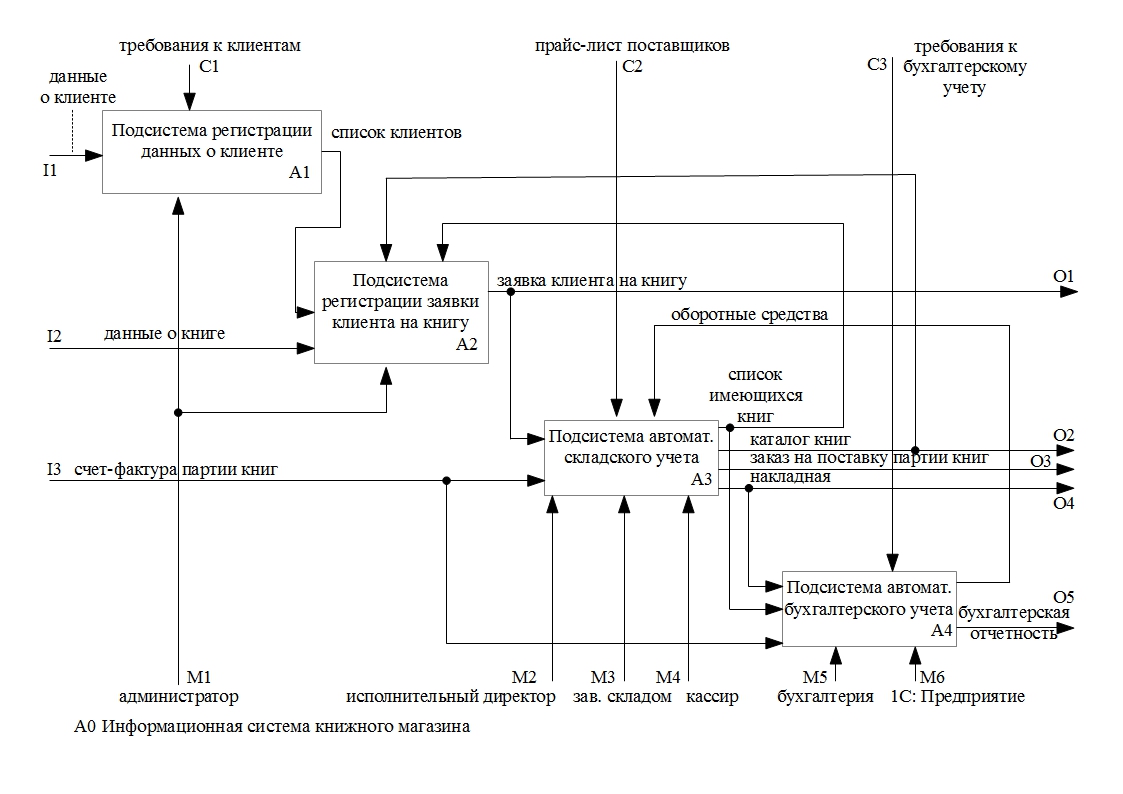
– весь отдел бухгалтерии (главный бухгалтер и бухгалтер).

Для полноценной работы ИС необходимо приобрести программный продукт «1С: Предприятие» с конфигурацией «1С: Бухгалтерия» (2 лицензии, сетевая версия).

Приложение А. Диаграмма А-0 «Информационная система книжного магазина»



Приложение Б. Диаграмма А0 «Информационная система книжного магазина»



Приложение В. Диаграмма А3 «Подсистема автоматизированного складского учета»

