**Лабораторная работа № 4**

**Тема работы: «Исследование структуры функциональной модели (IDEF0) на основе средств моделирования»**

**1. Цель работы**

Сформировать умения анализировать структуру функциональной модели (IDEF0). Сформировать умения составлять контекстную диаграмму.

**2. Задание**

Построить структуру функциональной модели разрабатываемой программной системы в соответствии с требованиями, изложенными в техническом задании. Построить контекстную диаграмму.

**3. Оснащение работы**

Техническое задание, ЭВМ, CASЕ-средство Bpwin для построения модели.

**4. Основные теоретические сведения**

Одним из средств, используемых на этапе структурного анализа, являются средства, иллюстрирующие функции, выполняемые системой. Эти средства соответствуют определенным видам моделей (диаграмм).

Наиболее распространенными являются IDEF0 – модели и соответствующие функциональные диаграммы (Integrated DEFenition).

Наиболее известной реализацией IDEF0 является методология SADT, разработанная Дугласом Россом. Основная идея методологии SADT – это построение древовидной функциональной модели.

Сначала функциональность описывается в целом – это называется контекстной диаграммой. При создании контекстной диаграммы формулируется цель моделирования, область (т.е., что будет рассматриваться, как компонент системы, а что как внешнее воздействие) и позиция, в соответствии с которой будет строиться модель.

Методология SADT представляет собой совокупность методов правил и процедур, предназначенных для построения функциональной модели объекта какой-либо предметной области. Такая модель отображает функциональную структуру объекта, т.е. производимые им действия и связи между этими действиями. Основные элементы этой методологии основаны на следующих концепциях:

Функции отображаются в виде блоков, а интерфейсы представлены дугами (стрелками), входящими в блок и выходящими из него.

При этом должны выполняться следующие правила:

– количество блоков на каждом уровне декомпозиции должно быть ограничено (как правило 3-6);

– связность диаграмм реализуется при помощи нумерации блоков;

– метки и наименования должны быть уникальными;

– соблюдение синтаксических правил для графики (блоков, дуг);

– правило определения роли данных (разделение входов и управлений).

В результате должна быть создана модель, состоящая из диаграмм, фрагментов текста, глоссария (справочника), имеющих ссылки друг на друга.

Главными компонентами модели являются диаграммы, на которых функции определяются блоками, а интерфейсы дугами. Место соединения дуги с блоком определяет тип интерфейса. Управляющая информация входит в блок сверху, информация, которая подвергается обработке – слева, результирующая информация показана справа, а механизм воздействия, осуществляющий операцию, определен дугой, входящей в блок снизу. Механизм может быть человеком, компьютером или др. устройством, помогающим выполнить данную функцию (рис. 3.1, 3.2).

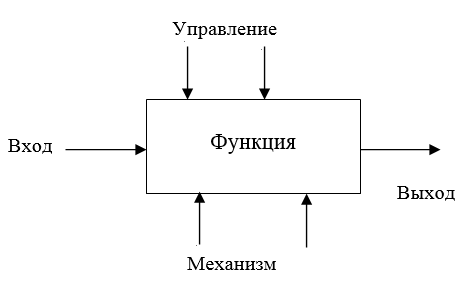


Рисунок 5.1 – Общий вид изображения функционального блока

Например:

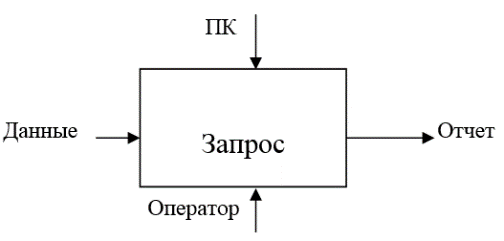


Рисунок 5.2 – Пример изображения функционального блока

Каждый компонент модели может быть декомпозирован на другой диаграмме, которая будет иллюстрировать внутреннее строение блока родительской диаграммы

Построение модели начинается с построения контекстной диаграммы, которая будет представлена одним блоком, представляющим систему в целом и интерфейсными дугами, представляющими связи с функциями за пределами системы. Имя, которое будет указано в блоке, является общим.

Затем система будет детализироваться на другой диаграмме в виде нескольких блоков, описывающих основные подфункции данной функции.

В результате декомпозиции будет выявлен полный набор подфункций, каждая из которых представляется, как блок, границы которого определены интерфейсными дугами.

Каждая детальная диаграмма является декомпозицией блока на более общей диаграмме, которая называется родительской.

Входные и выходные дуги для блока верхнего уровня являются входными и выходными дугами для диаграммы более низкого уровня, так как они представляют один и тот же компонент системы.

Например:

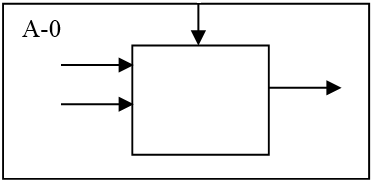


Рисунок 5.3 – Более общее представление (А-0 Контекстная диаграмма)

# 

Рисунок 5.4 – Более детальное представление (А0 диаграмма)

А0 является родительской диаграммой для диаграммы А4, а блок 4 является родительским (т.е. детализируемым) блоком.

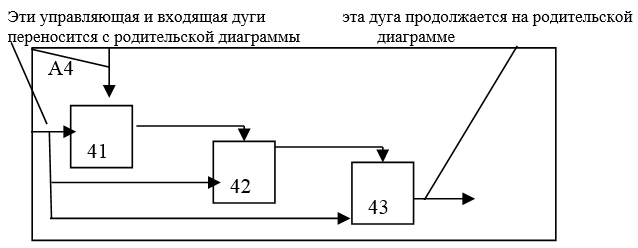


Рисунок 5.5 – Детализация блока 4 диаграммы А0

На диаграммах явно не указывается ни последовательность, ни время. Обратные связи, итерации, продолжающиеся процессы и перекрывающиеся по времени функции могут быть изображены в виде дуг, а обратные связи могут быть представлены комментариями, замечаниями и т.д.

Каждый блок на диаграмме имеет свой номер и может быть далее подробно описан диаграммой нижнего уровня, которая, в свою очередь, может быть детализирована при помощи необходимого числа диаграмм.

Рассмотрим пример построения иерархии функциональных диаграмм для программы ведения БД, в которой хранится информация о книгах, имеющихся в библиотеке.

Вначале строится контекстная диаграмма А-0, описывающая систему в целом.

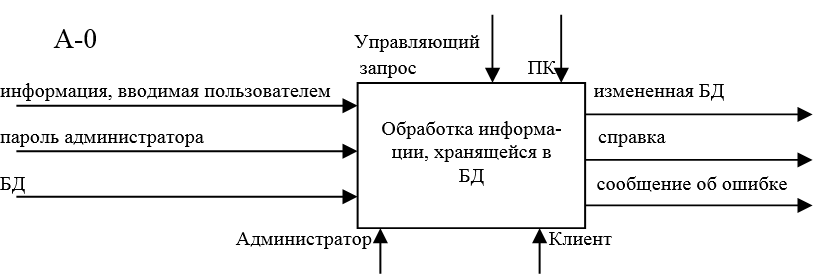


Рисунок 5.6 – Контекстная диаграмма

Затем строится диаграмма А0, на которой показываются основные подфункции разрабатываемой системы.

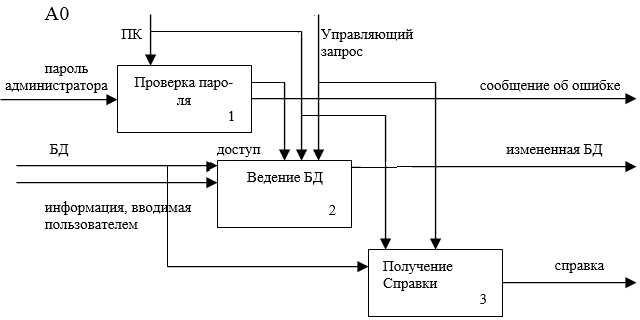


Рисунок 5.7 – Декомпозиция на основные подфункции

Каждый из блоков диаграммы А0 может быть детализирован на диаграммах более низкого уровня.

Например, при детализации блока 2 диаграммы А0 получим детализирующую диаграмму А2.

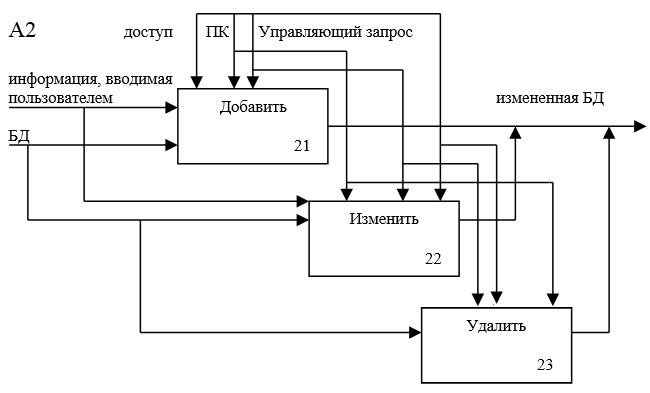
**

Рисунок 5.8 – Детализация блока 2 диаграммы А0

Состав диаграмм в каждом конкретном случае зависит от необходимой полноты описания разрабатываемой программной системы.

**5. Порядок выполнения работы**

1. Загрузить инструментальное средство Bpwin или Ramus.

2. Изучить теоретический материал в файле «Основы работы с CASE-средством» и проделать задание по построению контекстной диаграммы.

3. Определить структуру функциональной модели разрабатываемой системы по выбранной теме.

4. Построить контекстную диаграмму А-0.

**6. Форма отчета о работе**

*Лабораторная работа № \_\_\_*

*Номер учебной группы \_\_\_\_\_\_\_\_\_\_\_\_*

*Фамилия, инициалы учащегося*

*Дата выполнения работы*

*Тема работы:*

*Цель работы:*

*Оснащение работы:*

*Результат выполнения работы:*

**7. Контрольные вопросы и задания**

1. Изложите принципы структурного подхода к разработке программного обеспечения.

2. Охарактеризуйте методологию IDEF0-моделирования.

3. Что такое контекстная диаграмма?

4. Из чего состоит функциональный блок?

5. Что такое жизненный цикл программного продукта?

6. Дайте определение модели жизненного цикла ПО.

7. Приведите этапы разработки программного средства.

8. Для чего строят модели AS-IS и TO-BE?

9. Что такое бизнес-процесс?

**8. Рекомендуемая литература**

**Маклаков, С. В.** Bpwin и Erwin. CASE-средства разработки информационных систем / С. В. Маклаков. М.: Диалог-МИФИ, 2001.

**Маклаков, С. В.** Создание информационных систем с All Fussion Modeling Suite / С. В. Маклаков. М.: Диалог-МИФИ, 2003.

**Рудаков, А. В.** Технология разработки программных продуктов: учебное пособие для студентов учреждений среднего профессионального образования / А. В. Рудаков, Г. Н. Федорова. – 4-е изд., стер. – М.: Издательский центр «Академия»; 2014.