

ДИАГРАММА ПОТОКОВ ДАННЫХ

Цель работы: изучение основ структурного моделирования процессов.

Моделирование процессов систем в структурном анализе

Моделирование процесса касается графического представления функций, которые позволяют получать, обрабатывать, хранить и распространять данные между системой и ее окружением, а также между компонентами внутри системы. Распространенной формой такого описания являются диаграммы потока данных (Data Flow Diagrams, DFD).

Диаграммы потока данных – это традиционный метод моделирования процессов в анализе и дизайне систем, считающийся одним из основных инструментов, применяемых до широкого распространения унифицированного языка моделирования создания абстрактных моделей систем UML (Unified Modeling Language).

Основная цель построения диаграммы в данной методологии: визуализация процесса передачи объектов (информации) между участниками этого процесса, т.е. демонстрация того, как каждый элемент схемы преобразует свои входные данные в выходные и как связаны между собой эти элементы. Диаграмма наглядно демонстрирует пути, по которым циркулируют данные внутри проектируемой информационной системы, а также между пути следования информации между системой и внешним миром.

Набор взаимосвязанных и согласованных DFD диаграмм позволяет составить подробный список ожидаемых результатов при изучении и документировании процессов системы:

1. **Контекстная диаграмма** - показывает широту охвата системы, указывая, какие элементы находятся внутри, а какие вне ее. На ней изображены все заинтересованные внешние сущности, общающиеся с системой, сам процесс, отображающий ее целиком и основные потоки данных между системой и окружающим миром
2. **Диаграммы потоков данных системы** - определяют процессы перемещения и преобразования данных, воспринимающие входы и создающие выходы.

3. Полное описание каждого компонента диаграммы - данные для всех указанных объектов, включаются в словарь проекта или репозиторий CASE.

Таким образом, DFD диаграммы эволюционируют от более общего к более подробному описанию, по мере формирования лучшего понимания требований к системе.

Уровни представления DFD

Моделирование процессов начинается с обзорной DFD диаграммы, в которой кратко описывается проектируемая система – так называемый **верхний контекстный уровень**.

Логическая диаграмма потока данных, как следует из названия, рассказывает о событиях и данных, генерируемых в результате каждого такого события.

Тогда как **физическая диаграмма** потока данных описывает то, как должен быть представлен поток информации в системе.

Совместно логическая и физическая диаграммы могут полностью визуализировать состояние информационной системы.

Таким образом, любая диаграмма DFD отображает поток информации для процесса или системы, тогда как логическая составляющая позволяет понять «что» происходит, а физическая – «как» это происходит. Это две разные точки зрения на один и тот же поток данных, каждая из которых предназначена для визуализации и уточнения системы:

- Логический уровень DFD «as is» фиксирует текущие и необходимые действия, требуемые для процесса.
- Логический уровень DFD «to be» моделирует новый набор действий и функций.
- Физический уровень DFD «as is» отображает текущие ресурсы (программное обеспечение, оборудование, базы данных, людей и т.д.) для выполнения действий.
- Физический уровень DFD «to be» моделирует новую реализацию системы.

Нотации и элементы, используемые при DFD-моделировании

Диаграммы потоков данных стали известны широкой публике с конца 1970-х

годов благодаря книге «Структурное проектирование» Эда Йордана и Ларри Константина («Structured Design» Yourdon & Constantine, 1974).

Наиболее распространенные нотации (системы символов):

1. Peter Coad and Ed Yourdon (Coad and Yourdon methodology) – методология Коада и Йордана.
2. Ed Yourdon and Tom DeMarco (Yordon-DeMarco notation) – нотация Йордона-ДеМарко.
3. Chris Gane and Trish Sarson (Gene-Sarson DFD Symbols or notation) – нотация Гейна-Сарсона.
4. SSADM (Structured System Analysis and Design Methodology).

Так же, как и диаграммы IDEFO, диаграммы потоков данных моделируют систему как набор действий, соединенных друг с другом стрелками, иллюстрирующими не отношения между этими действиями, а реальное перемещение объектов (включая данные) между ними.

Методология диаграмм потоков данных (DFD) состоит из четырех элементов: внешних сущностей, процессов, хранилищ данных и потоков данных.

Однако, перечисленные элементы представляют различные точки зрения в логических и физических DFD, а также могут иметь разное визуальное оформление в зависимости от выбранной нотации. Например:

- в логических DFD процессами являются бизнес-операции, а в физических DFD процессами являются программы, ручные процедуры или другие способы обработки информации.
- в логических DFD хранилища данных представляют собой наборы информации, независимо от того, как они хранятся, а в физических DFD хранилищами данных являются базы данных, компьютерные файлы и бумажные файлы.

Синтаксис и семантика диаграмм потоков данных

В зависимости от используемой нотации графическое представление объектов элементов диаграмм (табл. 1) будет различным, но смысл понятий одинаковым.

Процесс (англ. Process / Activity) - функция обработки данных. Это элемент методологии DFD, который представляет собой преобразование входных потоков в

выходные в соответствии с определенным алгоритмом или последовательность действий, которую нужно предпринять, чтобы данные (входные в выходные) были обработаны. Его основные характеристики:

- Процесс именуется в виде словосочетания с активным глаголом в неопределенной форме, за которым следует существительное в винительном падеже, например: «Создать клиента» (а не «Создание клиента») или «Обработать заказ» (а не «Проведение заказа»).
- Процесс отличается от системы/подсистемы по полю наименование.

Графическое представление элемента DFD диаграммы «Процесс» в наиболее распространенных нотациях представлено на рисунке 1.

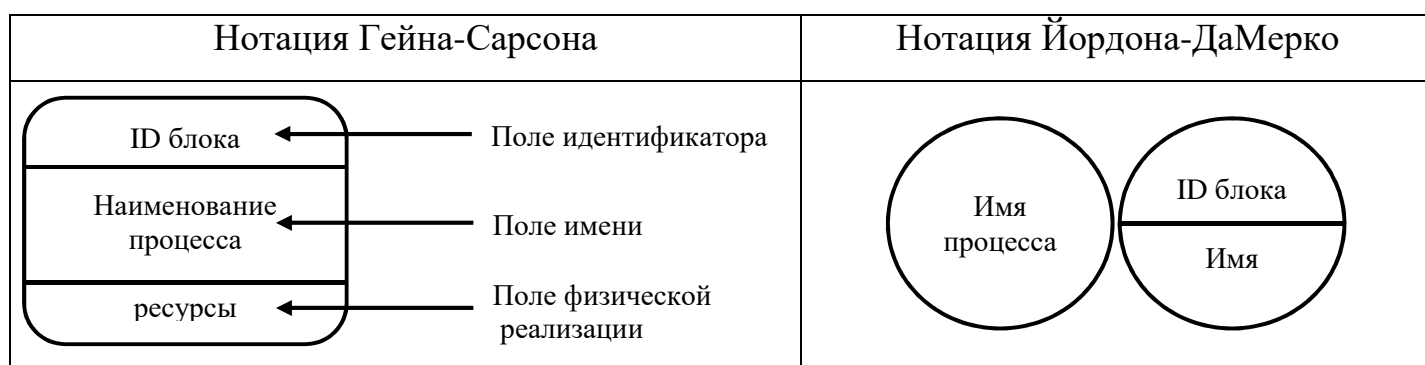


Рис.1 Элемент DFD диаграммы «Процесс».

Как и действия IDEF3, функциональные блоки DFD имеют входы и выходы, но не имеют управления и механизма исполнения как IDEFO. В некоторых интерпретациях нотации DFD механизмы исполнения IDEFO указываются как ресурсы и изображаются в нижней части фигуры.

Внешние сущности (англ. External Entity / External Reference). Это любые объекты, которые не входят в саму систему, но являются для нее источником либо получателем какой-либо информации после обработки данных. Это может быть человек, внешняя система, какие-либо внешние носители информации и хранилища данных. Одна и та же внешняя сущность может быть использована многократно на одной или нескольких диаграммах, одновременно предоставлять входы (функционируя как поставщик) и принимать выходы (функционируя как получатель).

Так как источники и приемники информации находятся за пределами моделируемой системы, многие из их характеристик опускаются. В частности, не рассматривается:

- Взаимодействие, которое происходит между источниками и приемниками.

- Что делает с информацией внешняя сущность и как она устроена.
- Как управлять или изменять внешнюю сущность, потому что с точки зрения системы данные, получаемые или обеспечиваемые ими, являются фиксированными.
- Как обеспечить источнику или приемнику прямой доступом к элементам хранения данных, потому что, как внешние агенты, они не могут непосредственно манипулировать потоками внутри системы.

Графическая интерпретация элемента DFD диаграммы «Внешняя сущность» в наиболее распространенных нотациях представлена на рисунке 2.



Рис.2 Элемент DFD диаграммы «Внешняя сущность».

Хранилище данных (англ. Data store). Внутреннее хранилище данных, которые поступили в систему или являются результатом обработки процессов в системе. В то время как потоки представляют объекты в процессе их передвижения, хранилища данных моделируют их во всех остальных промежуточных состояниях.

Графическая интерпретация элемента DFD диаграммы «Хранилище данных» в наиболее распространенных нотациях представлена на рисунке 3.

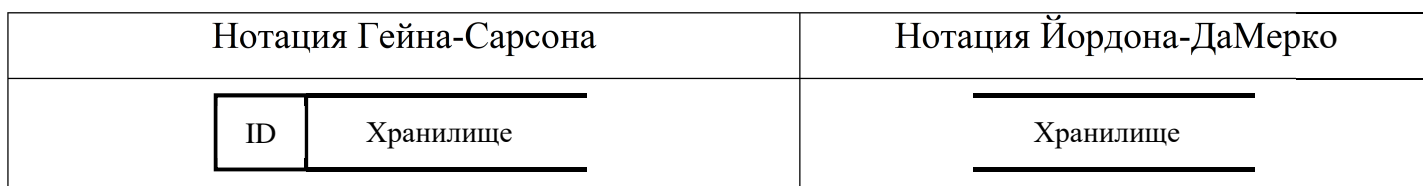


Рис.3 Элемент DFD диаграммы «Хранилище данных».

Поток данных (англ. Data flow). В нотации отображается в виде стрелок, которые показывают, какая информация входит, а какая исходит из того или иного блока на диаграмме.

Стрелки описывают передвижение (поток) объектов от одной части системы к другой. Поскольку все стороны обозначающего функциональный блок DFD прямоугольника равнозначны (в отличие от IDEFO), стрелки могут начинаться и заканчиваться в любой его части.

Графическое представление потока данных представлено на рисунке 4.

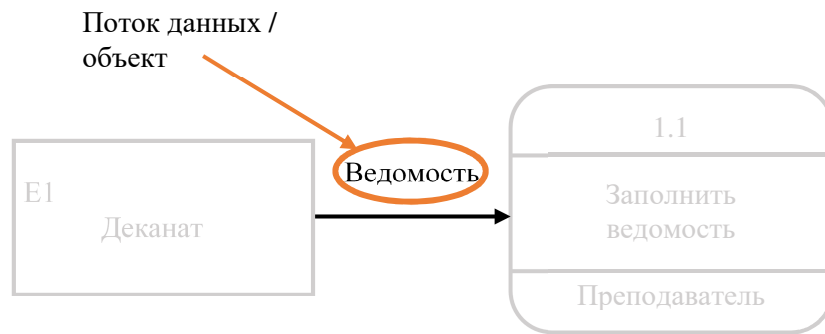


Рис.4 Элемент DFD диаграммы «Поток данных».

Декомпозиция DFD диаграмм.

Уровни DFD - модели можно наглядно представить в виде схемы на рисунке 5:



Рис.5 Схема представления уровней DFD диаграммы.

Т.е. декомпозиция проводится аналогичным с рассмотренными ранее нотациями IDEF0 и IDEF3 образом с иерархической нумерацией функциональных блоков:

1. Уровень 0 –0
2. Уровень 1 –1, 2, 3, ...
3. Уровень 2 –1.1, 1.2, ..., 2.1, 2.2, ...
4. Уровень 3 –1.1.1, 1.1.2, ..., 1.2.1, 1.2.2, ...
5. и т.д.

В DFD каждый номер функционального блока может включать в себя префикс, например А (англ. Activity), номер родительской диаграммы и собственно номер объекта, который уникальным образом идентифицирует его на диаграмме. Обозначения присваиваются также каждому хранилищу данных и каждой внешней сущности вне зависимости от их расположения на схеме. Причем в первом случае вместе с уникальным номером употребляется префикс D (англ. Data Store), а во втором префикс E (англ. External entity).

В таблице 1 представлен алгоритм задания нумерации разным элементам DFD-диаграммы.

Таблица 1. Нумерация элементов DFD диаграмм.

Системы и подсистемы	Функции и процессы
<div style="border: 1px solid black; border-radius: 10px; padding: 10px; margin: 10px;"> <div style="border-bottom: 1px solid black; padding-bottom: 5px;">A1</div> <div style="border-bottom: 1px solid black; padding: 5px 0 5px 20px;">Функция</div> </div>	<div style="border: 1px solid black; border-radius: 10px; padding: 10px; margin: 10px;"> <div style="border-bottom: 1px solid black; padding-bottom: 5px;">A2.1</div> <div style="border-bottom: 1px solid black; padding: 5px 0 5px 20px;">Функция</div> </div>
[Префикс] + собственный номер	[Префикс] + номер родительской подсистемы + собственный номер
Внешние сущности	Хранилища данных
<div style="border: 2px solid black; padding: 10px; margin: 10px;"> <div style="border-bottom: 1px solid black; padding-bottom: 5px;">E1</div> <div style="padding: 5px 0 5px 20px;">Внешняя сущность</div> </div>	<div style="border: 1px solid black; padding: 10px; margin: 10px;"> <div style="border-right: 1px solid black; padding-right: 10px; display: inline-block;">D1</div> <div style="padding: 0 10px 0 20px;">Хранилище</div> </div>
[Префикс] + номер	[Префикс] + номер

Стрелки на DFD-диаграммах могут быть разбиты (разветвлены) на части, и при этом каждый получившийся сегмент может быть переименован таким образом, чтобы показать декомпозицию данных, переносимых потоком. Пример декомпозиции потока представлен на рис. 6.

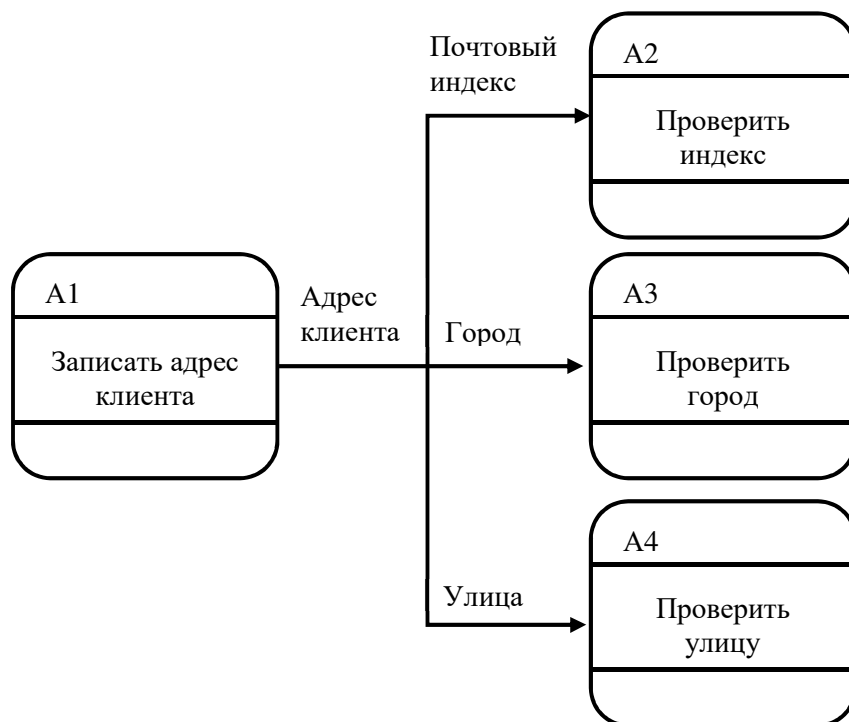


Рис.6 Пример декомпозиции данных.

Аналогично связи могут соединяться между собой для формирования так называемых комплексных объектов. Пример такого объединения приведен на рис. 7.

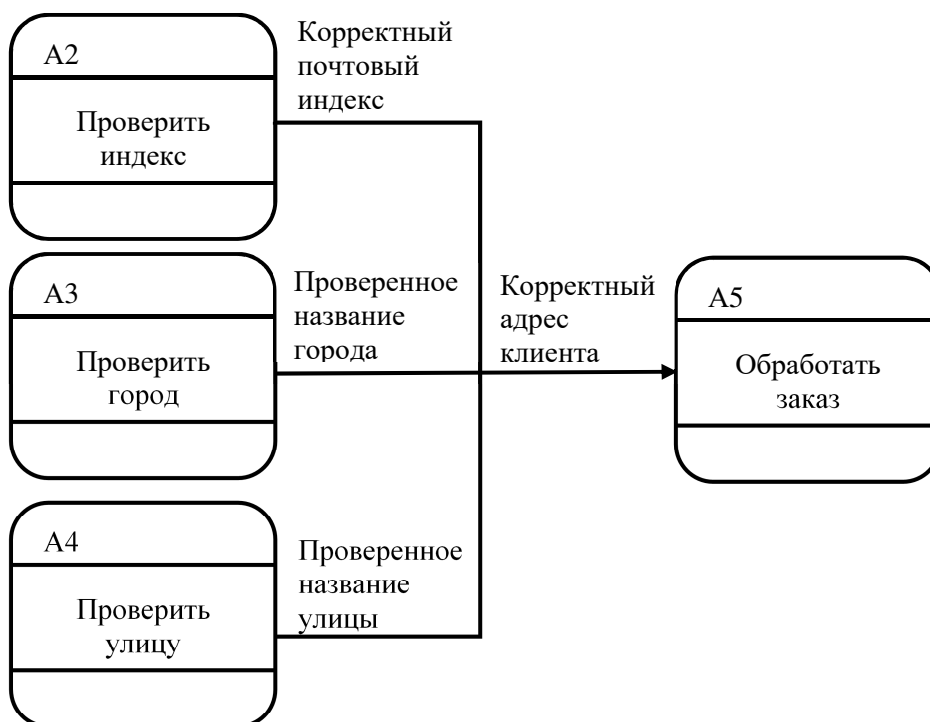


Рис.7 Пример объединения потоков.

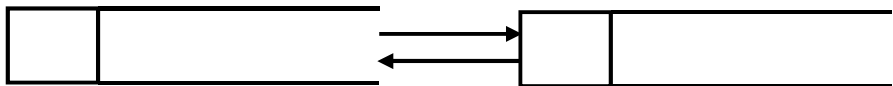
Распространенные ошибки при использовании методологии DFD

В следующей таблице приведены распространенные ошибки допускаемые при построении диаграмм потоков данных.

Таблица 2. Ошибки при построении DFD диаграмм.

У процесса есть выходящие потоки, но нет входящих	
Хранилище и внешний источник связаны напрямую	
Поток идет напрямую в двух направлениях	

Хранилища связаны
напрямую



Примечание:

- модель должна быть простой и легко читаемой;
- хранилище данных должно быть связано хотя бы с одним процессом;
- поток данных не должен существовать между двумя внешними объектами без прохождения процесса;
- процесс с входом, но без выхода считается процессом черной дыры;
- метки процессов должны быть глагольными фразами; хранилища данных представлены существительными;
- внешний объект должен быть связан хотя бы с одним процессом;
- нумерация не обязательно указывает порядок выполнения процессов и полезна для их идентификации при обсуждении с пользователями.
- хранилище данных не должно быть подключено к внешнему объекту, в противном случае это означает, что ему предоставляется прямой доступ к файлам данных.

Выводы

Диаграммы потоков данных (DFD) обеспечивают удобный способ описания передаваемой информации как между частями моделируемой системы, так и между системой и внешним миром. Это качество определяет область применения DFD — они используются для создания моделей информационного обмена и широко применяются при построении информационных систем.

Рассмотрим применение описанного подхода на контрольном примере ИС ВУЗа.

Информационная система ВУЗа

В данной информационной системе описана структура высшего учебного заведения. Данная структура включает в себя объекты, которые взаимодействуют между собой. Декан, возглавляющий факультет, контролирует учебный процесс. Факультет может включать в себя несколько кафедр. В свою очередь в состав кафедр может входить несколько групп студентов по определённым направлениям. Также в учебном процессе участвуют следующие категории преподавателей:

- ассистенты;
- преподаватели;
- старшие преподаватели;
- доценты;
- профессора.

Преподаватели, старшие преподаватели и ассистенты могут быть аспирантами. Профессора, доценты и старшие преподаватели могут возглавлять научные темы, а профессора еще организовывать научные направления. Преподаватели любой категории могли защитить кандидатскую, доценты и профессора – докторскую диссертацию. Преподаватели могут работать на должности доцента или профессора, если имеются соответствующие ученые звания.

В соответствии с учебным планом выстраивается учебный процесс. Учебный план устанавливает для студентов каждого года набора: дисциплины, которые преподаются на курсах, в каждом семестре, количество часов работы на каждый тип занятий и формы контроля.

Занятия подразделяются на следующие типы:

- лекция;
- семинар;
- лабораторная работа;
- консультация;
- курсовая работа;
- исследовательская работа и т.д.

Формы контроля включают в себя:

- зачет;
- дифференцированный зачет;
- экзамен.

Перед началом каждого семестра сотрудники деканата раздают каждой кафедре учебные поручения, где указано, какие кафедры должны вести в данном семестре, какие дисциплины и в каких группах. По учебным поручениям на кафедрах распределяют нагрузки, учитывая категории преподавателей. Ассистент не может читать лекции, а профессор не может вести лабораторные работы. По одной и той же дисциплине разные виды занятий может вести один или несколько преподавателей. Преподаватель может вести занятия по одной или нескольким дисциплинам в различных группах, не относящихся к его кафедре или факультету. В деканате хранятся сведения о зачетах и экзаменах, которые были проведены.

В конце обучения студент должен защитить дипломную работу. Руководителем дипломной работы может быть назначен преподаватель с той же кафедры и факультета, что и студент. За одним руководителем может быть закреплено несколько студентов.

Информационная система позволяет вывести: список и количество студентов определенных групп, сдавших экзамен или зачет по заданной дисциплине с определенной оценкой.

Проведем моделирование в нотации DFD процессов, выявленных ранее при анализе требований к информационной системе с точки зрения роли администратора.

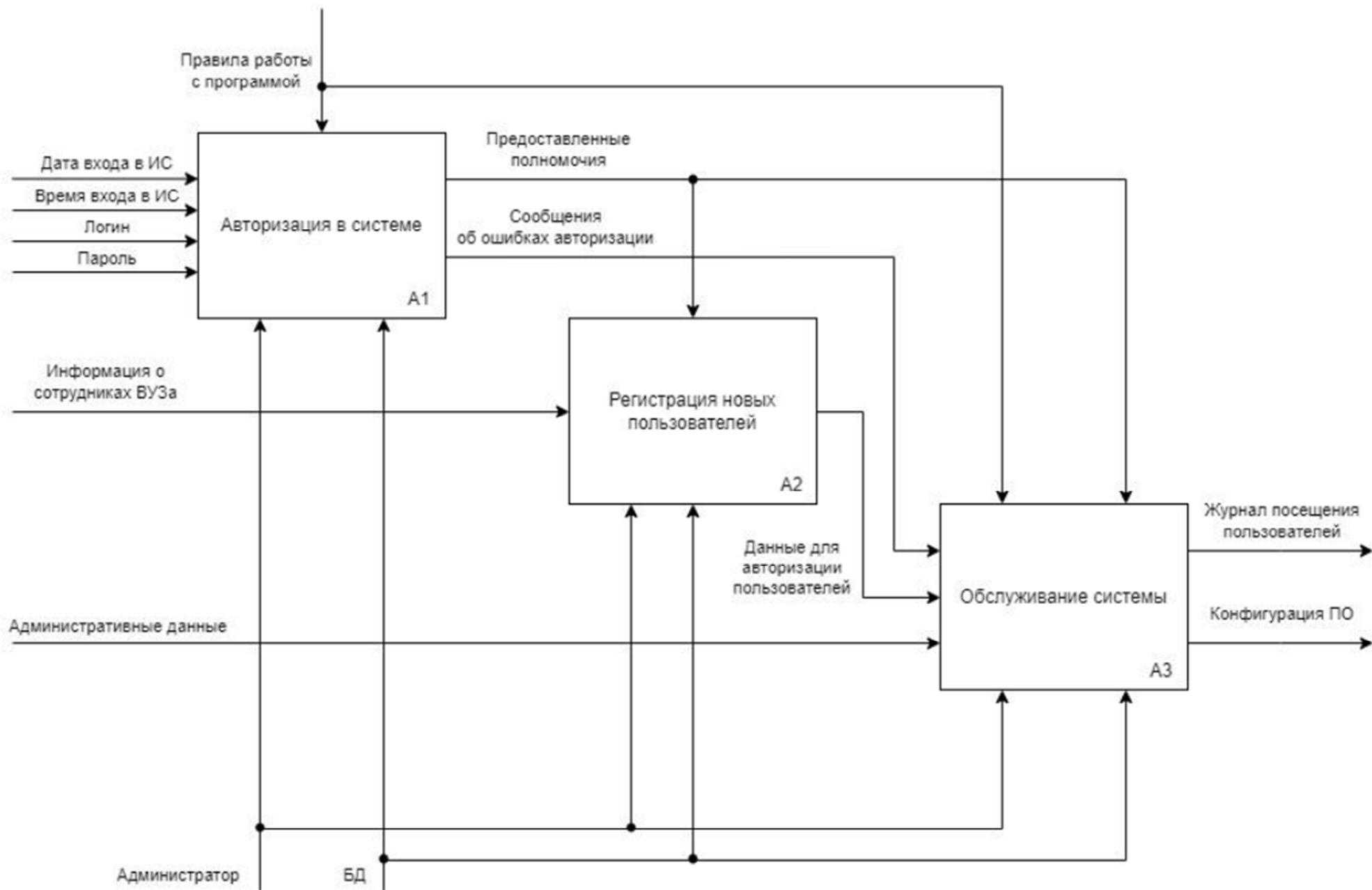


Рис. 8. Пример декомпозиции контекстной диаграммы ИС ВУЗа типа ТО ВЕ с точки зрения роли – администратор

Для рассмотренного примера диаграмма потоков данных уровня 0 может выглядеть следующим образом:



Рис. 9. Пример контекстной диаграммы в нотации DFD.

Контекстная диаграмма является точкой входа в модель потока данных и содержит один и только один процесс без указания хранилищ данных.

Основные подпроцессы и хранилища данных, составляющие информационную систему, продемонстрированы на диаграмме уровня 1, представленной на рисунке 10.

Из нее можно узнать следующее:

- Администратор может получить доступ к системе введя данные своей учетной записи.
- Введенные администратором данные проходят проверку верификации модулем безопасности системы.
- После получения доступа, администратор может осуществить изменение конфигурации системы или параметров учетных записей пользователей.
- Информация о попытках входа в систему или любых изменениях, затрагивающих ее, автоматически заносятся в электронный журнал событий.

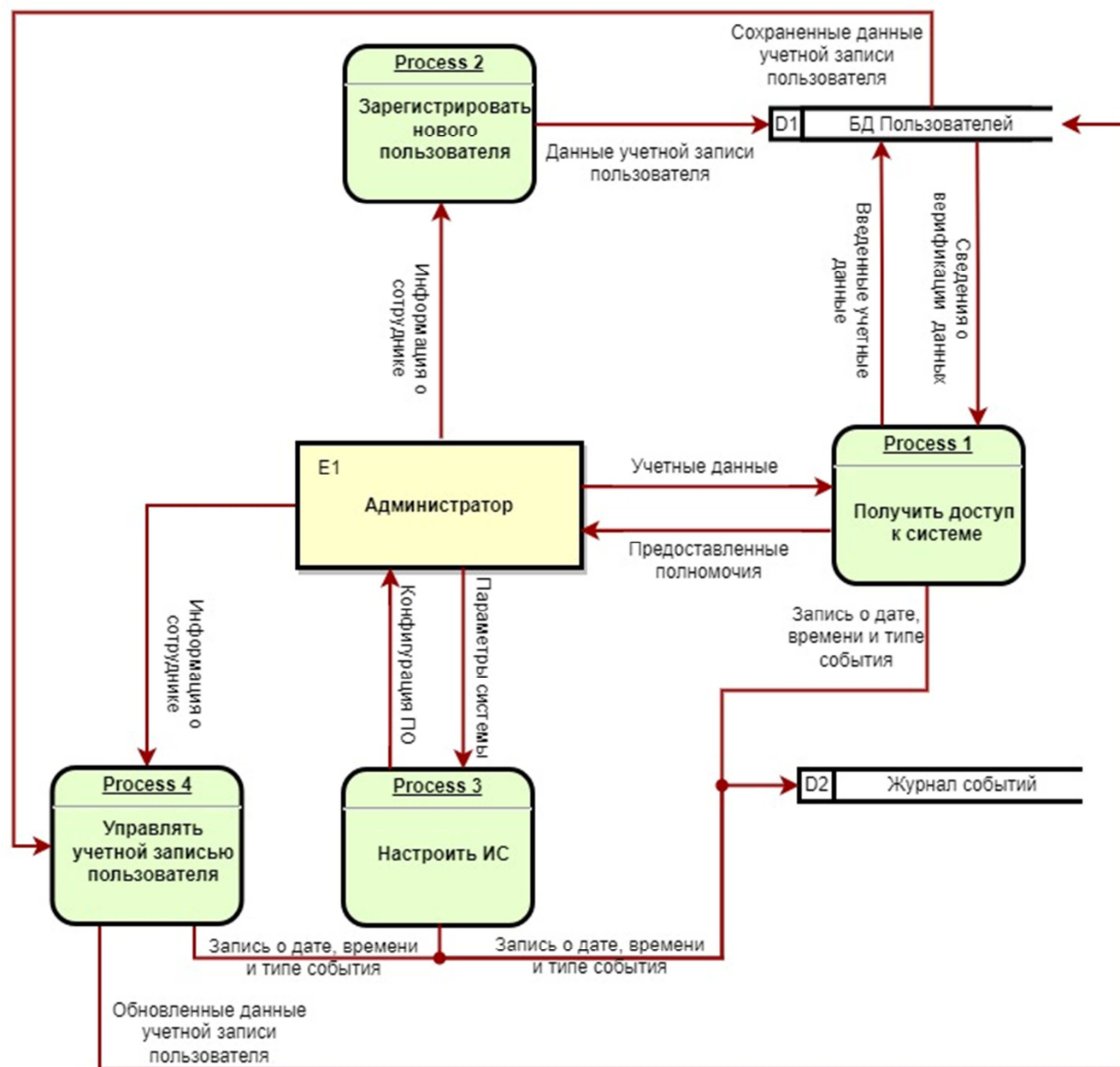


Рис. 10. Пример диаграммы DFD уровня 1.

Диаграмма декомпозиции процесса «Process 1», расположенного на первом уровне вышеприведенной схемы представлена на рисунке 11.

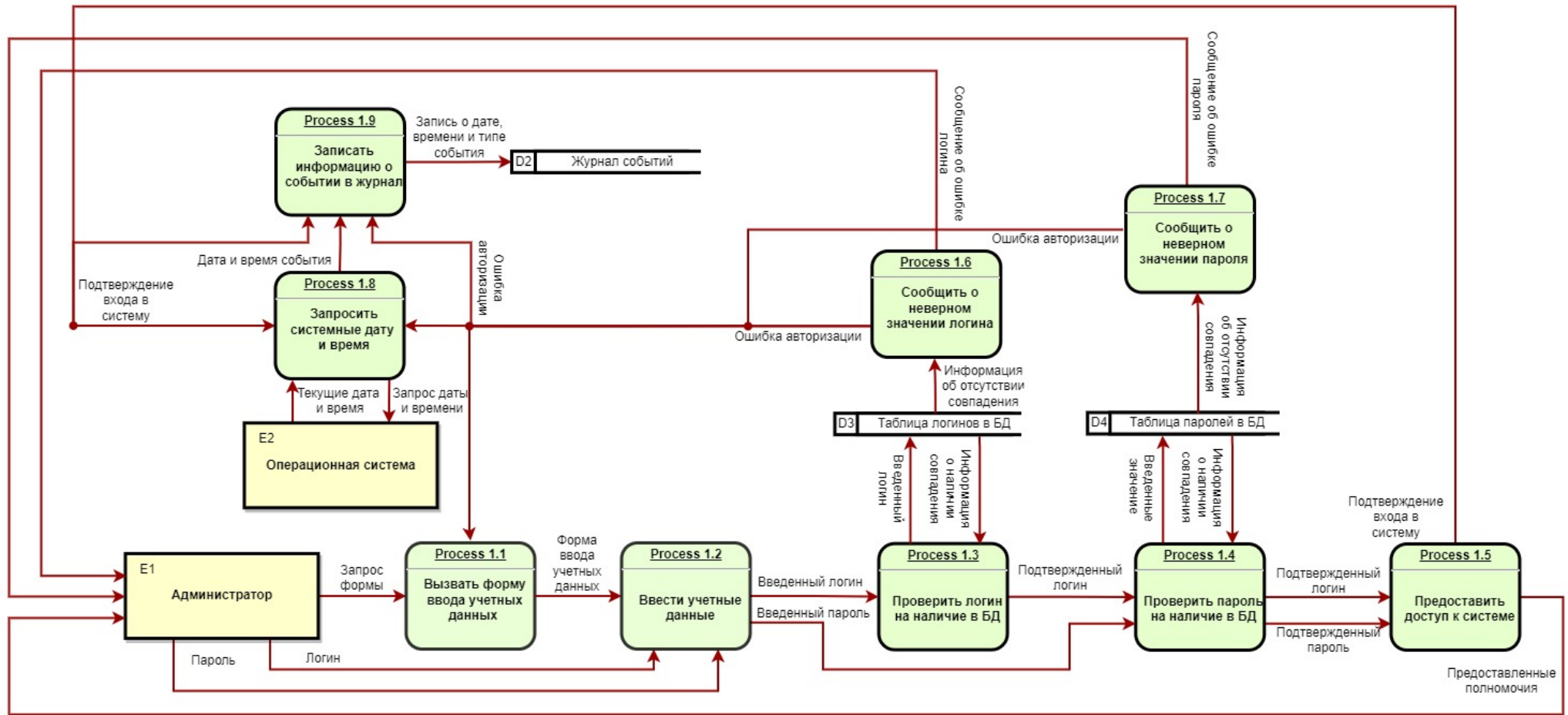


Рис. 11. Пример декомпозиции в нотации DFD.

На ней подробно изображены события, а также данные, необходимые и создаваемые для каждого из них, и их хранилища, происходящие в ходе процесса авторизации администратора в проектируемой информационной системе ВУЗа

Постановка задачи к практической работе

1. Изучить предлагаемый теоретический материал.
2. Предоставить преподавателю ранее созданную модель IDEF0 процессов ИС, соответствующую номеру индивидуального варианта, для получения задания на разработку диаграмм в нотации DFD.
3. Выполнить графическое моделирование указанных процессов

Содержание отчета

По выполненной работе составляется отчет в электронном виде, содержащий следующие структурные элементы:

- титульный лист;
- описание предметной области и вариант индивидуального задания;
- основные результаты выполненных предыдущих практических заданий, используемые для построения диаграмм;
- основная часть;
- выводы.

Требования к оформлению отчета:

- шрифт Times New Roman / Liberation Serif;
- размер шрифта 14 п.п;
- 1.5 междустрочный интервал;
- отступы 1.25;
- подписи к рисункам и таблицам обязательны.

Защита отчета по практической работе заключается в демонстрации самостоятельно полученных результатов и теоретических знаний в заданный срок с соблюдением установленных требований.