

Лабораторная работа

по дисциплине «Проектирование информационных систем»

Тема: Метод функционального моделирования SADT. Методология IDEF0.

Цель работы: Изучить теоретические основы структурного подхода к проектированию информационных систем. Освоить принципы построения IDEF0-диаграммы классов в программной среде Ramus Educational.

Задачи:

1. Ознакомиться с теоретическими вопросами структурного подхода к проектированию информационных систем.
2. Изучить диаграмму IDEF0 (Integration Definition for Function Modeling) для предметной области «Гостиница».
3. Построить с помощью программного средства Ramus Educational диаграмму IDEF0 согласно индивидуальному заданию (вариант получить у преподавателя).

2.1 Теоретические сведения

2.1.1 Общие положения структурного метода

Сущность структурного подхода к разработке ИС заключается в декомпозиции (разбиении) системы на автоматизируемые функции, которые в свою очередь делятся на подфункции, на задачи и так далее. Процесс декомпозиции продолжается вплоть до определения конкретных процедур. При этом автоматизируемая система сохраняет целостное представление, в котором все составляющие компоненты взаимосвязаны.

В основе структурного метода лежит несколько общих принципов:

- разбиение системы на множество независимых задач, доступных для понимания и решения;
- иерархическое упорядочивание, т.е. организация составных частей проблемы в древовидные структуры с добавлением новых деталей на каждом уровне.

К основным принципам относятся:

- *абстрагирование*, т.е. выделение существенных аспектов системы и отвлечение от несущественных;
- *формализация*, т.е. общее методологическое решение проблемы;
- *непротиворечивость*, состоящая в обосновании и согласовании элементов системы;
- *иерархическая структуризация данных*.

2.1.2 Метод функционального моделирования SADT

На основе метода SADT, предложенного Д. Россом, разработана методология IDEF0 (Icam DEFinition), которая является основной частью

программы ICAM (Интеграция компьютерных и промышленных технологий), проводимой по инициативе ВВС США. Методология IDEF0 является наиболее признанным эффективным средством анализа, конструирования и отображения бизнес-процессов, применяемым также и широко за пределами США.

Метод SADT применяется при моделировании широкого круга систем, для которых определяются требования и функции, после чего проводится их реализация.

Методология SADT представляет собой совокупность методов, правил и процедур, предназначенных для построения функциональной модели предметной области, которая отображает функциональную структуру, производимые функции и действия, а также связи между ними.

Результат применения метода SADT - модель, которая состоит из диаграмм, фрагментов текстов и глоссария со ссылками друг на друга. Все функции и интерфейсы представляются диаграммами в виде, соответственно, блоков и дуг. Место соединения дуги с блоком определяет тип интерфейса. Управляющая информация входит в блок сверху, в то время как информация, которая подвергается обработке (исходные данные), указывается с левой стороны блока, а результаты работы функции (выход, результат) - с правой стороны. Механизм, осуществляющий операцию (человек или автоматизированная система), задается дугой, входящей в блок снизу (см. Рис. 1).

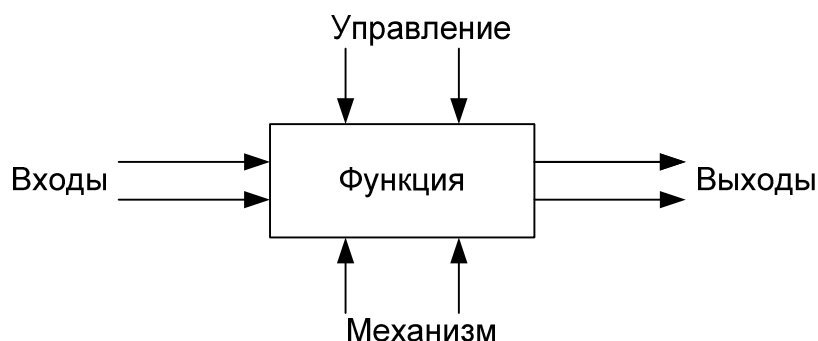


Рис. 1. Структура модели

Описание системы с помощью SADT называется **моделью**. **Субъектом** моделирования служит сама система. Однако моделируемая система никогда не существует изолированно: она всегда связана с окружающей средой. По этой причине в методологии SADT подчеркивается необходимость точного определения границ системы, 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. SADT-модель должна иметь единственный субъект.

С определением модели тесно связана позиция (называемая **точкой зрения**), с которой наблюдается система и создается ее модель. "Точку зрения" лучше всего представлять себе как место (роль, должность) человека

или объекта в рассматриваемой системе, на которое надо «встать», чтобы увидеть систему в действии и необходимой полноте. У конкретной модели может быть только **одна** точка зрения.

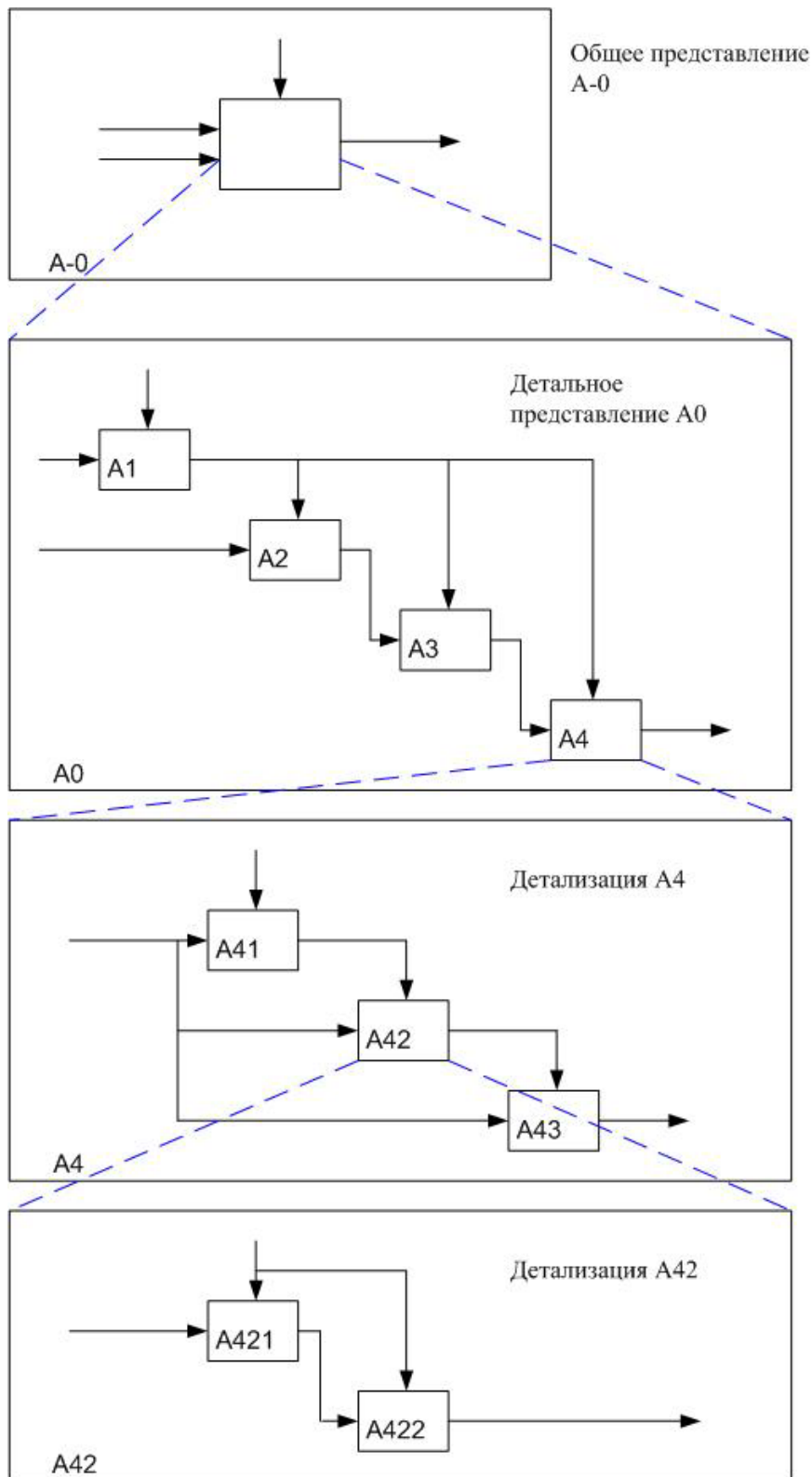


Рис. 2. Структура SADT-модели. Иерархия и декомпозиция диаграмм



Обычно вопросы для SADT-модели формулируются на самом раннем этапе проектирования, при этом основная суть этих вопросов должна быть выражена в одной-двух фразах, которые становятся **целью** модели.

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 представления. Цель становится критерием окончания моделирования. Конечным результатом этого процесса является набор тщательно взаимоувязанных описаний, начиная с описания самого верхнего уровня системы и заканчивая подробным описанием ее деталей или отдельных операций.

Каждое из таких тщательно взаимосогласованных описаний называется диаграммой и имеет определенный уровень детализации. SADT-модель объединяет и организует диаграммы в иерархические структуры, в которых диаграммы наверху модели менее детализированы, чем диаграммы нижних уровней. Другими словами, модель SADT можно представить в виде древовидной структуры диаграмм, где верхняя диаграмма является наиболее общей, а самые нижние – максимально детализированы (см. Рис. 2).

Каждый блок на диаграмме имеет свой номер. Блок любой диаграммы может быть детализирован диаграммой нижнего уровня, которая, в свою очередь, также может детализироваться с помощью необходимого числа диаграмм. Таким образом, формируется иерархия диаграмм. Для того чтобы указать положение любой диаграммы или блока в иерархии, им присваивают уникальные обозначения. Например, **A41** (**A** сокр. от **Activity**) является диаграммой, которая детализирует блок 1 на диаграмме **A4**. Аналогично, **A4** детализирует блок 4 на диаграмме **A0**, которая является самой верхней (родительской) диаграммой модели.

Некоторые дуги имеют начало в одном из блоков диаграммы и завершение в другом, у других же начало может исходить от границ диаграммы – дуги управления, механизма, дуги входа и выхода, перенесенные с родительской (верхнего уровня) диаграммы. Таким образом, источник или получатель этих пограничных дуг может быть обнаружен только на родительской диаграмме.

Также следует сказать о так называемых «туннельных дугах». Туннельные дуги означают, что данные, выраженные этими дугами не рассматриваются на следующем уровне детализации (как бы проходят «насквозь»). Если «туннель» расположен в месте соединения дуги с блоком , то данные этой дуги не обязательны на следующем уровне детализации. Если же «туннель» находится на противоположном конце дуги  - это значит, что данные дуги не описываются на родительской диаграмме. Граничные дуги должны продолжаться (дублироваться) на родительской диаграмме, делая ее полной и непротиворечивой (см. Рис. 3).

Для упрощения понимания приведенных диаграмм, следует расшифровать применяемую в IDEF систему обозначений, позволяющую

аналитику точно идентифицировать и проверять по дугам связи между диаграммами. Эта схема кодирования дуг - "ICOM" - получила название по первым буквам английских эквивалентов слов вход (**I**nput), управление (**C**ontrol), выход (**O**utput), механизм (**M**echanism).

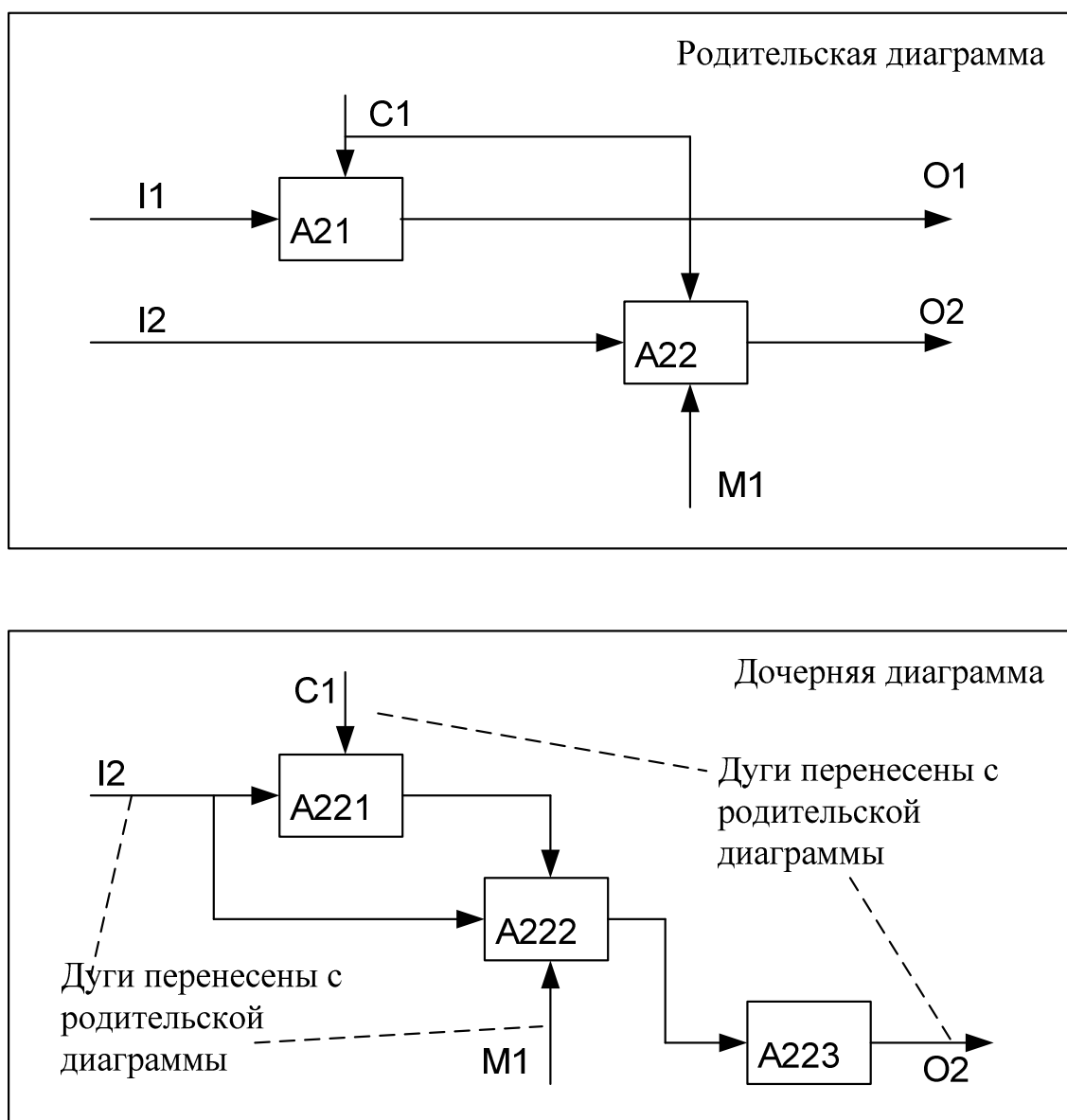


Рис. 3. Соответствие дуг родительской и дочерних диаграмм

2.2 Разработка модели IDEF0 в системе Ramus Educational

Программное обеспечение «Ramus» предназначено для использования в проектах, в которых необходимо описание бизнес-процессов предприятия. «Ramus» поддерживает методологии моделирования бизнес-процессов IDEF0 и DFD, а также имеет ряд дополнительных возможностей, призванных удовлетворить потребности команд разработчиков систем управления предприятиями. «Ramus» обладает гибкими возможностями построения отчетности по графическим моделям, позволяющие создавать отчеты в форме документов, регламентирующих деятельность предприятия.

Ramus Educational имеет достаточно интуитивный интерфейс пользователя, позволяющий быстро и просто создавать сложные модели.

2.2.1 Начало работы

1. Запустите программу Ramus Educational. В появившемся окне (см. Рис. 4) предлагается создать новый проект или открыть уже существующий.

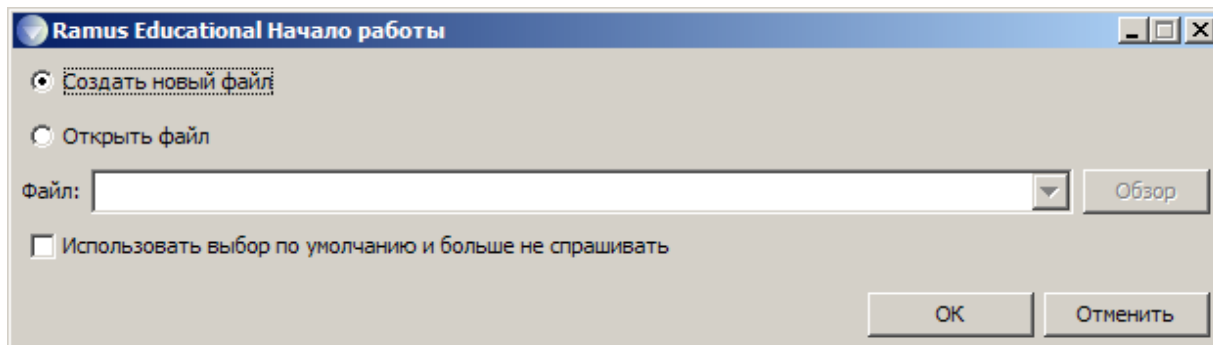


Рис. 4. Окно запуска Ramus Educational

2. После нажатия на кнопку «**ОК**» осуществляется запуск мастера проекта.

- На первом шаге (Рис. 5) в соответствующие поля необходимо внести сведения об авторе, названии проекта и модели, а также выбрать тип нотации модели (IDEF0 или DFD).
- На втором шаге вводится название организации, использующей данный проект.
- На третьем – дается краткое описание будущего проекта.
- Четвертый шаг позволяет создать несколько основных классификаторов (в данном случае можно пропустить этот шаг). Так как модели процессов реальных предприятий могут содержать значительное количество объектов (документы, персонал, функции и т.д.), то в Ramus предусмотрена возможность упорядочено хранить информацию об этих объектах в виде системы классификаторов. Классификация объектов упрощает поиск и обработку информации об объектах модели, а так же и об объектах непосредственно не представленных на диаграммах процессов, но относящихся к процессам предприятия.
- На пятом, заключительном, предлагается выбрать те из созданных классификаторов, элементы которых будут содержаться в перечне собственников процессов (пропустить данный шаг).

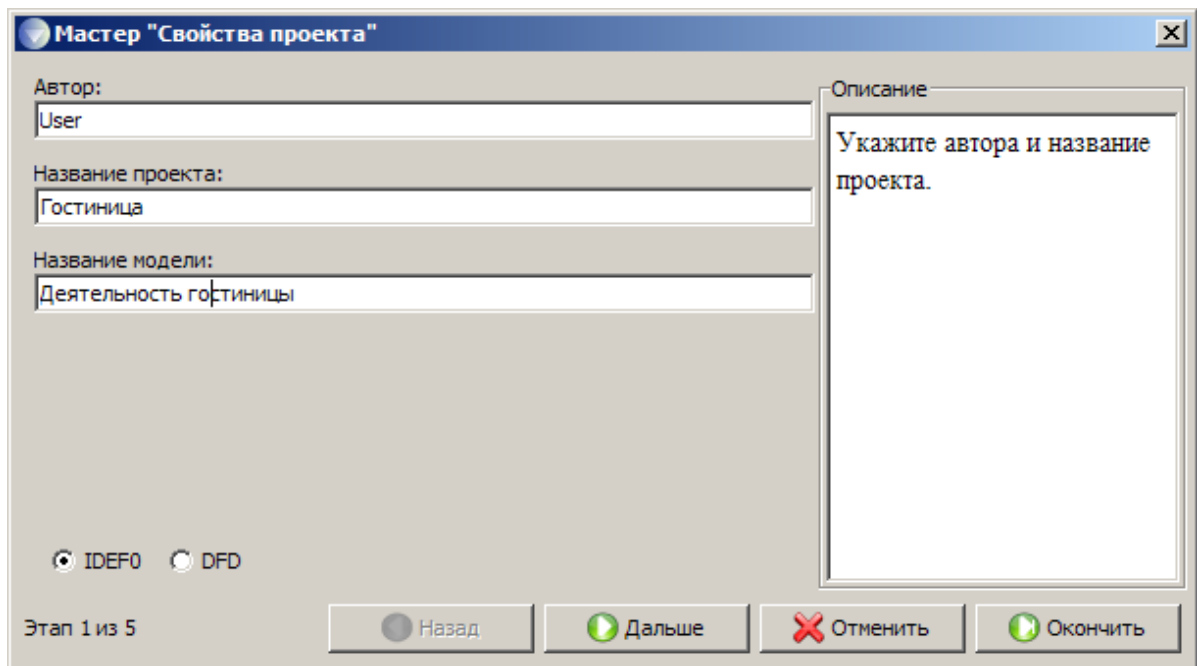


Рис. 5. Окно мастера создания проекта. Шаг 1

При необходимости можно завершить работу мастера, нажав кнопку **«Окончить»**.

После завершения работы мастера, откроется рабочее пространство **«Диаграммы»**, в котором можно приступить к рисованию графической модели (Рис. 6). В верхней части приводятся сведения о проекте, введенные пользователем посредством мастера диаграмм.

Программа Ramus Educational обладает гибким графическим интерфейсом, который можно настроить под нужды и предпочтения конкретного пользователя: ненужные окна можно закрыть/свернуть; можно менять их размеры и месторасположение; также можно группировать два и более окон в одном, при этом содержимое вложенных окон будет размещено на вкладках общего окна (данный функционал возможен не для всех комбинаций окон).

3. Сохраните созданную модель, выбрав опцию меню **«Файл»** → **«Сохранить как»**.

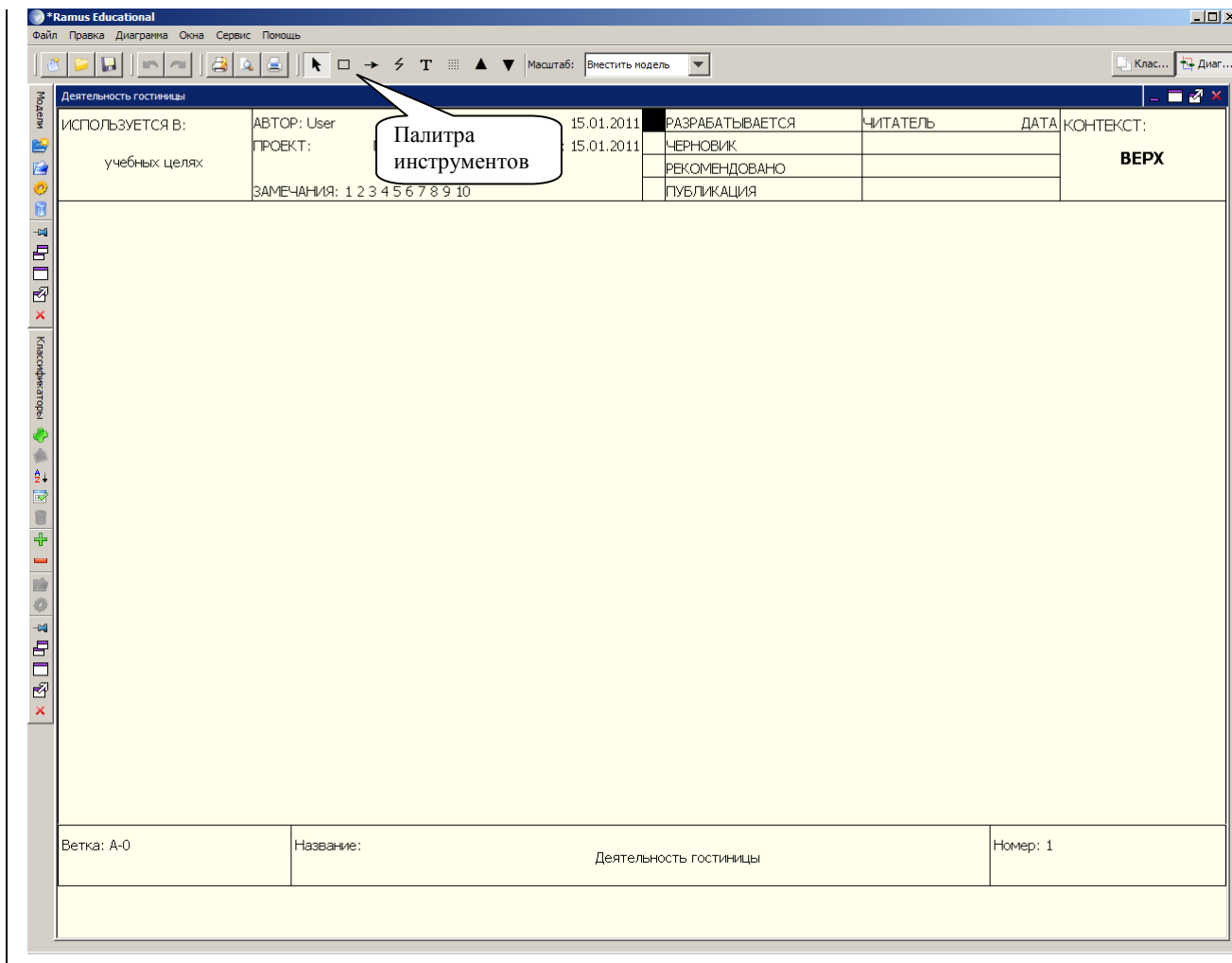


Рис. 6. Рабочее пространство «Диаграммы»

2.2.2 Создание контекстной диаграммы

1. На панели инструментов выберите пиктограмму функции (□) и мышью укажите месторасположение на рабочем пространстве.
2. Дайте данному функциональному блоку имя «**Описать деятельность ГОСТИНИЦЫ**».
3. Используя пиктограмму панели инструментов (→), создайте стрелки на контекстной диаграмме согласно Таблица 1.

Таблица 1. Контекстная диаграмма

Наименование стрелки	Тип
Заявки на заселение в гостиницу	ВХОД
Материально-техническое и кадровое обеспечение работы гостиницы	ВХОД
Нормативно-правовая база	управление
Гостиничные услуги	ВЫХОД

Наименование стрелки	Тип
Отчетность	ВЫХОД
Финансовые результаты деятельности	ВЫХОД
Услуги контрагентов	МЕХАНИЗМ
Инфраструктура гостиницы	МЕХАНИЗМ

4. В результате должна получиться контекстная диаграмма, показанная на Рис. 7.

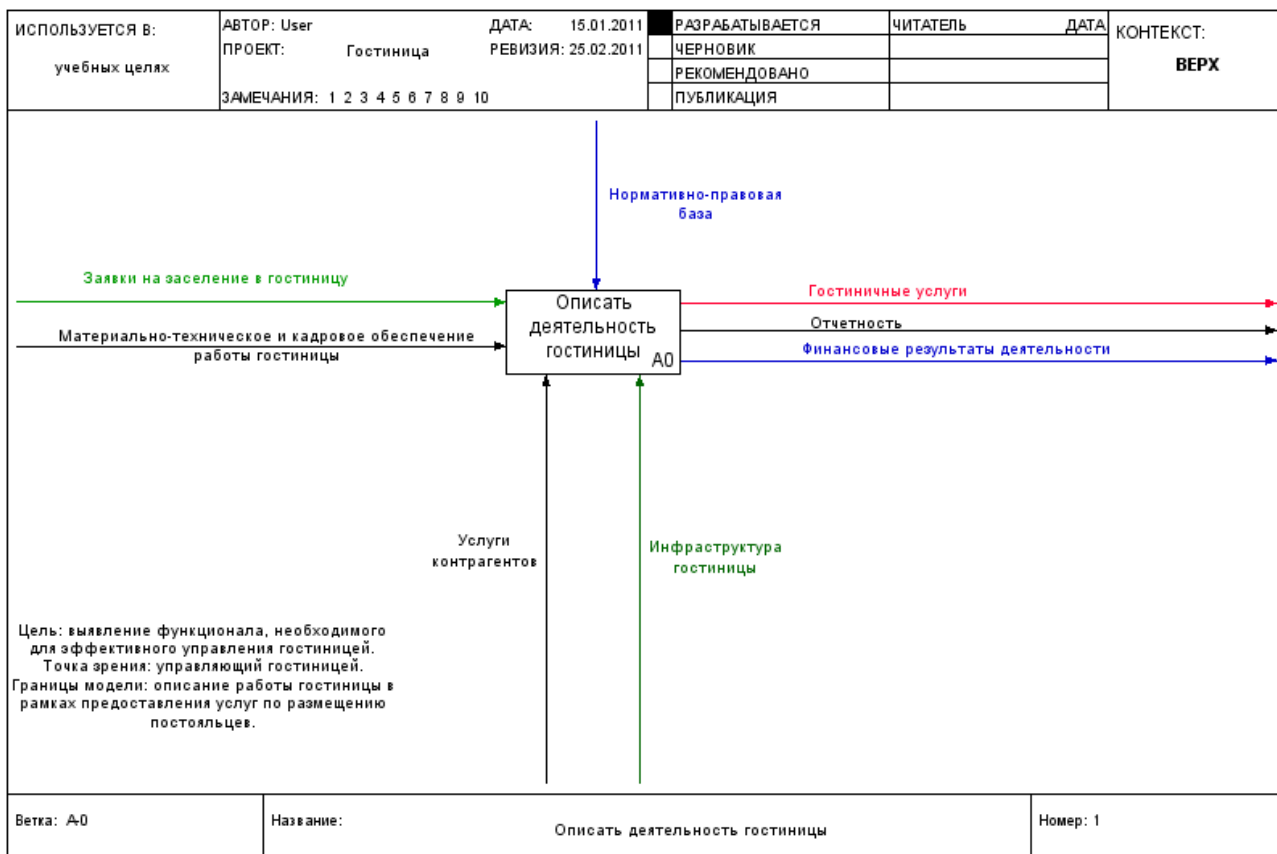



Рис. 7. Контекстная диаграмма предметной области «Гостиница»

2.2.3 Создание диаграммы декомпозиции

1. Выберите в палитре инструментов кнопку перехода на нижний уровень , в диалоговом окне «Создание новой диаграммы» (Рис. 8) установите количество функциональных блоков **3**, укажите тип диаграммы (**IDEF0**) и нажмите кнопку **OK**.

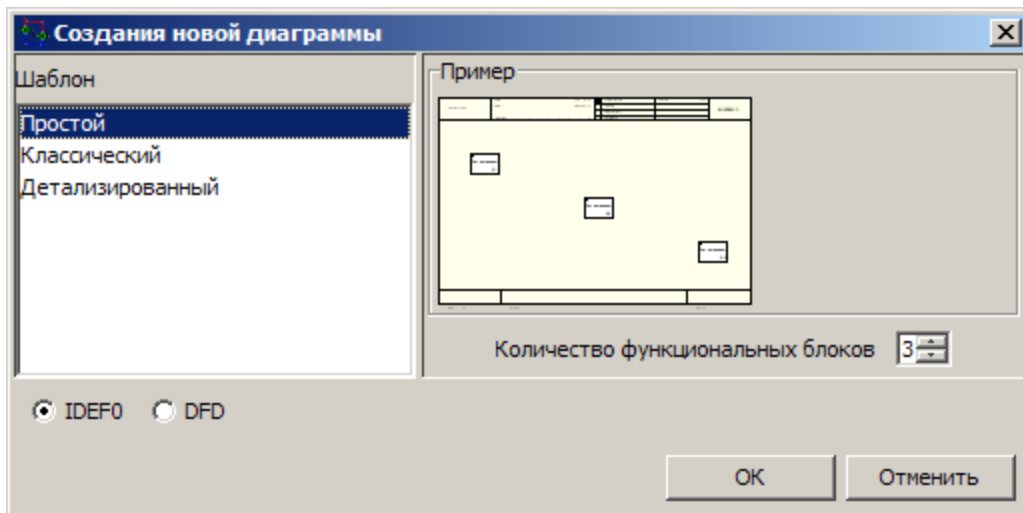


Рис. 8. Диалоговое окно создания детализирующей диаграммы

2. Автоматически будет создана диаграмма первого уровня декомпозиции (см. Рис. 9) с перенесенными в нее потоками родительской диаграммы.

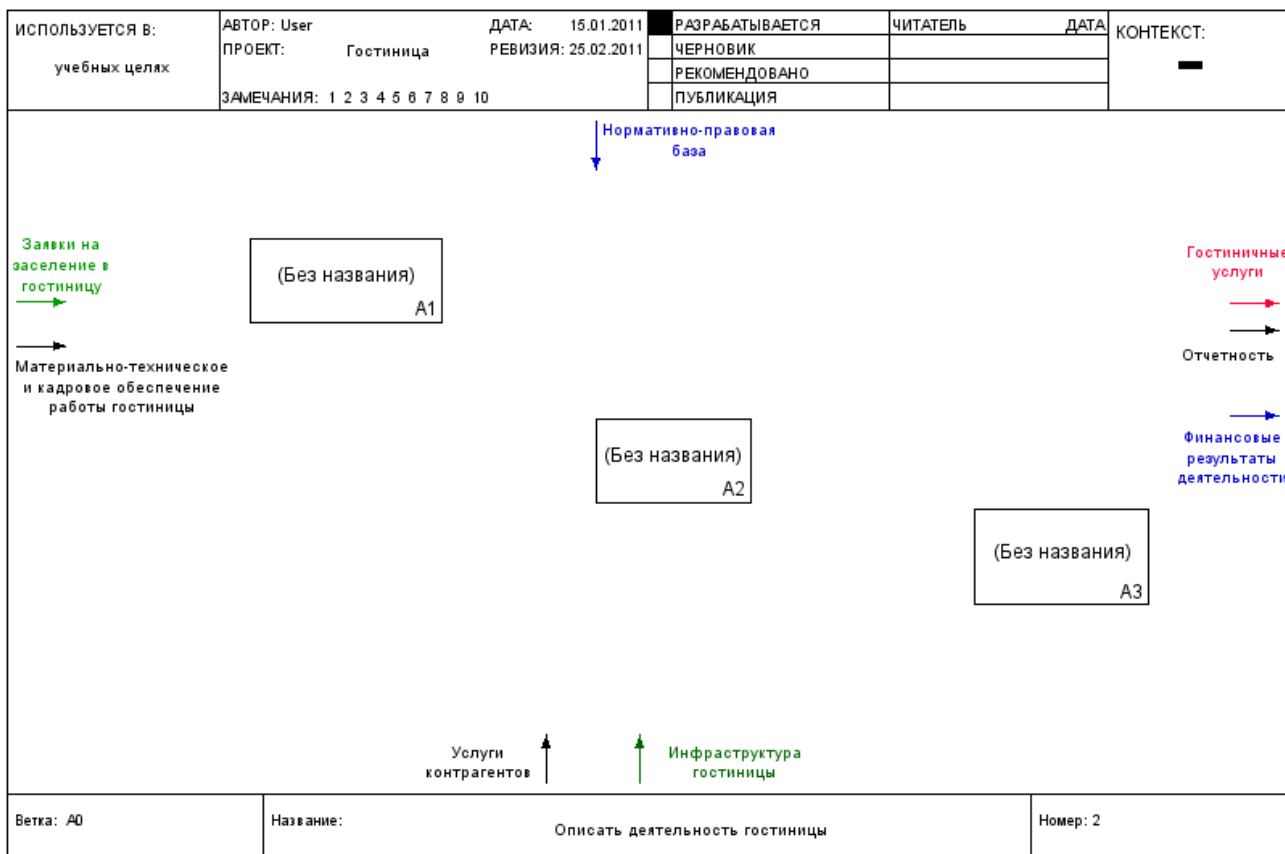


Рис. 9. Рабочее пространство детализирующей диаграммы

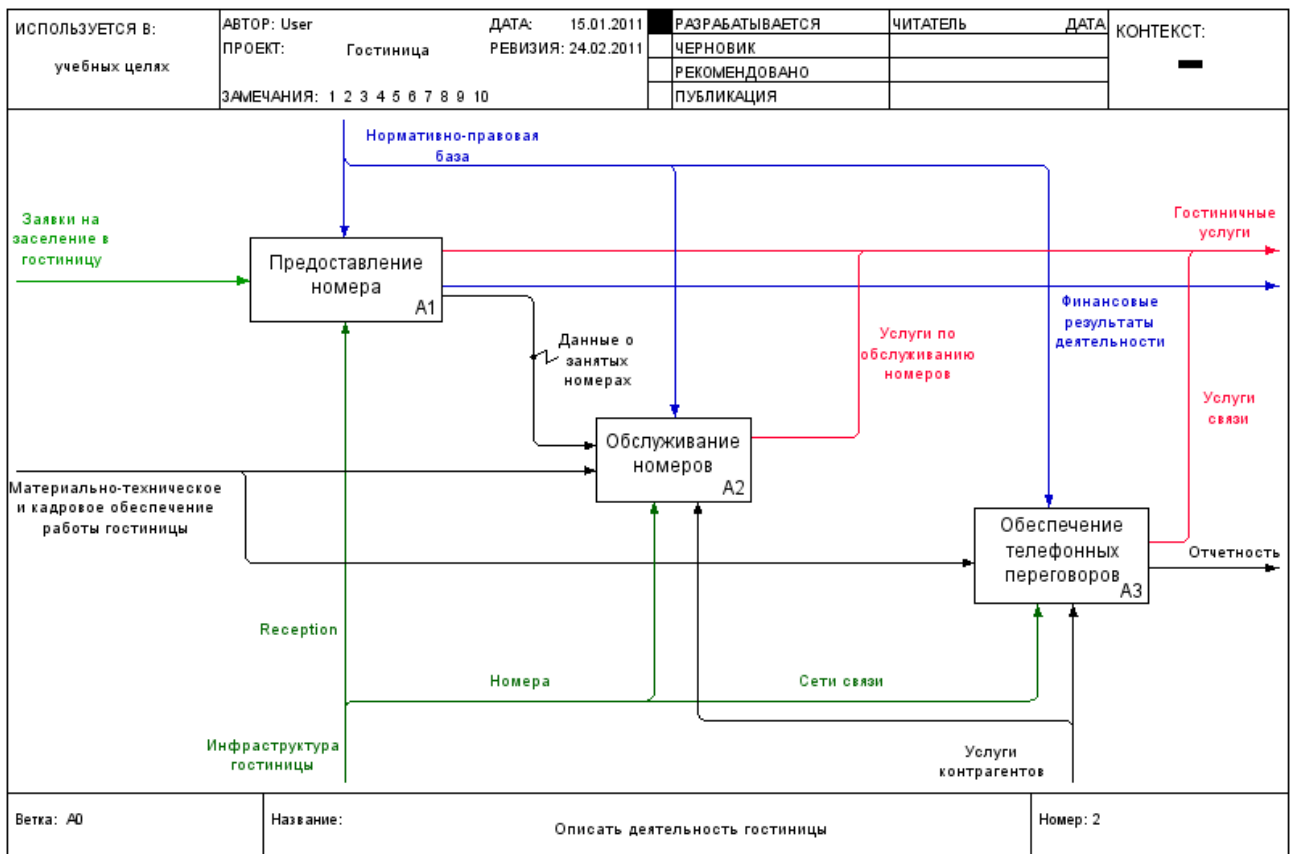


Рис. 10. Детализированная диаграмма первого уровня

3. Осуществите построение детализирующих диаграмм для функциональных блоков согласно представленной иерархии (см. Рис. 11).

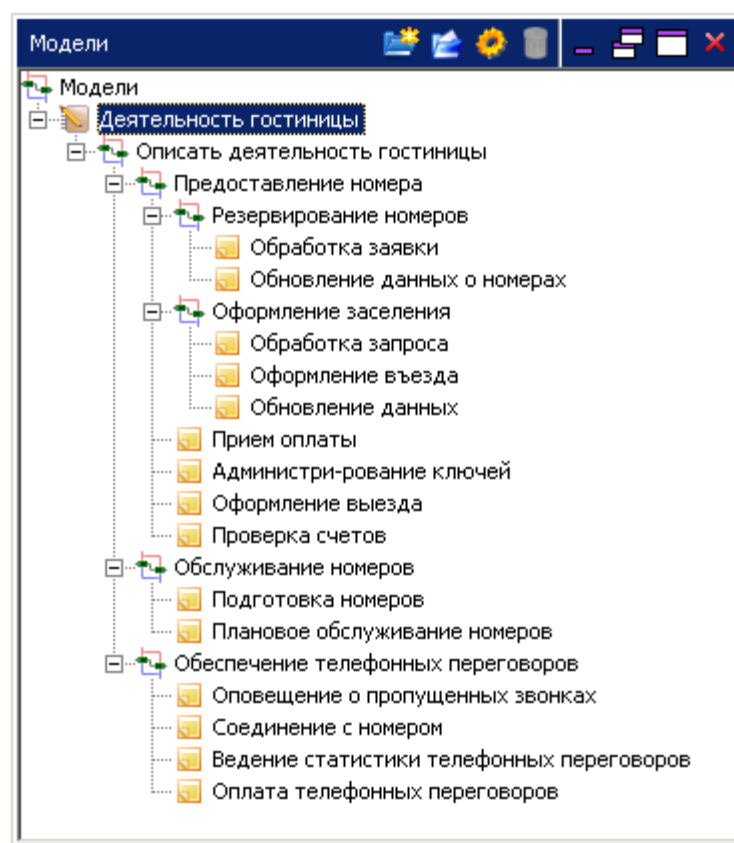


Рис. 11. Иерархия функциональных блоков модели

4. В результате должны быть разработаны IDEF0-диаграммы, представленные на Рис. 12 - Рис. 16.

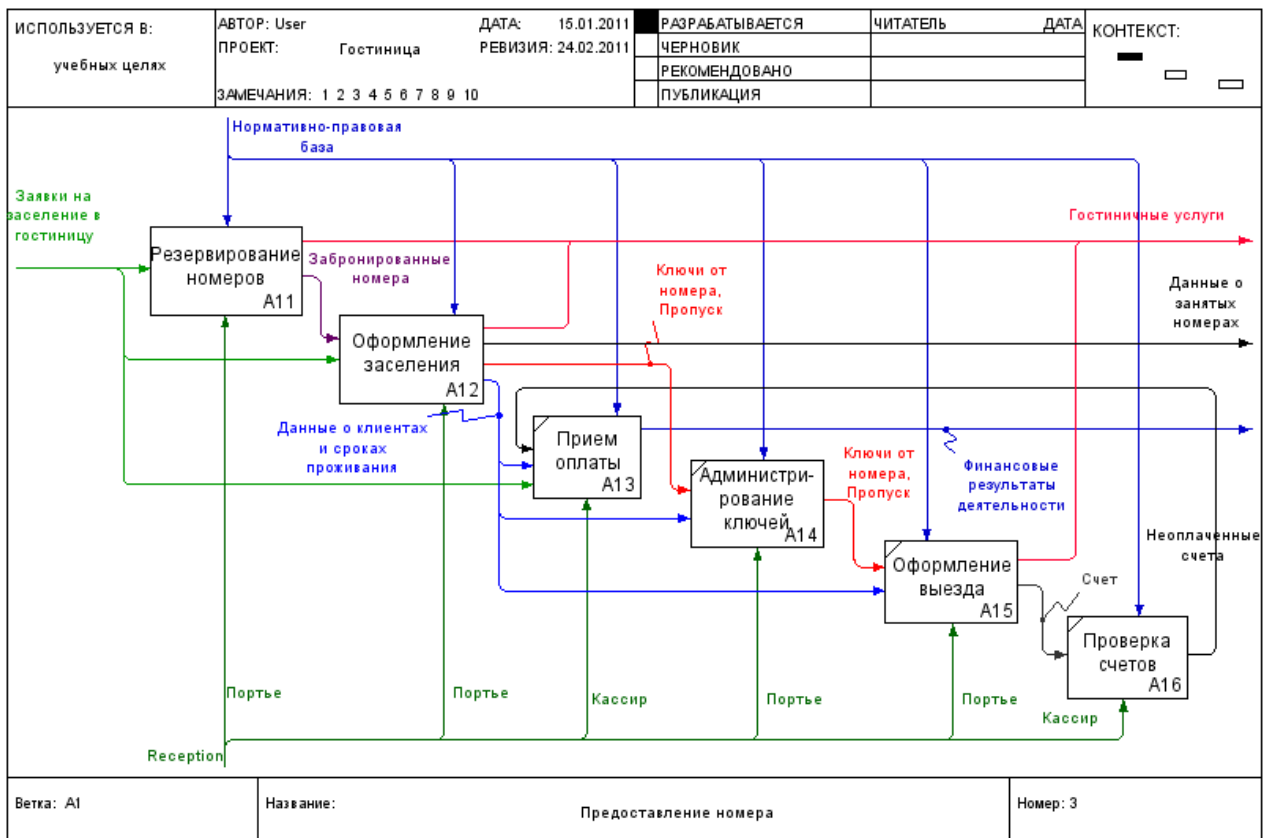


Рис. 12. Детализированная диаграмма работы «Предоставление номера»

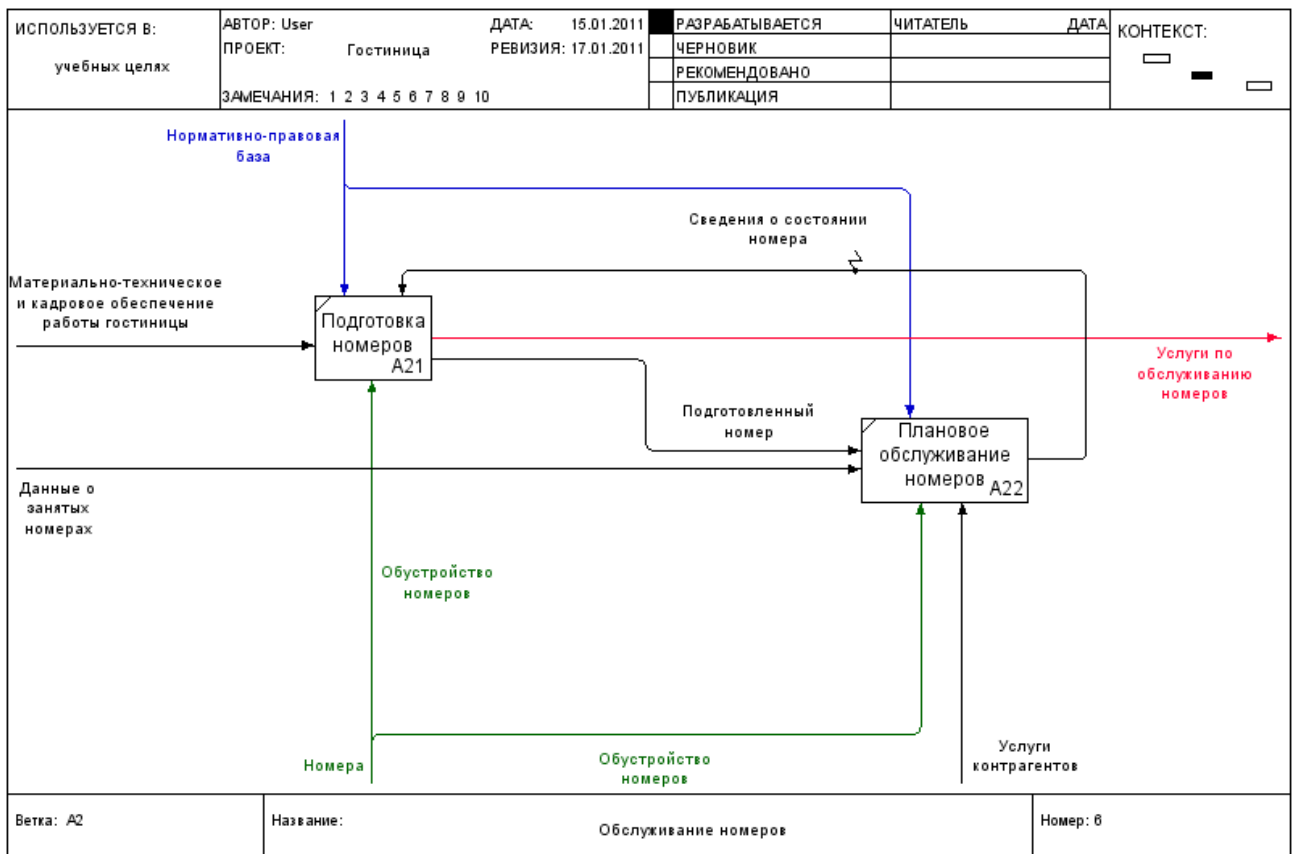


Рис. 13. Детализированная диаграмма работы «Обслуживание номеров»

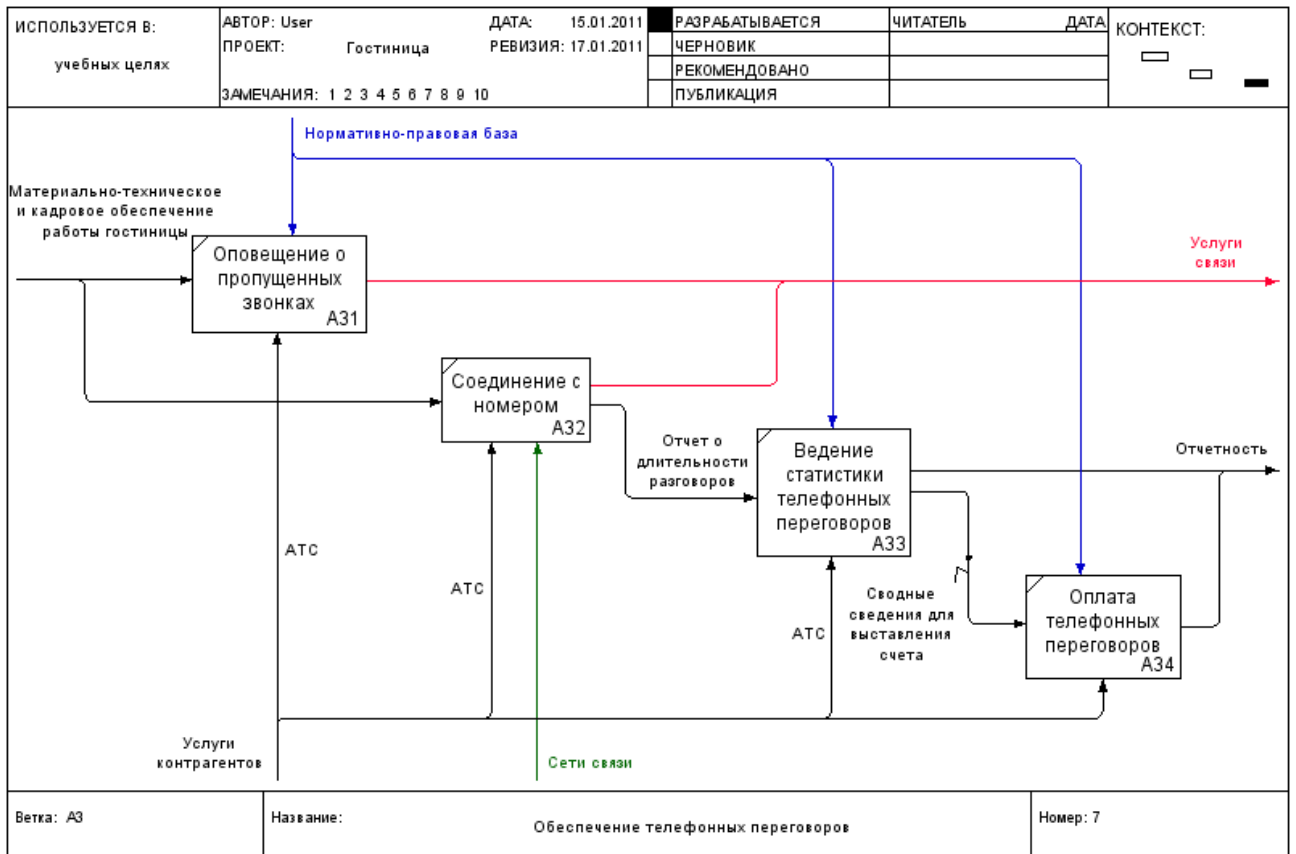


Рис. 14. Детализированная диаграмма работы «Обеспечение телефонных переговоров»

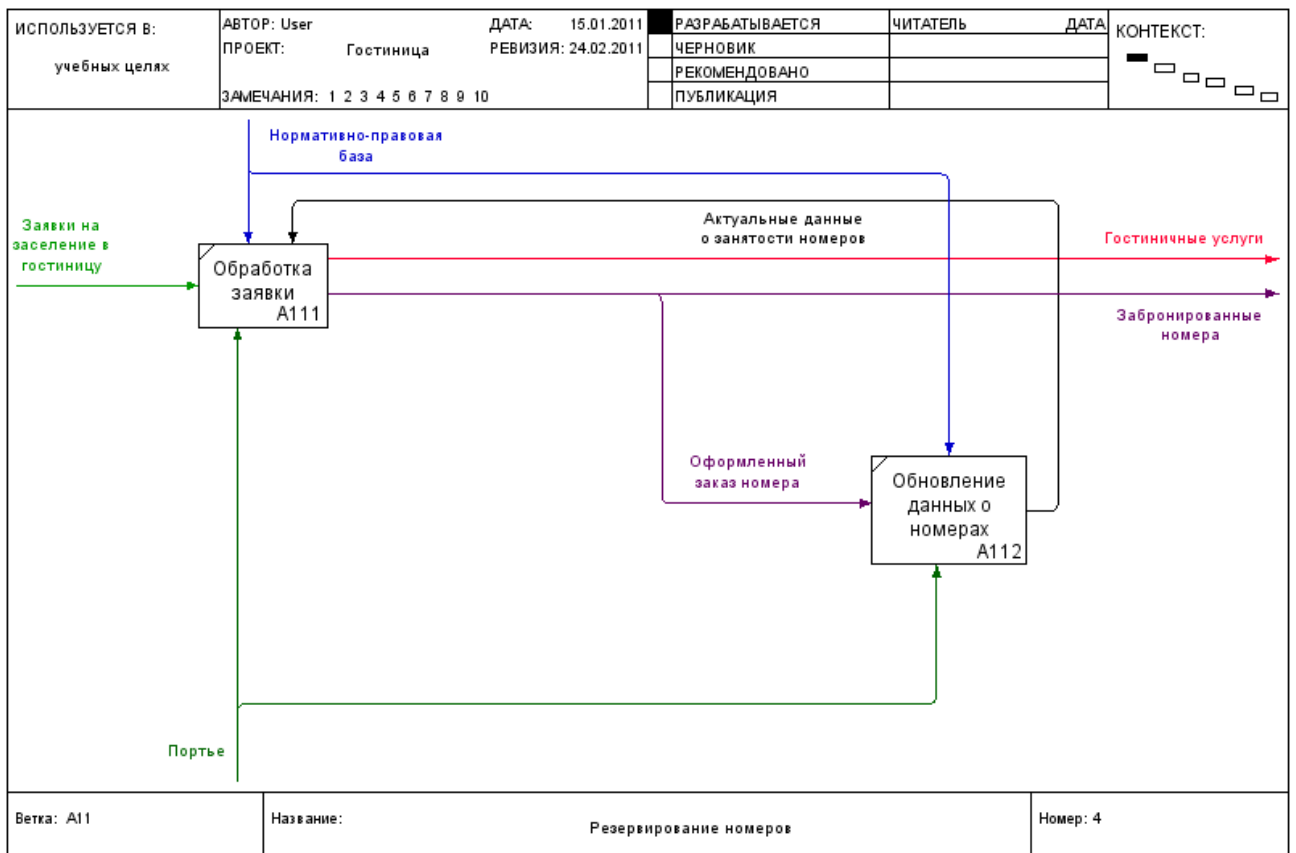


Рис. 15. Детализированная диаграмма работы «Резервирование номеров»

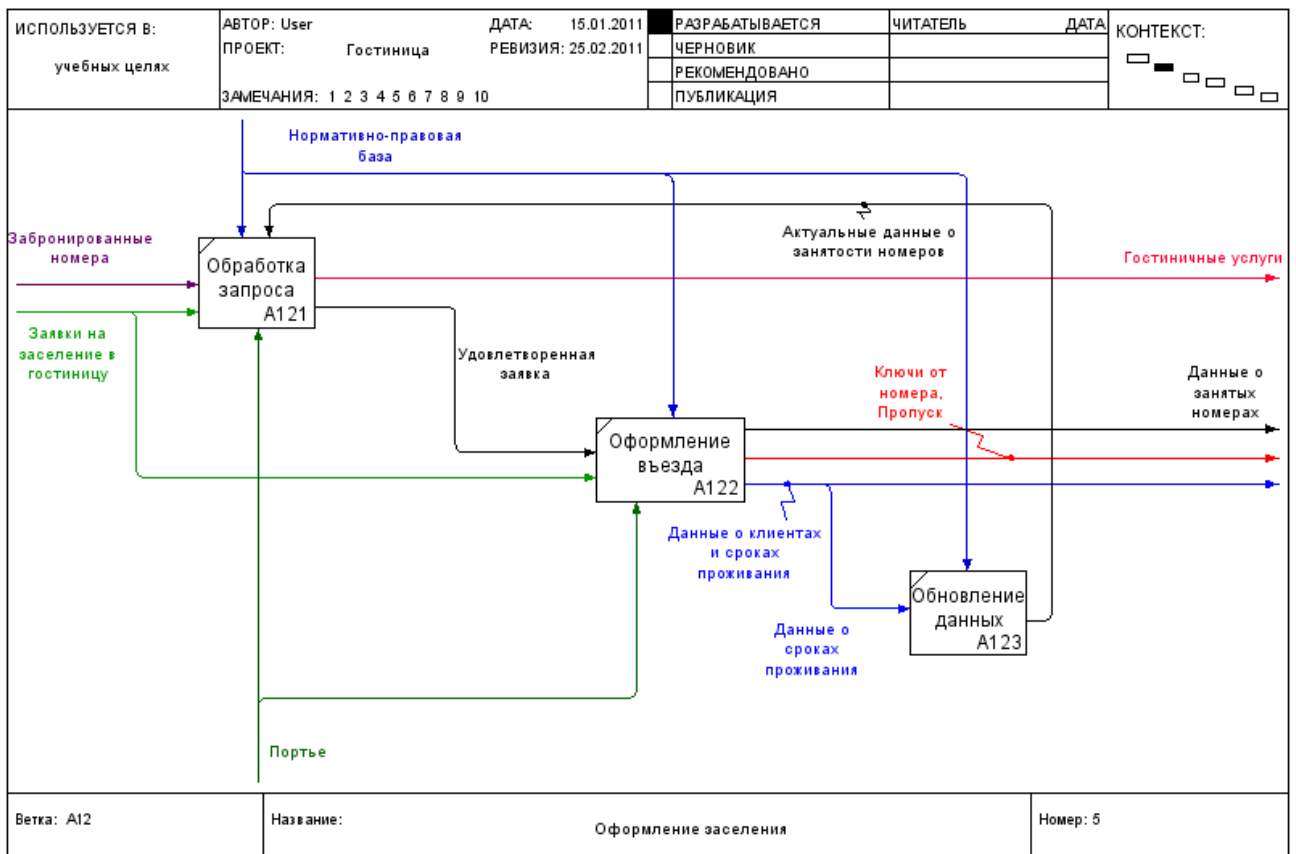


Рис. 16. Детализированная диаграмма работы «Оформление заселения»

5. В программе Ramus Educational предусмотрена возможность экспорта разработанных диаграмм в виде рисунков формата *.png, *.bmp или *.jpeg. Для этого в главном меню необходимо выбрать команду «**Диаграммы**» → «**Экспортировать как рисунки**». В появившемся окне указывается список экспортируемых рисунков, выбирается их формат и размер, а также путь для сохранения (см. Рис. 17).

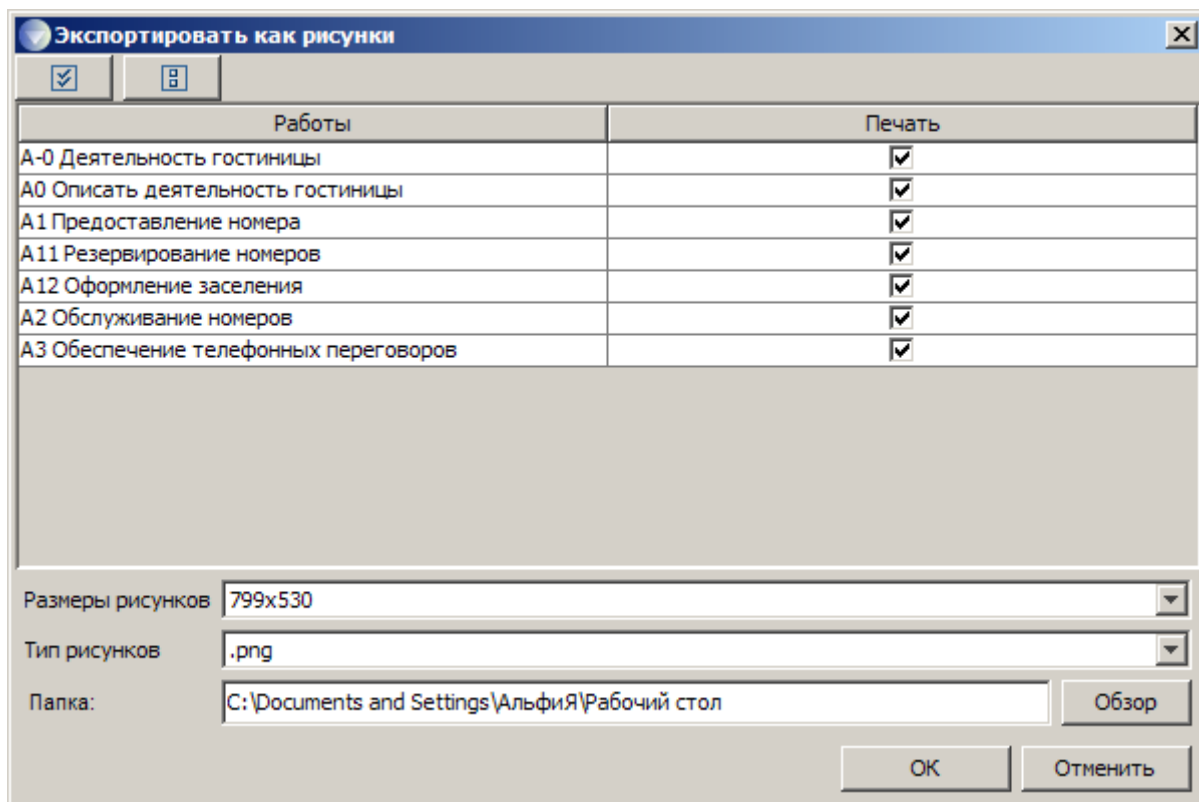


Рис. 17. Диалоговое окно экспорта диаграмм

6. Покажите преподавателю разработанные диаграммы.

Содержание работы

1. Ознакомиться с теоретическими вопросами построения IDEF0-диаграммы.
2. Изучить диаграмму IDEF0 для предметной области «Гостиница».
3. Осуществить построение IDEF0-диаграммы с помощью программного средства Ramus Educational согласно индивидуальному заданию (вариант получить у преподавателя).

Требования к отчету

Отчет по лабораторной работе оформляется в печатном виде. Защита работы включает в себя проверку знания студентом теоретического материала, а также практической части лабораторной работы.

Отчет должен включать:

- Разработанные IDEF0-диаграммы.
- Описание разработанных IDEF0-диаграмм.

Все примеры должны быть сохранены на сетевом диске или на диске студента.

ВАРИАНТЫ ЗАДАНИЙ

Вариант задания выбирается по сумме двух последних цифр зачетной книжки.

1. Автострахование.
2. Агентство по сдаче автомобилей в аренду.
3. Аренда коньков, роликов, велосипедов, лыж.
4. Аэропорт – пассажирское расписание и перевозки.
5. Банковская система вкладов (физических и юридических лиц) .
6. Банковская система кредитования (физических и юридических лиц).
7. Биллинг сотовой компании.
8. Ветеринарная лечебница.
9. Клуб обучения танцам.
10. Магазин косметики.
11. Машиностроительное предприятие: система по разработке и модификации изделий (ведение архива, стандартов и пр.).
12. Нефтеперерабатывающая компания.
13. Парикмахерская.
14. Поставка вин.
15. Приемная комиссия ВУЗа.
16. Производство мебели (прием индивидуальных и типовых заказов и изготовление).
17. Рекламное агентство.
18. Риэлторская компания: аренда; продажа первичного и вторичного жилья.
19. Санаторий.
20. Система управления проектом для IT-компании.
21. Складская логистика.
22. Спа-салон (услуги, обслуживающий персонал и пр.).
23. Страховая компания.
24. Такси.
25. Транспортная логистика.
26. Туристическое агентство (путешествия за рубеж).
27. Туристическое агентство (путешествия по России).
28. Учет оборудования на крупном промышленном предприятии.
29. Филармония.
30. Электронный проездной.