

Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное
учреждение высшего профессионального образования

**ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ**

Кафедра автоматизации обработки информации (АОИ)

О.И. Жуковский

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И АНАЛИЗ ДАННЫХ

**Методические указания
по выполнению лабораторных работ,
организации практических занятий
и самостоятельной работы**

2014

Корректор: Осипова Е.А.

Жуковский О.И.

Информационные технологии и анализ данных: методические указания по выполнению лабораторных работ, организации практических занятий и самостоятельной работы. — Томск: Факультет дистанционного обучения, ТУСУР, 2014. — 55 с.

© Жуковский О.И., 2014
© Факультет дистанционного
обучения, ТУСУР, 2014

СОДЕРЖАНИЕ

Введение.....	4
1 Принципы функционального моделирования.....	5
2 Создание функциональных моделей и диаграмм.....	6
2.1 Начало моделирования.....	6
2.2 Продолжение моделирования.....	10
2.3 Проверка диаграммы автором.....	12
2.4 Завершение моделирования.....	14
2.5 Дополнения к диаграммам и моделям.....	16
3 Автоматизация построения модели.....	19
4 Описание работы с Design/IDEF.....	22
4.1 Создание диаграмм методологии IDEF0.....	22
4.2 Создание словарей методологии IDEF0.....	39
5 Лабораторная работа.....	51
5.1 Сбор информации.....	51
5.2 Начало моделирования.....	52
5.3 Продолжение моделирования.....	53
5.4 Завершение моделирования.....	53
5.5 Список процессов для моделирования.....	53
Список литературы.....	55

ВВЕДЕНИЕ

В данном пособии рассматриваются основные понятия, методы и процедуры, связанные с построением функциональных моделей с использованием методологии IDEF0 (IDEF0).

IDEF0 используется для создания функциональной модели, которая является структурированным изображением функций производственной системы или среды, а также информации и объектов, связывающих эти функции.

1 ПРИНЦИПЫ ФУНКЦИОНАЛЬНОГО МОДЕЛИРОВАНИЯ

Описание системы с помощью IDEF0 называется **моделью**. В IDEF0-моделях используются как естественный, так и графический языки.

Одна и та же схема моделирования может быть использована для моделирования любого выбранного объекта. Универсальной единицей для неограниченного строго структурного анализа является блок (рис. 1):

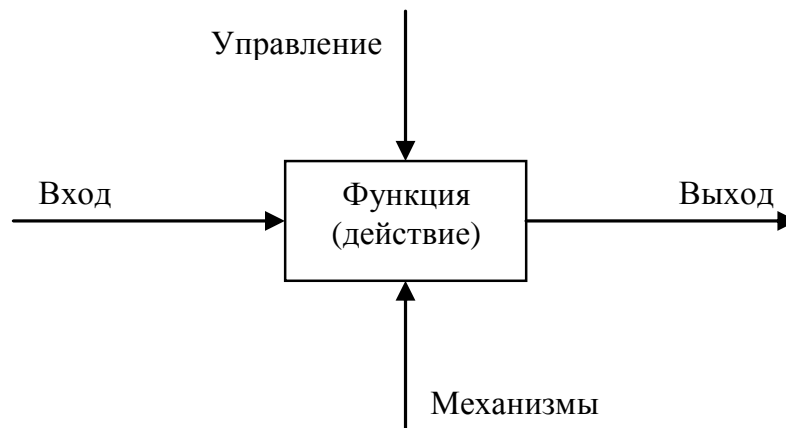


Рис. 1 — Пример блока

Вход при наличии **управления** преобразуется в **выход** с помощью **механизма**.

Выходы одного блока могут быть входами или управлениями (или механизмами) для других блоков. Блоки именуются, а дуги помечаются с использованием естественного языка. Дуги могут разветвляться и соединяться, а каждый блок может быть подвергнут декомпозиции, т. е. разделен как целое на свои составляющие на более детальной диаграмме. Входы, управления и выходы определяют интерфейсы между блоками, а механизмы позволяют при необходимости в определенной степени объединять объекты. Границы блоков и диаграмм должны быть согласованы, а возникающая иерархически взаимосвязанная совокупность диаграмм является моделью.

Диаграмма ограничивается 3—6 блоками. Вместо одной громоздкой модели используется несколько небольших взаимосвязанных моделей, значения которых взаимно дополняют друг друга, делая понятной структуризацию сложного объекта.

2 СОЗДАНИЕ ФУНКЦИОНАЛЬНЫХ МОДЕЛЕЙ И ДИАГРАММ

2.1 Начало моделирования

Начало моделирования означает создание диаграмм А-0 и А0, которые полностью рассказывают все об изучаемой системе с минимальной степенью детализации. Декомпозиция (диаграмма А0) освещает наиболее важные виды деятельности, обеспечивающие функционирование системы и объекты системы. Объединение (диаграмма А-0) трактует систему как «черный ящик», дает ей название и определяет наиболее важные входы, управления, выходы и механизмы.

Декомпозируя объект, нужно, прежде всего, обратить внимание на входные и выходные данные для всей системы. Декомпозиция всей системы начинается с составления списка основных типов данных и основных функций. Потом эти списки снабжаются комментариями для указания основных типов как данных, так и функций системы или их различных сочетаний. Наконец, списки с комментариями используются для создания диаграммы А0, которая затем обобщается с помощью диаграммы А-0.

Цель и точка зрения модели определяются на самой ранней стадии создания модели. Выбор цели осуществляется с учетом вопросов, на которые должна ответить модель, а выбор точки зрения — в соответствии с выбором позиции, с которой описывается система. Иногда цель и точку зрения можно выбрать до того, как будет сделана первая диаграмма. В рамках нашего курса цель модели разработки нового программного продукта можно определить заранее, потому что она очевидна в постановке задачи: «понять обязанности всех задействованных лиц так, чтобы организовать процесс разработки программного обеспечения». Настоятельно рекомендуется, как можно раньше определять цель и выбирать точку зрения новой модели. Но вначале попробуйте сформулировать ряд специфических вопросов, на которые модель должна ответить, чтобы убедиться, что цель сформулирована точно, и рассмотрите систему с нескольких различных точек зрения, прежде чем выбрать одну из них.

Иногда оказывается, что определить цель и точку зрения в самом начале моделирования чрезвычайно трудно. В таком случае следует составить списки данных и функций и, может быть, нарисовать диаграмму A-1, прообраз диаграммы A0. Сделав это, вы начнете чувствовать систему и установите, описывает ли ее диаграмма A0 с нужной точки зрения. Может быть, вам придется нарисовать несколько альтернативных A0-диаграмм, прежде чем появится достаточная уверенность для того, чтобы осуществить выбор правильной цели и точки зрения.

Списки объектов системы, создаваемые в ходе моделирования, в IDEF0 принято называть *списками данных*. Термин «данное» здесь употребляется как синоним слова «объект». Составление списка данных является начальным этапом создания каждой диаграммы функциональной IDEF0-модели. Правило заключается в том, чтобы вначале составить список данных, а потом список функций.

IDEF0-диаграммы представляют границы функций и ограничения, накладываемые на них, причем *ограничения должны присутствовать во всех системах*. Указывая вначале ограничения, выявляем естественную структуру системы. Без ограничений функциональная IDEF0-диаграмма представляет собой не более чем схему потоков данных. Без ограничительных дуг диаграммы не смогут рассказать читателю, почему аналитик выбрал именно данную декомпозицию. Благодаря тому, что в IDEF0 различаются входные дуги и дуги управления (информация, необходимая для пояснения декомпозиции), IDEF0-диаграммы ясно объясняют изучаемую систему и причину такой декомпозиции.

Закончив список данных, приступайте с его помощью к составлению *списка функций*. Для этого представьте себе функции системы, использующие тот или иной класс (тип) или набор данных. Помните, что несколько различных типов данных может использоваться одной функцией. Обозначьте, какие типы или наборы данных необходимы для каждой конкретной функции. Это позволит выделить данные сходных типов, которые затем можно объединить в метатипы.

Список функций должен находиться на одной странице со списком данных. При составлении исходного списка не пытайтесь объединять функции между собой. Вместо этого постарай-

тесь вначале сосредоточиться на каждой конкретной функции и ее отношении к группам данных. Кроме того, старайтесь подбирать такие функции, которые могли бы работать с наиболее общими типами данных из вашего списка.

Затем объединяйте функции в «агрегаты». Стремитесь к организации 3—6 функциональных группировок. Старайтесь, чтобы эти группировки имели один и тот же уровень сложности, содержали примерно одинаковый объем функциональности и функции в каждой из них имели сходные операции и цели.

Исходное содержание диаграммы А0 обеспечивают списки данных и функций. Для правильного описания системы содержанию надо придать форму. В IDEF0 это делается посредством построения диаграммы. Начинаящим авторам необходимо придерживаться определенного порядка: (1) расположите блоки на странице, (2) нарисуйте основные дуги, представляющие ограничения, (3) нарисуйте внешние дуги и (4) нарисуйте все оставшиеся дуги. Со временем накопленный опыт позволит вам отойти от этой процедуры и изображать блоки и дуги в соответствии с той идеей, которую вы хотите воплотить в диаграмме.

Правильное расположение блоков является самым важным этапом построения диаграммы. Блоки располагаются в соответствии с их доминированием (по степени важности или по порядку следования). Самый доминантный блок обычно располагается в верхнем левом углу, а наименее доминантный — в нижнем правом. Это приводит к расположению, при котором более доминантные блоки ограничивают менее доминантные, образуя ступенчатую схему. Доминирование имеет важнейшее значение для ясного представления процесса. Например, не имеет смысла говорить о контроле над выполнением задания до изготовления детали.

Затем изображают основные дуги, представляющие ограничения. Это является второй важной частью построения диаграммы А0. Рисуя эти дуги, проверяйте, действительно ли каждая из них оказывает влияние, соответствующее декомпозиции объекта. Проследите по списку данных, не отсутствуют ли какие-то дуги, представляющие ограничения. Если это так, вы, возможно, захотите проверить правильность декомпозиции.

Основными дугами, представляющими ограничения, всегда являются внешние дуги, т. е. дуги, представляющие данные, поступающие из непосредственного окружения диаграммы.

Следующим шагом в построении диаграммы является размещение остальных внешних дуг и назначение им соответствующих ИСОМ-кодов. Таким образом, все данные, входящие в систему или выходящие из нее, оказываются учтенными на рисунке. *Потеря внешней дуги — это ошибка интерфейса, одна из самых распространенных в системном анализе.* Занимаясь декомпозицией объекта, можно забыть об интерфейсных данных, потому что очень легко сосредоточиться на деталях. Начиная с изображения всех внешних дуг, вы повысите точность диаграммы, включив все интерфейсные данные. И, наконец, нарисуйте все остальные дуги, отражающие детали работы системы в целом. Во-первых, нарисуйте оставшиеся ограничения, действующие между блоками. Во-вторых, нарисуйте основной поток данных. В-третьих, уточните обратные связи в потоках данных. В заключение изобразите все обратные связи, вызываемые ошибочными ситуациями.

Здесь следует обратиться к одному очень важному моменту моделирования. На практике оказывается невозможным нарисовать диаграмму сразу набело. Поэтому рекомендуется вначале делать набросок диаграммы, а потом перерисовывать диаграмму набело, чтобы уточнить свое понимание, прояснить ситуацию и создать описание, которое могут посмотреть другие.

Обобщение является последним важным шагом начального этапа моделирования. Помните, что для любой IDEF0-диаграммы есть родительская диаграмма, содержащая ее контекст, где под контекстом понимается блок с набором входных дуг, дуг управления и выходных дуг. Верхняя диаграмма модели (т. е. диаграмма A0) не составляет исключения. Контекстом для нее служит диаграмма A-0, представляющая собой обобщение всей модели. Диаграмма A-0 имеет несколько предназначений. Во-первых, она объявляет общую функцию всей системы. Во-вторых, она дает множество основных типов или наборов данных, которые использует или производит система. В-третьих, A-0-диаграмма указывает взаимоотношения между основными

типами данных, проводя их разграничение. Таким образом, А-0-диаграмма представляет собой общий вид изучаемой системы.

При создании диаграммы А-0 используется информация, уже зафиксированная на диаграмме А0. Вначале в центре IDEF0-бланка рисуют один большой блок, название которого совпадает с названием диаграммы А0. В этот момент следует проверить, адекватно ли название отражает то, что делает система. Все внешние дуги диаграммы А0 изображаются на диаграмме А-0 входящими в соответствующую сторону блока. При этом проверьте, что название каждой дуги описывает то, чем обмениваются система и ее среда. После того как вы изобразите входные и выходные дуги, остановитесь на минуту, чтобы проверить точность описания потока данных. Нарисовав дуги управления, убедитесь, что именно они управляют тем, как система преобразует входные данные в выходные. Наконец, напишите цель и точку зрения модели под основным блоком и сверьте их с тем, что представляется блоком и его дугами.

Построение диаграммы А-0 свидетельствует об окончании начального этапа моделирования. Несмотря на ограниченное число описанных деталей, диаграммы А-0 и А0 представляют законченную картину, потому что они отражают все основные входы, управления, выходы и функции системы. Общий вид системы, полученный с помощью диаграмм А-0 и А0, — основная цель аналитика на начальном этапе построения IDEF0-модели.

2.2 Продолжение моделирования

Начальный этап моделирования включает определение объекта, цели и точки зрения модели, ограничения, накладываемые на объект, построение диаграммы верхнего уровня и ее обобщение, составление списков данных и функций, объединение функций в блоки, формирование с использованием списка данных взаимоотношений между блоками. Продолжение моделирования основывается на тех же методах и выводит модель на следующий уровень детализации. Для этого требуется создать отдельную диаграмму для, возможно, каждого блока диаграммы верхнего уровня, затем построить диаграммы для всех блоков новых диаграмм,

и так до тех пор, пока модель не будет описывать объект с нужной для достижения цели степенью детализации. Таким образом, продолжение моделирования является рекурсивным процессом.

Декомпозиция модели похожа на начальный этап моделирования, но проще его. При декомпозиции модели аналитик всегда находится в контексте, определенном блоком со своими дугами одной из диаграмм. Эта граница, называемая границей объекта, определена двумя способами. Во-первых, объект, цель и точка зрения каждой новой диаграммы уже определены на диаграмме А0. Во-вторых, каждый блок, декомпозируемый в новую диаграмму, уже является ограниченным объектом. Другими словами, он идентифицирует конкретную функцию и все данные, которые для нее требуются или ею порождаются. Строить диаграмму, исходя из этой информации, проще, потому что список данных создается на основе дуг, входящих в блок и выходящих из него, а также потому, что список функций подробно раскрывает название блока. Процесс декомпозиции ограниченного объекта состоит из следующих шагов: выбор блока диаграммы; рассмотрение объекта, определенного этим блоком; создание новой диаграммы; выявление недостатков новой диаграммы; создание альтернативных декомпозиций; корректировка новой диаграммы; корректировка всех связанных с ней диаграмм.

Декомпозиция начинается с чтения диаграммы А0 и определения самого содержательного блока. Это такой блок, декомпозиция которого выявит многие аспекты диаграммы А0 и будет оказывать большое влияние на будущие декомпозиции других блоков этой диаграммы. Новая диаграмма строится аналогично диаграммам А0 и А-0. Блоки размещаются в соответствии с доминированием (т. е. согласно взаимным ограничениям блоков), затем создаются основные дуги, представляющие ограничения, потом внешние и, наконец, внутренние дуги.

Практика показывает, что существенная доля ошибок приходится на интерфейсы. Для IDEF0 интерфейсными являются места соединения диаграмм со своими родителями. Вот почему каждую декомпозицию необходимо аккуратно соединять со своим родителем. Дуги выражают связи между блоками. Их вычерчивают не для показа последовательности действий. Они отра-

жают отношения между блоками, независимые от потенциального следования. Такой механизм приводит к реализации различных сценариев, активизируя блоки в различные моменты времени в зависимости от ситуации.

2.3 Проверка диаграммы автором

Построив диаграмму, попытайтесь самостоятельно выявить ее недостатки. Процесс авторской проверки дает новое направление работе — определение ее качества. На этапе декомпозиции возникает диаграмма, которая декомпозирует блок и его дуги. Аналитик пытается объяснить объект самому себе. Неудивительно, что результат может оказаться малодоступным для других. В работе, естественно, появляются жаргон и неявно подразумеваемые факты. При критической оценке аналитик абстрагируется от своей работы. Это позволяет взглянуть свежим взглядом на диаграмму с тем, чтобы информация, которую она несет, стала доступной не только ее автору, но и другим людям.

Вначале следует критически оценить блоки диаграммы. Определим функциональные аспекты диаграммы, задавая вопросы типа:

- Представляют ли блоки содержательную декомпозицию функции?

- Не выглядит ли диаграмма запутанной?
- Все ли блоки соответствуют точке зрения модели?
- Несут ли блоки достаточный объем новой информации?

Теперь зададим вопросы о связи диаграммы с ее родителем. При этом проверим, как диаграмма вписывается в модель.

- Все ли внешние дуги имеют ISOM-коды?
- Все ли ISOM-коды соединяют дуги с одним и тем же значением?

Вопросами о внутренних дугах обычно заканчивают поиск ошибок в диаграмме. Теперь, после разрешения основных вопросов, следует проанализировать детали диаграммы. Мы можем задать вопросы типа:

- Не слишком ли много внутренних дуг?
- Нет ли блоков без дуг управления?

- Нет ли блоков без выходных дуг?
- Все ли важные обратные связи отражены?

Корректировка новой диаграммы. Обычно, потратив время на вопросы к диаграмме, тестирование, выполнение альтернативных набросков, автор корректирует диаграмму. В ходе корректировки следите за правильным доминированием, выбором названий блоков, информативностью дуг и делайте пояснения. Помните, что теперь вы рисуете диаграмму, чтобы донести информацию в точном и понятном виде до читательской аудитории.

Для блоков обычно стараются выбрать содержательные названия. Однако в IDEF0 нет необходимости выражать все с помощью названий блоков, потому что о работе блока многое сообщают метки окружающих его дуг.

Рисуя дуги, старайтесь располагать их аккуратно, минимизируя число пересечений и максимизируя пространство между ними. Правильное графическое расположение вносит большой вклад в повышение наглядности и понятности диаграммы. Помечайте дуги ясно и точно. Хотя определенное количество слов передает информацию лучше. Вычерчивая дуги в порядке их значимости, вы сможете оценивать их в процессе рисования и избежите стремления механически присоединять все дуги ко всем блокам.

Закончив построение диаграммы, поясните ее важные аспекты с помощью замечаний или дополнительного материала. Проясняйте только те понятия, которые нельзя изобразить в виде блоков и дуг.

Исправление взаимосвязанных диаграмм. Создание диаграммы, ответы на связанные с ней вопросы и переделка ее обеспечивают более глубокое понимание родительской диаграммы и диаграмм-потомков вновь построенной диаграммы. Зафиксируйте свое понимание во время исправления диаграммы. Вам придется переносить информацию на другие диаграммы в трех ситуациях: при изменении меток внешних дуг, при появлении новых внешних дуг и при перераспределении функций. Перенесение необходимо, если изменилось название внешней дуги. Перенесение измененных меток внешних дуг немедленно обеспечива-

ет предоставление родительской диаграммой всех данных, необходимых диаграмме-потомку. Перенесение необходимо также, когда на новой диаграмме появляются новые входные или выходные дуги. Эти новые дуги должны, так или иначе, возникнуть на родительской диаграмме. Есть два пути сделать это: нарисовать новые дуги на родительской диаграмме или объединить дуги новой диаграммы в одну и изменить соответствующим образом метку дуги на родительской диаграмме. Делая это, соблюдайте правила соединения и разветвления дуг.

Перемещение блоков представляет самую сложную ситуацию. Оно происходит, когда функция (обычно на низком уровне модели) должна появиться, но не появляется на диаграмме, которую вы рисуете, а появляется на другой диаграмме модели, или наоборот. Перенести такую функцию, представленную блоком и всеми его дугами, с одной диаграммы на другую — нелегкое дело. Обычно это приводит к большим изменениям в метках дуг, появлению множества новых и исчезновению некоторых старых дуг. Иногда перемещение одного блока ведет к перемещению других блоков на различные диаграммы, вызывая целую волну изменений. Как правило, перемещение блока влечет за собой обилие технически сложной работы и может привести к ошибкам, если изменения не отслеживаются достаточно тщательно.

2.4 Завершение моделирования

Одна из наиболее частых проблем, возникающих в процессе реализации IDEF0-проектов, — когда же следует завершить построение конкретной модели? На этот вопрос не всегда легко ответить, хотя существуют некоторые эвристики для определения разумной степени полноты. В этом параграфе представлены правила, которыми пользуются опытные IDEF0-авторы для определения момента завершения моделирования. Однако хотелось бы предупредить, что приведенные здесь правила носят характер рекомендаций. Иногда даже опытные IDEF0-авторы, применив эти правила, обнаруживают впоследствии, что приняли неверное решение. Только длительная практика позволит вам приобрести знания, необходимые для принятия правильного решения об окончании моделирования.

Прекращение декомпозиции. Как правило, большинством IDEF0-авторов рекомендуется прекращать моделирование, когда уровень детализации модели удовлетворяет ее цель. Другими словами, вы должны закончить моделирование, когда почувствуете, что дальнейшее продвижение не будет удовлетворять информационные потребности проекта или вступит с ними в противоречие. Хотя интуитивно это правило понятно, ему трудно следовать, не оценив модель. В первое десятилетие использования IDEF0 для создания моделей в различных прикладных областях были разработаны некоторые критерии для определения момента завершения моделирования. Этот опыт показал, что для отдельной модели, которая создается независимо от какой-либо другой модели, декомпозиция одного из ее блоков должна прекращаться, если:

- 1) блок содержит достаточно деталей;
- 2) необходимо изменить уровень абстракции, чтобы достичь большей детализации блока;
- 3) необходимо изменить точку зрения, чтобы детализировать блок;
- 4) блок очень похож на другой блок той же модели;
- 5) блок очень похож на блок другой модели;
- 6) блок представляет тривиальную функцию.

Эти правила подчеркивают практические аспекты применения IDEF0 для описания систем реального мира с конкретной целью (например, понять работу телефонной станции, чтобы определить требования к ее программному обеспечению).

Одна из типичных ситуаций, встречающихся в конце моделирования, — это блок, который описывает систему с нужным уровнем подробности. Проверить достаточность деталей обычно совсем легко. Просто спросите себя, отвечает ли блок на все или на часть вопросов, составляющих цель модели. Если блок помогает ответить на один или более вопросов, то дальнейшая декомпозиция может не понадобиться.

Принятие решения о завершении моделирования. Вероятность принятия неправильного решения о завершении моделирования может быть уменьшена, если вы оцените каждый блок, который хотите детализировать, в соответствии с приведенными выше правилами. Поскольку большинство IDEF0-моделей обыч-

но содержат от 10 до 320 диаграмм, лучшее время для начала оценки блоков, когда модель достигает этих размеров. Если какая-то часть модели достигла уровня, не требующего дальнейшей декомпозиции, обращайтесь к своему собственному критерию для определения момента завершения моделирования, прежде чем декомпозировать блок, который еще не был детализирован.

Помните, что критерии, приведенные в этой главе, носят характер рекомендаций, следовательно, их нельзя применять механически. Иногда может случиться так, что в какой-то конкретной ситуации два правила будут противоречить друг другу. Разрешение таких конфликтов требует рассудительности и осмотрительности. А это результат накопленного опыта. Кроме того, мы советуем вести записи по сложным вопросам, по тому, как они были решены и что этому способствовало. Такие записи помогут вам впоследствии, если окажется, что моделирование следовало закончить раньше. Они помогут также при определении и выборе объектов будущего моделирования.

2.5 Дополнения к диаграммам и моделям

Одно из достоинств IDEF0-модели заключается в способе организации и представлении информации. Диаграмма, находящаяся на вершине модели, обобщает всю рассматриваемую систему. Диаграммы первого уровня представляют важнейшие подсистемы с их взаимосвязями, а диаграммы самого нижнего уровня представляют детализированные функции, с помощью которых, собственно, и работает система. Диаграммы законченной IDEF0-модели упорядоченно организуют все важные компоненты и детали системы. Опытные аналитики, используя преимущества организации, создают различные дополнения к ней. Дополнения и уточнения, которые не входят в сами диаграммы, обогащают информационное содержание модели. Поскольку дополнительная информация формально не является частью модели, IDEF0 рекомендует помещать такие материалы на отдельных страницах и соединять их с диаграммами модели.

Дополнения к диаграммам. Квалифицированные IDEF0-аналитики придают конкретную направленность дополнительной информации, чтобы выделять некоторый аспект или часть от-

дельной диаграммы с помощью их дополнительного описания. Таким образом, они повышают отдачу текстовых записей и графики, причем для этого требуется лишь небольшое количество деталей (модель обеспечивает контекст, необходимый для связи дополнительной информации с системой).

IDEF0-диаграммы могут быть дополнены информацией в виде текстов, рисунков и глоссариев. Текст обычно представляет собой рассказ об одной из частей диаграммы. Рисунки — это картинки, поясняющие отдельные моменты. Глоссарий — набор определений объектов и функций, представленных на диаграмме.

Дополнительная информация записывается или представляется на стандартных IDEF0-бланках. Поскольку дополнения уточняют конкретную диаграмму модели, для идентификации и связывания дополнительной страницы с диаграммой, к которой она относится, используется принятая в IDEF0 схема нумерации узлов. К номеру узла диаграммы добавляется буква и целое число. Буква определяет тип дополнения (Т — текст, Р — рисунок и Г — глоссарий), а число означает порядковый номер этой текстовой страницы среди других дополнительных страниц данной диаграммы.

Дополнение моделей. Иерархические наборы IDEF0-диаграмм, называемые «моделями», вводят объект описания и уточняют его регулярным, управляемым и понятным образом. Это обеспечивается тем, что диаграммы модели всегда организованы в соответствии с порядком нумерации узлов. Это означает, что первым идет узел А-0, вторым — узел А0, третьим — узел А1, четвертым — узел А11 и т. д. Такой порядок расположения IDEF0-диаграмм является копией «древовидной» структуры, часто встречающейся в математике и информатике. Полная IDEF0-модель обычно читается двумя способами. Первый способ — обзорное чтение, когда читаются все диаграммы верхнего уровня, прежде чем перейти к диаграммам следующего. Это называется чтением «в ширину». Второй способ — детальное чтение, когда читают отдельную ветвь дерева вплоть до диаграмм самого нижнего уровня. Это называется чтением «в глубину».

Чтобы помочь читателям правильно двигаться по древовидному набору диаграмм, разработаны дополнительные средства.

Примерами таких средств могут служить указатель диаграмм и указатель узлов. Они представляют собой составленные с отступами списки узлов или диаграмм (по типу оглавления), определяющих содержание модели. В указателе диаграмм перечисляются все диаграммы модели с приведенными (в отдельной строке) названием и номером узла каждой диаграммы. В указателе узлов перечисляются все блоки модели, причем в каждой отдельной строке записываются название и номер узла соответствующего блока. Таким образом, указатель узлов является просто более подробным, чем указатель диаграмм, списком.

3 АВТОМАТИЗАЦИЯ ПОСТРОЕНИЯ МОДЕЛИ

Пакет Design/IDEF (Meta Software Corp.) — графическая среда для проектирования и моделирования сложных систем широкого назначения, поддерживающая методологии описания и моделирования системных функций (IDEFO/ IDEF0), структур и потоков данных в системе (IDEF1, IDEF1X, E-R) и поведения системы (IDEF/CPN). Рассмотрим более подробно основные возможности пакета Design/IDEF.

Представление графики. Design/IDEF имеет быструю и высококачественную графику, включающую создание стандартных и пользовательских объектов, выравнивание и манипулирование объектами, выбор атрибутов графических объектов и текста. Дополнительно в Design/IDEF реализованы возможности, требуемые для редактирования и моделирования данных: построение связывающих линий типа «резинка», маршрутизация и сглаживание дуг и т.д.

Обеспечение непротиворечивости модели. Design/IDEF имеет встроенные возможности, дающие уверенность разработчику, что IDEF-модель будет точной, целостной и непротиворечивой на протяжении всего цикла ее создания. Например, при модификации текста, принадлежащего функциональному блоку или дуге в какой-то одной части модели, текст будет динамически скорректирован на всех страницах модели.

Поддержка Словаря Данных. Design/IDEF имеет встроенный Словарь Данных, который позволяет хранить информацию и создавать отчеты о функциях и потоках данных в IDEF-модели. Словарь дает возможность определять начальную информацию об объектах и предоставляет разнообразный набор функций сопровождения, восстановления и сохранения целостности файлов данных. Возможности словаря отличаются большой гибкостью и позволяют пользователю вводить неограниченное число параметров для каждого объекта. В сочетании с высококачественной печатью на лазерном принтере это позволяет разработчику созда-

вать документацию проекта, отвечающую самым высоким требованиям.

Генерация отчетов. Design/IDEF предоставляет возможность использовать пять видов отчетов для поддержки и анализа моделей:

- Отчет о контроле полноты модели.
- Отчет о функциях.
- Отчет о дугах.
- Отчет о ссылках.
- IDEF-отчет.

Все отчеты могут быть показаны на экране компьютера, отредактированы и распечатаны с помощью текстового редактора. Design/IDEF анализирует и отбирает данные для генерации текстового файла, содержащего информацию о диаграммах и Словаре.

Информация, содержащаяся в отчетах, может быть экспортирована для использования в других программах, таких как, например, электронные таблицы, настольные издательские системы и текстовые редакторы.

Организация коллективной работы. Design/IDEF поддерживает работу многочисленной группы разработчиков, создающих одновременно большую и сложную IDEF-модель. Подмодели легко интегрируются в одну большую модель.

Моделирование данных (IDEF1, IDEF1X и E-R-методологии). Design/IDEF дает также возможность создавать информационные модели, которые представляют как собственно данные, так и связи между ними в системе.

Информация, содержащаяся в IDEF-моделях, экспортируется в любую базу данных, а сами модели могут быть экспортированы в Design/CPN — пакет динамического моделирования и анализа сложных систем.

Как CASE-пакет по разработке программного обеспечения Design/IDEF поддерживает первые стадии создания программного продукта:

- Формулировку требований и целей проекта — определение того, что проектируемая система будет делать.

- Разработку спецификаций — формализованное описание требований.
- Создание проекта — определение подсистем и взаимодействий между ними.
- Документирование проекта — создание базы данных проекта, текстуальное описание составных частей проекта.
- Анализ проекта — проверка проекта на полноту и непротиворечивость.

Результатом работы пакета Design/IDEF является проект программной системы, состоящий из двух частей:

1. Проект функциональной структуры системы, содержащий иерархически связанные страницы с IDEF0-диаграммами и описывающий все модули (вплоть до элементарных функций) системы, их взаимосвязи, входные и выходные параметры.

2. Проект информационной структуры системы — логической модели ее базы данных, — описывающей все структуры и взаимосвязи данных.

Оба проекта проверяются на полноту и непротиворечивость, сопровождаются базой данных проекта и документацией.

4 ОПИСАНИЕ РАБОТЫ С DESIGN/IDEF

4.1 Создание диаграмм методологии IDEF0

Данный программный продукт является самым доступным и бесплатным средством автоматизации построения функциональных моделей согласно методологии IDEF0.

Установка и запуск Design/IDEF 3.7. Установка Design/IDEF не требует специальных инсталляционных пакетов. Все необходимые файлы расположены в каталоге IDEF37, который может быть расположен в любом месте жесткого диска, и пользователь должен иметь права записи на этот диск. Необходимое свободное пространство на диске должно составлять около 3.5 Мб. Более подробную информацию об отличительных особенностях данной версии от предыдущих можно получить в файле `rln37idf.wri`, расположенном в каталоге IDEF37¹. Руководство по использованию (`help`) так же расположено в каталоге IDEF37 в файле `idef37.hlp`.

Для запуска Design/IDEF необходимо в каталоге IDEF37 выбрать и запустить файл `idef.exe`.

Для корректного запуска Design/IDEF в среде Windows 7 необходимо установить виртуальную машину с Windows XP. В самом простейшем варианте это можно достичь, установив Windows Virtual PC и Windows XP Mode с официальных ресурсов Microsoft. Данные программные продукты распространяются бесплатно.

Создание новых диаграмм. Для создания новой диаграммы выберите пункт меню *File/New* или нажмите клавиши *Ctrl-N*. В появившемся окне (рис. 4.1) выберите из списка методологию (Methodology) IDEF0 и нажмите клавишу *OK*.

¹ Вся информация в каталоге представлена на английском языке.

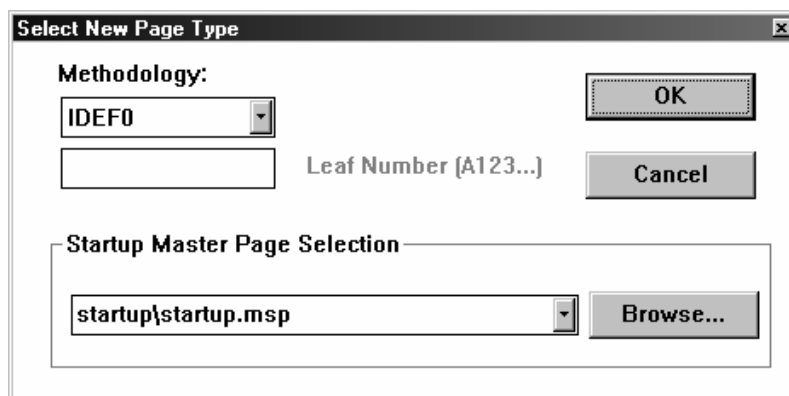


Рис. 4.1 — Создание новой диаграммы

В результате проделанного действия будет создан чистый лист диаграммы А-0, с одним блоком в центре. В правом нижнем углу блока будет подписан его идентификатор А0. Перед началом работы с проектом давайте заполним информационные графы диаграммы об этом проекте, такие как автор, проект. Для этого выберите в меню *Select* пункт *Page* или нажмите клавишу *F4*. В появившемся списке выберите мастерскую страницу — *Master P10000* (рис. 4.2).



Рис. 4.2 — Список страниц диаграммы

Мастерская страница содержит три секции (рис. 4.3):

- поле рабочей информации в верхней части страницы;
- поле сообщений, в котором рисуется диаграмма в центре;
- поля идентификации вдоль нижнего края страницы.

Заполните графы *AUTHOR* — автор и *PROJECT* — проект. Для ввода и редактирования текстовой информации используется указатель метки — специальные объекты без границ, размеры которых определяются текстом, напечатанным внутри них².

Выберите команду *Label* в меню *Create* (Указатель изменит форму на L). Поместите указатель метки в блок, находящийся слева от поля Рабочая версия, и отработайте кнопкой мыши, чтобы установить точку вставки. Поле состояния показывает, что текстовый режим сейчас включен.

² Перед началом ввода русского текста, необходимо в окне *IDEF Attributes*, которое вызывается через пункт меню *Edit/Set Attributes...* (*Shift-Ctrl-A*), установить русские шрифты для всех примитивов диаграммы.

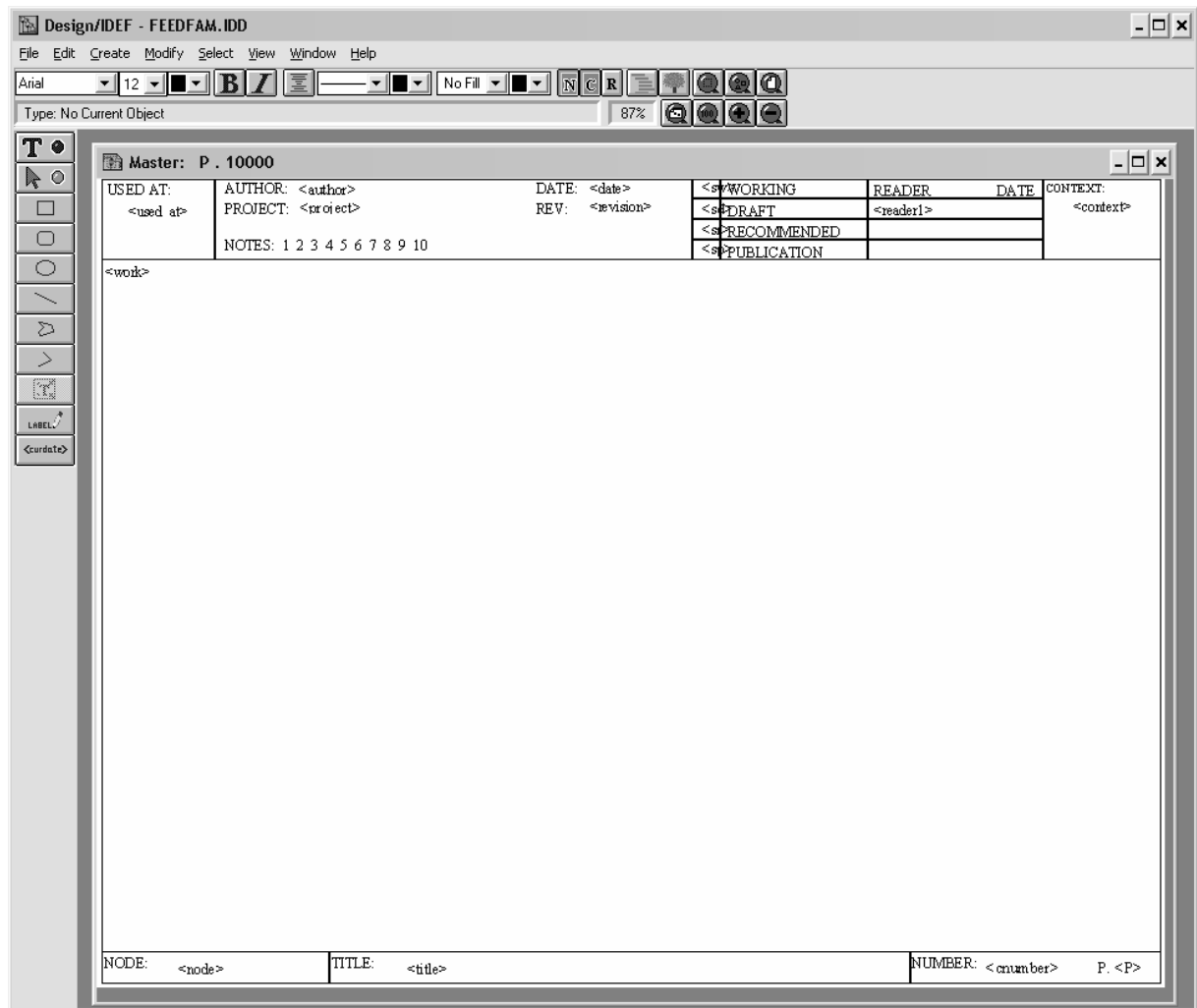


Рис. 4.3 — Мастерская страница Design/IDEF

Введите символ *X* слева от поля *Working* — *рабочая версия*. Отказом от продолжения выполнения режима работы является нажатие на *Escape*. Аналогичным образом заполните поля *AUTHOR* и *PROJECT*, после чего нажмите клавишу *Escape*, чтобы закончить создание меток.

На границах метки появятся черные квадратики (хэндлеры) как признак выделения, а указатель примет форму графического указателя.

Для изменения места положения метки:

- Выделите метку, отработав кнопкой мыши.
- Удерживая кнопку, переместите метку в новую позицию.
- Отпустите кнопку.

В диаграмме мастера заполните описанные выше поля путем выбора их указателем мыши. Далее таким же способом переключо-

читесь в диаграмму А-0, в которой вы увидите что поля, заполненные в мастере, приобрели те же значения и в диаграмме А-0.

Для того чтобы сохранить изменения, сделанные в диаграмме, выберите пункт меню *File/Save* или *File/Save As...* (сохранить файл с новым именем). Для печати текущей диаграммы служат пункты меню *File/Print...* (рис. 4.4) и *File/Print Setup...* (рис. 4.5).

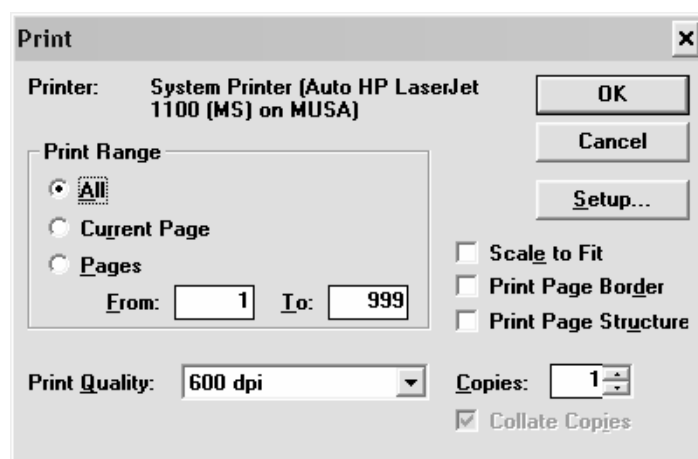


Рис. 4.4 — Окно печати диаграммы

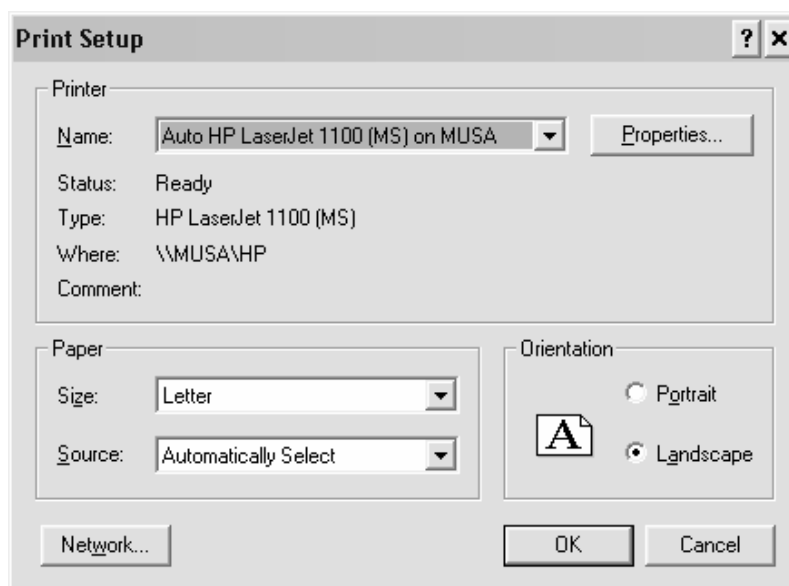





Рис. 4.5 — Окно установок печати

Создание текста в IDEF-блоках. Для ввода текста в блок А-0 необходимо включить текстовый режим. Выберите *Turn On Text* в меню *Modify* или нажмите клавишу *F2* (аналогичный результат может быть достигнут нажатием кнопки  в панели

инструментов, расположенной слева от диаграммы). Напечатайте в блоке A0 «Разработать программный продукт». Выберите *Turn Off Text* в меню *Modify* или нажмите клавишу *F2* (аналогичный результат может быть достигнут нажатием кнопки  в панели инструментов).

Стандартная контекстная IDEF-диаграмма включает формулировки цели и точки зрения модели. Введите их как метки. Выберите *Label* в меню *Create* или нажмите клавишу *F3* (аналогичный результат может быть достигнут нажатием кнопки  в панели инструментов). Указатель примет форму указателя метки.

Отработайте кнопкой ниже A-0-блока.

Напишите: «Цель: Организовать процесс разработки программного обеспечения».

Нажмите *Enter* и напишите: «Точка зрения: Руководитель разработки».

Нажмите *Escape*, чтобы закончить создание метки. Передвиньте метку в нижнюю часть страницы.

Создание IDEF-дуг. Дуги рисуются с помощью команды *Create/Arrow* и могут быть созданы только между блоком и другим блоком и блоком и меткой. Создайте и разместите метки аналогично тому, как показано на рис. 2.3 (диаграмма A-0).

Выберите *Create/Arrow*. Поместите указатель внутрь метки «Исследования рынка» около ее правой границы. Нажмите кнопку мыши и, не отпуская ее, переместите указатель до левой стороны A0-блока. Отпустите кнопку мыши, чтобы закончить создание дуги.


Указатель активен, пока не отмените его нажатием на *Esc*.



Аналогичным образом создайте дуги из других меток:

- Требования пользователей.
- Системные требования.
- Успешный программный продукт.
- Группа разработки.
- Оборудование.

Нечто похожее на рис. 2.3 должно получиться и у Вас.

Создание диаграммы первого уровня (A0). Каждая диаграмма в Вашей модели может быть названа и иерархически связана.

Для создания подстраницы диаграммы изображения следующего уровня необходимо выделить декомпозируемый блок. Выбрать в пункт меню *Create/Decompose* или нажать клавишу *Shift-PgDn* (кнопка панели инструментов ).

Команды *Select/Parent* (кнопка панели инструментов ) , *Select/Child* (кнопка панели инструментов ) , *Select/Page* позволяют перемещаться по иерархии диаграммы.


Выберите блок A0.


Выберите *Page/Decompose*. К диаграмме добавится новая подстраница. Для перехода на подстраницу выберите *Page/Child*. (Можно *Shift+Dn*, или двойной щелчок мыши.) Атрибуты подстраницы автоматически будут заполнены.

Текст меток («портовых узлов») блока A0 переносится на созданную подстраницу по ее краям соответственно расположению в родительском блоке.


Если IDEF-страница не помещается целиком на экране и Вам необходимо уменьшить ее размеры, а потом увеличить, то можно воспользоваться пунктом меню *View*. Приведем команды данного меню:

1. *Zoom...* — показывает окно, где можно установить параметры отображения диаграмм от 10 % до 400 %, и позволяет сохранить эту установку для будущих страниц.


2. *Zoom To Area* — позволяет увеличить выбранный фрагмент во всю просмотрную область — кнопка .

3. *Fit Page* — регулирует границы активной страницы так, что они приспособляются полностью к просмотрной области, — кнопка .


4. *Fit Page All* — регулирует границы всех открытых страницы так, что они приспособляются полностью к просмотрной области.


5. *Fit Object* — увеличивает или уменьшает размер страницы, так что объекты заполняют просмотрную область, — кнопка .

6. *Fit Object All* — увеличивает или уменьшает размеры всех открытых страниц, так что объекты заполняют просмотрную область.

7. *100 %* — отображает активную страницу в 100 %-ом масштабе — кнопка .

8. *100 % All* — отображает все открытые страницы в 100 %-ом масштабе.

9. *Enlarge* — увеличивает активную страницу пошагово вплоть до максимума 400 % — кнопка .

10. *Reduce* — уменьшает активную страницу пошагово вплоть до минимума 10 % — кнопка .

Размещение IDEF-блоков. Диаграмма первого уровня будет содержать три функции:

- Планирование и проектирование разработки продукта.
- Создание программ и документации.
- Маркетинг.

Команда *Create/Place Boxes...* — создаст и разместит заданное число блоков по диагонали страницы. Выберите *Create/Place Boxes...* Появится диалоговое окно, в котором выделено число 3, которое можно исправить до максимального числа. (Изменяется в *Edit/Set Options...* — параметр *Activities, Maximum Boxes* (рис. 4.6).) Отработайте кнопкой мыши на *OK* для размещения 3-х блоков вдоль диагонали страницы.

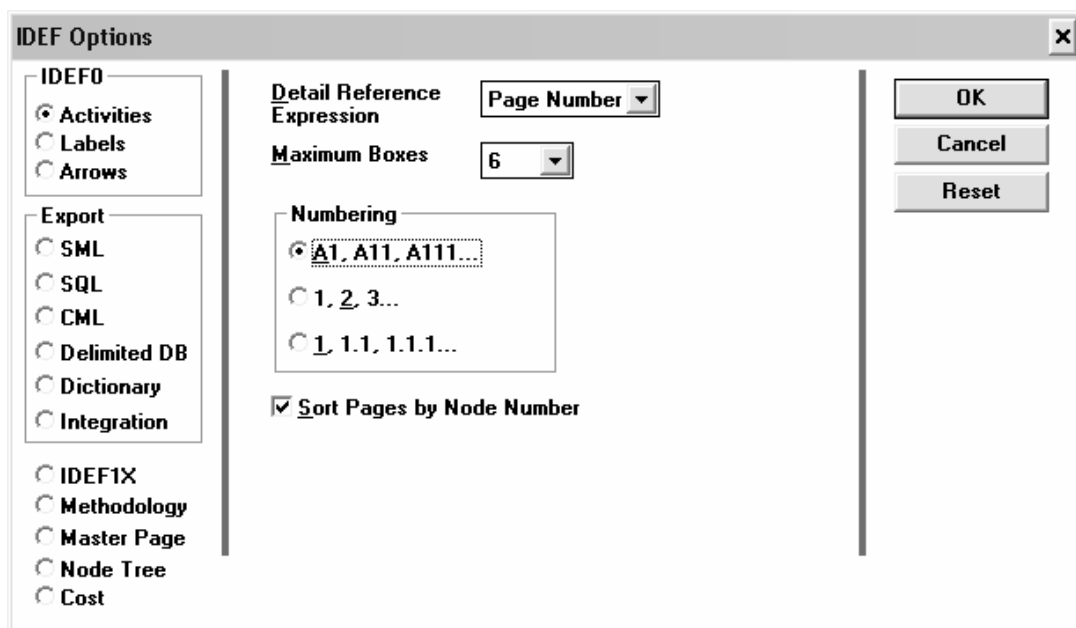


Рис. 4.6 — Окно изменения параметра *Activities*

Блоки нумеруются автоматически в соответствии с установками в *Edit/Set Options...* — параметр *Activities, Numbering*.

Выделите блок А1, чтобы вписать текст в блок, перейдите в режим ввода текста. Напечатайте: «Планирование и проектирование разработки продукта».

Выделите блок А2. Напечатайте: «Создание программ и документации».

Выделите блок А3. Напечатайте: «Маркетинг». Отключите режим ввода текста.

Выберите входной портовый узел «Исследования рынка» и расположите его слева от блока А1.

Выберите *Create/Arrow*. Не отпуская кнопки мыши, перемещайте указатель от правой стороны портового узла к центру левой стороны блока А1. Нажмите *Esc*, чтобы закончить создание дуг.

Поместите, как показано на рис. 2.4, оставшиеся портовые узлы: управляющий, механизма и входной и нарисуйте соответствующие дуги. Нажмите *Esc*, чтобы закончить создание дуг.


Создание ломаных дуг. Параметр *Arrows, Automatic Routing* в *Edit/Set Options...* соединяет блоки и метки, не расположенные на одной горизонтали или вертикали, с помощью ломаных дуг с прямыми углами.

Выберите *Create/Arrow*. Не отпуская кнопку мыши, соедините правую сторону блока А1 с левой стороной блока А2. Отпустите кнопку мыши, чтобы указать точку входа дуги. Аналогично нарисуйте вторую ломаную дугу от правой стороны А2 к левой стороне А3. Нажмите *Esc*.

Создание присоединенных меток. Чтобы пометить выходные дуги блоков А1 и А2, создадим для каждой дуги присоединенную метку.

Выберите *Create/Label*. Отработайте кнопкой мыши чуть правее вертикального сегмента дуги, идущей от блока А1 к блоку А2.

Напечатайте: «Проект системы, план документации». Нажмите *Esc*.

Выберите *Create/Attach Label* или нажмите кнопку панели инструментов . В поле состояние появится *Select Arrow for attach*. Поместите указатель на вертикальный сегмент дуги напротив метки. Отработайте кнопкой мыши на дуге.

Аналогично пометьте дугу A2-A3 меткой «Программный продукт, документация».

Помещение дуги в туннель. Вторая дуга, соединяющая блоки A1 и A2, не имеет отношения к следующему более низкому уровню модели, и ее необходимо поместить в т.н. «туннель».

Выделите дугу «План работы программистов». Выберите *Create/Tunnel*. Пометьте поле, соответствующее *Arrow Head*, в диалоговом окне *Tunnel Arrow* (рис. 4.7). Отработайте кнопкой мыши на *Ок*.

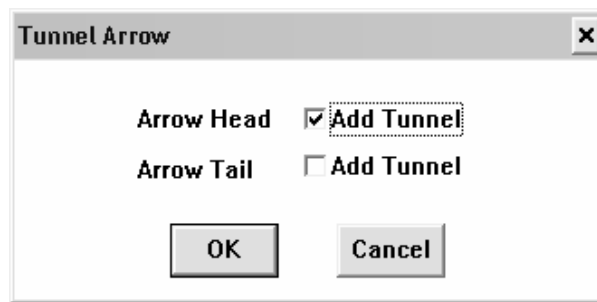


Рис. 4.7 — Окно помещения дуги в туннель

Декомпозируйте блоки A1 и A2 на функции, как показано на рис. 2.5—2.6.

Изменение текстовых меток на ICOM-метки. Перед тем как произвести декомпозицию блока A1, посмотрим, к чему приведет изменение параметра *Labels, Label Transfer* с *Text Label* (текстовые метки) на *ICOM Code* и наоборот (Данный параметр расположен в окне *IDEF Options* (рис. 4.8) и вызывается командой *Edit/Set Options...* или *Ctrl-Shift-O*.)

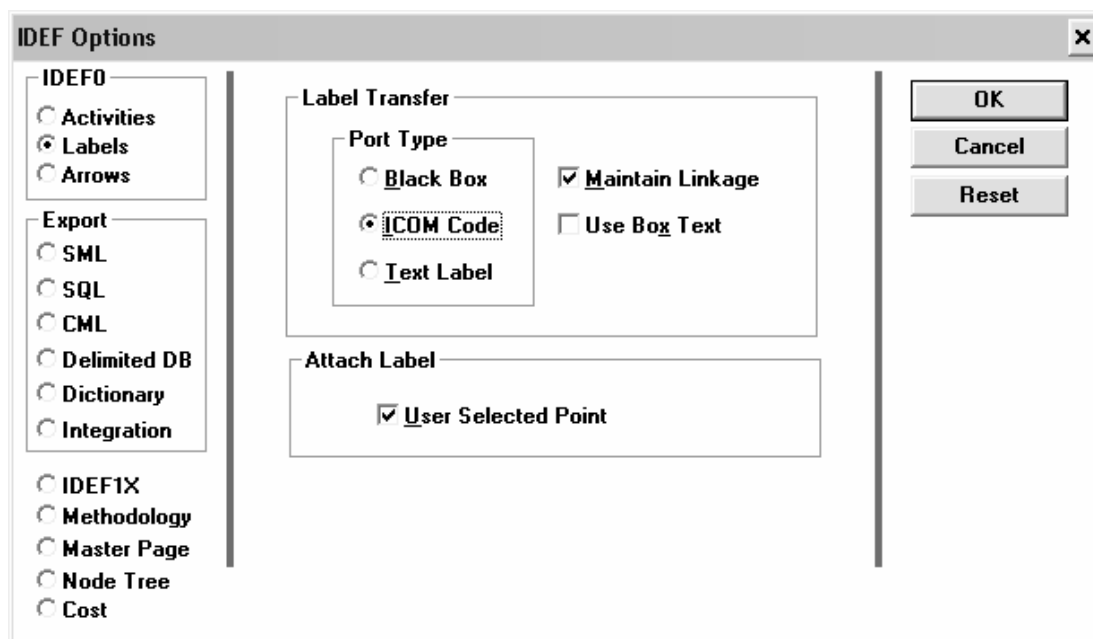


Рис. 4.8 — Окно изменения параметра *Labels*

В случае значения *Text Label* портовые узлы будут содержать текстовое название метки.

В случае значения *ICOM Code* портовые узлы будут содержать ICOM-код. Текст метки для соответствующей дуги будет помещен в структуру, которая называется «область» и является подчиненной соответствующему портовому узлу на диаграмме декомпозиции. Система обозначений состоит из букв, показывающих роль родительской дуги: I-INPUT, C-CONTROL, O-OUTPUT, M-MECHANISM. Номер после буквы указывает позицию дуги в группе дуг, выполняющих ту же роль.

Выберите команду *Edit/Set Options...*. Измените установку *Labels*, *Label Transfer* с *Text Label* на *ICOM Code*. Отработайте кнопкой мыши на *Ok*.

Рисование от портовых ICOM-узлов. Выделите и передвиньте портовый узел I1 влево от блока A11. Выберите *Create/Arrow*. Проведите дугу от портового узла I1 к входной левой стороне A11, затем нажмите *Esc* для завершения создания дуги. Поместите портовый узел C1 над блоком A11 и проведите дугу от C1 к A11.

Автоматическое размещение дуг используется для равномерного размещения дуг вдоль стороны блока.

Выберите команду *Edit/Set Attributes ...* и в окне *IDEF Attributes* установите параметр *Arrow, Autospace Arrows*. Отработайте кнопкой мыши на *Ок* (рис. 4.9).

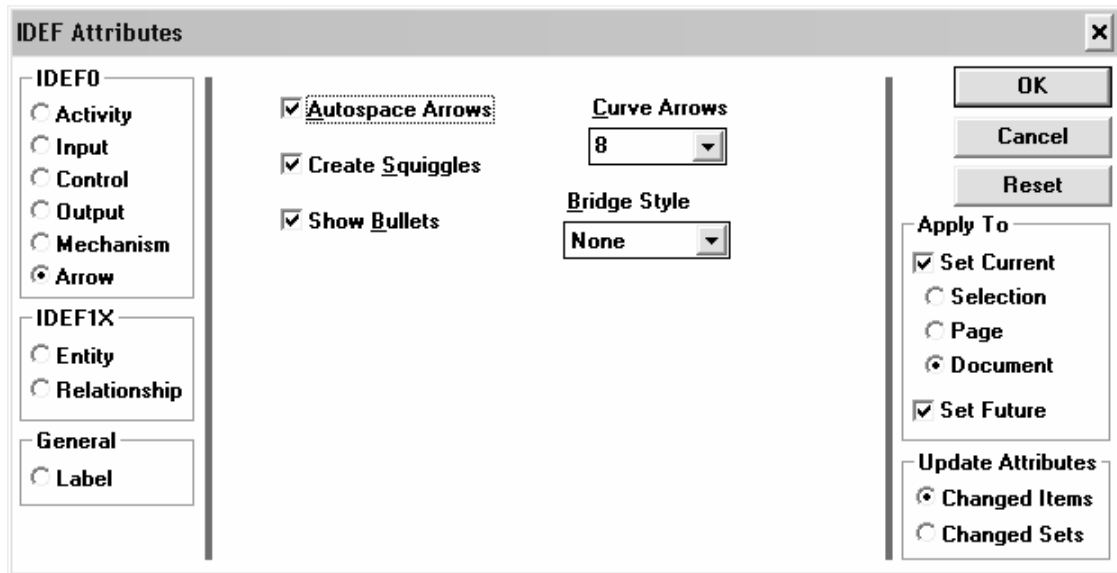



Рис. 4.9 — Окно установки параметров *Arrow*

Выберите *Create/Arrow*. Проведите дугу от правой стороны блока A11 к левой стороне A12 — центрирование будет выполнено автоматически. По вашему желанию вы можете оставить включенным данный параметр или отключить его.

Создание дополнительного блока на диаграмме. Выберите пункт меню *Create/IDEF Box*, или нажмите клавишу *F11*, или щелкните кнопкой мыши по кнопке , расположенной на панели инструментов. Установите указатель на диаграмме, куда Вы хотите поместить блок. Отработайте кнопкой мыши, и будет создан блок с размерами по умолчанию. Впишите его название. Отключите текстовый режим. Теперь необходимо внести перенумерацию блоков, для этого воспользуйтесь командой *Renumber Box...* меню *Edit* (блок должен быть выбран).

Теперь необходимо выровнять блоки и раздвинуть их вдоль диагонали.

Выделите и переместите блок A11 так, чтобы он частично накрыл блок A12. Выделите блок A12. Выберите *Modify/Align/Between*. Появится сообщение в поле состояние для указания блока, области или контура страницы в качестве ориентира для выравнивания.

Выделите A13. Сообщение предлагает выделить второй ориентир для выравнивания.


Выделите A11. Блок A12 будет размещен между A11 и A13.

В пункте меню *Modify/Align* есть и другие возможности для выравнивания.

Чтобы равномерно расположить все блоки на странице, надо сгруппировать их и использовать команды раздвижения из меню *Modify/Spread*. Выберите указателем мыши необходимые блоки, одновременно нажимая клавишу *Shift*. Вначале изменим размеры первых 4-х блоков до размера A15, который Вы увеличите. A15 должен быть отмечен последним в группе. Выберите *Modify/Same Size/Width and Height*. Результат: все члены группы совпадают с A15 по размеру.

Раздвижение блоков (команда *Modify/Spread*). У нее может быть три варианта:


- *Modify/Spread/Spread Horizontal*.
- *Modify/Spread/Spread Vertical*.
- *Modify/Spread/Spread Diagonal*.

Создание разветвлений. Выход блока A12 должен поступать на вход блоков A13 и A15. Для этого необходимо нарисовать выходную дугу блока A11 к входной стороне A13. Сохраняя выделение этой дуги, выберите *Create/Branch*, или нажмите комбинацию клавиш *Ctrl-Y*, или выберите кнопку в панели инструментов . Появится сообщение на указание блока или метки для разветвления. Поместите указатель на входную сторону блока A15 и отработайте кнопкой мыши. Ветвь дуги проведена. Создайте метки для дуг, как на рис 2.5. Присоедините их к соответствующим ветвям дуг.

Теперь создадим разветвляющуюся дугу от портового узла M1. Проведите дугу от портового узла M1 к нижней стороне блока A12. Выберите команду *Create/Branch*. Выделите сторону ме-

ханизма в A11, указав место присоединения разветвленной дуги. Создайте третью ветвь и присоедините ее к нижней стороне A13. Создайте и присоедините четвертую ветвь к нижней стороне A14, произведите то же самое действие и к блоку A15. Для каждой из ветвей из M1 создайте и присоедините метки (рис. 2.5).

Создание соединений дуг. Следующими должны быть нарисованы дуги, соединяющие выход O2 с двумя блоками A13 и A15.

Выберите *Create/Arrow*. Проведите дугу от правой стороны A13 к O1. Нажмите *Esc*. Сохраняя выделение дуги, выберите *Create /Join*, или нажмите комбинацию клавиш *Ctrl-J*, или выберите кнопку в панели инструментов . Появится сообщение указания блока или метки для объединения. Поместите указатель на середину правой стороны блока A15. Контур блока начал мерцать. Отработайте кнопкой мыши.

Сглаживание дуг. Выберите команду *Edit/Set Attributes ...* и в окне *IDEF Attributes* установите параметр *Arrow, CurveArrows* в значение 8 (рис. 4.9). Отработайте кнопкой мыши на *Ок*. Все углы ломаных стали закругленными.

Для того чтобы углы стали прямыми, не обходимо ввести значение 0.

Создание мостов. Выберите команду *Edit/Set Attributes ...* и в окне *IDEF Attributes* (рис. 4.9) установите параметр *Arrow, Bridge Style* в одно из трех значений: *None, Spaces, Arcs*. Отработайте кнопкой мыши на *Ок*. В результате, в зависимости от значения, в местах пересечения линий будут созданы или не созданы мосты.

Дорисуйте свою модель до состояния, представленного на рис. 2.3—2.6.

Прежде чем продолжить дальнейшую разработку модели посмотрим диаграммы для проверки IDEF синтаксиса с помощью Design/IDEF.

Выберите *File/Validate*. В появившемся окне (рис. 4.10) отметьте первые 5 полей, затем *Ок*. На экране появится список объектов, для которых нарушены синтаксические правила IDEF0.

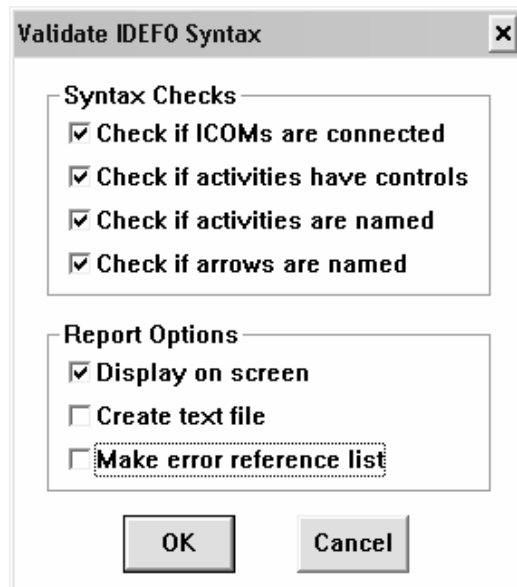


Рис. 4.10 — Окно проверки IDEF-синтаксиса

Если Вы все сделали правильно, то в результате у Вас должна появиться только одна ошибка: «The following boxes have no control arrow: A15 3 Проектирование документации». Исправьте обнаруженные ошибки.

Построение сегментированной дуги. Необходимость самостоятельной трассировки дуги может возникнуть при неудовлетворительном выполнении этой задачи программой. Выберите два несвязных блока для установления связи дугой и наметьте направление трассировки. Выберите *Create/Arrow* и нажмите кнопку мыши на границе исходного блока (можно метки), чтобы начать рисовать дугу. С нажатой кнопкой мыши перемещайте указатель дуги, рисуя ее первый сегмент; нарисовав его, отпустите кнопку мыши. Передвигайте мышь и отработывайте кнопкой для создания каждого дополнительного сегмента, завершив процесс на стороне блока либо метки назначения.

Перемещение дуги. Выделите любую дугу. Подведите указатель к метке у конца дуги и, нажав кнопку мыши, переместите указатель на новое место в стороне блока либо даже другого блока.

Добавление и удаление сегментов дуги.

Для добавления. Выделите дугу и перемещайте с нажатой кнопкой выделенную метку, находящуюся в середине одного из сегментов дуги. Это приведет к разделению сегмента. Передвигайте с нажатой кнопкой выделенную метку в новое место и отпустите кнопку.

Для удаления вершины дуги. Соединяя два сегмента в один, при перемещении выделяющей метки, находящейся на пересечении сегментов, нажмите пробел на клавиатуре.

Текстовые и FEO-страницы.

Чтобы добавить к IDEF-модели текстовую страницу, необходимо выбрать *Create/NEW PAGE*. Выбрать из списка *Methodology* значение *Text* и нажать *Ok* (рис. 4.11).

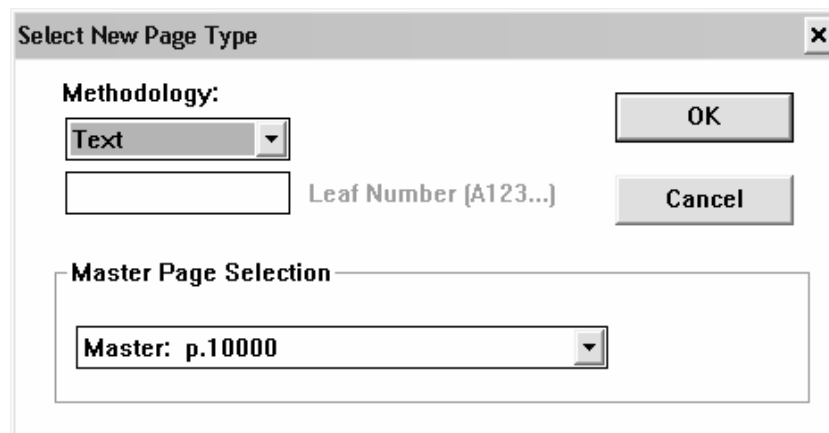


Рис. 4.11 — Окно создания новой текстовой страницы

Текст может создаваться внутри графических объектов. Простейший способ — создать текстовую метку *Create/Label* или блок *Create/Box*.

Если печатаете текст в блоке, то длина строки будет определяться шириной блока. Для того чтобы использовать под текст всю страницу, необходимо увеличить блок до размеров страницы.

Выберите *Create/Box*. Переместите указатель в левый верхний угол страницы, нажмите кнопку мыши, и указатель примет форму руки. Переместите указатель к нижнему правому краю страницы, отпустите кнопку мыши и включите текстовый режим. В верхней части страницы появится точка вставки для ввода текста.

На FEO-страницах (For Exposition Only — для экспозиции только) вы не обязаны придерживаться правил построения IDEF-диаграмм. Вы можете использовать другие объекты Design/IDEF (блоки с закругленными вершинами, эллипсы и многоугольники) и многое другое.

Чтобы создать FEO-страницу, выберите *Create/New Page*. Выберите поле *FEO* (рис. 4.12) и *Ok*.

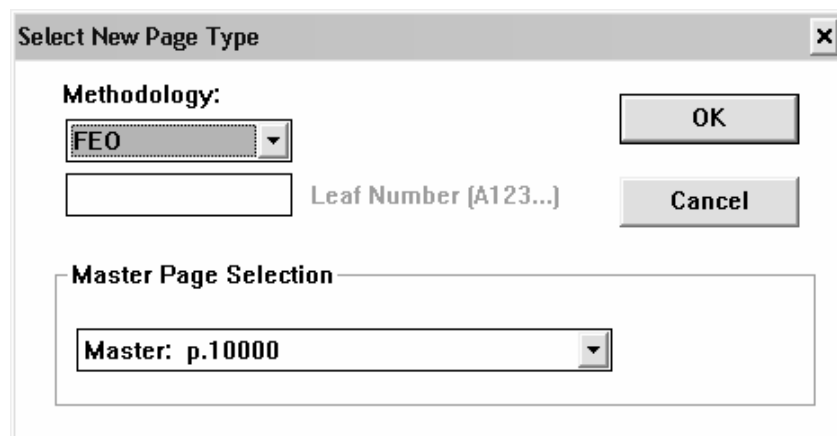


Рис. 4.12 — Окно создания новой FEO-страницы


Соединение FEO- и Text-страниц со страницей диаграммы осуществляется командой *Modify/Attach*. Соединение осуществляется с отдельной диаграммой модели, не создавая иерархической связи. Команда *Attach* связывает выделенный объект на текущей странице с другой страницей модели. Присоединенные объекты, такие как эллипсы и блоки с закругленными вершинами, могут работать как кнопки для перехода с одной страницы на другую.

Для создания кнопки присоединения, если у Вас уже создана FEO-страница, вернитесь к диаграмме, где должно быть сделано присоединение (к любому уровню). Создайте или выделите любой объект, отличный от IDEF-блока. Выберите *Modify/Attach*. Откроется окно *Page Structure*, и в поле состояния появится со-

общение *Clck on Page to be Attached*. Отработайте кнопкой на имени выбранной FEO-страницы, и Вы вернетесь к текущему объекту, который будет описан в поле состояния как *Attachtd Node*. Отработав дважды на присоединенном объекте, Вы переместитесь от исходной страницы к присоединенной.

Отсоединение по команде *Modify/Detach*.

Создание и работа с деревом узлов. Дерево узлов графически представляет иерархию функциональных блоков IDEF-модели. Информация размещается в отдельной странице.

Для создания страницы, содержащей дерево, выберите пункт меню *View/Node Tree*, или нажмите комбинацию клавиш *Shift-F4*, или нажмите на панели инструментов кнопку  (рис. 4.13).

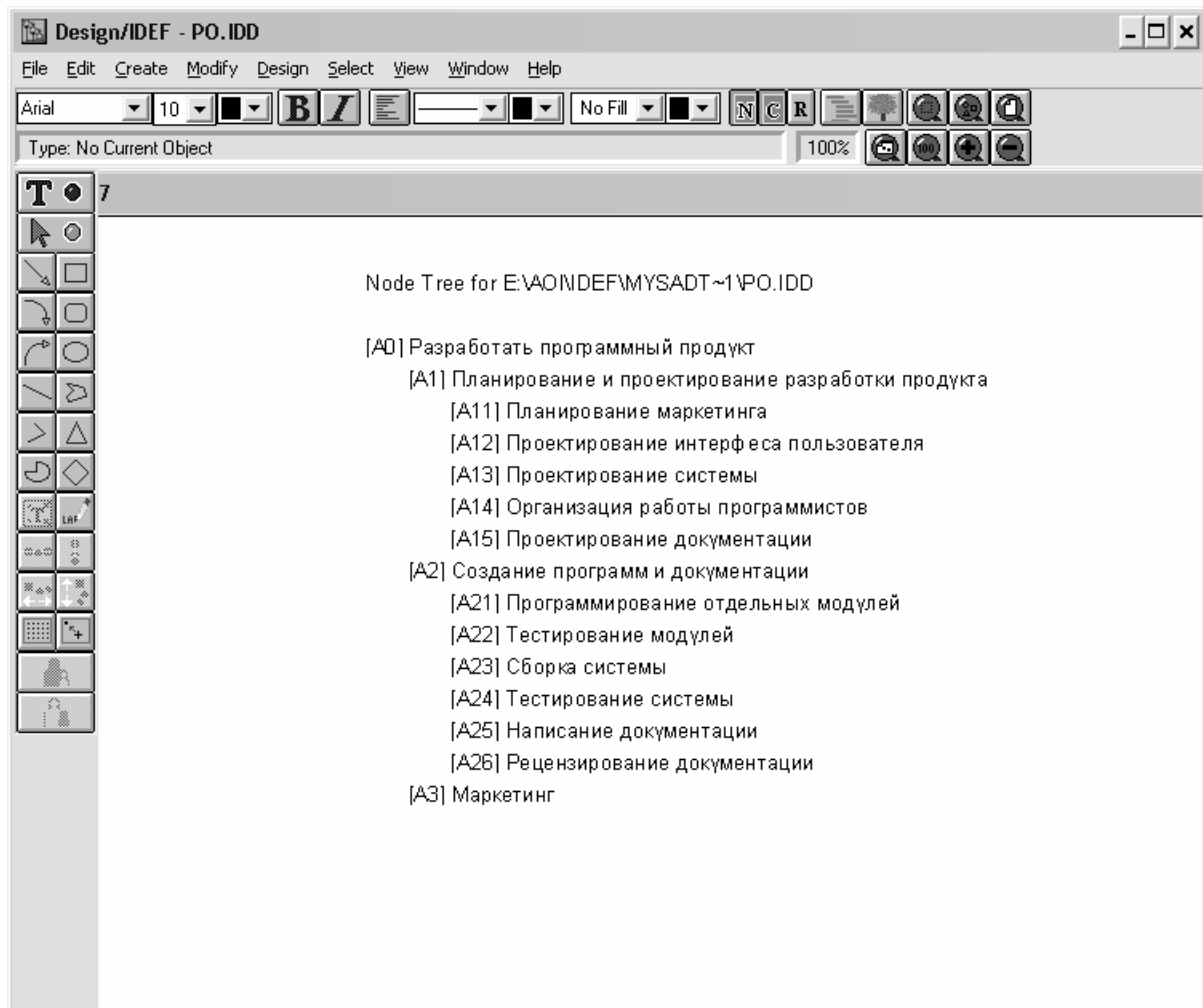


Рис. 4.13 — Пример дерева узлов

4.2 Создание словарей методологии IDEF0

В словаре данных Design/IDEF хранится информация о графических объектах IDEF-моделей. Можно создать единый словарь, в котором будут храниться данные многих моделей, или несколько словарей данных. Одна модель может быть присоединена к одному словарю данных. Имя словаря хранится в модели.

Создание нового словаря. Откройте файл со своей моделью. Отобразите диаграмму A1 модели. Выберите команду *Dict/Create...* Появится окно для наименования словаря.

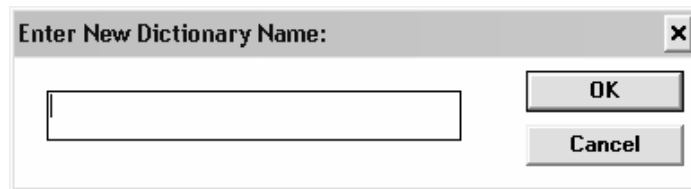


Рис. 4.14 — Окно создания нового словаря

Введите ваши инициалы в качестве имени словаря и щелкните кнопку *OK*. Появится окно ввода имени документа. Имя этого документа связывает IDEF-модель со словарем. В дальнейшем его будет невозможно изменить. Щелкните *OK* для имени, указанного по умолчанию.

Определение типов записей. Прежде чем вводить в словарь конкретную информацию об объектах модели, надо определить структуру типов записей данных, используемых для хранения этой информации. Тип записи выступает в роли шаблона; он устанавливает формат или образец для основной информационной единицы словаря — записи. Записи состоят из отдельных единиц данных, называемых полями. Запись может иметь до семи полей.

В нашем словаре мы создадим три типа записей: «Функция», «Кадры» и «Финансы».

Давайте создадим тип записи «Финансы». Для этого выберите команду *Dict/Define Schema...* Появится окно выбора типа записей — *Select Record Type* (рис. 4.15). Для нового словаря оно

пустое. Щелкните кнопку *New* для определения нового типа записи. Напечатайте имя «Финансы». Для каждой записи о финансах информация будет состоять из пяти статей расходов (полей записи) (рис. 4.16).

Выделите клавишей *Tab* или двойным щелчком мышки значение поля *Number Fields* и напечатайте значение 5.

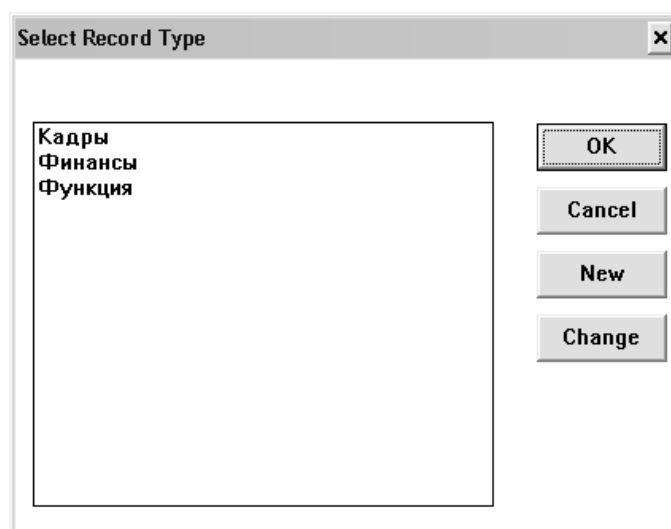


Рис. 4.15 — Окно выбора типа записи

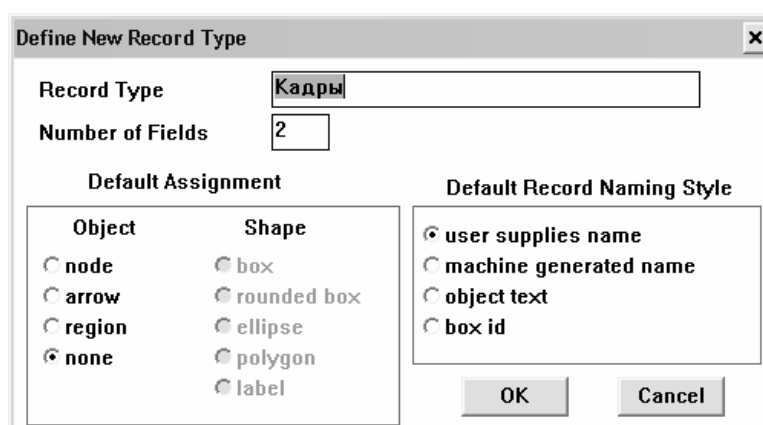


Рис. 4.16 — Окно Определения нового типа записи

Установку ассоциированных по умолчанию объектов и форм (*Default assignment*) менять не будем. Для параметра *object* сохранится значение «*none*» (никакой).

Установка по умолчанию имени записи в окошке *Default Record Naming Style* определяет, каким образом будет названа созданная запись. Если установлен *user suppliers name* (имя пред-

лагается пользователем), то вы сами будете в печатывать имя записи. В других случаях оно будет создаваться автоматически и записываться в словарь. Мы выберем первый режим. Щелкните кнопку *OK*, и появится диалоговое окно определения полей.

Напечатайте для каждого из следующих пяти полей их имена:

- Персонал.
- Ресурсы.
- Накладные расходы.
- Исследования рынка.
- Разное.

Для каждого поля установите тип данных (рис. 4.17):

Щелкните кнопку *undeclared* рядом с именем поля. Появится окно «Тип данных». Установите тип *real* для всех пяти полей. Щелкните *OK*.

Тип записей «Финансы создан».



Рис. 4.17 — Окно определения полей записи

Создайте второй тип записи «Кадры», который состоит из двух полей с именами «Служащий 1» и «Служащий 2» типа *String*.

Третий тип записи «Функция» будет содержать информацию о функциях модели, и поэтому при создании типа записи установите в области Стиля имен записей кнопку *object text* (текст объекта).

Тип записи будет состоять из трех полей: «Сотрудники», «Бюджет», «Обязанности». Из наименования полей мы видим, что поля записи «Сотрудники» и «Бюджет» дублируют информацию записей «Кадры» и «Финансы», привязывая ее к функции (блоку) модели. Вы можете связать эти поля с записью соответствующего типа с помощью структурной ссылки (типа данных *structure*). Установите для поля «Сотрудники» тип данных *structure* с типом записи «Кадры», для поля «Бюджет» тип *structure* с типом записи «Финансы» и для поля «Обязанности» тип *description* (поле типа *string* может иметь максимальный размер только 256 символов).

Итак, словарь создан. Словарь хранится на диске в виде трех файлов с расширениями .IDX, .DAT и .INF. Система снабжает словарные файлы внутренним номером версии для указания времени их создания или использования в модели. Этот номер сверяется и используется для согласованности с внутренним номером версии модели.

Выберите команду *Save* меню *File* для обновления номера версии вашей модели.

Другой способ определения типа записи. Создать новый тип записи можно и при использовании команды *Dict/CreateRecord* (создать запись об объекте), которая приведет к появлению окна Выбор типа записи. В этом окне щелчок на кнопку *New* приведет к появлению окна Определение нового типа записи. В этом режиме тип объекта, ассоциированный с данным типом по умолчанию, будет определен автоматически как тип текущего объекта вашей модели.

Создание записей. После определения типов записей можно создавать записи. Для создания записи типа «Кадры» прокрутите, если нужно, диаграмму вниз и выделите метку «Редактор» на диаграмме «Планирование и проектирование разработки продукта».

Выберите команду *Create Record* меню *Dict*. Появится окно *Select Record Type* (рис. 4.15). Выделите слово «Кадры» в списке типов записей. Щелкните кнопку *Ok*. Появится окно *Change Name*. Напечатайте имя этой записи «Группа взаимодействия» в поле имени записи. Щелкните кнопку *Ok*. Появится окно *Set Field*

Values (рис. 4.18). Введите значения для поля «Служащий 1» — Наталья Воротынцева, для «Служащий 2» — Николай Каменов. Щелкните кнопку *Ок*. Запись «Группа взаимодействия» готова.

Перейдите к диаграмме «Создание программ и документации».

Записи типа «Финансы» содержат информацию о бюджете и расходах. Создайте запись для метки «Расходы на документацию» для этого выделите метку «Расходы на документацию». Выберите команду *Create Record* меню *Dict*. Появится диалоговое окно *Select Record Type*. Выделите тип записи «Финансы». Щелкните *Ок*. Появится окно *Change Name*. Напечатайте Расходы на документацию в поле имени. Щелкните *Ок*. Появится окно Установка значений полей. Напечатайте данные:

- в поле Персонал 80000.00;
- в поле Ресурсы 1500.00;
- в поле Накладные расходы 500.00;
- в поле Исследование рынка 500.00;
- в поле Разное 500.00.

Щелкните кнопку *Ок*. Запись создана. Вы вернулись к диаграмме.

Field Name	Value	Data Type
Служащий 1	Наталья Воротынцева	string
Служащий 2	Николай Каменов	string

Рис. 4.18 — Окно создания записей

Запись типа «Функция» содержит два структурных поля данных. Такие записи создаются посредством связывания значений структурных полей с записями других типов.

Выделите блок A25 «Написание документации». Выберите команду *Create Record* меню *Dict*. Появится диалоговое окно

Select Record Type. Назначьте блок «Разработка документации» существующему типу записи. Выделите тип записи «Функция». Щелкните *Ok*. Появляется окно Установка значений полей.

Замечание. Так как вами был установлен автоматический режим именованной записей (*object text* для всех записей этого типа), имя «Разработка документации» автоматически назначается записи.

Поле «Сотрудники/Кадры» имеет тип данных структура, которую нужно связать с записью типа Кадры. Щелкните на кнопку *undefined* справа от «Сотрудники/Кадры». Появится окно *Change Name*. Вы можете напечатать имя записи или выбрать из существующих имен записей. Напечатайте «Группа интерфейса» заменяя *new name* для новой записи «Кадры». Щелкните *Ok*. Появится окно установки значений полей *Set Field Values. Level 1* показывает что записи имеют уровень вложенности записей. Запись уровня 1 имеет имя «Группа интерфейса» с полями записи типа Кадры. Эта группа включает Сильвию Нейв и Маранду Шекспир. Напечатайте ФИО этих сотрудников в поля значений сотрудников. Вы сейчас определили значения структурного поля. Щелкните *Ok* и вы вернетесь в окно Установка значений полей: Уровень 0.

Заполним поле «Бюджет/Финансы». Щелкните кнопку *undefined* справа от этого поля. Появится окно *Change Name*. Сейчас вы свяжите структурную ссылку с существующей записью. Выделите имя «Бюджет» документирования. Щелкните *Ok*. Появится окно установки значений полей *Set Field Values*. Значения полей в ней уже были определены. Щелкните *Ok*. Мы опять вернемся на уровень 0. Заполнение поля типа *Description*. Щелкните на кнопку *undefined* справа от поля «Обязанности». Появится окно текстового описания. Выделите слово *undefined*, чтобы заменить его. Напечатайте текст описания обязанностей. Детальная спецификация продукта. Компиляция и редактирование детальных спецификаций для функций программы и пользовательского интерфейса. Щелкните *Ok* и вернетесь к окну уровня 0. Обратите внимание, что в этом окне видимы только первые 20 символов текста. Для полного просмотра описания надо щелкнуть на поле значения для открытия окна описания. Запись Разработка документации создана. Щелкните *Ok* и вернетесь к модели.

Создание резервных копий для файлов словаря. Каждый открытый вами словарь автоматически сохраняется в словарных файлах. Информация словаря сохраняется на диске, с которым вы работаете. Удаление этих файлов с диска удаляет словарь, но не корректирует ссылающуюся на него информацию из соответствующей модели. Чтобы файлы были согласованы с моделью, необходимо сохранять свою модель после каждого ее открытия. Рекомендуется создавать резервные версии трех словарных файлов на другом диске. Нельзя переименовывать созданный словарь.

Редактирование существующего словаря. Откройте модель. Выберите пункт меню *Dict/Browse*. В диалоговом окне *Data Dictionary Browse* (рис. 4.19) приведен список всех типов записей и всех записей, определенных в текущем словаре. В этом окне можно узнать размер ссылок на записи словаря или отредактировать выделенную запись. Можно также создать список, определяющий все графические объекты текущей модели, которые ссылаются на конкретную словарную запись или тип записи. Вы можете удалить конкретную запись или тип записи, если словарь не содержит записей данного типа.

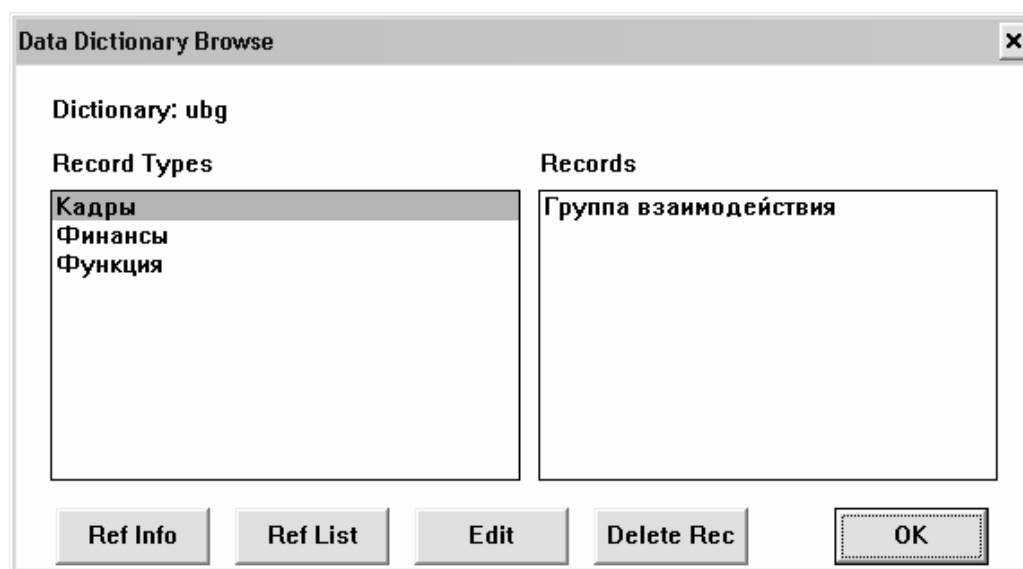


Рис. 4.19 — Окно редактирования словаря

Окно состоит из двух панелей: левое содержит список типов записей, правое — список записей текущего типа. Просмотрите базу данных словаря, выделяя имя каждого типа записей.

Редактирование имени записи. Если какая-либо запись была названа неправильно и должна быть переименована, выделите тип записи и соответствующую запись. Щелкните кнопку *Edit*. Появится окно *Set Field Values*. Щелкните кнопку *Change Name* справа от имени записи. Появится окно *Change Name*. Напечатайте в поле новое имя. Щелкните кнопку *Ок* и вы вернетесь в окно установки значений. Запись изменила название.

Щелкните *Ок*, и название изменится на диске в словарных файлах. Кнопка *Cancel* отменит внесение изменений в запись.

Изменение значений полей. Например, в «Группу взаимодействия» временно принят служащий для редактирования документации программ. Чтобы отразить этот факт, необходимо исправить информацию о бюджете.

Выделите тип записи «Финансы» и имя записи «Расходы на документацию».

Щелкните кнопку *Edit*. Появится окно установки значения поля. Выделите «80000.00» в поле Персонал. Напечатайте 85000.00. Щелкните кнопку *Ок*.

Просмотр информации о записях. Щелкнув мышкой на поле *Ref Info* (Информация о ссылках), Вы получите информацию о выделенной записи или типе записи. В окне о ссылках даются следующие сведения о ссылках:

- количество объектов в текущем документе;
- количество объектов в других документах, ссылающихся на данную запись;
- количество структурных ссылок из других записей словаря на данную ссылку.

Вы можете найти все объекты текущей модели, которые ссылаются на тип записи словаря, создав список ссылок. Для этого в окне просмотра *Data Dictionary Browse* выделите тип записи «Финансы». Щелкните кнопку *Ref List* (Список ссылок).

Удаление записи или типа записи. Если какая-либо запись не нужна, тогда выберите команду *Dict/Browse*, чтобы открыть окно просмотра. Выделите тип записи и запись. Щелкните кнопку *Delete Record*. В диалоговом окне появится информация о

ссылках на удаляемую запись. Щелкните кнопку *Ок*. Связь записи с меткой в модели разрывается, и запись удаляется из словаря.

Когда в словаре нет ни одной записи данного типа, в диалоговом окне *Browse* появляется кнопка *Delete Rec* для удаления типа записи. Тип записи не содержит никаких записей и не нужен в словаре. Выделите данный тип записи. Щелкните кнопку *Delete Rec*. Предупреждающее сообщение просит подтвердить удаление. Щелкните кнопку *Yes*. Данный тип записи удален из словаря

Команда *Dict/Edit record* появляется, когда в модели выделен графический объект и по нему имеется запись в словаре. Если этот объект еще не связан с записью словаря, появляется команда *Dict/Create record*. Этой командой можно изменить имя записи, присвоить ей другой тип записи, изменить значения полей или удалить ссылку между объектом и записью.

Вычисление стоимости на основе функций IDEF-диаграмм. Эта возможность позволяет Вам определить бюджетные категории или категории расходов функции для модели, рассматривать общую стоимость для функции и определять длительность и частоту функции.

Информация стоимости функции может экспортироваться и импортироваться в текстовые и табличные форматы.

Итак, Вы можете определить стоимость каждой функции в IDEF-модели. Общая стоимость функции представляет собой сумму компонент, которые распределены среди категорий функции. Вы определяете одну группу категорий функции для всей модели.

По умолчанию Вы определяете стоимость функции нижнего уровня, и *Design/IDEF* вычисляет стоимость функции родительских уровней.

Можно выполнить установки, позволяющие указать стоимости для любой функции в модели.

Кроме стоимостей, Вы можете ввести длительность и частоту выполнения каждой функции. Значение частоты может быть использовано как множитель для вычисления и просмотра стоимости или продолжительности Вашей модели.

Глоссарий позволяет ввести детальное описание для функции.

Использование элементов глоссария. Команды элементов глоссария показывают диалог, в котором Вы можете ввести следующую информацию о функциональных блоках:

- детальное определение функции;
- длительность выполнения применения функции;
- частоту выполнения функции;
- стоимость, продолжительность и процентное значение функции, которые Вы определили.

Диалог вычислений и просмотра общей стоимости суммирует индивидуальные стоимости функций.

Выберите блок, для которого хотите создать или редактировать запись. Выберите пункт меню *Glossary/Glossary Entry* (рис. 4.20). Для перехода между полями используйте клавишу *Tab* или мышь. Введите значения стоимости для различных категорий расходов функций. Введите значение длительности и частоты с одним знаком после десятичной точки. Введите величины стоимости категорий расходов функций для всех блоков нижнего уровня модели. При этом определяйте соответствующее значение единиц измерения (длительности — часы, стоимости — рубли) из соответствующих диалоговых окон, завершая диалог отработкой на *Ok*.

Рис. 4.20 — Пример описания блока в глоссарии

Частота представляет собой количество повторений выполнения функции в этой модели. Можно установить произведение частоты на стоимость и длительность при просмотре модели и выходной информации. Для этого необходимо установить информацию по вычислениям стоимости, которая будет представляться в модели. Для просмотра и изменения текущих установок просмотра стоимости. Выберите *Glossary/Cost Information*. В диалоговом окне (рис. 4.21) Вы наблюдаете следующие установки:

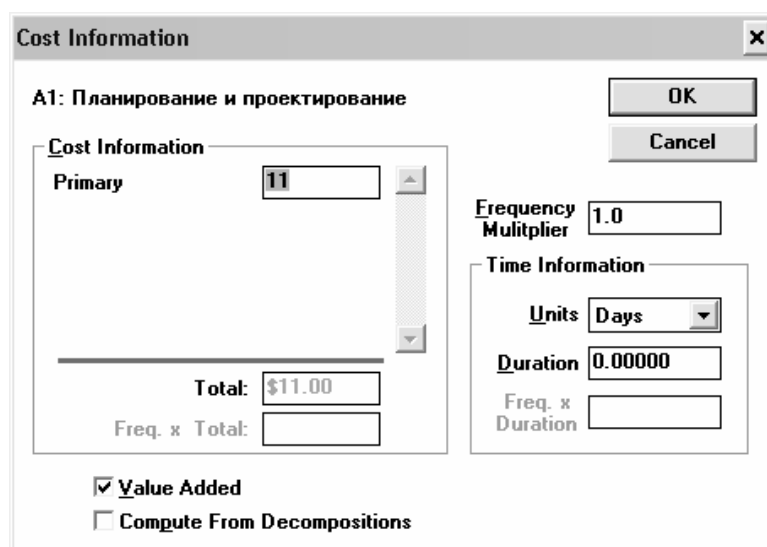


Рис. 4.21 — Окно просмотра и изменения установок стоимости

Value Added — если отмечена установка, стоимости для каждого родительского блока вычисляются как сумма на декомпозированной диаграмме. В этом режиме Вы можете только изменять компоненты стоимости для нижнего уровня модели. Если установка выключена — Вы можете вводить стоимость для любой функции, в том числе и родительской. Автоматических вычислений не производится *Compute From Decompositions* — если отмечена установка, стоимости для всех родительских функций замещаются автоматически, когда Вы вводите или изменяете стоимость для функций нижнего уровня. В случае неустановки данного режима для вычислений следует использовать *Glossary/Compute Now*, что более предпочтительно для больших моделей.

Use Frequency Multiplier — если установлен режим, Design/UDEF перемножает стоимость и продолжительность на частоту для представления в модели или в экспортном формате.

Попытайтесь выполнить различные установки и наблюдайте представление информации.

При помощи пунктов меню Glossary|IDEF 0/1X Integration имеется возможность связывать две модели IDEF 0 и IDEF 1X.

5 ЛАБОРАТОРНАЯ РАБОТА

Задание: Построить функциональную модель одного из приведенных в п. 5.5 процессов.

Выбор варианта контрольной работы осуществляется по общим правилам с использованием следующей формулы:

$$V = (N * K) \text{ div } 100,$$

где V — искомый номер варианта,
 N — общее количество вариантов,
 div — целочисленное деление,
 при $V = 0$ выбирается максимальный вариант,
 K — значение 2-х последних цифр пароля.

Модель может быть построена как с применением программ, автоматизирующих процесс построения моделей IDEF0 (Design/IDEF, BPWin), так и просто программ, позволяющих нарисовать диаграммы по правилам IDEF0, например Microsoft Visio.

После знакомства с методическим руководством в обязательном порядке следует просмотреть ресурс www.vernikov.ru, являющийся самым полным русскоязычным сводом стандартов по семейству IDEF-технологий (в случае отсутствия времени на доскональное изучение данного ресурса в полном объеме можно сразу перейти к разделу <http://vernikov.ru/biznes-modelirovanie/tehnologii-i-standarty.html>).

Выполнение задания проведите согласно следующим этапам:

5.1 Сбор информации

Наиболее подходящая стратегия выполнения данного этапа — самому придумать описание моделируемого процесса (см. раздел 2.1). Экспертом в данном случае будет выступать ваш преподаватель.

Предложите список лиц и документов, которые на ваш взгляд могли бы служить источниками информации при моделировании реального процесса. Обратите внимание на то, что це-

лью рассмотрения процесса в нашем случае является его автоматизация. Это будет в значительной мере определять возможные источники информации.

5.2 Начало моделирования

Создайте диаграммы A0 и A-0. Обратите внимание (см. раздел 2.2), что эти две диаграммы полностью рассказывают все о моделируемой системе с минимальной степенью детализации. Диаграмма A-0, часто называемая контекстной диаграммой, определяет все необходимые связи моделируемого процесса с окружающим миром.

В первую очередь создайте диаграмму A0 и, обобщив ее, создайте диаграмму A-0.

Особое внимание обратите на то, что каждая модель должна иметь определенную цель и создаваться с конкретной точки зрения. Выбор цели осуществляется с учетом вопросов, на которые должна ответить модель, а выбор точки зрения — в соответствии с выбором позиции, с которой описывается система.

Помните, что основа нашей работы это анализ данных о процессе с целью его автоматизации. Обращайте внимание на форму материальных объектов, участвующих в вашей модели. Так, например, в процессе продажи автомобиля нас интересует не «проданный автомобиль», а «акт продажи автомобиля». Форма материальных объектов должна позволять обрабатывать их с помощью компьютера. Механизмы, необходимые для реализации функций, приписанных блокам, покажут нам исполнителей процесса, чей труд необходимо автоматизировать.

Подготовьте список основных типов данных и функций, необходимый для дальнейшей декомпозиции системы. Начните заполнять словарь данных. Помните, что несколько различных типов данных могут использоваться одной функцией.

При создании диаграммы A0 строго следуйте рекомендациям раздела 2.2. Правильное расположение блоков является самым важным этапом построения диаграммы.

Построенные вами на данном этапе диаграммы A-0 и A0 должны представлять законченную картину, поскольку они отражают все основные входы, управления, выходы и функции системы.

5.3 Продолжение моделирования

Продолжение моделирования основывается на тех же методах, что и начальный этап, и выводит модель на следующий уровень детализации. Создайте отдельную диаграмму для, возможно, каждого блока диаграммы верхнего уровня, затем постройте для всех блоков новых диаграмм и так до тех пор, пока модель не будет описывать объект с нужной для достижения вашей цели степенью детализации.

Попытайтесь создать модель не менее чем с четырьмя уровнями детализации.

Не забывайте пополнять и корректировать словарь данных.

5.4 Завершение моделирования

Для определения момента завершения моделирования воспользуйтесь указаниями раздела 2.6.

По завершении моделирования подготовьте отчет, включающий:

- 1) описание моделируемого процесса,
- 2) список лиц и документов, которые на ваш взгляд могли бы служить источниками информации при моделировании реального процесса,
- 3) диаграммы построенной вами функциональной модели.

При использовании программ автоматизированного построения IDEF0 полученные в ходе создания модели файлы НЕ включаются в предоставляемый Вами отчет.

5.5 Список процессов для моделирования

1. Деятельность заправочной станции.
2. Работа фирмы, продающей компьютеры.
3. Ликвидация аварийной ситуации на нефтепроводе.
4. Ликвидация пожаров.
5. Мониторинг пожароопасности в лесном хозяйстве.
6. Процесс подготовки специалистов в учебном заведении.
7. Работа фирмы, продающей автомобили.
8. Работа библиотеки.

9. Управление гостиницей.
10. Управление автобусным парком.
11. Обеспечение питания в столовой.
12. Деятельность фирмы по перевозке грузов.
13. Организация выборной компании.
14. Выращивание сельскохозяйственной продукции.
15. Перевозки грузов на заказ.
16. Перевозка клиентов на заказ.
17. Строительство здания на заказ.
18. Организация туристической поездки клиентов.
19. Проведение выборов в городскую думу.
20. Ремонт бытовой техники на заказ.

СПИСОК ЛИТЕРАТУРЫ

1. Вендров А. М. CASE-технологии. Современные методы и средства проектирования информационных систем / А. М. Вендров. — М. : Финансы и статистика, 1998.
2. Калянов Г. Н. Консалтинг при автоматизации предприятий / Г. Н. Калянов. — М. : СИНТЕГ, 1997.
3. Калянов Г. Н. CASE-технологии. Консалтинг при автоматизации бизнес-процессов. — М. : Горячая линия, 2000.
4. <http://vernikov.ru/biznes-modelirovanie/tehnologii-i-standarty.html>
5. Дубейковский В. И. Эффективное моделирование с AllFusion Process Modeler 4.1.4 и AllFusion PM / В. И. Дубейковский — М. : ДИАЛОГ-МИФИ, 2007 — 384 с.
6. Дубейковский В. И. Практика функционального моделирования с AllFusion Process Modeler 4.1. Где? Зачем? Как? / В. И. Дубейковский. — М. : ДИАЛОГ-МИФИ, 2004 — 464 с.