**Методология функционального моделирования IDEFO**

Методология функционального моделирования IDEFO — это технология описания системы в целом как множества взаимозависимых действий, или функций. Важно отметить функциональную направленность IDEFO — функции системы исследуются независимо от объектов, которые обеспечивают их выполнение. "Функциональная" точка зрения позволяет четко отделить аспекты назначения системы от аспектов ее физической реализации. На рис. 2.1 приведен пример типовой диаграммы IDEFO.

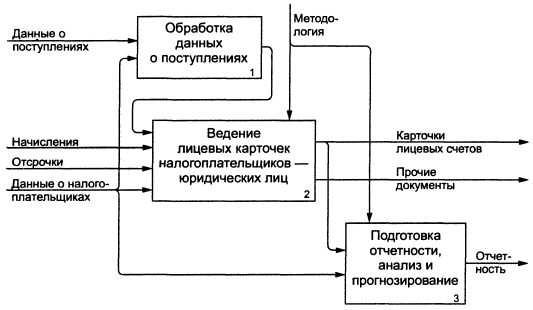


Рис. 2.1. Пример диаграммы IDEFO

Наиболее часто IDEFO применяется как технология исследования и проектирования систем на логическом уровне. По этой причине он, как правило, используется на ранних этапах разработки проекта, до IDEF3 моделирования для сбора данных и моделирования процесса "как есть". Результаты IDEFO анализа могут применяться при проведении проектирования с использованием моделей IDEF3 и диаграмм потоков данных.

**Синтаксис и семантика моделей IDEFO**

**Модели IDEFO**

IDEFO сочетает в себе небольшую по объему графическую нотацию (она содержит только два обозначения: блоки и стрелки) со строгими и четко определенными рекомендациями, в совокупности предназначенными для построения качественной и понятной модели системы.

Методология IDEFO в некоторой степени напоминает рекомендации, существующие в книгоиздательском деле, часто набор напечатанных моделей IDEFO организуется в брошюру (называемую в терминах IDEFO комплект), имеющую содержание, глоссарий и другие элементы, характерные для законченной книги.

Первый шаг при построении модели IDEFO заключается в определении назначения модели — набора вопросов, на которые должна отвечать модель. Набор вопросов можно сравнить с предисловием, в котором раскрывается назначение книги.

Границы моделирования предназначены для обозначения ширины охвата предметной области и глубины детализации и являются логическим продолжением уже определенного назначения модели. Как читающий модель, так и непосредственно ее автор должны понимать степень детальности ответов на поставленные в назначении модели вопросы.

Следующим шагом указывается предполагаемая целевая аудитория, для нужд которой создается модель. Зачастую от выбора целевой аудитории зависит уровень детализации, с которым должна создаваться модель. Перед построением модели необходимо иметь представление о том, какие сведения о предмете моделирования уже известны, какие дополнительные материалы и (или) техническая документация для понимания модели мог)гг быть необходимы целевой аудитории, какие язык и стиль изложения являются наиболее подходящими.

Под точкой зрения понимается перспектива, с которой наблюдалась система при построении модели. Точка зрения выбирается таким образом, чтобы учесть уже обозначенные границы моделирования и назначение модели. Однажды выбранная точка зрения остается неизменной для всех элементов модели. При необходимости могут быть созданы другие модели, отображающие систему с других точек зрения. Вот несколько примеров точек зрения при построении моделей: клиент, поставщик, владелец, редактор.

**Действия**

Действие, обычно в IDEFO называемое функцией, обрабатывает или переводит входные параметры (сырье, информацию и т.п.) в выходные. Поскольку модели IDEFO представляют систему как множество иерархических (вложенных) функций, в первую очередь должна быть определена функция, описывающая систему в целом — контекстная функция. Функции изображаются на диаграммах как поименованные прямоугольники, или функциональные блоки. Имена функций в IDEFO подбираются по сходным правилам с именами действий в IDEF3 — с использованием глаголов или отглагольных существительных. Важно подбирать имена таким образом, чтобы они отражали систему так, как если бы она обозревалась с точки зрения, выбранной для моделирования.

Пример функционального блока приведен на рис. 2.2.

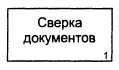


Рис. 2.2. Функциональный блок IDEFO

Выше мы определяли IDEFO модели как иерархическое множество вложенных блоков. Любой блок может быть декомпозирован на составляющие его блоки. Декомпозицию часто ассоциируют с моделированием "сверху вниз", однако это не совсем верно. Функциональную декомпозицию корректнее определять как моделирование "снаружи вовнутрь", в котором мы рассматриваем систему наподобие луковицы, с которой последовательно снимаются слои.

**Границы и связи**

Чтобы быть полезным, описание любого блока должно, как минимум, включать в себя описание объектов, которые блок создает в результате своей работы ("выхода"), и объектов, которые блок потребляет или преобразует ("вход").

В IDEFO также моделируются управление и механизмы исполнения. Под управлением понимаются объекты, воздействующие на способ, которым блок преобразует вход в выход. Механизм исполнения — объекты, которые непосредственно выполняют преобразование входа в выход, но не потребляются при этом сами по себе.

Для отображения категорий информации, присутствующих на диаграммах IDEFO, существует аббревиатура ICOM, отображающая четыре возможных типа стрелок:

I (Input) — вход — нечто, что потребляется в ходе выполнения процесса;

С (Control) — управление — ограничения и инструкции, влияющие на ход выполнения процесса;

О (Output) — выход — нечто, являющееся результатом выполнения процесса;

М (Mechanism) — исполняющий механизм — нечто, что используется для выполнения процесса, но не потребляется само по себе.

Рис. 2.3 показывает 4 возможных типа стрелок в IDEFO, каждый из типов соединяется со своей стороной функционального блока.

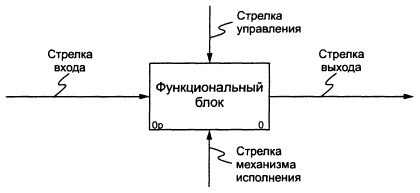


Рис. 2.3. Каждый тип стрелки соединяется со своей стороной функционального блока

Для названия стрелок, как правило, употребляются имена существительные. Стрелки могут представлять собой людей, места, вещи, идеи или собьггия. Как и в случае с функциональными блоками, присвоение имен всем стрелкам на диаграмме является только необходимым условием для понимания читателем сути изображенного. Отдельное описание каждой стрелки в текстовом виде может оказаться критическим фактором для построения точной и полезной модели.

Стрелки входа. Вход представляет собой сырье, или информацию, потребляемую или преобразуемую функциональным блоком для производства выхода. Стрелки входа всегда направлены в левую сторону прямоугольника, обозначающего в IDEFO функциональный блок. Наличие входных стрелок на диаграмме не является обязательным, так как возможно, что некоторые блоки ничего не преобразуют и не изменяют. Примером блока, не имеющего входа, может служить "принятие решения руководством", где для принятия решения анализируется несколько факторов, но ни один из них непосредственно не преобразуется и не потребляется в результате принятия какого-либо решения.

Стрелки управления. Стрелки управления отвечают за регулирование того, как и когда выполняется функциональный блок, и, если он выполняется, какой выход получается в результате его выполнения. Так как управление контролирует поведение функционального блока для обеспечения создания желаемого выхода, каэюдый функциональный блок долэ/сен иметь, как минимум, одну стрелку управления. Стрелки управления всегда входят в функциональный блок сверху.

Управление часто существует в виде правил, инструкций, законов, политики, набора необходимых процедур или стандартов. Влияя на работу блока, оно непосредственно не потребляется и не трансформируется в результате. Может оказаться, что целью функционального блока является как раз изменение того или иного правила, инструкции, стандарта и т.п. В этом случае стрелка, содержащая соответствующую информацию, должна рассматриваться не как управление, а как вход функционального блока.

Управление можно рассматривать как специфический вид входа. В случаях, когда неясно, относить ли стрелку к входу или к управлению, предпочтительно относить ее к управлению до момента, пока неясность не будет разрешена.

Стрелки выхода. Выход — это продукция или информация, получаемая в результате работы функционального блока. Каждый блок долэюен иметь, как минимум, один выход. Действие, которое не производит никакого четко определяемого выхода, не должно моделироваться вообще (по меньшей мере, должно рассматриваться в качестве одного из первых кандидатов на исключение из модели).

При моделировании непроизводственных предметных областей выходами, как правило, являются данные, в каком-либо виде обрабатываемые функциональным блоком. В этом случае важно, чтобы названия стрелок входа и выхода были достаточно различимы по своему смыслу. Например, блок "Прием пациентов" может иметь стрелку "Данные о пациенте" как на входе, так и на выходе. В такой ситуации входящую стрелку можно назвать "Предварительные данные о пациенте", а исходящую — "Подтвержденные данные о пациенте".

Стрелки механизма исполнения. Механизмы являются ресурсом, который непосредственно исполняет моделируемое действие. С помощью механизмов исполнения могут моделироваться: ключевой персонал, техника и (или) оборудование. Стрелки механизма исполнения могут отсутствовать в случае, если оказывается, что они не являются необходимыми для достижения поставленной цели моделирования.

Комбинированные стрелки. В IDEFO существует пять основных видов комбинированных стрелок: выход — вход, выход — управление, выход — механизм исполнения, выход — обратная связь на управление и выход — обратная связь на вход.

Стрелка выход — вход применяется, когда один из блоков должен полностью завершить работу перед началом работы другого блока. Так, на рис. 2.4 формирование счета должно предшествовать приему заказа.

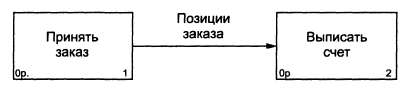


Рис. 2.4. Комбинация стрелок выход — вход

Стрелка выход — управление отражает ситуацию преобладания одного блока над другим, когда один блок управляет работой другого. На рис. 2.5 принципы формирования инвестиционного портфеля управляют поведением брокеров на бирже.

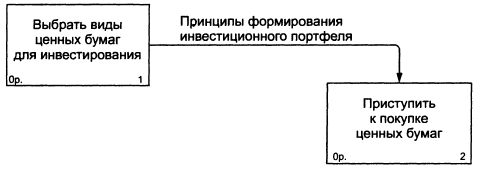


Рис. 2.5. Комбинированная стрелка выход — управление

Стрелки выход — механизм исполнения встречаются реже и отражают ситуацию, когда выход одного функционального блока применяется в качестве оборудования для работы другого блока. На рис. 2.6 зажим, устройство, используемое для закрепления детали во время ее сборки, должно быть собрано для того, чтобы выполнить сборку детали.

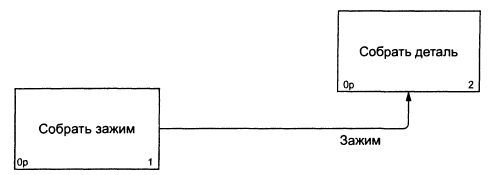


Рис. 2.6. Комбинированная стрелка выход — механизм исполнения

Обратные связи на вход и на управление применяются в случаях, когда зависимые блоки формируют обратные связи для управляющих ими блоков. На рис. 2.7 получаемая от брокеров информация о текущих биржевых курсах применяется для корректировки стратегии игры на бирже.



Рис. 2.7. Комбинированная стрелка выход — обратная связь на управление

Стрелка выход — обратная связь на вход обычно применяется для описания циклов повторной обработки чего-либо. Рис. 2.8 может служить примером применения стрелки такого типа. Кроме того, связи выход — обратная связь на вход могут применяться в случае, если бракованная продукция может заново использоваться в качестве сырья, как это происходит, например, при производстве оконного стекла, когда разбитое в процессе производства стекло перемалывается и переплавляется заново вместе с обыкновенным сырьем.

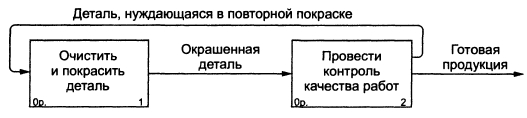


Рис. 2.8. Комбинированная стрелка выход — обратная связь на вход

Разбиение и соединение стрелок. Выход функционального блока может использоваться в нескольких других блоках. Фактически чуть ли не главная ценность IDEFO заключается в том, что эта методология помогает выявить взаимозависимости между блоками системы. Соответственно IDEFO предусматривает как разбиение, так и соединение стрелок на диаграмме. Разбитые на несколько частей стрелки могут иметь наименования, отличающиеся от наименования исходной стрелки. Исходная и разбитые (или объединенные) стрелки в совокупности называются связанными. Такая техника обычно применяется для того, чтобы отразить использование в процессе только части сырья или информации, обозначаемых исходной стрелкой (рис. 2.9). Аналогичный подход применяется и к объединяемым стрелкам.

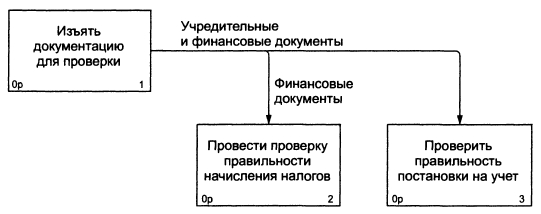


Рис. 2.9. Разбитая на две части и переименованная стрелка

**Туннели**

Понятие связанные стрелки используется для управления уровнем детализации диаграмм. Если одна из стрелок диаграммы отсутствует на родительской диаграмме (например, ввиду своей несущественности для родительского уровня) и не связана с другими стрелками той же диаграммы, точка входа этой стрелки на диаграмму или выхода с нее обозначается туннелем. На рис. 2.10, например, стрелка "корпоративная информационная система" — важный механизм исполнения для данной диаграммы, но, возможно, она более нигде не используется в модели. Туннель в данном случае используется как альтернатива загромождению родительских диаграмм помещением на них несущественных для их уровня стрелок.

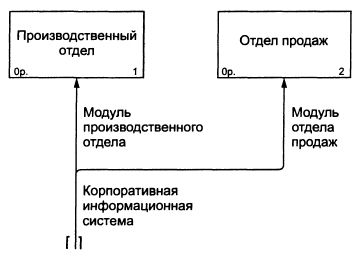


Рис. 2.10. Пример применения туннеля

Кроме того, туннели применяются для отражения ситуации, когда стрелка, присутствующая на родительской диаграмме, отсутствует в диаграмме декомпозиции соответствующего блока. На рис. 2.11 туннель у стрелки "модуль производственного отдела" обозначает, что на диаграмме декомпозиции производственного отдела отсутствует стрелка механизма управления с соответствующим наименованием.



Рис. 2.11. Еще один пример применения туннеля

**Построение моделей IDEFO**

В этом подразделе мы рассмотрим методику построения моделей IDEFO более подробно.

**Диаграммы**

На рис. 2.12 типовая диаграмма IDEFO показана вместе с находящейся на ее полях служебной информацией. Служебная информация состоит из хорошо выделенных верхнего и нижнего колонтитулов (заголовка и "подвала"). Элементы заголовка используются для отслеживания процесса создания модели. Элементы "подвала" отображают наименование модели, к которой относится диаграмма, и показывают ее расположение относительно других диаграмм модели.

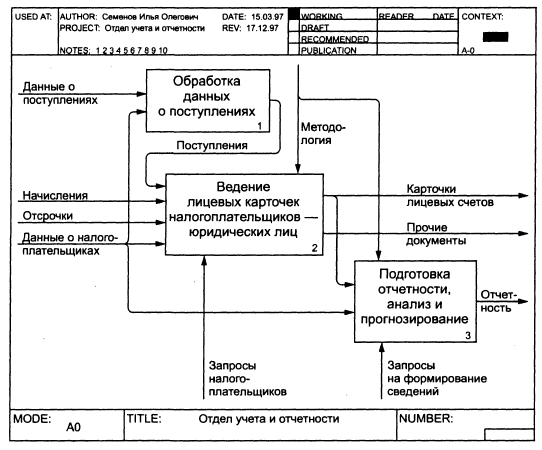


Рис. 2.12. IDEFO-диаграмма со служебной информацией на полях

Все элементы заголовка диаграммы перечислены в табл. 2.1.

Таблица 2.1. Элементы заголовка диаграммы IDEFO

|  |  |
| --- | --- |
| Поле | Назначение |
| USED AT | Используется для отражения внешних ссылок на данную диаграмму (заполняется на печатном документе вручную) |
| Author, date, project | Содержит ФИО автора диаграммы, дату создания, дату последнего внесения изменений, наименование проекта, в рамках которого она создавалась |
| Notes 1 ... 10 | При ручном редактировании диаграмм пользователи могут зачеркивать цифру каждый раз, когда они вносят очередное исправление |
| Status | Статус отражает состояние разработки или утверждения данной диаграммы. Это поле используется для реализации формального процесса публикации с шагами пересмотра и утверждения |
| Working | Новая диаграмма, глобальные изменения или новый автор для существующей диаграммы |
| Draft | Диаграмма достигла некоторого приемлемого для читателей уровня и готова для представления на утверждение |
| Recommended | Диаграмма одобрена и утверждена. Какие-либо изменения не предвидятся |
| Publication | Диаграмма готова для окончательной печати и публикации |
| Reader | ФИО читателя |
| Date | Дата знакомства читателя с диафаммой |
| Context | Набросок расположения функциональных блоков на родительской диаграмме, на котором подсвечен декомпозируемый данной диаграммой блок. Для диаграммы самого верхнего уровня (контекстной диаграммы) в поле помещается контекст ТОР |

Все элементы "подвала" диаграммы перечислены в табл. 2.2.

Таблица 2.2. Элементы "подвала" диаграммы IDEFO

|  |  |
| --- | --- |
| Поле | Назначение |
| Node | Номер диаграммы, совпадающий с номером родительского функционального блока. |
| Title | Имя родительского функционального блока. |
| Number (еще называют С-Number) | Уникальный идентификатор данной версии данной диаграммы. Таким образом, каждая новая версия данной диаграммы будет иметь новое значение в этом поле. Как правило, C-Number состоит из инициалов автора (которые предполагаются уникальными среди всех аналитиков проекта) и последовательного уникального идентификатора, например SDO005. При публикации эти номера могут быть заменены стандартными номерами страниц. Если диаграмма замещает другую диаграмму, номер заменяемой диаграммы может быгь заключен в скобки — SDO005 (SDO004). Это обеспечивает хранение истории изменений всех диаграмм модели. |

**Цикл "эксперт-аналитик"**

Подобно циклу автор-редактор, применяющемуся в книгоиздательском деле, диаграммы IDEFO пересматриваются и изменяются для обеспечения точности отражения предметной области и улучшения своего качества.

Для каждого рецензента автором, как правило, подготавливается свой набор диаграмм. Предложения по изменениям и исправлениям возвращаются рецензентами автору для внесения их в модель. При возникновении разногласий между автором и рецензентом спорная диаграмма обычно рассылается всем рецензентам для достижения группового консенсуса.

Формально механизм рецензирования и модификации диаграмм поддерживается полями Status и нумерацией диаграмм, контроль истории изменений — полем Field (см. табл. 2.1).

**Построение моделей**

Ни одна модель не должна строиться без ясного осознания объекта и целей моделирования. Выбранное определение цели моделирования должно отвечать на следующие вопросы:

* Почему моделируется данный процесс?
* Что выявит данная модель?
* Как ознакомившиеся с этой моделью смогут ее применить?

Следующее предложение может служить примером формулирования цели моделирования. Выявить задачи каждого работника компании и понять в целом взаимосвязь между отдельно взятыми задачами для разработки руководства по обучению новых сотрудников. Модели строятся для того, чтобы ответить на набор поставленных вопросов. Такие вопросы формулируются на ранних стадиях моделирования и впоследствии служат основой для четкого и краткого определения цели моделирования. Примерами таких вопросов могут быть:

* Каковы задачи менеджера?
* Каковы задачи клерка?
* Кто контролирует работу?
* Какая технология нужна для выполнения каждого шага? и т.п.

**Точка зрения**

С методической точки зрения при моделировании полезно использовать мнение экспертов, имеющих разные взгляды на предметную область, однако каждая отдельно взятая модель должна разрабатываться исходя из единственной заранее определенной точки зрения. Часто другие точки зрения вкратце документируются в прикрепленных диаграммах FEO (см. ниже) исключительно для наглядности изложения.

Точку зрения нужно подбирать достаточно аккуратно, основой для выбора должна служить поставленная цель моделирования. Наименованием точки зрения может быть наименование должности, подразделения или роли (например, руководитель отдела или менеджер по продажам). Как и в случае с определением цели моделирования, четкое определение точки зрения необходимо для обеспечения внутренней целостности модели и предотвращения постоянного изменения ее структуры. Может оказаться необходимым построение моделей с разных точек зрения для детального отражения всех особенностей вьщеленных в системе функциональных блоков.

**Границы моделирования**

Одним из положительных результатов построения функциональных моделей оказывается прояснение границ моделирования системы в целом и ее основных компонентов. Хотя и предполагается, что в процессе работы над моделью будет происходить некоторое изменение границ моделирования, их вербальное (словесное) описание должно поддерживаться с самого начала для обеспечения координации работы участвующих в проекте аналитиков. Как и при определении цели моделирования, отсутствие границ затрудняет оценку степени завершенности модели, поскольку границы моделирования имеют тенденцию к расширению с ростом размеров модели.

Границы моделирования имеют два компонента: ширину охвата и глубину детализации. Ширина охвата обозначает внешние границы моделируемой системы. Глубина детализации определяет степень подробности, с которой нужно проводить декомпозицию функциональных блоков.

Чтобы облегчить правильное определение границ моделирования при разработке моделей IDEFO, существенные усилия затрачиваются на разработку и рецензирование контекстной диаграммы IDEFO (диаграммы "самого верхнего" уровня). Иногда даже прибегают к построению дополнительной диаграммы для отображения уровня, более высокого, чем контекстный, для данной модели, что позволяет обозначить систему, внутри которой располагается объект для моделирования.

Существенные затраты на разработку контекстной диаграммы вполне оправданы, поскольку она является своего рода "точкой отсчета" для остальных диаграмм модели и вносимые в нее изменения каскадом отражаются на все лежащие ниже уровни.

Когда границы моделирования понятны, становятся ясными и причины, по которым некоторые объекты системы не вошли в модель.

**Выбор наименования контекстного блока**

Рекомендуемой последовательностью действий при построении модели "с нуля" являются: формулирование цели моделирования, выбор точки зрения, определение границ моделирования. Наименование контекстного блока — функционального блока самого высокого уровня — обобщает определение границ моделирования.

Правила подбора имени для контекстного блока в целом не отличаются от общих правил наименования функциональных блоков, поэтому для них обычно подбирают обобщающие названия, типа "Управление отделом по работе с клиентами", "Обработка заказов" и т.п.

**Определение стрелок на контекстной диаграмме**

Стрелки диаграмм IDEFO обычно проще проектировать в следующем порядке: выход, вход, механизм исполнения, управление. Каждый функциональный блок обозначает отдельную функцию, и эта функция часто имеет ясно и кратко описываемые результаты работы.

Наличие неясностей при анализе выходов того или иного функционального блока — возможный сигнал необходимости проведения реинжиниринга рассматриваемого бизнес-процесса.

Определение выходов. После идентификации возможных выходов полезно провести анализ модели на предмет покрытия ею всех возмоэюных сценариев поведения процесса. Это означает, что если существует вероятность возникновения той или иной ситуации в ходе процесса, модель отражает возможность возникновения такой ситуации.

Многие начинающие аналитики забывают отразить негативные результаты работы функциональных блоков. Например, блок "Провести экзамен по вождению" определенно произведет поток водителей, только что получивших права, но вполне правомерно ожидать и потока лиц, не сдавших экзамен. Негативные результаты часто используются в качестве обратных связей, анализ на их наличие должен проводиться для каждого блока. Важным также является необходимость включения в модель спорных стрелок, принятие решения о наличии которых в модели вполне можно переложить на плечи рецензирующих модель экспертов.

Определение входов. Входы можно рассматривать как особым образом преобразуемые функциональными блоками для производства выхода сырье или информацию. В производственных отраслях определить, как входное сырье преобразуется в готовую продукцию, обычно довольно просто. Однако при моделировании информационных потоков входной поток данных может представляться не потребляемым и не обрабатываемым вообще. Случаи, когда входящие и исходящие стрелки называются в точности одинаково, крайне редки и в основном указывают на бесполезность данного блока для системы в целом или на некорректный выбор имени для исходящей стрелки. Решением может служить применение более подробного описания для входящих и исходящих потоков данных. Например, вход может иметь название "Предварительный диагноз пациента", а выход — "Уточненный диагноз пациента".

Определение механизмов исполнения. После создания входов и выходов можно приступить к рассмотрению механизмов исполнения, или ресурсов, относящихся к функциональному блоку. В понятие механизма исполнения входят персонал, оборудование, информационные системы и т.п. Например, функциональный блок "Собрать деталь" может потребовать использования какого-либо оборудования, например гаечного ключа. При приеме экзаменов на водительские права механизмом исполнения является инспектор ГИБДД. Как правило, определить механизмы исполнения для функциональных блоков довольно просто.

Определение управления. Должно быть определено управление, контролирующее ход работы функционального блока. Все функциональные блоки в IDEFO должны иметь хотя бы одно управление. В случаях, когда не ясно, относить ли стрелку к входу или к управлению, следует ее рисовать как управление. Важно помнить, что управление можно рассматривать как особую форму входа функционального блока.

Когда контекстная диаграмма представляется завершенной, попробуйте задать следующие вопросы:

* Обобщает ли диаграмма моделируемый бизнес-процесс?
* Согласуется ли диаграмма с границами моделирования, точкой зрения и целью моделирования?
* Подходит ли выбранный уровень детализации стрелок для контекстного блока? (Обычно на контекстной диаграмме рекомендуется рисовать не более шести стрелок каждого типа.)

**Нумерация блоков и диаграмм**

Все функциональные блоки IDEFO нумеруются. В номерах допускается использование префиксов произвольной длины, но в подавляющем большинстве моделей используется префикс А. Номер блока проставляется за префиксом. Контекстный блок всегда имеет номер АО.

Префикс повторяется для каждого блока модели. Номера используются для отражения уровня декомпозиции, на котором находится блок. Блок АО декомпозируется в блоки А1, А2, A3 и т.д. А1 декомпозируется в А11, А12, А13 и т.д. А11 декомпозируется в А111, А112, А113 и т.д. Для каждого уровня декомпозиции в конец номера добавляется одна цифра.

**Связь между диаграммой и ее родительским функциональным блоком**

Функциональный блок декомпозируется, если необходимо детально описать его работу. При декомпозиции блока полезно рассмотреть его жизненный цикл, это поможет определить функциональные блоки получающейся "детской" диаграммы. Например, жизненный цикл блока "Поджарить бифштекс" может выглядеть как следующая последовательность: "Подготовить продукты", "Отбить мясо", "Разогреть масло" и т.д.

При моделировании IDEFO важно иметь в виду, что граница детской диаграммы есть граница родительского функционального блока. Это означает, что вся работа выполняется блоками самого нижнего уровня. В отличие от иерархии, применяемой в сфуктурном программировании, блоки верхнего уровня не являются субъектами управления для блоков нижнего уровня. Это означает, что в IDEFO дети — это те же объекты, что и их родители, только показанные с большей детализацией.

Действия генерального директора компании на диаграммах IDEFO могут отражаться рядом с действиями простых рабочих. На концах граничных стрелок (начинающихся или заканчивающихся за пределами диаграммы) детских диаграмм помещаются коды ICOM, чтобы показать, где находится соответствующая стрелка на родительской диаграмме (рис. 2.13). Они нужны для проверки целостности модели и могут быть полезны, когда порядок расположения стрелок на детской диаграмме отличается от порядка их расположения на родительской диаграмме. Код ICOM состоит из латинской буквы I, С, О или М и числа, показывающего расположение стрелки на родительской диаграмме в порядке сверху вниз или слева направо.

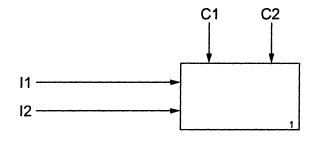


Рис. 2.13. IСОМ-КОДЫ на граничных стрелках

**Два подхода к началу моделирования ("в ширину" и "в глубину")**

Модели могут проектироваться с использованием подхода "в ширину", когда каждая диаграмма максимально детализируется перед своей декомпозицией, и с подходом "в глубину", когда сначала определяется иерархия блоков, а затем создаются соединяющие их стрелки. Естественно, возможно применение комбинации этих подходов, причем иерархия блоков может иногда немного измениться после того, как нарисованы стрелки. Это происходит из-за того, что создание стрелок может изменить понимание внутренней архитектуры моделируемого объекта.

**Когда остановиться?**

Сформулированная цель моделирования содержит вопросы, на которые должна отвечать модель. Когда становится возможным получение ответов на них с помощью модели, модель считается удовлетворяющей поставленным требованиям и рассматривается как завершенная. При построении декомпозиции первого уровня нужно следить за тем, чтобы все блоки на диаграмме лежали внутри определенных ранее границ моделирования. Перед декомпозированием блока нужно удостовериться, не приведет ли это к превышению установленной ранее глубины детализации для данной модели. Еще одно правило состоит в том, что моделирование IDEFO должно продолжаться до тех пор, пока стрелки предшествования (вход и выход) преобладают на диаграммах.

При необходимости дальнейшей детализации отдельных процессов могут быть использованы диаграммы IDEF3.

**Другие диаграммы IDEFO**

В дополнение к контекстным диаграммам и диаграммам декомпозиции при разработке и представлении моделей могут применяться другие виды IDEFO-диаграмм.

Дерево модели. Это обзорная диаграмма, показывающая структуру всей модели. На рис. 2.14 приведен фрагмент такой диаграммы. Обычно вершина дерева соответствует контекстному блоку, под вершиной выстраивается вся иерархия блоков модели. Однако не запрещается назначать вершиной произвольный блок, помещая под ним все его детские блоки. Из-за высокой итеративности функционального моделирования можно ожидать, что дерево модели будет неоднократно изменяться существенным образом до тех пор, пока не будет получена его стабильная версия. Обзор модели с использованием дерева помогает сконцентрироваться на функциональной декомпозиции модели.

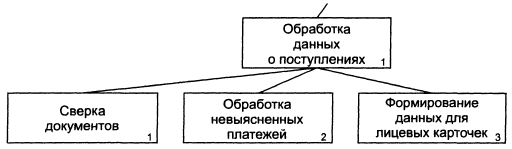


Рис. 2.14. Фрагмент дерева модели

Презентационные диаграммы. Презентационные диаграммы (For Exposition Only diagrams — FEO diagrams) часто включают в модели, чтобы проиллюстрировать другие точки зрения или детали, выходящие за рамки традиционного синтаксиса IDEFO. Диаграммы FEO допускают нарушение любых правил построения диаграмм IDEFO в целях выделения важных с точки зрения аналитика частей модели. Естественно, если диаграмма FEO включена в модель исключительно для отображения другой точки зрения на систему, она скорее всего будет выглядеть как обыкновенная диаграмма IDEFO, удовлетворяя всем ограничениям IDEFO.

Один из способов использования FEO-диаграмм состоит в отделении функционального блока от его окружения посредством создания диаграммы с единственным блоком и всеми относящимися к нему стрелками наподобие контекстной диаграммы (рис. 2.15). Это может оказаться полезным в ситуациях, когда необходимо быстро получить информацию об интерфейсе (стрелках) функционального блока, а соответствующая диаграмма декомпозиции содержит слишком много объектов.

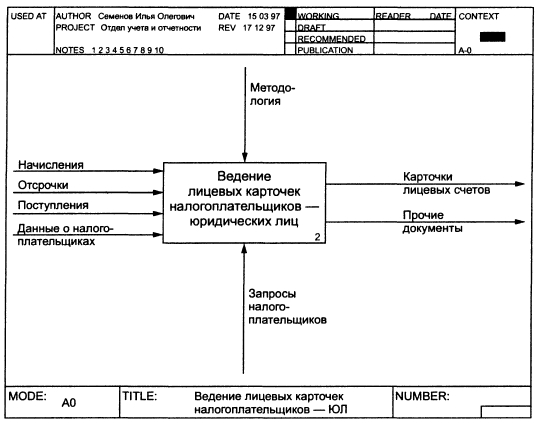


Рис. 2.15. Диаграмма FEO для выделения функционального блока и его стрелок

Кроме того, встречаются следующие виды презентационных диаграмм:

* копия диаграммы IDEFO, которая содержит все функциональные блоки, и стрелки, относящиеся только к одному из функциональных блоков, — это позволяет отразить взаимодействие между этим блоком и другими объектами диаграммы;
* копия диаграммы IDEFO, которая содержит все функциональные блоки, и стрелки, непосредственно относящиеся только к входу и (или) к выходу родительского блока;
* различные точки зрения, как правило, на глубину одного уровня декомпозиции.

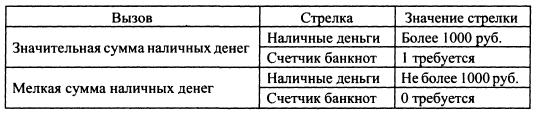
**Взаимосвязь моделей IDEFO и IDEF3**

**Действия, выполняемые в функциональных блоках**

Как правило, при работе с пластиковой картой клиент не производит всех доступных ему при этом действий, выполняя ограниченный набор операций. Например, при оплате покупки не производится снятие наличных, а при проверке баланса состояние счета вообще не изменяется (это верно, конечно, только в случае, если карта обслуживается приличным банком). Мы можем декомпозировать функциональный блок "Обработка операций с пластиковыми картами", создав дополнительные блоки для оплаты покупок, снятия наличных, проверки баланса и т.п. Вместо этого можно создать отдельные модели IDEF3 для каждого из этих действий. Это, в частности, полезно, если в дальнейшем предполагается заняться оцениванием соответствующих операций по тем или иным параметрам.

Более простой альтернативой предложенным выше двум подходам может служить так называемая таблица вызова (activation table), описывающая различные комбинации входов, выходов, управлений и механизмов исполнения для каждого способа вызова функционального блока на исполнение. Вызов — это уникальная конфигурация значений входа, управления и требований к механизмам исполнения (табл. 2.3). Каждому вызову присваивается уникальное имя в пределах блока и перечисляются значения различных стрелок. Комбинация значений стрелок должна быть уникальной для каждого вызова, из чего следует, что для каждого вызова любые две одинаковые стрелки не могут иметь одинаковых значений.

Таблица 2.3. Таблица вызовов для блока "Подсчитать наличные''



Информация о вызовах из табл. 2.3 также дает определенную информацию о стрелках управления данного функционального блока. Например, мы можем предположить, что политика банка при подсчете сумм наличных заключается в использовании счетчиков банкнот для сумм, превышающих 1000 руб.

**Создание моделей IDEF3 для отображения блоков IDEFO**

Для иллюстрирования вызовов листовых функциональных блоков IDEFO (т.е. блоков, не имеющих диаграмм декомпозиции) может быть применено построение моделей IDEF3. Если развитие модели IDEFO предполагается аналитиками именно таким способом, моделями IDEF3 должен быть тщательно документирован каждый возможный вызов функционального блока. Соответствующие таблицы вызовов (наподобие табл. 2.3) можно будет получить впоследствии из соответствующих диаграмм IDEF3.

\* \* \*

Итак, методология функционального моделирования IDEFO — это технология описания системы в целом как множества взаимозависимых действий, или функций. IDEFO имеет функциональную направленность. IDEFO — функции системы исследуются независимо от объектов, которые обеспечивают их выполнение. Одной из основных идей моделей IDEFO является построение двух видов моделей: "как есть" и "как должно быть". Это нужно при проведении реинжиниринга бизнес-процессов организации. Кроме того, IDEFO обеспечивает удобный язык обмена информацией о моделируемой системе.