Р 50.1.028—2001

РЕКОМЕНДАЦИИ ПО СТАНДАРТИЗАЦИИ

Информационные технологии поддержки жизненного цикла продукции

МЕТОДОЛОГИЯ ФУНКЦИОНАЛЬНОГО МОДЕЛИРОВАНИЯ

Издание официальное

ГОССТАНДАРТ РОССИИ
Москва

Предисловие

1 РАЗРАБОТАНЫ Научно-исследовательским Центром CALS-технологий «Прикладная Логистика» при участии Всероссийского научно-исследовательского института стандартизации (ВНИИстандарт)

ВНЕСЕНЫ Техническим комитетом по стандартизации ТК 431 «CALS-технологии»

2 ПРИНЯТЫ И ВВЕДЕНЫ В ДЕЙСТВИЕ Постановлением Госстандарта России от 2 июля 2001 г. № 256-ст

3 ВВЕДЕНЫ ВПЕРВЫЕ

4 ПЕРЕИЗДАНИЕ. Январь 2003 г.

© ИПК Издательство стандартов, 2001
© ИПК Издательство стандартов, 2003

Настоящие рекомендации не могут быть полностью или частично воспроизведены, тиражированы и распространены в качестве официального издания без разрешения Госстандарта России

Содержание

1 Область применения 1
2 Определения 1
3 Сокращения 3
4 Концепция IDEF0 3
5 Синтаксис графического языка IDEF0 4
5.1 Блок 4
5.2 Стрелка 5
5.3 Синтаксические правила 5
6 Семантика языка IDEF0 6
6.1 Семантика блоков и стрелок 6
6.2 Имена и метки 7
6.3 Сводка семантических правил для блоков и стрелок 7
6.4 Диаграммы IDEF0 7
6.5 Контекстная диаграмма верхнего уровня 7
6.6 Дочерняя диаграмма 8
6.7 Родительская диаграмма 9
6.8 Текст и глоссарий 10
6.9 Диаграммы-иллюстрации (FEO) 11
7 Свойства диаграмм 11
7.1 Стрелки как ограничения 11
7.2 Параллельное функционирование 11
7.3 Ветвление и слияние сегментов стрелок 11
7.4 Отношения блоков на диаграммах 12
8 Отношения между блоками диаграммы и другими диаграммами (окружающей средой) . . . 14
8.1 Граничные стрелки 14
8.2 ICOM-кодирование граничных стрелок 15
8.3 Стрелки, помещенные в «туннель» 16
9 Правила построения диаграмм 17
10 Ссылочные выражения (коды) 21
10.1 Номера блоков 22
10.2 Узловые номера 22
10.3 Перечень узлов 22
10.4 Дерево узлов 23
11 Методика разработки функциональных моделей в среде IDEF0 23
11.1 Общие положения 23
11.2 Классификация функций, моделируемых блоками IDEF0 24
11.3 Организационно-технические структуры и механизмы IDEF0-моделей 25
11.4 Управление — особый вид процесса, операции, действия 27
11.5 Типизация функциональных моделей и IDEF0-диаграмм 27
12 Организация процесса функционального моделирования и управление проектом 28
12.1 Общие положения 28
12.2 Состав участников проекта и структура их взаимодействия 30
12.3 Заключительные замечания 33
Приложение А Стандартный бланк методологии IDEF0 и правила его заполнения 34
Приложение Б Метамодель 37
Приложение В Функциональная модель предприятия 42

Введение

Постоянное усложнение производственно-технических и организационно-экономических систем — фирм, предприятий, производств и других субъектов производственно-хозяйственной деятельности — и необходимость их анализа с целью совершенствования функционирования и повышения эффективности обусловливают необходимость применения специальных средств описания и анализа таких систем. Эта проблема приобретает особую актуальность в связи с появлением интегрированных компьютеризированных производств и автоматизированных предприятий.

В США в конце 70-х годов была предложена и реализована Программа интегрированной компьютеризации производства ICAM (Integrated Computer Aided Manufacturing), направленная на увеличение эффективности промышленных предприятий посредством широкого внедрения компьютерных (информационных) технологий.

Реализация программы ICAM потребовала создания адекватных методов анализа и проектирования производственных систем и способов обмена информацией между специалистами, занимающимися такими проблемами. Для удовлетворения этой потребности в рамках программы ICAM была разработана методология моделирования IDEF (ICAM Definition), позволяющая исследовать структуру, параметры и характеристики производственно-технических и организационно-экономических систем. Общая методология IDEF состоит из трех частных методологий моделирования, основанных на графическом представлении систем:

IDEF0 используется для создания **функциональной модели,** отображающей структуру и функции системы, а также потоки информации и материальных объектов, преобразуемые этими функциями;

IDEF1 применяется для построения **информационной модели,** отображающей структуру и содержание информационных потоков, необходимых для поддержки функций системы;

IDEF2 позволяет построить **динамическую модель** меняющихся во времени поведения функций, информации и ресурсов системы.

К настоящему времени наибольшее распространение и применение имеют методологии IDEF0 и IDEF1 (IDEF1X).

Методология IDEF0, особенности и приемы применения которой описываются в настоящих рекомендациях, основана на подходе, получившем название SADT (Structured Analysis & Design Technique — метод структурного анализа и проектирования). Основу этого подхода и методологии IDEF0 составляет графический язык описания (моделирования) систем.

В связи с расширяющимся применением информационных технологий и, в частности, CALS-технологий в народном хозяйстве Российской Федерации в настоящих рекомендациях приводятся основные сведения о методологии IDEF0 и графическом языке описания моделей, а также практические указания по методике разработки таких моделей.

РЕКОМЕНДАЦИИ ПО СТАНДАРТИЗАЦИИ

Информационные технологии поддержки жизненного цикла продукции

МЕТОДОЛОГИЯ ФУНКЦИОНАЛЬНОГО МОДЕЛИРОВАНИЯ

Continuous acquisition and life-cycle support.
Methodology of functional modelling

Дата введения 2002—07—01

1 Область применения

Настоящие рекомендации по стандартизации предназначены для использования при анализе и синтезе производственно-технических и организационно-экономических систем методами функционального моделирования в различных отраслях экономики. Рекомендации содержат описание комплекса средств для наглядного представления широкого спектра деловых, производственных и других процессов и операций предприятия на любом уровне детализации, а также организационные и методические приемы применения этих средств.

2 Определения

В настоящих рекомендациях применяют следующие термины:

2.1 **блок:** Прямоугольник, содержащий имя и номер и используемый для описания функции.

2.2 **ветвление:** Разделение стрелки на два или большее число сегментов. Может означать «развязывание пучка» (см. 2.27).

2.3 **внутренняя стрелка:** Входная, управляющая или выходная стрелка, концы которой связывают источник и потребителя, являющиеся блоками одной диаграммы. Отличается от граничной стрелки.

2.4 **входная стрелка:** Класс стрелок, отображающих вход IDEF0-блока, то есть данные или материальные объекты, которые преобразуются функцией в выход. Входные стрелки связываются с левой стороной блока IDEF0.

2.5 **выходная стрелка:** Класс стрелок, отображающих выход IDEF0-блока, то есть данные или материальные объекты, произведенные функцией. Выходные стрелки связываются с правой стороной блока IDEF0.

2.6 **глоссарий:** Список определений для ключевых слов, фраз и аббревиатур, связанных с узлами, блоками, стрелками или с моделью IDEF0 в целом.

2.7 **граничная стрелка:** Стрелка, один из концов которой связан с источником или потребителем, а другой не присоединен ни к какому блоку на диаграмме. Отображает связь диаграммы с другими блоками системы и отличается от внутренней стрелки.

2.8 **декомпозиция:** Разделение моделируемой функции на функции-компоненты.

2.9 **дерево узлов:** Представление отношений между родительскими и дочерними узлами модели IDEF0 в форме древовидного графа. Имеет то же значение и содержание, что и перечень узлов (см. 2.23).

2.10 **диаграмма А—0 (А минус ноль):** Специальный вид (контекстной) диаграммы IDEF0, состоящей из одного блока, описывающего функцию верхнего уровня, ее входы, выходы, управление, и механизмы, вместе с формулировками цели модели и точки зрения, с которой строится модель.

2.11 **диаграмма:** Часть модели, описывающая декомпозицию блока.

Издание официальное

2.12 **диаграмма-иллюстрация** (FEO): Графическое описание, используемое для сообщения специфических фактов о диаграмме IDEF0. При построении диаграмм FEO можно не придерживаться правил IDEF0.

2.13 **дочерний блок:** Блок на дочерней (порожденной) диаграмме.

2.14 **дочерняя диаграмма:** Диаграмма, детализирующая родительский (порождающий) блок.

2.15 **имя блока:** Глагол или глагольный оборот, помещенный внутри блока и описывающий моделируемую функцию.

2.16 **интерфейс:** Разделяющая граница, через которую проходят данные или материальные объекты; соединение между двумя или большим числом компонентов модели, передающее данные или материальные объекты от одного компонента к другому.

2.17 **код ICOM** (аббревиатура Input — вход, Control — управление, Output — выход, Mechanism — механизм): Код, обеспечивающий соответствие граничных стрелок дочерней диаграммы со стрелками родительского блока; используется для ссылок.

2.18 **контекст:** Окружающая среда, в которой действует функция (или комплект функций на диаграмме).

2.19 **контекстная диаграмма:** Диаграмма, имеющая узловой номер A—n (A минус n) ($n \geq 0$), которая представляет контекст модели. Диаграмма A—0, состоящая из одного блока, является необходимой (обязательной) контекстной диаграммой; диаграммы с узловыми номерами A—1, A—2, (A минус 1, A минус 2) . . . , — дополнительные контекстные диаграммы.

2.20 **метка стрелки:** Существительное или оборот существительного, связанные со стрелкой или сегментом стрелки и определяющие их значение.

2.21 **модель IDEF0:** Графическое описание системы, разработанное с определенной целью (см 2.46) и с выбранной точки зрения (см 2.39). Комплект документов IDEF0, которые изображают функции системы с помощью графики (диаграмм), текста и глоссария.

2.22 **номер блока:** Число (0 — 6), помещаемое в правом нижнем углу блока и однозначно идентифицирующее блок на диаграмме.

2.23 **перечень узлов:** Список, часто ступенчатый, показывающий узлы модели IDEF0 в упорядоченном виде. Имеет то же значение и содержание, что и дерево узлов (см 2.9).

2.24 **примечание к модели:** Текстовый комментарий, являющийся частью диаграммы IDEF0 и используемый для записи факта, не нашедшего графического изображения.

2.25 **родительская диаграмма:** Диаграмма, которая содержит родительский блок.

2.26 **родительский блок:** Блок, который подробно описывается дочерней диаграммой.

2.27 **связывание/развязывание:** Объединение значений стрелок в составное значение (связывание в «пучок»), или разделение значений стрелок (развязывание «пучка»), выраженные синтаксисом слияния или ветвления стрелок.

2.28 **сегмент стрелки:** Сегмент линии, который начинается или заканчивается на стороне блока, в точке ветвления или слияния, или на границе (несвязанный конец стрелки).

2.29 **семантика:** Значение синтаксических компонентов языка.

2.30 **синтаксис:** Структурные компоненты или характеристики языка и правила, которые определяют отношения между ними.

2.31 **слияние:** Объединение двух или большего числа сегментов стрелок в один сегмент. Может означать «связывание пучка» (см 2.27).

2.32 **С-номер:** Номер, создаваемый в хронологическом порядке и используемый для идентификации диаграммы и прослеживания ее истории; может быть использован в качестве ссылочного выражения при определении конкретной версии диаграммы. Обычно С-номер состоит из инициалов автора модели и хронологических данных (даты создания очередной версии диаграммы).

2.33 **стрелка:** Направленная линия, состоящая из одного или нескольких сегментов, которая моделирует открытый канал или канал, передающий данные или материальные объекты от источника (начальная точка стрелки), к потребителю (конечная точка с «наконечником»). Имеется четыре класса стрелок: входная, выходная, управляющая стрелка механизма (включает стрелку вызова). (См. сегмент стрелки, граничная стрелка, внутренняя стрелка).

2.34 **стрелка вызова:** Вид стрелки механизма, который обозначает обращение из блока данной модели (или части модели) к блоку другой модели (или другой части той же модели) и обеспечивает связь между моделями или между разными частями одной модели.

2.35 **стрелка механизма:** Класс стрелок, которые отображают механизмы IDEF0, то есть средства, используемые для выполнения функции; включает специальный случай стрелки вызова. Стрелки механизмов связываются с нижней стороной блока IDEF0.

2.36 **стрелка, помещенная в туннель (туннельная стрелка):** Стрелка (со специальной нотацией), не удовлетворяющая обычному требованию, согласно которому каждая стрелка на дочерней диаграмме должна соответствовать стрелкам на родительской диаграмме. Туннельные стрелки — дополнительное средство графического языка, облегчающее чтение и понимание диаграмм.

2.37 **текст:** Любой текстовый (не графический) комментарий к графической диаграмме IDEF0.

2.38 **тильда:** Небольшая ломаная (волнистая) линия, используемая для соединения метки с конкретным сегментом стрелки или примечания модели с компонентом диаграммы.

2.39 **точка зрения:** Указание на должностное лицо или подразделение организации, с позиции которого разрабатывается модель. Для каждой модели точка зрения единственная.

2.40 **узел:** Блок, порождающий дочерние блоки; родительский блок. (См. перечень узлов, дерево узлов, узловой номер, узловая ссылка, номер узла диаграммы).

2.41 **узловая ссылка:** Код, присвоенный диаграмме для ее идентификации и определения положения в иерархии модели; формируется из сокращенного имени модели и узлового номера диаграммы с дополнительными расширениями.

2.42 **узловой номер диаграммы:** Часть узловой ссылки диаграммы, которая соответствует номеру родительского блока.

2.43 **узловой номер:** Код, присвоенный блоку и определяющий его положение в иерархии модели; может быть использован в качестве подробного ссылочного выражения.

2.44 **управляющая стрелка:** Класс стрелок, которые в IDEF0 отображают управления, то есть условия, при выполнении которых выход блока будет правильным. Данные или объекты, моделируемые как управления, могут преобразовываться функцией, создающей соответствующий выход. Управляющие стрелки связываются с верхней стороной блока IDEF0.

2.45 **функция:** Деятельность, процесс или преобразование (моделируемые блоком IDEF0), идентифицируемое глаголом или глагольной формой, которая описывает, что должно быть выполнено.

2.46 **цель:** Краткая формулировка причины создания модели.

3 Сокращения

Сокращения, принятые в настоящих рекомендациях:

ICAM — интегрированная компьютеризация производства.

ICOM — вход (Input), управление (Control), выход (Output), механизм (Mechanism).

IDEF0 — методология, используемая для создания функциональной модели.

IDEF1 — методология, используемая для создания информационной модели.

IDEF2 — методология, используемая для создания динамической модели.

FEO — диаграмма-иллюстрация.

4 Концепция IDEF0

Методология IDEF0 основана на следующих концептуальных положениях.

4.1 Модель — искусственный объект, представляющий собой отображение (образ) системы и ее компонентов. Считается, что

М *моделирует* **А,** *если* **М** *отвечает на вопросы относительно* **А**.

Здесь **М** — модель, **А** — моделируемый объект (оригинал). Модель разрабатывают для понимания, анализа и принятия решений о реконструкции (реинжиниринге) или замене существующей, либо проектировании новой системы. **Система** представляет собой совокупность взаимосвязанных и взаимодействующих частей, выполняющих некоторую полезную работу. Частями (элементами) системы могут быть любые комбинации разнообразных сущностей, включающие людей, информацию, программное обеспечение, оборудование, изделия, сырье или энергию (энергоносители). Модель описывает, что происходит в системе, как ею управляют, что она преобразует, какие средства использует для выполнения своих функций и что производит.

4.2 Блочное моделирование и его графическое представление. Основной концептуальный принцип методологии IDEF — представление любой изучаемой системы в виде набора взаимодействующих и взаимосвязанных блоков, отображающих процессы, операции, действия (определения — см. ниже), происходящие в изучаемой системе. В IDEF0 все, что происходит в системе и ее элементах, принято называть **функциями.** Каждой функции ставится в соответствие **блок**. На **IDEF0-диаграмме,** основном документе при анализе и проектировании систем, блок представляет собой прямоугольник. Интерфейсы, посредством которых блок взаимодействует с другими блоками или с внешней

по отношению к моделируемой системе средой, представляются **стрелками,** входящими в блок или выходящими из него. Входящие стрелки показывают, какие условия должны быть одновременно выполнены, чтобы функция, описываемая блоком, осуществилась.

4.3 Лаконичность и точность. Документация, описывающая систему, должна быть точной и лаконичной. Сведения о свойствах и характеристиках системы в форме традиционных текстов в этом смысле неудовлетворительны, поскольку зачастую содержат избыточную информацию, допускают неоднозначное толкование и т.д. Графический язык позволяет лаконично, однозначно и точно показать все элементы (блоки) системы и все отношения и связи между ними, выявить ошибочные, лишние или дублирующие связи и т.д.

4.4 Передача информации. Средства IDEF0 облегчают передачу информации от одного участника разработки модели (отдельного разработчика или рабочей группы) к другому. К числу таких средств относятся:

- диаграммы, основанные на простой графике блоков и стрелок, легко читаемые и понимаемые;
- метки на естественном языке для описания блоков и стрелок, а также глоссарий и сопроводительный текст, уточняющие смысл элементов диаграммы;
- последовательная декомпозиция диаграмм, строящаяся по иерархическому принципу, при котором на верхнем уровне отображаются основные функции, а затем происходит их детализация и уточнение;
- древовидные схемы иерархии диаграмм и блоков, обеспечивающие обозримость модели в целом и входящих в нее деталей, что особенно важно при моделировании больших систем.

4.5 Строгость и формализм. Разработка моделей IDEF0 требует соблюдения ряда строгих формальных правил, обеспечивающих преимущества методологии в отношении однозначности, точности и целостности сложных многоуровневых моделей. Эти правила описываются ниже. Здесь отмечается только основное из них: на всех стадиях и этапах разработки и корректировки модели должны строго, формально соблюдаться синтаксические и семантические правила графического языка, а результаты — тщательно документироваться с тем, чтобы при ее эксплуатации не возникало вопросов, связанных с неполнотой или некорректностью документации. Программный продукт Design/IDEF 3.7 (и более поздние версии) фирмы Meta Software Corporation поддерживает автоматическое соблюдение большинства из перечисленных правил.

4.6 Итеративное моделирование. Разработка модели в IDEF0 представляет собой пошаговую, итеративную процедуру. На каждом шаге итерации разработчик предлагает вариант модели, который подвергают обсуждению, рецензированию и последующему редактированию, после чего цикл повторяется. Такая организация работы способствует оптимальному использованию знаний системного аналитика, владеющего методологией и техникой IDEF0, и знаний специалистов — экспертов в предметной области, к которой относится объект моделирования.

4.7 Отделение «организации» от «функций». При разработке моделей следует избегать изначальной «привязки» функций исследуемой системы к существующей организационной структуре моделируемого объекта (предприятия, фирмы). Это помогает избежать субъективной точки зрения, навязанной организацией и ее руководством. Организационная структура должна явиться результатом использования (применения) модели. Сравнение результата с существующей структурой позволяет, во-первых, оценить адекватность модели, а во-вторых — предложить решения, направленные на совершенствование этой структуры.

5 Синтаксис графического языка IDEF0

Набор структурных компонентов языка, их характеристики и правила, определяющие связи между компонентами, представляют собой синтаксис языка. Компоненты синтаксиса IDEF0 — блоки, стрелки, диаграммы и правила. Блоки представляют функции, определяемые как деятельность, процесс, операция, действие или преобразование (см. ниже). Стрелки представляют данные или материальные объекты, связанные с функциями. Правила определяют, как следует применять компоненты; диаграммы обеспечивают формат графического и словесного описания моделей. Формат образует основу для управления конфигурацией модели.

5.1 Блок

Блок описывает функцию. Типичный блок показан на рисунке 1. Внутри каждого блока помещаются его имя и номер. Имя должно быть активным глаголом или глагольным оборотом, описывающим функцию. Номер блока размещается в правом нижнем углу. Номера блоков используются для их идентификации на диаграмме и в соответствующем тексте.

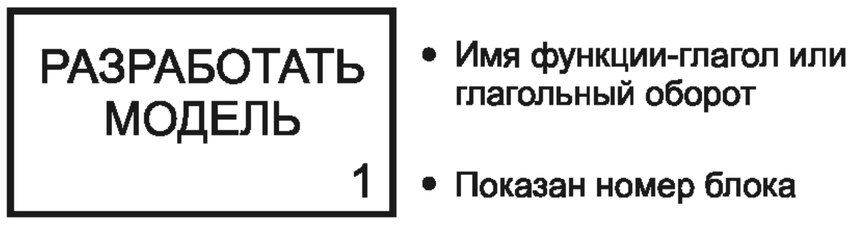


Рисунок 1

5.2 Стрелка

Стрелка формируется из одного или нескольких отрезков прямых и наконечника на одном конце. Как показано на рисунке 2, сегменты стрелок могут быть прямыми или ломаными; в последнем случае горизонтальные и вертикальные отрезки стрелки сопрягаются дугами, имеющими угол 90°. Стрелки не представляют поток или последовательность событий, как в традиционных блок-схемах потоков или процессов (потоковых диаграммах). Они лишь показывают, какие данные или материальные объекты должны поступить на вход функции для того, чтобы эта функция могла выполняться.

5.3 Синтаксические правила

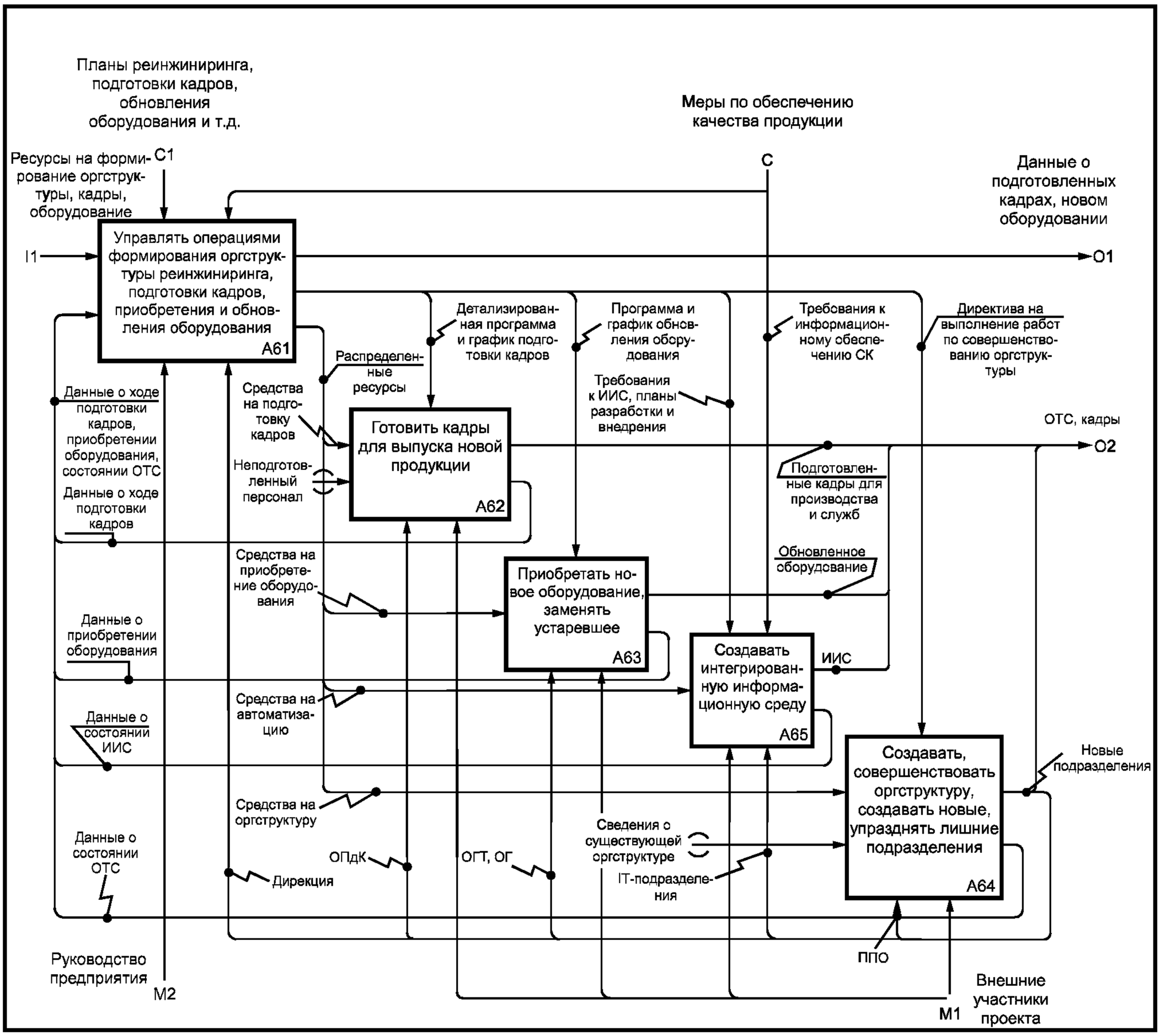


Рисунок 2

5.3.1 Б л о к и

Для блоков установлены следующие синтаксические правила:

- размеры блоков должны быть достаточными для того, чтобы включить имя и номер блока.
- блоки должны быть прямоугольными, с прямыми углами;
- блоки должны быть нарисованы сплошными линиями.

5.3.2 С т р е л к и

Для стрелок установлены следующие синтаксические правила:

- ломаные стрелки изменяют направление только под углом 90°;
- стрелки должны быть нарисованы сплошными линиями. Можно использовать линии различной толщины;
- стрелки могут состоять только из вертикальных или горизонтальных отрезков; отрезки, направленные по диагонали, не допускаются;
- концы стрелок должны касаться внешней границы функционального блока, но не должны пересекать ее;
- стрелки должны присоединяться к блоку на его сторонах. Присоединение в углах не допускается.

6 Семантика языка IDEF0

Семантика определяет содержание (значение) синтаксических компонентов языка и способствует правильности их интерпретации. Интерпретация устанавливает соответствие между блоками и стрелками с одной стороны и функциями и их интерфейсами — с другой.

6.1 Семантика блоков и стрелок

Поскольку IDEF0 есть методология функционального моделирования, имя блока, описывающее функцию, должно быть глаголом или глагольным оборотом. Например имя блока «Выполнить проверку» означает, что блок с таким именем превращает непроверенные детали в проверенные. После присваивания блоку имени, к соответствующим его сторонам присоединяются входные, выходные и управляющие стрелки, а также стрелки механизма, что и определяет наглядность и выразительность изображения блока IDEF0 (см. рисунок 3).

Чтобы гарантировать точность модели, следует использовать стандартную терминологию. Блоки именуются глаголами или глагольными оборотами, и эти имена сохраняются при декомпозиции. Стрелки и их сегменты, как отдельные, так и связанные в «пучок», помечаются существительными или оборотами существительного. Метки сегментов позволяют конкретизировать данные или материальные объекты, передаваемые этими сегментами, с соблюдением синтаксиса ветвлений и слияний.

Каждая сторона функционального блока имеет стандартное назначение с точки зрения связи блок/стрелки. В свою очередь, сторона блока, к которой присоединена стрелка, однозначно определяет ее роль. Стрелка (и), входящая (ие) в левую сторону блока, — **вход(ы).** Входы преобразуются или расходуются функцией, чтобы создать то, что появится на ее выходе. Стрелка(и), входящая (ие) в блок сверху, — **управление(я).** Управление (я) определяет (ют) условия, необходимые функции, чтобы произвести правильный выход. Стрелка (и), покидающая (ие) блок справа, — **выход(ы),** то есть данные или материальные объекты, произведенные функцией.

Стрелки, подключенные к нижней стороне блока, представляют **механизмы,** то есть все то, с помощью чего осуществляется преобразование входов в выходы. Стрелки, направленные вверх, идентифицируют средства, поддерживающие выполнение функции. Другие средства могут наследоваться из родительского блока. Стрелки механизма, направленные вниз, являются стрелками вызова. Стрелки вызова обозначают обращение из данной модели или из данной части модели к блоку, входящему в состав другой модели или другой части модели, обеспечивая их связь, то есть разные модели или разные части одной и той же модели могут совместно использовать один и тот же элемент (блок).

Стандартное расположение стрелок показано на рисунке 3.

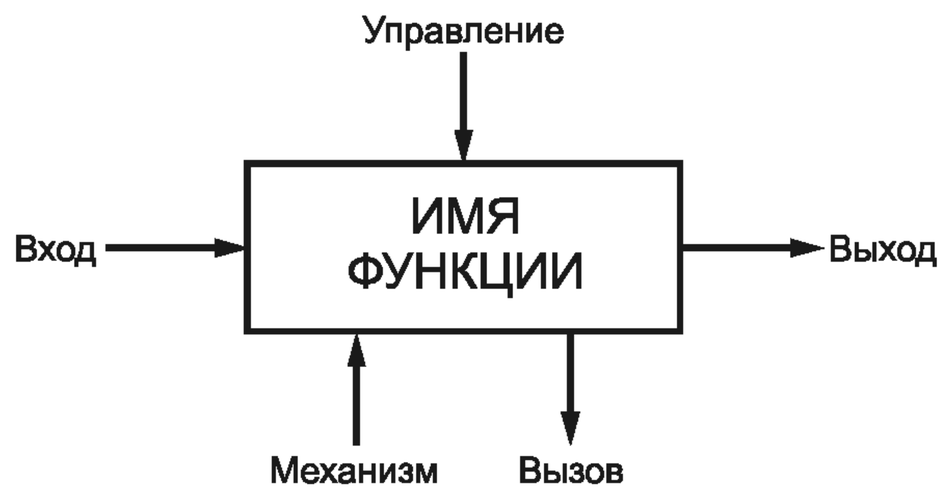


Рисунок 3

6.2 Имена и метки

Как указывалось, имена функций — глаголы или глагольные обороты.

Примеры таких имен:

производить детали	планировать ресурсы	наблюдать
наблюдать за выполнением	проектировать систему	эксплуатировать
разработать детальные чертежи	изготовить компонент	проверять деталь

Стрелки идентифицируют данные или материальные объекты, необходимые для выполнения функции или производимые ею. Каждая стрелка должна быть помечена существительным или оборотом существительного, например:

Спецификации	отчет об испытаниях	бюджет
Конструкторские требования	конструкция детали	директива
Инженер-конструктор	плата в сборе	требования

Пример размещения меток стрелок и имени блока показан на рисунке 4.

6.3 Сводка семантических правил для блоков и стрелок

а) Имя блока должно быть глаголом или глагольным оборотом.

б) Каждая сторона функционального блока имеет стандартное отношение блок/стрелки:

- входные стрелки должны связываться с левой стороной блока;
- управляющие стрелки должны связываться с верхней стороной блока;
- выходные стрелки должны связываться с правой стороной блока;
- стрелки механизма (кроме стрелок вызова) должны указывать вверх и подключаться к нижней стороне блока;

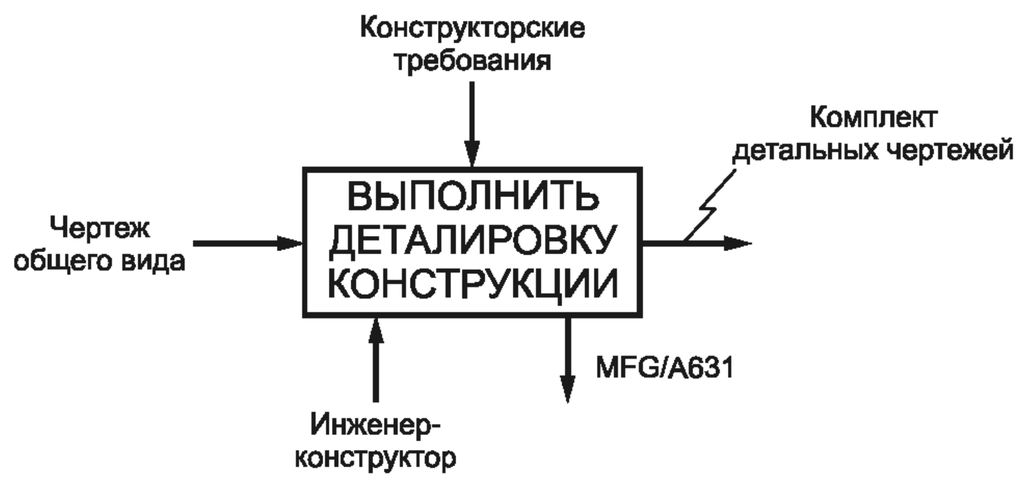


Рисунок 4

- стрелки вызова механизма должны указывать вниз, подключаться к нижней стороне блока, и помечаться ссылкой на вызываемый блок.

в) Сегменты стрелок, за исключением стрелок вызова, должны помечаться существительным или оборотом существительного, если только единственная метка стрелки не относится к стрелке в целом.

г) Чтобы связать стрелку с меткой, следует использовать ломаную молниеобразную выносную линию ( ).

д) В метках стрелок **не должны** использоваться следующие термины: функция, вход, управление, выход, механизм, вызов.

6.4 Диаграммы IDEF0

IDEF0-модели состоят из документов трех типов: графических диаграмм, текста и глоссария. Эти документы имеют перекрестные ссылки друг на друга. Графическая диаграмма — главный компонент IDEF0-модели, содержащий блоки, стрелки, соединения блоков и стрелок и ассоциированные с ними отношения. Блоки представляют основные функции моделируемого объекта. Эти функции могут быть разбиты (декомпозированы) на составные части и представлены в виде более подробных диаграмм; процесс декомпозиции продолжается до тех пор, пока объект не будет описан на уровне детализации, необходимом для достижения целей конкретного проекта.

Диаграмма верхнего уровня обеспечивает наиболее общее описание объекта моделирования. За этой диаграммой следует серия дочерних диаграмм, дающих более детальное представление об объекте.

6.5 Контекстная диаграмма верхнего уровня

Каждая модель должна иметь контекстную диаграмму верхнего уровня, на которой объект моделирования представлен единственным блоком с граничными стрелками. Эта диаграмма называется А—0 (А минус ноль). Стрелки на этой диаграмме отображают связи объекта моделирования с окружающей средой. Поскольку единственный блок представляет весь объект, его имя — общее для всего проекта. Это же справедливо и для всех стрелок диаграммы, поскольку они представляют

полный комплект внешних интерфейсов объекта. Диаграмма А—0 устанавливает область моделирования и ее границу. Пример диаграммы А—0 показан на рисунке 5.

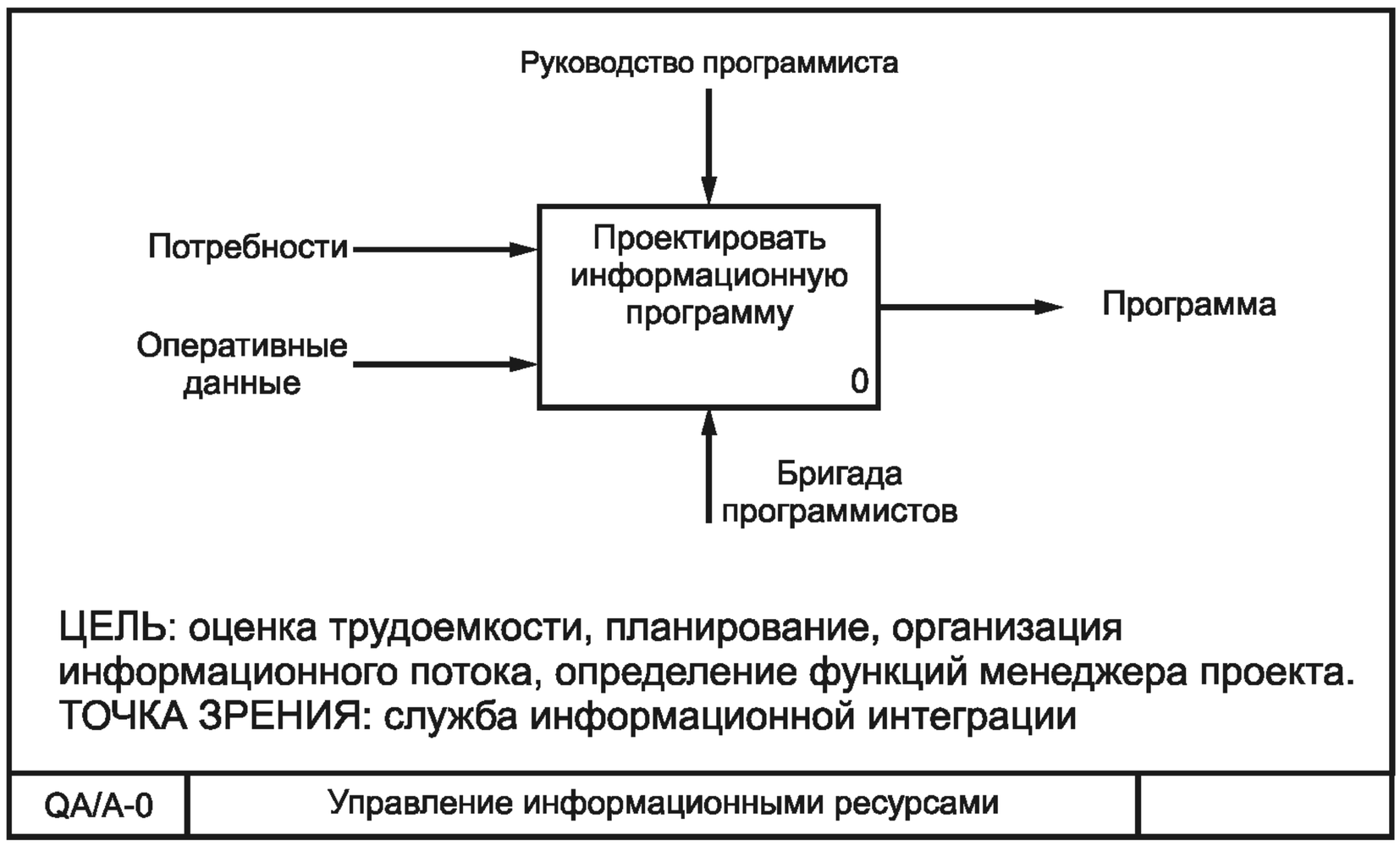


Рисунок 5

Контекстная диаграмма А—0 также должна содержать краткие утверждения, определяющие точку зрения должностного лица или подразделения, с позиций которого создается модель, и цель, для достижения которой ее разрабатывают. Эти утверждения помогают руководить разработкой модели и ввести этот процесс в определенные рамки. Точка зрения определяет, что и в каком разрезе можно увидеть в пределах контекста модели. Изменение точки зрения приводит к рассмотрению других аспектов объекта. Аспекты, важные с одной точки зрения, могут не появиться в модели, разрабатываемой с другой точки зрения на тот же самый объект.

Формулировка цели выражает причину создания модели, то есть содержит перечень вопросов, на которые должна отвечать модель, что в значительной мере определяет ее структуру. Наиболее важные свойства объекта обычно выявляются на верхних уровнях иерархии; по мере декомпозиции функции верхнего уровня и разбиения ее на подфункции, эти свойства уточняются. Каждая подфункция, в свою очередь, декомпозируется на элементы следующего уровня, и так происходит до тех пор, пока не будет получена релевантная структура, позволяющая ответить на вопросы, сформулированные в цели моделирования. Каждая подфункция моделируется отдельным блоком. Каждый родительский блок подробно описывается дочерней диаграммой на более низком уровне. Все дочерние диаграммы должны быть в пределах области контекстной диаграммы верхнего уровня.

6.6 Дочерняя диаграмма

Единственная функция, представленная на контекстной диаграмме верхнего уровня, может быть разложена на основные подфункции посредством создания **дочерней диаграммы.** В свою очередь, каждая из этих подфункций может быть разложена на составные части посредством создания дочерней диаграммы следующего, более низкого уровня, на которой некоторые или все функции также могут быть разложены на составные части. Каждая дочерняя диаграмма содержит дочерние блоки и стрелки, обеспечивающие дополнительную детализацию родительского блока.

Дочерняя диаграмма, создаваемая при декомпозиции, охватывает ту же область, что и родительский блок, но описывает ее более подробно.

Таким образом, дочерняя диаграмма как бы вложена в свой родительский блок. Эта структура иллюстрируется рисунком 6.

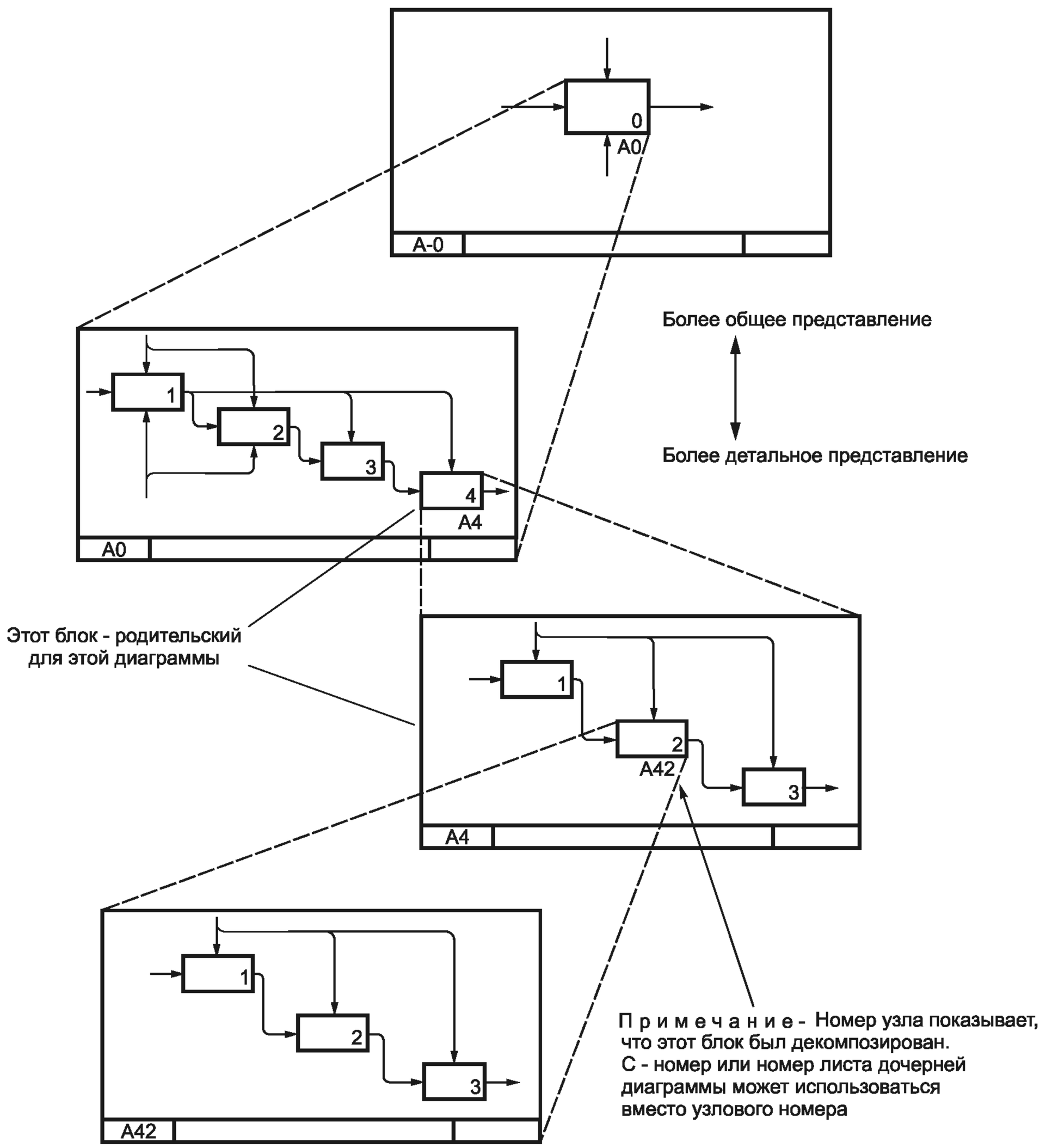


Рисунок 6

6.7 Родительская диаграмма

Родительская диаграмма — та, которая содержит один или более родительских блоков. Иными словами, это диаграмма верхнего уровня, на которой располагается декомпозируемая функция со всеми ее связями. Каждая обычная (неконтекстная) диаграмма является также дочерней диаграммой, поскольку, по определению, она подробно описывает некоторый родительский блок. Таким образом, любая диаграмма может быть как родительской диаграммой (содержать родительские блоки), так и дочерней (подробно описывать собственный родительский блок). Аналогично, блок может быть как родительским (подробно описываться дочерней диаграммой), так и дочерним (появляющимся на дочерней диаграмме). Основное иерархическое отношение существует между родительским блоком и дочерней диаграммой, которая его подробно описывает (рисунок 6).

То, что блок является дочерним и раскрывает содержание родительского блока на диаграмме предшествующего уровня, указывается специальным ссылочным кодом, написанным ниже правого нижнего угла блока. Этот ссылочный код может формироваться несколькими способами, из которых самый простой заключается в том, что код, начинающийся с буквы А (по имени диаграммы А—0), содержит цифры, определяемые номерами родительских блоков. Например показанные на рисунке 7 коды означают, что диаграмма является декомпозицией 1-го блока диаграммы, которая, в свою очередь, является декомпозицией 6-го блока диаграммы А0, а сами коды образуются присоединением номера блока.

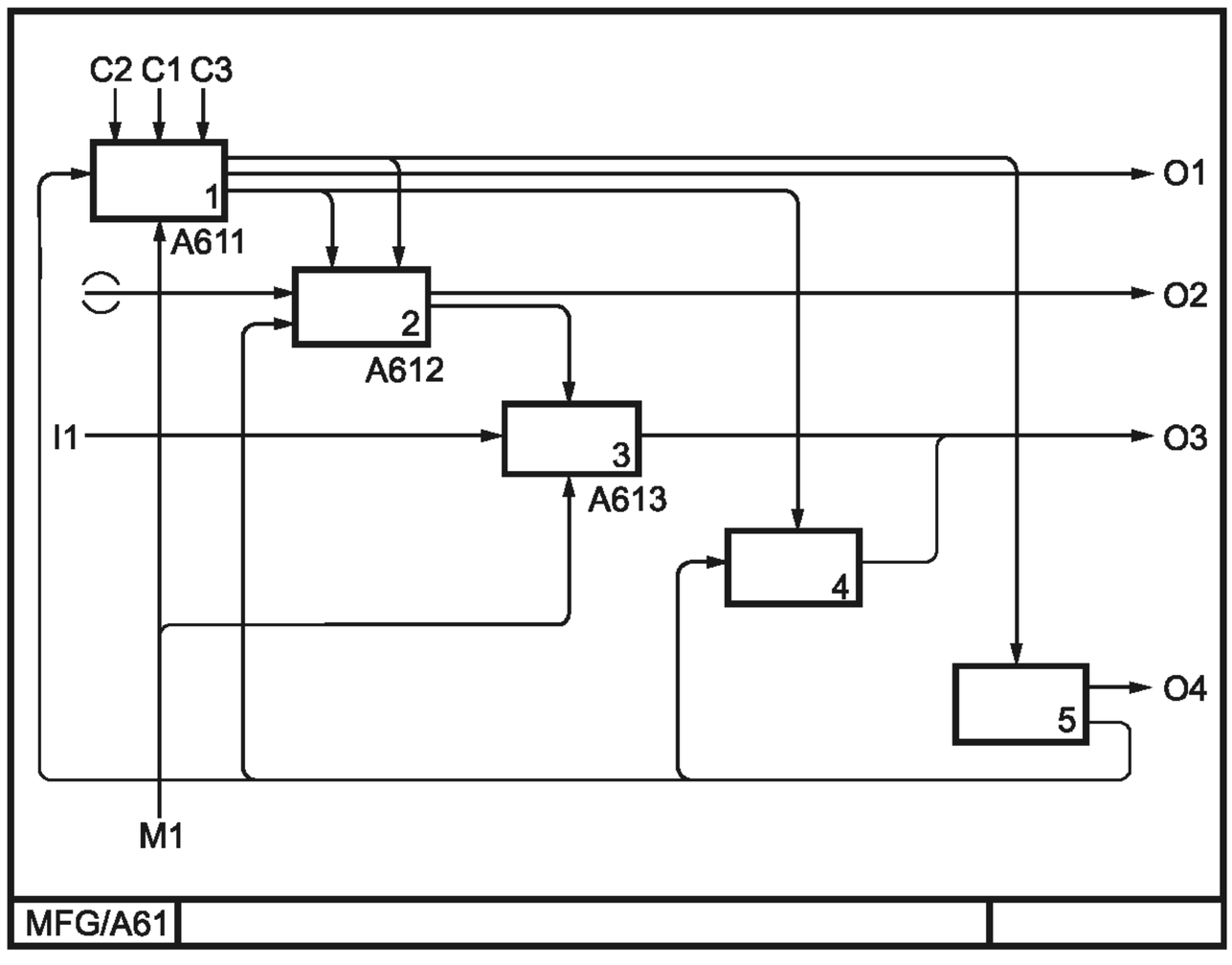
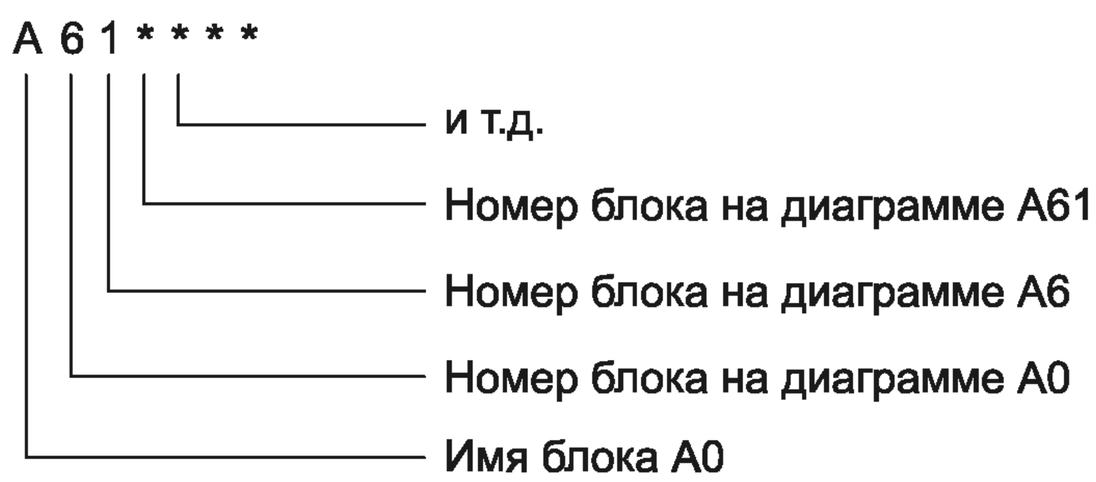


Рисунок 7

Следовательно, код формируется так:



6.8 Текст и глоссарий

Диаграмме может быть поставлен в соответствие структурированный текст, представляющий собой краткий комментарий к содержанию диаграммы.

Текст используется для объяснений и уточнений характеристик, потоков, внутриблочных соединений и т.д. Текст не должен использоваться для описания и без того понятных блоков и стрелок на диаграммах. При большом объеме текст располагается на отдельном листе модели (текстовой странице).

Глоссарий предназначен для определения аббревиатур, ключевых слов и фраз, используемых в качестве имен и меток на диаграммах. Глоссарий определяет понятия и термины, которые должны быть одинаково понимаемы всеми участниками разработки и пользователями модели, чтобы правильно интерпретировать ее содержание. Глоссарий составляется с любой необходимой степенью подробности.

6.9 Диаграммы-иллюстрации (FEO)

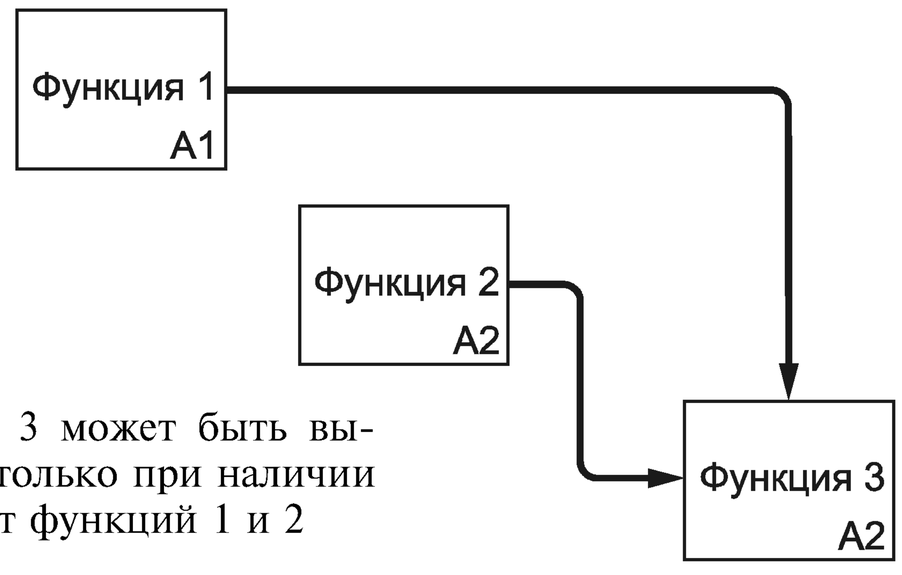
Эти диаграммы используются в качестве дополнений, поясняющих специфику содержания основных диаграмм в случаях, когда это необходимо. Диаграмма FEO не должна подчиняться синтаксическим правилам IDEF0.

7 Свойства диаграмм

7.1 Стрелки как ограничения

Стрелки на диаграмме IDEF0, представляя данные или материальные объекты, одновременно задают своего рода ограничения (условия). Входные и управляющие стрелки блока, соединяющие его с другими блоками или внешней средой, описывают условия, которые должны быть выполнены для того, чтобы реализовалась функция, записанная в качестве имени блока.

Рисунок 8 иллюстрирует случай, при котором «функция 3» может быть выполнена только после получения данных, выработанных «функцией 1» и «функцией 2».



Функция 3 может быть выполнена только при наличии данных от функций 1 и 2

Рисунок 8

7.2 Параллельное функционирование

Различные функции в модели могут быть выполнены параллельно, если удовлетворяются необходимые ограничения (условия). Как показано на рисунке 9, один блок может создать данные (а) или материальные объекты (б), необходимые для параллельной работы нескольких блоков.

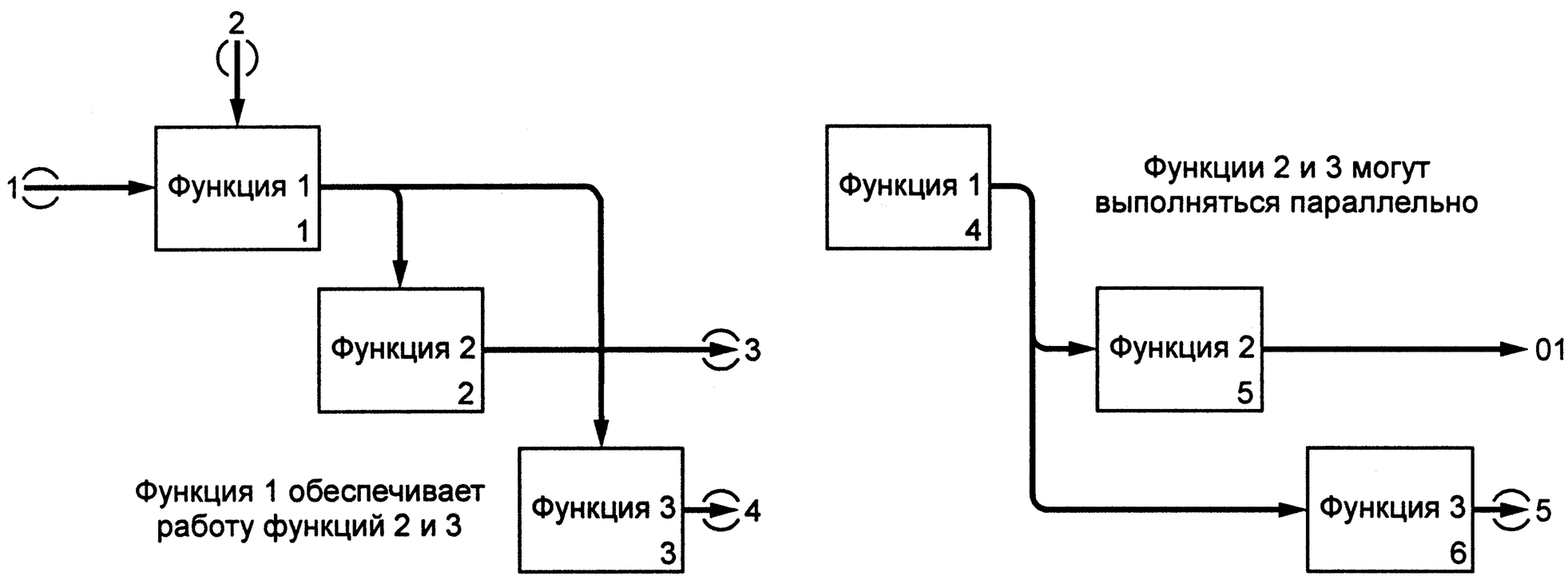


Рисунок 9

7.3 Ветвление и слияние сегментов стрелок

Ветвление и слияние стрелок призвано уменьшить загруженность диаграмм графическими элементами (линиями).

Чтобы стрелки и их сегменты правильно описывали связи между блоками-источниками и

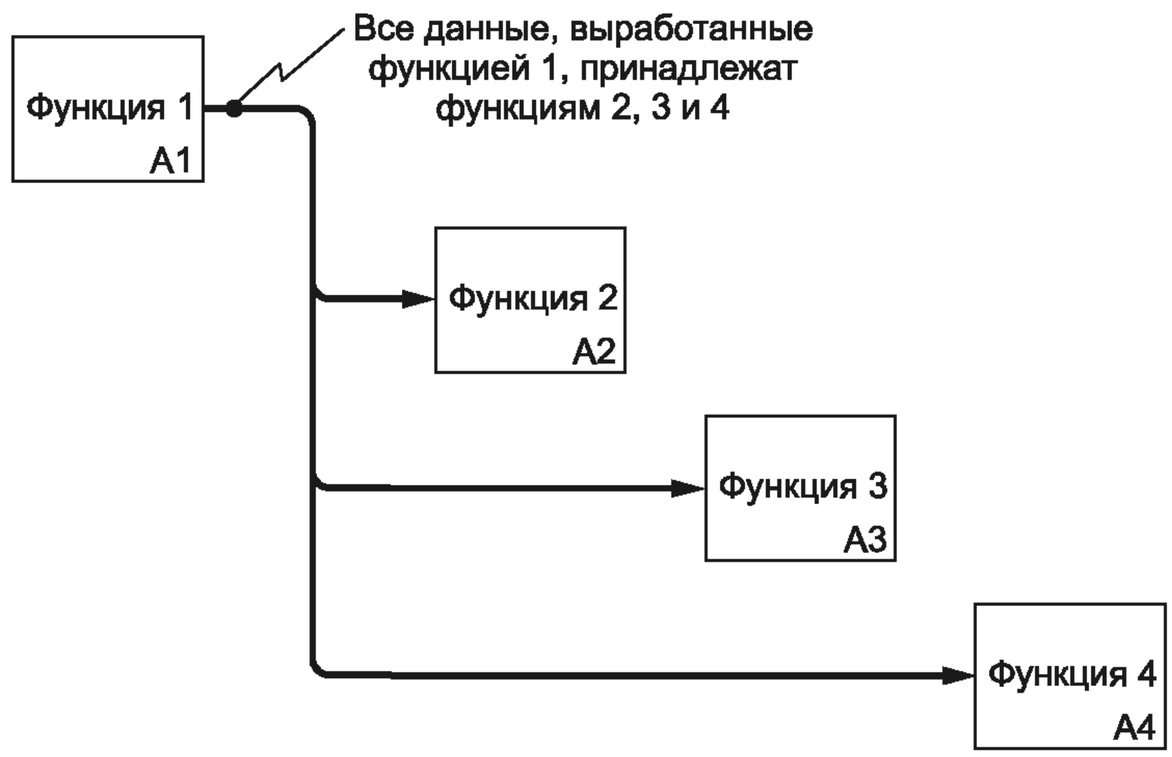


Рисунок 10

блоками-потребителями, используется аппарат меток. Метки связываются с сегментами посредством тильд. При этом между сегментами возникают определенные отношения, описанные ниже:

- непомеченные сегменты (рисунок 10) содержат все объекты, указанные в метке стрелки перед ветвлением (то есть все объекты принадлежат каждому из сегментов);
- сегменты, помеченные после точки ветвления (рисунок 11), содержат все объекты, указанные в метке стрелки перед ветвлением, или их часть, описываемую меткой каждого конкретного сегмента;
- при слиянии непомеченных сегментов объединенный сегмент стрелки содержит все объекты, принадлежащие сливаемым сегментам и указанные в общей метке стрелки после слияния (рисунок 12);
- при слиянии помеченных сегментов (рисунок 13) объединенный сегмент содержит все или некоторые объекты, принадлежащие сливаемым сегментам и перечисленные в общей метке после слияния; если общая метка после слияния отсутствует, это означает, что общий сегмент передает все объекты, принадлежащие сливаемым сегментам;

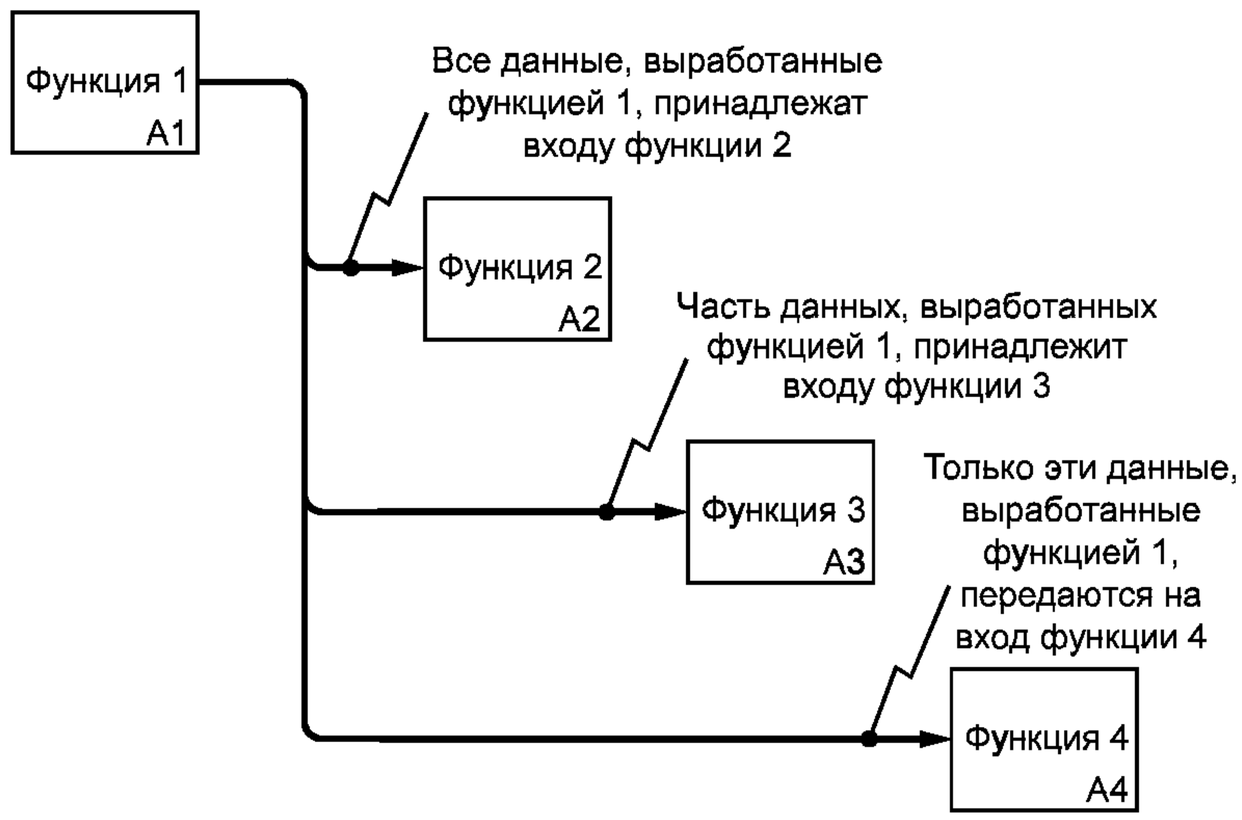


Рисунок 11

7.4 Отношения блоков на диаграммах

В методологии IDEF0 существует шесть типов отношений между блоками в пределах одной диаграммы:

- доминирование;
- управление;
- выход — вход;
- обратная связь по управлению;
- обратная связь по входу;
- выход — механизм.

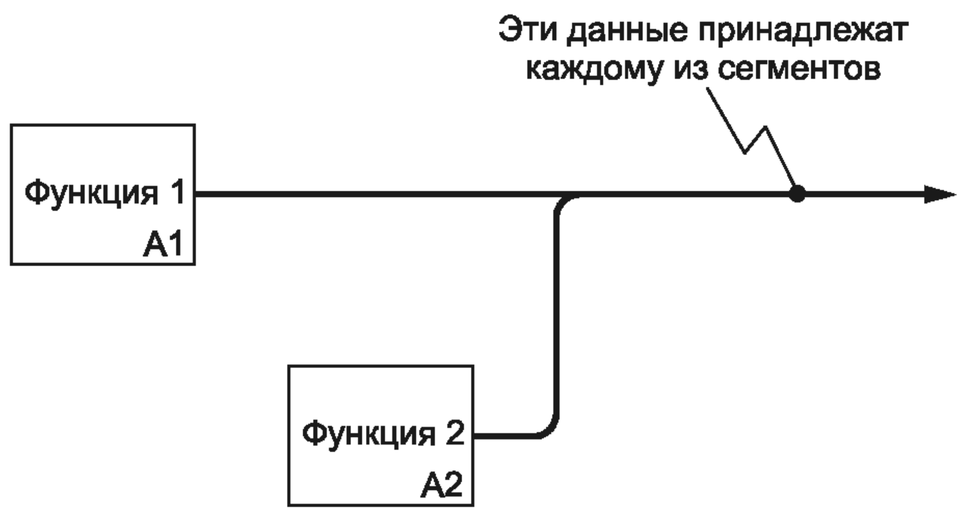


Рисунок 12

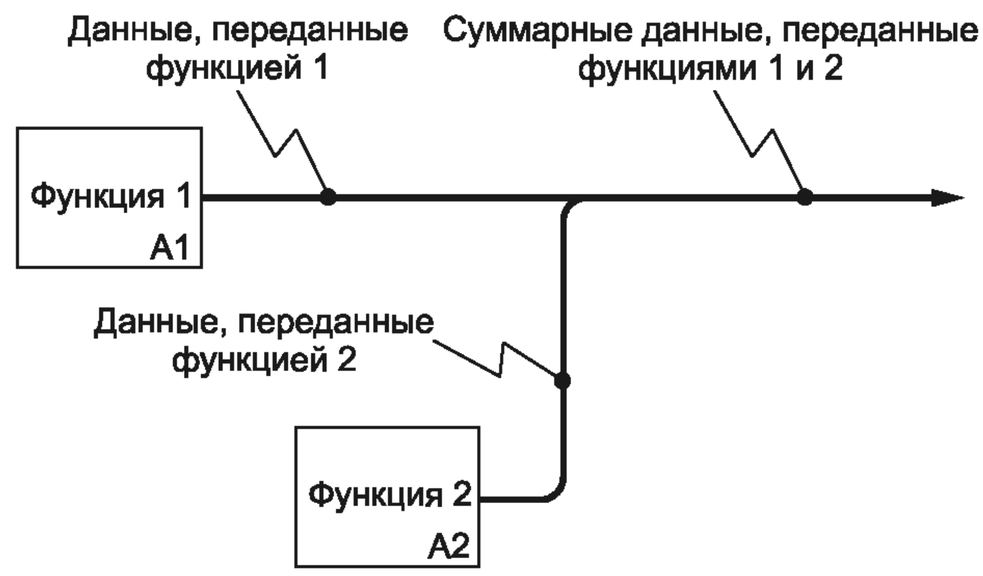


Рисунок 13

Первое из перечисленных отношений определяется взаимным расположением блоков на диаграмме. Предполагается, что блоки, расположенные на диаграмме выше и левее, «**доминируют**» над блоками, расположенными ниже и правее. «**Доминирование**» понимается как влияние, которое один блок оказывает на другие блоки диаграммы.

Остальные пять отношений описывают связи между блоками и изображаются соответствующими стрелками.

Отношения управления и выход — вход являются простейшими, поскольку отражают прямые взаимодействия, которые понятны и очевидны.

Отношение управления (рисунок 14) возникает, когда выход одного блока служит управляющим воздействием на блок с меньшим доминированием.

Отношение выход — вход (рисунок 15) возникает при соединении выхода одного блока с входом другого блока с меньшим доминированием.

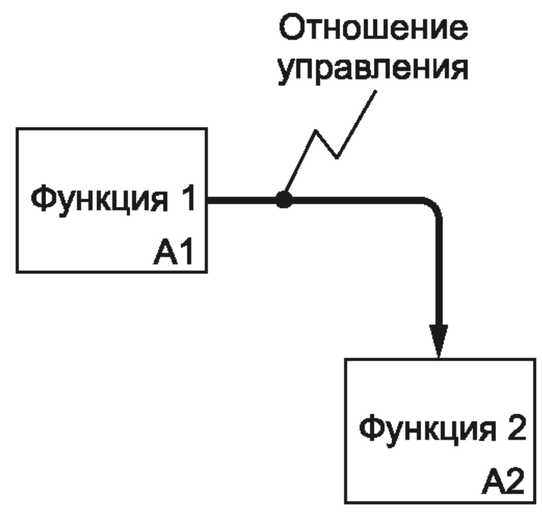


Рисунок 14

Обратная связь по управлению и обратная связь по входу являются более сложными типами отношений, поскольку они представляют итерацию (выход функции влияет на будущее выполнение других функций с большим доминированием, что впоследствии влияет на исходную функцию).

Обратная связь по управлению (рисунок 16) возникает, когда выход некоторого блока создает управляющее воздействие на блок с большим доминированием.

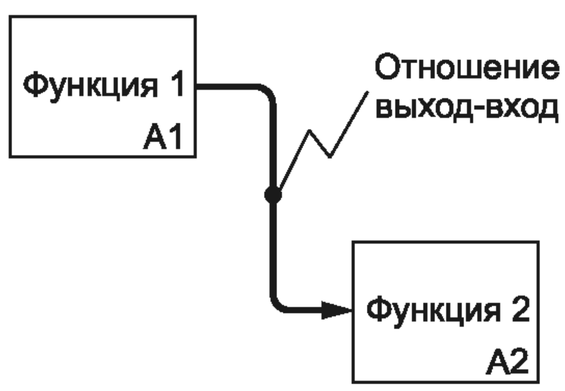


Рисунок 15

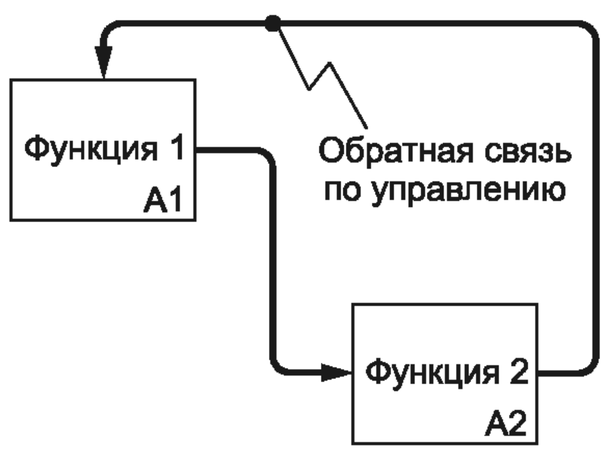


Рисунок 16

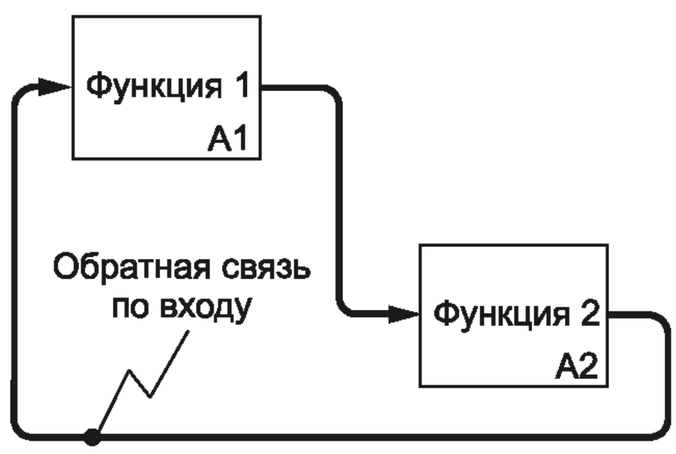


Рисунок 17

Отношение обратной связи по входу (рисунок 17) имеет место, когда выход блока становится входом другого блока с **большим** доминированием.

Связи «выход — механизм» (рисунок 18) отражают ситуацию, при которой выход одной функции становится средством достижения цели для другой. Связи «выход — механизм» возникают при отображении в модели процедур пополнения и распределения ресурсов, создания или подготовки

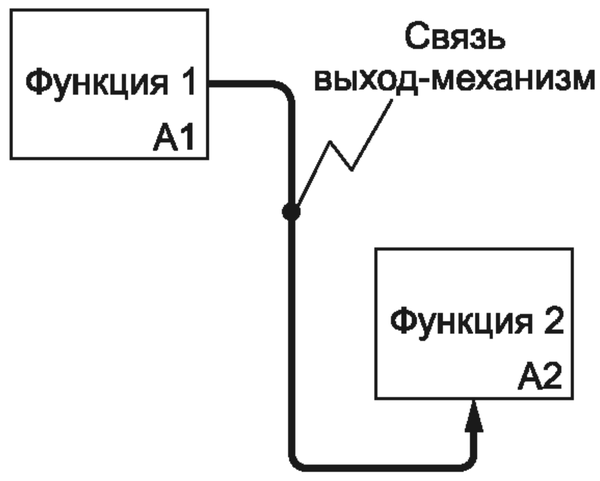


Рисунок 18

средств для выполнения функций системы (например приобретение или изготовление требуемых инструментов и оборудования, обучение персонала, организация физического пространства, финансирование, закупка материалов и т.д.; подробнее — см. 11.3).

8 Отношения между блоками диаграммы и другими диаграммами (окружающей средой)

Все описанные выше отношения отображаются внутренними стрелками, то есть такими, у которых оба конца связаны с блоками диаграммы. Отношения между блоками диаграммы и другими диаграммами, являющимися по отношению к рассматриваемой диаграмме окружающей средой (окружением), описываются граничными стрелками (см. 2.7). Обе ситуации отражены на рисунке 19.

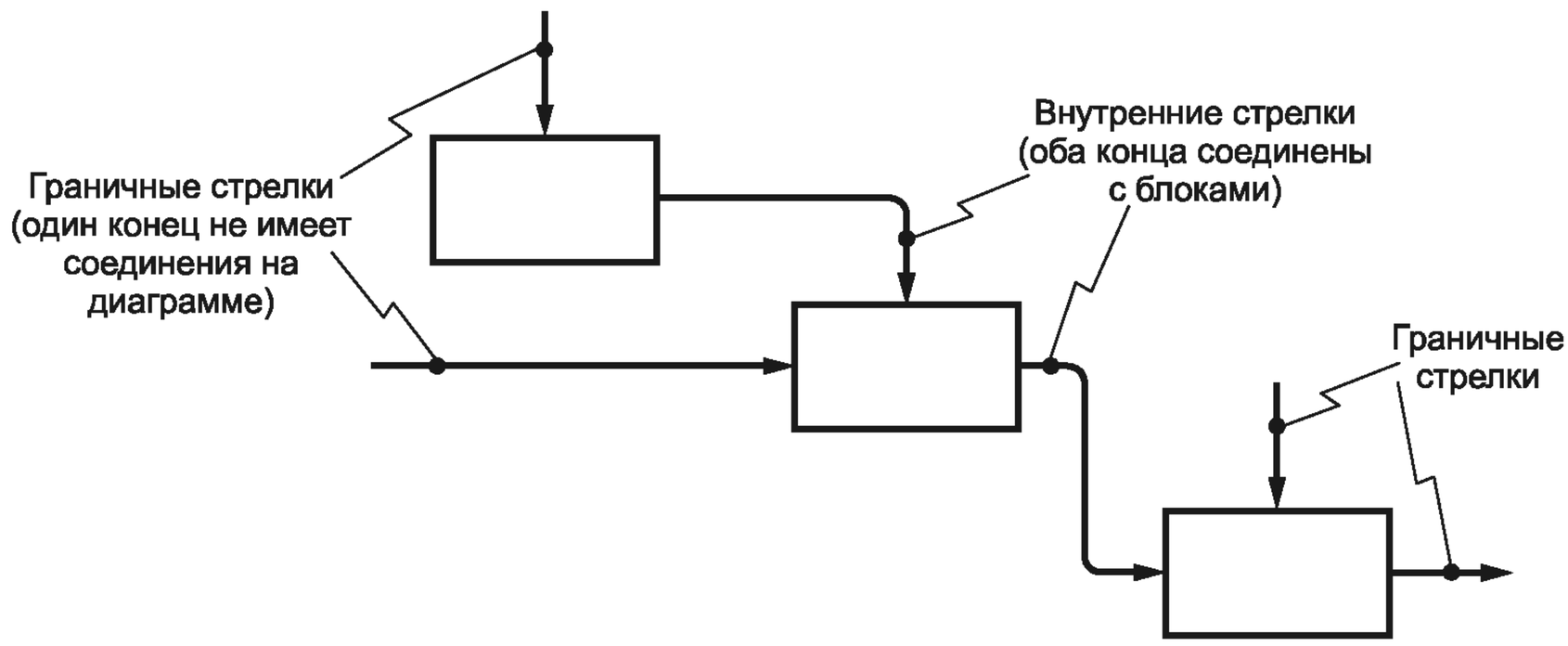


Рисунок 19

8.1 Граничные стрелки

На обычной (неконтекстной) диаграмме граничные стрелки представляют входы, управления, выходы или механизмы родительского блока диаграммы. Источник или потребитель граничных стрелок можно обнаружить, только изучая родительскую диаграмму. Все граничные стрелки на дочерней диаграмме (за исключением стрелок, помещенных в туннель, см. 2.36) должны соответствовать стрелкам родительского блока, как показано на рисунке 20.

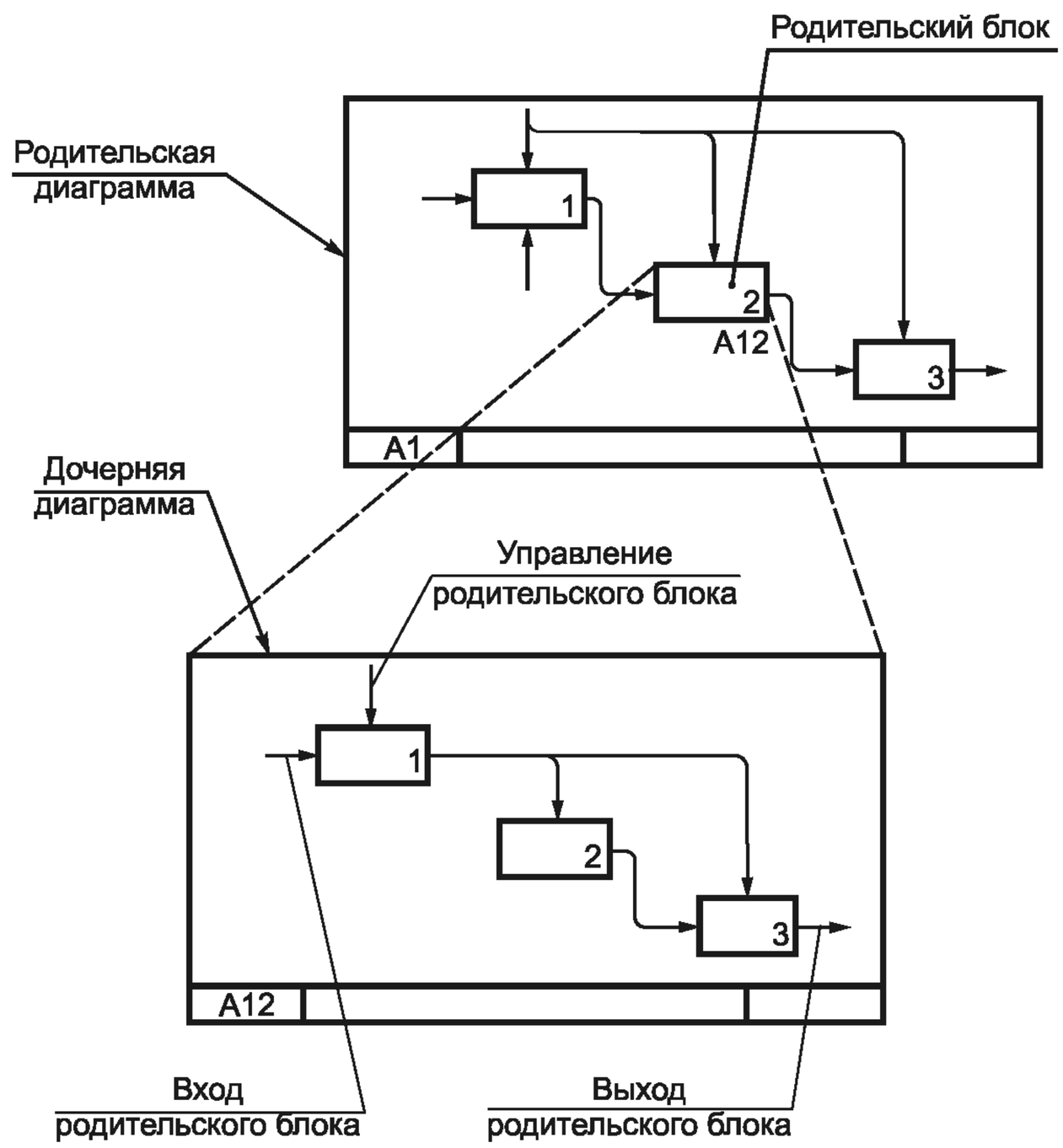


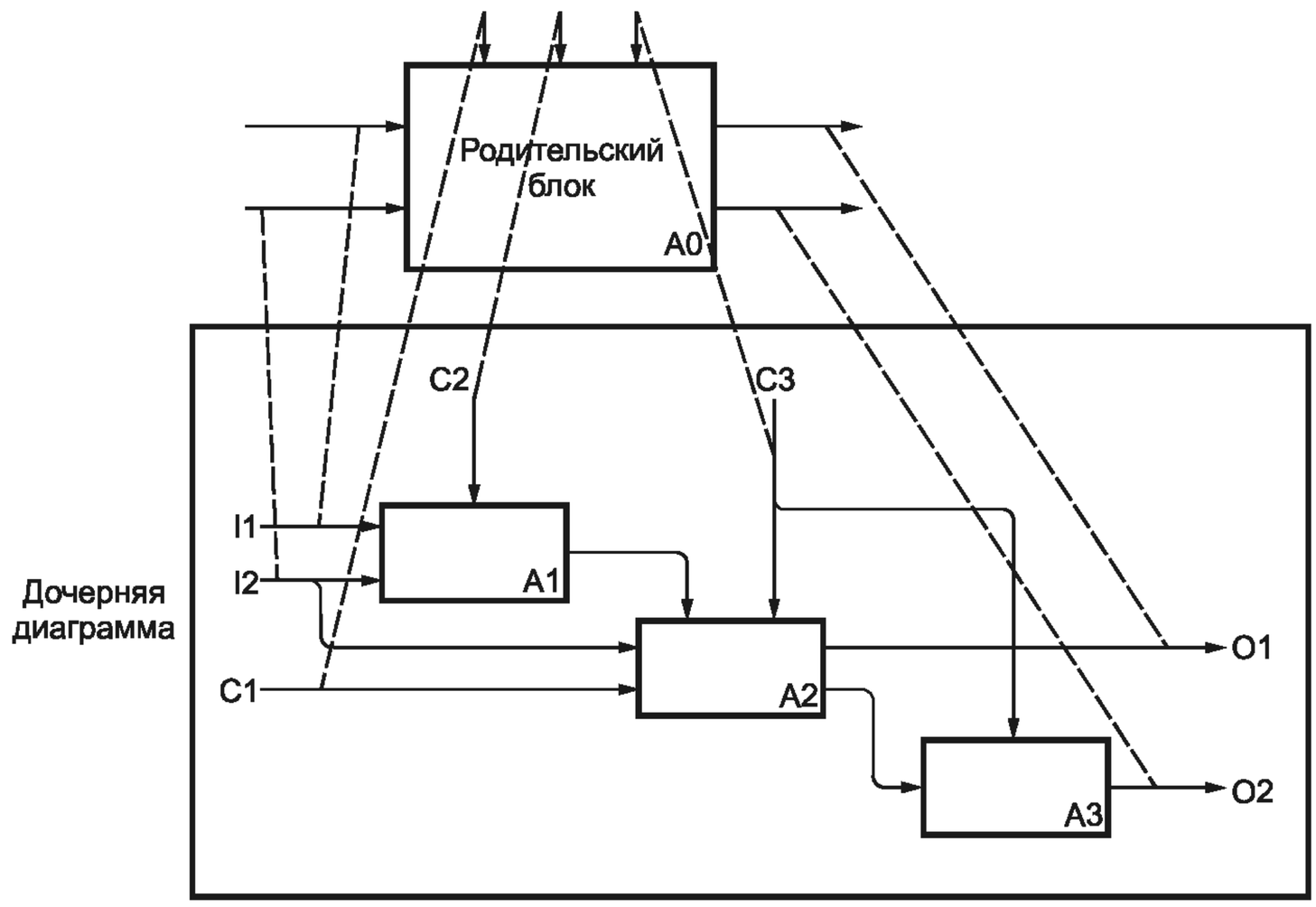
Рисунок 20

8.2 ICOM-кодирование граничных стрелок

ICOM-коды связывают граничные стрелки на дочерней диаграмме со стрелками родительского блока. Нотация, названная ICOM-кодом, определяет значения соединений. Буквы I, C, O или M, приведенные около несвязанного конца граничной стрелки на дочерней диаграмме, идентифицируют стрелку как Вход (**I**nput), Управление (**C**ontrol), Выход (**O**utput) или Механизм (**M**echanism) в родительском блоке. Буква следует за числом, определяющим относительное положение точки подключения стрелки к родительскому блоку; это положение определяется слева направо или сверху вниз. Например, код «3C» возле граничной стрелки на дочерней диаграмме указывает, что эта стрелка соответствует третьей (считая слева) управляющей стрелке родительского блока.

Это кодирование связывает каждую дочернюю диаграмму со своим родительским блоком. Если блоки на дочерней диаграмме подвергаются дальнейшей декомпозиции и подробно описываются на дочерних диаграммах следующего уровня, то на каждую новую диаграмму назначаются новые ICOM-коды, связывающие граничные стрелки этих диаграмм со стрелками их родительских блоков.

Иногда буквенные ICOM-коды, определяющие роли граничных стрелок (вход, управление, механизм), могут меняться при переходе от родительского блока к дочерней диаграмме. Например управляющая стрелка в родительском блоке может быть входом на дочерней диаграмме. Аналогично, вход родительского блока может быть управлением для одного или нескольких дочерних блоков. Примеры изменения ролей стрелок можно видеть на рисунке 21.



П р и м е ч а н и е — Штриховые линии показывают отношения между граничными стрелками и стрелками родительского блока.

Рисунок 21

8.3 Стрелки, помещенные в «туннель»

Туннель — круглые скобки в начале и/или в конце стрелки. Туннельные стрелки означают, что данные, выраженные этими стрелками, не рассматриваются на родительской диаграмме и/или на дочерней диаграмме.

Стрелка, помещенная в туннель там, где она присоединяется к блоку (рисунок 22), означает, что данные, выраженные этой стрелкой, не обязательны на следующем уровне декомпозиции.

Стрелка, помещаемая в туннель на свободном конце (рисунок 23), означает, что представляемые ею данные отсутствуют на родительской диаграмме.

Более детально эта ситуация поясняется на рисунке 24.

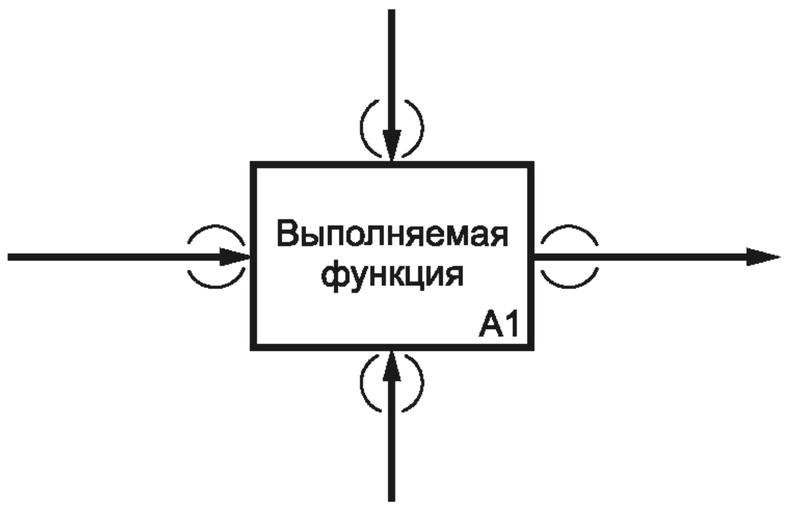


Рисунок 22

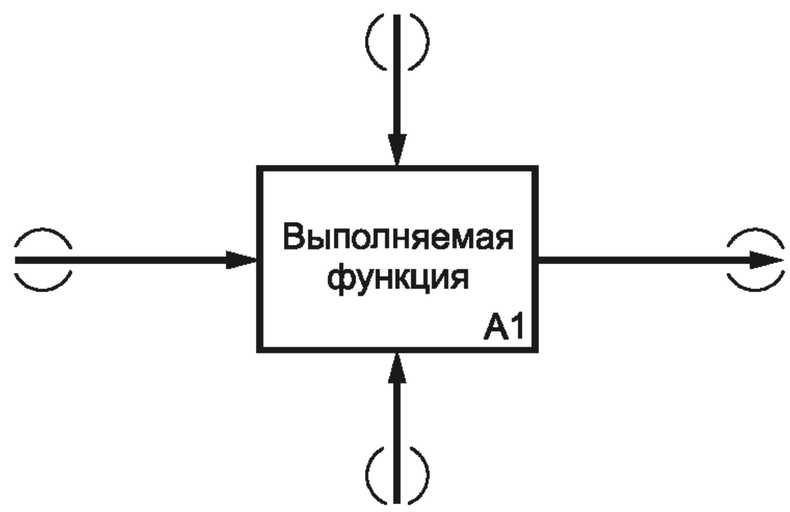


Рисунок 23

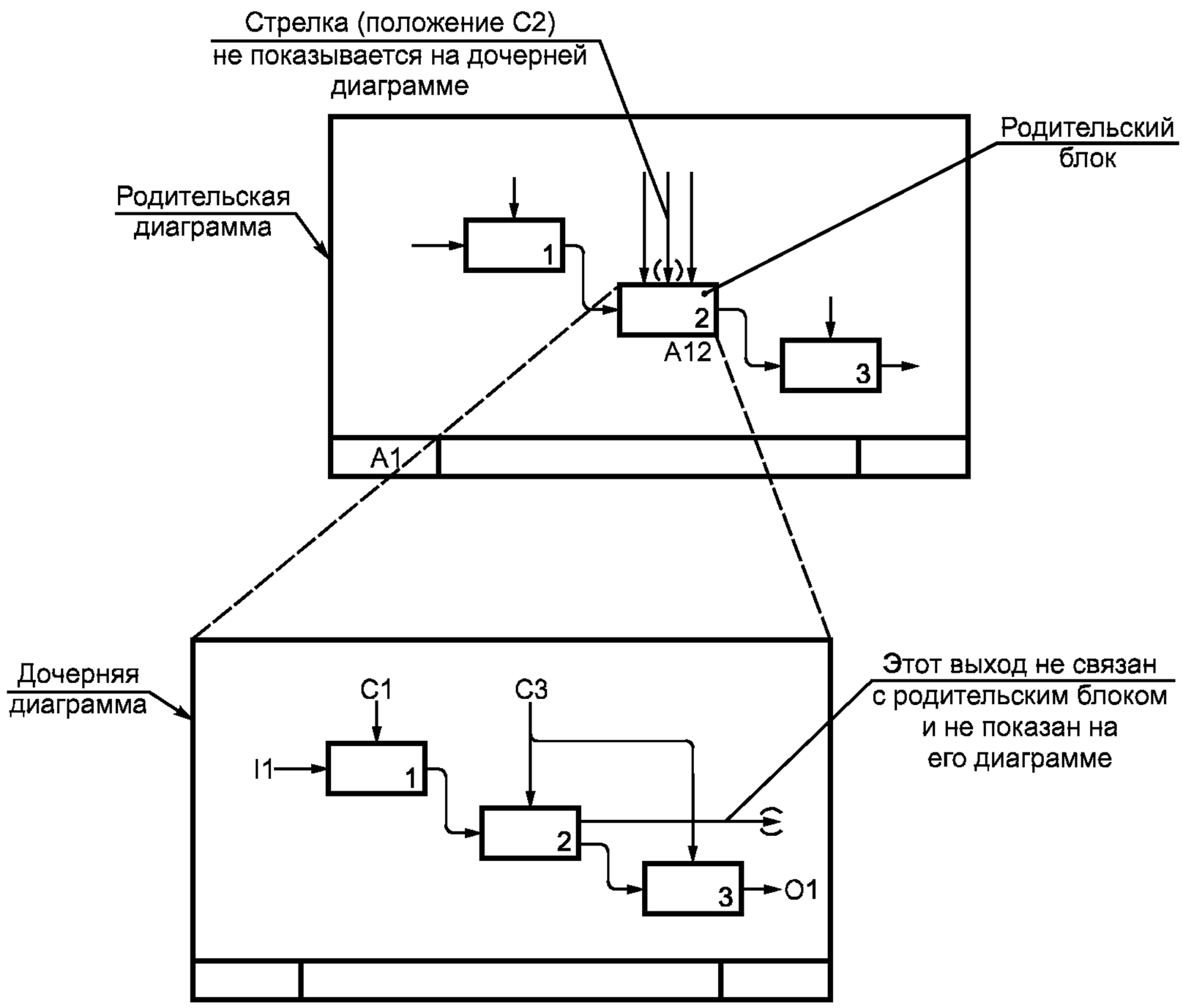


Рисунок 24

9 Правила построения диаграмм

При построении диаграмм необходимо выполнять следующие правила.

1 В составе модели должна присутствовать контекстная диаграмма А—0, которая содержит только один блок. Номер единственного блока на контекстной диаграмме А—0 должен быть 0.

2 Блоки на диаграмме должны располагаться по диагонали — от левого верхнего до правого нижнего угла диаграммы в порядке присвоенных номеров. Блоки на диаграмме, расположенные вверху слева, «**доминируют**» над блоками, расположенными внизу справа. «**Доминирование**» понимается как влияние, которое блок оказывает на другие блоки диаграммы. Расположение блоков на листе диаграммы отражает авторское понимание доминирования. Таким образом, топология диаграммы показывает, какие функции оказывают большее влияние на остальные. При параллельном функционировании блоков отношение доминирования часто носит формальный характер, и диагональное расположение блоков лишь способствует читаемости диаграммы.

3 Диаграммы (кроме диаграммы А—0) должны содержать не менее трех и не более шести блоков. Эти ограничения поддерживают сложность диаграмм на уровне, доступном для чтения, понимания и использования.

Диаграммы с количеством блоков менее трех вызывают серьезные сомнения в необходимости декомпозиции родительской функции. Диаграммы с количеством блоков более шести сложны для восприятия читателями и вызывают у автора трудности при внесении в нее всех необходимых графических объектов и меток.

4 Каждый блок неконтекстной диаграммы получает номер, помещаемый в правом нижнем углу; порядок нумерации — от верхнего левого к нижнему правому блоку (от 1 до 6).

5 Каждый блок, подвергнутый декомпозиции, должен иметь ссылку на дочернюю диаграмму; ссылка (например узловой номер, С-номер или номер страницы) помещается под правым нижним углом блока.

6 Имена блоков (выполняемых функций) и метки стрелок должны быть уникальными. Если метки стрелок совпадают, это значит, что стрелки отображают тождественные данные.

7 При наличии стрелок со сложной топологией целесообразно повторить метку для удобства ее идентификации.

8 Следует обеспечить максимальное расстояние между блоками и поворотами стрелок, а также между блоками и пересечениями стрелок для облегчения чтения диаграммы. Одновременно уменьшается вероятность того, что две разные стрелки будут перепутаны.

9 Блоки всегда должны иметь хотя бы одну управляющую и одну входную стрелку, но могут не иметь выходных стрелок.

10 Если одни и те же данные служат и для управления, и для входа, вычерчивается только стрелка управления. Этим подчеркивается управляющий характер данных и уменьшается сложность диаграммы.

11 Максимально увеличенное расстояние между параллельными стрелками облегчает размещения меток, их чтение и позволяет проследить пути стрелок (рисунок 25).

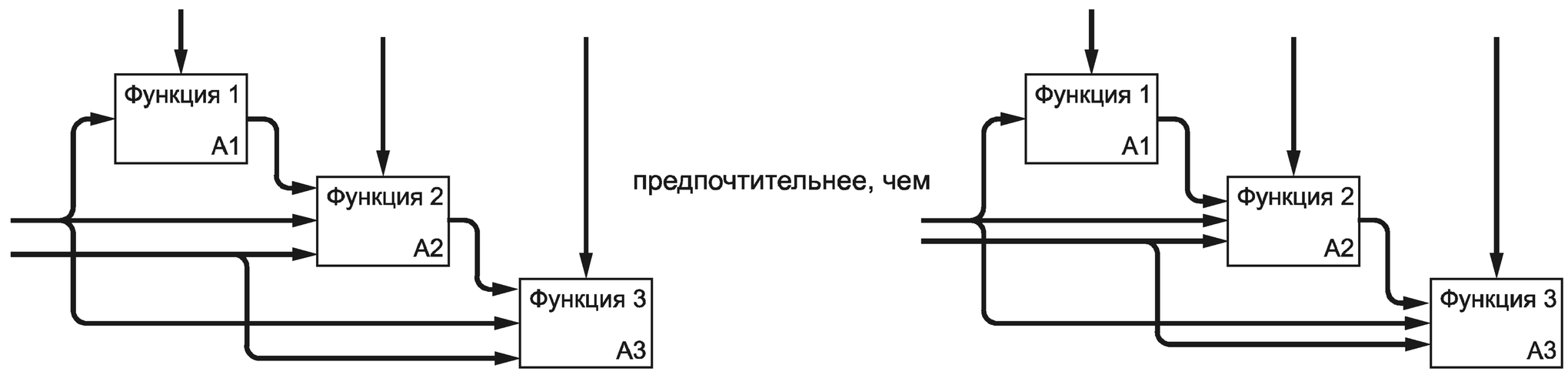


Рисунок 25

12 Стрелки связываются (сливаются), если они представляют сходные данные и их источник не указан на диаграмме (рисунок 26).

Рисунок 26

13 Обратные связи по управлению должны быть показаны как «вверх и над» (рисунок 27а).

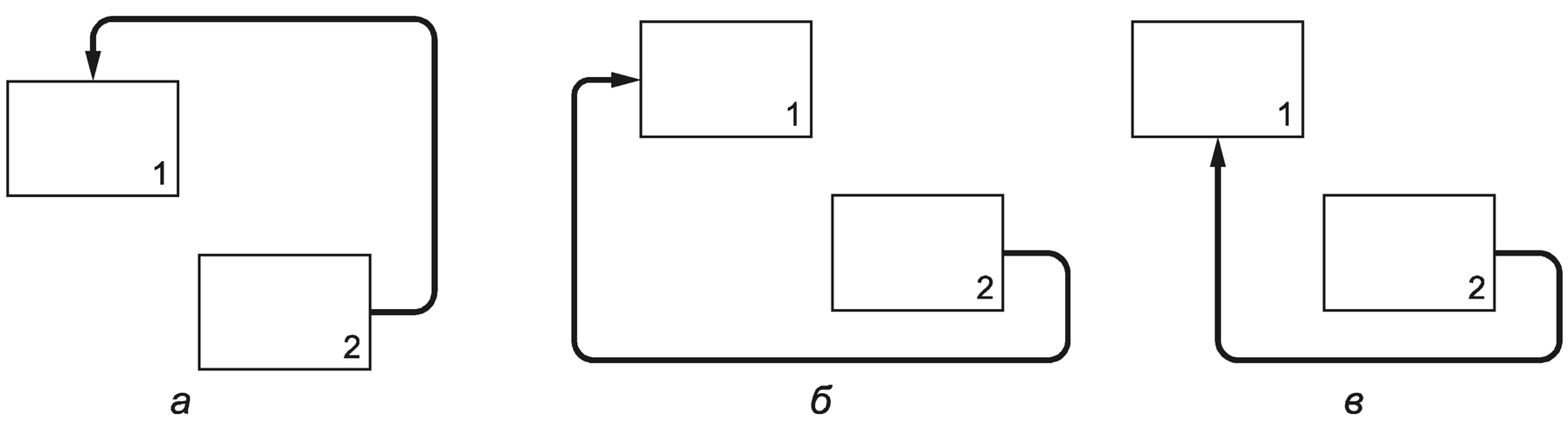


Рисунок 27

Обратные связи по входу должны быть показаны как «вниз и под» (рисунок 27б). Так же показываются обратные связи посредством механизма (рисунок 27в). Таким образом обеспечивается показ обратной связи при минимальном числе линий и пересечений.

14 Циклические обратные связи для одного и того же блока изображаются только для того, чтобы их выделить. Обычно обратную связь изображают на диаграмме, декомпозирующей блок. Однако иногда требуется выделить повторно используемые объекты (рисунок 28).

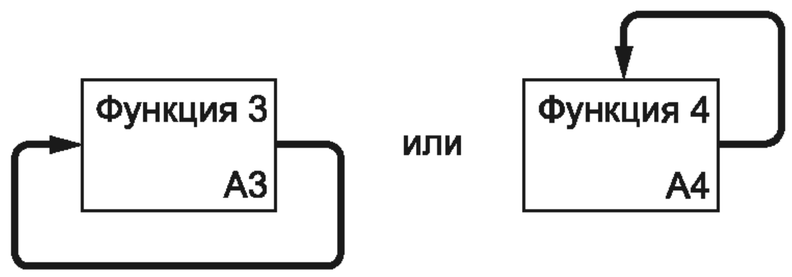


Рисунок 28

15 Стрелки объединяются, если они имеют общий источник или приемник, или они представляют связанные данные. Общее название лучше описывает суть данных. Следует минимизировать число стрелок, касающихся каждой стороны блока, если, конечно, природа данных не слишком разнородна (рисунок 29).

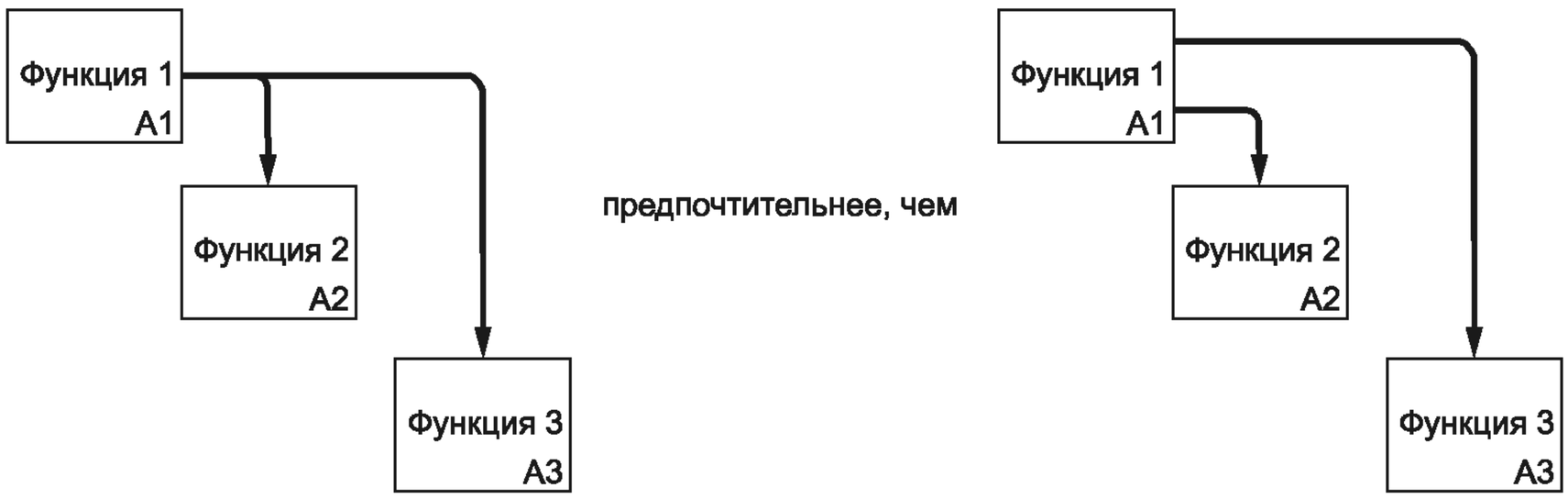


Рисунок 29

16 Если возможно, стрелки присоединяются к блокам в одной и той же позиции. Тогда соединение стрелок конкретного типа с блоками будет согласованным и чтение диаграммы упростится (рисунок 30).

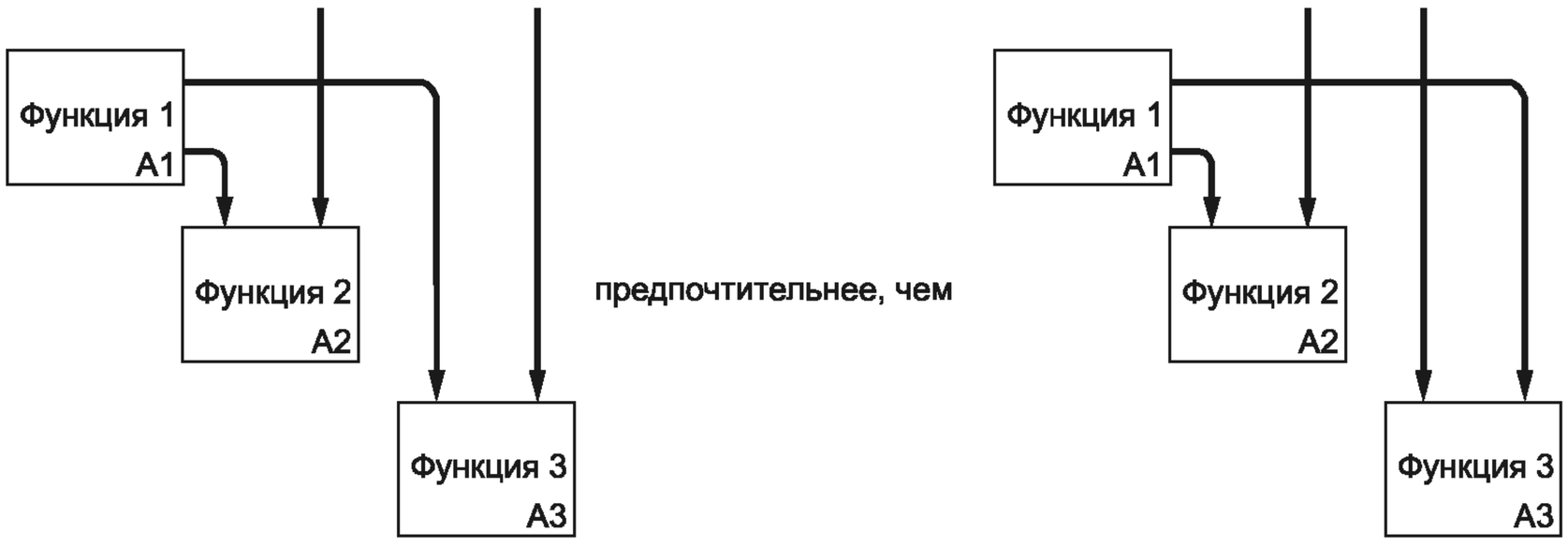


Рисунок 30

17 При соединении большого числа блоков необходимо избегать необязательных пересечений стрелок (рисунок 31). Следует минимизировать число петель и поворотов каждой стрелки (рисунок 32).

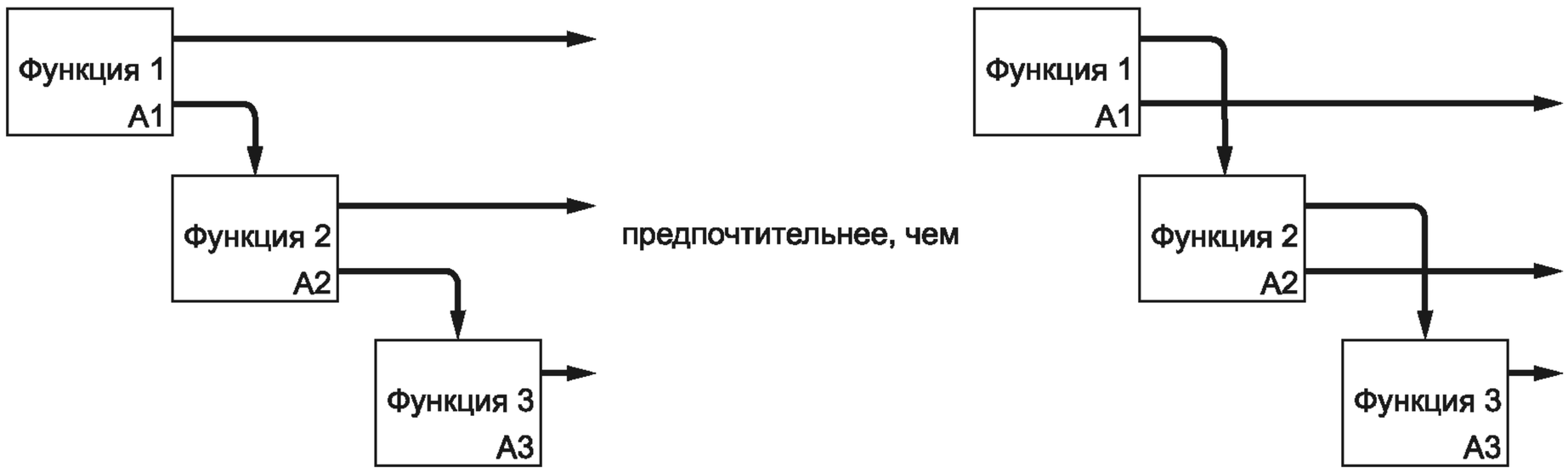


Рисунок 31

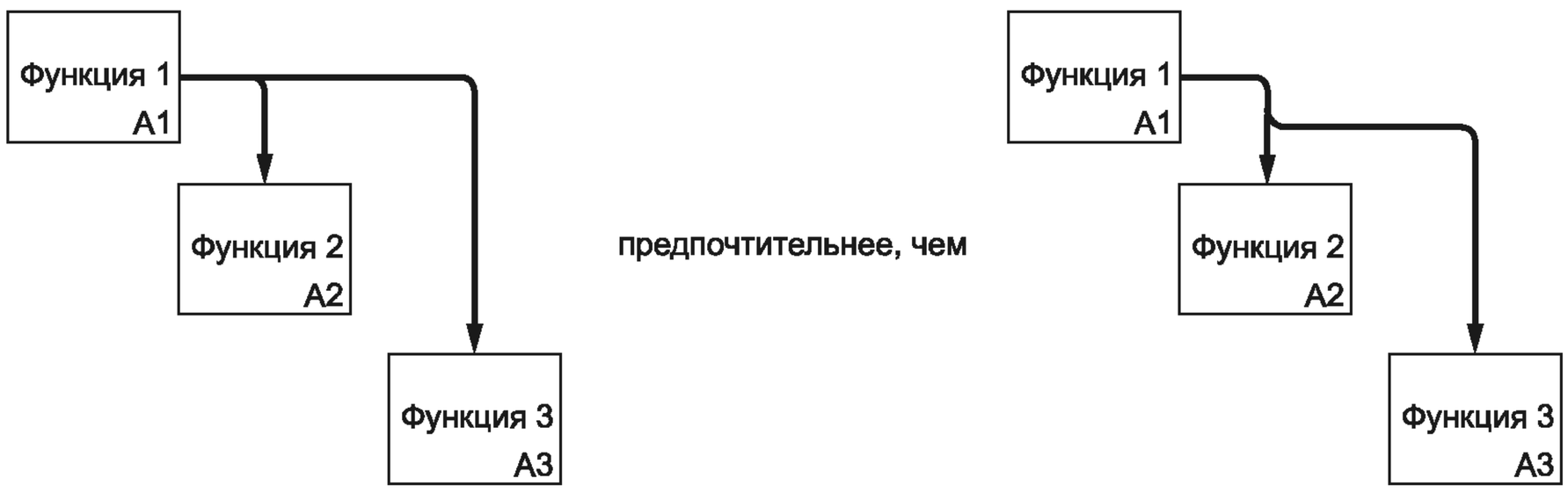


Рисунок 32

18 Блоки (функции) дочерней диаграммы являются сопряженными через среду (диаграмму или родительский блок), если они имеют связи с источником, генерирующим данные, без конкретного определения того, как отдельные части данных относятся к какому-либо блоку (рисунок 33).

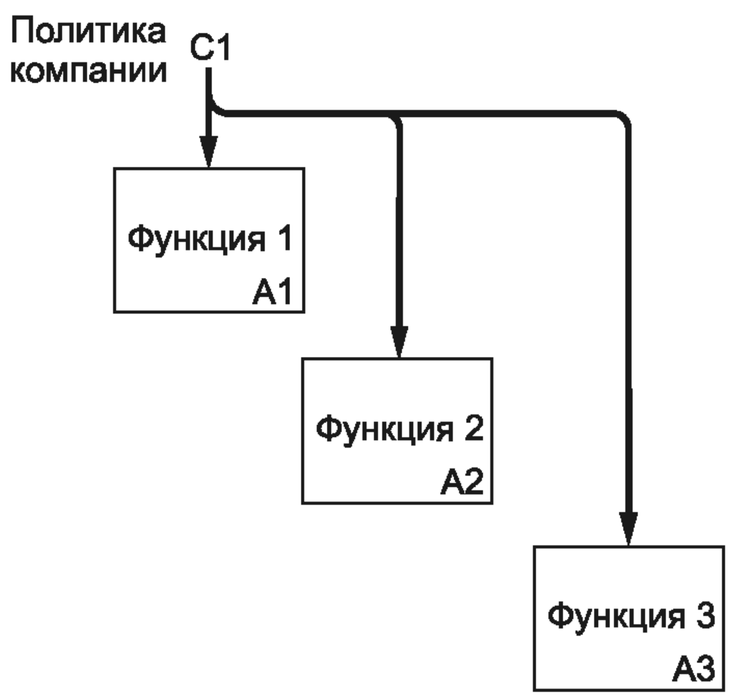


Рисунок 33

19 Тип интерфейса, показанный на рисунке 34, предпочтителен, поскольку в этом случае определяются конкретные данные, относящиеся к каждому блоку.

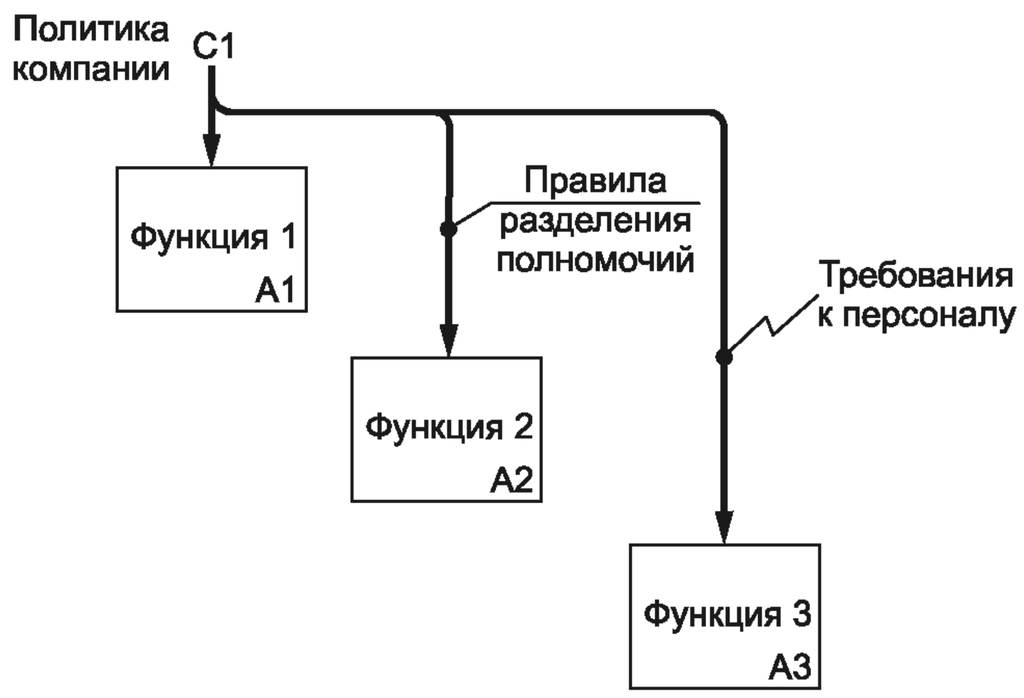


Рисунок 34

20 Необходимо использовать (там, где это целесообразно) выразительные возможности ветвящихся стрелок (рисунок 35).

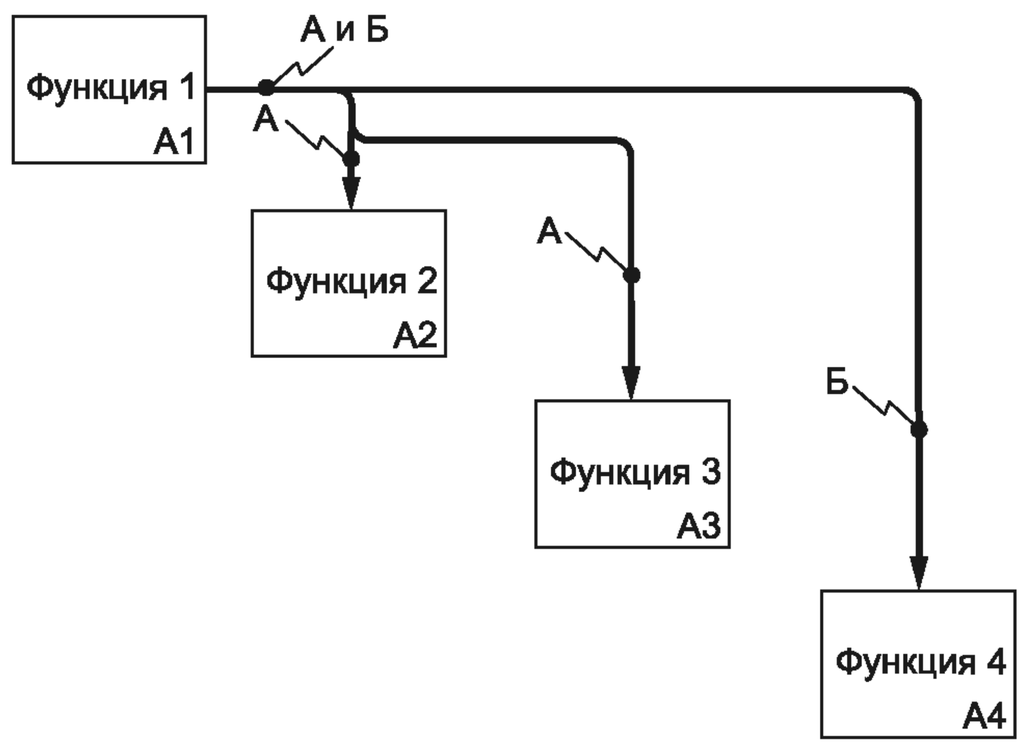


Рисунок 35

21 Каждая диаграмма IDEF0 изображается на стандартном бланке, именуемом мастер-страницей. Бланк снабжен верхним и нижним штампами, содержащими информацию как о конкретной диаграмме, так и в целом о проекте, в состав которого входит диаграмма. Форма бланка с расшифровкой всех полей штампов приведена в приложении А.

10 Ссылочные выражения (коды)

Ссылочные выражения (коды) присваиваются всем элементам модели: диаграммам, блокам, стрелкам и примечаниям. Ссылочные выражения затем могут использоваться в различных контекстах для точного указания на нужный элемент модели.

Основное ссылочное выражение — узловой номер, который появляется там, где выполняется декомпозиция функционального блока и создается его подробное описание на дочерней диаграмме. Все остальные ссылочные коды базируются на узловых номерах.

10.1 Номера блоков

Каждому блоку на диаграмме присваивается номер, помещаемый в нижнем правом внутреннем углу блока. Эта система нумерации необходима для однозначной идентификации блоков в пределах диаграммы и для генерации узловых номеров. Эти номера используются также для ссылок на блоки в тексте и глоссарии.

На контекстной диаграмме А—0 единственному блоку присваивается номер 0 (нуль). На всех других диаграммах блоки нумеруются от 1 до 6, начиная с верхнего левого блока (при их диагональном размещении) и кончая нижним правым блоком. Если некоторые блоки на диаграмме размещены не по диагонали, то сначала нумеруются «диагональные» блоки (также начиная с левого верхнего блока), а затем — «недиагональные» блоки, начиная с нижнего правого против часовой стрелки.

10.2 Узловые номера

Узловой номер базируется на положении блока в иерархии модели. Обычно узловой номер формируется добавлением номера блока к номеру диаграммы, на которой он появляется. Например узловой номер блока 2 на диаграмме А25—А252. Все узловые номера IDEF0 начинаются с заглавной буквы, например «А». Когда родительский блок подробно описывается дочерней диаграммой, узловые номера родительского блока и дочерней диаграммы совпадают.

Контекстные диаграммы и дочерняя диаграмма верхнего уровня — исключения в вышеуказанной схеме узловой нумерации. Каждая модель IDEF0 имеет контекстную диаграмму верхнего уровня — диаграмму А—0. Эта диаграмма содержит единственный «высший блок», который является уникальным родителем всей модели и несет уникальный номер 0 (нуль) и узловой номер А0. Каждая модель IDEF0 должна также иметь по крайней мере одну дочернюю диаграмму, содержащую декомпозицию блока А0 на 3 . . . 6 дочерних блоков. Этим блокам присваиваются уникальные узловые номера А1, А2, А3, . . . , А6. Таким образом, последовательность [А0, А1, . . . , А2, . . . , А3, . . .] начинает нумерацию узлов для любой модели.

Например, модель может иметь следующие узловые номера:

. . .	
А—1	Дополнительная контекстная диаграмма
А—0	Обязательная контекстная диаграмма верхнего уровня (содержащая высший блок А0)
А0	Верхняя дочерняя диаграмма
А1, А2, . . . , А6	Дочерние диаграммы
А11, А12, . . . , А16, . . , А61, . . . , А66	Дочерние диаграммы
А111, А112, . . . , А161, . . . , А611, . . . , А666	Дочерние диаграммы
. . .	Дочерние диаграммы нижнего уровня

Узловой номер используется также для обозначения того, что блок подвергнут декомпозиции. В этом случае узловой номер, совпадающий с номером дочерней диаграммы, помещается под правым нижним углом блока на родительской диаграмме (рисунок 36).

10.3 Перечень узлов

Перечень узлов представляет информацию о входящих в модель узлах в форме списка, напоминающего обычное оглавление и отражающего иерархическую структуру модели, как показано на рисунке 36.

А0 Производить продукт
- А1 Планировать производство
  - А11 Выбрать технологию производства
  - А12 Оценить требуемое время и затраты на производство
  - А13 Разработать производственные планы
  - А14 Разработать план вспомогательных действий
- А2 Разрабатывать и управлять графиком выпуска и ресурсами
  - А21 Разработать основной график
  - А22 Разработать график координации работ
  - А23 Оценивать затраты и приобретать ресурсы
  - А24 Следить за выполнением графика и расходом ресурсов
- А3 Планировать выпуск продукции

Рисунок 36

10.4 Дерево узлов

Разработанная модель IDEF0 со всеми уровнями структурной декомпозицией может быть представлена на единственной диаграмме в виде дерева узлов, дополняющего перечень узлов. Для изображения этого дерева нет стандартного формата. Единственное требование состоит в том, что вся иерархия узлов модели должна быть представлена наглядно и понятно. Пример дерева узлов показан на рисунке 37.

Рисунок 37

11 Методика разработки функциональных моделей в среде IDEF0

В предыдущих разделах описаны инструментальные возможности методологии IDEF0 как средства функционального моделирования производственно-технических и организационно-экономических систем. В настоящем разделе кратко излагаются некоторые методические приемы построения моделей, облегчающие практическое применение этой методологии.

11.1 Общие положения

Как уже отмечалось во введении, объектами функционального моделирования и структурного анализа по методологии IDEF0 являются организационно-экономические и производственно-технические системы.

Функциональный блок, отображающий моделируемую систему в целом (блок A0), и блоки на любом уровне декомпозиции являются преобразующими блоками. **Преобразующий блок** — блок IDEF0-диаграммы, преобразующий входы в выходы под действием управлений при помощи «механизмов» (см. разделы 2, 4). **Преобразование** — цель и результат работы любого блока на диаграмме любого уровня декомпозиции.

Преобразованию в блоке могут подвергаться материальные и информационные объекты, образующие соответствующие потоки.

Материальный поток — непрерывное или дискретное множество материальных объектов, распределенное во времени.

Информационный поток — множество информационных объектов, распределенное во времени.

Информация, участвующая в процессах, операциях, действиях и деятельности в целом, может быть классифицирована на три группы:

- ограничительная;

- описательная;
- предписывающая (управляющая).

Ограничительная информация — сведения о том, **что нельзя делать:**

а) никогда, ни при каких обстоятельствах (кроме, быть может, форс-мажорных), в любой фазе жизненного цикла и на любом этапе функционирования системы в целом;

б) в рамках функционирования конкретного блока.

Ограничительная информация содержится в законах, подзаконных актах, международных, государственных и отраслевых стандартах, а также в специальных внутренних положениях и документах предприятия, в частности, в технических требованиях, условиях, регламентах и т.д.

Описательная информация — сведения об атрибутах объекта (потока), преобразуемого функциональным блоком. Содержится в чертежах, технических и иных описаниях, реквизитах и других документах, являясь неотъемлемым компонентом объекта в течение всего жизненного цикла. Эта информация сама преобразуется (изменяется) в результате выполнения функции.

Предписывающая (*управляющая*) информация — сведения о том, **как, при каких условиях и по каким правилам** следует преобразовать объект (поток) на входе в объект (поток) на выходе блока. Содержится в технологических (в широком смысле) инструкциях, руководствах, документах, определяющих «настройки» и характеристики блока.

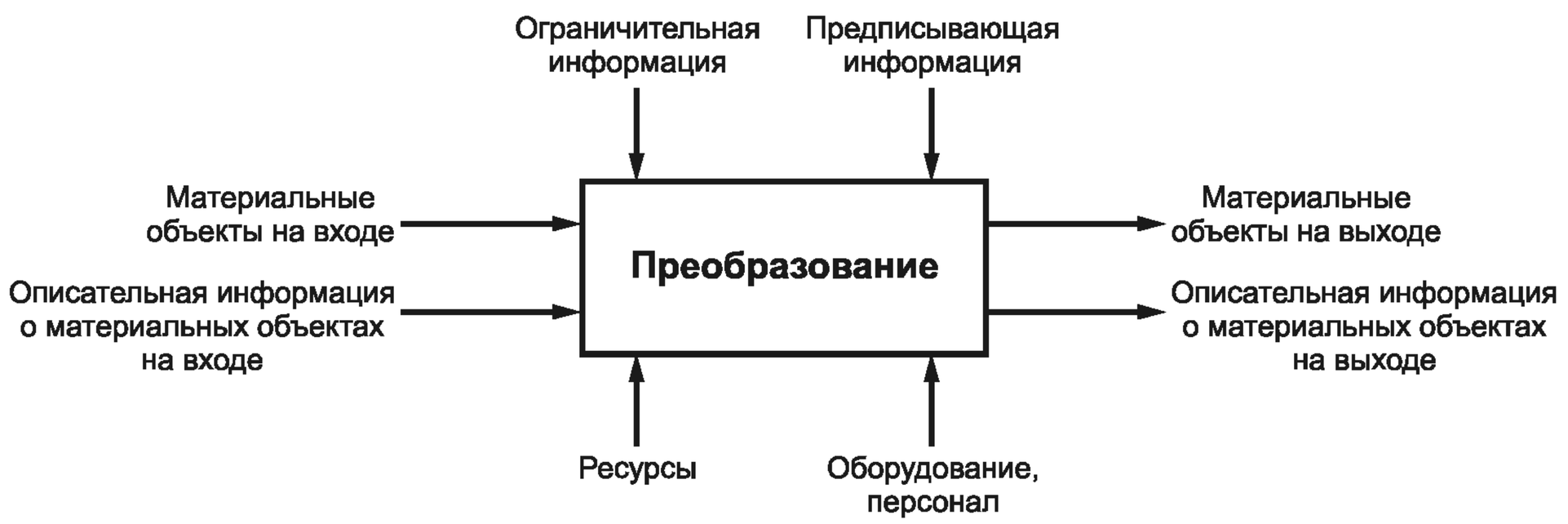


Рисунок 38

Схематическое изображение связей преобразующего блока в соответствии с соглашениями системы IDEF0 показано на рисунке 38. Ограничительная и предписывающая информация изображается стрелками, присоединяемыми к блоку на стороне управления, а описательная информация поступает на вход блока и формируется на его выходе, отображаясь стрелками входа и выхода соответственно.

Материальный поток и описывающий его информационный поток везде, где это не вызывает недоразумений, можно изображать одной стрелкой.

11.2 Классификация функций, моделируемых блоками IDEF0

Практика построения функциональных моделей требует введения классификации явлений и событий, отображаемых в моделях. Такая классификация облегчает выбор глубины декомпозиции моделируемых систем и способствует выработке единообразных подходов и приемов моделирования в конкретных предметных областях.

В настоящих рекомендациях предлагается классификация, ориентированная на достаточно широкий круг организационно-экономических и производственно-технических систем. Классификация делит все функции таких систем на четыре основных и два дополнительных вида. Каждая рубрика в классификации представляет собой класс преобразующих блоков, экземпляры которого возникают и используются при моделировании конкретной системы.

а) Основные виды функций:

1 Деятельность (синонимы: *дело, бизнес*) — совокупность **процессов,** выполняемых (протекающих) последовательно или/и параллельно, преобразующих множество материальных или/и информационных потоков во множество материальных или/и информационных потоков с другими свойствами. **Деятельность** осуществляется в соответствии с заранее определенной и постоянно корректируемой **целью,** с потреблением финансовых, энергетических, трудовых и материальных **ресурсов,** при выполнении **ограничений** со стороны внешней среды.

В модели IDEF0 деятельность описывается блоком А0 на основной контекстной диаграмме А—0.

При моделировании крупных, многопрофильных структур (фирм, организаций, предприятий), которые по своему статусу занимаются различными **видами деятельности,** последние представляют собой **различные экземпляры класса «деятельность»** и могут найти отражение в дополнительной контекстной диаграмме А—1. В этом случае общая модель сложной структуры будет состоять из ряда частных моделей, каждая из которых относится к конкретному виду деятельности.

2 Процесс (синоним: *бизнес-процесс*) — совокупность последовательно или/и параллельно выполняемых **операций,** преобразующая материальный или/и информационный потоки в соответствующие потоки с другими свойствами. **Процесс** протекает в соответствии с управляющими **директивами,** вырабатываемыми на основе **целей деятельности.** В ходе процесса потребляются финансовые, энергетические, трудовые и материальные **ресурсы** и выполняются **ограничения** со стороны других процессов и внешней среды.

3 Операция — совокупность последовательно или/и параллельно выполняемых **действий,** преобразующих объекты, входящие в состав материального или/и информационного потока, в соответствующие объекты с другими свойствами. Операция выполняется: а) в соответствии с **директивами,** вырабатываемыми на основе директив, определяющих протекание процесса, в состав которого входит операция; б) с потреблением всех видов необходимых **ресурсов;** в) с соблюдением ограничений со стороны других операций и внешней среды.

4 Действие — преобразование какого-либо свойства материального или информационного объекта в другое свойство. Действие выполняется в соответствии с **командой,** являющейся частью **директивы** на выполнение операции, с потреблением необходимых ресурсов и с соблюдением ограничений, налагаемых на осуществление операции.

б) Дополнительные виды функций:

5 Субдеятельность — совокупность нескольких процессов в составе деятельности, объединенная некоторой частной целью (являющейся «подцелью» деятельности).

6 Подпроцесс — группа операций в составе процесса, объединенная технологически или организационно.

Понятия группы а) образуют естественную иерархию блоков на IDEF0-диаграммах при декомпозиции, предусматривая четыре уровня последней. При анализе сложных видов деятельности могут потребоваться промежуточные уровни декомпозиции, основанные на применении функций группы б).

Уровни декомпозиции, детализирующие действия, естественно считать состоящими из **элементарных или простых функций.**

В приложении Б приведены IDEF0-диаграммы, показывающие описанную в классификации иерархию функций в виде абстрактной метамодели. Из нее видно, как эти функции взаимодействуют между собой на разных уровнях декомпозиции. Метамодель служит шаблоном, применение которого может облегчить создание реальной модели в конкретной предметной области.

11.3 Организационно-технические структуры и механизмы IDEF0-моделей

Все функции, входящие в приведенную выше классификацию, находятся между собой в отношениях иерархической подчиненности по принципу «сверху вниз»: деятельность — субдеятельность — процесс — подпроцесс — операция — действие. Согласно методологии IDEF0 каждая функция выполняется посредством механизма. В большинстве систем, анализируемых при помощи функциональных моделей, такими механизмами служат организационно-технические структуры. Одним из концептуальных принципов функционального моделирования (см. 4.7) является «отделение «организации» от функций». Вместе с тем анализ показывает что между иерархией функций (преобразований) и иерархией механизмов существует соответствие, иллюстрируемое рисунком 39.

Элементы иерархии механизмов определяются следующим образом.

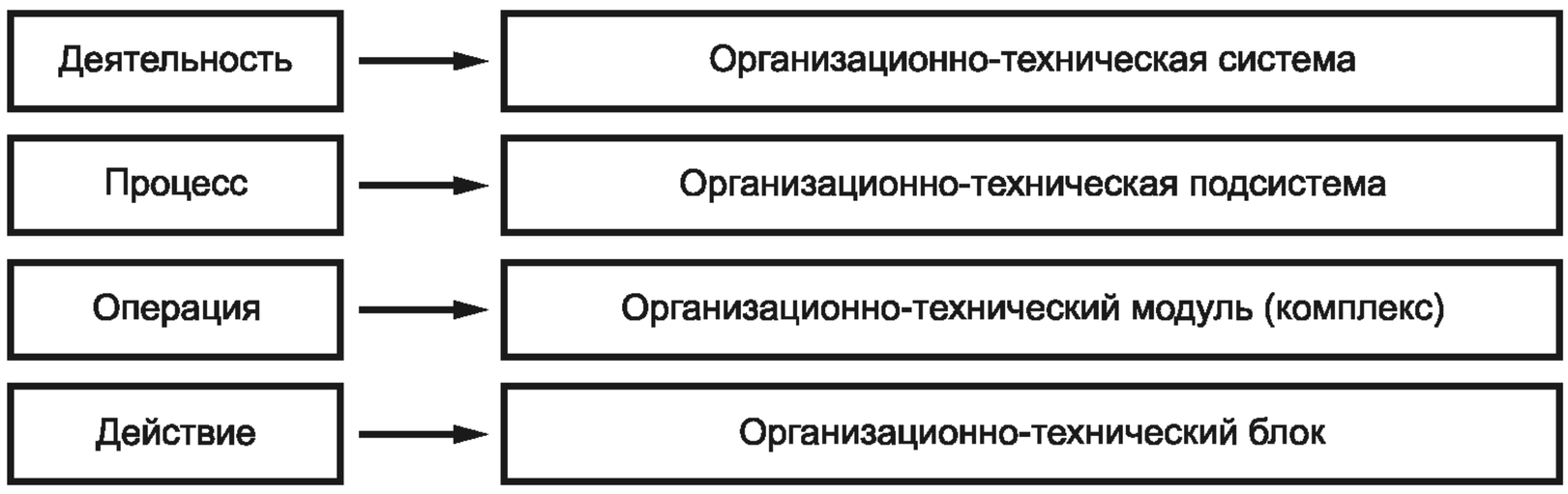


Рисунок 39

Организационно-техническая система — организационная структура, персонал и комплекс технических средств (оборудование), необходимые для осуществления **деятельности.**

Организационно-техническая подсистема — часть организационно-технической системы, обеспечивающая протекание **процесса** (субдеятельности).

Организационно-технический комплекс (модуль) — часть организационно-технической подсистемы, предназначенная для выполнения **операции.**

Организационно-технический блок — часть организационно-технического комплекса, обеспечивающая выполнение **действия.**

Таким образом, при корректном построении модели (без априорной привязки к «организации») появляется возможность связать ее блоки на разных уровнях декомпозиции с объектами организационно-технической структуры, выступающими в качестве механизмов. В этом случае **организационно-техническая структура становится результатом функционального моделирования.**

Э — энергия, П — персонал, О — оборудование, Ф — финансы

Рисунок 40

Во многих моделях находит или должно находить отражение явление, состоящее в формировании или специфической настройке (перестройке) механизмов в ходе деятельности. Это явление часто именуется **реинжинирингом** производства и/или бизнес-процессов на предприятии (в организации).

Явление отражается в модели как субдеятельность, поскольку почти всегда состоит из нескольких процессов. Укрупненная схема этой субдеятельности приведена на рисунке 40. Согласно схеме **входом** и одновременно потребляемым **ресурсом** субдеятельности являются **финансы,** преобра- зуемые в другие виды ресурсов — энергетические, трудовые, материальные (оборудование, вспомогательные материалы и т.д.). (см. приложение Б).

Механизм любого уровня обеспечивает выполнение деятельности (процесса, операции, действия), потребляя ресурсы: финансовые, энергетические, трудовые, непосредственно или с помощью промежуточных преобразований (рисунок 40), то есть специфических процессов, которые можно назвать поддерживающими, обеспечивающими или **вспомогательными** (по аналогии с вспомогательными производствами, цехами, участками на машиностроительном предприятии) по отношению к **основным процессам,** где происходят преобразования, однозначно обусловленные целью деятельности.

Существенный признак вспомогательного процесса: этот процесс не создает конечного продукта деятельности и, следовательно, прибыли.

11.4 Управление — особый вид процесса, операции, действия

Один из общих принципов методологии IDEF0 требует, чтобы к каждому блоку на диаграмме была присоединена хотя бы одна управляющая стрелка, отображающая условия правильного функционирования блока (см. раздел 9). Ниже сформулирован ряд определений и методических положений, которыми следует руководствоваться при отражении управлений на функциональных моделях.

Управление деятельностью — процесс, состоящий как минимум из следующих **операций:**

- формулирование целей деятельности;
- оценивание ресурсов, необходимых для осуществления деятельности и их сопоставление с имеющимися ресурсами;
- сбор информации об условиях протекания и фактическом состоянии деятельности («глобальная» обратная связь);
- выработка и принятие решений, направленных на достижение целей, в частности, решений о распределении ресурсов по процессам, входящим в состав деятельности; оформление решений в виде директив на управление процессами;
- реализация решений (исполнение директив) и оценка их результатов («локальная обратная связь»);
- корректировка (в случае необходимости, например при нехватке ресурсов) ранее сформулированных целей (самонастройка, адаптация).

Управление процессом — операция, состоящая как минимум из следующих **действий:**

- анализ директивы на управление процессом, ее декомпозиция на директивы управления операциями;
- сбор (прием по каналам связи) информации о ходе выполнения операций, ее обобщение и формирование сведений о состоянии процесса; передача данных в подсистему управления деятельностью;
- сопоставление информации о ходе операций с данными директив и выработка локальных решений, направленных на устранение отклонений;
- корректировка (в случае необходимости) директив на выполнение операций.

Управление операцией — действие, состоящее в выработке на основании директивы на управление операцией **команд** на управление действиями, в реализации этих команд, оценке результатов выполнения, передаче необходимой информации в комплекс управления процессом, корректировке команд в случае необходимости.

Блоки управления должны присутствовать на каждой IDEF0-диаграмме (кроме тех, которые являются декомпозициями самих таких блоков). Через них осуществляются управляющие воздействия на остальные блоки диаграммы. Именно эти блоки воспринимают ограничивающую и предписывающую информацию и преобразуют ее в соответствующие директивы и команды. Имена блоков управления, как правило, содержат глагол «Управлять . . . ».

Стрелки, исходящие из блока с именем «Управлять . . . », описывают **централизованную схему** управления (управленческую «вертикаль»). Возможны варианты структур, в которых выходная информация одного из блоков является управляющей для другого. Это отображает **децентрализацию** управления («горизонтальные» связи) (см. приложение Б).

11.5 Типизация функциональных моделей и IDEF0-диаграмм

Эффективность и производительность труда разработчиков функциональных моделей могут быть повышены за счет применения типовых моделей и отдельных диаграмм, ориентированных на применение в конкретных предметных областях. Так, например, на основе представлений о жизненном цикле продукции (изделия) можно предложить типовую диаграмму уровня А0 для промышленного предприятия, которая может иметь вид, схематически показанный на рисунке 41.

Фрагмент типовой модели промышленного предприятия в формате IDEF0 дан в приложении В.

Аналогичные типовые модели могут быть разработаны для других видов бизнеса (оказание услуг, транспорт, банковское дело, финансовая деятельность и т.д.).

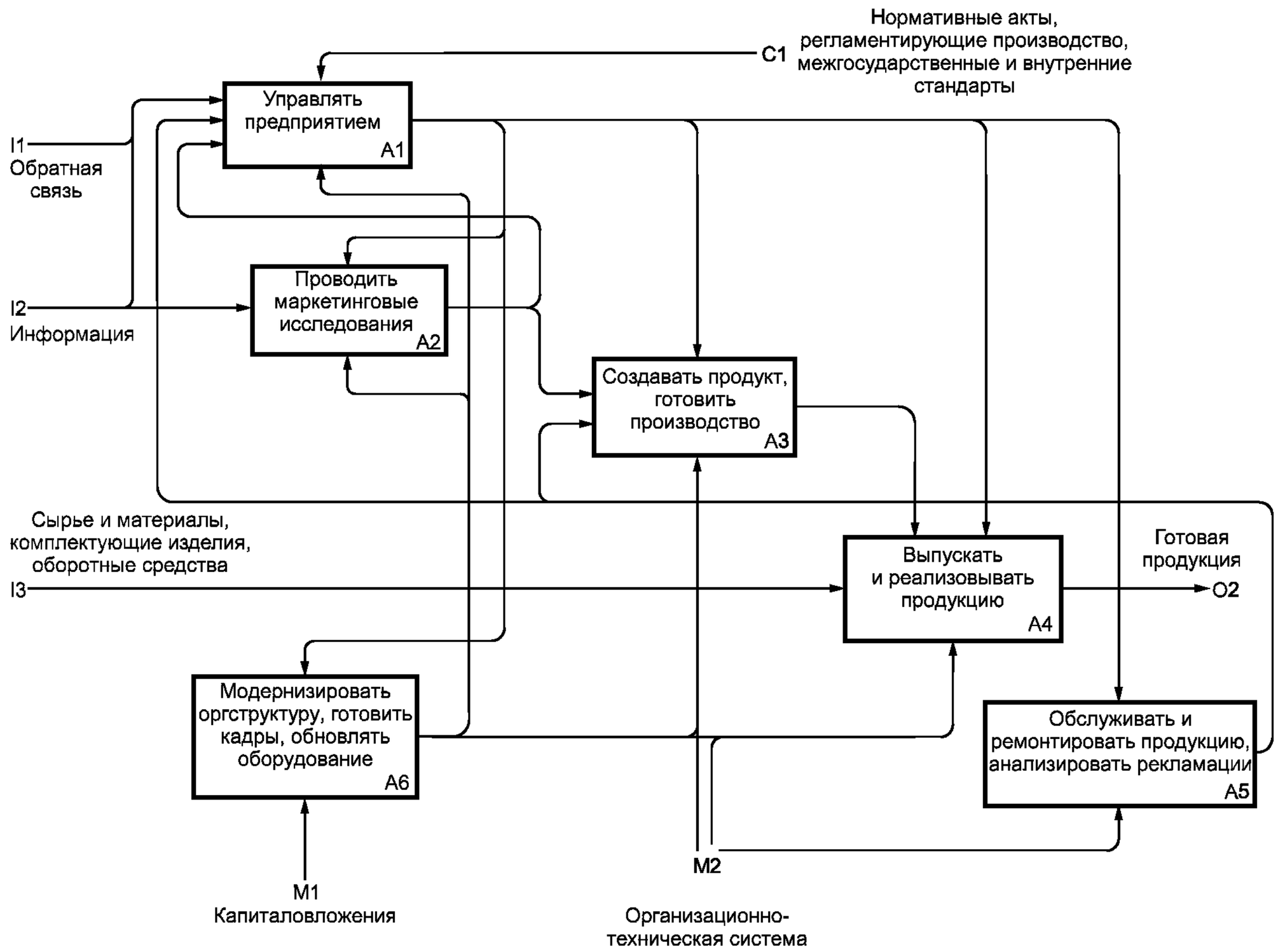


Рисунок 41

12 Организация процесса функционального моделирования и управление проектом

12.1 Общие положения

Для эффективного моделирования и получения результатов в соответствии со сроками и сметами управление проектом должно представлять собой процесс, в ходе которого координируется работа авторов, экспертов и тех, кто принимает окончательную версию модели системы или ее части.

Это должен быть процесс, в полной мере использующий возможности методологии, основанной на разделении функций участников проекта и итеративном характере рецензирования, в ходе которого проверяется корректность диаграмм и/или моделей, а также соответствие их поставленной цели и точке зрения.

IDEF0-модель есть результат скоординированной коллективной работы, при которой авторы создают первоначальные диаграммы, основанные на собранной информации об объекте моделирования, и передают их другим участникам проекта для рассмотрения и формулирования замечаний. Порядок, изложенный ниже, требует, чтобы каждый эксперт, у которого есть замечания к диаграмме, сделал их письменно и передал автору диаграммы. Этот цикл продолжается до тех пор, пока диаграммы, а затем и вся модель не будут приняты. Процесс моделирования иллюстрируется рисунком 42. Диаграмма отражает тот факт, что этот процесс — итеративная процедура, приводящая к точному описанию системы.

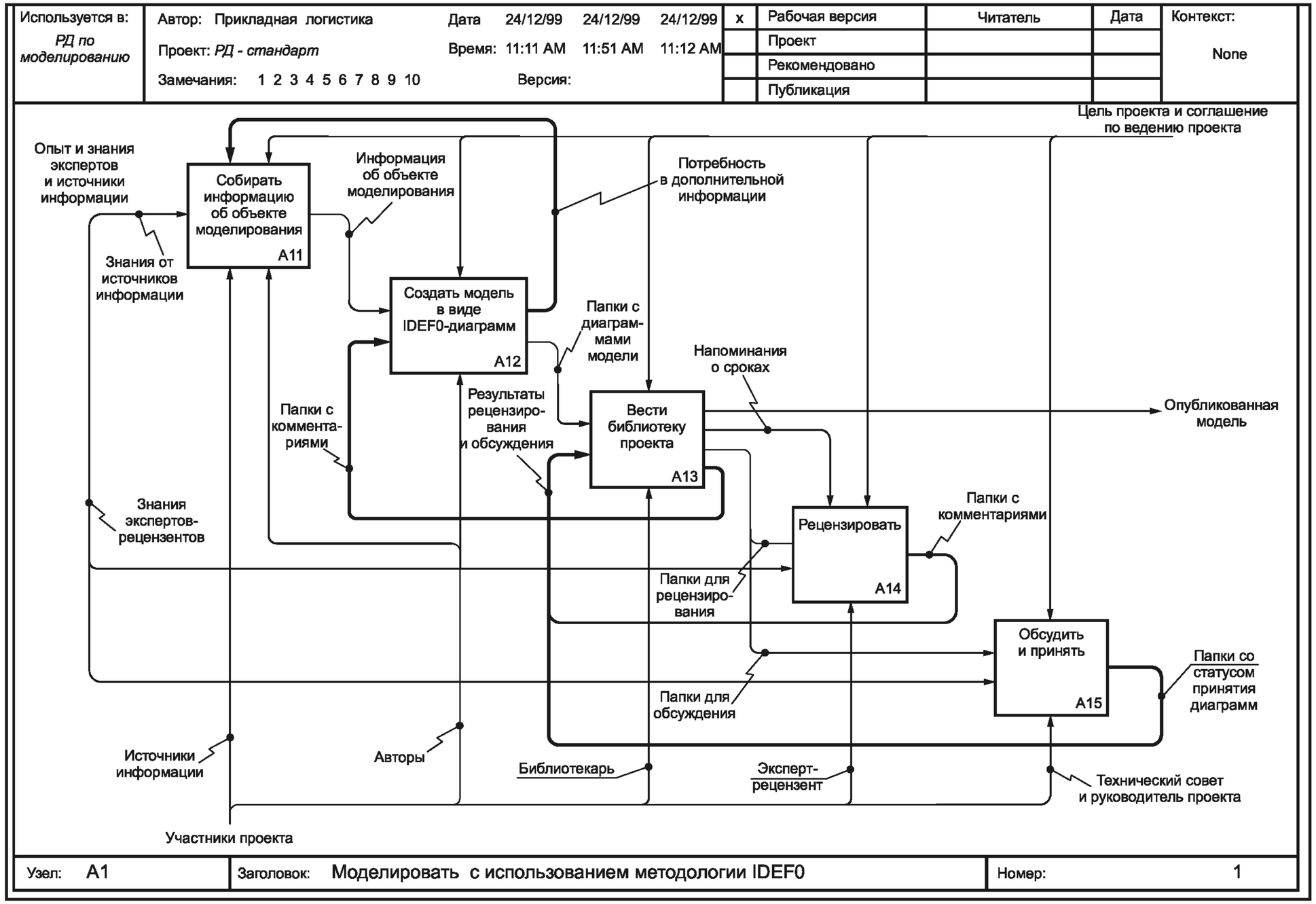

Используется в:
РД по моделированию
Автор: Прикладная логистика
Проект: РД - стандарт
Замечания: 1 2 3 4 5 6 7 8 9 10
Дата 24/12/99 24/12/99 24/12/99
Время: 11:11 AM 11:51 AM 11:12 AM
Версия:
x
Рабочая версия
Проект
Рекомендовано
Публикация
Читатель
Дата
Контекст:
None
Цель проекта и соглашение по ведению проекта
Опыт и знания экспертов и источники информации
Собирать информацию об объекте моделирования
A11
Информация об объекте моделирования
Потребность в дополнительной информации
Знания от источников информации
Создать модель в виде IDEF0-диаграмм
A12
Папки с диаграм-мами модели
Напоминания о сроках
Результаты рецензиро-вания и обсуждения
Папки с коммента-риями
Вести библиотеку проекта
A13
Опубликованная модель
Знания экспертов-рецензентов
Рецензировать
A14
Папки с комментариями
Папки для рецензиро-вания
Обсудить и принять
A15
Папки со статусом принятия диаграмм
Папки для обсуждения
Источники информации
Авторы
Библиотекарь
Эксперт-рецензент
Технический совет и руководитель проекта
Участники проекта
Узел: A1
Заголовок: Моделировать с использованием методологии IDEF0
Номер: 1

Рисунок 42

Ценность модели (проекта) определяется ее приемлемостью для экспертов.

Эта приемлемость достигается следующими путями:

1) постоянным рецензированием экспертами развивающейся модели, что обеспечивает необходимый уровень соответствия — адекватности — модели существующему моделируемому объекту (если модель отражает состояние «как есть») или предполагаемому (состояние «как должно быть») в том понимании, которое соответствует мнению экспертов;

2) периодическим обсуждением диаграмм, частей модели и модели в целом на техническом совете, решение которого (оформленное в виде протокола) позволяет автору продолжить уточняющее моделирование или закончить его ввиду достаточности детализации и приемлемости проекта (модели).

Если в процессе моделирования выявляется несогласованность оценок экспертов, то она должна быть преодолена, чтобы получить модель, адекватно представляющую объект моделирования или какую-то его часть.

Методология IDEF0 предусматривает необходимость сохранения записей обо всех решениях и альтернативных подходах по мере того, как они возникают на протяжении проекта.

Копии диаграмм, разработанные автором, критически (конструктивно) анализируются компетентными экспертами, которые заносят свои замечания и предложения непосредственно на копиях диаграмм. Авторы отвечают на каждое замечание письменно на тех же копиях.

Предложения принимаются или отвергаются письменно с указанием причин. После внесения изменений и исправлений старые варианты диаграмм остаются в архиве проекта.

В процессе чтения диаграмм ничто не должно предполагаться в модели по умолчанию, а также не должны делаться выводы, выходящие за пределы действия и утверждения модели. Это побуждает автора к тщательному комментированию и иллюстрированию каждого добавляемого к модели фрагмента, чтобы при чтении и интерпретации модели ее толкование было однозначным и соответствующим поставленной цели и установленной точке зрения без личного присутствия автора и его дополнительных пояснений.

12.2 Состав участников проекта и структура их взаимодействия

В коллектив, занимающийся проектированием (моделированием), должны входить следующие участники:

- руководитель проекта;
- авторы (разработчики) модели;
- технический совет;
- эксперты в предметной области;
- библиотекарь.

Дополнительный специфический участник проекта — «Источники информации».

При проведении работ с привлечением сторонних организаций может создаваться технический совет, обеспечивающий взаимодействие всех участников проекта, работающих как в составе проектирующей организации, так и вне ее. Выполняемая функция («роль», которую выполняет участник проекта) не зависит от должности. Один и тот же человек может выполнять несколько функций. Однако «роль» каждого участника проекта индивидуальна, должна быть определена и зависит от рассматриваемой части проекта. Структура взаимодействия участников проекта приведена на рисунке 43.

Принципы коллективной работы в IDEF0-методологии гарантируют, что окончательная версия IDEF0-модели будет верной, так как модель корректируется по результатам рецензирования частей модели, оформленных в виде папок. Более подробная детализация достигается построением необходимого количества диаграмм. По новым частям модели делаются новые замечания, вносятся новые изменения. Окончательная модель соответствует представлениям автора и экспертов о системе, смоделированной с данной точки зрения и для данной цели.

Руководитель проекта и разработчики модели (авторы) должны быть главными исполнителями. Хотя конечной целью разработчика является получение одобрения модели техническим советом, утверждает результаты руководитель проекта. Таким образом, обеспечивается согласованность интересов авторов, рецензентов, совета и руководителя проекта.

12.2.1 Руководитель проекта

Руководитель проекта — лицо, осуществляющее административное управление проектом. Руководитель проекта должен выполнять следующие основные функции:

- выбирать разработчиков модели (авторов). Главным аспектом данной функции является достижение руководителем проекта и разработчиком модели согласия относительно основополага-

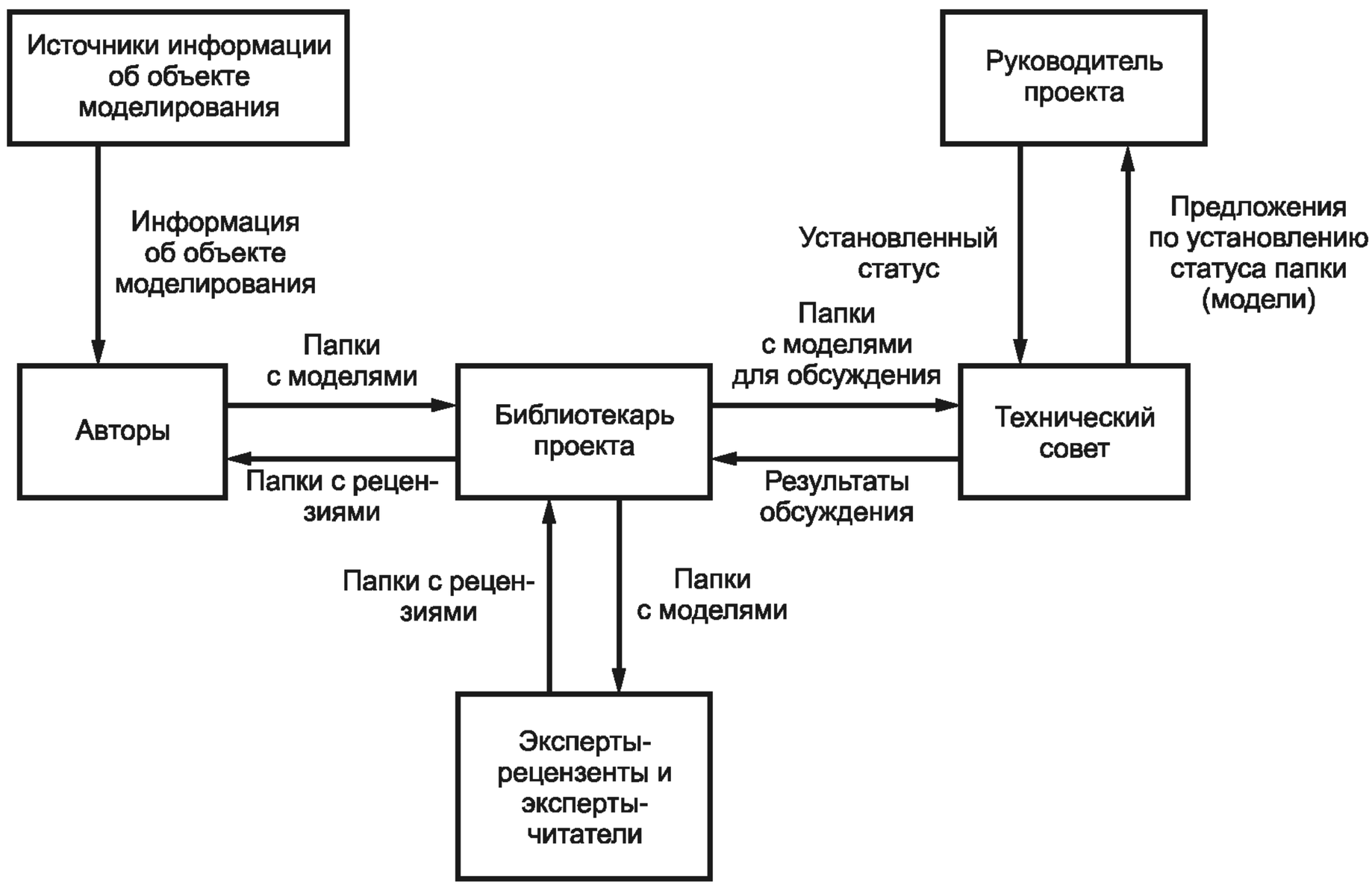


Рисунок 43

ющих правил, в соответствии с которыми выполняется моделирование. Сюда входят использование методологии, порядок и степень контроля за разработчиком модели со стороны руководителя проекта, область действия и ориентация разрабатываемой модели;

- определять обязательные источники информации, на которые разработчик модели будет опираться при построении модели. Этими источниками могут быть либо люди, осведомленные о различных аспектах рассматриваемой сферы деятельности, либо документы, освещающие предметную область моделирования. Эти источники определяются на начальной стадии моделирования, но список их в процессе моделирования должен пересматриваться и уточняться, поскольку по мере построения модели потребности в информации меняются;
- выбирать экспертов, чьи знания будут использованы разработчиком для получения оценки (одобрения) диаграмм и частей модели. Список экспертов составляется на начальной стадии проекта и уточняется о мере необходимости;
- формировать технический совет. Руководитель проекта по должности является председателем технического совета. Этот совет под его председательством периодически собирается для обсуждения существенных вопросов, рецензирования и определения статуса модели и ее частей;
- присваивать статус рассматриваемой советом части модели.

12.2.2 Разработчики (авторы) модели

Разработчики (авторы) модели — лица, создающие IDEF0-модели. Разработчик создает модель на основе материала, собранного из источников информации.

Разработчик должен:

- собирать исходные данные от обязательных источников информации, определенных руководителем проекта; при недостаточности собранных сведений автор вправе использовать любые другие источники информации с обязательным указанием ссылок на них;
- обучать (при необходимости) основам IDEF0-моделирования руководителя проекта, экспертов (рецензентов и читателей) и других членов технического совета для обеспечения правильного понимания ими моделей, создаваемых авторами;
- оформлять модель в виде IDEF0-диаграмм;
- организовать разработку модели.

В период подготовки проекта разработчик вместе с руководителем проекта изучает и устанавливает область действия модели. Затем он намечает план проекта, то есть состав и последовательность работ, которые необходимо выполнить для достижения поставленных целей.

Руководитель проекта обеспечивает разработчика списком источников информации и списком

экспертов, к которым разработчик может обратиться. Разработчик должен удостовериться, что со всеми участниками проекта установлен необходимый контакт.

Исходную информацию разработчик собирает из источников, установленных руководителем проекта. Природа этой информации во многом зависит от стадии разработки модели. Источниками информации могут служить люди и документы. Разработчик должен понимать, что каждый эксперт-источник информации смотрит на информацию со своей точки зрения. Разработчик должен стараться увидеть смысл и структуру информации глазами ее источника. Синтезируя эти точки зрения в процессе сравнения и противопоставления, разработчик создает адекватный образ объекта моделирования.

Второй функцией разработчика является помощь в технике моделирования всем, кому она может понадобиться. Эта помощь заключается в общем ориентировании членов технического совета, источников информации и экспертов, ознакомлении их с навыками чтения модели, а также с навыками моделирования.

Третьей функцией является оформление модели в виде IDEF0-диаграмм. Для рецензирования разработчик оформляет папки с диаграммами для передачи их в библиотеку проекта.

Разработчик организует построение модели. Для поддержки принятых разработчиком решений и регистрации вклада каждого участника записи исходной информации, собранной в процессе моделирования, сохраняются в течение определенного времени после завершения проекта. Это позволяет разработчику следить за тем, чтобы исследуемая область была охвачена со всех сторон. Зная, кто и в каких областях поставлял информацию и как это происходило, разработчик может оценить степень соответствия стадии моделирования исходным целям.

12.2.3 Технический совет

Этот элемент организации процесса создания моделей предлагает арбитражные решения по моделированию и рекомендации по установлению статуса диаграмм, части и/или модели в целом (статусы: «Рабочий проект», «Эскиз», «Рекомендовано» и «Публикация»).

Технический совет проводит политику проекта через рекомендации и замечания авторам, предложения по становлению статуса руководителю проекта и подготовку компромиссных решений в конфликтных ситуациях, которые могут возникнуть в процессе проекта. В техническом совете должно быть несколько специалистов с высоким уровнем компетентности, способных отстоять свои решения перед высшим руководством объекта моделирования.

Технический совет формируется из экспертов и профессионалов, знакомых с предметной областью моделирования. Руководитель проекта формирует этот совет и является его председателем. Поскольку основной причиной, побуждающей создавать модель, является необходимость повышения эффективности объекта моделирования, важно, чтобы в совете были представлены все службы, имеющие отношение к рассматриваемой в проекте предметной области. Полезно включать в совет экспертов из смежных подразделений объекта моделирования, не входящих в исследуемую область, но связанных с ней. Эти эксперты помогают адекватно оценить влияние окружающей среды на объект моделирования.

Эксперты могут быть членами совета. Эксперту обычно показывают только ограниченные фрагменты модели на промежуточных стадиях, в то время как совет должен принять решение по всей модели. Иногда в совет могут входить лица, играющие роль источников. В силу очевидной противоречивости интересов нецелесообразно, чтобы в совет входили разработчики модели.

12.2.4 Эксперт

Эксперт — выбираемое руководителем проекта лицо, обладающее специальными знаниями некоторых аспектов моделируемой области. Его опыт в предметной области, к которой относится моделируемый объект, позволяет делать полезные критические замечания в процессе создания модели.

Эксперты призваны критически оценивать создаваемую по частям модель. Это осуществляется в ходе нескольких циклов изучения с использованием читательских папок (цикл автор/читатель). Папки обеспечивают эксперта набором информации, предназначенным для описания законченного фрагмента моделируемого объекта. С помощью папок эксперту предоставляется информация в наглядном виде. В процессе рецензирования ему может понадобиться заполнить пробелы или даже завершить изложение материала, представленного в папке. Хотя папка во многом основывается на интерпретации разработчиком ранее полученной информации, комментарии экспертов служат ценным материалом для уточнения модели. В информационных папках перед экспертом должны ставиться конкретные, четко сформулированные вопросы, связанные с моделированием.

Главной задачей эксперта является оценка адекватности модели соответствующей предметной

области. Экспертная оценка является основным средством в достижении консенсуса среди изучающих модель экспертов. Одобренная модель — это модель, согласованная с экспертами. Если эксперты согласны с тем, что модель или ее часть адекватно представляет рассматриваемый объект, то модель считается одобренной. Если есть не согласившиеся с этим, то их мнение должно обязательно фиксироваться, и модель считается неправильной, пока не доказано обратное. Для достижения консенсуса авторы учитывают комментарии и замечания экспертов при пересмотре той части модели, к которой эти замечания относятся.

Эксперты подразделяются на две группы:

- эксперты-рецензенты;
- эксперты-читатели.

Эксперт-рецензент — член коллектива разработчиков, знающий предметную область моделирования, специализирующийся на некоторой конкретной функции предприятия и ответственный за обеспечение критических комментариев относительно разрабатываемой модели. Эксперт-рецензент должен знать IDEF0-методологию и уметь делать письменные структурированные замечания в рассылаемых папках. Он является постоянным и активным участником цикла автор/читатель.

Эксперт-читатель — член коллектива разработчиков, профессионально знающий предметную область моделирования, понимающий IDEF0-методологию и умеющий читать IDEF0-диаграммы. Эксперт-читатель знакомится с документацией (IDEF0-папкой), не делая письменных комментариев. От экспертов-читателей авторы получают замечания с помощью опроса.

12.2.5 Библиотекарь

Библиотекарь — лицо, ответственное за хранение документации, изготовление копий, координацию обмена письменной и/или электронной информацией (рассылка папок, получение рецензий, регистрация и публикация диаграмм и модели).

12.2.6 Источники информации

Исходная информация для IDEF0-модели поступает к разработчику из разных источников: от людей и от документов. Люди, являющиеся источниками информации, обладают конкретными знаниями о частных свойствах объекта моделирования, управлении или ходе бизнес-процесса, и их участие в моделировании может быть ограничено несколькими минутами опроса. Однако именно эти источники обеспечивают основу для моделирования. Информация, предоставляемая ими, используется для создания модели, а восприятие этой информации обеспечивает разработчику понимание, необходимое для построения адекватной модели.

Руководитель проекта подбирает источники информации, исходя из направления проекта и потребностей разработчика. По мере развития процесса моделирования потребности в информации изменяются, и список источников информации руководитель проекта пересматривает. Собираемая разработчиком информация должна как можно точнее фиксироваться.

Каждый источник воспринимает предметную область по-своему, и на разработчике лежит ответственность за правильный отбор информации. Особенно это относится к источникам-документам.

Источники-документы отражают состояние объекта моделирования в некоторый момент времени, и их роль крайне важна, но для их эффективного использования необходима значительная работа, связанная с интерпретацией, пониманием и подтверждением содержащихся в них сведений.

Лица, выступающие в роли источников информации, могут оказывать разработчику модели дополнительную помощь, объясняя, как сообщенная ими информация поступает, интерпретируется или используется. Разработчик должен воспользоваться этой помощью для понимания того, как восприятие информации одного источника связано с восприятием другого.

12.3 Заключительные замечания

1 Функциональная модель — плод коллективного труда всех участников процесса моделирования.

2 Создание моделей, адекватно отражающих предметную область, возможно лишь при выполнении обязательных условий:

- IDEF0-диаграммы следует разрабатывать в точном соответствии с IDEF0-методологией;
- при моделировании должен быть организован итеративный процесс рецензирования каждого фрагмента модели и модели в целом;
- начинать следующий уровень декомпозиции можно лишь после полного завершения работы над родительской диаграммой, то есть после присвоения ей статуса «Публикация».

ПРИЛОЖЕНИЕ А
(обязательное)

Стандартный бланк методологии IDEF0 и правила его заполнения

А.1 Мастер-страница (Master Page)

Мастер-страница содержит шаблон (бланк), который копируется на каждую страницу модели (рисунок А.1). Мастер-страница содержит три области.

1 Область проектной информации — в верхней части страницы.

2 Рабочая область, в которой располагается диаграмма или другая информация — в центральной части страницы.

3 Область идентификации — вдоль нижнего края страницы.

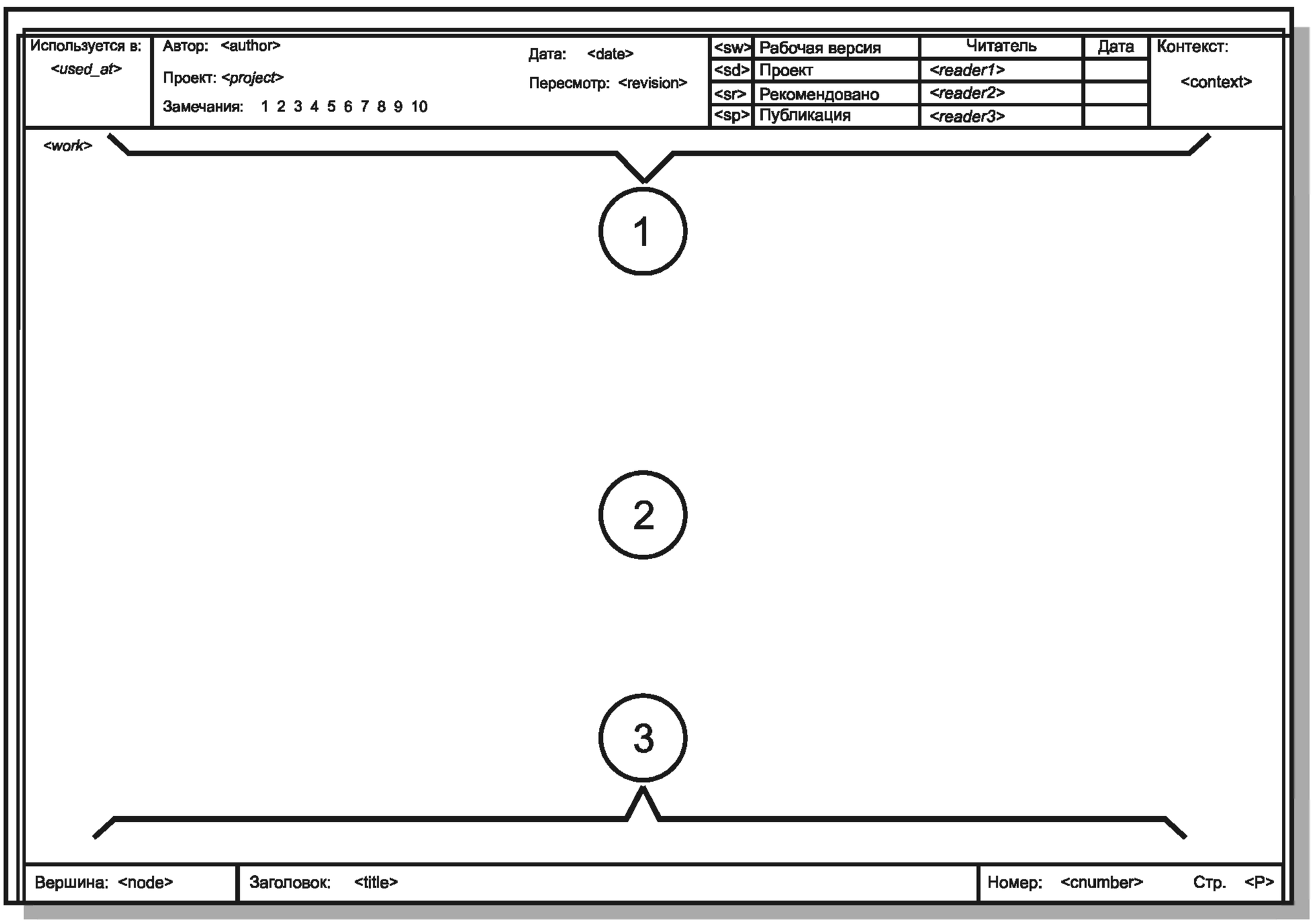


Рисунок А.1 — Мастер-страница

А.2 Стандартный бланк

Стандартный бланк диаграммы (рисунок А.2) формируется на базе мастер-страницы и обеспечивает единообразное оформление и хранение всех документов, относящихся к модели. Бланк имеет формат А4.

Стандартный бланк содержит следующие информационные поля.

Рабочее поле

Поле содержит всю информацию, относящуюся к диаграмме: изображения блоков и стрелок, метки, примечания и т.д. Оно служит также для размещения глоссария, текстовых комментариев, замечаний, иллюстраций, эскизов и т.д. Автор не должен использовать никаких других бумаг, кроме стандартного бланка.

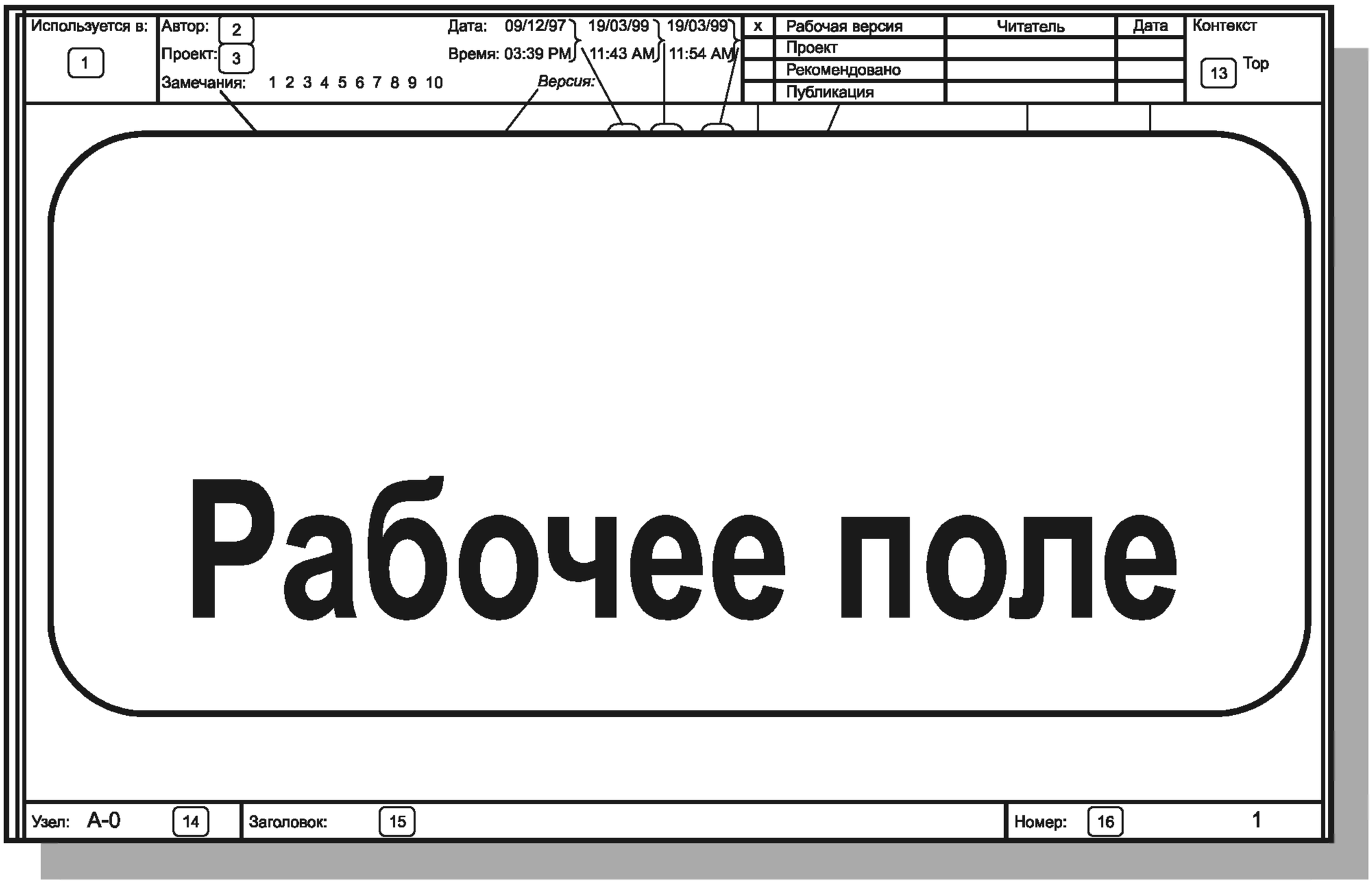


Рисунок А.2 — Стандартный бланк

Область проектной информации

1 Поле «Используется в:»

Содержит название общей модели, составной частью которой является настоящая модель, или область ее применения.

2 Поле «Автор»

В этом поле указывают имя автора диаграммы или наименование организации—разработчика проекта.

3 Поле «Проект»

Содержит название проекта или его аббревиатуру.

4 Поле «Замечания»

Используется для указания количества замечаний на текущей странице. Чтобы указать количество замечаний, соответствующее число перечеркивают крестом (символом «Х») или обводят это число эллипсом.

5 Поле «Версия»

Используется для указания информации о версии диаграммы.

6 Дата и время создания страницы.

7 Дата и время внесения последнего изменения на странице.

8 Текущая дата и время.

9 Поля для установления статуса страницы.

В зависимости от присвоенного статуса (см. ниже) в соответствующем поле ставится символ «Х».

10 Наименования устанавливаемых статусов.

Наименования статуса, присваиваемого диаграмме по мере ее продвижения по циклам «автор — читатель». Приняты следующие виды статусов диаграмм в зависимости от уровня их готовности.

«Рабочая версия» — диаграмма с большим числом изменений. Новые диаграммы являются рабочими версиями.

«Проект» — диаграмма имеет меньше изменений по сравнению с предыдущим уровнем, что свидетельствует о достижении некоторого согласия с экспертами. Статус «Проект» присваивается диаграмме по предложению руководителя работ, до ее рассмотрения экспертами и одобрения техническим советом.

«Рекомендовано» — диаграмма и сопутствующие ей тексты прорецензированы и утверждены техническим советом. В нее не предполагается внесение существенных изменений.

«Публикация» — материал, который может быть представлен заказчику модели и рассылаться для окончательной печати и опубликования.

11 Поля для подписей читателей (экспертов) диаграммы.

12 Дата прочтения диаграммы экспертом.

13 Поле «Контекст»

Содержит миниатюрную картинку страницы диаграммы (без стрелок), содержащей родительский блок, который на этой картинке затемнен. Это позволяет иметь перед глазами структуру родительской диаграммы и место на ней декомпозируемого блока, не вызывая диаграмму на экран.

Область идентификации

14 Поле «Узел»

В поле помещается номер родительского блока для этой страницы диаграммы.

15 Поле «Заголовок»

В поле помещается текст имени родительского блока, в результате декомпозиции которого создана текущая страница.

16 Поле «Номер»

Поле содержит номер данной страницы, на который имеется ссылка на родительской диаграмме (под декомпозируемым блоком). Это может быть С-номер или номер страницы (см. раздел 6).

ПРИЛОЖЕНИЕ Б
(справочное)

Метамодель

На рисунках Б.1 — Б.5 представлены диаграммы метамодели (см. раздел 11).

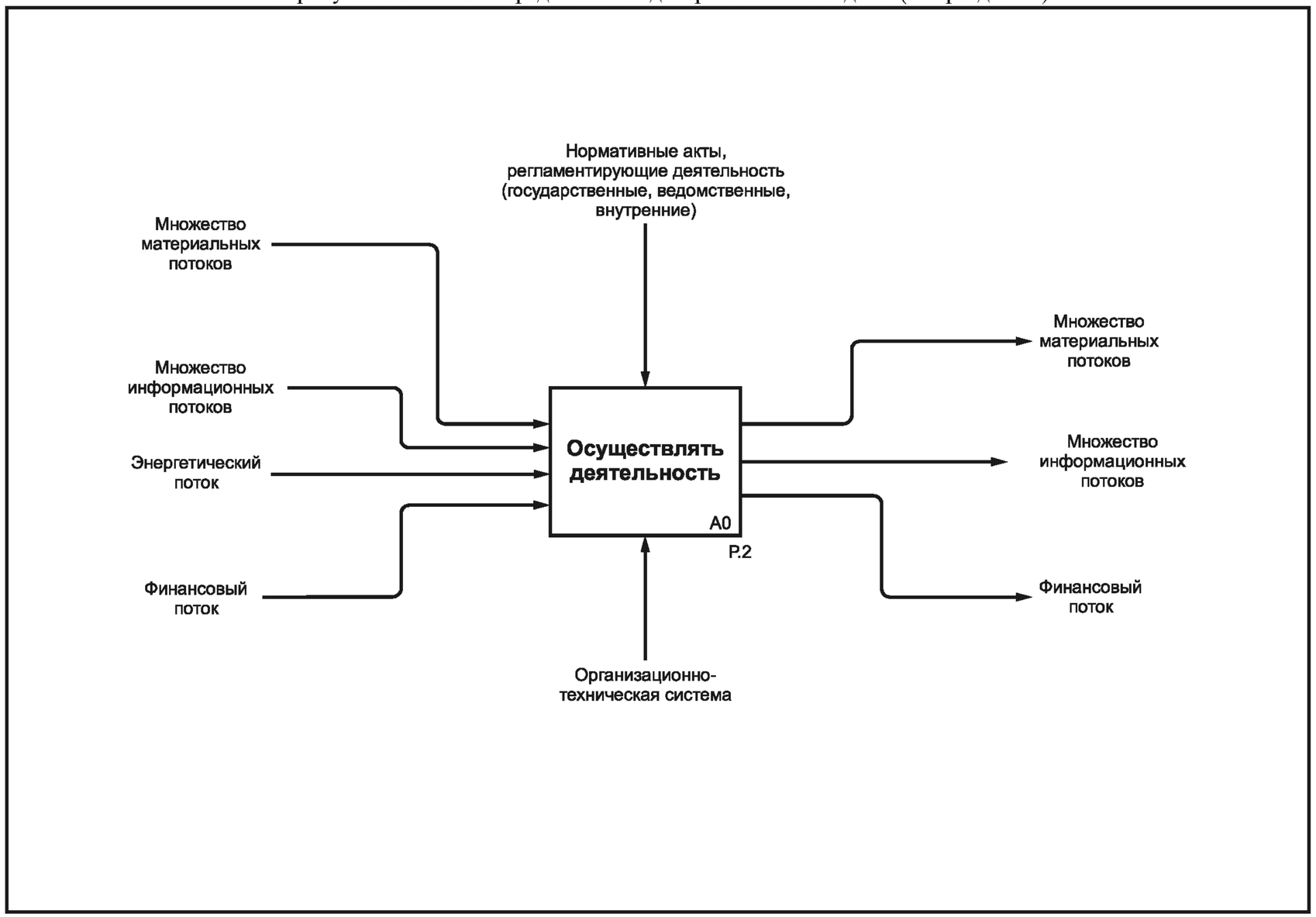


Рисунок Б.1 — Контекстная диаграмма

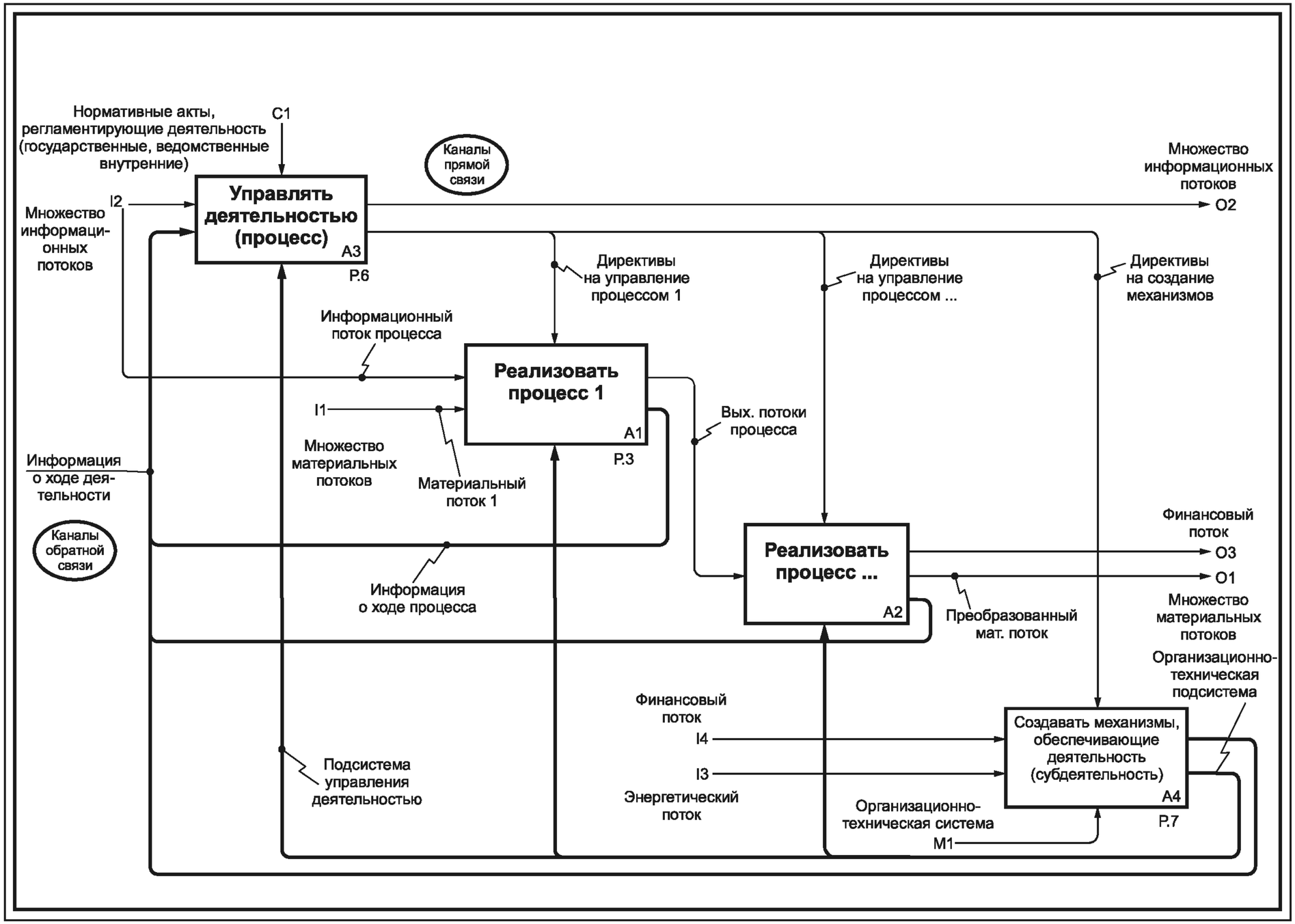


Рисунок Б.2 — Декомпозиция контекстной диаграммы

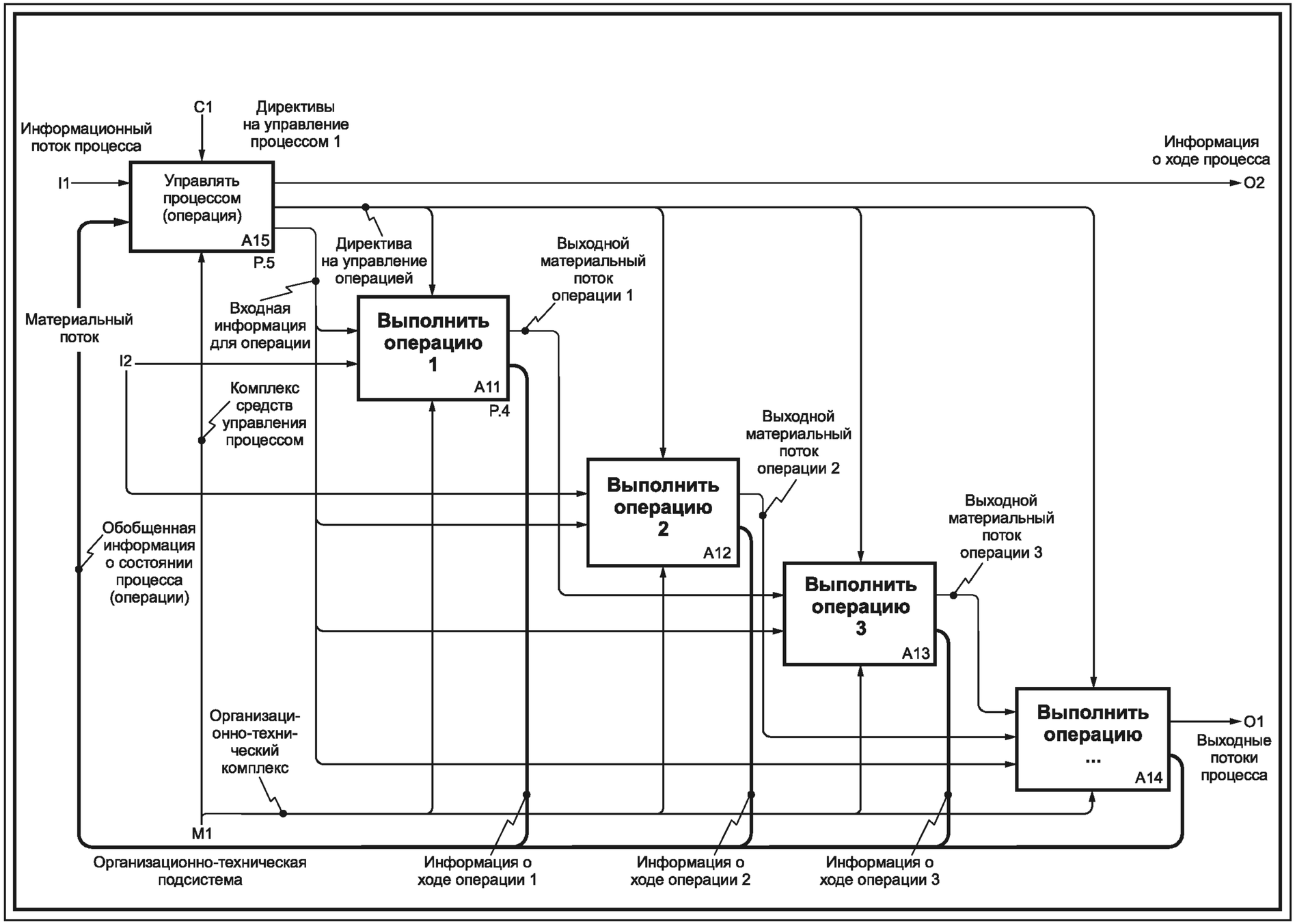


Рисунок Б.3 — Декомпозиция блока А1

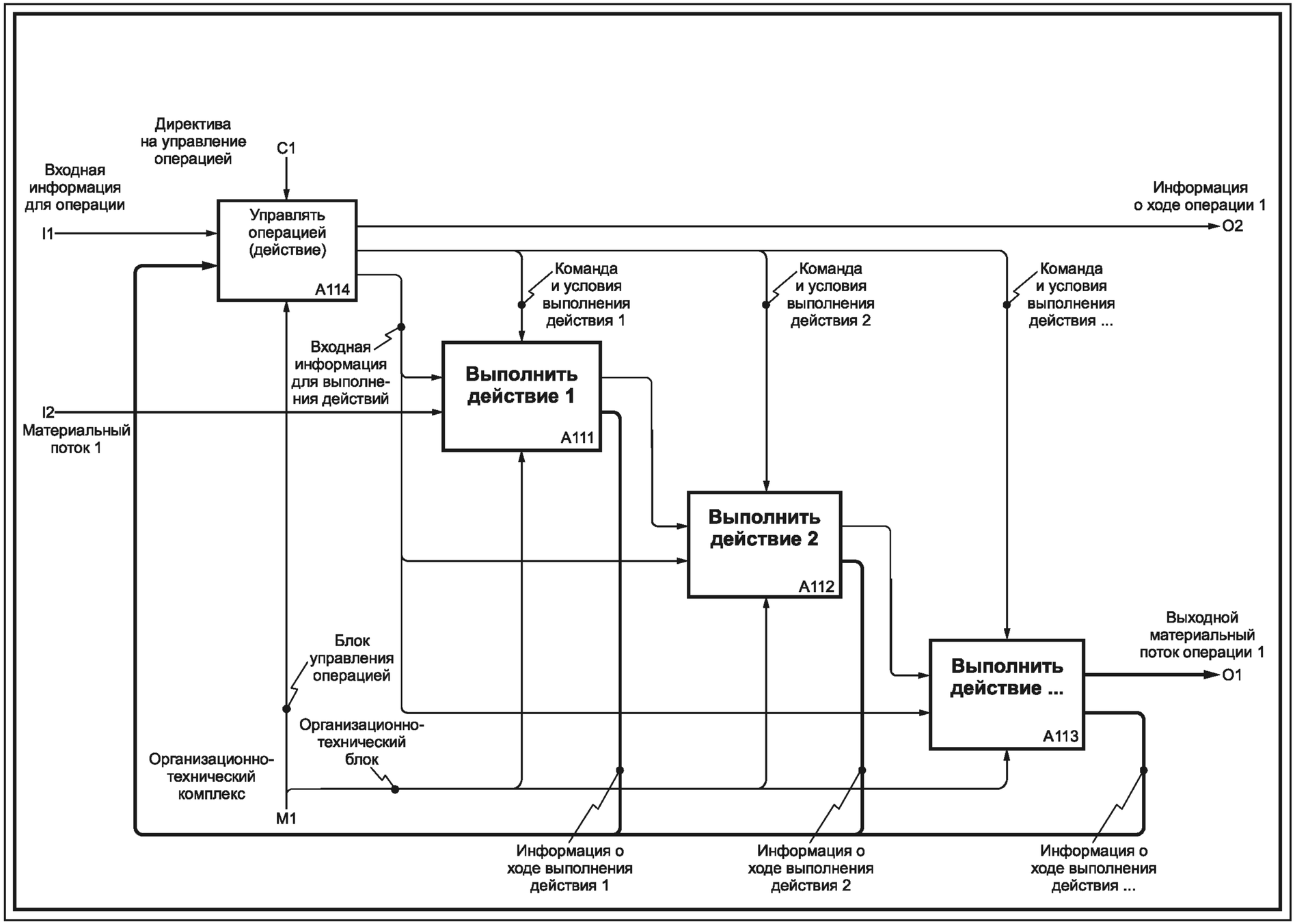


Рисунок Б.4 — Декомпозиция блока A11

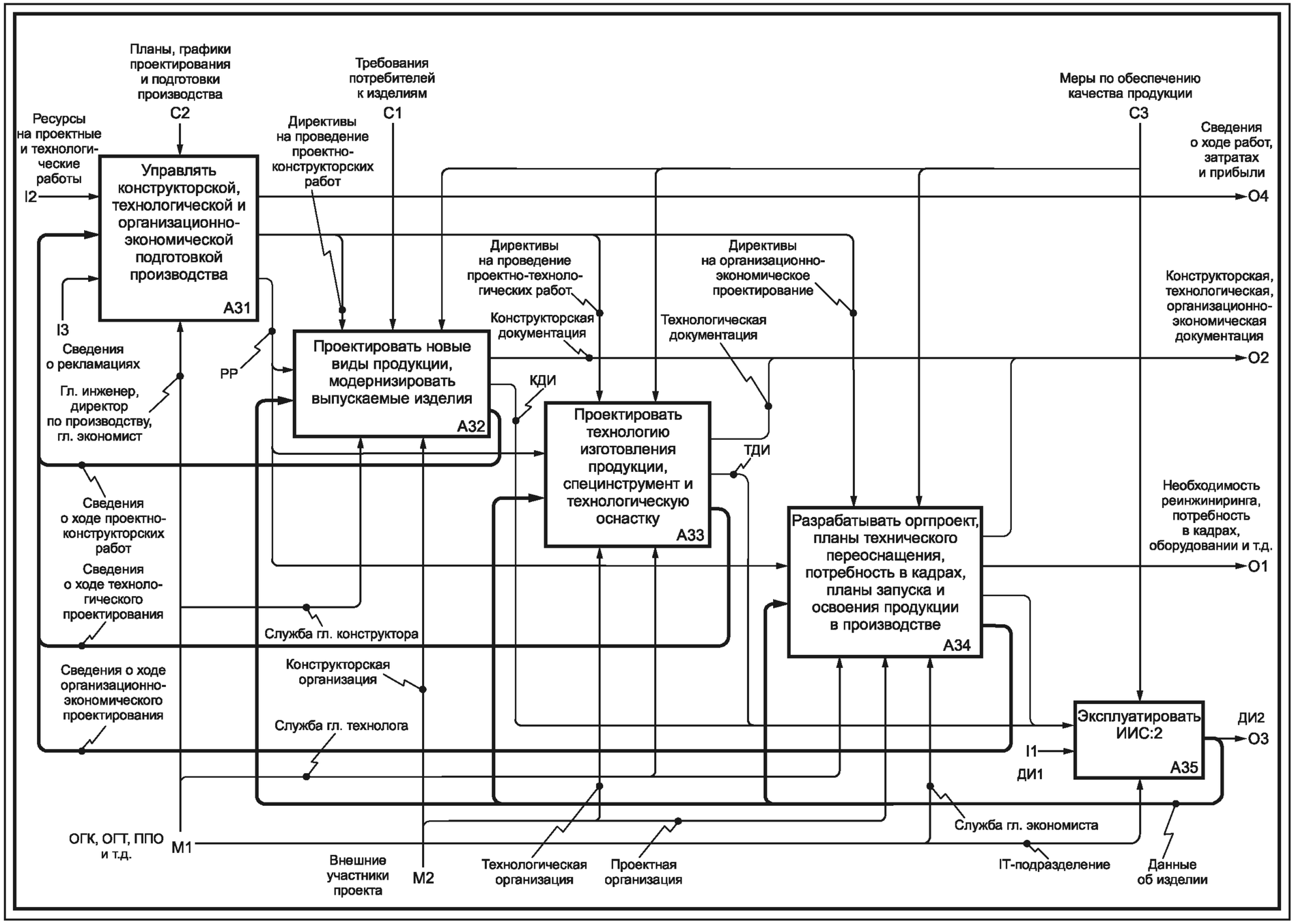


Рисунок Б.5 — Декомпозиция блока А3

ПРИЛОЖЕНИЕ В
(справочное)

Функциональная модель предприятия

На рисунках В.1 — В.7 представлены диаграммы фрагмента типовой функциональной модели промышленного предприятия в формате IDEF0.

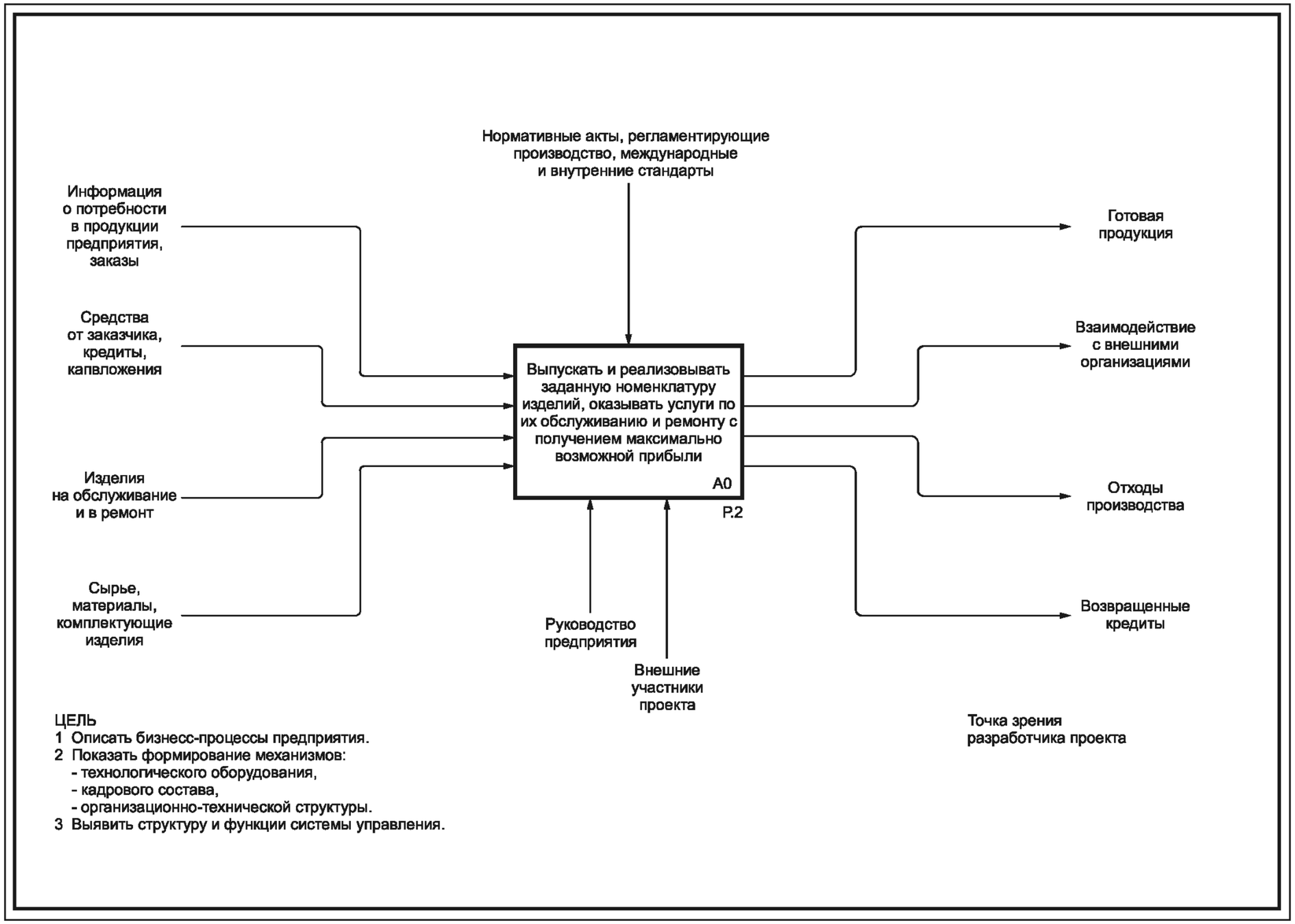


Рисунок В.1 — Контекстная диаграмма

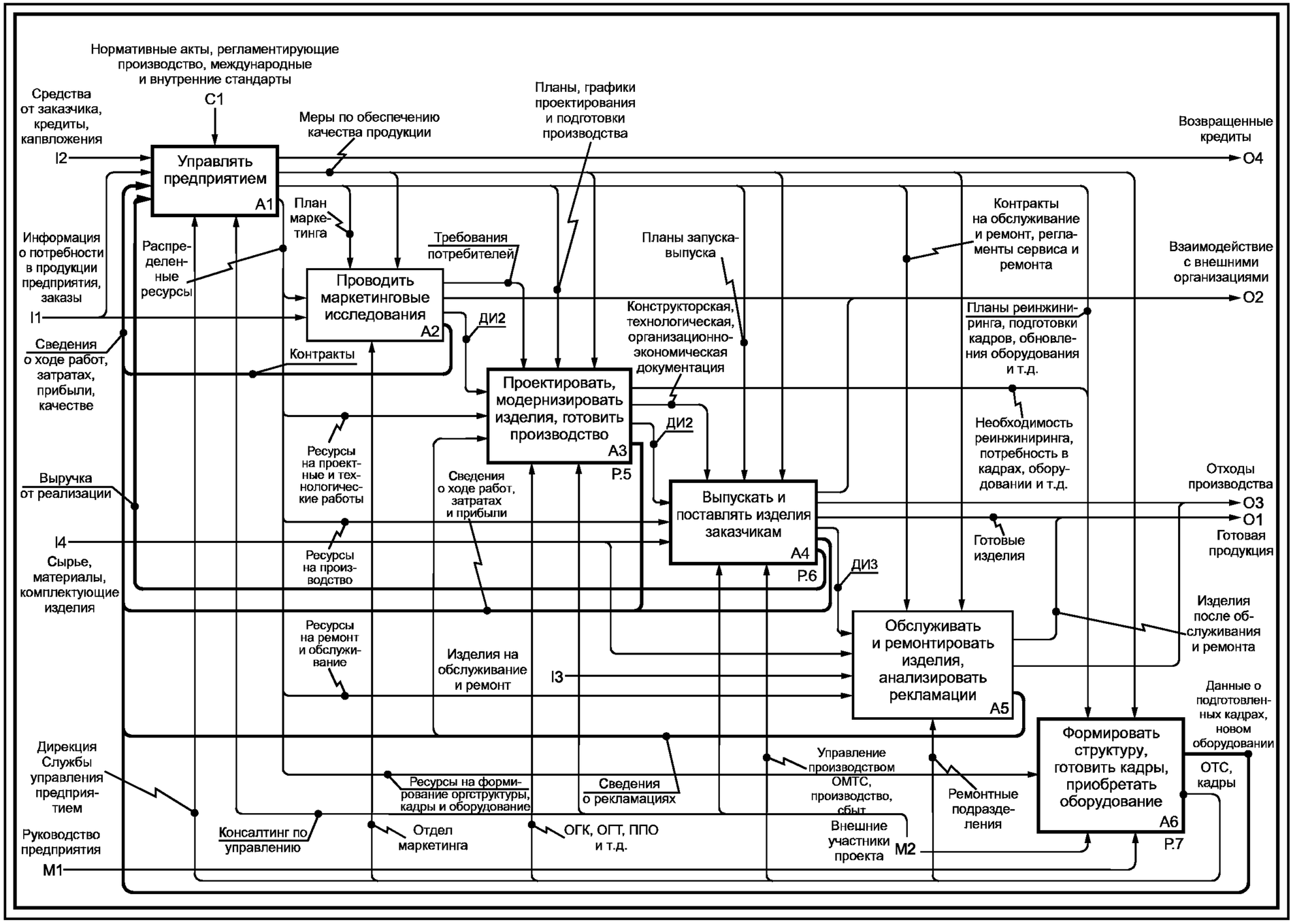


Рисунок В.2 — Декомпозиция контекстной диаграммы

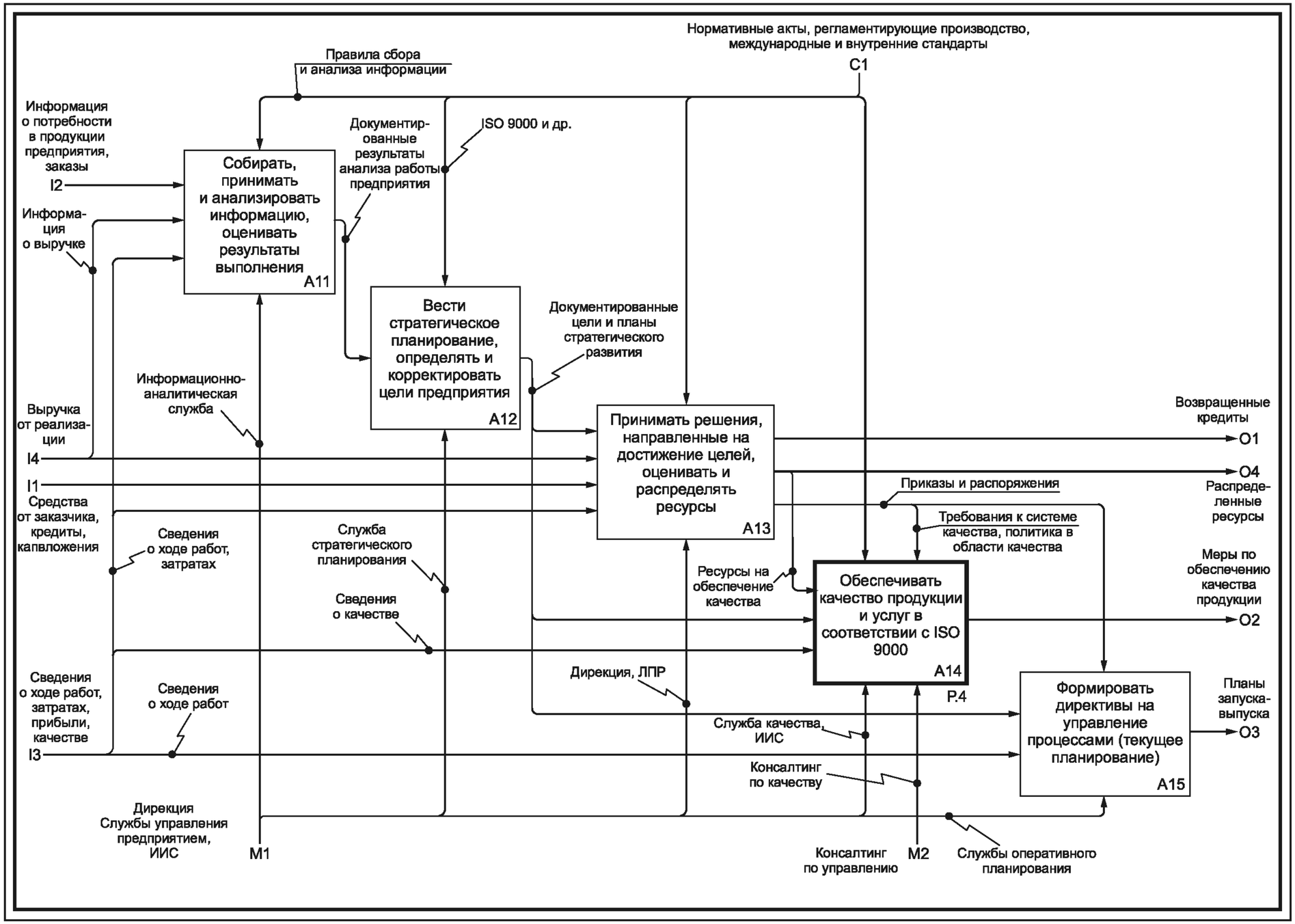


Рисунок В.3 — Декомпозиция блока А1

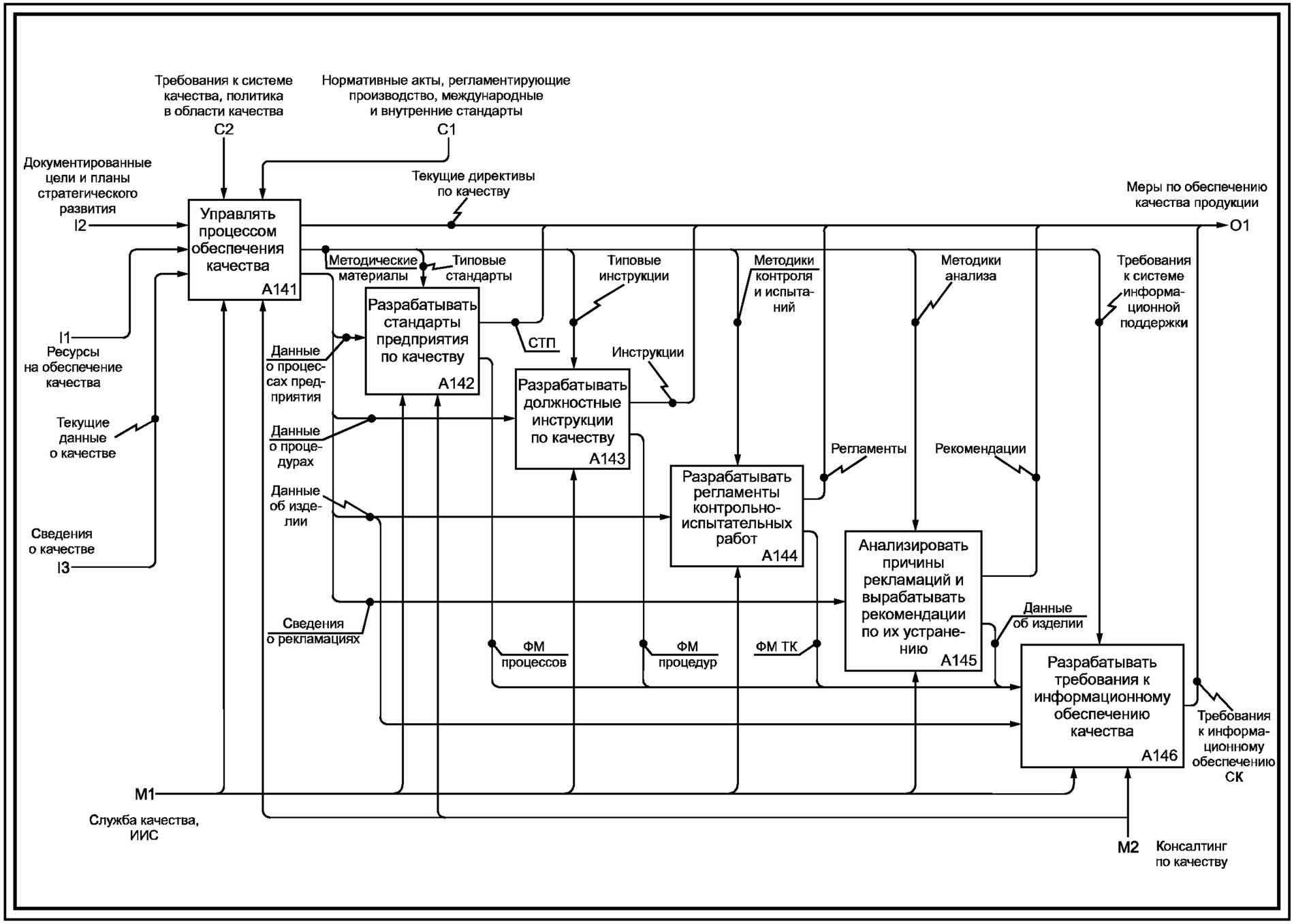


Рисунок В.4 — Декомпозиция блока А14

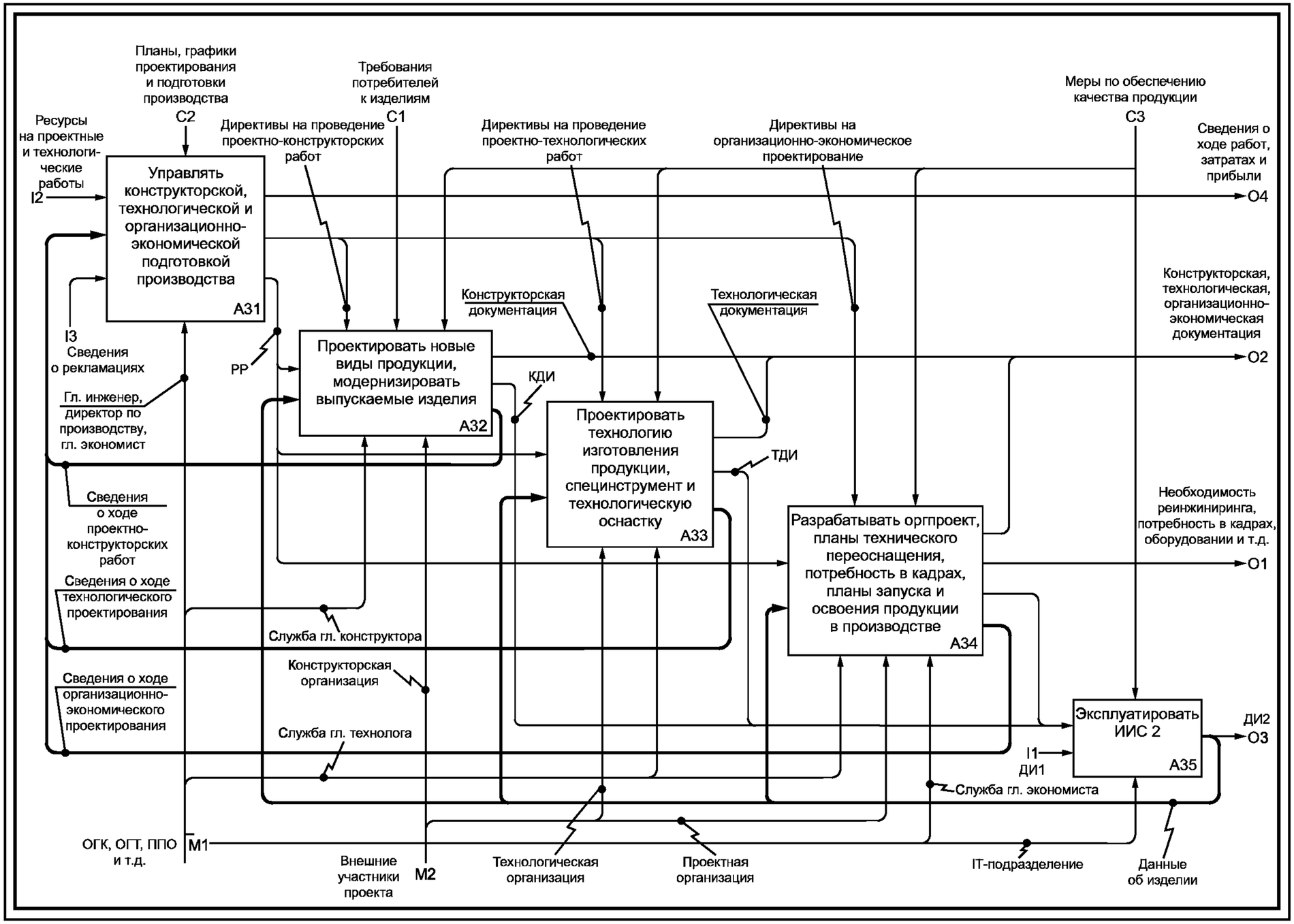


Рисунок В.5 — Декомпозиция блока А3

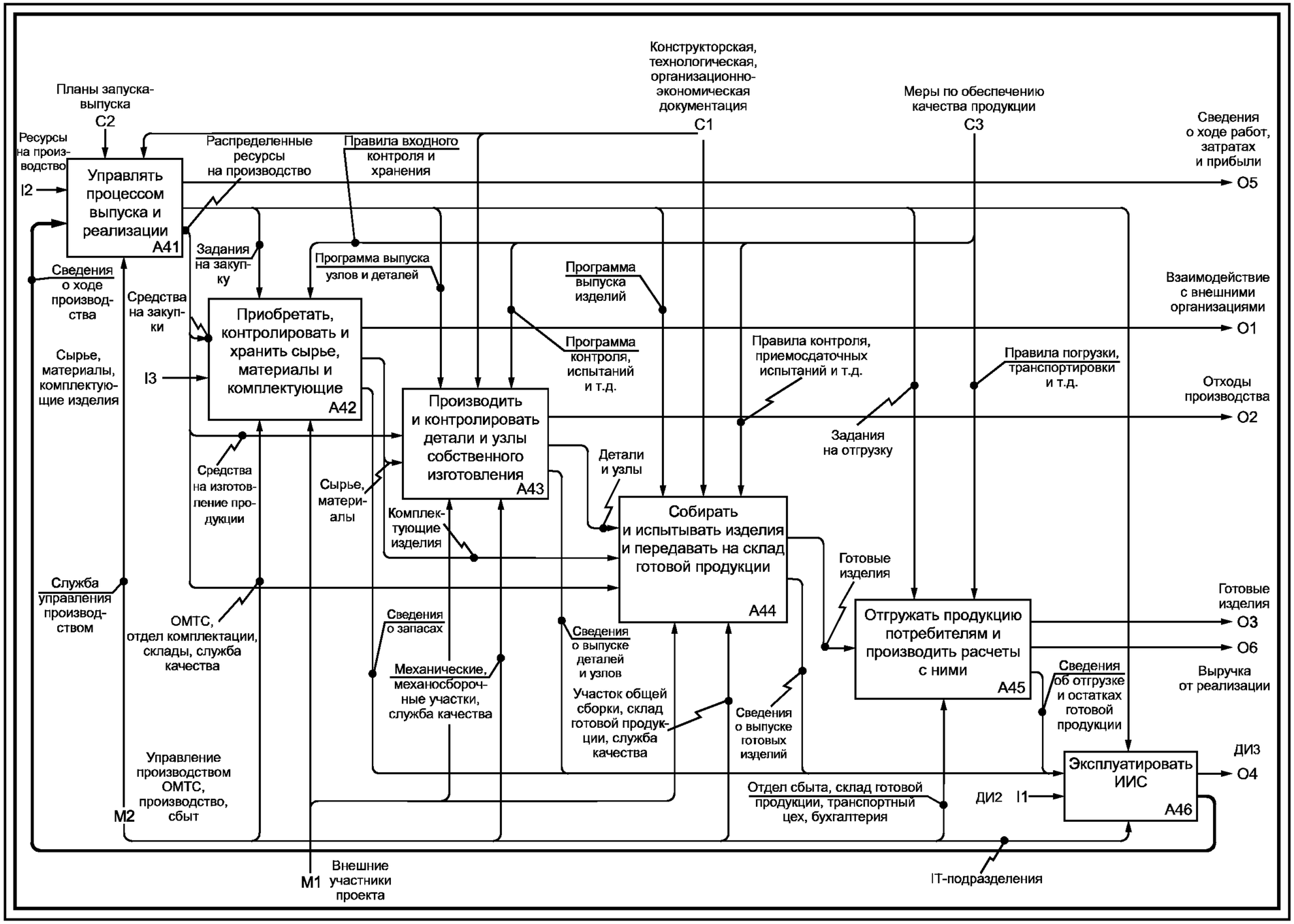


Рисунок В.6 — Декомпозиция блока А4

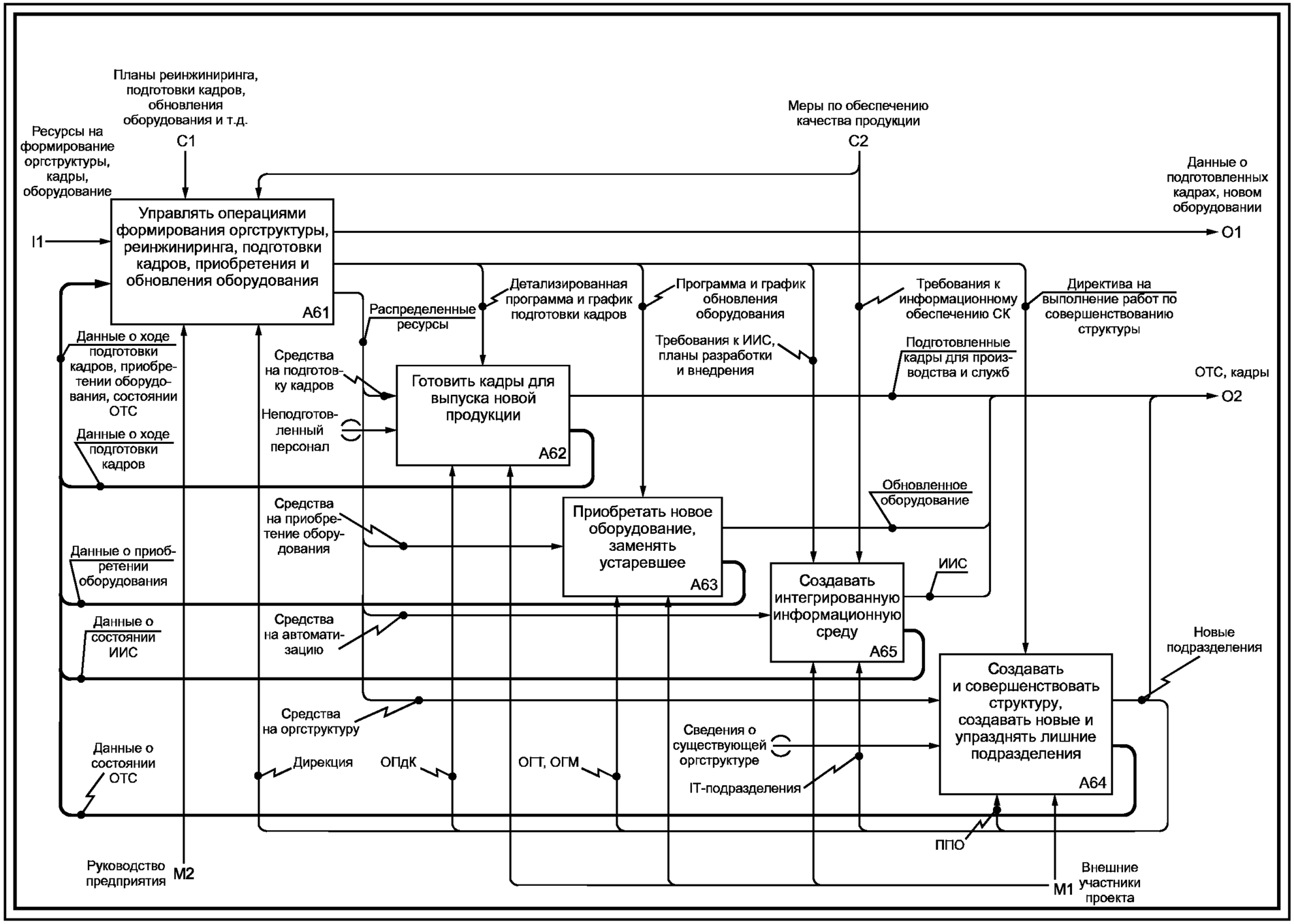


Рисунок В.7 — Декомпозиция блока А6

УДК 656.072:681.3:006.354 ОКС 25.040.40 П87 ОКСТУ 4002

Ключевые слова: моделирование, функция, структура, процесс, бизнес-процесс, реинжиниринг, операция, действие, графический язык, диаграмма, блок, стрелка, метка, вход, выход, управление, механизм, глоссарий

РЕКОМЕНДАЦИИ ПО СТАНДАРТИЗАЦИИ

Информационные технологии поддержки жизненного цикла продукции

МЕТОДОЛОГИЯ ФУНКЦИОНАЛЬНОГО МОДЕЛИРОВАНИЯ

Р 50.1.028—2001

Редактор *В.П. Огурцов*
Технический редактор *О.Н. Власова*
Корректор *Т.И. Кононенко*
Компьютерная верстка *Е.Н. Мартемьяновой*

Изд. лиц. № 02354 от 14.07.2000. Подписано в печать 12.02.2003. Формат 60×84 $^{1}/_{8}$.
Бумага офсетная. Гарнитура Таймс. Печать офсетная. Усл. печ. л. 6,05. Уч.-изд. л. 5,80. Тираж 70 экз. Зак. 52.
Изд. № 3041/4. С 9724.

ИПК Издательство стандартов, 107076, Москва, Колодезный пер., 14.
http://www.standards.ru e-mail: info@standards.ru
Набрано и отпечатано в ИПК Издательство стандартов.