

Сравнение некоторых современных методологий, применяемых для описания бизнес-процессов (*IDEF0, IDEF3, ARIS eEPC, ПОСТ нотация*)

Некоторые сокращения

IDEF (Integration Definition Metodology) - Объединение Методологических Понятий.

ARIS (Architecture of Integrated Information Systems) - Архитектура Интегрированных Информационных Систем.

eEPC (Extended Event Driven Process Chain) - расширенная нотация описания цепочки процесса, управляемого событиями.

ПОСТ (нотация)- Процесс Объект Связь Технология.

PFDD (Process Flow Description Diagrams) - диаграммами Описания Последовательности Этапов Процесса.

OSTN (Object State Transition Network) диаграммами Состояния Объекта в и его Трансформаций Процессе).

Введение

Неотъемлемым этапом разработки информационной системы является построение модели бизнес-процессов организации. Обычно для этого применяются нотации, согласно которым создаются процессные схемы (в данном случае часто предпочтение отдается графической интерпретации, как наиболее наглядной), которые содержат информацию о структуре деятельности организации.

В настоящее время разработано большое число различных нотаций, каждая из которых относительно остальных имеет определенные характеристические свойства (простым языком достоинства и недостатки). Эти свойства и определяют применимость той или иной нотации, которая в каждом конкретном случае зависит от рода деятельности организации и от того, какие именно аспекты деятельности должны быть описаны, какие типы моделей должны быть построены. Поэтому встает вопрос выбора подходящей нотации.

Сравним некоторые популярные методологии применяемые в настоящее время для описания бизнес процессов. Для достижения поставленной цели в работе рассмотрены нотации IDEF0, IDEF3, ARIS eEPC, ПОСТ нотация. Приведено краткое описание основных черт нотаций, которые важны для построения модели бизнес процессов организации, которая может использоваться впоследствии при разработке модели бизнес процессов организации. Затем будут сформулированы требования к нотациям,

которые, по мнению автора, наиболее подходят для оценки модели бизнес процессов организации, строящейся при разработке информационной системы для организации, деятельность которой связана с организацией документооборота.

Семейство методологий IDEF

Методология IDEF представляет собой семейство методов применяемых для процесса моделирования. Эта технология активно используется, начиная с конца 1980-х годов. Они нашли применение не только в бизнесе. Некоторые стандарты семейства IDEF приняты в качестве федеральных стандартов в США.

Ниже приведены некоторые основные методологии этого семейства:

- IDEF0 – представляет собой графический язык, позволяющий сформировать модель системы, в которой описывается функции и структура системы, подчиненность объектов системы, а также потоки информации и материальных объектов, связывающие эти объекты.
- IDEF1 – позволяет построить информационную модель, которая отображает структуру и содержание информационных потоков, обеспечивающих функционирование системы. Данная методология предоставляет возможность визуализации и анализа структуры и взаимосвязи этих потоков.
- IDEF1X – расширение методологии IDEF1, предназначенное для построения реляционных структур.
- IDEF2 – методология позволяет динамически моделировать эволюцию системы. Анализ динамических систем сопряжен с множеством трудностей, поэтому данный стандарт не нашел достаточно серьезного применения;
- IDEF3 – позволяет документировать технологические процессы. В IDEF3 описывается сценарий и последовательность действий для каждого процесса, таким образом, возможно подробно детализировать каждую функцию, используемую в IDEF0-диаграмме;
- IDEF4 – методология построения объектно-ориентированных систем.
- IDEF5 – стандарт, позволяющий анализировать онтологию рассматриваемой системы. Методологии IDEF5 описания онтологии системы включает в себя утверждение словаря терминов и описание правил и ограничений, согласно которым на базе введенной терминологии формируются достоверные утверждения, описывающие состояние системы. На основе этих утверждений с использованием словаря и правил формируются новые утверждения, позволяющие сделать вывод о состоянии системы в некоторый момент времени, что в конечном итоге дает возможность прогнозировать и требуемым образом корректировать процесс эволюции системы.

Методология IDEF0.

Функциональный блок (Activity Box).

Операции на схеме IDEF0 изображаются в виде функциональных блоков. Каждый блок должен иметь название, которое отображается в его теле. Оно должно быть выражено глаголом или глагольным оборотом, описывающим функцию. Помимо названия каждый блок имеет номер. Номера блоков используются для их идентификации на диаграмме и в тексте описания. На схеме функциональный блок изображается в виде прямоугольника.

Стороны функционального блока неравнозначны. При построении диаграммы каждая сторона выполняет определенную функцию:

- правая сторона обозначает результат процесса (Output);
- левая сторона обозначает входные элементы (Input);
- верхняя сторона имеет значение управляющего регламента (Control);
- нижняя сторона обозначает механизм исполнения операции (Mechanism).

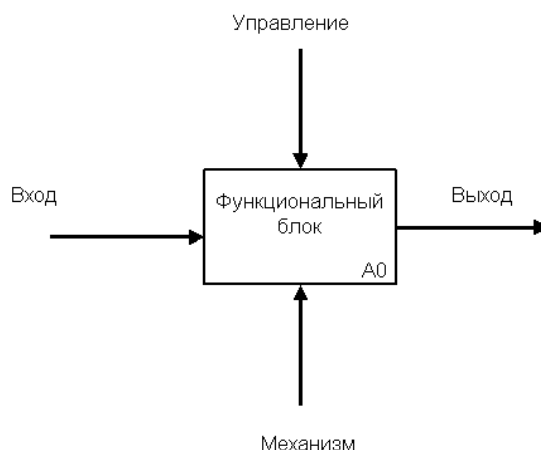


Рис. 1.

То, к какой стороне блока подключен объект, определяет его роль в данном действии.

Интерфейсная дуга (Arrow).

Интерфейсная дуга (иначе поток, стрелка) изображает объекты, которые влияют на функцию, отображенную данным функциональным блоком, одним из четырех вышеописанных способов (Output, Input, Control, Mechanism). В отличие от традиционных блок-схем процессов в IDEF0 стрелки не представляют поток или последовательность событий. Они лишь показывают, какие данные или материальные объекты должны поступить на вход функции для того, чтобы эта функция могла

выполняться. На диаграмме интерфейсные дуги изображаются в виде однонаправленных стрелок.

Началом и концом каждой интерфейсной дуги (кроме дуг, входящих в первый блок и исходящих из последнего) могут являться только функциональные блоки, причем в качестве источника выступает выходная (Output) сторона блока, а в качестве приемника - произвольная входная (Input, Control, Mechanism).

Для каждого блока обязательно должны присутствовать управляющая интерфейсная дуга (регламент, определяющий выполнение процесса или операции) и выходная интерфейсная дуга (результат выполнения процесса или операции). Необходимость наличия управляющих интерфейсных дуг в диаграммах IDEF0 отличает их от диаграмм потоков других нотаций.

Правила построения диаграмм в стандарте IDEF0.

В общем случае методология может характеризоваться по широте охвата в описании моделируемого объекта и по степени понятности и удобства восприятия. Причем данные характеристики связаны обратной зависимостью: с увеличением глубины и функциональности модели она становится более загруженной и сложной для восприятия. Для решения этой проблемы в методологию включают строгие стандарты, регламентирующие составления моделей, например требование к формату стрелки или блока. В приведенном примере использование различных типов стрелок и блоков для объектов с различными свойствами позволит сделать модель более информативной для человека, знакомого со стандартом, избежав нагромождения дополнительных элементов и комментариев [2].

Одним из приемов, предназначенных для облегчения восприятия модели, применяющимся при построении диаграмм процессов и потоков, является декомпозиция. Он заключается в представлении сложных действий в виде последовательности простых шагов.

Построение модели IDEF0 принято начинать с представления системы в виде единственного функционального блока. Подключаемые к нему интерфейсные дуги представляют собой объекты (ресурсы, регламенты, результат всего процесса), задействованные в процессе. Такая диаграмма именуется контекстной диаграммой верхнего уровня, и имеет номер «А-0».

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.

Стандарт жестко регламентирует методику образования номеров блоков. Номер каждого блока состоит из двух частей: префикса, единого для всех функциональных блоков в пределах одной диаграммы, и собственно номера блока. Префикс диаграммы совпадает с номером блока, детализацией которого является эта диаграмма.

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. Глоссарий предназначен для определения аббревиатур (акронимов), 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.

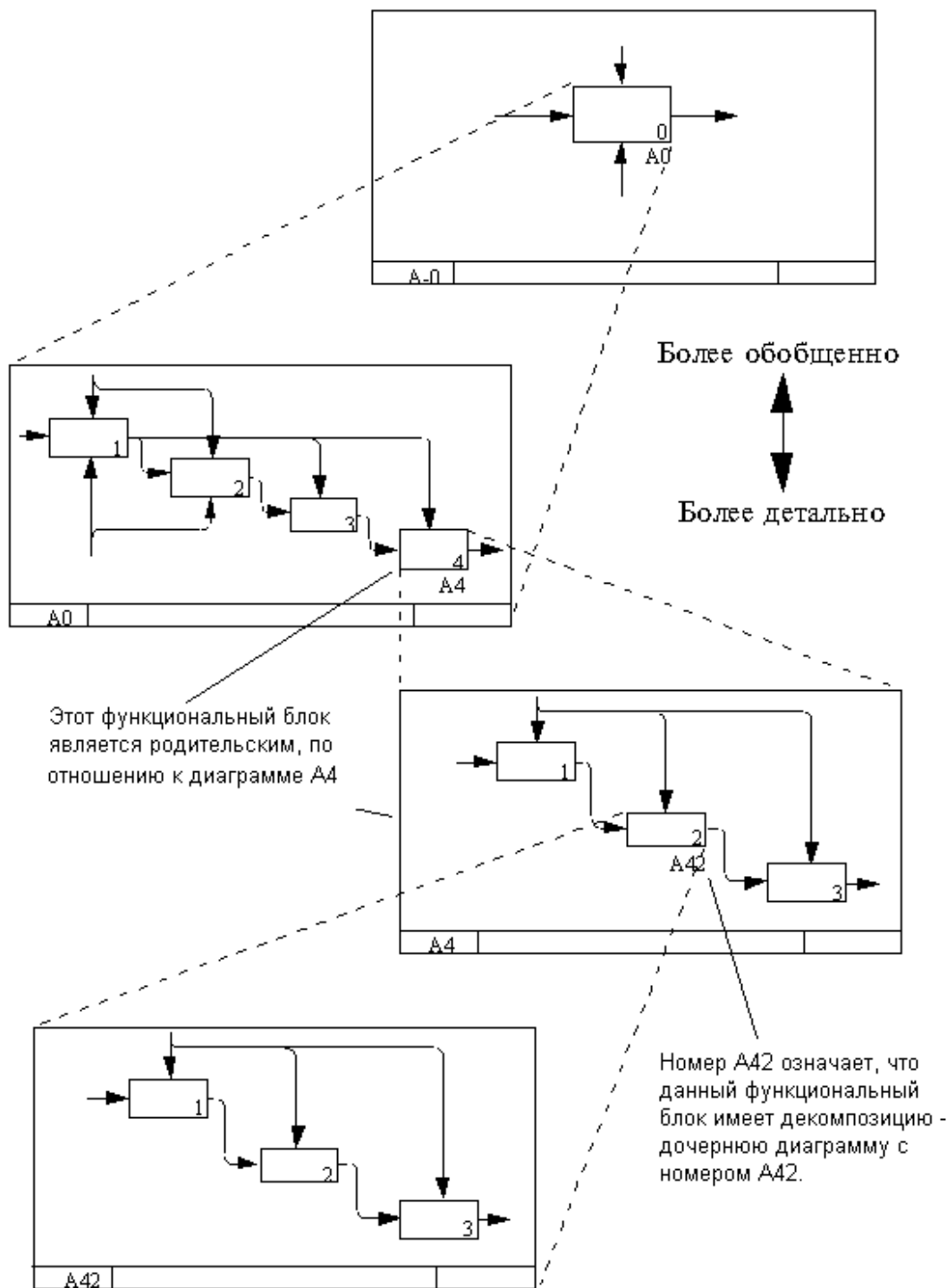


Рис. 2.

Блоки на диаграмме должны располагаться по диагонали в порядке присвоенных номеров – от левого верхнего угла диаграммы до правого нижнего. Блоки на диаграмме, расположенные вверху слева «доминируют» над блоками, расположенными внизу справа. «Доминирование» означает влияние, которое блок оказывает на те блоки, с которыми он взаимодействует. Доминирование между блоками отражает авторское понимание описываемого процесса.

Диаграммы более низкого уровня (не являющиеся контекстными) должны содержать от трех и до шести блоков. Это ограничение предназначено для удобства

использования диаграмм. Опыт показывает, что такого количества блоков на диаграмме оказывается достаточно для построения модели [9].

Также число интерфейсных дуг, подходящих к одному функциональному блоку, как и число дуг, выходящих из одного блока, не должно превышать дуг четырех.


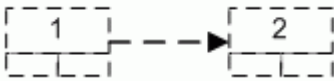

Стандарт IDEF3

Стандарт IDEF0, который был рассмотрен ранее, является развитием классического DFD – подхода и предназначен для описания бизнес-процессов верхнего уровня. Для описания временной последовательности и алгоритмов выполнения работ стандарт IDEF0 не подходит [7]. Для решения этой задачи стандарт IDEF0 получил дальнейшее развитие, в результате чего был разработан стандарт IDEF3, который входит в семейство стандартов IDEF.

Стандарт IDEF3 предназначен для описания бизнес-процессов нижнего уровня и содержит объекты – логические операторы, с помощью которых показывают альтернативы и места принятия решений в бизнес-процессе, а также объекты – стрелки, с помощью которых показывают временную последовательность работ в бизнес-процессе.

В стандарте IDEF3 связи между работами делятся на три типа, обозначения, названия и смысл которых, приведены в таблице.

Таблица 1. Типы связей между работами в стандарте IDEF3.

Связь предшествования		Применяется, когда второе действие начинает выполняться после завершения первого.
Связь отношения		Применяется, когда второе действие может начаться и даже закончиться до того момента, когда закончится выполнение первого действия
Связь потоков объектов		Применяется, когда второе действие начинает выполняться после завершения первого. При этом результатом первого действия является объект, название которого написано над стрелкой. Этот объект подается на вход второго действия.

Помимо наличия нескольких типов связей между работами в стандарте IDEF3 есть логические операторы, которые в данном случае называются перекрестками. Они также делятся на несколько типов: "Исключающий ИЛИ", "И" и "ИЛИ".

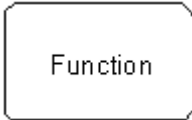


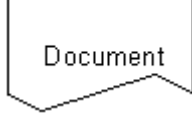
Перекрестки "И" и "ИЛИ" подразделяются еще на два подтипа – синхронные и асинхронные. Перекрестки синхронного типа обозначают, что последующие работы запускаются одновременно после завершения работы предшествующей. Перекрестки асинхронного типа требований к одновременности не предъявляют.

Диаграммы методологии ARIS

Система ARIS является комплексом средств для анализа и моделирования деятельности предприятия и разработки систем автоматизации. Состоящая из множества модулей, она позволяет осветить различные аспекты деятельности организации. Для описания бизнес-процессов в виде последовательности действий применяется модуль ARIS eEPC.

Следует заметить, что использование множества блоков и связей различных типов, загромождает модель, делает ее менее понятной и наглядной. Для решения задач, поставленных в работе, достаточно рассмотреть использование небольшого набора основных объектов. Эти объекты приведены в таблице.

Таблица 2. Обозначение некоторых объектов в нотации ARIS eEPC.

Функция	Описание	Графическое обозначение
Функция	Объект «Функция» служит для описания операций, выполняемых подразделениями или отдельными сотрудниками организации.	
Событие	Объект «Событие» служит для описания реальных состояний системы.	
Организационная единица	Объект, отражающий различные организационные звенья предприятия (например, управление или отдел)	
Документ	Объект, отражающий реальные носители информации, например бумажный документ	

Скелет процесса в методологии ARIS eEPC представляет собой набор событий и функций. Происшедшее событие влечет за собой выполнение некоторой операции, результат которой представляет собой некоторое событие и т.д. Скелет позволяет получить общее представление о ходе процесса. В случае ветвления процесса применяются логические операторы.

Для более глубокого изучения процесса в диаграмму можно добавить объекты других типов, например, исполнителей операций, какие объекты создавались или видоизменялись в процессе операции.

Существуют некоторые правила, регламентирующие построение диаграмм в нотации ARIS eEPC:

1. только событие может инициировать функцию;
2. результатом выполнения любой функции является событие (из 1. и 2. следует, что процесс обязан начинаться и заканчиваться с функции);
3. любая функцию должна иметь только одну входящую стрелку со стороны событий, инициирующих данную функцию, и только одну исходящую стрелку, являющейся результатом выполнения операции.
4. Так как событие «не может» принимать решение (принятие решения обязательно осуществляется в некоторой функции), применение операторов дизъюнкции OR и XOR после события запрещено.

На рисунках приведены примеры описания в нотации eEPC. На первом рисунке представлен скелет процесса, а на втором - использование множества объектов различных типов.

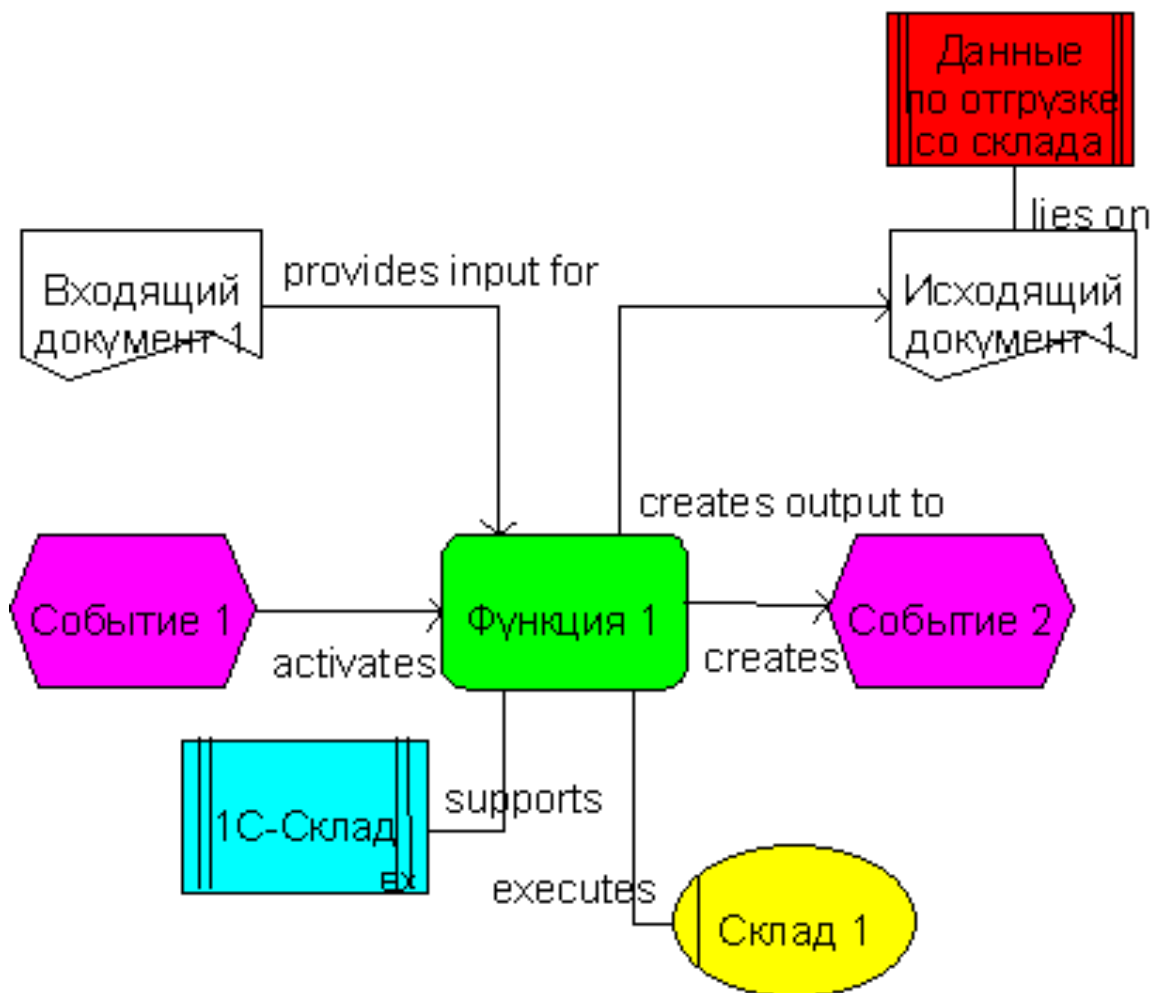


Рис. 3.

ПОСТ-модели для представления диаграмм процессов

Диаграмма процесса в ПОСТ-нотации как и в методологии ARIS eEPC изображает последовательность действий процесса. Одним из отличий от ARIS является то, что результатом операции и инициатором операции является не событие, а объект. Это оправдано тем, что для рассмотрения события человеком ему должны быть представлены сведения о нем. А передача сведений осуществляется посредством объектов.

В нотации ПОСТ функции и действия носят название «Превращение» и обозначаются, как показано на рисунке. При этом считается, что набор объектов поступающий, на вход операции преобразуется в набор объектов на выходе операции.

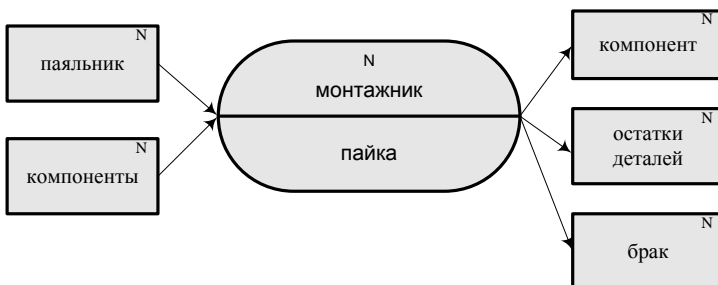
Превращения обозначаются овалами. В верхней части овала указывается исполнитель (например, человек или машина), в нижней действие, которое должно быть выражено отглагольным существительным, например, корректировка, установка.



На диаграмме объекты обозначаются как прямоугольники. Описание объекта производится с помощью существительных и прилагательных, например, буровая установка, отгрузочная накладная.

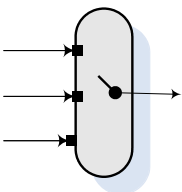


Тогда схема элементарного процесса, выглядит, например, так, как показано на рисунке.

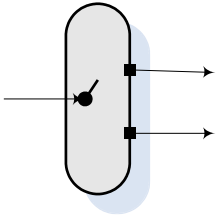


Для реализации альтернативного выбора, а также других типов ветвлений процесса применяются переключатели.

Символы соединения/переключения бывают двух типов: символы отождествления и символы растождествления. Символы отождествления используются для обозначения ситуаций, когда несколько входных объектов могут быть равнозначно взаимозаменены. Т.е. так, что при замене одного входного объекта другим процесс существенно не изменится.



Символы растождествления используются для обозначения ситуаций, в которых объект в рамках его универсальности может быть использован как входящий объект различных превращений, при условии, что использование экземпляров одного и того же объекта в разных превращениях существенно не меняет их. Использование переключателей позволяет процедурно определить использование альтернатив при разработке процессных сетей.



Начало любого процесса становится возможно при условии наличия необходимого и достаточного набора входных компонентов. Каждый из компонентов входного набора является объектом, необходимым для того, чтобы процессор (человек-исполнитель или технологическое устройство) мог осуществить превращение. В результате будет получен набор компонентов результата.

На ПОСТ-диаграммах прямоугольник всегда обозначает объект, который может являться как компонентом результата (одного процесса), так компонентом начала (другого процесса) в зависимости от того, какое из связанных с ним превращений рассматривается.

Правила, регламентирующие составление процессных схем в ПОСТ нотации:

1. нельзя напрямую соединять объекты, так как тогда не ясно, какое превращение преобразует один объект в другой;
2. нельзя напрямую соединять превращения, так как тогда не ясно, какой объект передан из одной операции в другую;
3. нельзя напрямую соединять стрелкой два переключателя, так как теряется определённость передачи: становится не понятно, из выхода какого из группы процессов взят объект и в какой из альтернативной группы последующих процессов направлен;
4. нельзя разветвлять соединительную стрелку: так как в данном случае теряется реальный смысл ветвления: некий объект «вдруг и как бы самопроизвольно размножился и поступил на переработку сразу во многие процессы».

ПОСТ-модель является иерархически организованной совокупностью диаграмм. Каждое из превращений в случае необходимости может быть детализировано на другой диаграмме нижележащего слоя, то есть каждый элемент (превращение) этой диаграммы может пониматься далее как отдельно определенный структурированный сложный процесс.

Как можно видеть ПОСТ нотация очень простая. В ней применяется малое количество различных символов: имеется всего два типа синтаксических знаков и только один тип стрелки. Несмотря на это нотация полна и позволяет описать практически большинство реальных бизнес-процессов [10].

Сравнение различных методологий описания бизнес-процессов

Планируется, что выбираемая нотация будет использована для описания бизнес-процессов организации, деятельность которой связана с документооборотом. Поэтому на диаграммах должны быть отображены объекты, которые обрабатываются в действиях бизнес-процесса, а также исполнители этих действий.

В результате были сформулированы требования, которым должны удовлетворять выбранная нотация:

1. модель бизнес-процесса должна отображаться в графическом виде;
2. бизнес-процесс должен описываться в виде последовательности операций;
3. на диаграмме должны быть указаны исполнители операций, а также объекты (документы), которые создавались или изменялись в результате операций;
4. не требуется указывать регламенты, управляющие выполнением конкретной операции.

Одним из основных недостатков методологии IDEF0 является отсутствие логических символов. Исследование области применения нотации показало наличие большего числа ветвлений описываемых процессов. Еще одним недостатком нотации является сложность для восприятия человеком плохо знакомым с данной нотацией (сложность обычно вызывает восприятия дуг на диаграмме [9]). Для комфортного ее восприятия требуется довольно большой опыт работы с данной методологией. Следует отметить также избыточность нотации, так как согласно пункту 4 наличие управляющих дуг в диаграмме не требуется. То есть придется отклониться от строгого соответствия стандарту. При этом смысл его использования становится неочевидным.

Сложные ветвления позволяет реализовать нотация IDEF3. Эта нотация является достаточно простой и понятной для восприятия. Автор были рассмотрены два типа диаграмм, формируемых в данной нотации. Диаграмма PFDD описывает последовательность операций, но не удовлетворяет требованиям, предъявляемым к нотации в пункте 3 (описание исполнителя операции и потоков документов). Использование диаграмм потоков OSTN позволяет лишь частично решить проблему. Тем более использование диаграмм двух типов существенно затрудняет восприятие модели, и, таким образом, не удовлетворяет пункту 4.

Нотацию ARIS eEPC можно считать расширением IDEF3 [9]. Она лишена недостатков, описанных в предыдущем абзаце. Данная нотация полностью удовлетворяет пунктам 1-3. Последнее предложение относится и к ПОСТ нотации. Следует сравнить данные нотации с позиции удобства и простоты использования.

Ранее в работах уже неоднократно упоминалась важная черта ПОСТ нотации – ее понятность на интуитивном уровне, позволяющая ей стать незаменимым средством

для разработки модели бизнес-процессов предприятия, выступая в качестве языка общения заказчика и разработчика, понятного им обоим [2, 6, 7, 10]. Другие нотации требуют серьезного изучения, что в случае, когда со стороны заказчика выступает группа лиц, весьма проблематично и затратно. Практика показывает, что разработка модели бизнес-процессов ведется с использованием средств MS Office и простейших графических редакторов. В результате при рассмотрении чуть более сложного бизнес-процесса между разработчиком и заказчиком сразу возникает непонимание.

Однако эта положительная черта имеет также побочный эффект, который связан с возможностью преобразования ПОСТ нотации в программный код. Это позволило бы существенно повысить ее эффективность и расширить область ее применения. Действительно, в этой нотации синтаксически различимы только два типа знаков – объекты и действия над объектами. Следовательно, автоматически преобразоваться они могут только в два соответствующих класса программных объектов. Надписи внутри неформальны и не позволяют преобразовать их в программный код. То есть, для более серьезных задач необходимы более сложные семиотические системы [6].

Тот уровень описания модели с использованием нотации ARIS что в работе носит название «скелета» диаграммы (описание лишь событий и действий) по простоте эквивалентен моделям в ПОСТ нотации. Но добавление дополнительных элементов (согласно пункту 3) существенно усложняет диаграмму.

Следует заметить, что ПОСТ нотация не только удовлетворяет всем предъявляемым требованиям, но ее применение задействует все элементы и нюансы данной нотации (все от типов структурных элементов, до надписей на диаграмме существенно и значимо). Ничего остается незадействованным.

Приведенные доводы позволяют заключить, что ПОСТ нотацию можно рекомендовать для построения модели бизнес-процессов организации. В случае требований к модели, совпадающих с рассмотренными в данной работе, из рассмотренных в работе нотаций ПОСТ нотация наиболее подходит для построения такой модели.

Выводы

Были рассмотрены некоторые нотации, применяющиеся в настоящее время для описания модели бизнес-процессов организации: IDEF0, IDEF3, ARIS eEPC, ПОСТ нотация. Были сформулированы критерии, предъявляемые к нотациям. Критерии формулировались для модели организации, деятельность которой связана с организацией документооборота. В результате сделан вывод, что из рассмотренных в работе нотаций ПОСТ нотация наиболее подходит для построения такой модели.

Помимо этого приведенные доводы позволяют заключить, что ПОСТ нотацию можно рекомендовать для построения модели бизнес-процессов организации, требования к которой несколько отличаются от требований, рассмотренных в работе.