

Жаринов В.Н.

АКТУАЛЬНЫЕ НАПРАВЛЕНИЯ ИНФОРМАТИЧЕСКОЙ ДЕЯТЕЛЬНОСТИ

Вводный цикл (извлечение)

Версия 09.1

ВВЕДЕНИЕ В ДОКУМЕНТ

Общие положения

1. Файл содержит выполняемый автоматизированным способом (в форме машинного оригинала МО) беловик целевого документа или его части (неотъемлемой), выделенной для удобства работы.

Документ в целом, кроме основного содержания, может включать приложения. Содержание документа, приложения (его выделенной части) составляют текст и/или иллюстрации (графчасть).

Конкретное наполнение файла определяется по его имени (полный формат имен см. [шаблон документа](#))¹.

2. Содержание документа, приложения подразделено на структурные элементы по иерархии; ее высшие 4 уровня стандартны. Элементы обычно имеют многоуровневую нумерацию и заголовки-абзацы, входящие в оглавление; возможны также элементы без нумерации, в т.ч. не входящие в оглавление, в т.ч. с заголовками в тексте.

В тексте применяются типовые приёмы оформления, описанные в [п/р 1.1 документа|шаблона](#).

3. В файл части из документа, приложения выделяется элемент структуры стандартного уровня иерархии (или ряд соседних элементов одного уровня) целиком (с заголовками).

Для многофайлового МО в имени каждого файла указаны индексы входящих элементов (формат: разделы <ЧN>, подразделы <пPNN>, пункты <ПNNN>, подпункты <пПNNNN>); файл первой части является *головным*.

При наличии приложений их форму (способ выполнения) указывают в отметках о наличии в составе единственного (или головного) файла основного документа (виды способов и формат отметок см. [шаблон](#)).

Приложения в МО могут выполняться как отдельные файлы *ПрилN* (что указывается в их отметках о наличии).

При наличии иллюстраций в документе, приложении (части) они также м.б. выполнены разными способами. Подрисовочные подписи включаются в оглавление для удобства поиска рисунков в документе.

Иллюстрации в МО могут содержаться в отдельном файле графчасти *Рисунки*; тогда текст содержится в файле *Текст*, и в нём дублируется подпись к каждой иллюстрации по месту её упоминания для отсылки к графчасти.

4. Оригинал документа (части) выполнен как настоящий файл (имя см. [поле внизу](#)) и другие необходимые (детальный состав многофайлового документа см. [п. 1.1.4](#) в <настоящем файле|головном файле *Ч.1 Введ.*>).

Текст подготовлен в среде OpenOffice.org Writer или иной программы, совместимой по файлам; иллюстрации выполнены в той же программе и/или иными средствами, включая захват машобразов для МО.

Подлинник выполняется как твёрдая копия с заменой и/или добавлением листов к твёрдой копии предыдущих версий, либо как машинный образ файлов оригинала по листам, с которого делаются твёрдые дубликаты.

5. Все права защищены их обладателями. Документ, а равно любая его часть в любой форме адресованы лицам, которые указаны автором как его адресаты и (или) третьим лицам, участвующим в совместной деятельности по соглашению между автором и указанными лицами; иное возможно только с письменного разрешения автора.

Документ предназначен для учебных, информационных, научных или культурных целей в соответствии с действующим законодательством РФ, включая, но не ограничиваясь, п.1 Ст.1274 ч.4 ГК РФ². Содержание документа используется «как есть», без к.-л. изменений. ПОЛЬЗОВАТЕЛЮ РАЗРЕШАЕТСЯ: создать резервную копию каждого файла оригинала (при предоставлении только подлинника – каждого его листа) на случай утраты; делать одну твёрдую копию МО для правомерного пользования, включая замену утраченных (испорченных, потерянных) листов; цитировать документ в объёмах и порядке, разрешённых нормами авторского права РФ. ПОЛЬЗОВАТЕЛЬ ОБЯЗАН: использовать оригинал (подлинник) и его копии (резервную и/или твёрдую) только лично и как указано выше; при цитировании документа ссылаться на источник³. Иное воспроизведение документа или любой его части невозможно без письменного разрешения.

Информация, содержащаяся в документе, получена из открытых источников, рассматриваемых автором как надёжные. Возможное наличие секретных, конфиденциальных, а равно иных сведений ограниченного доступа следует рассматривать как результат предположения на массивах открытых сведений. Имея в виду возможные человеческие и технические ошибки, автор не может гарантировать абсолютную точность и полноту приводимых сведений, и не несёт ответственности за возможные последствия, связанные с их использованием.

¹ Переменные части текста даются как поля в '< >', заменяемые на описание; общая часть (корень) поля пишется как есть, а изменяемые части как '*'. Файлы МО с однокоренным именем относятся к одному элементу структуры.

² Федеральный закон № 230-ФЗ от 18 декабря 2006 г.

³ Если цитата состоит полностью из сведений, цитирующих другой источник – дать ссылку на первоисточник.

Назначение, сведения о версиях, языковые соглашения

1. Документ предназначен для представления содержания документа (его неотъемлемой части) как загружаемого инфордока в составе веб-ресурса.

2. Версии документа выпускаются по мере обновления содержания цикла.

3. В тексте документа употребляются следующие типовые обозначения и сокращения:

англ.	английский;
букв.	буквально;
в т.ч.	в том числе;
жарг.	жаргонный;
и т.д.	и так далее;
и т.п.	и тому подобное;
к.-л.	какой-либо;
напр.	например;
нем.	немецкий;
см.	смотри;
т.е.	то есть;
т. зр.	точка зрения;
т.о.	таким образом;
разд.	раздел (документа);
п/р	подраздел (документа);
п.	пункт (документа);
п/п	подпункт (документа);
устар.	устаревший (о предмете, понятии);
фр.	французский;

Оглавление

ПРИЛОЖЕНИЕ 2. ЭЛЕМЕНТЫ ЯЗЫКОВОГО ОБЕСПЕЧЕНИЯ ВИЗУАЛИЗАЦИИ ДЕЯТЕЛЬНОСТИ.....	4
2. ДЕКЛАРАТИВНЫЕ ЯЗЫКИ.....	5
2.1. Язык схем данных.....	5
<i>2.1.1. Алфавит языка.....</i>	<i>5</i>
<i>2.1.2. Правила языка.....</i>	<i>6</i>
2.2. Язык атрибуции данных.....	7
<i>2.2.1. Алфавит языка.....</i>	<i>7</i>
<i>2.2.2. Правила языка.....</i>	<i>9</i>
3. ИМПЕРАТИВНЫЕ ЯЗЫКИ.....	10
3.4. Язык протоколов деятельности.....	10
<i>3.4.1. Алфавит языка.....</i>	<i>10</i>
<i>3.4.2. Правила языка.....</i>	<i>10</i>
4. ЯЗЫКИ ОПИСАНИЯ ИСПОЛНИТЕЛЕЙ.....	12
4.1. Структурно-топологический язык.....	12
<i>4.1.1. Алфавит языка.....</i>	<i>12</i>
<i>4.1.2. Правила языка.....</i>	<i>12</i>

5. ЯЗЫКИ ОБОБЩЁННОЙ ФОРМАЛИЗАЦИИ.....	14
5.2. Язык синтаксических диаграмм.....	14
5.2.1. Алфавит языка.....	14
5.2.2. Правила языка.....	14
5.2.3. Конструкции языка.....	16
5.2.3.1. Базовые структуры содержания.....	16
5.2.3.2. Диоформы синтаксических диаграмм.....	18

Приложение 2.

ЭЛЕМЕНТЫ ЯЗЫКОВОГО ОБЕСПЕЧЕНИЯ ВИЗУАЛИЗАЦИИ ДЕЯТЕЛЬНОСТИ

В приложении кратко описаны искусственные языки представления знаний, применяемые для описания систем различного рода. Данные описания можно рассматривать как предложения по языковым стандартам, в частности, в силу полуформальности определений; тем не менее они отражают реалии применения соответствующих языков и после усвоения могут составить основу инструментария представления профессиональных знаний.

Приложение предназначено для обучения работе с языками и обсуждения их содержания с целью уточнения состава языков и их определений.

В данное извлечение включены языки, не подлежащие первоочередной реализации в составе системы (среды) разработки и документирования процессов, либо необязательные для такой реализации вообще (в силу второстепенной роли в формализации знаний, недостаточной определённости и других причин).

2. ДЕКЛАРАТИВНЫЕ ЯЗЫКИ

2.1. Язык схем данных

Служит для графического описания состава и взаимосвязи объектов данных конкретной задачи, решаемой с применением информатических машин (систем, комплексов, сетей).

2.1.1. Алфавит языка

Алфавит образуют символы, перечисленные в таблице ниже:

Обозначение	Наименование, характеристика значения
	Ручной ввод – объект входных данных в символьной форме, значения элементов (реквизитов, величин) которого задает оператор непосредственно в порядке, определённом <i>сценарием диалога</i> с информатиной (или иным исполнителем).
	Документ – входной символьный объект в натуральной форме, реквизиты которого <i>организованы</i> (структурированы и упорядочены) согласно формуляр-образцу (шабл) документа. Вид документа (бумажный, фото-, фоно- и т.п.) указывается контуром из числа показанных или другим стандартным.
	Массив – предметный объект, представленный как содержимое области основной (адресуемой) памяти информатины, организованное согласно типу массива.
	Файл – предметный объект, представленный как содержимое носителя внешней памяти информатины, организованное по правилам файловой системы машины для данного носителя.
	Видеограмма – выходной символьный объект, в котором значения элементов представлены образами на устройстве отображения данных согласно экранной форме объекта.
	Машинограмма – выходной символьный объект, значения реквизитов которого представлены твёрдыми копиями на натуральном носителе данных (бумаге или др.), организованными согласно конструктиву (печатной форме) объекта.
	Входная кодограмма – предметный объект, значения элементов которого представлены сигналами на входе к-л. интерфейса информатины, упорядоченными согласно одному из протоколов обмена по этому интерфейсу (средствами внешнего отправителя или самой машины).
	Выходная кодограмма – предметный объект, значения элементов которого представлены сигналами на выходе интерфейса информатины, упорядоченными (средствами машины) согласно протоколу обмена.
	Связь – сопряжение объекта и процесса. Метка указывает конкретные элементы объекта, передаваемые по связи.
	Процесс – символизирует решение задачи (подзадачи) как преобразование входных объектов в выходные. Нижняя форма применяется, если нужно указать имя процесса (напр. для процессов подзадач декомпозированной задачи).

2.1.2. Правила языка

2.1.2.1. Схемы данных представляют собой орграфы, причём рёбрами являются линии связи, а все остальные символы языка трактуются как вершины различных типов. Направление на схеме специально не указывается, т.к. соответствует привычному направлению чтения.

Схемы составляются по следующим основным правилам.

- схема образуется из объектов и процессов, причём объекты связаны только через процессы;
- схема состоит по крайней мере из одного процесса, одного входного и одного выходного объектов данных (конкретный состав объектов определяется содержанием процесса);
- среди входных объектов выделяются те, которые содержат условно-постоянные данные;
- основные признаки объектов: стабильности элементов (указание для данных – слово «Справочник»); природы данных (для отчуждённых знаний, напр. – слово "Знания"); сочинитель может вводить и иные признаки, описав их как дополнительные правила языка;
- объект *Массив* употребляется только когда его данные содержатся внутри одной машины (системы с общим полем ОП) и только если предполагается, что другие задачи (источники/потребители этих данных) всегда выполняются в течение одного сеанса с данной задачей.
- Имя объекта (метка) на реальной схеме указывает его тип; метки д. б. краткими, уникальными для всей совокупности задач описываемой системы (напр. организации, использующей язык). Для наглядности метку рекомендуется производить от полного имени объекта (документа, сообщения) по общеязыковым правилам (см. [Разд.1 Прил.7](#)).
- в задаче могут использоваться разные формы одного объекта. Тогда на схеме помещают одинаково именованные обозначения этого объекта;
- список *ИмяЭл* применяется, когда нужно указать на использование в процессе части содержимого составного объекта (напр., конкретных реквизитов документа);
- для первоначальной схемы, когда точно неизвестно, к какому виду отнести объект, можно обозначить его как наиболее часто встречающийся (первый в подгруппе);
- схема данных м.б. укрупнённой (задача отображается как единое целое, т.е. одним блоком «Процесс») или детальной;
- на детальной схеме выделяются подпроцессы по этапам преобразований ресурсов (на схеме появляется несколько уровней линий процессов, связанных через объекты промежуточных данных) и/или по независимым друг от друга функциям задачи/этапа (на одном уровне размещается несколько линий с разрывами, т.е. линия процесса как бы разомкнута на отрезки). Каждая линия помечается именем соответствующего подпроцесса (функции).

Основным типом компоновки схем является вертикальная, при которой:

- схема в целом строится и читается сверху вниз; символы процессов ориентированы горизонтально, а связи, соответственно – вертикально;
- символы входных объектов располагаются в ряд над символом «своего» процесса, а символы выходных – также в ряд под ним.

Для удобства размещения в заданном формате допускается горизонтальная компоновка схемы, при этом она строится и читается справа налево, т.е. символы объектов расположены по столбцам, символы процессов ориентированы вертикально, а связи – горизонтально.

2.1.2.2. Язык исходит из классификации объектов данных задачи по ряду оснований, прежде всего по роли в процессе (входные/выходные/промежуточные, что отражается в компоновке схемы), далее по форме представления (символьная/предметная, отражается видом) и источнику поступления/выдачи (отражается разными типами объектов); далее по степени стабильности значений их элементов (отражается признаком и группировкой входных объектов).

Для структурирования дополнительно применяются элементы КогниСтиль (прежде всего скобки и/или ограничители).

2.2. Язык атрибуции данных

Служит для описания свойств объектов данных (элементарных, сложных) с т. зр. формального исполнителя (напр. информшины) в текстово-табличной форме. Употребляется, как правило, совместно с языком схем данных.

2.2.1. Алфавит языка

Алфавит образуют значения атрибутов данных. Идентификаторы атрибутов и значений перечислены в таблице ниже:

Значение (имя)	Характеристика	Примечания
ОСНОВНЫЕ АТТРИБУТЫ		
Имя элемента		
<текст-имени>	Представляет естественное имя для человека; даётся по правилам родного языка	
Метка элемента		
"Имя_элемента"	Представляет формальное имя для моделей (формул, алгоритмов, схем); дается по общеязыковым правилам	(см. п/р 1.1 Прил. 7)
Система кодирования (показателя составного реквизита)		
"Текст"["("<Имя_яз">")"]	Указывает на запись данных в алфавите некоего языка; "Имя_яз" – краткое наименование языка: «рус», «матем.» и т. п. В машинном представлении знакам алфавита соответствуют коды заданной разрядности; возможны разные кодировки одного алфавита.	При наличии вариантов кодировки указать нужный
<N->"ичная"	Указывает систему счисления числовых данных; <N-> – корень названия СС: десятичная, 16-ричная и т. д.	
<"Форм." "Слов.">	Указывает способ кодирования данных-величин той или иной формальной логики (напр. булевой): формальный – знаками (кодами) некоего математического алфавита по числу возможных значений величины, словесный – словами некоего естественного или искусственного (напр. машинного) языка.	Возможные коды булевых величин: формальный – цифрами 0/1; словесный русский – да/нет; машинный – избранными значениями байта, машслова.
Тип данных (показателя составного реквизита)		
"Симв."	Обозначает символьный тип (данные рассматриваются как строки знаков или кодов в некоей кодировке)	Кодируется как текст
"Числ."	Обозначает числовой тип (данные рассматриваются как числа в некоторой системе счисления)	Кодируется как число
"Логич."	Обозначает логический тип (данные рассматриваются как значения величин некоторой логики)	Обычно – булевой
"Дата"	Для данных времени в специальном формате (напр. машинной базы данных, динатаба)	Длина поля фиксирована

Значение (имя)	Характеристика	Примечания
"Прим."	Для данных типа <i>Примечание</i> (машинных текстов для человека, не обрабатываемых автоматически)	Только в некоторых СУБД
Длина показателя		
m["/"]n]	Числовой код, где: m – предельная длина строки символов, словесного кода логич. величины или целой части числа; n – длина дробной части (только для числа)	При переменной длине показателя
Ключ сортировки		
<k>	Числовой код k=1,2,...N, определяемый для реквизита объекта (составного – некоторых или всех); значения указывают очередность алгоритмического упорядочения массива объектов: сначала по реквизитам с ключом 1, потом по 2 и т.д.	Для массивов объектов, подлежащих сортировке
ДОПОЛНИТЕЛЬНЫЕ АТТРИБУТЫ		
Способ ввода (получения значения реквизита)		
"Свободный"	Адресант задачи (обычно человек-оператор) указывает любое значение допустимого типа (может не проверяться адресатом).	
"Выбор"	Указывает, что человеку предлагается список значений для ввода; он выбирает одно значение.	
"Программный"	Указывает, что значение получается как результат обработки по некоторому алгоритму, входящему в решение данной задачи.	
Метод контроля		
"Совпадение"	Указывает, что адресатом задачи (напр. машиной) принимается допустимое значение реквизита (единственное или одно из возможных).	
"«Вилка»"	Указывает, что значение (числовое) проверяется адресатом на соответствие некоторому диапазону; ввод, не попадающий в диапазон, отвергается с выдачей сообщения адресанту.	
"Логический"	Указывает, что адресатом проверяется, выполнено ли некоторое соотношение результатов ввода с другими величинами (в частности, соответствие заданному типу данных).	
Параметры контроля		
{определение-i}	перечисляются значения-уставки и/или логические условия (правила) для i-того контролируемого реквизита.	

2.2.2. Правила языка

Атрибуция данных проводится по следующим правилам.

А. Организация состава объекта м.б. описана одним из двух способов:

- формальным – посредством структурной схемы объекта, показывающей отношения его элементов (напр. в виде классификационной диаграммы);
- конструктивным – посредством конструктива объекта; показывающего конфигурацию и вид элементов (напр. в виде шаблона документа, чертежа предмета).

На практике можно сочетать оба способа.

В частном случае объект может состоять из единственного элемента.

В общем случае описываемый элемент объекта (напр., реквизит документа) является *составным*, т.е. включает *основание* (постоянную часть, определяющую значение всего реквизита) и *показатель* (значение, возможно, переменное); показателей м.б. несколько, напр. значение величины и обозначение единицы измерения значения.

Основанием составного реквизита служит его имя; оно же указывается и в конструктиве объекта данных (натуральном бланке, экранной/печатной форме).

Б. Основные атрибуты конкретного объекта (элемента составного объекта) организуются в виде записи. Для каждого атрибута выбирается одно из возможных значений. Записи об элементах объекта сводятся в таблицу вида:

Имя реквизита	Метка	Тип данных	Система кодирования	Длина показателя	Ключ сортировки

В. Дополнительные атрибуты вводятся (выборочно) для реквизитов, значения которых получаются извне задачи (скажем, заполняются оператором при машинном решении) и определяют способы и критерии контроля правильности значений. Эти атрибуты также организуются в виде записей об элементах объекта, сводимых в таблицу вида:

Метка	Способ ввода	Метод контроля	Параметры контроля

Атрибут *Метка* указывает на принадлежность дополнительных атрибутов объекту, который имеет ту же метку в основной таблице. Для любого другого атрибута допускается одно и более из возможных значений (что означает множественность ввода и/или контроля).

Г. Допустима общая характеристика для группы полей (класса реквизитов) одной строкой (под обобщённым именем реквизита).




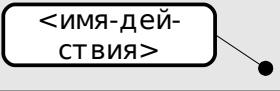
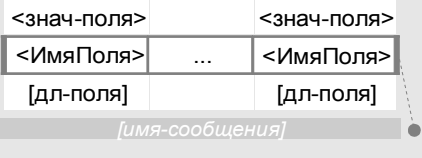

3. ИМПЕРАТИВНЫЕ ЯЗЫКИ

3.4. Язык протоколов деятельности

Служит для графического представления процесса решения задачи (подзадачи) как ряда взаимодействий между набором сущностей (объектов). Рекомендуются для начальной формализации процессов взаимодействия между объектами (напр. инфокоммуникационного); входит как составная часть в различные методологии моделирования.

3.4.1. Алфавит языка

Алфавит образуют символы, перечисленные в таблице ниже:

Обозначение	Наименование, краткая характеристика
	ПД1. Сущность – объект, взаимодействующий с другими сущностями диаграммы (порождающий события для них).
	ПД2. Внешняя сущность – объект, взаимодействующий с сущностью как часть её внешней среды.
	ПД3. Ось – отражает временной интервал процесса, описываемого диаграммой; начало отсчёта (t=0) в верхней точке оси.
	ПД4. Функция – часть процесса, исполняемого сущностью, порождающая сообщение другой сущности (или самой себе).
	ПД5. Сообщение – объект передачи в рамках события; содержание перехода. Поля м.б. расшифрованы в легенде диаграммы.
	Переход – событие взаимодействия объектов; интерпретируется как сообщение, посылаемое другому объекту или самому себе (обозначается петлёй).

3.4.2. Правила языка

Протокольные диаграммы представляют собой орграфы, причём все объекты, кроме дуг перехода, являются вершинами. На схеме указываются направления рёбер (стрелками связей).

- диаграмма состоит по крайней мере из двух объектов (с осями) и одного перехода;
- сущности располагаются в ряд по правилу «чем правее, тем позже включается во взаимодействие»;
- ось совмещается с серединой нижней стороны блока сущности;
- точки сопряжения переходов с осями упорядочиваются согласно моментам отправки/получения сообщений; разность начальной и конечной точек одного перехода отражает время на его совершение (для межосевого перехода это ведёт к наклону линии сверху вниз);
- переходы между осями отражают процессы в связях между сущностями, разомкнутые и петлевые переходы (только по оси одной сущности) — вышестоящие (прикладные) процессы сущности, приводящие к необходимости взаимодействия её с др. сущностью;
- сообщение может рассматриваться либо как вершина, включаемая в разрыв линии перехода, либо как метка, сопрягаемая с этой линией (зависит от авторского стиля);
- в табличном описании сообщений верхняя строка содержит значения полей (в принятой символической системе кодирования; могут не указываться) средняя строка — имена полей, нижняя строка — длины полей (в битах);

- время диаграммы м .б. как относительным, так и абсолютным; во втором случае вводится дополнительная ось-шкала времени;
- длительности на диаграмме могут обозначаться размерными линиями.



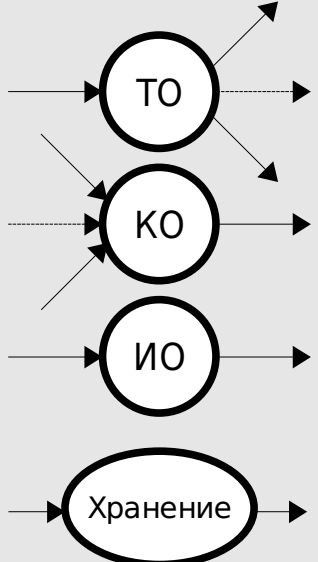
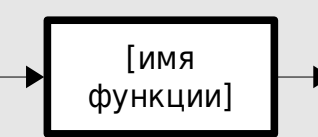
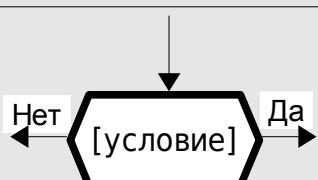

4. ЯЗЫКИ ОПИСАНИЯ ИСПОЛНИТЕЛЕЙ

4.1. Структурно-топологический язык

Графически описывает структуру задачи или произвольного исполнителя (коллектива людей, комплекса машин, человеко-машинной системы). Является инструментом структурно-топологического метода описания территориально рассредоточенных и/или функционально распределённых систем.

4.1.1. Алфавит языка

Алфавит образуют символы, перечисленные в таблице ниже:

Обозначение	Наименование, характеристика значения
	<p>СТ1. Узел обмена – реализует передачу потоков данных (вещества, энергии) следующих видов (см. сверху вниз):</p> <ul style="list-style-type: none"> • трансляционную – с ветвлением входящего потока (для данных – возможно, с расщеплением содержания); • коллекторную – с объединением входящего потока (для данных – возможно, с удалением дублирующих элементов); • индивидуальную (парную) – между двумя соседними по топологии узлами; • хранения — между узлами обмена с задержкой выдачи относительно приёма на любое конечное время, превышающее время прохождения потока между узлами.
	<p>СТ2. Функциональный узел (блок) – реализует преобразование входного потока в выходной, сосредоточенное в пространстве.</p>
	<p>СТ3. Решающий узел (блок) – реализует проверку условия и выбор пути в зависимости от того, выполнилось условие или нет. Необязателен в схемах исполнителей.</p>
	<p>Дуга – обозначает передачу обрабатываемых данных (вещества, энергии) или данных (сигналов) управления между соединяемыми узлами. Может указывать реальную конфигурацию пути передачи в пространстве.</p>

4.1.2. Правила языка

Схемы представляют собой оргграфы, причём рёбрами являются дуги, а остальные символы языка трактуются как вершины различных типов. Направление на диаграмме указывается стрелками дуг.

Схемы строятся по следующим правилам:

- размещение вершин и направление связей на схеме произвольно, но при описании реальной системы рекомендуется соотносить их с организацией системы на местности;
- допустимы пересечения дуг;

- блоки связаны друг с другом только через узлы обмена;
- дуга может иметь метку, определяющую передаваемое содержание.



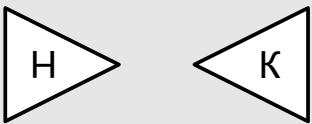


5. ЯЗЫКИ ОБОБЩЁННОЙ ФОРМАЛИЗАЦИИ

5.2. Язык синтаксических диаграмм

Графически описывает правила порождения множества (подмножества) текстов на некотором языке общения исполнителя задачи (коллектива людей, комплекса машин, человеко-машинной системы). Применяется для визуального задания языковых стандартов в программировании и др. областях формализации.

5.2.1. Алфавит языка

Алфавит образуют символы, перечисленные в таблице ниже:

Обозначение	Наименование, характеристика значения
	СД1,2 Начало, Конец – указывают исходную и конечную вершины диаграммы, с которых всякий раз начинается и заканчивается её чтение, т.е. порождение очередного текста.
	СД3 Жёсткое поле – служит для записи постоянного значения, т.е. подтекста, всякий раз входящего в текст, порождаемый при чтении диаграммы.
	СД4 Гибкое поле – служит для записи переменного значения; содержимое поля либо неформально описывает смысл текста, либо является именем уточняющей диаграммы, формально описывающей всё возможное содержание гибкого поля.
	СД5 Узел исходный – служит для разветвления линий на диаграмме; по смыслу является вершиной выбора дальнейшего пути.
	СД6 Узел конечный – служит для объединения линий на диаграмме; по смыслу является древесной точкой.
	Линия – соединяет поля друг с другом или поле с вершиной, указывая порядок чтения диаграммы. Разные формы линий (сверху вниз) применяются в разных конструкциях диаграммы, а именно: <ul style="list-style-type: none"> • двойная линия – присоединяет перед началом уточняющей диаграммы гибкое поле её имени; имеет смысл «<имя> описывается диаграммой». • продолжения строки и перехода на новую строку; • пути ветвления – простого (обхода) и составного (нагруженного полем или конструкцией из полей, узлов и линий); • петли цикла (простой и составной).

Вершины индексированы по типам (назначениям на диаграмме).

5.2.2. Правила языка

Синтаксические диаграммы представляют собой орграфы, причём рёбрами являются линии и двойные линии, а остальные символы являются вершинами. Направление на диаграмме указывается стрелками соединительных линий.

Диаграммы строятся по следующим правилам.

- диаграммы делятся на *обычные* и *уточняющие*, используемые для декомпозиции обычных;
- уточняющая диаграмма является детализацией синтаксиса некоторого гибкого поля другой диаграммы (обычной или уточняющей); допускается многоуровневое уточнение;
- обычная диаграмма состоит из начала, конца и содержания; уточняющая дополнительно содержит гибкое поле названия диаграммы, присоединённое двойной линией к началу слева;

- поле имеет один вход и один выход для линий; несколько входов и/или выходов у поля указывается через присоединённые конечный и/или исходный узлы;
- содержание диаграммы в простейшем случае составляет одно поле, соединённое линиями с началом и концом; более сложное содержание образуется композицией из базовых конструкций трёх видов: цепочки, ветвления и/или циклы (см. таблицу далее); в содержание входят также значения полей, т.е. одинаковые структуры с различием значения хотя бы одного поля считаются разными;
- содержание диаграммы на диосцене организуется по принципу текста, т.е. разбивается на строки полей и/или конструкций, составляемые и читаемые слева направо и сверху вниз;
- вход поля располагается на левой стороне контура (или верхней, когда поле является первым от границы диосцены), а выход на правой (или нижней, когда поле является последним); рекомендуется располагать вход посередине стороны;
- цепочка включает последовательность жёстких и/или гибких полей, соединённых линиями; поля располагаются слева направо, при достижении границы диосцены (рабочего листа) цепочка продолжается ниже под предыдущими строками;
- ветвление включает исходный узел, из которого начинаются два и более путей чтения диаграммы; далее по ходу диаграммы каждый путь завершается на к.-л. конечном узле вместе с путями, исходящими из того же самого или других исходных узлов; содержание пути может составлять линия, любое поле или конструкция; условие ветвления не формулируется, т.к. выбор направления при чтении предполагается произвольным (случайным);
- цикл образуется из цепочки конечного узла, поля (конструкции) и исходного узла, в которой вводятся один, два и более путей замыкания узлов (петли), направленные против направления в цепочке; содержание пути замыкания может составлять линия, любое поле или конструкция, причём содержание разных путей замыкания одной пары узлов д.б. различным; условие выхода из цикла не формулируется явно, т.к. число повторений цикла при чтении предполагается произвольным;
- допускаются цепочки из исходных и/или конечных узлов в любых сочетаниях;
- не допускаются пересечения линий вне вершин (устраняются введением цепочек узлов) и/или разрывы диаграммы на несколько диосцен (для устранения на сложной диаграмме выбираются фрагменты, которые можно рассматривать как поля, и каждый из них выносится как уточняющая диаграмма на отдельную диосцену);
- допускается включать в уточняющую диаграмму гибкое поле, одноимённое с именем самой диаграммы; тем самым реализуется рекурсивный принцип порождения текста;

Также используются следующие приёмы эргономизации диаграмм и порождаемых ими текстов:

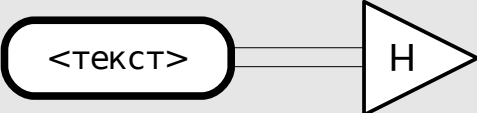


- изломы линий допускаются только под прямым углом;
- разные линии не должны накладываться друг на друга даже частично;
- возможно преобразование связей диаграммы по правилам шампур-метода;
- можно формулировать явно условия ветвления в конечных узлах; при этом графика узла заменяется на контур иконы Выбор (И5 в дракон-алфавите), повернутый на 90° (поворота можно избежать, применив иную организацию конструкций, см. п. 5.2.3);
- вход и выход поля должны лежать на одной оси (по шампуру); при стандартной ориентации синтдиаграмм эта ось горизонтальна;
- не следует переносить по строкам цепочки полей, содержащиеся в путях ветвления или цикла; рекомендуется выносить фрагменты таких цепочек как уточняющие диаграммы;
- можно не изображать стрелки у любых линий, кроме завершающих петли циклов, т.к. другие линии направлены в естественном (для европейских языков) порядке чтения текста;

- организация диаграммы как текста удобна при описании текстовых форм, реквизиты которых располагаются также построчно; в иных случаях диаграммы целесообразно располагать так, как на обычных схемах (структурных, принципиальных), т.е. не переносить цепочки полей/конструкций по строкам;
- вместо жёсткого поля, содержащего текстовый фрагмент для человека (напр. в составе машинного сообщения) можно сочинить ряд вариаций фрагмента, которые не меняют его общего смысла, и организовать их как ветвление на ряд жёстких полей.

5.2.3. Конструкции языка

5.2.3.1. Базовые структуры содержания

Аналогично любым маршрутным языкам (напр. схемам алгоритмов), для синтдиаграмм можно определить базовые (элементарные) структурные конструкции, как показано ниже:

Обозначение конструкции	Наименование, характеристика значения
	<p>1СД. Заголовок — указывает имя обычной или уточнённой синтдиаграммы.</p>
	<p>2СД. Ветвление (простое) — показывает выбор из двух возможностей. Условие ветвления (обозначено В) в общем случае м.б. любым, обеспечивая порождение текста, определяемого содержанием выбранной возможности.</p> <p>Одна из возможностей м.б. пустой (соответствующий путь от исходного узла к конечному не нагружен).</p>
	<p>3СД. Цикл (простой) — показывает повтор тела. Условие цикла в общем случае м.б. любым, обеспечивая порождение текста в виде цепочки произвольной длины, в которой повторяются пары из содержания предусловного (индекс Д на конце) и постусловного (П) подтел цикла.</p> <p>Либо предусловное, либо постусловное подтело цикла м.б. пустым (т.е. путь - ненагруженным).</p>

В таблице ниже приведены примеры сложных конструкций.

Видно, что для «лестничных» форм можно использовать явно-условные вершины И5 в стандартной ориентации (в цикле — с переносом побочного выхода), что упрощает их чтение.

Представляется очевидным, что с помощью вложенного цикла можно порождать любой текст в метаалфавите, образованном значениями блоков (конструкций) 1...N, если есть возможность ответить на сформулированный вопрос условия в каждом цикле.

Обозначение конструкции	Наименование, характеристика значения
	<p>4СД. Ветвление (сложное) — показывает выбор из N возможностей. Условия ветвления в общем случае м.б. любыми, обеспечивая порождение текста, определяемого содержанием выбранной возможности.</p> <p>В частном случае число возможностей равно двум, и ветвление будет простым.</p> <p>Одна из возможностей м.б. пустой (соответствующий путь от исходного узла к конечному не нагружен).</p> <p>Нижний вариант «лесенкой» применяется для удобства компоновки (выбрана организация, обеспечивающая порядок возможностей сверху вниз, как в верхнем варианте).</p>
	<p>5СД. Цикл (вложенный) — показывает перебор по N телам. В данном случае все циклы типа ПОКА.</p> <p>Условия каждого цикла в общем случае м.б. любыми, обеспечивая порождение текста в виде цепочки произвольной длины, в которой комбинируются ряды произвольного числа повторов содержания каждого тела текущей вложенности; ряды следуют в порядке вложения циклов (изнутри наружу).</p> <p>Вопрос цикла i-того вложения имеет смысл «Нужно ли в текущей позиции текста содержание i-того тела?»; ответ «нет» на вопрос N ведёт к выходу из конструкции, а на любой другой — к условию следующего уровня.</p> <p>Нижний вариант «лесенкой» применяется для удобства компоновки (выбрана организация, обеспечивающая порядок вложения сверху вниз, как в верхнем варианте).</p>

Аналогично конструкции 5СД можно определить вложение циклов ДО (постусловных); при этом поля как бы перемещаются на один маршрут вниз, и пустой маршрут оказывается наверху (в цикле N). Чтение такой конструкции отличается тем, что начинается с порождения содержания из вершины 1П, а не с проверки условия; т.о. это содержание будет входить в результирующий текст безусловно.

Очевидным образом можно преобразовать конструкции 2СД и 3СД в «лестничную» форму; результат показан в следующей таблице.

Одновременно конструкции даны информатизованными (с условными вершинами-развилками типа И5).

Обозначение конструкции	Наименование, характеристика значения
	<p>2СДи. Ветвление (простое информатизованное) — явная (алгоритмичная) форма конструкции. Условие ветвления имеет конкретную формулировку.</p> <p>Одна из возможностей м.б. пустой (соответствующий путь от исходного узла к конечному не нагружен).</p>
	<p>3СДи. Цикл (простой информатизованный) — явная (алгоритмичная) форма конструкции. Условие цикла имеет конкретную формулировку.</p> <p>Либо предусловное, либо постусловное подтело цикла м.б. пустым (т.е. путь — ненагруженным).</p>

По такому же принципу можно информатизовать сложные конструкции.

5.2.3.2. Диоформы синтаксических диаграмм

Простейшей формой синтдиаграммы на плоскости служит цепочка, как показано в таблице:

Обозначение конструкции	Наименование, характеристика значения
	<p>ПСД. Цепочка — примитивная форма с простым следованием от начала к концу (здесь не показаны).</p> <p>В общем случае всё содержание диаграммы рассматривается как цепочка полей (и/или конструкций).</p> <p>Если физическая протяжённость цепочки превышает ширину диосцены, то она м.б. перенесена подобно строке текста.</p>