

Главная страница	
Рабочие материалы	
Литература	
Теоретическая информатика	
Семиотика	
(Многоаспектная семантика)	04.10.16

Многоаспектная семантика полиморфного языка

Статья. Библиографические данные:

<p>Житников А.П. Концепция многоаспектной семантики полиморфного языка описания и моделирования параллельных алгоритмов / Принятие решений в условиях неопределенности: Межвуз. научн. сб. – Уфа, 2003. С. 113 – 121.</p>

Это обработанная копия оригинала статьи:
добавлены содержание статьи, два рисунка, нумерация рисунков и т.п.

СОДЕРЖАНИЕ

Полиморфный язык описания параллельных алгоритмов	2
Знаковые формы текстов алгоритмов.....	2
Семантические причинные цели полиморфизма	3
Уточнение концепции семантического треугольника	4
Расщепление общего отношения обозначения.....	5
Содержание и форма комплексов обозначения.....	6
Представительность объектов (явлений)	7
Компоненты состава концепта (описания)	8
Прямое структурное моделирование	8
Многоаспектная система интерпретации	10
ЛИТЕРАТУРА.....	11

КОНЦЕПЦИЯ МНОГОАСПЕКТНОЙ СЕМАНТИКИ ПОЛИМОРФНОГО ЯЗЫКА ОПИСАНИЯ И МОДЕЛИРОВАНИЯ ПАРАЛЛЕЛЬНЫХ АЛГОРИТМОВ

Кратко излагаются принципы формирования многоуровневой многоаспектной системы семантики для комплекса сопряженных языков описания и моделирования параллельных алгоритмов в разных визуальных и синтаксических формах и лексических системах.

Ключевые слова: параллельный алгоритм, алгоритмический полюсник, алгоритмический язык, полиморфный синтаксис, многоаспектная семантика, семантический треугольник, денотат, концепт.

Полиморфный язык описания параллельных алгоритмов

Для инженерных приложений алгоритмов недостаточно применение отдельных методов и средств их описания. Необходимо широкое разнообразие форм их представления разного назначения (широкий полиморфизм) при обеспечении их конструктивной взаимосвязи в рамках комплексных системных концепций.

Разрабатывается и проходит поэтапную концептуальную отладку интегрированный (по конструктивным средствам) полиморфный язык структурно-функционального описания основных классов параллельных алгоритмов и систем их реализации (материализации).

В работе [1] отражаются исходные принципы синтаксиса полиморфного алгоритмического языка для описания и моделирования параллельных алгоритмов и техники согласованных полиморфных алгоритмических построений с применением их программной поддержки (макетные программные разработки).

Далее излагаются исходные аспекты формирования семантики полиморфного языка, в основу построения которой принимаются общие концепции семиотики – теории знаков и знаковых систем [2-3].

Знаковые формы текстов алгоритмов

Используются одномерные и двухмерные обобщенные *тексты алгоритмов* (ТА) – объединенные общим смыслом знаковые комплексы типа процедурных и объектных предписаний. Синтаксисом поэтапно определяются следующие согласованные формы записи текстов параллельных алгоритмов (разных видов и структурных классов, [Рис. 1](#)).

1) ЛТА: *Литерные* тексты алгоритмов:

ФТА: *Формульные* тексты алгоритмов – структурные и функциональные формулы (компактные тексты). Синтаксические формы: инфиксная, префиксная, постфиксная и комбинированные формы.

ВТА: **Вербальные** (словесные) тексты алгоритмов – взаимно обратимые аналоги подмножеств языков программирования разных синтаксических форм и лексических групп (на основе ФТА как шаблонов): алгол-подобные, си-подобные, оккам-подобные тексты и др.

2) ГТА: **Графические** тексты алгоритмов (обычно двухмерные):

ССА: **Структурные** схемы алгоритмов – блок-схемы, штрих-схемы, а также граф-схемы (пока ограниченно).

ДИА: Линейные и сетевые временные **диаграммы** исполнения алгоритмов – структурные и метрические (масштабируемые) диаграммы.

3) ТТА: **Табличные** тексты описания алгоритмов (ограниченно).

Используются разные формы исполнения описаний алгоритмов:

- ГИ: Горизонтальное исполнение и ВИ: Вертикальное исполнение.
- ПФ: Прямая форма и ОФ: Обратная форма (обратная запись).

Для обеспечения необходимой гибкости построений задаются:

- исходные **опорные** взаимно обратимые формы описаний;
- любые **производные** формы – в четкой связи с общей основой: упрощения и дополнения, видоизменения и адаптация к различным условиям и средствам, разным стандартам и традициям и т.п.

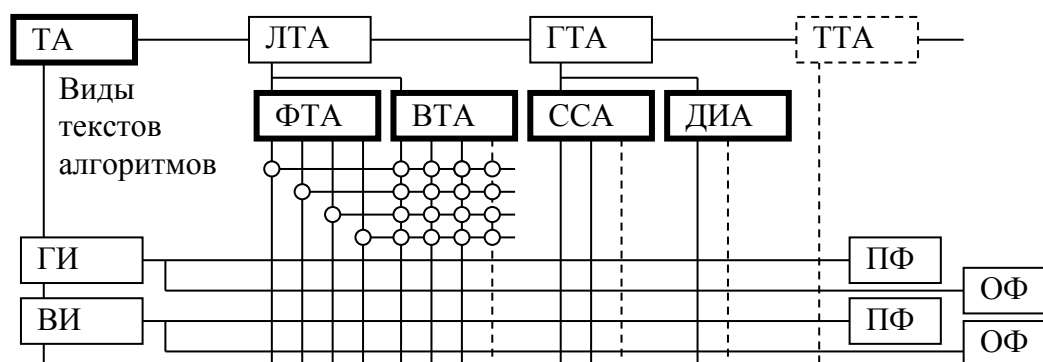


Рис. 1. Знаковые формы представления алгоритмов

Семантические причинные цели полиморфизма

Целью полиморфных построений не является просто охват разных видов, форм и многочисленных вариантов представлений алгоритмов, которые вводятся и используются произвольно по соображениям разных потребностей и удобств, традиций, субъективных предпочтений и т.п. (гибкий дружественный сервис). Полиморфизм является необходимостью по принципиальным причинным обстоятельствам:

1) Конструктивное описание семантики – это текст, который имеет некоторый синтаксис. Необходима **многоуровневая система** синтаксических построений с последовательным уточнением семантики: каждый последующий уровень синтаксиса отражает некоторую семантику синтаксиса предыдущего уровня.

2) Разные виды и формы текстов, кроме некоторой общей синтаксической информации, отражают разные аспекты семантики. При этом необходима не просто последовательная цепь по уровням синтаксиса и их семан-

тики, а **многомерная сеть** синтаксических построений с разветвлением и пересечением различных направлений по различным аспектам многоаспектной общей семантики.

Уточнение концепции семантического треугольника

Анализ вопроса выявляет исходную общую проблему семантики: конструктивный анализ и развитие концептуальной схемы семантического треугольника в концепции двойного значения знака в семиотике [2, 3] для исключения неясностей и разнотолков его интерпретации, проблем применения, критики двойного значения знака [4] и т.п. (Рис. 2, Рис. 3)

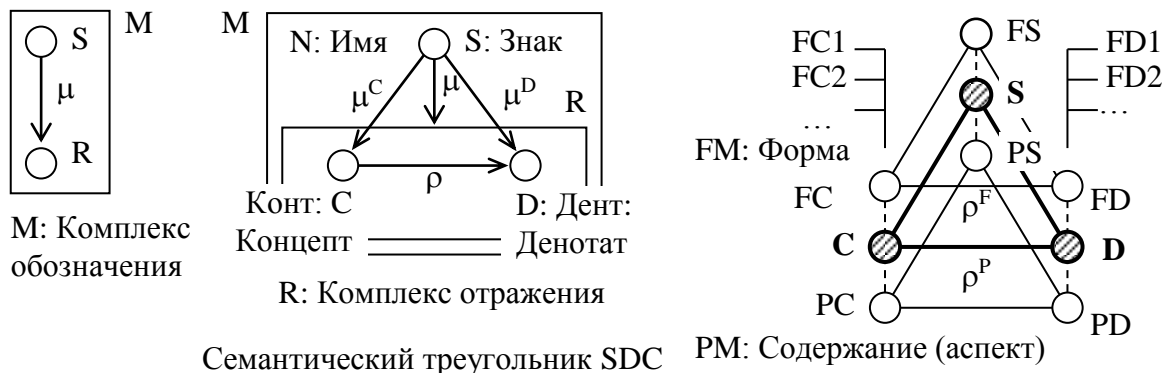


Рис. 2. Семантический треугольник и сопутствующие конфигурации

Вводятся исходные категории связи различных явлений (сущностей) – объектов, процессов, свойств, отношений в составе информационных процессов и систем (в их знаковой форме):

$M = (S/N \mu R)$: **Комплекс обозначения** с отношением обозначения μ : знак S (в частности, имя N) обозначает (μ) комплекс отражения R .

S : **Знак** или, в частности, $S = N$: **Имя** (словесный знак устной или письменной речи – слово, словосочетание, термин, идентификатор и т.п.) – это обозначающий объект (обозначающее явление);

например: имя дискретного процесса или реализующего его объекта.

$R = (C \rho D)$: **Комплекс отражения** с отношением отражения ρ : концепт (описание) или конт C отражает (ρ) денотат или дент D .

Комплекс (информационного) отражения R – это одно общее (денто-контное) значение знака S (концепция одного значения) с двумя его частными значениями D и C (концепция двух значений):

D : **Дент** (денотат) – обозначаемый отражаемый объект (явление):

дентовое частное значение знака; например, обозначаемый знаком (именем) отражаемый в некотором описании процесс или объект.

C : **Конт** (концепт) – обозначаемый отражающий объект (явление):

контное частное значение знака – понятие, концепция дента и т.п.; например: обозначаемый знаком (именем) алгоритмический (или, в частности, программный) текст описания процесса или объекта.

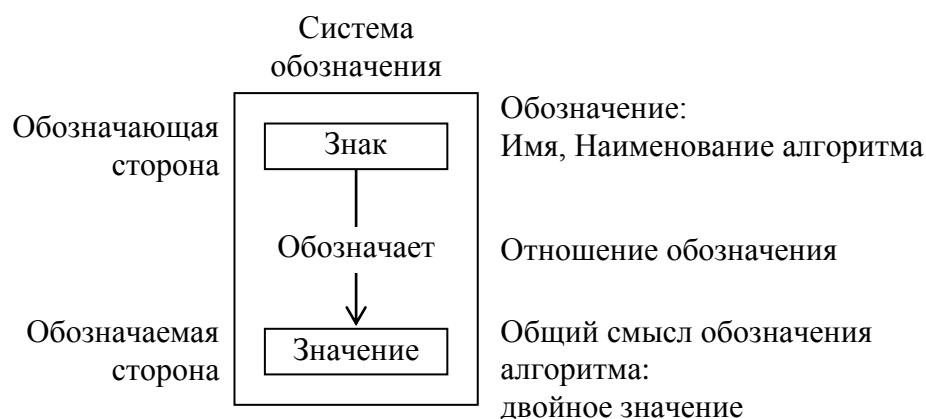


Рис. 3. Первичная схема системы обозначения для некоторого алгоритма

Расщепление общего отношения обозначения

Детализация записи отношения обозначения:

$M = (S \mu R) = (S \mu (C \rho D)) = ((S \mu^C C) \rho (S \mu^D D))$ – частные отношения обозначения μ^D и μ^C : общий знак S обозначает (μ^D) отражаемый дент D и, одновременно, обозначает (μ^C) отражающий конт C .

Формируется **семантический треугольник SDC**: знак (имя), денотат (дент), концепт (конт) – классическая фигура семиотики. Удобные краткие термины *дент* и *конт* вводятся по аналогии с работой [2, 3].

Принимается интерпретация: двойное значение знака – это двойственный общий смысл, то есть двойственная общая семантика знака.

Знак $S = N$ (имя, наименование) некоторой сущности имеет два значения (двойное значение):

D: Дент: Обозначаемая именем отражаемая сущность (объект, процесс, признак – свойство или отношение и т.п.) некоторого типа (класса).

C: Конт: Обозначаемый именем отражающий объект, представляющие его описание.

Например, имя алгоритма (программы) имеет два значения:

C: Конт: Обозначаемый именем текст алгоритма (программы), предопределяющий (и отражающий) управляемый некоторый вычислительный процесс (класс процессов в общем случае).

D: Дент: Обозначаемый именем вычислительный процесс (некоторого класса), предопределяемый (и отражаемый) текстом алгоритма (программы).

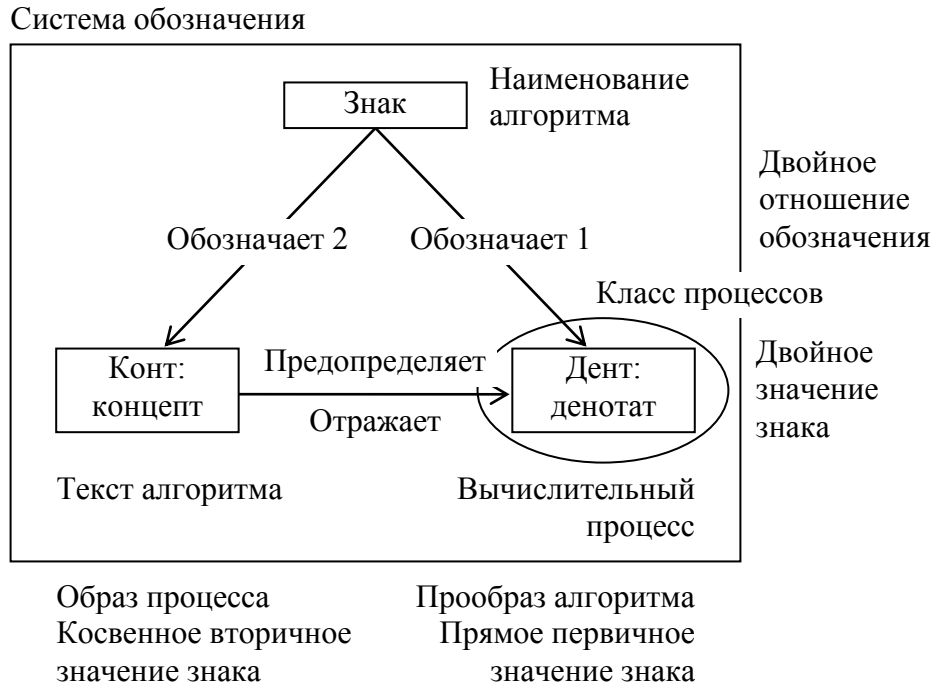


Рис. 4. Вторичная схема системы обозначения алгоритма. Семантический треугольник

Содержание и форма комплексов обозначения

Вводится концепция содержания и формы объектов (явлений):

P: **Содержание** (аспект) объекта (явления) – комплекс признаков объекта (явления), проявляемый в некотором определенном отношении, некоторая точка зрения на объект (явление) и то, что видимо из нее.

F: **Форма** организации объекта (явления), его содержания.

Основные соотношения:

$$PM = P(S \mu R) = PS \mu^P P(C \rho D) = PS \mu^P (PC \rho^P PD)$$

$$FM = F(S \mu R) = FS \mu^F F(C \rho D) = FS \mu^F (FC \rho^F FD)$$

1) $PC \rho^P PD$: **Содержание** PC (определенный комплекс признаков) **конта** C, то есть **информация** в составе описания отражает (ρ^P) **содержание** PD (определенный комплекс признаков) **дента** D.

Например. Аспекты конструкции, физических принципов, обобщенной структуры устройства и процессов работы автомата, его переменные величины, рабочие функции и отношения отражаются соответствующим информационным содержанием его описания.

2) $FC \rho^F FD$: Общая **форма** FC **конта** (описания) имеет, в общем случае разные частные формы FC1, FC2 и т.д., и, при этом:

а) имеет разные **целевые формы**, которые отражают (ρ^F) специфику **форм реализации дента** – FD1, FD2 и т.д.; например:

- формы реализации алгоритмов – аппаратная (машинная), программная, организационная и комбинированная реализации предполагают разные формы их описания (и задачу системных обобщений);
- формы машинной реализации устройства и работы некоторого автомата

(механика, электрика, гидравлика, пневматика) представлены разными стандартными формами документации;

б) имеет разные **знаковые формы**; например: соответственные и взаимно-обратимые виды формул, схем, вербальных текстов, табличных форм – содержат общую целевую информацию, но имеют разные прочие полезные свойства и используются с разными целями.

Информационное содержание РС контя (описания) С всегда представлено в некоторой **знаковой форме** FC.

Знак S (имя N – слово, словосочетание, термин) имеет некоторую (знаковую) форму FS и, в общем случае, некоторое собственное содержание PS, например: слова "треугольник" или "треугольничек". В частности, это содержание пустое, например, для знаков A, B, C.

Представительность объектов (явлений)

Вводится концепция представительности (замены) дента (Рис. 5).

$M = (I \pi D) = ((S, C) \pi D)$ – комплекс отражения M (семантический треугольник) переопределяется как **комплекс представления**:

представляющий комплекс I = (S, C): знак (имя) и конт (описание) представляет (π) и, в частности, заменяет **представляемый** дент D, то есть обозначаемый отражаемый объект (явление) в составе информационных процессов и систем (реализуемых в знаковой форме).

Например. Программный объект D, расположенный в оперативной памяти, представлен (заменяется) в программной документации именем S = N и текстом описания программы, то есть контом C.

Разные функции отношения представительности:

$M = ((S, C) \pi D) = ((S \pi^S D), (C \pi^C D))$ – расщепление отношения;

$(S \pi^S D)$ – знак S представляет (π^S) дент D в целом;

$(C \pi^C D)$ – конт C (описание) представляет (π^C) дент D в некоторых деталях (соответственно некоторому аспекту PD в комплексе M).

Уточнение для представительности дента контом (описанием):

$\pi^C = \pi' \pi'' = \pi^{PD} \pi''$ – состав транзитивного отношения;

$(C \pi^C D) = (C \pi^{PD} \pi'' D) = (C \pi^{PD} PD \pi'' D) = ((C \pi^{PD} PD) \pi'' D)$,

$(C \pi^{PD} PD)$: конт C представляет (π^{PD}) некоторый аспект PD дента D.

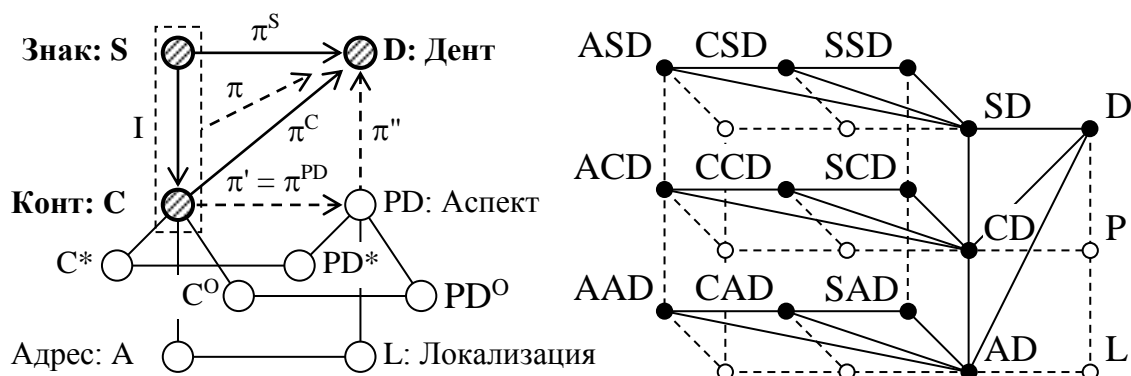


Рис. 5. Расширения концепции семантического треугольника

Компоненты состава концепта (описания)

В составе отражаемого аспекта PD и отражающего его конт С могут выделяться различные типовые компоненты (концепция множественности составляющих значений знака, Рис. 5):

1)) В частности, выделяются типовые компоненты [2, 3]:

а) Дент может иметь некоторую *локализацию* L в пространстве-времени. В составе конта этому соответствует определенный пространственно-временной *адрес* A в некоторых координатах.

б) В общем случае конт С = (С*, С⁰) включает две компоненты:
 С*: **Внешний** конт – например, описание интерфейса подпрограммы списком входных (in) и выходных (out) параметров и их типов: внешнее описание алгоритмического полюсника (черный ящик);
 С⁰: **Внутренний** конт – тело описания подпрограммы "begin ... end": внутреннее описание алгоритмического полюсника (белый ящик).

2) Возможна визуализация простого исчисления комбинированных объектов произвольной длины и их точная интерпретация:

- двухмерная фигура: SD, CD, AD – знак, конт, адрес дента D;
- трехмерная фигура: SAD – имя адреса дента (адресной переменной или указателя дента), CSD – конт (описание) имени дента, AAD и т.д.;
- четырехмерная гиперфигура: объекты ACSD, AAAD, CCCD и т.п.

3) Имена и указатели объектов (объекты данных, процедур и т.п.):

D = xxxx – алгоритмический (например, программный) объект.

ND = nnnn – имя (знак S = N) объекта D.

AD = aaaa – адрес A объекта D или адресная переменная типа указателя (местоимение типа "тот, который там" – по месту A → L).

NAD = rrrr – имя N адреса A (адресного указателя) объекта D.

N'D = *rrrr / c = rrrr^ / pas – псевдо-имя N' объекта D.

N'AD = &nnnn / c = @nnnn / pas – псевдо-имя N' адреса A объекта D.

ND ~ N'D, NAD ~ N'AD – сходные сущности с разными свойствами.

Составляющие модельные объекты:

D, ND, N'D, AD, NAD, N'AD – оболочковые объекты (контейнеры), представляющие собой поля для размещения знаковых комплексов;

xxxx, nnnn, *rrrr; aaaa, rrrr, &nnnn – объекты наполнения оболочек.

Для объектов D и AD возможны концепты (описания типов и т.д.).

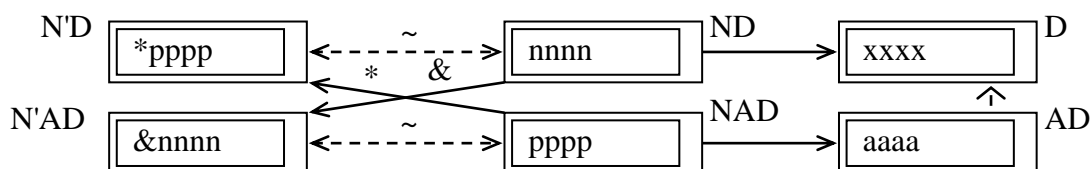


Рис. 6. Имена и указатели объектов

Прямое структурное моделирование

Изложенные аспекты являются основой формирования конструктивной системы семантики для полиморфного языка параллельных алгорит-

мов и систем их (аппаратной, программной, организационной и комбинированной) реализации. Далее в основу семантики полиморфного языка принимается взаимно-однозначное соответствие структуры параллельных процессов, реализующих их объектов и предопределяющих их алгоритмов – взаимное прямое структурное моделирование процессов, объектов и алгоритмов, включая компоненты ():

У: **Поток управления** – цепи (сети) передачи управления;

Д: **Поток данных** – цепи (сети) обработки данных.

Соотношения изоморфизма частных операционных сущностей:

$O'' \sim O'$ – структура реализации изоморфна структуре процесса;

$O''' \sim O'$, $O''' \sim O''$ – структуры процедурных и объектных описаний изоморфны структурам процессов и объектов соответственно.

$O' \sim O'' \sim O''' \sim O'''' \Leftrightarrow O$ – все виды взаимно изоморфных операционных сущностей отображаются на единое унифицированное описание O обобщенной операционной сущности.

Обобщенная операционная сущность представлена структурными формулами и структурными схемами и может интерпретироваться как описание любого типа частной операционной сущности.

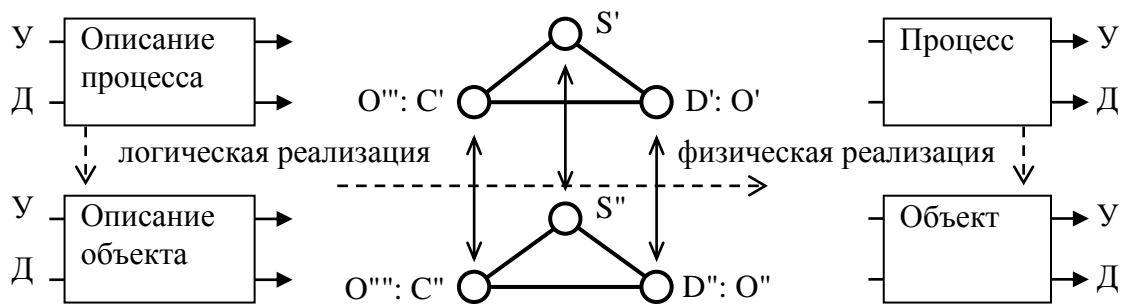


Рис. 7. Семантические треугольники процессной и объектной реализации алгоритма

Алгоритм $O''' = (A :: B)$ – это операционное предписание операционному исполнителю O'' выполнить комплекс операций O' (дискретный процесс) в определенном (последовательном или параллельном) порядке в соответствии с определенными функциями [1] (Рис. 8):

$$O = (A :: B) = ((A: Y_a = F_a(X_a)) :: (B: Y_b = F_b(X_b)))$$

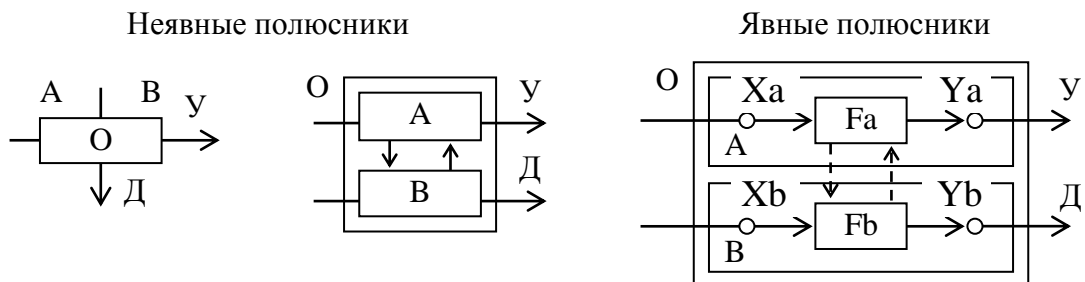


Рис. 8. Полюсники потока управления и потока данных

Процедурное предписание O''' исполнителю изоморфно интерпретируется как объектное предписание $O'''' \sim O'''$ пред-исполнителю (например,

транслятору) подготовить исполнителя O'' как реализацию алгоритма для последующего многократного его исполнения O' : это включает (в терминах [2]) создание статусных объектов (объектов переменных данных) X , Y и функциональных объектов F .

Многоаспектная система интерпретации

На основе изложенных концепций принимается многоаспектная сеть семантики полиморфного языка, включая:

- уровни интерпретации;
- типы интерпретации разных форм по уровням.

Выделяются уровни интерпретации (Рис. 8):

- **неявные** алгоритмические **полюсники** (обобщенные операторы A , B без отражения переменных в составе потоков) – основа методов описания общей операционной структуры объектов, процессов и алгоритмов управления; это исходные объекты интерпретации;
- **явные** алгоритмические **полюсники** – детализация операционных объектов (переменные X , Y функциональные и преобразователи F): подготовка функциональной семантики описания структур;
- **функции** явных структурных полюсников – функциональная семантика $y = F(x)$ описания работы полюсников (во времени).

Каждый последующий уровень интерпретирует предыдущий. На каждом уровне выделяются группы интерпретации разных форм:

ФТА: Формульные тексты – исходный объект интерпретации;

ССА: Структурные схемы, включая схемы виртуальной аппаратной реализации (материализации) алгоритмов – основа решения концептуальных проблем понимания и объяснения (модельных) механизмов действия алгоритмов и их строгого функционального обоснования;

ДИА: Временные диаграммы – представления процессов во времени;

ВТА: Вербальные тексты алгоритмов – программно подобные описания и расширенные вербальные описания (детальная интерпретация).

Исходная многомерная сеть многоаспектной семантики условно сводится к упрощенной плоской (двухмерной) схеме для ее последующего наполнения и развертывания по разным аспектам:

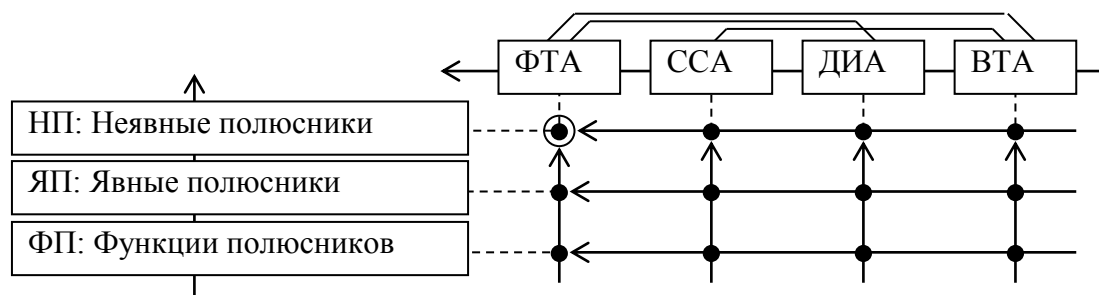


Рис. 9. Уровни интерпретации и знаковые формы представления алгоритмов

ЛИТЕРАТУРА

1. Житников В.П., Шерыхалина Н.М., Житников А.П. Программно-методический комплекс "Параллельные алгоритмы и программы". // Материалы научно-технического совещания "Высокопроизводительные вычислительные ресурсы России: состояние и перспективы развития". – Уфа: УГАТУ, 2003. С. 149 - 159.
2. Зверев Г.Н. Основания теоретической информатики. 1. Элементы системологии. – Уфа: УГАТУ, 1995. 61 с.
3. Зверев Г.Н. Основания теоретической информатики. 2. Теоретическая семиотика и лингвистика. – Уфа: УГАТУ, 1995. 31с.
4. Переверзев В.Н. Логистика. Справочная книга по логике. – М.: Мысль, 1995.