

1. Графодинамическая парадигма обработки информации

Основной задачей данного раздела является определение графодинамической парадигмы обработки информации как наиболее перспективной основы для создания интеллектуальных систем нового поколения, в которых должны поддерживаться сложноструктурированные базы знаний большого объема, сложные стратегии и механизмы решения трудноформализуемых задач, высокий уровень гибкости используемых стратегий и механизмов решения задач, высокий уровень интегрируемости различных моделей переработки знаний, асинхронизм и высокий уровень параллелизма. Кроме того, здесь же осуществляется уточнение основных понятий, связанных с графодинамической парадигмой и уточнение предлагаемого подхода к созданию комплекса средств, ориентированных на поддержку интеллектуальных систем нового поколения.

Данный раздел может быть использован в качестве учебного пособия по дисциплине «Инструментальные средства и технологии проектирования интеллектуальных систем» для студентов специальности «Искусственный интеллект».

1.1. Концепция графодинамических моделей

Ключевые понятия: графодинамические параллельные асинхронные модели; формальная модель обработки информации; интеграция; интерпретация.

Одной из характерных особенностей современного состояния информатики является наличие огромного многообразия способов, парадигм, принципов, методов организации обработки информации. Основным источником такого многообразия на сегодняшний день являются работы в области искусственного интеллекта. При этом следует подчеркнуть, что многообразие способов организации обработки информации находит отражение не только во множестве языков программирования, во множестве программных инструментальных средств проектирования интеллектуальных систем, но и в аппаратных средствах, в архитектурах компьютеров новых поколений.

Для того чтобы разобраться во всем многообразии способов организации обработки информации, необходимо, во-первых, иметь возможность абстрагироваться от несущественных деталей реализации и, во-вторых, иметь возможность так трактовать сравниваемые способы обработки информации, чтобы были исключены их несущественные отличия, несущественные особенности.

Таким образом, важнейшим подходом к исследованию различных способов организации обработки информации в современных системах является исследование соответствующих формальных (абстрактных) моделей.

Основными особенностями рассматриваемых в данной книге моделей являются:

- графодинамический характер, т.е. представление перерабатываемой информации в виде нелинейных конструкций и структурная перестраиваемость системы связей между элементами памяти (абстрактной или аппаратно реализованной) в ходе переработки хранимых в этой памяти информационных конструкций;
- ассоциативность, т.е. использование ассоциативного метода доступа к требуемым фрагментам перерабатываемой информации;
- параллелизм;
- асинхронность;
- возможность обработки информации с помощью операций высокого уровня.

Модели обработки информации указанного класса будем называть **графодинамическими параллельными асинхронными моделями**. Нетрудно заметить, что перечисленные выше свойства этих моделей тесно связаны друг с другом и являются полной противоположностью свойствам классических фон-Неймановских моделей обработки информации, лежащих в основе семейства традиционных компьютеров. К числу свойств классических моделей относятся:

- 1) линейный характер перерабатываемых информационных конструкций, линейный характер связи между элементами памяти и структурно фиксированный характер изменения состояния памяти;
- 2) адресный метод доступа к требуемым фрагментам перерабатываемой информации, в основе которого лежит фиксированное, т.е. неизменяемое в процессе обработки информации, взаимно однозначное соответствие между множеством элементов памяти и множеством их имен (идентифи-

- каторов), – такие имена (в качестве которых обычно используются натуральные числа) называются адресами элементов памяти;
- 3) последовательный характер переработки;
 - 4) синхронный, т.е. основанный на централизованном управлении, характер организации обработки информации;
 - 5) низкий уровень операций обработки информации.

Подробнее о перечисленных свойствах классических моделей обработки информации см. в работе [129] (*Глушков В.М..1974пр-РекурМиВТ*).

Термин "графодинамика" заимствован нами из работы [10] (*Айзерман М.А..1988ст-ДинамПКАС*).

Известными примерами графодинамических моделей обработки информации, в частности, являются алгоритмы Колмогорова [259] (*Колмогоров А.Н..1958ст-кОпредА*), вегетативная машина Борщева [64] (*Борщев В.Б.1989ст-ВегетМ*), всевозможные графовые грамматики [379; 376; 377; 378] (*Петров С.В..1978ст-ГрафоГиА* ; *Петров С.В..1977ст-ГрафоГиЗГ* ; *Петров С.В..1977ст-НормаФГГ* ; *Петров С.В..1977ст-СтандФГГ*).

1.1.1. Понятие формальной модели обработки информации

Ключевые понятия: формальная модель обработки информации, первичные элементы, синтаксические правила, начальная информационная конструкция, операции формальной модели, формальная модель, язык, абстрактная машина.

Приведем классическое определение формальной модели.

Определение 1.1.1.1. [390; 392; 401] (*Поспелов Д.А.1981кн-ЛогикЛМ* ; *Поспелов Д.А.1986кн-СитуаУ* ; *Поспелов Д.А.ред.1990спр-ИскусИ-К2*). **Формальная модель обработки информации F** , называемая также формальной системой, исчислением, определяет процесс переработки информационной конструкции и задается четверкой:

$$F = \langle T, P, S, W \rangle,$$

где

T – множество **первичных элементов** (терминальных элементов, базовых элементов, элементарных конструкций) формальной модели;

P – множество **синтаксических правил** формальной модели, которые определяют множество синтаксически правильных (правильно построенных) информационных конструкций (конструктивных объектов), перерабатываемых в рамках данной формальной модели;

S – **начальная (исходная) информационная конструкция** формальной модели, т.е. начальное состояние перерабатываемой информационной конструкции, которое иногда называют совокупностью аксиом;

W – множество **операций формальной модели**, т.е. правил построения новых информационных конструкций из уже построенных, правил преобразования (модификации) текущего состояния перерабатываемой информационной конструкции.

Операции формальной модели иногда называют правилами вывода, которые не следует отождествлять с правилами логического вывода, поскольку формальные модели могут быть не только логическими.

Нетрудно заметить, что понятие формальной модели включает в себя три аспекта:

- представление (кодирование) информации в виде некоторых информационных конструкций, устройство этих информационных конструкций, их соотношение с описываемой предметной областью, т.е. устройство языка, используемого формальной моделью, его синтаксис и семантика;
- построение начальной информационной конструкции формальной модели, представляющей собой исходное описание некоторой конкретной предметной области. Для формальной модели переработки знаний, отражающей логико-семантический уровень интеллектуальной системы, начальная информационная конструкция называется исходным состоянием базы знаний;
- организация переработки информационных конструкций. Таким образом, наряду с приведенным выше определением формальной модели можно дать следующее эквивалентное определение.

Определение 1.1.1.2. Формальная модель F задается тройкой $F = \langle L, S, C \rangle$, где

L – язык формальной модели F с присущими ему синтаксисом и семантикой;

S – начальная информационная конструкция формальной модели F , которая должна принадлежать языку L ;

C – абстрактная машина обработки информации, определяющая операции (правила преобразования) конструкций языка L . Множество операций абстрактной машины C в точности соответствует множеству операций формальной модели, которое в определении 1.1.1.1 обозначено символом W .

Формальная модель рассматривает процесс обработки информации как процесс преобразования информационной конструкции, хранимой в памяти абстрактной машины. Следовательно, текущее состояние такого процесса полностью определяется текущим состоянием перерабатываемой информационной конструкции, т.е. текущим состоянием памяти абстрактной машины.

Приведенное определение формальной модели обработки информации условно разбивает модель обработки информации на модель представления информации и модель преобразования информации (модель манипулирования информационными конструкциями).

Язык L определяется множеством информационных конструкций, которое называется множеством синтаксически правильных (правильно построенных) конструкций этого языка. Описание синтаксиса языка формальной модели должно быть конструктивным определением множества синтаксически правильных его конструкций, которое соответственно задается а) множеством первичных элементов (базовых элементов, элементарных, атомарных конструкций) языка и б) множеством синтаксических правил (см. определение 1.1.1.1). То, как соотносится произвольная синтаксически правильная конструкция этого языка с фрагментом предметной области, описываемой этой конструкцией, будем называть денотационной семантикой языка, а соотношение конкретной конструкции языка с описываемым этой конструкцией фрагментом предметной области будем называть денотационной семантикой указанной конструкции.

Абстрактная машина C задается а) абстрактной памятью (абстрактной запоминающей средой), в которой хранятся перерабатываемые информационные конструкции, и б) множеством операций. Текущее состояние абстрактной памяти представляет собой текущее состояние перерабатываемой информационной конструкции. В этом смысле абстрактная память есть нестационарная (динамическая, изменяющаяся во времени) информационная конструкция. Структура памяти абстрактной машины, ее "статические" свойства определяются синтаксисом языка L . Принципы изменения состояния памяти абстрактной машины, т.е. "динамические" свойства хранимых в памяти информационных конструкций, характер преобразования информационных конструкций определяются операциями абстрактной машины. На одной и той же абстрактной машине могут быть реализованы разные формальные модели, отличающиеся друг от друга разными начальными информационными конструкциями, которые задают разное исходное состояние памяти абстрактной машины. Таким образом, каждому сочетанию абстрактной машины и языка соответствует целое семейство формальных моделей, использующих указанный язык и реализуемых на указанной абстрактной машине. Могут существовать формальные модели, отличающиеся разными начальными информационными конструкциями, разными языками, но имеющие одинаковые операции. Такие формальные модели также могут быть реализованы на одной и той же абстрактной машине. Могут существовать формальные модели, отличающиеся разными начальными информационными конструкциями, разным набором операций, но имеющие одинаковые языки.

Язык, которому однозначно ставится в соответствие набор операций, т.е. определенная абстрактная машина, будем называть языком с фиксированной операционной семантикой. Операционная семантика такого языка задается соответствующей абстрактной машиной. Все остальные языки будем называть языками с нефиксированной операционной семантикой. Языками с фиксированной (четко заданной) операционной семантикой являются все языки программирования. В отличие от этого языки представления знаний могут иметь нефиксированную операционную семантику. Это означает, что одному и тому же языку представления знаний могут быть поставлены в соответствие разные методы решения задач в рамках этого языка.

Рассматриваемая нами трактовка формальной модели дает возможность четко выделить три этапа разработки конкретных формальных моделей:

- разработка языка (языка программирования или языка представления знаний);
- разработка абстрактной машины (машины реализации хранимых программ или машины переработки знаний);
- разработка начальной информационной конструкции начального состояния памяти абстрактной машины (конкретной программы вместе с ее конкретными исходными данными или исходного состояния некоторой базы знаний).

Таким образом, рассматриваемая трактовка формальной модели дает возможность явно связать формальную модель с главными компонентами инструментальных средств, обеспечивающих ее реализацию, – с соответствующим языком и с соответствующей абстрактной машиной.

Кроме того, используемое понятие абстрактной машины дает возможность с общих позиций рассмотреть принципы организации обработки информации в формальных моделях самого различного вида, а также дает возможность исследовать новые архитектуры компьютеров следующих поколений, в частности, компьютеров, ориентированных на реализацию интеллектуальных систем.

1.1.2. Классификация формальных моделей обработки информации

Ключевые понятия: формальная модель реализации хранимых программ, формальная модель переработки знаний, формальная модель переработки данных, формальные модели реализации хранимых процедурных программ, формальные модели реализации хранимых непроцедурных программ, формальная модель переработки символьных конструкций, формальная модель переработки графовых конструкций, графодинамические формальные модели.

Первым признаком классификации формальных моделей обработки информации является тип денотационной семантики начальных информационных конструкций.

Если начальная информационная конструкция формальной модели включает в себя программы, описывающие методы решения некоторых классов задач, и исходные данные для ряда конкретных задач из указанных классов, а операциями формальной модели являются операции реализации таких программ, то соответствующую формальную модель будем называть **формальной моделью реализации хранимых программ**.

Если начальная информационная конструкция формальной модели включает в себя фактографические (экстенциональные) знания, относящиеся к некоторой предметной области, и знания о свойствах и законах этой предметной области (интенциональные знания) [242] (*Кандрашина Е. Ю..1989кн-Пред3оВиП*), а операциями формальной модели являются операции переработки знаний, то соответствующую формальную модель будем называть **формальной моделью переработки знаний**. Такие операции могут поддерживать самые различные стратегии решения задач (стратегии рассуждений) – классические и неклассические, дедуктивные и индуктивные, четкие и нечеткие, строгие и правдоподобные. Частным видом формальных моделей переработки знаний являются формальные модели логического вывода.

Если в начальной информационной конструкции нельзя выделить ни фактографические и интенциональные знания, ни программы, то соответствующую формальную модель будем называть **формальной моделью переработки данных**. Система операций такой формальной модели есть не что иное, как программа, отражающая некоторый метод решения некоторого класса задач, а начальная информационная конструкция такой модели есть не что иное, как исходные данные для конкретной задачи из указанного класса. Классическим примером формальных моделей переработки данных являются различные неуниверсальные машины Тьюринга с разными системами команд и с разными исходными состояниями памяти.

Формальные модели реализации хранимых программ в зависимости от типа программ делятся на **формальные модели реализации хранимых процедурных программ** и **формальные модели реализации хранимых непроцедурных программ**. Подчеркнем, что каждая формальная модель реализации хранимых непроцедурных программ, строго говоря, включает в себя некоторую формальную модель логического вывода, начальная информационная конструкция которой формируется в результате связывания непроцедурной программы с исходными данными конкретной задачи. Отме-

тим при этом, что для реализации непроцедурных программ обычно используется достаточно простой набор операций логического вывода. Таким образом, непроцедурная программа с семантической точки зрения представляет собой совокупность знаний, являющихся общими для некоторого класса предметных областей и достаточными для решения однотипных (в определенном смысле слова) задач, задаваемых в рамках указанных предметных областей.

Вторым признаком классификации формальных моделей является структурный тип перерабатываемых информационных конструкций.

Если перерабатываемые информационные конструкции являются цепочками (строками) символов, т.е. символьными конструкциями, то соответствующую формальную модель будем называть **формальной моделью переработки символьных конструкций**.

Если перерабатываемые информационные конструкции являются графовыми конструкциями, то соответствующую формальную модель будем называть **формальной моделью переработки графовых конструкций**.

Третьим признаком классификации формальных моделей является характер возможных преобразований перерабатываемых информационных конструкций – возможность изменения связей между элементами этих конструкций. По этому признаку формальные модели делятся на формальные модели переработки структурно фиксированных информационных конструкций и формальные модели переработки структурно перестраиваемых информационных конструкций.

Основным предметом исследования данной работы являются формальные модели переработки графовых структурно перестраиваемых информационных конструкций, которые будем называть **графодинамическими формальными моделями**.

По типу используемых формальными моделями абстрактных машин они делятся на последовательные и параллельные модели, синхронные и асинхронные и т.д.

1.1.3. Интеграция и интерпретация формальных моделей обработки информации

Ключевые понятия: интеграция, интерпретация.

Важнейшими соотношениями между формальными моделями являются соотношения **интеграции и интерпретации**.

Интеграция двух формальных моделей заключается в построении некоторой третьей модели, гарантирующей решение всех задач, решаемых в интегрируемых моделях, и дополнительно обеспечивающей решение некоторых новых задач, для которых требуется совместное использование интегрируемых моделей.

Интеграция формальных моделей в общем случае предполагает:

- перекодировку начальных информационных конструкций интегрируемых моделей;
- соединение перекодированных начальных информационных конструкций;
- преобразование каждой операции интегрируемой модели в эквивалентную операцию интегрированной формальной модели.

Формальные модели, которым соответствует достаточно простая технология интеграции, будем называть **легко интегрируемыми**.

Интерпретация произвольной формальной модели сводится:

- к выбору некоторого языка программирования;
- к описанию на этом языке программы реализации каждой операции интерпретируемой формальной модели в виде демонической (периодически иницируемой) программы;
- к разработке системы операций, обеспечивающих реализацию хранимых программ, написанных на указанном языке программирования;

- к формированию начальной информационной конструкции, включающей в себя начальную информационную конструкцию интерпретируемой формальной модели (возможно, каким-либо способом перекодированную) и тексты демонических программ, описывающих операции интерпретируемой формальной модели.

1.1.4. Семиотические модели обработки информации

Ключевые понятия: семиотическая модель, метаоперация, метаязык, семиотическая абстрактная машина.

В ходе расширения сферы применения интеллектуальных систем стало ясно, что формальные модели отражают не все аспекты обработки информации в интеллектуальных системах. Так, например, при обучении, при самообучении, при адаптации к изменяющейся внешней среде интеллектуальная система должна иметь возможность пересматривать начальную информационную конструкцию, модифицировать язык описания внешней среды и даже модифицировать правила решения задач. В основе интеллектуальных систем нового поколения должны лежать такие модели обработки информации, в которых не фиксируется ни начальная информационная конструкция, ни язык, ни абстрактная машина. Такие модели называются **семиотическими моделями** или семиотическими системами [392] (*Поспелов Д. А. 1986кн-СитуаУ*).

Семиотические модели носят открытый, гибкий, легко перестраиваемый, модифицируемый, самоизменяемый, адаптируемый, эволюционируемый, немонотонный, динамический характер.

В рамках семиотической модели процесс обработки информации рассматривается не только как процесс преобразования информационной конструкции по правилам некоторой формальной модели, но и как процесс перехода от одних формальных моделей к другим.

Текущее состояние процесса обработки информации, описываемого семиотической моделью, в общем случае определяется некоторым множеством взаимодействующих друг с другом формальных моделей, а точнее, текущим состоянием всех формальных моделей, входящих в это множество. Текущее состояние каждой формальной модели определяется текущим состоянием перерабатываемой информационной конструкции (т.е. текущим состоянием памяти соответствующей абстрактной машины), текущим состоянием используемого языка и текущим состоянием используемой системы операций. Таким образом, семиотическую модель можно трактовать как метамодель, описывающую некоторое семейство формальных моделей и, в частности, задающую правила перехода от одних формальных моделей к другим формальным моделям из этого семейства. С другой стороны, семиотическую модель можно трактовать как систему динамических (изменяющихся) формальных моделей.

Определение 1.1.4.1. Семиотическая модель M может быть задана парой

$$M = \langle FS, WF \rangle,$$

где

FS – одна или несколько формальных моделей, определяющих исходное состояние семиотической модели. В исходном состоянии семиотической модели каждая формальная модель, входящая в семейство FS , может быть либо активной (инициированной, непосредственно реализуемой), либо пассивной (неинициированной, нереализуемой). При этом 1) активных формальных моделей, реализуемых параллельно (одновременно), может быть сколько угодно, 2) эти формальные модели могут иметь общие фрагменты перерабатываемых информационных конструкций;

WF – **метаоперации** (метаправила) модификации семейства формальных моделей, определяющих текущее состояние семиотической модели. Эти метаоперации обеспечивают: 1) перевод пассивных формальных моделей в активное состояние; 2) перевод активных формальных моделей в пассивное состояние; 3) ликвидацию формальных моделей; 4) синтез новых или модификацию имеющихся формальных моделей.

Синтез новой формальной модели или модификация имеющейся формальной модели в общем случае предполагает синтез (или модификацию) языка, синтез (или модификацию) начальной информационной конструкции, синтез (или модификацию) набора операций.

Каждая метаоперация, входящая во множество WF , может задавать любые сочетания различных видов модификации формальных моделей (активизацию, прерывание, ликвидацию формальных мо-

делей, модификацию языка, модификацию начальной информационной конструкции и модификацию операций). Поэтому разбить множество метаправил на подмножества в соответствии с видами модификации формальных моделей представляется возможным не для всех семиотических моделей.

Одним из вариантов реализации семиотической модели $M = \langle FS, WF \rangle$ является построение формальной метамодели $F = \langle L, S, C \rangle$, обеспечивающей интерпретацию семиотической модели M . Построение формальной метамодели F включает в себя:

- создание **метаязыка** L , обеспечивающего описание всевозможных формальных моделей (описание синтаксиса и семантики различных языков, описание семантики различных операций, представление начальных информационных конструкций формальных моделей, представление текущего состояния формальных моделей, т.е. текущего состояния перерабатываемых информационных конструкций), обеспечивающего описание соотношений между формальными моделями, определяющими одно состояние или разные состояния реализуемой семиотической модели, и обеспечивающего описание семантики метаопераций реализуемой семиотической модели, т.е. метаопераций, входящих во множество WF ;
- построение начальной информационной конструкции S , включающей в себя полное описание всех формальных моделей, входящих во множество FS , и полное описание всех метаопераций, входящих во множество WF ;
- создание универсальной абстрактной машины C , обеспечивающей интерпретацию всевозможных формальных моделей по их описаниям на метаязыке L , а также интерпретацию всевозможных метаопераций по их описаниям на метаязыке L . Такие абстрактные машины, использующие метаязык рассмотренного класса, обеспечивающие параллельную интерпретацию нескольких формальных моделей и легко переходящие с интерпретации одних формальных моделей на интерпретацию других формальных моделей с новыми языками, начальными информационными конструкциями и операциями, будем называть **семиотическими абстрактными машинами**. Разработка таких машин является одним из важнейших современных направлений развития работ в области искусственного интеллекта.

1.1.5. Графодинамические ассоциативные модели обработки знаний и известные виды моделей представления знаний

Ключевые понятия: графодинамические модели переработки знаний, сетевые модели, графовая конструкция.

Рассмотрим соотношение **графодинамических моделей переработки знаний** с известными классами моделей представления знаний, к числу которых относятся сетевые модели представления знаний, логические, фреймовые, продукционные модели.

Очевидно, что сетевые модели представления знаний, в которых знания представляются в виде семантических сетей, устроенных тем или иным способом, имеют к графодинамическим моделям переработки знаний самое непосредственное отношение, ибо семантическая сеть есть частный вид графовой конструкции.

Недостатки, обычно приписываемые **сетевым моделям** [120] (*Георгиев В.О.1993ст-МоделПЗ*), относятся не столько к самим моделям, сколько к сложностям их реализации на традиционных (фон-Неймановских) компьютерах. Если же вести речь о специальных графодинамических компьютерах, ориентированных на реализацию графодинамических моделей, то преимущество моделей такого класса становится очевидным, ибо их выразительные возможности намного превышают возможности других моделей. Тем более, что графодинамические модели представления и переработки знаний не являются альтернативой фреймовым, логическим и продукционным моделям, поскольку вполне могут существовать графодинамические фреймовые модели (в рамках которых фреймы представляются в виде графовых конструкций), графодинамические логические модели (в рамках которых логические высказывания представляются в виде графовых конструкций, а логический вывод рассматривается как манипуляция такими конструкциями), графодинамические продукционные модели (в рамках которых перерабатываемая продукциями информация и сами продукты представляются в виде графовых конструкций). Более того, в рамках графодинамической парадигмы легко представить себе комплексные модели, сочетающие в себе и фреймовые, и логические, и продукционные аспекты, т.е. модели, в которых интегрируются достоинства перечисленных типов моделей и нейтрали-

зуются их недостатки. О целесообразности создания таких комплексных моделей говорится в целом ряде работ [401; 120] (*Поспелов Д.А. ред. 1990спр-ИскусИ-К2 ; Георгиев В.О. 1993ст-МоделПЗ*).

В основе фреймовых моделей представления знаний лежит системно-структурная трактовка описываемой предметной области, при которой предметная область рассматривается как совокупность взаимосвязанных подсистем, каждая из которых описывается в виде соответствующего фрейма. Связи между фреймами могут быть самыми различными, в частности, они могут иметь достаточно сложную иерархическую структуру. **Графовые конструкции** являются удобным средством представления фреймовых структур, поэтому фреймовые модели часто считают частным случаем сетевых. Соответствующие языки представления знаний условно можно назвать графовыми фреймовыми языками представления знаний.

Графодинамические модели обработки информации хорошо согласуются также и с логическими моделями, так как в этих моделях ничто не запрещает в качестве перерабатываемых информационных конструкций использовать графовые конструкции. В частности, ничто не запрещает строить так называемые графовые логические языки, лежащие в основе графодинамических логических моделей представления и переработки знаний. В указанных логических языках графового типа логические высказывания (логические формулы) представляются не в виде символьных конструкций, как в известных логических языках, а в виде графовых конструкций. Наиболее сложным при создании графовых логических языков является выбор удобного способа представления логических связей и кванторов.

Графодинамическая парадигма является единственно возможной основой для формального рассмотрения и реализации параллельной переработки знаний. Это обусловлено тем, что параллельная переработка знания всегда есть одновременная переработка различных и заранее непредсказуемых фрагментов этого знания и одновременная реализация сразу нескольких стратегий решения одной и той же задачи. Любой фрагмент перерабатываемого знания не может быть выделен из этого знания таким образом, чтобы у этого фрагмента отсутствовали связи с остальной частью знания. Такая связь, в частности, может осуществляться через имена (идентификаторы), присваиваемые различным объектам представляемого знания. Указанное свойство знания можно назвать его связностью или целостностью. Из этого свойства следует, что при переработке каждого фрагмента знания в общем случае может непредсказуемо измениться не только внутренняя структура этого фрагмента, но и его внешние связи (в том числе и метасвязи) с остальной частью перерабатываемого знания, т.е. при переработке каждого фрагмента знания в общем случае требуется учет не только его внутренней структуры, но и его контекста (ближайшего окружения). При последовательной переработке какого-либо знания нет необходимости говорить о различных фрагментах этого знания – в каждый момент времени в качестве перерабатываемого фрагмента можно в этом случае рассматривать все перерабатываемое знание. При параллельной переработке знания картина меняется, ибо приходится говорить об одновременно перерабатываемых фрагментах этого знания. При этом процессы переработки указанных фрагментов могут взаимодействовать друг с другом в силу связности знания – при переработке одного фрагмента могут измениться также его внешние связи, которые непосредственно входят в состав другого одновременно обрабатываемого фрагмента.

Таким образом, процесс параллельной переработки знания есть взаимодействие (своеобразная "интерференция") одновременно выполняемых локальных процессов, каждый из которых осуществляет переработку некоторого фрагмента знания. Причем взаимодействие этих локальных процессов осуществляется на основе анализа внешних связей каждого фрагмента. Следовательно, для организации параллельной переработки знания необходимы 1) явное представление всевозможных связей между потенциально обрабатываемыми фрагментами знания и 2) поддержка возможности изменения этих связей. Очевидно, что графодинамическая парадигма переработки знаний этим требованиям удовлетворяет.

Графодинамическая парадигма является также единственно возможной основой для развитых форм ассоциативного доступа к хранимой в памяти информации [269] (*Кохонен Т. 1980кн-АссоцП*). Как известно, без развитых форм ассоциативного доступа к требуемым фрагментам перерабатываемой сложноструктурированной информации, имеющим произвольный размер, произвольную конфигурацию и произвольную "привязку" к остальной части хранимой информации, невозможна реализация наиболее перспективных моделей параллельной переработки сложноструктурированной информации – асинхронных моделей [208] (*Ершов А.П. ред. 1992кн-АлгорМОиА*).

Резюме к подразделу 1.1

Завершая данный подраздел, отметим следующее.

1. Преимущества графодинамических моделей обработки информации по сравнению с другими видами моделей обработки информации (символьными моделями) заключаются в следующем:

- графодинамические модели легко поддерживают развитые формы ассоциативного доступа;
- графодинамические модели легко поддерживают параллельную реализацию операций;
- графодинамические модели легко поддерживают асинхронную реализацию операций;
- графодинамические модели проще интегрируются, так как интеграция начальных информационных конструкций интегрируемых графодинамических моделей может быть сведена к простому склеиванию некоторых элементов интегрируемых графовых конструкций при полном сохранении этих конструкций;
- графодинамические модели легко интерпретируются друг другом, потому что благодаря ассоциативному доступу легко обеспечить независимость программ, описывающих методы реализации операций интерпретируемой модели, от перерабатываемых ими данных, т.е. от начальной информационной конструкции интерпретируемой модели;
- в графодинамических моделях легко поддерживается работа со сложноструктурированной информацией любой степени сложности.

2. Преимущество графовых средств представления информации по сравнению с символьными средствами и преимущество графодинамических моделей обработки информации отмечается в целом ряде работ. "Всякий раз, когда с задачей удается связать граф, обсуждение резко упрощается и большие фрагменты словесного описания заменяются манипуляциями с картинками" [317] (*Манин Ю.И. 1979кн-ДоказИИ*).

3. В основе графодинамических моделей обработки информации лежит понятие графового языка, подробно рассматриваемое в подразделе 1.2, и понятие графодинамической абстрактной машины, подробно рассматриваемое в подразделе 1.3.

4. Наиболее перспективным классом графодинамических моделей для реализации интеллектуальных систем нового поколения являются графодинамические параллельные асинхронные модели. Параллелизм для графодинамических моделей вполне естественен, а возможность эффективной организации параллельной обработки является одним из главных достоинств графодинамических моделей. Асинхронность организации выполнения операций является естественным свойством моделей переработки знаний (не только графодинамических). Кроме того, без асинхронной организации невозможно обеспечить гибкость модели. Но только в рамках графодинамических моделей представляются широкие возможности для реализации асинхронной переработки знаний, благодаря поддержке развитых форм ассоциативного доступа.

5. Предметом исследования в данной работе являются не только сами модели обработки информации, но и соотношения между ними, а точнее, такие соотношения, которые лежат в основе проектирования (синтеза) новых моделей (в частности, путем интеграции нескольких моделей), а также в основе их практической реализации (в частности, путем интерпретации, т.е. путем сведения реализуемой модели к эквивалентной модели более простого вида).

6. В данной работе исследуются не только принципы (технологии) построения графодинамических моделей параллельной асинхронной переработки знаний, но и принципы практической реализации этих моделей. Суть такой реализации сводится к построению иерархии интерпретирующих графодинамических моделей параллельной асинхронной обработки информации, в рамках которой обеспечивается сведение графодинамических моделей параллельной асинхронной переработки знаний к графодинамическим моделям более простого вида с четкой технологией перехода от уровня к уровню.

1.2. Графовые языки

Ключевые понятия: графовый язык, графовая структура, реляционная структура.

В данном подразделе рассматриваются основные понятия, связанные с нетрадиционным классом языков – классом **графовых языков**, текстами которых являются **графовые структуры**.

Графовые языки составляют основу графодинамических формальных моделей обработки информации, исследуемых в данной работе и неформально рассмотренных в подразделе 1.1. Формальное рассмотрение графовых языков дает возможность уточнить понятие графодинамической формальной модели, являющееся центральным в данной работе.

1.2.1. Понятие реляционной структуры как уточнение понятия предметной области и понятия информационной конструкции

Ключевые понятия: реляционная структура, множество, кортеж, атрибут, универсум, отношение подчинения, множество реляционной структуры, кортеж реляционной структуры, атрибут реляционной структуры, первичный элемент реляционной структуры, отношение реляционной структуры, область определения отношения, проекция отношения, классическое отношение, классическая реляционная структура, гомоморфизм, изоморфизм, нестационарная реляционная структура.

Понятие **реляционной структуры** нами используется как способ формального уточнения предметных областей различного вида, т.е. как способ задания их структуры. Нас будут интересовать сложноструктурированные предметные области и соответствующие им сложноструктурированные реляционные структуры.

На предметные области не накладывается никаких ограничений. В частности, предметной областью может быть информационный объект, являющийся описанием какой-то другой предметной области. Информационному объекту можно поставить в соответствие две реляционные структуры. Первая из них определяет внутреннюю структуру (внутреннее строение, синтаксис) информационного объекта. Такую реляционную структуру будем называть информационной конструкцией. Вторая структура задает семантику информационного объекта, т.е. его соотношение с описываемой им предметной областью. Вторая из указанных реляционных структур является метаконструкцией, поскольку с формальной точки зрения является описанием определенного соотношения между двумя реляционными структурами – между информационной конструкцией, определяющей структуру соответствующего информационного объекта, и реляционной структурой, определяющей структуру предметной области, описываемой указанным информационным объектом. Каждый язык (в том числе и каждый графовый язык) формально задается некоторым множеством информационных конструкций, имеющих общие синтаксические и семантические свойства. Введение понятия реляционной структуры позволяет с единых позиций рассматривать различные языки, сравнивать их друг с другом и классифицировать.

Для уточнения понятий сложноструктурированной предметной области и сложноструктурированной информационной структуры вводится понятие реляционной структуры. Это понятие является обобщением классического понятия алгебраической модели [316] (*Мальцев А.И. 1970кн-АлгебС*), а также таких понятий, как клубная система [65] (*Борщев В.Б. 1983ст-СхемыНКС*), гиперсеть [385] (*Понков В.К. 1986ст-ГиперИиХС*), П-граф [213] (*Ефимова С.М. 1983ст-ОднойФМ*).

Реляционные структуры позволяют работать со структурами существенно более сложного вида, чем алгебраические модели, – иерархическими, многоуровневыми структурами.

В основе понятия реляционной структуры лежат понятия **множества, кортежа и атрибута**. Необходимость глубокого семантического анализа реляционных структур заставляет:

- 1) четко противопоставлять понятие знака объекта и понятие самого объекта, обозначаемого этим знаком. Под объектами могут пониматься какие-то отдельные предметы описываемой предметной области, множества, кортежи, целые реляционные структуры;
- 2) ввести специальное бинарное асимметричное отношение принадлежности, связывающее знаки множеств с элементами множеств, обозначаемых этими знаками.

Множество задается знаком (обозначением, именем) множества, перечнем (набором) его элементов, семейством пар отношения принадлежности, связывающих знак множества с каждым элементом этого множества. Элементами множества могут быть объекты любой природы. В зависимости от количества элементов множества бывают пустыми, конечными (1-элементными, 2-элементными и т.д.), бесконечными.

Отношение принадлежности не является однозначным. Одному множеству может принадлежать несколько элементов. Элемент, принадлежащий одному множеству, может также принадлежать и другому множеству. В указанное семейство пар отношения принадлежности одна и та же пара может входить несколько раз, т.е. элементы множества могут входить в его состав многократно. Поэтому отношение принадлежности не может быть отнесено к числу классических бинарных отношений.

Следует четко отличать понятие самого множества (как математической конструкции) от понятия знака множества. Знак одного множества может быть элементом другого множества. Кроме того, знак множества может быть одним из элементов того множества, которое он сам обозначает.

Кортеж задается знаком кортежа, перечнем его элементов (компонентов), семейством пар отношения принадлежности, связывающих знак кортежа с каждым элементом этого кортежа, семейством множеств, элементами которых являются указанные выше пары отношения принадлежности. Такие множества будем называть атрибутами кортежей.

Атрибут a_t кортежа t задается знаком атрибута a_t , перечнем его элементов, каждый из которых является парой отношения принадлежности, связывающего знак кортежа t с его элементами, имеющими атрибут a_t , семейством пар отношения принадлежности, связывающих знак атрибута a_t с каждым его элементом.

Элементами кортежей могут быть объекты любой природы. В общем случае кортежу принадлежит несколько элементов. Элемент, принадлежащий одному кортежу, может принадлежать другому кортежу. Элементы кортежа могут входить в его состав многократно. Знак одного кортежа может быть элементом другого кортежа. Более того, знак кортежа может быть одним из элементов того кортежа, который он сам обозначает.

Каждый атрибут определяет роль соответствующих элементов кортежа в рамках всего кортежа. Каждая пара отношения принадлежности, связывающая знак кортежа с каким-либо его элементом, может либо принадлежать одному атрибуту, либо принадлежать сразу нескольким атрибутам, либо не принадлежать ни одному атрибуту. Элемент кортежа может одновременно являться атрибутом этого же кортежа. Атрибут кортежа может одновременно являться элементом этого же кортежа.

Таким образом, кортеж – это множество, каждому элементу которого ставится в соответствие атрибут этого элемента (в частности, номер), под которым он принадлежит данному кортежу. Атрибут элемента кортежа есть свойство относительное, ибо при интеграции кортежей в единую реляционную структуру элементы одного кортежа могут быть элементами других кортежей и иметь при этом другие атрибуты. Следует отметить, что в предлагаемой трактовке кортежа множество его атрибутов является произвольным множеством, в то время как для классического определения кортежа атрибутами являются номера элементов, которые условно будем обозначать символами $1_$, $2_$, $3_$ и т.д.

Кортеж, имеющий элементы c_1, c_2, \dots, c_n , атрибутами которых в рамках данного кортежа соответственно являются $a_{1_}, a_{2_}, \dots, a_{n_}$, в символической записи будем представлять следующим образом: $\square a_{1_} : c_1, a_{2_} : c_2, \dots, a_{n_} : c_n \square$. При этом кортежи классического вида, атрибутами которых являются номера их компонентов, т.е. кортежи $\square 1_ : c_1, 2_ : c_2, \dots, n_ : c_n \square$, будем представлять также и общепринятым образом: $\square c_1, c_2, \dots, c_n \square$.

Утверждение 1.2.1.1. Любой кортеж можно представить в виде классического кортежа, т.е. кортежа, атрибутами которого являются номера его элементов.

Для этого нужно построить взаимно однозначное соответствие между множеством атрибутов неклассического кортежа и отрезком натурального ряда чисел от 1 до n , т.е. пронумеровать атрибуты.

Кортежи друг от друга могут отличаться:

- 1) набором атрибутов (классические кортежи, неклассические кортежи с числовыми атрибутами, неклассические кортежи с нечисловыми атрибутами);
- 2) количеством элементов (унарные, бинарные, тернарные кортежи и т.д.);
- 3) наличием или отсутствием элементов, имеющих одинаковые атрибуты;
- 4) наличием или отсутствием многократного вхождения каких-либо элементов под разными или одинаковыми атрибутами;

- 5) наличием или отсутствием элементов кортежа, являющихся одновременно атрибутами этого же кортежа;
- 6) тем, является ли знак кортежа элементом обозначаемого им кортежа.

Определение 1.2.1.1. Универсум H , построенный на множестве P с атрибутами A , определяется рекурсивно на основе следующих утверждений:

- 1) $P \subset H$. Элементы множества P будем называть первичными (терминальными) элементами универсума H . Само множество P будем называть базовым множеством универсума H ;
- 2) $A \subset H$; $A / P = \emptyset$. Множество A формально трактуется как множество знаков атрибутов;
- 3) если hi есть подмножество универсума H ($hi \subset H$), то знак этого подмножества также является элементом универсума H ;
- 4) если hi есть произвольный элемент универсума H ($hi \in H$), а $ai_$ есть произвольный элемент множества A ($ai_ \in A$), то знак одноэлементного (унарного) кортежа $\square ai_ : hi \square$ также является элементом универсума H ;
- 5) если hi есть произвольный элемент универсума H , $ai_$ есть произвольный элемент множества A , hj есть элемент универсума, являющийся знаком кортежа, то элементом универсума H также является знак кортежа, полученного в результате присоединения к кортежу hj нового элемента hi с атрибутом $ai_$;
- 6) никаких других элементов множество H не содержит.

Таким образом, в состав универсума H входят следующие элементы:

- первичные (терминальные) элементы;
- знаки атрибутов, которые следует отличать от самих атрибутов;
- вторичные (производные) элементы, являющиеся знаками множеств, состоящих только из первичных элементов универсума, – такие множества будем называть простыми;
- вторичные элементы, являющиеся знаками кортежей, состоящих только из первичных элементов универсума, – такие кортежи будем называть простыми;
- вторичные элементы, являющиеся знаками множеств, в состав которых входит хотя бы один вторичный элемент (знак множества или знак кортежа), – такие множества будем называть метамножествами;
- вторичные элементы, являющиеся знаками кортежей, в состав которых входит хотя бы один вторичный элемент, – такие кортежи будем называть метакортежами.

Завершая рассмотрение понятия универсума, отметим то, что это понятие можно считать обобщением известного понятия шкалы множеств [77] (*Бурбаки Н.1965кн-ТеориМ*).

Определение 1.2.1.2. Пусть hi , hj – произвольные элементы универсума. Будем говорить, что hj принадлежит hi в том, и только в том случае, если либо hj есть элемент множества, знаком которого является hi , либо hj есть элемент кортежа, знаком которого является hi .

Из приведенного определения следует, что каждый элемент универсума, которому принадлежит хотя бы один элемент этого универсума, является вторичным элементом, т.е. является либо знаком множества, либо знаком кортежа.

Транзитивное замыкание отношения принадлежности назовем отношением **подчинения**, которое, соответственно, определяется рекурсивно на основании следующих утверждений:

- 1) если hj принадлежит вторичному элементу hi , то hj подчинен элементу hi ;
- 2) если hj подчинен элементу hi , а hk подчинен элементу hj , то hk подчинен элементу hi .

Теперь перейдем к определению понятия реляционной структуры.

Определение 1.2.1.3. Реляционная структура G задается кортежем:

$\square P, A, K, R, D \square$,

где

- P – множество первичных элементов реляционной структуры G (которые в рамках этой структуры отмечаются атрибутом *primaryEl_*);
- A – множество знаков атрибутов реляционной структуры G (которые в рамках этой структуры отмечаются атрибутом *attr_*);
- K – множество знаков связей реляционной структуры G (которые в рамках этой структуры отмечаются атрибутом *conn_*);
- R – множество знаков отношений реляционной структуры G (которые в рамках этой структуры отмечаются атрибутом *rel_*). Знаки связей и знаки отношений реляционной структуры будем называть вторичными элементами этой структуры;
- D – множество элементов неопределенного типа реляционной структуры G (которые в рамках этой структуры не имеют атрибутов).

При этом должны выполняться следующие условия:

- множества P, A, K, R, D попарно не пересекаются;
- вторичные элементы реляционной структуры G представляют собой знаки множеств или кортежей и являются вторичными элементами универсума H , построенного на множестве P с атрибутами A , т.е. $(K \cup R) \subset H$. Это условие назовем свойством иерархичности реляционных структур, которое заключается в том, что элементами вторичных элементов реляционной структуры являются элементы (в том числе и вторичные элементы) этой же реляционной структуры;
- каждый элемент реляционной структуры G , не являющийся знаком ее отношения, должен быть подчинен знаку хотя бы одного отношения этой реляционной структуры. Понятие отношения подчинения, заданного на элементах реляционной структуры, аналогично понятию отношения подчинения, заданному на элементах универсума (см. определение 1.2.1.1). Это свойство назовем свойством целостности реляционных структур.

Множество первичных элементов реляционной структуры также будем называть базовым множеством этой реляционной структуры.

Множество K знаков связей реляционной структуры разбивается на два подмножества:

$K = S \cup T$; $S \cap T = \emptyset$,

где

- S – множество знаков неупорядоченных связей реляционной структуры G , которым в рамках этой конструкции приписывается атрибут *connSet_*;
- T – множество знаков упорядоченных связей (кортежей) реляционной структуры G , которым в рамках этой структуры приписывается атрибут *connTuple_*.

Вторичные элементы, знаки которых принадлежат множеству $S \cup R$, будем называть множествами реляционной структуры G .

Множество реляционной структуры – это множество, которое целиком, как математическая структура, входит в состав (является фрагментом) этой реляционной структуры. Это означает, что в состав указанной реляционной структуры входят знак указанного множества, все элементы этого множества, все пары отношения принадлежности, связывающие знак множества с его элементами.

Кортеж реляционной структуры – это кортеж, который целиком, как математическая структура, входит в состав (является фрагментом) этой реляционной структуры вместе со всеми его элементами, со знаками всех используемых им атрибутов и со всеми соответствующими ему парами отношения принадлежности.

Атрибут реляционной структуры – это атрибут, используемый по крайней мере одним кортежем, входящим в состав указанной реляционной структуры.

Первичный элемент реляционной структуры – это такой элемент, который не является ни знаком множества, целиком входящего в состав указанной реляционной структуры, ни знаком кортежа, цели-

ком входящего в состав этой структуры, ни знаком атрибута такого кортежа. Знаки множеств, знаки кортежей и знаки атрибутов могут быть первичными элементами реляционной структуры, но в этом случае соответствующие множества и кортежи как математические структуры не считаются фрагментами указанной реляционной структуры. Даже если в число первичных элементов реляционной структуры входит знак некоторого множества, а также все элементы этого множества, то считается, что пары отношения принадлежности, связывающего знак указанного множества с его элементами, в состав указанной реляционной структуры не входят.

Отношение реляционной структуры – это специально выделенное (чаще всего бесконечное) множество реляционной структуры.

Следует подчеркнуть, что пары отношения принадлежности, связывающие знаки атрибутов реляционной структуры и вторичные элементы реляционной структуры с элементами обозначаемых ими множеств или кортежей, входят в состав реляционной структуры, хотя формально ее элементами не являются.

Определение 1.2.1.4. Пусть r – одно из отношений некоторой реляционной структуры и пусть элементами этого отношения являются вторичные элементы указанной реляционной структуры. Тогда множество m будем называть **областью определения отношения r** в том и только в том случае, если выполняются следующие условия:

- 1) каждый элемент кортежа, принадлежащего отношению r , является элементом множества m ;
- 2) каждый элемент множества, принадлежащего отношению r , является элементом множества m ;
- 3) каждый элемент множества m является либо элементом некоторого кортежа, принадлежащего отношению r , либо элементом некоторого множества, принадлежащего отношению r .

В область определения отношения могут входить не только первичные элементы и элементы неопределенного типа, но и вторичные элементы реляционной структуры, если среди кортежей или множеств, принадлежащих отношению r , имеются метакортежи или метамножества.

Определение 1.2.1.5. Множество l будем называть **проекцией отношения r** по атрибуту a в том и только в том случае, если:

- 1) каждый элемент с атрибутом a кортежа, принадлежащего отношению r , является элементом множества l ;
- 2) каждый элемент множества l является элементом с атрибутом a кортежа, принадлежащего отношению r .

Отношения реляционной структуры бывают классическими (простыми) и неклассическими.

Классическое отношение r представляет собой подмножество декартова произведения $P \times P \times \dots \times P$, где P – множество первичных элементов, т.е. $r \subseteq P \times P \times \dots \times P$.

Если количество сомножителей равно 1, т.е. $r \subseteq P$, то классическое отношение называется унарным. Если количество сомножителей равно 2, т.е. $r \subseteq P \times P$, то классическое отношение называется бинарным. Если количество сомножителей равно 3, т.е. $r \subseteq P \times P \times P$, то классическое отношение называется тернарным и т.д.

Следовательно, классическое отношение – это либо некоторое множество первичных элементов ($r \subseteq P$), либо некоторое множество простых (т.е. состоящих из первичных элементов) кортежей, состоящих при этом из одинакового количества элементов, которым ставятся в соответствие одинаковые наборы атрибутов (натуральные числа от 1 до n , где n – количество элементов кортежа).

Неклассические отношения реляционной структуры отличаются от классических следующим:

- элементами неклассического отношения могут быть элементы реляционной структуры разного типа (в частности, неклассическому отношению могут одновременно принадлежать как первичные элементы, так и знаки кортежей с разным количеством элементов, знаки множеств с разным количеством элементов, в том числе знаки отношений, а также знаки атрибутов);

- элементами неклассического отношения могут быть не только знаки простых кортежей, элементами которых являются первичные элементы реляционной структуры, но и знаки метакортежей, среди элементов которых имеется хотя бы один вторичный элемент реляционной структуры. Кроме того, элементами неклассических отношений могут быть знаки как простых множеств, так и метамножеств;
- элементами неклассического отношения могут быть не только знаки кортежей, в которых каждому элементу соответствует свой атрибут, но и знаки кортежей, в которых некоторые элементы имеют одинаковые атрибуты, не имеют атрибутов или имеют несколько атрибутов;
- элементами неклассического отношения могут быть не только знаки классических кортежей, но и знаки кортежей, для которых набор атрибутов представляет собой произвольный набор натуральных чисел, а также знаки кортежей, в которых в качестве атрибутов используются не только натуральные числа.

Отношения, в область определения которых могут входить знаки отношений, а элементами могут быть знаки метакортежей и метамножеств, будем называть метаотношениями.

Реляционные структуры могут быть самыми разными в зависимости от того, какие отношения входят в их состав и как эти отношения связаны друг с другом. Важными характеристиками здесь являются арность, симметричность, однозначность и область определения отношений (так, например, одно отношение реляционной структуры может полностью или частично входить в область определения другого отношения).

Реляционную структуру, все отношения которой являются классическими, будем называть **классической реляционной структурой**. Такую структуру обычно называют алгебраической моделью или реляционной системой [316] (*Мальцев А.И. 1970кн-АлгебС*).

Переход от классических реляционных структур к неклассическим (сложноструктурированным) реляционным структурам – это переход к иерархическим, многоуровневым структурам, в которых имеют место связи не только между первичными элементами, но и между связями, а также между целыми конструкциями. Это обстоятельство является очень важным при представлении знаний в интеллектуальных системах, ибо в них часто приходится иметь дело со сложноструктурированными, иерархическими предметными областями.

Особо отметим то, что знак кортежа, представляющего некоторую реляционную структуру, может входить в число вторичных элементов другой реляционной структуры, которую будем называть реляционной метаконструкцией. Более того, в число отношений реляционной метаконструкции может входить отношение "быть знаком реляционной структуры" (т.е. быть знаком кортежа, представляющего реляционную структуру). Таким образом, наша трактовка реляционной структуры позволяет, не выходя за рамки реляционной структуры, описывать сами реляционные структуры и связи между ними.

Примерами реляционных метаконструкций являются структуры, описывающие всевозможные морфизмы между реляционными структурами. Каждый такой морфизм – это соответствие между множествами элементов двух реляционных структур, удовлетворяющее тем или иным свойствам. Важнейшими примерами морфизмов являются гомоморфизмы и изоморфизмы.

Определение 1.2.1.6. Пусть $G1$ – реляционная структура, представляющая собой кортеж, множество элементов которого в соответствии с их атрибутами разбивается на семейство подмножеств $\square P1, A1, K1, R1, D1 \square$, а $G2$ – реляционная структура, представляющая собой кортеж, множество элементов которого разбивается на семейство подмножеств $\square P2, A2, K2, R2, D2 \square$ (см. определение 1.2.1.3). Соответствие между множеством элементов реляционной структуры $G1$ и множеством элементов реляционной структуры $G2$ будем называть **гомоморфизмом** в том и только в том случае, если оно удовлетворяет следующим условиям:

- 1) если $x \in P1$, то $x^* \in P2 \cup D2$;
- 2) если $a \in A1$, то $a^* \in A2$;
- 3) если $k \in K1$, то $k^* \in K2$;
- 4) если $r \in R1$, то $r^* \in R2$;
- 5) если $d \in D1$, то $d^* \in P2 \cup A2 \cup K2 \cup R2 \cup D2$;
- 6) если $k \in r \in R1$, то $k^* \in r^* \in R2$;

7) если $x \in k \in K1$, то $x^* \in k^* \in K2$;

8) если $k = \square \dots, a : x, \dots \square$; $k \in K1$, то $k^* = \square \dots, a^* : x^*, \dots \square$; $k^* \in K2$.

Здесь x^* , a^* , k^* , r^* , d^* есть образы элементов x , a , k , r , d в рамках рассматриваемого соответствия.

Определение 1.2.1.7. Реляционная структура $G1$ **гомоморфна** реляционной структуре $G2$ в том и только в том случае, если существует гомоморфизм между ними.

Определение 1.2.1.8. Соответствие между множеством элементов реляционной структуры $G1$ и множеством элементов реляционной структуры $G2$ будем называть **изоморфизмом** в том и только в том случае, если оно является гомоморфизмом между $G1$ и $G2$, а также гомоморфизмом между $G2$ и $G1$.

Определение 1.2.1.9. Реляционные структуры $G1$ и $G2$ **изоморфны** в том и только в том случае, если существует изоморфизм между ними.

Наряду с понятием реляционной структуры введем понятие **нестационарной реляционной структуры**, т.е. реляционной структуры, изменяющейся во времени. В этом смысле обычные реляционные структуры можно назвать стационарными. Для нестационарных реляционных структур вводится понятие ситуативного отношения, т.е. отношения, которое в разных состояниях нестационарной реляционной структуры состоит из разных элементов (разных кортежей, разных множеств или разных первичных элементов).

Важнейшим приемом описания нестационарной реляционной структуры является ее представление в виде обычной (стационарной) реляционной метаконструкции, описывающей соотношение во времени между различными состояниями нестационарной реляционной структуры, каждое из которых описывается конкретной стационарной реляционной структурой. Состояние нестационарной реляционной структуры будем также называть ситуациями этой структуры. Таким образом, описание нестационарных реляционных структур также приводит к сложноструктурированным иерархическим реляционным структурам.

В стационарной реляционной метаконструкции, описывающей нестационарную реляционную структуру, применяется целый ряд метаотношений, заданных на множестве оболочек стационарных реляционных структур, описывающих различные состояния нестационарной структуры. К числу этих метаотношений относятся: метаотношение "Следовать сразу за"; метаотношение "Следовать за через некоторый промежуток времени"; метаотношение "Причина - следствие"; метаотношение "Следовать за с перекрытием во времени"; метаотношение "Происходить одновременно"; метаотношение, связывающее начальную и конечную ситуацию некоторого процесса, метаотношение, связывающее ситуацию с подситуацией, которая является частным случаем (этапом, стадией) для первой ситуации.

Важно подчеркнуть, что при создании сложных интеллектуальных систем обычно приходится иметь дело именно с неклассическими сложноструктурированными иерархическими реляционными структурами самого разного вида, и в частности с реляционными метаконструкциями.

1.2.2. Линейные тексты

Ключевые понятия: линейный текст, символьная конструкция.

Перейдем к рассмотрению информационных конструкций, которые представляют собой реляционные структуры, описывающие структуру информационных объектов, т.е. объектов, которые сами являются описаниями каких-либо предметных областей. Никаких ограничений на структуру информационных объектов не накладывается и, соответственно этому, не накладывается никаких ограничений и на структуру информационных конструкций. Единственное, что в информационных конструкциях невозможно, – это то, чтобы их первичными элементами были предметы описываемой предметной области. Первичными элементами информационных конструкций могут быть не сами предметы описываемой предметной области, а их знаки. Первичными элементами информационных конструкций могут быть также символы, из которых строятся знаки предметов описываемой предметной области.

В данном пункте рассматривается один из видов информационных конструкций – **линейные тексты** (символьные информационные конструкции), называемые также символьными конструкциями, линейными информационными конструкциями, цепочечными информационными конструкциями, цепочками символов, строками символов, которые являются традиционным видом информационных конструкций.

Определение 1.2.2.1. Пусть G_s – информационная конструкция, задаваемая кортежем, множество элементов которого в соответствии с их атрибутами разбивается на семейство подмножеств $\square P_s, A_s, K_s, R_s, D_s \square$ (см. определение 1.2.1.3). Информационную конструкцию

G_s будем называть **символьной конструкцией** в том и только в том случае, если выполняются следующие условия:

- 1) $A_s = \{ 1_-, 2_- \}$;
- 2) если $k \in K_s$, то $k = \square 1_- : pi, 2_- : pj \square$; $pi, pj \in P_s$;
- 3) $R_s = R_p \cup \{ K_s \}$, где R_p есть семейство унарных отношений, заданных на множестве P_s ;
- 4) не существует $pi, pj, pe \in P_s$ таких, что $\square 1_- : pi, 2_- : pj \square \in K_s$,
 $\square 1_- : pi, 2_- : pe \square \in K_s$;
- 5) не существует $pi, pj, pe \in P_s$ таких, что $\square 1_- : pi, 2_- : pj \square \in K_s$,
 $\square 1_- : pe, 2_- : pj \square \in K_s$;
- 6) $D_s = \emptyset$.

Первичные элементы символьной конструкции (т.е. элементы множества P_s) будем называть символами. Каждый бинарный асимметричный кортеж, принадлежащий множеству K_s , есть связь непосредственного соседства символов в строке. При этом атрибут “1_” указывает на предшествующий символ, а атрибут “2_” указывает на последующий символ. Унарные отношения, входящие в состав R_p , будем называть типами символов, а все семейство унарных отношений R_p будем называть алфавитом символов.

На множестве символьных конструкций задается целый ряд отношений – отношение равенства, отношение включения, отношение конкатенации.

Символьные конструкции равны, если 1) они изоморфны, 2) у них совпадают алфавиты символов, 3) типы символов в рамках указанного изоморфизма соответствуют сами себе. Очевидно, что множество всевозможных символьных конструкций может быть разбито на классы равных символьных конструкций.

Символьная конструкция, определяемая семейством множеств $\square P_s, A_s, K_s, R_s, D_s \square$, связная, если симметризация и транзитивное замыкание отношения $\{ K_s \}$ приводят к декартову произведению $P_s \times P_s$.

Символьная конструкция, определяемая семейством множеств $\square P_s, A_s, K_s, R_s, D_s \square$, является результатом конкатенации связных символьных конструкций, определяемых семейством множеств $\square P_{si}, A_{si}, K_{si}, R_{si}, D_{si} \square$ и семейством множеств $\square P_{sj}, A_{sj}, K_{sj}, R_{sj}, D_{sj} \square$, если

- 1) $P_s = P_{si} \cup P_{sj}$;
- 2) $K_s = K_{si} \cup K_{sj} \cup \{ k \}$;
- 3) $R_s = R_{si} \cup R_{sj}$;
- 4) $k = \square 1_- : i, 2_- : j \square$, где
 i – самый правый символ первой символьной конструкции;
 j – самый левый символ второй символьной конструкции.

Утверждение 1.2.2.1. Любую реляционную структуру можно представить в виде эквивалентного линейного текста.

Существует большое число способов (языков) представления (кодирования) произвольных реляционных структур в виде символьных конструкций. Определим один из таких языков, который условно на-

зовем Lrs . Этот язык является прообразом формального языка SCBs, который подробно рассмотрен ниже в разделе 2.

Определение 1.2.2.2. Будем утверждать, что реляционная структура, задаваемая семейством множеств $\square P, A, K, R, D \square$, и символьная конструкция, принадлежащая языку Lrs и определяемая семейством множеств $\square Ps, As, Ks, Rs, Ds \square$, эквивалентны тогда, и только тогда, когда существует взаимно однозначное соответствие между множеством $P \cup A \cup K \cup R \cup D$ (множеством всех элементов исходной реляционной структуры) и некоторым множеством S , каждый элемент которого представляет собой множество равных символьных конструкций, входящих в состав конструкции $\square Ps, As, Ks, Rs, Ds \square$ и являющихся идентификаторами (именами) соответствующих элементов исходной (представляемой) реляционной структуры. При этом должны выполняться следующие условия:

- 1) если $y \in r, r \in R$, то в символьной конструкции присутствует строка вида $r^* !; y^*$, где $r^*, y^* \in S$, r^* и y^* – идентификаторы элементов y и r ;
- 2) если в символьной конструкции присутствует строка вида $r !; y^*$, где $r^*, y^* \in S$, то существуют элементы реляционной структуры $y \in r, r \in R$, для которых r^* и y^* являются идентификаторами;
- 3) если связка $k, k \in K$ является неупорядоченной и состоит из элементов y_1, y_2, \dots, y_n , т.е. $y_1, y_2, \dots, y_n \in k$, то в символьной конструкции присутствует строка вида $k^* !; y_1^*, y_2^*, \dots, y_n^*$, где $k^*, y_1^*, y_2^*, \dots, y_n^*$ – идентификаторы элементов k, y_1, y_2, \dots, y_n ;
- 4) если в символьной конструкции присутствует строка вида $k^* !; y_1^*, y_2^*, \dots, y_n^*$, где $k^*, y_1^*, y_2^*, \dots, y_n^* \in S$, то в реляционной структуре существует неупорядоченная связка $k, k \in K$, состоящая из элементов $y_1, y_2, \dots, y_n, y_1, y_2, \dots, y_n \in k$, для которых $y_1^*, y_2^*, \dots, y_n^*$ являются идентификаторами, k^* является идентификатором элемента k ;
- 5) если связка $k, k \in K$ является кортежем, состоящим из элементов y_1, y_2, \dots, y_n , которые, соответственно, имеют атрибуты $a_{1_}, a_{2_}, \dots, a_{n_}$, то в символьной конструкции присутствует строка вида $k^* !; a_{1_}^* : y_1^*, a_{2_}^* : y_2^*, \dots, a_{n_}^* : y_n^*$, где $k^*, a_{1_}^*, y_1^*, a_{2_}^*, y_2^*, \dots, a_{n_}^*, y_n^*$ идентификаторы элементов $k, a_{1_}, y_1, a_{2_}, y_2, \dots, a_{n_}, y_n$;
- 6) если в символьной конструкции присутствует строка вида $k^* !; a_{1_}^* : y_1^*, a_{2_}^* : y_2^*, \dots, a_{n_}^* : y_n^*$, где $a_{1_}^*, y_1^*, a_{2_}^*, y_2^*, \dots, a_{n_}^*, y_n^* \in S$, то в реляционной структуре существует связка $k, k \in K$, являющаяся кортежем, состоящим из элементов y_1, y_2, \dots, y_n , которые, соответственно, имеют атрибуты $a_{1_}, a_{2_}, \dots, a_{n_}$, причем $k^*, a_{1_}^*, y_1^*, a_{2_}^*, y_2^*, \dots, a_{n_}^*, y_n^*$ являются идентификаторами элементов $k, a_{1_}, y_1, a_{2_}, y_2, \dots, a_{n_}, y_n$.

Заметим, что в символьной конструкции, которая эквивалентна некоторой реляционной структуре, кроме идентификаторов элементов этой реляционной структуры присутствуют специальные вспомогательные строки символов, выполняющие разделительные и ограничительные функции. К таким специальным символьным конструкциям относятся: “!”, “=”, “(”, “)”, “□”, “□”, запятая “,”, точка с запятой “;”.

Рассмотренные символьные конструкции противопоставляются графовым структурам, к числу которых будем относить все информационные конструкции, не являющиеся символьными. При определении графовых структур не уточняется, что из себя представляют первичные элементы этих структур, как конкретно выглядят связки графовых структур. Уточнение всего этого приводит к самым различным вариантам представления и изображения графовой структуры. Так, например, можно говорить о матричных представлениях (матрицы смежности, матрицы инцидентности), о списковых, о графических (топологических) представлениях.

1.2.3. Нелинейные тексты

Ключевые понятия: бинарная информационная конструкция, отношение принадлежности, однородная информационная конструкция.

Если рассматривать информационные конструкции, являющиеся реляционными структурами общего вида, как способ представления информации в памяти систем обработки информации, то следует отметить недостаточный конструктивизм такого представления, ибо трудно построить соответствие между элементами реляционной структуры общего вида и элементами абстрактной или реальной памяти, в которой эта структура может храниться и преобразовываться. Для того чтобы иметь такое соответствие, нужно ограничить (а в идеале – зафиксировать) набор атрибутов и отношений информационной конструкции, не нарушая при этом семантической мощности. Перейдем к частным видам информационных конструкций, удовлетворяющим данному требованию.

Определение 1.2.3.1. Информационную конструкцию Gg , описывающую реляционную структуру G и представляющую собой кортеж, множество элементов которого в соответствии с их атрибутами разбивается на семейство подмножеств $\square Pg, Ag, Kg, Rg, Dg \square$, будем называть **бинарной информационной конструкцией**, если выполняются следующие условия:

- 1) элементы множества Pg взаимно однозначно соответствуют элементам описываемой реляционной структуры G . Каждый элемент множества Pg , т.е. каждый первичный элемент структуры Gg , считается семантически эквивалентным соответствующему элементу описываемой реляционной структуры G ;
- 2) множество Ag фиксировано: $Ag = \{ 1_-, 2_- \}$;
- 3) связки реляционной структуры Gg представляют собой бинарные кортежи, имеющие вид $\square 1_- : k, 2_- : g \square$, где $k \in Pg, g \in (Pg \cup Kg \cup Dg)$, при этом один и тот же кортеж во множество Kg может входить несколько раз (такие кортежи называются кратными). От множества Kg можно перейти ко множеству Kg^* , $Kg^* \subseteq Pg \times (Pg \cup Kg^* \cup Dg)$, оставив для каждой группы кратных кортежей по одному представителю;
- 4) $Rg = Rt \cup Ri$, где Rt – множество знаков унарных отношений (для каждого $r \in Rt$ имеет место $r \subseteq (Pg \cup Dg)$), Ri – множество знаков бинарных отношений (для каждого $r \in Ri$ имеет место $r \subseteq Kg$). Унарные отношения будем также называть метками первичных и неопределенных элементов бинарной информационной конструкции.

Если элемент описываемой реляционной структуры есть ее первичный элемент, представляющий собой некий предмет соответствующей предметной области, то семантическим эквивалентом этого элемента в бинарной информационной конструкции является знак указанного предмета. Если элемент описываемой реляционной структуры есть знак множества или кортежа (атрибуты и отношения считаются частными видами множеств), то семантическим эквивалентом этого элемента в бинарной информационной конструкции является знак семантически эквивалентного множества или кортежа, т.е. множества или кортежа, состоящего из тех, и только тех элементов бинарной информационной конструкции, которые семантически эквивалентны элементам указанного выше множества или кортежа, входящего в состав описываемой реляционной структуры.

Утверждение 1.2.3.1. Каждую реляционную структуру можно представить в виде эквивалентной бинарной информационной конструкции.

Доказательство этого утверждения сводится к построению алгоритма, обеспечивающего для произвольной реляционной структуры построение эквивалентной бинарной информационной конструкции, а также к построению обратного алгоритма, обеспечивающего для произвольной бинарной информационной конструкции построение эквивалентной реляционной структуры общего вида.

Дадим определение эквивалентности реляционных структур общего вида и бинарных информационных конструкций, на основании которого указанные алгоритмы легко могут быть получены.

Определение 1.2.3.2. Пусть G – реляционная структура, представляющая собой кортеж, определяемый семейством множеств $\square P, A, K, R, D \square$, а Gg – бинарная информационная конструкция, представляющая собой кортеж, определяемый семейством множеств $\square Pg, Ag, Kg, Rg, Dg \square$, удовлетворяющих следующим условиям:

- 1) $Pg = Pv \cup Pk \cup Pd$, $Pv / Pk = \emptyset$, $Pv / Pd = \emptyset$, $Pk / Pd = \emptyset$;
- 2) $Ag = \{ 1_-, 2_- \}$ (по определению бинарной конструкции);
- 3) если $y \in Kg$, то $y = \square 1_- : k, 2_- : g \square$, $k \in Pk$, $g \in Pg$;
- 4) $Rg = Rt \cup Ri$, $Rt / Ri = \emptyset$,
 $Rt = \{ Pv, Pk \} \cup Rtr$,
 $Ri = \{ p \} \cup Ria$,
 где p – отношение принадлежности элемента множеству;
- 5) если $y \in r$, $r \in Rt$, то $y \in Pg$;
- 6) если $y \in r$, $r \in Ri$, то $y \in Kg$;
- 7) $Dg = \emptyset$.

Будем говорить, что реляционная структура G и бинарная информационная конструкция Gg эквивалентны тогда и только тогда, когда существуют взаимно однозначные соответствия между P и Pv ($Pv \subseteq Pg$), между A и Ria ($Ria \subseteq Rg$), между K и Pk ($Pk \subseteq Pg$), между R и Rtr ($Rtr \subseteq Rg$), между D и Pd ($Pd \subseteq Pg$) и при этом выполняются следующие условия:

- 1) если $x \in r$, $r \in R$, то $x^* \in r^*$, $r^* \in Rtr$ и наоборот. Здесь x^* , r^* – образы объектов x , r в рамках указанных соответствий;
- 2) если в рамках реляционной структуры имеет место $x \in k$, $k \in K$, где k – вторичный элемент реляционной структуры, являющийся множеством, то $k^* \in Pk$, $x^* \in Pg$, $\square 1_- : k^*, 2_- : x^* \square \in p$, $\square 1_- : k^*, 2_- : x^* \square \in Kg$ и наоборот;
- 3) если в рамках реляционной структуры имеет место $k \in K$, $a \in A$, $k = \square \dots, a : x, \dots \square$, то $k^* \in Pk$, $a^* \in Ria$, $x^* \in Pg$, $\square 1_- : k^*, 2_- : x^* \square \in a^*$, $\square 1_- : k^*, 2_- : x^* \square \in Kg$ и наоборот.

Здесь x^* , a^* , k^* – образы объектов x , a , k в рамках указанных соответствий.

Элементы множества Pg будем называть узлами бинарной информационной конструкции, а элементы множества Kg – дугами бинарной информационной конструкции. Элементы множества Pv – знаки первичных элементов исходной реляционной структуры, элементы множества Pd – знаки элементов неопределенного типа исходной реляционной структуры, элементы множества Pk – знаки связок (упорядоченных или неупорядоченных) исходной реляционной структуры. Подчеркнем, что здесь имеется в виду не связка бинарной конструкции, а связка эквивалентной реляционной структуры общего вида, которая сама в бинарную конструкцию не входит, а входит знак (имя) этой связки.

Как было отмечено, отношение p трактуется как **отношение принадлежности** элемента множеству. Каждое из остальных отношений, входящих в семейство Ri , есть отношение принадлежности элемента кортежу под определенным атрибутом (каждому такому отношению соответствует свой атрибут принадлежности). Итак, переход от реляционной структуры общего вида к эквивалентной бинарной конструкции суть:

- 1) замена первичных элементов и элементов неопределенного типа на их знаки;
- 2) сведение произвольного набора атрибутов к двум атрибутам;
- 3) сведение вторичных элементов произвольного вида к бинарным кортежам;
- 4) сведение произвольного набора отношений к унарным и бинарным отношениям.

В процессе такого перехода атрибуты исходной реляционной структуры "превращаются" в соответствующие отношения принадлежности, вторичные элементы – в первичные, отношения исходной реляционной структуры – в унарные отношения, элементы неопределенного типа – в первичные.

Нетрудно заметить, что рассмотренный переход от реляционных структур общего вида к эквивалентным бинарным конструкциям есть своего рода декомпозиция реляционных структур, в результате чего неявно заданное в реляционной структуре общего вида отношение принадлежности, скрытое внутри вторичных элементов реляционной структуры (знаков множеств и знаков кортежей), приобретает в рамках бинарной конструкции явное очертание. Более того, отношение принадлежности со всеми его модификациями может интерпретироваться как топологическое присоединение (того или иного вида) одного элемента бинарной конструкции к другому. Это обстоятельство является очень важным, так как позволяет говорить не только об алгебраических и теоретико-множественных свойствах бинарных конструкций, но и об их топологических свойствах. Кроме того, простая топологическая интерпретация

бинарных конструкций, в частности простая возможность их графического изображения, позволяет говорить о переработке бинарных конструкций более конструктивно, чем о переработке реляционных структур общего вида.

Так, например, первичным элементам бинарной информационной конструкции можно поставить в соответствие элементы памяти, в которой эта конструкция хранится и перерабатывается. Тогда набор унарных отношений, которым принадлежит первичный элемент (т.е. набор его меток), есть не что иное, как текущее состояние соответствующего ему элемента памяти. А набор дуг (пар) принадлежно-сти, которыми первичный элемент связан с другими элементами, есть не что иное, как система связей, которыми соответствующий элемент памяти связан с другими элементами памяти в текущий момент времени. При этом в процессе переработки бинарной информационной конструкции в самом общем случае может происходить как изменение состояния элементов памяти, так и изменение связей между ними.

Переход от реляционных структур общего вида к эквивалентным бинарным информационным конструкциям можно считать одним из способов упрощения (канонизации) реляционных структур. Дальнейший анализ бинарных информационных конструкций на предмет их упрощения приводит к понятию однородных информационных конструкций – реляционных структур, не имеющих меток и имеющих единственное отношение (бинарное асимметричное отношение принадлежности).

Определение 1.2.3.3. Информационную конструкцию Gg , описывающую реляционную структуру G и представляющую собой кортеж, множество элементов которого в соответствии с их атрибутами разбивается на семейство подмножеств $\square Pg, Ag, Kg, Rg, Dg \square$, будем называть **однородной информационной конструкцией**, если выполняются следующие условия:

- 1) элементы множества Pg взаимно однозначно соответствуют элементам описываемой реляционной структуры G . Каждый элемент множества Pg , т.е. каждый первичный элемент конструкции Gg , считается семантически эквивалентным соответствующему элементу описываемой реляционной структуры G ;
- 2) множество Ag фиксировано: $Ag = \{ 1_-, 2_- \}$;
- 3) связки реляционной структуры Gg представляют собой либо простые бинарные кортежи принадлежности, имеющие вид $\square 1_- : k, 2_- : q \square$, где $k \in Pk, q \in Pg, (Pg = Pv \ 0 \ Pk \ 0 \ Pd)$, либо бинарные метакортежи принадлежности, имеющие вид $\square 1_- : k, 2_- : c \square$, где $k \in Pk, c \in Kg$. При этом кортеж c может быть как простым кортежем, так и метакортежем. Связки однородной конструкции будем называть дугами этой конструкции (соответственно простыми дугами и метадугами);
- 4) $Rg = \{ Kg \}$. Семейство отношений реляционной структуры Gg включает в себя единственное отношение – отношение принадлежности, представляющее собой множество знаков всех бинарных асимметричных кортежей принадлежности.

Утверждение 1.2.3.2. Каждую бинарную информационную конструкцию можно представить в виде эквивалентной однородной информационной конструкции.

Иными словами, любую бинарную информационную конструкцию можно свести к реляционной структуре с единственным отношением – бинарным асимметричным отношением принадлежности.

Это утверждение легко доказывается на основании следующего определения эквивалентности бинарной информационной конструкции и однородной информационной конструкции.

Определение 1.2.3.4. Пусть Gg – бинарная информационная конструкция, представляющая собой кортеж, определяемый семейством множеств $\square Pg, Ag, Kg, Rg, Dg \square$, удовлетворяющих условиям:

- 1) $Rg = Rt \ 0 \ Ri, \quad Rt / Ri = \emptyset, \quad Ri = \{ P \} \ 0 \ Ria$.
Здесь p – отношение принадлежности элемента множеству, обозначаемому некоторым узлом в рамках исходной бинарной информационной конструкции;
- 2) $Dg = \emptyset$,
а Gq – однородная информационная конструкция, представляющая собой кортеж, определяемый семейством множеств $\square Pq, Aq, Kq, Rq, Dq \square$, удовлетворяющих условиям:

$$1) Pq = Pqv \ 0 \ Pqk, Pqv / Pqk = \emptyset, \\ Pqk = Pqrt \ 0 \ Pqri, Pqrt / Pqri = \emptyset.$$

Здесь Pqv – множество предметных узлов однородной конструкции, Pqk – множество знаков множеств и знаков кортежей;

$$2) Kq = Kqg \ 0 \ Kqrt \ 0 \ Kqri, Kqg / Kqrt = \emptyset, \\ Kqg / Kqri = \emptyset, Kqrt / Kqri = \emptyset;$$

$$3) Dq = \emptyset.$$

Будем говорить, что бинарная информационная конструкция Gg и однородная информационная конструкция Gq эквивалентны тогда и только тогда, когда существуют взаимно однозначные соответствия между Pg и Pqv ($Pqv \subseteq Pg$), между Kg и Kqg ($Kqg \subseteq Kg$), между Rt и $Pqrt$ ($Rt \subseteq Rg, Pqrt \subseteq Pq$), между Ria и $Pqri$ ($Ria \subseteq Rg, Pqri \subseteq Pq$) и при этом выполняются следующие условия:

1) если $k !; 1_ : x, 2_ : y; k \in Kg$, то $k^* !; 1_ : x^*, 2_ : y^*; k^* \in Kqg$ и наоборот. Здесь k^*, x^*, y^* – образы объектов k, x, y в рамках указанных выше соответствий;

2) если $k \in Kg, k \in r, r \in Ria$, то $k^* \in Kqg, r^* \in Pqri$, $1_ : r^*, 2_ : k^* \in Kqri$ и наоборот. Здесь k^*, r^* – образы объектов k, r в рамках указанных соответствий;

3) если $t \in Pg, t \in r, r \in Rt$, то $t^* \in Pqv, r^* \in Pqrt$, $1_ : r^*, 2_ : t^* \in Kqrt$ и наоборот. Здесь t^*, r^* – образы объектов t, r .

Итак, в результате перехода от бинарной информационной конструкции произвольного вида к однородной информационной конструкции каждое унарное отношение исходной бинарной информационной конструкции "превращается" в первичный элемент, являющийся знаком соответствующего множества первичных элементов, а каждое бинарное отношение r исходной бинарной информационной конструкции, относящееся к классу Ria , "превращается" в первичный элемент, являющийся знаком соответствующего множества пар (дуг) принадлежности, каждая из которых связывает знак некоторого кортежа исходной реляционной структуры общего вида с некоторым элементом указанного кортежа, имеющим в рамках этого кортежа атрибут, соответствующий отношению r .

Следует подчеркнуть то, что все вторичные элементы (дуги) однородной информационной конструкции имеют одинаковую и четкую теоретико-множественную интерпретацию. Каждая такая дуга связывает знак некоторого множества (из которого дуга выходит) с одним из элементов этого множества.

Если от однородной информационной конструкции осуществить точно такой же переход, какой мы осуществляли от произвольной реляционной структуры к эквивалентной бинарной информационной конструкции, то получится информационная конструкция, которую назовем предельной. В предельной информационной конструкции дуги (пары) принадлежности (как простые дуги, так и метадуги) будут сведены к кортежам инцидентности, каждый из которых связывает знак дуги принадлежности либо со знаком некоторого множества, либо с первичным элементом, принадлежащим этому множеству. Соответственно этому отношению принадлежности однородной информационной конструкции (не являющееся простым отношением) сводится к двум (простым) бинарным отношениям инцидентности. Одно из этих отношений связывает знаки дуг принадлежности со знаками множеств (т.е. с элементами, из которых эти дуги выходят). А каждая пара инцидентности, принадлежащая другому из этих отношений, связывает знак дуги принадлежности с первичным элементом предельной информационной конструкции, в который эта дуга входит, т.е. с элементом множества, обозначаемого первичным элементом, из которого указанная дуга выходит.

В основе дальнейшего рассмотрения графодинамических моделей обработки информации используются однородные информационные конструкции в силу того, что они имеют очень простую базовую семантическую интерпретацию, носящую теоретико-множественный характер и непосредственно не зависящую от описываемой предметной области.

О перспективности рассмотренных в данном пункте бинарных и однородных информационных конструкций свидетельствует большой интерес к бинарным представлениям (бинарным моделям) баз данных [80; 539; 401] (Вагин В.Н. 1989кн–ДедукиО ; Цаленко М.Ш. 1989кн–Модел СвБД ; Поспелов Д.А. ред. 1990спр–ИскусИ–К2).

Рассматривая соотношение произвольной описываемой реляционной структурой с некоторой эквивалентной символьной информационной конструкцией (см. определение 1.2.2.2), а также с эквивалентной бинарной информационной конструкцией (см. определение 1.2.3.2) и с эквивалентной однородной информационной конструкцией (см. определение 1.2.3.4), мы фактически определяем денотационную семантику произвольных информационных конструкций, относящихся 1) к одному из возможных символьных фактографических языков, 2) к графовому фактографическому языку бинарных информационных конструкций и 3) к графовому фактографическому языку однородных информационных конструкций.

Особенность перечисленных языков заключается в том, что, во-первых, все они являются фактографическими языками и, во-вторых, все они обеспечивают представление фактографической (экстенциональной) информации о реляционных структурах любого вида. Графовый фактографический язык бинарных информационных конструкций и графовый фактографический язык однородных информационных конструкций относятся к классу графовых семантических языков, которые будут рассмотрены в пункте 1.2.6.

1.2.4. Денотационная семантика текстов

Ключевые понятия: денотационная семантика информационной конструкции, знак, Базовая семантическая информационная конструкция, семантическая сеть.

Каждый информационный объект является моделью (описанием) некоторой предметной области и, следовательно, имеет определенные связи с описываемой предметной областью. Внутренняя структура информационного объекта определяется соответствующей информационной конструкцией. Внутренняя структура предметной области определяется соответствующей реляционной структурой. Совокупность связей информационной конструкции с реляционной структурой, определяющей структуру описываемой предметной области, будем называть **денотационной семантикой информационной конструкции**.

Ключевым понятием денотационной семантики информационных конструкций является понятие **знака**. Знак – это минимальный семантически значимый фрагмент информационной конструкции. Знак может быть представлен либо элементом информационной конструкции, либо фрагментом информационной конструкции, состоящим из нескольких элементов. Знак в рамках информационной конструкции представляет (заменяет, изображает) нечто из описываемой предметной области. Этим нечто может быть либо конкретный предмет (объект) описываемой предметной области, либо конкретная связь, имеющая место в описываемой предметной области (это может быть связь между объектами, другими связями, понятиями, фрагментами предметной области), либо конкретный фрагмент описываемой предметной области, либо конкретное понятие этой предметной области. То, что обозначается знаком, называется его денотатом или денотационной семантикой этого знака. Множество знаков, входящих в состав информационной конструкции, однозначно соответствует множеству своих денотатов, т.е. каждому знаку не может соответствовать несколько денотатов. При этом указанное соответствие между множеством знаков, входящих в информационную конструкцию, и множеством их денотатов не обязано быть взаимно однозначным. Таким образом, разные знаки могут иметь один и тот же денотат, т.е. обозначать одно и то же. Такие знаки будем называть синонимичными.

Рассмотрение денотационной семантики информационных конструкций различного вида основано на понятии базовой семантической информационной конструкции.

Определение 1.2.4.1. Базовая семантическая информационная конструкция – это такая информационная конструкция, которая изоморфна некоторому фрагменту реляционной структуры, определяющей структуру описываемой предметной области. Все элементы базовой семантической информационной конструкции являются знаками, семантически эквивалентными соответствующим элементам указанного представляемого (кодируемого) фрагмента реляционной структуры.

Информационная конструкция называется семантической информационной конструкцией или **семантической сетью**, если морфизм между этой конструкцией и семантически эквивалентной ей базовой семантической информационной конструкцией является взаимно однозначным соответствием между множествами элементов указанных конструкций:

- 1) каждый первичный элемент базовой семантической конструкции является знаком, обозначающим некоторый конкретный предмет описываемой предметной области и взаимно однозначно соответствующим этому предмету;
- 2) каждый знак связки базовой семантической конструкции обозначает некоторую конкретную связь, имеющую место в описываемой предметной области, и взаимно однозначно соответствует этой связи;
- 3) каждый знак атрибута базовой семантической конструкции взаимно однозначно соответствует некоторому относительному понятию (множеству однотипных ролей, выполняемых в рамках определенных связей соответствующими компонентами этих связей);
- 4) каждый знак отношения базовой семантической конструкции взаимно однозначно обозначает некоторое понятие описываемой предметной области (некоторое множество знаков аналогичных в том или ином смысле предметов или связей).

Конкретная связь, имеющая место в описываемой предметной области, не обязательно должна быть простой, т.е. не обязательно должна быть связью между предметами описываемой предметной области. В число связей описываемой предметной области входят также некоторые фрагменты этой предметной области, состоящие из некоторого количества предметов, некоторого количества связей. К числу связей относятся также и ролевые структуры (фреймы).

Построение базовой семантической информационной конструкции, соответствующей описываемой предметной области, является важнейшим этапом формального описания предметной области, а также важнейшим этапом формирования базы знаний.

В качестве примера рассмотрим базовую семантическую информационную конструкцию, соответствующую геометрии Евклида. Первичными элементами этой конструкции являются знаки конкретных геометрических точек, конкретных прямых, конкретных плоскостей. Вторичными элементами этой конструкции являются знаки конкретных отрезков, треугольников, окружностей и т.д. Каждая такая геометрическая фигура трактуется как множество точек, удовлетворяющее определенным требованиям. Вторичными элементами рассматриваемой информационной конструкции являются также знаки связей инцидентности, конгруэнтности; связей сравнения по длине, по площади, по объему; связей, каждая из которых связывает тройку точек, одна из которых лежит между двумя другими, и т.д. Отношениям рассматриваемой информационной конструкции соответствуют такие абсолютные понятия, как "быть геометрической точкой", "быть прямой", "быть отрезком", "быть треугольником", "быть инцидентными геометрическими фигурами", "быть конгруэнтными геометрическими фигурами" и т.д. Атрибутами рассматриваемой информационной конструкции являются такие относительные понятия, как "быть более длинной линией", "быть менее длинной линией", "быть точкой, лежащей между" и т.д.

Таким образом, построить базовую семантическую информационную конструкцию описываемой предметной области – это, прежде всего, сформировать понятийный аппарат указанной предметной области, что требует глубокого анализа этой предметной области.

В основе нашего рассмотрения денотационной семантики информационных конструкций лежит следующее свойство базовой семантической информационной конструкции.

Утверждение 1.2.4.1. Любую фактографическую информацию (любые экстенциональные знания) о любой предметной области можно представить в виде фрагмента базовой семантической информационной конструкции.

Если от базовой семантической информационной конструкции перейти к эквивалентным информационным конструкциям частного вида, то получатся различные виды семантических конструкций – семантическая бинарная конструкция, семантическая однородная конструкция. Введенные семантические конструкции различного вида можно считать различными вариантами уточнения понятия семантической сети.

Если рассматривать множество информационных конструкций, эквивалентных некоторой базовой семантической информационной конструкции, то последнюю можно считать формальным уточнением денотационной семантики указанных конструкций.

Важнейшим свойством семантических конструкций различного вида является их ассоциативность, т.е. наличие достаточно простой процедуры, позволяющей для любых объектов или связей описываемой предметной области выделить их окрестность по отношению принадлежности. Другими словами, ас-

социативность семантических конструкций – это наличие простых процедур, с помощью которых легко находятся ответы на следующие вопросы:

- какими связями связан данный предмет с другими предметами;
- в каких связях заданный предмет выполняет заданную роль;
- какие предметы и под какими атрибутами участвуют в заданной связи;
- каким классам принадлежит заданный предмет или заданная связь;
- какими связями связано между собой некоторое число заданных предметов и (или) связей.

Такая ассоциативность семантических конструкций становится возможной благодаря тому, что каждый знак (знак какого-либо предмета, какой-либо связи, какого-либо понятия) в каждую семантическую конструкцию входит однократно.

Итак, семантическая конструкция есть такая информационная конструкция (общего или частного вида), в которой:

- 1) в качестве знаков используются элементы этой конструкции, а не более сложные ее фрагменты, состоящие из нескольких элементов;
- 2) отсутствует синонимия знаков, т.е. разные знаки не могут иметь совпадающие денотаты.

Очевидно, далеко не каждая информационная конструкция является семантической конструкцией.

Классическим примером несемантических информационных конструкций являются символьные конструкции (см. пункт 1.2.2.). Во-первых, знаки в символьной конструкции представлены идентификаторами – фрагментами символьной конструкции, состоящими в общем случае из нескольких символов. Это обусловлено тем, что алфавит символов в символьных конструкциях конечен и обычно состоит из небольшого числа унарных отношений, заданных на первичных элементах символьных конструкций, тогда как количество предметов, связей, понятий, обозначаемых знаками информационных конструкций, является, в общем случае, неограниченным. Во-вторых, в символьных конструкциях при описании нетривиальных предметных областей принципиально невозможно избежать синонимии знаков, т.е. многократного вхождения в символьную конструкцию знаков, обозначающих одно и то же.

Обычно синонимичные знаки в символьных конструкциях представляются равными (совпадающими, одинаковыми) строками символов. Неравные строки символов также могут быть синонимичными знаками. В символьных конструкциях встречаются омонимичные идентификаторы, т.е. идентификаторы, которые являются равными (совпадающими) строками символов, но имеют разную денотационную семантику.

В основе дальнейшего рассмотрения графодинамических моделей обработки информации будем использовать однородные информационные конструкции. Благодаря их однородности существенно сокращается номенклатура механизмов обработки этих конструкций на низших уровнях.

Первым семантическим свойством однородных информационных конструкций является то, что они имеют простую базовую семантическую интерпретацию, носящую теоретико-множественный характер и непосредственно не зависящую от описываемой предметной области. Каждый первичный элемент однородной информационной конструкции является либо знаком некоторого предмета описываемой предметной области (первичные элементы будем называть знаками предметов или предметными узлами), либо знаком некоторого множества, состоящего из первичных и (или) вторичных элементов однородной информационной конструкции. Таким образом, все первичные элементы (узлы) однородной информационной конструкции являются знаками двух типов: знаками предметов или знаками множеств. Причем элементами указанных множеств могут быть только первичные и вторичные элементы однородной информационной конструкции. Каждый вторичный элемент (дуга) однородной информационной конструкции является позитивным высказыванием о принадлежности некоторого первичного или вторичного элемента однородной информационной конструкции некоторому множеству, обозначаемому узлом, из которого указанная дуга выходит (т.е. тем узлом, который является первым компонентом этой дуги).

Таким образом, множество дуг однородной информационной конструкции определяет классифицирующее отношение, аналогичное АКО-отношению (a kind of...) и ISA-отношению (is a...), которые рассмотрены в работе [242] (*Кандрашина Е.Ю.. 1989кн-Предс3оВиП*). Эти классифицирующие отношения не следует путать с классифицирующим отношением, которое связывает знаки множеств не с элементами этих множеств, а со знаками их подмножеств (подклассов). Такое классифицирующее отношение будем на-

зывать родовидовым отношением. Классифицирующие отношения являются основой для наследования свойств, что играет важную роль в решении большого числа задач.

Вторым семантическим свойством однородных информационных конструкций является то, что они всегда представляют собой семантические информационные конструкции, причем даже тогда, когда они являются информационными метаконструкциями, описывающими структуру несемантических информационных конструкций. Так, например, переходя от символьной информационной конструкции к ее эквивалентному представлению в виде однородной информационной конструкции, эта символьная информационная конструкция "превращается" в семантическую сеть, которая описывает синтаксис указанной символьной конструкции и которую можно легко нарастить семантической сетью, аналогичным образом устроенной и описывающей семантику этой символьной конструкции.

Итак, однородные информационные конструкции всегда являются семантическими информационными конструкциями. При этом по семантическим признакам выделяется два класса однородных информационных конструкций:

- 1) однородные информационные конструкции, описывающие структуру (синтаксис) информационных конструкций, не являющихся семантическими. Денотатами (описываемыми предметными областями) таких однородных метаконструкций являются реляционные структуры, определяющие внутреннюю структуру несемантических информационных конструкций;
- 2) однородные информационные конструкции, описывающие структуру семантических информационных конструкций, которые, в свою очередь, описывают произвольные предметные области. Денотатами таких однородных информационных конструкций можно считать как указанную семантическую информационную конструкцию, так и предметную область, описываемую этой семантической информационной конструкцией.

Исследуя семантические свойства информационных конструкций, можно говорить о фактографических и логических высказываниях (фактографических и логических информационных конструкциях). Фактографические высказывания непосредственно представляют структуру описываемой предметной области, т.е. представляют конкретные факты, имеющие место в описываемой предметной области. В отличие от этого логические высказывания представляют информацию о свойствах и законах описываемой предметной области. Благодаря этому появляется возможность компактного описания бесконечных предметных областей с помощью информационных конструкций, состоящих из конечного числа элементов.

Отличительными особенностями логических высказываний по сравнению с фактографическими являются:

- 1) появление логических переменных наряду со знаками конкретных предметов и связей предметной области (указанные знаки иногда называют логическими константами). Семантически каждая логическая переменная является знаком произвольного элемента из некоторого множества, которое может содержать как логические константы, так и логические переменные. Указанное множество элементов называют областью возможных значений соответствующей переменной;
- 2) появление простых (атомарных) высказываний и сложных (конъюнктивных, дизъюнктивных, имплицативных) высказываний;
- 3) появление позитивных, негативных и нечетких высказываний;
- 4) появление кванторных высказываний, т.е. высказываний, свободные переменные которых связываются кванторами (кванторами существования, кванторами всеобщности и т.д.);
- 5) появление формальных теорий для стационарных и нестационарных предметных областей. Формальная теория – это конъюнктивное высказывание, изначально считающееся истинным. Соответственно этому позитивность, негативность и нечеткость компонентов (логических множителей) формальных теорий определяет истинные, ложные и неопределенные высказывания в рамках данной формальной теории.

Основным принципом представления информации о законах какой-либо предметной области (интенциональных знаний о предметной области) является формальное рассмотрение законов не самой описываемой предметной области, а универсального фактографического высказывания, которое включает в себя всю фактографическую информацию описываемой предметной области. Таким образом, логические высказывания по своей сути являются информационными метаконструкциями, т.е. конструкциями, описывающими другие информационные конструкции.

Для рассмотренных выше информационных конструкций фактографического типа выделим следующие классы:

- 1) базовые семантические информационные конструкции;
- 2) семантические конструкции различного вида, не являющиеся базовыми (в первую очередь, однородные информационные конструкции);
- 3) символные конструкции.

Совершенно аналогичные классы можно выделить и для информационных конструкций логического типа. Для этого в перечисленные фактографические высказывания необходимо дополнительно ввести средства представления логических переменных, логических связей, кванторов, высказываний, формальных теорий.

1.2.5. Классификация языков

Ключевые понятия: символный язык, графовый язык, семантический язык, сложноструктурированная предметная область, фактографический язык, логический язык, нестационарная предметная область, семантическая мощность языка.

С формальной точки зрения каждый конкретный язык представляет собой некоторое множество (являющееся бесконечным для практически интересных языков) информационных конструкций, которые имеют общие синтаксические свойства (т.е. общие принципы своего внутреннего устройства), а также общие денотационно-семантические свойства (т.е. общие принципы своего соотношения с описываемыми предметными областями). Общие синтаксические свойства информационных конструкций, принадлежащих языку, будем называть синтаксисом этого языка, а общие денотационно-семантические свойства информационных конструкций, принадлежащих языку, – денотационной семантикой этого языка. Синтаксис языка чаще всего задается как конструктивное определение множества так называемых синтаксически правильных (правильно построенных) конструкций соответствующего языка. Множество синтаксически правильных конструкций языка – это условное расширение множества конструкций, принадлежащих данному языку. Конструктивное определение множества синтаксически правильных конструкций языка задается следующими множествами:

- 1) множеством элементарных (атомарных, примитивных) конструкций;
- 2) множеством синтаксических правил, указывающих, как из имеющихся синтаксически правильных конструкций можно построить также синтаксически правильную конструкцию.

Денотационная семантика языка чаще всего формально задается как метод (в частности, алгоритм) перевода произвольной конструкции описываемого языка (языка-объекта) на некий другой язык, денотационная семантика которого считается известной и является достаточно простой. Языками с наиболее простой денотационной семантикой являются языки, конструкции которых представляют собой семантические конструкции.

С прагматической точки зрения каждому языку можно поставить в соответствие присущий только этому языку метод представления информации, являющейся описанием некоторого класса областей, т.е. метод построения информационных конструкций. Каждый такой метод ограничивает (уточняет) вид (внутреннее строение) информационных конструкций и соотношение информационных конструкций с описываемыми предметными областями.

Классификация языков осуществляется в соответствии с тем, какими синтаксическими и семантическими особенностями обладают информационные конструкции, принадлежащие этим языкам.

Первый признак классификации языков – линейность языковых конструкций. По этому признаку языки делятся на символные (линейные) и графовые (нелинейные, сетевые). **Символьные языки** – это языки, которым принадлежат только символные конструкции. Все остальные языки будем называть графовыми.

Во множестве **графовых языков**, в свою очередь, можно выделить следующие подклассы: языки информационных конструкций общего вида, языки бинарных информационных конструкций, языки однородных информационных конструкций, языки предельных информационных конструкций. Из сказанного ранее (см. пункт 1.2.3.) следует, что от каждого графового языка, принадлежащего одному из перечисленных классов, достаточно легко перейти к эквивалентному графовому языку, который принадлежит любому другому из перечисленных классов графовых языков.

Важнейшим свойством символьных языков является то, что для каждого такого языка при конечном фиксированном для всех конструкций языка алфавите символов (разным языкам могут соответствовать разные алфавиты) обеспечивается возможность формирования неограниченного количества знаков, соответствующих неограниченному количеству денотатов.

Предложенный нами язык однородных информационных конструкций также обеспечивает открытый характер при одинаковом для всех языковых конструкций наборе отношений, задающих эти конструкции.

По второму признаку классификации языки делятся на семантические и несемантические. **Семантические языки** – это языки, которым принадлежат только семантические конструкции. Все остальные языки считаются несемантическими.

Символьная информационная конструкция может быть семантической сетью, если описываемая предметная область имеет линейную структуру и если связи между символами в символьной информационной конструкции ставятся в соответствие связям между предметами в описываемой предметной области. Но далеко не каждая фактографическая конструкция может быть одновременно и семантической информационной конструкцией, и символьной информационной конструкцией. В частности, не существует символьного семантического языка, обеспечивающего эквивалентное представление произвольных информационных конструкций общего вида.

Третий признак классификации языков – степень сложности структур описываемых предметных областей. По этому признаку выделяют языки, ориентированные на описание сложноструктурированных предметных областей. В **сложноструктурированных предметных областях** отсутствуют ограничения на вид связей. В этих предметных областях связи могут быть между предметами, связями, фрагментами предметной области, относительными понятиями, абсолютными понятиями.

Четвертый признак классификации языков – наличие средств описания свойств предметных областей. По этому признаку языки делятся на фактографические и логические. **Фактографический язык** – это язык, обеспечивающий представление только фактографической информации (экстенциональных знаний), т.е. информации о фактах, имеющих место в описываемой предметной области [242] (*Кандрашина Е.Ю.. 1989кн-ПредсЗоВиП*). Следовательно, фактографическому языку принадлежат только фактографические высказывания. Логический язык – это язык, обеспечивающий представление не только фактографической информации, имеющей отношение к некоторой предметной области, но и информации о свойствах и законах этой предметной области. Такую информацию называют интенциональными знаниями [242] (*Кандрашина Е.Ю.. 1989кн-ПредсЗоВиП*). **Логический язык** – это язык, которому принадлежат как фактографические, так и логические высказывания (как фактографические, так и логические информационные конструкции).

Особенностями логических языков, отличающими их от языков фактографических, являются наличие логических переменных, наличие сложных (конъюнктивных, дизъюнктивных, имплицативных) высказываний, негативных и нечетких высказываний, кванторных высказываний, наличие формальных теорий. Логические языки могут быть самыми различными как по синтаксическим, так и по семантическим особенностям. Классическими примерами логических языков являются языки предикатов 1-го порядка (языки 1-й степени) и языки предикатов 2-го порядка (языки 2-й степени) [316] (*Мальцев А.И. 1970кн-АлгебС*).

Пятый признак классификации языков – наличие фактора времени. По этому признаку языки делятся на языки описания стационарных предметных областей и языки описания нестационарных предметных областей, т.е. предметных областей, состояние которых меняется во времени. **Нестационарная предметная область** формально трактуется как множество упорядоченных во времени стационарных предметных областей, называемых состояниями (ситуациями). Каждому состоянию нестационарной предметной области ставится в соответствие информационная конструкция, описывающая это состояние. В целом описание нестационарной предметной области есть метаописание системы информационных конструкций, каждая из которых описывает одно из состояний нестационарной предметной области. Следовательно, язык описания нестационарных предметных областей должен включать в себя язык описания стационарных предметных областей. Описание нестационарной предметной области можно также трактовать как метаописание некоторой нестационарной информационной конструкции. При этом за основу такого метаописания можно брать не только систему состояний нестационарной информационной конструкции, но и систему взаимодействующих переходных процессов, осуществляющих переход от одного состояния к другому. Важнейшим классом языков описания нестационарных предметных областей являются языки процедурного программирования, описывающие преобразование информационных конструкций.

Шестой признак классификации языков – наличие метаязыковых средств, достаточных для полного самоописания языка. По этому признаку языки делятся на стратифицированные, у которых указанные средства отсутствуют, и нестратифицированные, которые являются собственными метаязыками и которые, следовательно, не имеют четкой грани между языком-объектом и метаязыком, а также между информационной конструкцией-объектом и метаконструкцией.

Седьмой признак классификации языков – семантическая мощность. **Семантическая мощность языка** – это все то, что может быть описано с помощью конструкций этого языка. Языки с неограниченной семантической мощностью, т.е. языки, в которых имеется потенциальная возможность представления любой информации, будем называть универсальными. Согласно рассматриваемому признаку классификации, языки делятся на универсальные и специализированные. В свою очередь, специализированные языки могут сравниваться по своей семантической мощности, т.е. по многообразию описываемых предметных областей, а также по многообразию свойств, описываемых в этих областях. Универсальные языки можно рассматривать как результат интеграции всевозможных специализированных языков, поэтому при создании универсального языка принципиально важной является разработка методов интеграции специализированных языков в рамках создаваемого универсального языка. В основе такой интеграции, в частности, может лежать выделение языка-ядра, общего для всех интегрируемых языков. Универсальные языки по определению должны быть языками, ориентированными на описание сложноструктурированных предметных областей, логическими языками, языками описания нестационарных предметных областей, нестратифицированными языками.

Подчеркнем, что языки представления знаний для интеллектуальных систем нового поколения должны быть универсальными, а также открытыми языками, т.е. языками, позволяющими достаточно легко интегрировать самые различные специализированные языки.

Важнейшим классом специализированных языков являются языки программирования, включающие в себя языковые средства представления самих программ и языковые средства представления информационных конструкций (данных), которые перерабатываются в процессе реализации этих программ. Каждая программа есть описание некоторого метода решения произвольной задачи из определенного класса задач. Существуют два принципиально разных класса методов решения задач – процедурные и не процедурные методы. Процедурный метод решения произвольной задачи из определенного класса задач – это явная декомпозиция всего процесса решения задачи на иерархическую систему более простых и, в конечном счете, элементарных процессов. Таким образом, описание процедурного метода решения некоторого класса задач (такое описание будем называть процедурной программой) есть не что иное, как описание некоторого класса нестационарных информационных конструкций. Следовательно, языки процедурного программирования следует отнести к классу языков описания нестационарных предметных областей (см. пятый признак классификации языков). В отличие от этого не процедурный метод решения произвольной задачи из определенного класса задач явно не рассматривает сам процесс решения задачи, а представляет собой информационную конструкцию, которая, будучи соединена с исходными данными задачи из указанного класса, предоставляет решателю (абстрактной машине) дополнительную информацию, достаточную для решения произвольной задачи соответствующего класса. При этом способ использования решателем этой дополнительной информации может быть самый различный.

Важным классом специализированных языков для интеллектуальных систем являются языки спецификаций программ. Язык спецификаций программ – это метаязык, обеспечивающий описание программ, и в частности описание денотационной семантики программ, т.е. формальное определение класса задач, решаемых с помощью каждой программы. Наличие описаний спецификаций программ, имеющихся в памяти интеллектуальной системы, дает возможность интеллектуальной системе найти для появившейся у нее задачи программу, обеспечивающую решение этой задачи, если, конечно, такая программа имеется.

1.2.6. Семантические сети и семантические графовые языки

Ключевые понятия: графовый язык, графовый семантический язык представления знаний.

Семантические языки, т.е. языки, которым принадлежат только семантические информационные конструкции, являются основным объектом исследований в данной работе. Все семантические языки с нетривиальной семантической мощностью являются **графовыми языками**.

Можно говорить о целом семействе семантических графовых языков, называемых также языками семантических сетей. В частности, в соответствии с типологией семантических информационных конструкций, рассмотренных выше, можно выделить следующие семантические языки: язык базовых семантических информационных конструкций, язык бинарных информационных конструкций, язык однородных информационных конструкций.

Примерами графовых семантических языков (как специализированных, так и языков, претендующих на универсальность) также являются:

- предложенный В.С.Лозовским сетевой язык представления фреймов [401] (*Поспелов Д.А.ред.1990спр-ИскусИ-К2*);
- язык растущих пирамидальных сетей [125] (*Гладун В.П.1987кн-ПланиР*);
- способ организации семантических сетей, используемый В.Н.Вагиным для поддержки параллельного логического вывода [80; 401] (*Вагин В.Н.1989кн-ДедукиО ; Поспелов Д.А.ред.1990спр-ИскусИ-К2*);
- способы организации семантических сетей, рассмотренные в работе [445] (*Скрэгг П.1983ст-СеманСкМП*);
- используемый Э.Ф.Скороходько сетевой способ описания синтаксиса и семантики текстов естественного языка [444] (*Скороходько Э.Ф. 1983кн-СеманСуАОТ*);
- используемый Р.Шенком сетевой способ описания семантики текстов естественного языка [554] (*Шенк Р.1980кн-ОбрабКИ*);
- графовый язык представления информации в памяти вегетативной машины, предложенной В.Б.Борщевым [66] (*Борщев В.Б.1990ст-ПаралАВ*);
- графовый язык представления информации в памяти абстрактной сетевой машины, предложенной А.М.Степановым [452] (*Степанов А.М.1981пр-ФрейМИПСВ*);
- графовый язык представления алгоритмов А.Н.Колмогорова и данных, перерабатываемых этими алгоритмами [259] (*Колмогоров А.Н..1958ст-кОпреда*);
- графовые языки представления различным образом устроенных продукционных программ (систем продукций), ориентированных на переработку семантических сетей [276] (*Кузнецов В.Е.1989кн-ПредсВЭВМНП*);
- графовые логические языки, являющиеся различными способами сетевого представления логических высказываний, т.е. различными способами введения логических связей и кванторов в семантическую сеть [631; 80] (*Hendrix G.G.1979art-EncodKiP ; Вагин В.Н.1989кн-ДедукиО*).

К достоинствам семантических графовых языков представления знаний, по сравнению с другими способами представления знаний, относятся:

- компактность, обусловленная тем, что в отличие от символического текста в семантической сети знак каждого объекта или понятия описываемой предметной области присутствует только в одном экземпляре и состоит из одного элемента;
- ассоциативность, заключающаяся в существовании простых процедур поиска элементов семантической сети, связанных заданным образом с заданными элементами;
- наличие простой возможности введения метаинформации в семантическую сеть путем простого наращивания исходной семантической сети метасетью без какого-либо изменения исходной семантической сети;
- возможность рассмотрения описываемых предметных областей одновременно на неограниченном числе уровней детализации;
- приспособленность к поддержке структур любого вида, и в частности к поддержке сложноструктурированных знаний;
- приспособленность к интеграции самых различных специализированных языков и самых различных моделей представления знаний;
- приспособленность к представлению различного рода лингвистических знаний (о синтаксисе, о семантике, о прагматике естественных языков), что делает эффективным использование семантических графовых языков для создания естественно-языковых интерфейсных подсистем в интеллектуальных системах;
- приспособленность к параллельной асинхронной переработке знаний, что делает эффективным использование семантических графовых языков для создания интеллектуальных систем, поддерживающих сложноструктурированные знания и сложные логические операции.

Подробнее о достоинствах семантических графовых языков см. в работах [127; 125; 444; 80] (*Гладун В.П.1977кн-ЭврисПвСС ; Гладун В.П.1987кн-ПланиР ; Скоруходько Э.Ф.1983кн-СеманСиАОТ ; Вагин В.Н.1989кн-ДедукиО*).

В данной работе мы будем рассматривать графовый семантический язык представления знаний, ориентированный на описание сложноструктурированных предметных областей, обеспечивающий представление логических высказываний самого различного вида и обладающий высокой степенью открытости. Кроме того, мы будем рассматривать различные графовые языки параллельного асинхронного программирования, легко интегрируемые в состав указанного выше графового семантического языка представления знаний. Напомним при этом, что программы, представляющие собой описания различных методов решения различных классов задач, составляют важнейшую часть баз знаний интеллектуальных систем.

Предлагаемый в данной работе **графовый семантический язык представления знаний**, обладающий указанными выше свойствами, назван языком SCL (Semantic Code Logic) – см. раздел 5. В качестве базового языка представления знаний (языка-ядра), на основе которого осуществляется интеграция всевозможных специализированных языков в состав языка SCL, предлагается язык SC (Semantic Code) – см. раздел 4.

Резюме к подразделу 1.2

В подразделе 1.2 графовые языки рассматриваются в сравнении с символьными языками и в контексте общей типологии языков. Кроме того, формально рассмотрены синтаксические и семантические свойства информационных конструкций, принадлежащих графовым языкам.

В целях формального уточнения понятия предметной области и понятия информационного объекта введено понятие реляционной структуры как обобщение классического понятия алгебраической модели. Рассмотрены наиболее интересные виды информационных конструкций: бинарные, однородные, символьные информационные конструкции.

В целях формального уточнения денотационной семантики информационных конструкций введено понятие базовой семантической информационной конструкции. Денотационная семантика информационной конструкции определяется как ее соотношение с эквивалентной базовой семантической информационной конструкцией. Информационные конструкции произвольной структуры, имеющие наиболее простой вид соотношения с эквивалентными им базовыми семантическими информационными конструкциями, определены как семантические конструкции.

Графовые языки в настоящее время не являются практически используемыми языками, несмотря на их преимущества по сравнению с символьными языками. В данной работе мы пытаемся это устранить. При этом нас будут интересовать не просто графовые языки, а графовые семантические языки, ибо языки именно этого класса дают достаточно "прозрачную" возможность реализации моделей параллельной асинхронной переработки знаний путем их сведения к более простым и в конечном счете к непосредственно реализуемым моделям.

Для того чтобы оценить практическую значимость идеи создания и реализации графовых языков, сделаем небольшой экскурс в становление теории графов как области математики. Особенность теории графов заключается в том, что исследуемые ею математические объекты были хорошо известны и до нее. Это конечные алгебраические модели частного вида. Но на эти алгебраические модели теория графов взглянула, условно говоря, "топологическим" взглядом. И сразу оказался обнаженным для этих структур целый ряд аспектов, имеющих, как скоро выяснилось, большое практическое значение. Конечно, графы как картинки, иллюстрирующие ту или иную задачу, использовались задолго до появления теории графов. Практика показала плодотворность теоретико-графового взгляда на большое количество задач, и не только при постановке задач, но и при рассмотрении процесса их решения. Но все это в настоящее время сдерживается определенной сложностью погружения теоретико-графового взгляда на решаемую задачу в современные технологии программирования, которые в конечном счете скрывают, камуфлируют то, что теоретико-графовый взгляд пытается обнажить. С появлением графовых языков представления знаний и графовых языков программирования это противоречие снимается.

Наиболее острая потребность в графовых языках ощущается в CAD-CAM-задачах, в задачах структурного распознавания, в задачах, связанных с обработкой иерархической сложноструктурированной информации.

В заключение сделаем несколько общих примечаний о графовых структурах.

Примечание 1. Графовый способ представления дискретной информации обладает универсализмом в том смысле, что любое представление дискретной информации можно рассматривать как графовую структуру. Следовательно, есть все основания считать понятие графовой структуры и понятие текста (в общем виде) тождественными понятиями. Действительно, когда мы говорим о графе, мы всегда подразумеваем некую предметную область, которую этот граф описывает, будь то схема улиц города, сеть железных дорог, принципиальная электрическая схема, структура молекулы какого-либо органического соединения или система состояний конечного автомата. А текст всегда можно рассматривать как граф, вершины и связи которого являются элементами текста (разумеется, элементы текста физически могут быть представлены самым различным образом). Так, например, текст, записанный на ленте машины Тьюринга, есть не что иное, как орграф с помеченными вершинами. Вершины этого орграфа есть ячейки ленты, в которые записывается тот или иной символ, а его дуги физически реализуются смежностью (соседством) ячеек ленты и связывают каждую ячейку ленты с непосредственно следующей за ней ячейкой.

Графовая трактовка текста встречается в целом ряде работ по формальным системам. Текстовая трактовка графовой структуры также встречается в ряде работ. Так, например, А.А.Зыков рассматривает графовую структуру как "средство описания тех или иных взаимосвязей между математическими объектами" [228] (*Зыков А.А. 1969 кн-ТеориКТ*).

Итак, графовая структура можно рассматривать как объект или процесс, являющийся дискретной моделью некоторого другого объекта или процесса. Под дискретностью модели здесь понимается не физическое свойство объектов, являющихся дискретными моделями, а тот факт, что для восприятия информации, содержащейся в дискретной модели, достаточно изучить ее строение только до определенного уровня. Так, например, изменение написания какого-либо символа в рукописном тексте, не переходящее грани возможного его написания, не изменяет содержащейся в тексте информации, т.е. не является существенным для текста.

Примечание 2. Одну и ту же информацию можно представлять различными видами графов. Отсюда следуют два вывода:

- необходимо разрабатывать стандарты;
- среди этих стандартов нужно выявлять наиболее удобные способы графowego представления информации.

Примечание 3. До сих пор в основном мы пользовались такими языками, которые являются способами представления дискретной информации в виде линейных (цепочечных) графовых структур. Сюда относятся все естественные языки, подавляющее большинство искусственных языков. Не исключением является и способ представления информации в памяти современных ЭВМ.

Примечание 4. Выход за рамки линейности текстов [317] (*Манин Ю.И. 1979 кн-ДоказИН*) при разработке языков представления дискретной информации представляет значительные удобства.

Нелинейная графовая структура является естественной и удобной формой описания любой сложной системы (стационарной, нестационарной, абстрактной), в которой удается выделить некоторое множество элементов системы и некоторый набор отношений, заданных на этом множестве элементов. Эффективность нелинейного способа описания систем может быть проиллюстрирована следующими примерами.

Формализация описания систем, исследуемых органической химией, привела к графовой форме описания молекулы органических соединений.

Описание электронной системы оказалось удобнее всего строить в виде принципиальной электрической схемы, которая представляет собой не что иное, как граф.

Примечание 5. Среди способов (языков) нелинейного представления дискретной информации наиболее удобным для переработки является представление в виде семантических сетей. Об этом свидетельствует интенсивное развитие теории семантических сетей, которая сформировалась в рамках исследований по системам искусственного интеллекта, системам автоматического перевода с одного естественного языка на другой и которая рассматривается сейчас как формальный аппарат исследования сложных процессов обработки информации (расознавания образов, планирования целенаправленных действий, обнаружения закономерностей и т.д.).

Примечание 6. В рамках способа представления информации при помощи семантических сетей, в свою очередь, также возможен полиморфизм. Это требует разработки стандартных способов представления семантических сетей, т.е. специальных языков семантических сетей. Одним из таких языков является рассматриваемый ниже язык SC (Semantic Code).

Рассмотрение интеллектуальных систем на семантическом уровне является наиболее перспективным подходом к формальному уточнению понятия интеллектуальной системы. Такой подход развивается в работах И.П.Кузнецова [287; 288] (*Кузнецов И.П.1978кн-МеханОСИ ; Кузнецов И.П.1986кн-СеманП*), В.П.Гладуна [125; 127] (*Гладун В.П.1987кн-ПланиР ; Гладун В.П.1977кн-ЭврисПвСС*), Р.Шенка [554] (*Шенк Р.1980кн-ОбрабКИ*), Э.В.Попова [387] (*Попов Э.В.1982кн-ОбщенСЭВМнЕЯ*) и многих других авторов.

Рассмотрение семантического уровня информационно-логической организации интеллектуальных систем позволяет в большой степени абстрагироваться от несущественных деталей технической реализации этих систем и, таким образом, позволяет сконцентрировать внимание на принципиальных особенностях их организации.

Рассмотрение интеллектуальных систем на семантическом уровне прежде всего требует выяснения того, что следует называть семантическим языком представления знаний. Проблема представления (организации) знаний в памяти интеллектуальной системы носит ключевой характер, так как от способа представления знаний, а точнее, от степени близости этого представления к семантическому самым существенным образом зависят простота и удобство их переработки [341; 386; 515; 127] (*Минский М.1978кн-СтрукДПЗ ; Попов Э.В.1976кн-АлгорОИР ; Хант Э.1978кн-ИскусИ ; Гладун В.П.1977кн-ЭврисПвСС*). Суть проблемы представления знаний заключается не в том, чтобы суметь представить знания (представить их можно, например, и на естественном языке), а в том, чтобы представить эти знания в памяти так, чтобы ими можно было достаточно удобно пользоваться. Следовательно, проблему представления и организации знаний нельзя решать в отрыве от логики системы и от организации ее памяти. Так, например, представление знаний существенным образом зависит от способа организации доступа к памяти. Хорошо организованные знания должны быть записаны в память так, чтобы достаточно просто можно было найти требуемую в текущий момент информацию (требуемый фрагмент знаний).

Наиболее перспективной в этом плане является ассоциативная память. При вводе информации в ассоциативную структурно перестраиваемую память не возникает проблемы распределения информации по памяти, так как при обращении к нужному фрагменту информации, хранимой в ассоциативной памяти, не требуется знать, в каком месте памяти находится этот фрагмент. Особенность ассоциативной организации знаний интеллектуальной системы заключается в том, что здесь требуется реализация ассоциативного доступа к таким фрагментам хранимых знаний, которые имеют в общем случае произвольный размер, вид и структуру. Реализация такого доступа к фрагментам знаний оказывается возможной, если знания представить в виде семантической сети. Поэтому семантическую сеть иногда называют ассоциативной сетью. Семантическая сеть (которую иногда также называют семантической моделью предметной области, смысловым графом, концептуальным графом, сетью концептуальной зависимости) представляет собой специальным образом организованную графовую структуру, вершины которой однозначно соответствуют понятиям представляемой предметной области, а связи между вершинами однозначно соответствуют связям между этими понятиями. Преимущество семантических сетей по сравнению с другими способами представления знаний обусловлено их компактностью, однозначностью и ассоциативностью [559] (*Шуберт Л.1979ст-УсилеВМСС*).

О преимуществах семантических сетей свидетельствует удобство использования и широкое распространение теоретико-графовых методов решения различного рода задач [274; 425; 228] (*Кристофидес Н.1978кн-ТеориГ ; Рейнгольд Э..1980кн-КомбиАТиП ; Зыков А.А.1969кн-ТеориКГ*), а также активная разработка языковых [320; 446] (*Маркевичус Р.1977ст-ЯзыкПдОГ ; Скриган Н.И..1979пр-СредсОИГС*), программных [44] (*Белоглазов В.Н..1974ст-СистеАРЗ*) и аппаратных [193] (*Додонов А.Г..1979ас-УстроДИГ*) средств решения теоретико-графовых задач.

Связь теоретико-графовых методов с семантическими сетями обусловлена тем, что теоретико-графовая трактовка задачи всегда неявно подразумевает формализацию перерабатываемых данных не просто в виде некоторой графовой структуры, а в виде такой графовой структуры, которая обладает вполне определенными семантическими свойствами, а именно – является семантической сетью, хотя явно о семантических сетях в теории графов речь не идет. Этим обстоятельством обусловлены естественность, наглядность и удобство теоретико-графовой трактовки задачи, в чем легко убедиться,

сравнив содержательную запись какого-либо теоретико-графового алгоритма с записью программы, обеспечивающей его реализацию на современных ЭВМ.

Кроме традиционных приложений в органической химии, в электротехнике теоретико-графовые методы в настоящее время широко используются в социологии, экономике, теории вероятностей, генетике, математической лингвистике, проектировании дискретных устройств и т.д. Об эффективности теоретико-графовых методов говорят также результаты, полученные при разработке структурного подхода к распознаванию образов, который позволяет сократить сложность процедуры распознавания по сравнению с другими подходами и обеспечивает распознавание в тех случаях, когда другие подходы оказываются неприемлемыми [512] (*Фу К.С.1977кн-СтрукМвРО*). Как отмечается в работе [43] (*Белов В.В..1976кн-ТеориГ*), "возможность приложения теории графов к столь различным областям заложена, в сущности, уже в самом понятии графа, сочетающего в себе теоретико-множественные, комбинаторные и топологические аспекты".

1.3. Абстрактные графодинамические ассоциативные машины

Ключевые понятия: абстрактная машина обработки информации; графодинамическая параллельная асинхронная абстрактная ассоциативная машина.

Цель данного подраздела – рассмотреть графодинамические параллельные асинхронные абстрактные машины с общих позиций, как подкласс абстрактных машин обработки информации, и показать, что этот подкласс абстрактных машин является наиболее перспективным для проектирования интеллектуальных систем нового поколения, а следовательно, и для проектирования компьютеров нового поколения, ориентированных на поддержку интеллектуальных систем.

1.3.1. Абстрактные машины обработки информации и соответствующие им операции, элементарные процессы и микропрограммы

Ключевые понятия: абстрактная машина обработки информации, память, операция.

Как было отмечено выше, каждому языку можно поставить в соответствие его синтаксис, отражающий свойства внутреннего строения конструкций этого языка, и денотационную семантику, отражающую общие свойства соотношения между конструкциями языка и описываемой ими предметной областью. Некоторым языкам можно поставить в соответствие также и методы манипулирования их конструкциями, направленные на решение различных задач. Такие методы манипулирования языковыми конструкциями (модели обработки информации, модели вычислений, модели решения задач) могут быть различными. Так, например, разным языкам программирования и разным языкам представления знаний могут соответствовать абсолютно разные принципы организации решения задач, абсолютно разные "системы мышления" [563] (*ЯзыкПада-1981кн*).

Правила манипулирования языковыми конструкциями, соответствующие некоторому языку и направленные на решение различных задач в рамках этого языка, называются его операционной семантикой [358] (*Ненейвода Н.Н.1983ст-СеманАЯ*). Операционная семантика различных языков может существенно отличаться друг от друга. Более того, некоторым наиболее мощным языкам (например, языкам представления знаний) может быть поставлено в соответствие несколько операционных семантик, т.е. некоторые языки допускают достаточно большую свободу в выборе методов (приемов, стратегий) решения задач.

Для того чтобы уточнить (формализовать) понятие способа организации процесса решения задач, для того чтобы разобраться во всем многообразии таких способов, вводится понятие абстрактной машины обработки информации (абстрактной информационной машины). Важность этого понятия отмечается в ряде работ, в частности в работе [482] (*Тыгу Э.Х.1984кн-КонцеП*). Заметим также, что самому термину "абстрактная машина обработки информации" в других работах могут соответствовать такие термины, как абстрактный решатель задач, абстрактная вычислительная машина, абстрактный вычислитель, гипотетическая информационная машина, виртуальная информационная машина, абстрактный интерпретатор некоторого языка, абстрактная машина манипулирования конструктивными объектами, абстрактный компьютер. Введение понятия абстрактной машины преследует цель с некоторых единых позиций охватить все многообразие моделей вычисления, способов организации процесса решения задач, задаваемых самими различными языками программирования,

языками представления знаний, самыми различными архитектурами вычислительных систем. На основе понятия абстрактной машины различные способы организации вычислительного процесса можно трактовать как различные виды абстрактных машин, что дает формальную основу для сравнения различных способов организации обработки информации.

Итак, задать тот или иной способ организации обработки информации – это задать ту или иную абстрактную машину обработки информации. Не случайно поэтому различные абстрактные машины использовались для различных уточнений понятия алгоритма (машины Тьюринга, машины Поста).

В основе предлагаемой в данной работе трактовки абстрактной машины обработки информации лежит стремление привести к общему виду абстрактные машины самого различного вида. В частности, это выражается в том, что рассматриваемые нами абстрактные машины не содержат никаких устройств управления (которые являются уникальными для каждого вида абстрактных машин) и осуществляют управление взаимодействием операций только через память. Нетрудно показать, что к такому виду можно привести любую абстрактную машину путем формирования специальных управляющих структур в памяти.

Определение 1.3.1.1. Абстрактная машина обработки информации C задается парой $C = \langle CS, W \rangle$, где

CS – некоторым образом организованная **память** (запоминающая среда), в которой хранятся перерабатываемые информационные конструкции;

W – множество **операций** (правил вывода), выполняемых над указанной памятью.

Язык, которому принадлежат информационные конструкции, хранимые и преобразуемые в памяти абстрактной машины обработки информации, и есть тот самый язык, операционная семантика которого определяется этой абстрактной машиной. Указанный язык вместе с соответствующей абстрактной машиной и информационной конструкцией, принадлежащей этому языку и определяющей начальное состояние памяти абстрактной машины, задают конкретную формальную модель обработки информации (см. пункт 1.1.1).

Память абстрактной машины обработки информации с формальной точки зрения есть некоторым образом устроенная нестационарная информационная конструкция, трактуемая как процесс преобразования хранимой в памяти стационарной информационной конструкции, определяющей текущее состояние реализуемой формальной модели. Формальное рассмотрение указанной нестационарной информационной конструкции осуществляется путем декомпозиции этой конструкции на систему определенным образом взаимодействующих между собой непрерываемых элементарных процессов. В разных абстрактных машинах обработки информации эта декомпозиция осуществляется разным образом и в известной мере носит условный (договорной) характер.

Множество всевозможных элементарных процессов абстрактной машины обработки информации разбивается на классы и каждому такому классу ставится в соответствие активный исполнитель, осуществляющий реализацию каждого элементарного процесса из соответствующего класса, если в текущем состоянии перерабатываемой информационной конструкции возникнут необходимые и достаточные условия выполнения этого элементарного процесса. Указанный активный исполнитель будем называть операцией или процессором абстрактной машины обработки информации. Абстрактная машина обработки информации в общем случае может иметь неограниченное число операций. Таким образом, элементарный процесс обработки информации в абстрактной машине – это процесс реализации одной из операций этой абстрактной машины.

В абстрактной машине обработки информации каждой ее операции ставится в соответствие программа, описывающая то, как осуществляется выполнение произвольного элементарного процесса из соответствующего для этой операции класса элементарных процессов. Указанные программы будем называть микропрограммами операций. Микропрограмма каждой операции абстрактной машины должна описывать три этапа выполнения элементарного процесса [80] (*Вагин В.Н. 1989кн–ДедукиО*):

- 1) проверку условия выполнения элементарного процесса, т.е. поиск соответствующего фрагмента перерабатываемой информационной конструкции, над которым указанный элементарный процесс должен быть выполнен;
- 2) собственно выполнение элементарного процесса (преобразование перерабатываемой информационной конструкции);

- 3) оформление результата, включающее в себя формирование информационных конструкций, описывающих тип полученного результата, удаление вспомогательных информационных конструкций, создаваемых только для выполнения данного элементарного процесса.

В абстрактных машинах обработки информации, поддерживающих параллельное и асинхронное выполнение элементарных процессов, первый этап их выполнения может оказаться существенно сложнее второго.

В некоторых элементарных процессах второй и даже третий этапы их выполнения могут отсутствовать. Такие элементарные процессы будем называть нерезультативными. Нерезультативный элементарный процесс – это безуспешная попытка найти в текущем состоянии памяти некоторую информационную конструкцию, являющуюся достаточной для того, чтобы этот элементарный процесс закончился результативно. Нерезультативные элементарные процессы никак не влияют на процесс переработки информационных конструкций, хранимых в памяти абстрактной машины. Тем не менее нерезультативные элементарные процессы абстрактной машины необходимо исследовать, так как одним из направлений повышения эффективности абстрактной машины является снижение частоты появления таких процессов, чего можно добиться 1) путем подбора системы приоритетов на множестве операций, 2) путем совершенствования структуры перерабатываемых информационных конструкций, 3) путем совершенствования микропрограмм операций.

Микропрограмма каждой операции абстрактной машины представляет собой описание семантики соответствующей операции этой машины. Систему операций абстрактной машины можно считать первым уровнем декомпозиции абстрактной машины, а систему соответствующих микропрограмм – вторым уровнем ее декомпозиции. Микропрограммы не должны непосредственно храниться в памяти этой абстрактной машины. Хотя очевидно, что память абстрактной машины, интерпретирующей некоторую другую абстрактную машину, должна хранить микропрограммы всех операций интерпретируемой абстрактной машины.

Операции абстрактной машины взаимодействуют друг с другом только через память абстрактной машины и в этом смысле являются автономными (самостоятельными). Операции абстрактной машины реализуются асинхронно, и если в текущем состоянии памяти абстрактной машины для нескольких операций одновременно существуют условия их реализации, то эти операции могут быть реализованы параллельно. Каждая операция абстрактной машины реагирует на соответствующую этой операции ситуацию, возникающую в текущем состоянии памяти абстрактной машины. Таким образом, синхронизация выполнения операций абстрактной машины, и в частности организация их последовательного выполнения, осуществляется через память этой машины с помощью специальных управляющих структур данных, описывающих текущее состояние процесса обработки информации. Следовательно, управление последовательностью выполнения элементарных процессов абстрактной машины обработки информации осуществляется децентрализованным образом и непосредственно входит в компетенцию каждой операции.

Из сказанного следует, что микропрограммы всех операций абстрактной машины носят активный, демонический характер, т.е. инициируются самопроизвольно в произвольные моменты времени, независимо друг от друга и без каких-либо внешних причин. Другими словами, для того чтобы инициировать микропрограмму любой операции абстрактной машины, никакого явного обращения к ней из другой микропрограммы, никакого явного ее вызова не требуется.

Язык, операционная семантика которого определяется абстрактной машиной обработки информации, будем называть внутренним языком этой абстрактной машины. Наряду с внутренним языком абстрактная машина обработки информации может иметь несколько внешних языков, с помощью которых осуществляются ввод информации в память и вывод информации из памяти абстрактной машины. Такой обмен информацией с пользователями машины обработки информации поддерживается соответствующими операциями ввода и вывода информационных конструкций. Эти операции осуществляют трансляцию вводимых информационных конструкций с некоторого внешнего языка абстрактной машины обработки информации на ее внутренний язык, а также трансляцию выводимых информационных конструкций с внутреннего языка на тот или иной внешний язык. Очевидно, что такие операции-трансляторы могут быть достаточно сложными, если внешние языки абстрактной машины обработки информации сильно отличаются от ее внутреннего языка.

Противопоставление внешнего и внутреннего языка является для абстрактных машин относительным, ибо для любого способа представления информационных конструкций, который является внешним

для одной абстрактной машины, можно построить другую абстрактную машину, в которой указанный способ представления будет внутренним.

Кроме операций ввода и вывода информационных конструкций, предназначенных для взаимодействия с пользователем, абстрактная машина обработки информации может взаимодействовать с внешней средой также с помощью различных рецепторных и эффекторных операций. Рецепторные операции осуществляют формирование информационных конструкций в памяти абстрактной машины путем описания (на внутреннем языке) текущего состояния соответствующих рецепторных подсистем. Эффекторные операции осуществляют воздействие на внешнюю среду путем изменения состояния соответствующих эффекторных подсистем.

Завершая рассмотрение понятия абстрактной машины обработки информации, определяющей операционную семантику некоторого языка путем его непосредственной интерпретации, подчеркнем, что это понятие имеет также большую практическую ценность, поскольку строгое описание абстрактной машины обработки информации (перечисление всех ее операций и соответствующих им микропрограмм) содержит всю информацию, необходимую для реализации указанного языка любым выбранным способом (программным, аппаратным, аппаратно-программным).

Понятие абстрактной машины обработки информации является также важной методологической опорой при проектировании вычислительных машин с нетрадиционной (не фон-Неймановской) архитектурой, особенно параллельных машин. "Если мы занимаемся исследованием предельных теоретических машин, а не практическим инженерным анализом существующих устройств, то необходимо абстрагироваться от многих реальных деталей и особенностей систем. Большей частью это абстрагирование заходит так далеко, что остается лишь скелетное представление о структуре последовательности событий внутри машины, т.е. некоторого рода "символическая" или "информационная" структура машины. При таком уровне абстракции мы игнорируем геометрию расположения частей. Мы игнорируем вопросы, касающиеся энергии. Мы даже забываем время на последовательность отдельных, не связанных между собой моментов и вообще игнорируем пространство как таковое! Может ли вообще столь абстрактная теория быть теорией чего-либо? Как ни странно, может. Выделяя только те вопросы, которые касаются логических выводов из определенного вида причинно-следственных отношений, мы можем сконцентрировать наше внимание на немногих действительно фундаментальных проблемах. Разобравшись в этих вопросах, мы можем вернуться в мир практики, где столь ясное понимание сути дела было бы невозможным из-за множества несущественных деталей" [342] (*Минский М. 1971кн-ВычисИА*).

Использование понятия абстрактной машины для уточнения различных моделей вычислений осуществляется в работе [24] (*Ахо А. 1979кн-ПострИАВА*).

Понятие абстрактной машины как средства формального рассмотрения операционной семантики языков программирования, причем не обязательно процедурного программирования, используется в работах [5; 520; 408] (*Агафонов В.Н. 1990кн-СпецИП ; Хендерсон П. 1983кн-ФункцП ; Прайт Т. 1979кн-ЯзыкиП*).

О трактовке машин Тьюринга как о частном виде параллельных асинхронных абстрактных машин см. в работе [5] (*Агафонов В.Н. 1990кн-СпецИП*).

1.3.2. Классификация абстрактных машин обработки информации

Ключевые понятия: абстрактные машины с символьной памятью; абстрактные машины с графовой памятью; абстрактные машины со структурно фиксированной памятью; абстрактные машины со структурно перестраиваемой памятью; абстрактные машины переработки знаний; абстрактные машины реализации хранимых программ; последовательные абстрактные машины обработки информации; параллельные абстрактные машины обработки информации; синхронные абстрактные машины обработки информации; асинхронные абстрактные машины обработки информации; абстрактные машины обработки информации, в памяти которых описание элементарных информационных процессов не осуществляется; абстрактные машины обработки информации, управляемые потоком команд; абстрактные машины обработки информации, управляемые потоком данных.

Как было отмечено в пункте **1.1.2**, тип абстрактной машины и тип внутреннего языка этой машины полностью определяют тип соответствующей формальной модели. Типология языков рассмотрена в пункте **1.2.5**. Классификацию абстрактных машин будем проводить по следующим признакам.

1. Структурный тип информационных конструкций, хранимых в памяти абстрактной машины. По этому признаку можно выделить:

- **абстрактные машины с символьной** (линейной) **памятью**, т.е. абстрактные машины, в памяти которых могут храниться только символьные конструкции;
- **абстрактные машины с графовой** (нелинейной) **памятью**, т.е. абстрактные машины, в памяти которых непосредственным образом могут храниться графовые конструкции.

2. Характер изменения состояния памяти, возможность изменения связей между элементами памяти. По этому признаку можно выделить:

- **абстрактные машины со структурно фиксированной памятью**, обработка информации в которых сводится только к изменению состояния элементов памяти;
- **абстрактные машины со структурно перестраиваемой** (модифицируемой) **памятью**, обработка информации в которых сводится к изменению состояния элементов памяти и к изменению связей между ними.

Абстрактные машины со структурно фиксированной и структурно перестраиваемой памятью могут перерабатывать как символьные, так и графовые информационные конструкции. Абстрактные машины с графовой структурно перестраиваемой памятью будем называть графодинамическими. Такие машины являются основным объектом рассмотрения в данной работе.

В структурно фиксированной памяти жестко задана структура элементов памяти, система связей между ними. Поэтому в структурно фиксированной памяти возникает проблема размещения хранимой информационной конструкции в памяти, т.е. проблема наложения информационной конструкции на структуру памяти. Кроме того, при переработке информации в структурно фиксированной памяти часто возникает проблема переразмещения (перераспределения) информационных конструкций, хранимых в памяти.

Структурно фиксированная память является неудобной даже для переработки символьных конструкций. Так, например, в символьной структурно фиксированной памяти весьма громоздко реализуются такие элементарные и часто используемые преобразования, как вставка строки символов или удаление подстроки. Реализация этих преобразований в символьной структурно фиксированной памяти требует достаточно утомительных действий по перезаписи (сдвигу) хранимых в памяти строк символов. Очевидно, что действий по перезаписи хранимой в памяти информации можно вообще избежать, если разрешить менять не только состояние элементов памяти, но и связи между ними. Благодаря этому становится возможной вставка (запись в память) новых информационных конструкций, а также удаление (стирание в памяти) фрагментов перерабатываемой информационной конструкции, оказавшихся в какой-то момент лишними, без какой-либо перезаписи остальной части хранимой в памяти информационной конструкции.

3. Уровень организации доступа к нужным фрагментам перерабатываемой информационной конструкции, который определяется количеством усилий, затрачиваемых программистом (при составлении программы), и количеством усилий, затрачиваемых операциями абстрактной машины на выделение (локализацию, поиск) требуемого фрагмента перерабатываемой информационной конструкции. Вид доступа к требуемым фрагментам определяется 1) ограничениями на размеры и структуру этих фрагментов и 2) значимыми признаками искомого фрагмента, на основании которых осуществляется поиск, т.е. тем, что должна знать абстрактная машина о требуемом фрагменте хранимой информационной конструкции для того, чтобы выделить этот фрагмент в рамках всей хранимой информационной конструкции. В настоящее время наиболее известными методами доступа являются **последовательный доступ, адресный** (прямой) **доступ, доступ по ключу** и **обобщенный ассоциативный доступ**.

Первые три из перечисленных методов доступа предполагают расчленение хранимой информационной конструкции на некоторые области (зоны, блоки) фиксированной или переменной длины, к которым и осуществляется непосредственный доступ. При последовательном методе доступа на множестве указанных областей задается бинарное отношение непосредственного соседства, являющееся отношением строгого порядка. Описание области, к которой осуществляется непосредственный доступ, заключается в указании того, где она находится по отношению к области, обозреваемой в текущий момент (справа от нее или слева). Таким образом, последовательный метод доступа в каждый

момент времени обеспечивает возможность доступа только к двум смежным областям хранимой информационной конструкции. Адресный метод доступа устраняет этот недостаток путем условного задания взаимно однозначного отображения множества областей хранимой информационной конструкции во множество имен, называемых адресами этих областей. Описание области, к которой осуществляется непосредственный доступ, здесь заключается в указании ее адреса. Адресный метод доступа, как и последовательный, требует того, чтобы программист точно знал, где в памяти находится каждая область хранимой информационной конструкции. Это вызывает проблему распределения памяти, которая заключается в том, что сам программист должен "раскладывать по полочкам" перерабатываемую им информацию. Это, очевидно, требует значительных накладных расходов.

Поскольку адрес требуемой области перерабатываемой информационной конструкции не всегда легко определить, требуемую область удобнее задавать не ее адресом, а некоторой известной ее подструктурой (например, известными значениями некоторых разрядов). В этом случае программисту не требуется знать, в каком месте памяти находится необходимая ему область [508] (*Фостер К.1981кн-АссоцПП*). Такой метод доступа называется доступом по ключу (по ключевому набору заранее известных признаков).

Общим недостатком первых трех методов доступа является наличие определенных ограничений на структуру и "размеры" областей (единиц доступа) хранимой информационной конструкции. Снятие каких бы то ни было ограничений на размеры и структуру областей информационных конструкций, требующее отсутствия разбиения этих объектов на области, предпринято при организации обобщенного ассоциативного метода доступа (доступа по значению) [187] (*Дейт К.1980кн-ВведеВСБД*). Здесь описание требуемой области есть указание ее структуры (с точностью до изоморфизма) или некоторой ее подструктуры, дополненной указанием некоторых свойств. С логической точки зрения такое описание требуемого фрагмента информационной конструкции представляет собой не что иное, как его определение. Следовательно, такой язык описания (задания) фрагментов можно с полным основанием считать декларативным языком запросов. Подчеркнем, что обобщенный ассоциативный метод доступа оказывается единственно возможным для языка программирования, специально ориентированного на переработку баз данных, поскольку программист в этом случае не может заранее знать, где и как хранятся требуемые ему данные (это оказывается возможным, только когда сам программист организует для себя все необходимые для него данные). Таким образом, обобщенный ассоциативный метод доступа обеспечивает произвольную перемещаемость данных в памяти и независимость данных от программ, их использующих. Следует также заметить, что переработку базы данных с использованием обобщенного ассоциативного метода доступа логически очень просто организовать как вычисление, управляемое потоком данных.

Нетрудно заметить, что все виды доступа можно формально проинтерпретировать как частный случай обобщенного ассоциативного доступа. Очевидно также, что снятие ограничений на вид и размер требуемых фрагментов информационной конструкции позволяет существенно увеличить множество реализуемых абстрактной машиной операций.

Нетрудно заметить, что в полном, наиболее развитом варианте обобщенный ассоциативный метод доступа реализуем только на основе использования графовых языков и графодинамической памяти.

Подчеркнем, что все исследуемые в данной работе абстрактные машины используют обобщенный ассоциативный метод доступа к фрагментам хранимой информационной конструкции.

4. Денотационная семантика перерабатываемых информационных конструкций и смысл операций абстрактной машины. По этому признаку можно выделить **абстрактные машины переработки знаний и абстрактные машины реализации хранимых программ.**

5. Возможность или невозможность одновременного выполнения нескольких результативных элементарных процессов. По этому признаку можно выделить:

- **последовательные абстрактные машины обработки информации** (абстрактные машины обработки информации, осуществляющие последовательное выполнение элементарных процессов);
- **параллельные абстрактные машины обработки информации.**

Подчеркнем, что абстрактные машины обработки информации в общем случае считаются параллельными, что следует из определения абстрактных машин. Следовательно, последовательные абстрактные машины считаются абстрактными машинами частного вида. Согласно определению абстрактной машины, ее параллельность или последовательность определяется не самими операциями, а осо-

бенностями хранимой в памяти информационной конструкции, ибо каждая операция абстрактной машины считается абсолютно автономной и может реализовываться, как только в памяти возникнет ситуация, удовлетворяющая условию ее применения. При этом реализуется операция абсолютно независимо от того, реализуется в это время какая-то другая операция или нет. Следовательно, для последовательной абстрактной машины характерно то, что в ее памяти никогда не возникает ситуация, которая бы одновременно удовлетворяла условиям применения сразу нескольких операций.

Необходимо отметить, что в параллельных машинах обработки информации понятие текущего состояния ее памяти становится неопределенным, так как может не существовать ни одного момента, в который бы не выполнялось ни одного элементарного процесса. То есть память параллельной машины обработки информации всегда может находиться в переходном состоянии, когда до завершения каждого элементарного процесса начинается и не заканчивается какой-либо другой элементарный процесс или несколько элементарных процессов.

6. Характер взаимодействия элементарных процессов (насколько произвольным может быть выбор момента начала выполнения элементарных процессов). По этому признаку можно выделить:

- **синхронные абстрактные машины обработки информации;**
- **асинхронные абстрактные машины обработки информации.**

Из определения абстрактных машин обработки информации следует, что в общем случае такие машины являются асинхронными. Синхронность в абстрактной машине обработки информации достигается с помощью специальных управляющих (синхронизирующих) информационных конструкций, которые явно фиксируют факты завершения элементарных процессов того или иного вида и соответствующим образом анализируются при проверке условий выполнения элементарных процессов.

Суть асинхронного взаимодействия элементарных процессов абстрактной машины обработки информации заключается в том, что в каждый момент времени могут существовать условия инициирования нескольких элементарных процессов. Но последовательность выполнения этих процессов произвольна. Таким образом, в асинхронной машине обработки информации наличие условия выполнения того или иного элементарного процесса означает не предписание (команду), а только разрешение на выполнение этого процесса.

7. Характер описания в памяти и инициирования элементарных информационных процессов. По этому признаку можно выделить:

- **абстрактные машины обработки информации, в памяти которых описание элементарных информационных процессов не осуществляется** ни в каком виде (этот класс абстрактных машин обработки информации полностью совпадает с абстрактными машинами переработки данных);
- **абстрактные машины обработки информации, управляемые потоком команд;**
- **абстрактные машины обработки информации, управляемые потоком данных;**
- **абстрактные машины обработки информации, управляемые потоком целей.**

Команда, представляющая собой инициированный оператор хранимой процедурной программы, содержит полную информацию, необходимую для выполнения соответствующего элементарного информационного процесса.

В абстрактной машине обработки информации, управляемой потоком данных, хранимая (уже не процедурная, а точнее, функциональная) программа также представляет собой совокупность операторов. Но условием реализации каждого из этих операторов является не его явное инициирование, а вычисленность всех его операндов.

В абстрактных машинах обработки информации, управляемых потоком целей, операции и элементарные процессы имеют дело уже не с операторами, каждый из которых задает преобразование заранее известного типа, а с описаниями целей, способ достижения которых определяется не только самими целями, но и их контекстами.

8. Степень открытости (гибкости, модифицируемости) абстрактных машин.

Для открытых абстрактных машин характерно то, что добавление в этих машинах новых операций или модификация имеющихся не требует модификации остальных операций либо требует их локальной модификации.

Высокая степень открытости абстрактных машин является важнейшим требованием, предъявляемым к современным системам обработки информации, особенно к современным системам переработки знаний.

Принятая трактовка абстрактных машин создает все необходимые предпосылки для создания абстрактных машин с высокой степенью открытости, так как эта трактовка основана 1) на отсутствии какого-либо непосредственного взаимодействия между операциями (отсутствие вызова одной операции из другой), 2) на использовании самопроизвольного инициирования операций, 3) на использовании только одного способа фиксации текущего состояния машины – через текущее состояние памяти, 4) на организации взаимодействия и синхронизации элементарных процессов только через память абстрактной машины.

Тем не менее следует подчеркнуть, что наиболее высокого уровня открытости можно добиться только от абстрактных машин, имеющих графовую структурно перестраиваемую память. Высокий уровень открытости таких абстрактных машин обеспечивается высоким уровнем их ассоциативности. Именно поэтому данная работа, направленная на создание максимально открытых средств переработки знаний, базируется на исследовании графодинамических абстрактных машин.

1.3.3. Графодинамические параллельные асинхронные абстрактные машины как наиболее перспективный класс абстрактных машин для проектирования сложных интеллектуальных систем

Преимущество использования графодинамических параллельных асинхронных абстрактных машин в качестве инструмента для создания интеллектуальных систем нового поколения обусловлено следующими обстоятельствами:

- 1) в графодинамических машинах принципиально проще реализуется ассоциативный метод доступа к перерабатываемой информации;
- 2) в графодинамических машинах существенно проще поддерживается открытый характер как самих машин, так и реализуемых на них формальных моделей. В частности, на графодинамических машинах существенно проще реализуются семиотические модели;
- 3) графодинамические параллельные асинхронные машины являются удобной основой для интеграции самых различных моделей обработки информации, ибо на базе графодинамических параллельных асинхронных абстрактных машин легко реализуются не только графодинамические, но и символные формальные модели, не только параллельные, но и последовательные модели, не только асинхронные, но и синхронные модели, не только различные модели логического вывода, но и всевозможные модели реализации хранимых процедурных программ.

Остальные достоинства графодинамических параллельных асинхронных машин обусловлены достоинствами графовых информационных конструкций и графовых языков (см. подраздел 1.2). Подчеркнем при этом, что нас будут интересовать в первую очередь такие графодинамические параллельные асинхронные машины, внутренними языками которых являются семантические языки.

Резюме к подразделу 1.3

Соображения, приведенные в подразделе 1.3, позволяют сделать вывод о том, что графодинамические параллельные асинхронные абстрактные машины являются наиболее перспективной основой для реализации интеллектуальных систем нового поколения. При этом особое внимание следует уделить трем видам графодинамических параллельных асинхронных абстрактных машин, определяющим три уровня параллельных интеллектуальных систем:

- 1) графодинамическим параллельным асинхронным абстрактным машинам логического вывода. Перерабатываемая информация (перерабатываемые знания) в таких машинах представляется в виде семантических конструкций (семантических сетей) на некотором графовом семантическом языке представления знаний. А операциями в этих машинах являются операции логического вывода, поддерживающие самые различные стратегии решения задач в базах знаний;

- 2) графодинамическим параллельным асинхронным абстрактным машинам реализации хранимых процедурных программ, специально ориентированным на интерпретацию графодинамических параллельных асинхронных абстрактных машин логического вывода;
- 3) графодинамическим распределенным (!) асинхронным абстрактным машинам реализации хранимых процедурных программ, определяющим крупнозернистую архитектуру графодинамического параллельного асинхронного компьютера, ориентированного на поддержку всевозможных графодинамических параллельных асинхронных моделей обработки информации. Операции абстрактных машин данного уровня, в отличие от абстрактных машин второго уровня, имеют ограниченную область действия, т.е. имеют доступ не ко всей памяти, а только к какой-то ее области. Следовательно, вся память абстрактной машины разбивается на области, каждой из которых ставится в соответствие набор работающих над ней операций. При этом наборы операций, работающих над разными областями памяти, абсолютно аналогичны и отличаются только областью действия операторов. Таким образом, графодинамическая распределенная асинхронная абстрактная машина реализации хранимых процедурных программ представляет собой коллектив графодинамических абстрактных машин (которые будем называть модулями распределенной машины), взаимодействующих между собой путем обмена сообщениями. Особенности распределенных абстрактных информационных машин являются а) наличие в каждом модуле входных и выходных буферов для приема и передачи сообщений, б) наличие специальных операций пересылки сообщений из выходного буфера модуля-передатчика во входной буфер модуля-приемника (модуля-адресата), в) наличие специальных операций обработки принятых сообщений.

В настоящее время исследования по графовым языкам, графодинамическим машинам и графодинамическим формальным моделям в основном носят частный теоретический характер и пока не привели к широкому появлению практически пригодных и конкурентоспособных графовых языков программирования, графовых языков представления знаний, графодинамических логик и графодинамических компьютеров. К числу работ, связанных с графодинамическими моделями обработки информации, можно отнести работы по графовым грамматикам [376; 11; 379] (*Петров С.В..1977ст-ГрафоГиЗГ ; Айзерман М.А..1977ст-ДинамПКАС ; Петров С.В..1978ст-ГрафоГиА*), работы по исследованию логического вывода в семантических сетях [80] (*Вагин В.Н.1989кн-ДедукИО*), работы по исследованию алгоритмов А.Н.Колмогорова [259; 174; 175] (*Колмогоров А.Н..1958ст-кОпредА ; Григорьев Д.Ю.1976ст-АлгорКСМ ; Григорьев Д.Ю.1977ст-ТеорЕвдМТ*), работы А.Шенхаге по исследованию абстрактных машин с модифицируемой памятью [674] (*Schonhage A.1980art-StoraMM*), работы В.Б.Борщева по вегетативной машине [65] (*Борщев В.Б.1983ст-СхемыНКС*), работы А.М.Степанова по абстрактной сетевой машине [452; 453] (*Степанов А.М.1981пр-ФреймИПСВ ; Степанов А.М.1981пр-ЭкснеСП*).

Выводы к разделу 1

1. Поскольку решение задач в интеллектуальных системах требует использования самых различных методов (процедурных программ, непроцедурных программ, баз знаний), важной проблемой является не только разработка каждого из этих методов, но и разработка способов их эффективной интеграции, т.е. эффективного совместного использования в рамках одной интеллектуальной системы. Поэтому актуальным является создание принципов, в рамках которых такая интеграция была бы достаточно легко реализуемой.

2. На современном этапе развития языков программирования и языков представления знаний особую актуальность приобретает совершенствование таких их свойств, как открытость (модифицируемость), ассоциативность доступа к перерабатываемой информации, поддержка параллельной асинхронной реализации операций.

3. Совершенствование методов описания операционной семантики самых различных языков и, соответственно, методов интерпретации этих языков настоятельно требует введения общего понятия абстрактной машины, в рамках которого можно было бы достаточно легко специфицировать всевозможные методы организации обработки информации.

4. Вводится понятие реляционной структуры, являющееся способом формального уточнения сложноструктурированных предметных областей, сложноструктурированных информационных объектов и представляющее собой обобщение таких понятий, как алгебраическая модель, гиперсеть, клубная

система. По семантическому принципу реляционные структуры разбиваются на два типа: 1) реляционные структуры, определяющие структуру различных предметных областей, и 2) реляционные структуры, определяющие структуру (синтаксис) различных информационных объектов. Реляционные структуры 2-го типа называются информационными конструкциями.

5. Вводится понятие графовой информационной конструкции. Рассматриваются графовые информационные конструкции частного вида (бинарные информационные конструкции и однородные информационные конструкции), являющиеся различными способами кодирования произвольных реляционных структур.

6. Уточняется понятие семантической сети (семантической графовой информационной конструкции).

7. На современном этапе развития языков программирования и языков представления знаний перспективным является класс графовых языков, обеспечивающих представление информации в виде тем или иным способом устроенных семантических сетей. Графовые языки поддерживают открытость, ассоциативность, параллельность, асинхронизм. Графодинамическая парадигма обработки информации является наиболее перспективной основой для создания интеллектуальных систем нового поколения, в которых должны поддерживаться сложноструктурированные базы знаний большого объема, сложные стратегии и механизмы решения трудноформализуемых задач, гибкость (модифицируемость) используемых стратегий и механизмов решения задач, интегрируемость различных моделей переработки знаний, асинхронизм и высокий уровень параллелизма.

8. Рассмотрены основные понятия, связанные с графодинамической парадигмой обработки информации, – понятие графодинамической формальной модели, понятие графовой конструкции, понятие семантической графовой конструкции, понятие графового языка, понятие графового семантического языка, понятие графодинамической абстрактной машины, понятие графодинамической параллельной асинхронной абстрактной машины.

9. Графодинамические модели обработки информации, графовые языки программирования, графовые языки представления знаний, графодинамические абстрактные машины – все это является естественным результатом эволюции традиционных моделей обработки информации, направленной на поддержку сложноструктурированности, гибкости, модифицируемости, асинхронности, ассоциативности, параллельности.

10. Основными направлениями работ по созданию комплекса средств, ориентированных на поддержку интеллектуальных систем нового поколения, являются: создание базового графового языка, обеспечивающего легкую интегрируемость различных графодинамических формальных моделей, построенных с использованием этого языка; создание графового логического языка расширяемого типа и соответствующей ему графодинамической абстрактной машины логического вывода; создание графовых языков программирования и соответствующих им графодинамических абстрактных машин, обеспечивающих легкую интегрируемость с графодинамическими моделями логического вывода; создание графового языка программирования и соответствующей ему графодинамической абстрактной машины для эффективной интерпретации всевозможных графодинамических параллельных моделей; создание графодинамического параллельного асинхронного ассоциативного компьютера, ориентированного на эффективную интерпретацию всевозможных графодинамических параллельных моделей.

11. В основе формального рассмотрения интеллектуальных систем на самом верхнем уровне лежит специальный класс моделей обработки информации, называемых моделями логического вывода. Основными особенностями таких моделей являются:

- 1) рассмотрение обработки информации на семантическом уровне;
- 2) сложность операций, семантически трактуемых как операции логического вывода, как содержательно осмысливаемые механизмы решения задач, поддерживающие ту или иную стратегию решения;
- 3) объективно присущие этим моделям параллельность и асинхронность реализации операций.

Добавим к этому то, что модели логического вывода, соответствующие практически полезным интеллектуальным системам, оперируют перерабатываемыми информационными конструкциями (базами знаний) большого размера и сложной структуры. Последнее обусловлено не только тем, что практически интересные модели логического вывода часто бывают модифицируемыми, но и тем, что даже в классических (немодифицируемых) моделях логического вывода практически интересные модели обычно имеют дело со знаниями, которые описывают сложноструктурированные предметные области.

Известные модели представления и переработки знаний а) имеют свои достоинства и недостатки, б) не противоречат друг другу, т.е. не являются альтернативными, и в) дополняют друг друга, поскольку каждая из этих моделей акцентирует свое внимание не на всех, а только на некоторых аспектах представления знаний.

Следовательно, одним из направлений повышения эффективности известных моделей представления знаний является их интеграция, т.е. создание комплексной модели представления знаний, в рамках которой гармонично бы сочетались и логическая модель представления знаний, и сетевая (графовая) модель представления знаний, и фреймовая, и продукционная модель, а также различные модели представления процедурных знаний, модели представления и реализации программ. О целесообразности интеграции различных моделей представления знаний говорится, в частности, в работе [401] (*Поспелов Д.А. ред. 1990 спр-ИскусИ-К2*).

Построение комплексной (интегрированной) модели представления знаний целесообразно начинать с создания базовой (унифицированной, стандартизированной) сетевой модели, которая бы являлась ядром, основой всего комплекса моделей представления знаний. Целесообразность такого подхода обусловлена тем, что основным препятствием для интеграции различных моделей представления знаний является отсутствие базового языка.

В случае создания удачного базового графового языка, т.е. базовой сетевой (графовой) модели представления знаний, интеграция различных моделей представления знаний может быть осуществлена путем погружения в эту базовую модель.

Важнейшими дополнительными требованиями, предъявляемыми к комплексной модели представления знаний, являются:

- гибкость, т.е. легкая возможность модификации модели;
- параллельность, асинхронизм и адекватность аппаратной поддержке в компьютерах нового поколения, ориентированных на параллельную и асинхронную переработку знаний.

Из сказанного следует, во-первых, то, что модели представления и переработки знаний удобнее всего строить как графодинамические, и, во-вторых, то, что реализовывать модели переработки знаний естественнее не на традиционных (фон-Неймановских) вычислительных машинах, а на графодинамических параллельных асинхронных машинах, которые с полным основанием можно считать вычислительными машинами нового поколения, специально ориентированными на использование в интеллектуальных системах.

12. Рассмотрены семиотические модели обработки информации, являющиеся для интеллектуальных систем важнейшим классом моделей, на основе которых, в частности, осуществляется реализация всевозможных псевдофизических логик. Поскольку основным путем реализации семиотических моделей обработки информации является их сведение к формальным моделям, необходимы языки с развитыми метаязыковыми средствами, с помощью которых можно было бы описывать правила модификации языка и правила модификации системы операций. Быть основой для таких метаязыков – это одно из основных требований, предъявляемых к базовому графовому языку SC (Semantic Code), рассматриваемому в разделе 4. Формальные модели обработки информации, осуществляющие интерпретацию семиотических моделей, являются типичным примером моделей переработки сложноструктурированных знаний, которые носят иерархический характер со сложным переплетением информации и метаинформации.

13. Наиболее перспективным классом формальных моделей обработки информации являются графодинамические параллельные асинхронные модели, так как они:

- наиболее близки к семантическому уровню рассмотрения и обработки информации;
- достаточно просто интерпретируют друг друга;
- легко интегрируются.

Соответственно этому перспективным классом языков и абстрактных машин являются графовые языки и графодинамические параллельные асинхронные абстрактные машины.

К числу графодинамических формальных моделей можно отнести также и нейросетевые модели, обработка информации в которых сводится к изменению состояния формальных нейронов и (при обучении нейронной сети) к изменению веса связей между ними.

Понятие графодинамической модели обработки информации, понятие графового языка и понятие графодинамической параллельной асинхронной абстрактной машины являются центральными понятиями всей данной работы.