

РОССИЙСКАЯ АКАДЕМИЯ НАУК
ИНСТИТУТ ПРОБЛЕМ ИНФОРМАТИКИ

А.В. ИЛЬИН, В.Д. ИЛЬИН

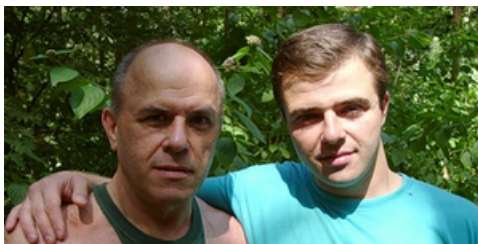
ОСНОВЫ ТЕОРИИ S-МОДЕЛИРОВАНИЯ



Москва
ИПИ РАН
2009

*Ильин Владимир
Дмитриевич*

*Ильин Александр
Владимирович*



Доктор технических наук,
профессор.
Зав. лабораторией

Кандидат технических наук.
Старший научный сотрудник
лаборатории

«Методологических основ информатизации»
в Институте проблем информатики РАН.

Автор более 100 трудов по
автоматизации
программирования,
информатизации,
S-моделированию, системам
знаний. Научный консультант и
автор статей БРЭ. Научный
редактор и автор статей
Энциклопедии информатики
ИНФОПЕДИЯ.

Автор более 30 трудов по
конструированию программ и
сервисов, S-моделированию,
системам знаний. Автор статей
БРЭ и ИНФОПЕДИИ.

УДК 004

Ильин А.В., Ильин В.Д.

Основы теории s-моделирования

– М.: ИПИ РАН, 2009. – 143с. – ISBN 978-5-902030-78-2

Теория s-моделирования включает язык специфицированного описания расширяемой системы понятий символьного моделирования произвольных объектов, описание ядра этой системы и классов базовых задач построения моделей и манипулирования ими. Теория рассматривается как методологическая платформа проектирования человеко-машинной среды символьного моделирования и информатизации.

Для информатиков, ИТ-разработчиков, преподавателей вузов и аспирантов.

*Издано по решению Учёного совета
Института проблем информатики Российской академии наук
(ИПИ РАН)*

© Ильин А.В., Ильин В.Д., 2009

ISBN 978-5-902030-78-2

Alexander V. Ilyin, Vladimir D. Ilyin

Basics of the Theory of S-modeling

– М.: IPI RAN, 2009. – 143p. – ISBN 978-5-902030-78-2

The theory of S-modeling includes language of the specified description for expandable concepts system of the symbol modeling of arbitrary objects, description of this system' core and basic task classes of constructing models and manipulating them. The theory is considered as a methodological platform for designing the human-machine environment of symbol modeling and informatization.

For computer scientists, IT-developers, university teachers and graduate students.

Issued by decision of the Academic Council of the Institute of Informatics Problems of the Russian Academy of Sciences (IPI RAN)

© Vladimir D. Ilyin, Alexander V. Ilyin, 2009

ISBN 978-5-902030-78-2

Оглавление

ПРЕДИСЛОВИЕ.....	7
О книге: размещение, структура.....	7
А. ТЕОРИЯ S-МОДЕЛИРОВАНИЯ: ЯЗЫК, ЯДРО, КЛАССЫ ЗАДАЧ.....	9
А.1. TSM-КОМПЛЕКС СРЕДСТВ ОПИСАНИЯ S-МОДЕЛЕЙ.....	10
А.1.1. Уровни вложенности фрагментов описания.....	10
А.1.2. Выделения.....	11
А.1.3. Сокращения.....	12
А.1.4. Умолчания.....	12
А.1.5. Формулы.....	12
А.1.6. Специализация и обобщение типов.....	15
А.1.7. Применимость.....	16
А.2. ОБЩИЙ МЕТОД И КЛАССЫ БАЗОВЫХ ЗАДАЧ.....	17
А.2.1. Инструмент познания.....	17
А.2.2. Эпоха S-моделирования.....	19
А.2.3. Общий метод	20
А.2.4. Классы базовых задач.....	22
А.3. СИМВОЛЫ, КОДЫ, СИГНАЛЫ.....	26
А.3.1. S-символы.....	28
А.3.1.1. Виды символов.....	31
А.3.1.2. Специализация и обобщение.....	37
А.3.1.3. Типы символов.....	39
А.3.2. S-коды.....	42
А.3.3. S-сигналы.....	44
А.3.4. Символы, коды, сигналы и эффективность деятельности.....	44
А.4. S-(СООБЩЕНИЕ, ДАННЫЕ, ИНФОРМАЦИЯ).....	48
А.4.1. S-сообщение.....	48
А.4.2. S-данные.....	52
А.4.3. S-информация.....	54

А.4.4. О понятийной неразберихе в «трёх соснах (сообщение, данные, информация)».....	55
А.4.5. Суть неудач машинного перевода.....	58
А.5. S-МОДЕЛИ СИСТЕМ ПОНЯТИЙ И СИСТЕМ ЗНАНИЙ.....	60
А.5.1. S-модель системы понятий.....	61
А.5.2. S-знание.....	64
А.5.3. S-модель системы знаний.....	66
А.6. ЗАДАЧНЫЕ КОНСТРУКТИВНЫЕ ОБЪЕКТЫ (S-ЗАДАЧИ) И СИСТЕМА ЗНАНИЙ ОБ S-ЗАДАЧАХ.....	69
А.6.1. Задачный конструктивный объект (s-задача).....	70
А.6.2. Система знаний об s-задачах.....	74
А.7. ИСЧИСЛЕНИЕ S-ЗАДАЧ И КОНСТРУИРОВАНИЕ РАЗРЕШАЮЩИХ СТРУКТУР.....	79
А.7.1. Исчисление задачных конструктивных объектов (исчисление s-задач).....	79
А.7.2. Разрешающие структуры на задачных графах.....	82
Б. МЕТОДОЛОГИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ПРОЕКТИРОВАНИЯ S-СРЕДЫ И ИНФОРМАТИЗАЦИИ.....	86
Б.1. S-МАШИНЫ.....	87
S-интерфейс.....	89
Б.2. S-СРЕДА.....	93
Б.3. S-СЕТЬ.....	99
Б.4. ИНТЕРНЕТ.....	106
Б.5. S-ДОМЕН.....	114
Б.6. S-СЕРВИС.....	117
Б.7. ГИПЕРТЕКСТ.....	118
Б.8. ВЕБ.....	121
Б.9. О ПРОБЛЕМЕ S-ЗАЩИТЫ.....	125
Б.10. О МЕТОДОЛОГИИ ИНФОРМАТИЗАЦИИ.....	129
Б.10.1. Информатизация.....	130
Б.10.2. Информационные ресурсы.....	133
Б.10.3. О методологии ситуационной информатизации.....	134
ЛИТЕРАТУРА.....	139

ПРЕДИСЛОВИЕ



S-моделирование (см. с.18)— вид моделирования, в котором применяются все реализуемые в человеко-машинной среде *виды символов* (аудио, визуальные и др. (см. с.29)) и которое осуществляется по правилам, соответствующим *классам базовых задач* (см. с.23), сформулированным в теории \geq s-моделирования.

Книга, представленная в форме гипермедийного документа, не только содержит изложение основ теории, но и служит демонстрационным примером применения этой теории.

В бумажной форме без потери части содержания и важных свойств гипермедийного документа (включая навигационные)¹ материал книги представить невозможно.

О книге: размещение, структура

Представлена первая редакция основ теории s-моделирования, созданной авторами в рамках исследований, посвящённых открытой распределённой гипермедийной системе знаний информатики (названной СИНФ [Ильин В.Д., Соколов И.А. 2007], [Ильин В.Д. 2009, 1]). Ядро СИНФ — *Энциклопедия информатики ИНФОПЕДИЯ* и журнал *ИНФОРМАТИКА: S-моделирование*. Оба эти издания в экспериментальном режиме действуют с апреля

1 В оформлении не применяются переносы, красноточные отступы и прочие способы экономии площади бумажных страниц.

2008. Для открытого обсуждения книга доступна на сайте журнала ИНФОРМАТИКА: S-моделирование.

После предисловия размещена основная часть книги (А), включающая разделы, содержащие описание основ теории. За ней — часть Б, объединяющая разделы, посвящённые методологическому обеспечению проектирования s-среды и информатизации (см. с.130) [Ильин В.Д. 2008, 3].

Оформление материала

Разметка текста (выделение фрагментов и др.) и формализованная запись соответствуют правилам унифицированного описания s-моделей (см. с.11).

Библиографические ссылки оформлены как перекрёстные ссылки и представлены в следующем формате:

[<Фамилия И.О. Автора>.< Год издания>,<номер публикации в году издания>].

Если публикация единственная, то поле номера публикации не заполняется. Если авторов несколько, то их ФИО разделяются запятыми.

Список литературы упорядочен по алфавиту и годам публикаций (по алфавиту — прямой, по годам — обратный порядок следования).

/ Навигация

Из оглавления можно переместиться к началу любого раздела. Для возврата в начало оглавления — ссылка **Огл** в нижнем колонтитуле (справа от стрелки). Чтобы переместиться к началу определения понятия, входящего в число ключевых, в круглых скобках указан номер страницы, являющийся ссылкой.

А. ТЕОРИЯ S-МОДЕЛИРОВАНИЯ: ЯЗЫК, ЯДРО, КЛАССЫ ЗАДАЧ



Теория *s-моделирования* включает язык специфицированного описания (см. с.11) расширяемой системы понятий, описание ядра этой системы и классов базовых задач (см. с.23) построения моделей произвольных объектов, а также манипулирования моделями.

Определены виды *символов* (*аудио, визуальных* и др.), применимых для построения *сообщений* (см. с.49) в человеко-машинной среде (*s-среде*; см. с.93). Для видов определены типы символов (рассчитанные на восприятие человеком). Определены виды, классы и типы *кодов* (см. с.43) и *сигналов* (см. с.45) рассчитанных на *s-машины* (см. с.87): компьютеры, коммуникаторы и др.). Для типов символов и кодов определены связывающие их отношения (*специализации, обобщения* и др.). Сформулированы правила построения в *s-среде* систем символов, символьных конструкций и соответствующих им систем кодов и кодовых конструкций. Определены унифицированные *s-модели систем понятий* (см. с.62) и *систем знаний* (см. с.67). Сформулированы *классы базовых задач s-моделирования*.

♦ В теории *s-моделирования* модели изучаются как сущности, имеющие взаимосвязанные представления в трёх мирах: *символьном, кодовом и сигнальном*. ♦

Теория S-моделирования рассматривается как методологическая платформа проектирования *s-(машин, среды)* и *информатизации* (см. с.130) различных видов деятельности.

A.1. TSM-КОМПЛЕКС СРЕДСТВ ОПИСАНИЯ S-МОДЕЛЕЙ



□ *TSM-комплекс средств описания s-моделей* (англ. TSM) – расширяемый набор средств унифицированного описания s-моделей систем понятий (см. с.62) и систем знаний (см. с.67). Включает средства записи формул, выделения частей гипермедийных описаний *s-моделей* и замены выбранными сокращениями часто повторяющихся фрагментов. □

Разработан на основе языка спецификации задачных конструктивных объектов [Ильин В.Д. 1989], развитого в

[Ильин А.В. 2007] и [Ильин А.В., Ильин В.Д. 2009].

A.1.1. Уровни вложенности фрагментов описания

□ *Вложенный фрагмент* – часть описания, включающая не менее одного полного абзаца (без заголовка или с заголовком). □

Выделяется косыми (slashes), размещаемыми в начале фрагмента: /*k*/ (*k* – номер уровня вложенности).

Для первого и второго уровней значения *k* не указываются (/ – первый уровень вложенности; // – второй); для третьего и последующих уровней можно указывать (начало фрагмента третьего уровня можно обозначить как /// или как /3/).

А.1.2. Выделения

Для выделения определений, замечаний, примеров, имен понятий и отдельных частей описания используются следующие средства:

\triangleleft <фрагмент описания> \triangleleft \approx часть описания с фиксированными в ее пределах обозначениями (здесь и далее символ \approx заменяет слово означает);

\square <фрагмент описания> \square \approx утверждение (определение, аксиома и др.);

\diamond <фрагмент описания> \diamond \approx замечание;

\odot <фрагмент описания> \odot \approx пример;

Δ <фрагмент описания> Δ \approx рекомендация или комментарий составителя описания;

$\{S\langle\text{фрагмент описания}\rangle\langle\text{список}\rangle S\}$ \approx здесь <фрагмент описания> \approx набранный курсивом текст (может быть пустым), который следует интерпретировать как расширенный префикс $s\text{-}\langle\text{текст}\rangle$ для выделенных курсивом элементов списка;

$\odot\{S\text{модель}\langle\text{список}\rangle S\}$ – здесь расширенным префиксом служит $s\text{-модель}$;

$\{S\langle\text{список}\rangle S\}$ – здесь префикс $s\text{-}$ \odot .

Курсивом выделяются:

- первые вхождения названий понятий [определяемых или определённых (последние могут быть гиперссылками)];
- фрагменты описания, к которым автор хочет привлечь внимание;

➤ формулы.

А.1.3. Сокращения

Для часто повторяющихся названий понятий:

СМ \approx символьное моделирование;

S-моделирование \approx СМ произвольных объектов в
человеко-машинной среде;

s-машина \approx машина, помогающая создавать и применять
s-модели;

s-среда \approx совокупность взаимодействующих людей и
управляемых ими s-машин, предназначенная для решения задач
S-моделирования.

А.1.4. Умолчания

Так как в s-среде имеем дело только с s-моделями, вместо
s-модель символа, s-модель кода, s-модель сообщения, s-модель
информации и т.д., пишем s-символ, s-код, s-сообщение,
s-информация и т.д. Слово s-модель не опускаем лишь там, где
может возникнуть контекстная неясность.

А.1.5. Формулы

Для теоретико-множественных и других формул применяется
одноэтажная форма записи.

/ Индексы, пометы

Не накладывается никаких ограничений на максимальное число
индексов для переменных и помечающих символов (помет).

Все индексы и пометы записываются в строчку внутри квадратных скобок, следующих сразу за индексируемой (или / и помеченной) переменной.

◊ Квадратные скобки в формульных фрагментах описаний зарезервированы только для индексов и помет.◊

Индексы, определяющие элемент массива, отделяются запятыми, индексированные индексы – косой чертой «/».

Верхний индекс от нижнего отделяется точкой с запятой «;». Если в описании индекса точка с запятой не встречается, то индекс считается нижним. Если сразу после точки с запятой стоит закрывающая квадратная скобка, то задан только верхний индекс.

$\star x[out; j = 1...n] \approx$ вектор x из n компонент, имеющий помету out ;

$a[inp; i = 1...m, j = 1...n] \approx$ матрица a размера $m*n$, имеющая помету inp ;

$c[';1] \approx$ c -один со штрихом (штрих «'» – верхняя помета, 1 – нижний индекс);

$d[j/i;] \approx d$ с верхним индексированным индексом j i -тое (чтобы показать отсутствие нижних индексов, поставлена точка с запятой, за которой сразу следует закрывающая квадратная скобка);

$d[j/i] \approx d$ с нижним индексированным индексом j i -тое (отсутствие точки с запятой указывает на отсутствие верхних индексов) \star .

/ Теоретико – множественные

$a: elem A \approx a$ является элементом множества A ;

$a, b: elem\ C \approx a, b$ — элементы множества C (число элементов, разделённых запятыми, может быть любым);

$A: set\ a \approx A$ — множество, содержащее элемент a ;

$A < B$ (когда оговорено, что A и B рассматриваются как множества)
 $\approx A$ — подмножество B ;

$B = D \approx$ множества D и B совпадают;

$C \leq B \approx C$ является подмножеством B или совпадает с ним;

$B > A \approx B$ содержит A ;

$A \geq E \approx A$ содержит E или совпадает с E ;

$A \wedge B \approx$ пересечение множеств A и B ;

$A \setminus B \approx$ разность множеств A и B ;

$A * B \approx$ декартово произведение множеств A и B ;

$R \leq A * B \approx$ бинарное отношение, заданное на множествах A и B .
Символ 0 обозначает пустое множество или нуль (в зависимости от контекста);

символ $\#$ обозначает «не равно».

☼ Если $x: elem\ X, y: elem\ Y$ и $x = y$, то $x: elem\ (X \wedge Y)$; если $X: set\ x, Y: set\ y$ и пара $(x, y): elem\ R$, где $R \leq A * B$, то $(X * Y) \wedge (A * B) \# 0$. ☼

Аргументы функции размещаются в круглых скобках, стоящих сразу за идентификатором, обозначающим функцию.

☼ $f(x) \approx f$ от x ;

$f[max;](x[i=1...n]) \approx f$ с верхней пометой max от $x[i=1...n]$. ☼

При записи операций символы «+», «-», «*», «/» обозначают соответственно сложение, вычитание, умножение, деление, а символ «**» – возведение в степень.

Для записи суммы вместо заглавной «сигмы» используется *sum*; при этом индекс суммирования, его начальное и конечное значения записываются в квадратных скобках справа от *sum*.

☼ $sum[i=1...n]$ ≈ сумма по *i* от 1 до *n*.☼

$x[j = 1...n] \geq 0$ ≈ вектор, все компоненты которого неотрицательны

А.1.6. Специализация и обобщение типов

Тип *X* ≈ множество *X*, элементы которого имеют фиксированные набор атрибутов и семейство допустимых операций.

Может иметь *подтипы*, называемые *специализациями типа X*, и *надтипы*, называемые *обобщениями типа X*.□

/ Специализация типа

□*Специализация типа X* – это порождение подтипа $X[::rule]$ (здесь sdвоенное двоеточие :: - символ специализации) с семейством связей, расширенным добавлением связи *rule*. Выделяет подмножество $X[::rule]$ множества *X*. *Специализацией* называем и результат $X[::rule]$ этого порождения ($X > X[::rule]$).□

// Специализация типа, заданная последовательностью добавляемых связей

$X[::(rule1)::rule2]$ – специализация типа $X[::rule1]$ по связи *rule2*. Число специализирующих связей в последовательности не ограничено. При этом имена связей, предшествующие

последнему, заключены в круглые скобки, а перед открывающей скобкой каждой пары скобок – сдвоенное двоеточие.

/ Обобщение типа

□ *Обобщение типа Z* – это порождение его надтипа $Z[\#rule]$ путём ослабления (здесь # – символ ослабления) связи *rule* из семейства связей, соответствующей типу Z . Исключение связи считаем её предельным ослаблением. □

A.1.7. Применимость

TSM рассчитан на формирование строчных описаний посредством QWERTY-клавиатуры.

А.2. ОБЩИЙ МЕТОД И КЛАССЫ БАЗОВЫХ ЗАДАЧ



□ **S-МОДЕЛИРОВАНИЕ** (англ. S-modeling) – символьное моделирование произвольных объектов в человеко-машинной среде (*s-среде*) представляет собой комплексную методологию решения задач *s-* (*представления, преобразования, распознавания, конструирования, интерпретации, обмена, сохранения, накопления, поиска и защиты*) *s-сообщений*. Предмет информатики [Ильин В.Д., Соколов И.А. 2008].

Обозначение — *s-modeling*. □

А.2.1. Инструмент познания

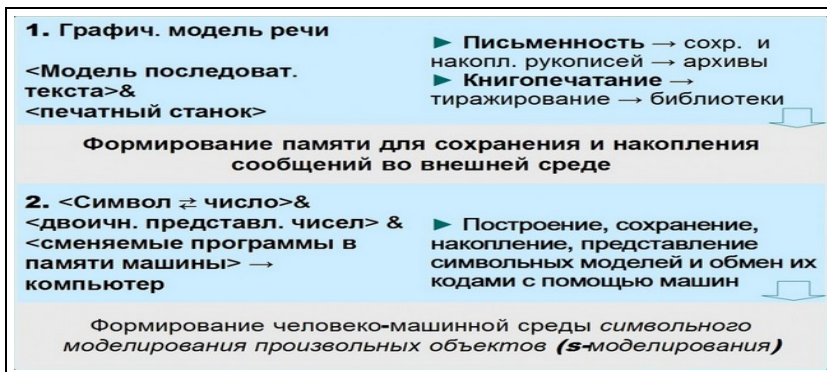
Изобретение символов и построенных из них символьных моделей сообщений, представление и накопление таких моделей во внешней среде стали важными (а, возможно, и ключевыми) средствами формирования и развития разумного человека. На длинном и трудном пути от наскальных рисунков, через рукописные тексты, книгопечатание, звукозапись, фотографию, кино и телевидение роль символьных моделей сообщений, сохраняемых во внешней среде, постоянно росла.

Их доминирующая роль в интеллектуальной деятельности определяется не только компактностью и выразительностью, но и тем, что не существует ограничений на типы носителей, применяемых для сохранения символьных моделей. Ими могут

быть память человека, бумажный лист, матрица цифровой фотокамеры, память цифрового диктофона или ещё что-то.

Затраты на построение, копирование, передачу, сохранение и накопление символьных моделей несопоставимо меньше, чем аналогичные затраты, связанные с несимвольными моделями (☼ макетами судов, зданий и др. ☼).

Символьное моделирование не только сопровождает абстрактное мышление, но и служит инструментом его совершенствования (позволяя на время отвлечься от деталей, чтобы чётче увидеть главное).



☼ Достаточно вспомнить, какое ускорение получило развитие математики после введения буквенных символов для записи формул (до того их записывали, используя разговорный язык). Примерами символьных моделей могут служить чертежи машин, записи музыкальных композиций, шахматных партий и т.д. ☼

Компактность и выразительность символьных моделей позволяют эффективно сочетать детализацию и обобщения в процессе рассуждений. Символьные модели – испытанный

инструментарий механизма ассоциаций, от продуктивности которого зависят судьбы изобретений и открытий.

А.2.2. Эпоха S-моделирования

Изобретение программируемой машины для поддержки процессов символического моделирования (*s-машины* (см. с.87), названной компьютером), изменило представления о возможностях машинной поддержки символического моделирования.

♦Вспомним пресловутое изобретение колеса, которое по сути было изобретением способа соединения колеса с неподвижной осью.♦

Истинная ценность компьютера не в том, что он быстро вычисляет и много запоминает. Это лишь технические характеристики, обязательные для исполнения роли средства построения *s-среды* (см. с.93).

♦Начало компьютерной эпохи стало стартом колоссальных по значению и динамике перемен в технологиях построения, преобразования, распознавания, интерпретации, сохранения, накопления, передачи, поиска и защиты символических моделей различных сообщений. Впервые люди стали применять машины во всех работах, связанных с символическими моделями сообщений [Ильин В.Д. 2009, 4].

Одновременно с компьютерной стартовала эпоха *символического моделирования произвольных объектов в человеко-машинной среде* (эпоха *S-моделирования* (см. с.18)).♦

С тех пор конкурентоспособная часть человечества пристально следит за новыми информационными технологиями и средствами их реализации, а понятие конкурентоспособности

теперь прочно связано с умением создавать и применять s-модели для повышения продуктивности своей деятельности.

А.2.3. Общий метод

Особое место в развитии символьного моделирования принадлежит идее его формализации, заключающейся в том, чтобы строить символьные модели по определенным правилам из заранее определенных элементов. Эта идея издавна реализуется в математических методах символьного моделирования. Однако метод формализации, применяемый в математике для получения формальных систем, нельзя перенести на s-моделирование, так как s-модели не являются формальными системами. Объясним подробнее это важное замечание.

В s-моделировании *задача* (см. с.71) имеет более широкий смысл, чем в математике. Задачи s-(представления, распознавания, преобразования, конструирования, интерпретации, обмена, сохранения, накопления, поиска, защиты) сообщений не являются математическими. Математический арсенал недостаточен для того, чтобы их можно было сформулировать и решить как математические задачи. Дело не только в том, что в математике главенствует формальное доказательство (существования, единственности решения), а в s-моделировании – конструктивное доказательство (существования s-модели; а о единственности вообще речь не идет). Важно другое: неформальность s-моделей – их полезное отличие, связанное с возможностью привлечения неформализованного знания человека-эксперта для управления процессами S-моделирования (☼методология интерактивного² преобразования ресурсов по изменяемым системам правил [Ильин А.В., Ильин В.Д. 2004] – одно из подтверждений☼).

2 см. [Ильин А.В. 2008, 1]

S-моделирование предполагает представление символов и построенных из них s-моделей в двух формах, одна из которых рассчитана на интерпретацию человеком, другая (в форме кодов (см. с.43)) – на интерпретацию программой s-машины.

Множество символов, применимых для построения s-моделей – это множество элементарных конструктивных объектов, каждый из которых наделен набором атрибутов и совокупностью допустимых операций. Построение конструкций из элементов этого множества определено системой правил конструирования s-моделей.

□*Общий метод s-моделирования* – конструктивное доказательство существования s-модели, представимой в двух формах, одна из которых рассчитана на интерпретацию человеком, а другая – s-машиной. □

Необходимое условие реализации s-моделирования предполагает существование удовлетворяющих требованиям s-(представления, распознавания, преобразования, конструирования, интерпретации, обмена, сохранения, накопления, поиска и защиты) s-сообщений:

1. языка описания s-моделей, рассчитанного на человека;
2. s-машинного языка (языка, рассчитанного на s-машину (см. с.87));
3. программ s-преобразования s-моделей на языке для человека в описания на s-машинном языке.

♦Формальное символьное моделирование в математике не стеснено требованиями 1 – 3. Конечно, языку математического моделирования можно поставить в соответствие язык описания s-моделей [✧*Пролог* (логика предикатов первого порядка), *Лисп* (λ -исчисление)✧]. Развитие языков s-моделирования,

рассчитанных на человека, направляется стремлением использовать композиции различных типов символов, библиотеки и средства конструирования программ [Ильин В.Д. 1989], [Ilyin V.D. 1995], [Ильин А.В. 2007] и сервисов (см. с.117).♦

A.2.4. Классы базовых задач

Изучение свойств и закономерностей s-моделирования необходимо, чтобы выявить классы задач, подлежащих исследованию. Назовём их *классами базовых задач s-моделирования*. На данном этапе исследований определены восемь классов.

1. S-представление (представление s-сообщений)

♦ Системы символов, кодов, адресации, s-машинных команд; протоколы взаимодействия и др.; языки и системы программирования; языки запросов, спецификации задач, систем понятий и др. ♦

2. S-преобразование (преобразование s-сообщений)

♦ Символ \rightleftharpoons код; речь \rightleftharpoons текст; аналоговое \rightleftharpoons цифровое; несжатое \rightleftharpoons сжатое; назашифрованное \rightleftharpoons зашифрованное; преобразование форматов (♦ *.doc \rightleftharpoons *.pdf; *.raw \rightleftharpoons *.jpg ♦) и др. ♦

3. S-распознавание (распознавание s-сообщений)

♦ По описанию свойств и форм представления; сопоставлением с образцами (♦ биометрический контроль ♦) и др. ♦

4. S-конструирование (конструирование s-сообщений)

♦ Построение по определённым правилам (используя ранее созданные элементы): программ s-машин, моделей чипсетов, s-(машин, сетей, сервисов) и др. ♦

1. Представление моделей произвольных объектов, рассчитанных на восприятие человеком и s-машинами (см. с.87), связано с изобретением языков s-сообщений (см. с.49), удовлетворяющих определённым требованиям. В этом классе изучаются системы символов (см. с.29) и кодов, используемые соответственно в человеко- и s-машинно-ориентированных

языках. К первым относим языки спецификации, программирования, запросов, ко вторым – системы s-машинных команд. Этот класс включает также задачи представления s-данных (см. с.53). В него входят задачи представления моделей систем понятий (см. с.62), на которых интерпретируются сообщения. На верхнем уровне задачной иерархии этого класса находится представление моделей систем знаний (см. с.67).

2. Преобразование типов и форм представления s-моделей позволяет устанавливать соответствия между моделями. Задачи преобразования типов (☼ речевой в текстовый и обратно и др.☼) и форм (☼ аналоговой в цифровую и обратно; несжатой в сжатую и обратно; одной формы представления документа в другую: *.doc в *.pdf ☼) – необходимое дополнение к задачам представления моделей.

3. Сообщение не может быть интерпретировано, если оно не распознано получателем. Необходимым, но не достаточным условием распознавания является представление сообщения в формате, известном получателю. При выполнении этого условия для распознавания сообщения решаются задачи сопоставления с моделями-образцами, либо сопоставления свойств распознаваемой модели со свойствами моделей-образцов.

4. К задачам этого класса относятся задачи конструирования моделей систем понятий, языков, систем знаний, интерпретаторов сообщений на моделях систем понятий; моделей задач [Ильин В.Д. 1989], [Ilyin V.D. 1995], [Ильин А.В. 2007], программирования (как технологии), взаимодействия в s-среде (см. с.93); моделей архитектур s-машин, s-сетей [Ильин В.Д. 2009, 5], сервис-ориентированных архитектур; моделей сообщений и средств их построения,

документов и документооборота. На верхнем уровне иерархии этого класса находятся задачи конструирования моделей s-среды и технологий S-моделирования.

5. Интерпретация s-сообщения предполагает существование принятого сообщения, модели системы понятий, на которой оно должно интерпретироваться, и механизма интерпретации.

✧Глядя на веб-страницу (см. с.121) [Ильин В.Д. 2006, 1] на экране монитора, человек интерпретирует это сообщение, используя системы понятий, хранящиеся в его памяти. Для микропроцессора s-машины сообщениями, подлежащими интерпретации, служат коды (см. с.43) s-машинных команд и данных; для компилятора – код исходного текста программы.✧

5. S-интерпретация (интерпретация s-сообщений)

✧Коды s-машинных команд интерпретируют микропроцессоры s-машин; исходные тексты программ - трансляторы систем программирования; запросы браузеров – веб-серверы.✧

6. S-обмен (обмен s-сообщениями)

✧Взаимодействие в s-среде (человек \rightleftharpoons s-машина; s-машина \rightleftharpoons s-машина); s-сетевые технологии обмена сообщениями (в сетях различных архитектур, масштаба и др.); межсетевое взаимодействие, развитие Интернета; сервис-ориентированные архитектуры и др.✧

7. S-сохранение, накопление и поиск (сохранение, накопление и поиск s-сообщений)

✧Модели архитектур памяти s-машин, накопителей, носителей; файловых систем; технологий сохранения, накопления, поиска и др.✧

8. S-защита (защита s-сообщений и программно-аппаратных средств s-среды от вредоносных воздействий, несанкционированного доступа и применения)

✧Предотвращение проектных, реализационных и эксплуатационных уязвимостей; программно-аппаратная защита и др.✧

6. В этом классе изучаются задачи взаимодействия в s-среде (человек – машина; машина – машина) с типизацией: отправителей и получателей; средств отправки, передачи и получения сообщений; сред передачи сообщений. Изобретаются системы правил обмена сообщениями (s-сетевые протоколы);

архитектуры s-сетей [Ильин В.Д. 2009, 5],
сервис-ориентированные архитектуры; системы
документооборота.

7. Этот класс включает связанные между собой задачи
сохранения, накопления и поиска. Изучаются и типизируются
память и накопители, механизмы управления ими; формы
сохранения и накопления; носители, методы сохранения,
накопления и поиска; базы данных и библиотеки программ.
Изучаются модели предмета поиска (по образцу, по признакам,
по описанию свойств) и методов поиска.

8. Задачи этого класса включают: предотвращение и
обнаружение уязвимостей; контроль доступа; защиту от
вторжений, вредоносных программ [Ильин В.Д. 2006, 2],
перехвата сообщений и несанкционированного применения.

А.3. СИМВОЛЫ, КОДЫ, СИГНАЛЫ



В *s-среде* (см. с.93) с символами (см. с.29) и построенными из них сообщениями (см. с.49) имеют дело люди, с кодами (символов и сообщений) — программные средства, а с сигналами (физическими реализациями кодов) — аппаратные средства *s-машин*.

□ *СИМВОЛ в науке и технике* (англ. Symbol in Science and Technology) – заменитель некоторого объекта (обозначающий заменяемый объект), принадлежащий определённому набору, предназначенному для формирования *сообщений* по заданным правилам. Каждый элемент такого набора наделён совокупностью свойств (одинаковой для всех элементов набора), обеспечивающей применимость в заданной среде формирования, передачи, интерпретации, сохранения сообщений и манипулирования ими (копирования, поиска и др.). □

☼ В русском алфавите буква *а* – заменитель речевого звука, применяемый для формирования текстовых сообщений.



В наборе символов, применяемых в языке системы знаний (см. с.67) информатики СИНФ, изображение (слева) заменяет полное название ИНФОПЕДИИ, представленное текстом *Энциклопедия информатики ИНФОПЕДИЯ*. ☼

☼ Среда современного бумажного документооборота включает средства формирования рукописных и печатных сообщений на бумаге, пересылки их посредством авиа- или иной

(неэлектронной) почты, копирования с помощью устройств для бумажных документов, поиска по карточкам каталогов, хранения в ящиках, шкафах или на полках.✧

✧Электронный документооборот реализован в s-среде, где конструктивными элементами для построения сообщений служат s-символы (см. с.29). В s-среде человек создаёт сообщения с помощью программ (текстовых, графических и видео редакторов) s-машин. Сохраняет эти сообщения на накопителях s-машин, манипулирует s-кодами сохранённых сообщений (копирует, перемещает, удаляет, переименовывает их) – всё это он делает с помощью s-машинных программ.

Пересылка сообщений на любое расстояние осуществляется с использованием различных сетевых сервисов (см. с.117) [электронной почты (e-mail), транспортировки файлов (ftp) и др.]. Поиск сообщений выполняется также с помощью специальных программ, установленных на s-машине пользователя, и сетевых поисковых сервисов. Электронные библиотеки и другие хранилища s-сообщений несопоставимы по уровню совершенства с «бумажными» предками.✧

/ Предложенное истолкование понятия СИМВОЛ отличается от известных в науке и технике

1. Символ рассматривается как элемент набора символов, предназначенного для формирования сообщений по заданным правилам.
2. Роль набора символов может выполнять что угодно [если это "что угодно" наделено совокупностью свойств, обеспечивающей применимость в заданной среде формирования, передачи, интерпретации, сохранения

сообщений и манипулирования ими (копирования, поиска и др.)).

☼ Набор жестовых символов языка глухонемых (Sign Language); система Брайля для слепых (используются фактурные символы, рассчитанные на осязание посредством пальцев рук); набор звуковых символов музыкальной композиции и соответствующий ему набор графических символов нотного письма ([Нотное письмо](#)); набор графических символов шахматной нотации (Chess Notation). ☼

А.3.1. S-символы

□ *S-СИМВОЛ* (англ. S-symbol) – заменитель некоторого объекта, принадлежащий определённому набору, предназначенному для формирования *s-сообщений* по заданным правилам. Каждый элемент такого набора наделён совокупностью свойств (одинаковой для всех элементов набора), обеспечивающей применимость в *s-среде* для формирования, передачи, распознавания, интерпретации, сохранения сообщений и манипулирования ими (копирования, поиска и др.). Представлен в двух формах, одна из которых рассчитана на распознавание и интерпретацию человеком, другая (в форме *s-кода*) – программой *s-машины* (см. с.87). Реализуемое в *s-среде* (см. с.93) средство представления произвольного объекта, наделённое набором свойств, обеспечивающих возможности:

1. распознавания и интерпретации человеком;
2. применения без участия изобретателя;
3. неограниченного числа воспроизведений по заданным правилам (копирования), удаления и перемещения в *s-среде*;

4. применения в задачах s-моделирования [s-(представления, распознавания, преобразования, конструирования, интерпретации, обмена, сохранения, накопления, поиска, защиты)].

Обозначение в S-моделировании (см. с.18) — *s-symbol*.

Специализация *символа* по среде реализации.□

Не предполагается никаких ограничений на виды и типы заменяемых символами объектов: они могут иметь любую физическую сущность, размещение, происхождение и назначение. Символы одного вида могут заменять символы другого вида (то же справедливо и для типов символов).

Воспроизводимость. Оригинальный символ, хранящийся только в памяти человека, который его изобрел, приобретает шанс стать воспроизводимым, когда он представлен в s-среде. Это означает, что новый символ описан его изобретателем с использованием других символов, считающихся известными. Другими словами, воспроизводимость предполагает обязательную опору на уже известные символы и средства их изготовления и применения.

✧Предположим, необходимо создать новую иконку (англ. icon), расширяющую некоторый существующий набор иконок. Для изготовления используем шаблон (англ. template) для иконок этого типа. Этот шаблон является графическим s-символом, рассчитанным на изготовление множества подобных ему s-символов.✧

Дискретность. Символы, используемые человеком, дискретны. Дискретность символов – необходимое условие их различимости. Она так же важна для человека, как для s-машин важна различимость s-кодов.

Пороговая различимость. Определяет допустимое уменьшение размеров визуальных s-символов, продолжительности звучания аудио s-символов и т.д.

/ Неискусственные средства производства и приёма сообщений человеком

// Для производства сообщений человек использует:

1. *органы речи* (голосовые и речевые команды и извещения);
2. *части тела, производящие различные движения* (жесты рук; движения пальцев рук, головы, лица, ног и др.);
3. *глаза* (указание на расположение объекта, его оценка; реакция на полученное сообщение и др.).

♦К настоящему времени (декабрь 2009) в s-среде относительно продуктивно используются движения пальцев рук (посредством клавиатуры, мыши, стилуса, сенсорного экрана и др.).

Распознавание звуковых команд и извещений находится в стадии становления (большинство существующих реализаций пока не обладает качеством, приемлемым для массового применения). Использование глаз как средства производства сообщений, направляемых s-машине, исследовано менее всего.♦

// Для приёма сообщений от s-машин человек использует:

1. *зрение,*
2. *слух,*
3. *осязание,*
4. *обоняние.*

◊ К настоящему времени в s-среде сравнительно продуктивно используются зрение (восприятие текста, неподвижных и подвижных изображений и др.) и слух (восприятие речи, музыки и др.). Осязание используется для приёма вибровывода мобильного телефона, в игровых устройствах и др.; обоняние осваивается (выпускаются приборы для распознавания запахов).◊

А.3.1.1. Виды символов

Каждому виду символов соответствует средство приёма сообщений, которым наделён человек:

1. *визуальный* (зрение);
2. *аудио* (слух);
3. *тактильный* (осязание);
4. *запаховый* (обоняние).

Вид символов делится на типы.

□ Типу $ts[A[at,op]]$ символов соответствует множество A символов, для которых определены набор атрибутов at и семейство op допустимых операций. □

□ **ВИЗУАЛЬНЫЙ ВИД S-СИМВОЛОВ** (англ. Sight S-symbols) — совокупность s-символов, применимых для построения s-сообщений, содержащих неподвижные и подвижные изображения, рассчитанные на восприятие человеком посредством зрения, а программами s-машин — посредством устройств графического и видео ввода.

Обозначение в S-моделировании — $s-symbol[visual]$.

Специализация *s-symbol* по параметру *means* (средства)³:
s-symbol[visual] ≈ s-symbol[::means=sight].□

Значение *sight* выделяет из множества *s*-символов те, которые рассчитаны на зрительный аппарат человека и устройства графического и видео ввода в составе *s*-машин (☼веб-камеры и др.☼). Вывод *s*-сообщений, содержащих визуальные *s*-символы и рассчитанных на приём человеком, осуществляется на экраны мобильных устройств, мониторов и др.

Виду визуальный соответствуют типы:

- графический [служит для построения *s*-сообщений, содержащих неподвижные изображения (фотографии, схемы и др.)];
- видео (для построения *s*-сообщений, содержащих подвижные изображения).

□**АУДИО ВИД S-СИМВОЛОВ** (англ. Audio S-symbols) — совокупность *s*-символов, применимых для построения *s*-сообщений, содержащих звуки, рассчитанные на звукопроизводящие и звукоприёмные средства человека и *s*-машин.

Обозначение в *S*-моделировании — *s-symbol[audio]*.

Специализация *s-symbol* по параметру *means* (средства):
s-symbol[audio] ≈ s-symbol[::means=audio].□

Значение *audio* выделяет из множества *s*-символов те, которые рассчитаны на звукопроизводящие и звукоприёмные средства человека и *s*-машин (голосовой и слуховой аппараты человека; музыкальные инструменты и др.; аудио устройства ввода

3 См. с.16

(микрофон и др.) и вывода (звуковые колонки и др.) в составе s-машин и др.).

Виду *аудио* соответствуют типы:

- *речевой* (для построения s-сообщений, содержащих речевые фрагменты; имеет специализации по национальным языкам и др.);
- *музыкальный* [для построения s-сообщений, содержащих музыкальные фрагменты; имеет специализации по инструментальным средствам (вокал, фортепиано и др.) и др.];
- *специальный (х)* [для построения s-сообщений, содержащих специальные (неречевые и немзыкальные) аудио команды и извещения; ☆специальный (хлопки), специальный (свист) и др.☆].

□ **ТАКТИЛЬНЫЙ ВИД S-СИМВОЛОВ** (англ. Touch S-symbols) — совокупность s-символов, применимых для построения s-сообщений, включающих прикосновения к элементам интерфейсных устройств s-машин, изменения фактуры и температуры их поверхности, рассчитанные на приём человеком посредством осязания, а программами s-машин — посредством клавиатур, сенсорных экранов и других тактильных устройств ввода.

Обозначение в S-моделировании — *s-symbol[touch]*.

Специализация *s-symbol* по параметру *means* (средства):
s-symbol[touch] ≈ s-symbol[::means=touch]. □

Значение *touch* выделяет из множества s-символов подмножество s-символов, рассчитанных на осязательный

аппарат человека и тактильные устройства ввода (☼клавиатура, сенсорный экран и др.☼) и вывода (☼механизм вибровывода мобильного телефона и др.☼) в составе s-машин.

Виду тактильный соответствуют типы:

- *кинетический* (служит для передачи сообщений путём перемещения элементов устройств, предназначенных для интерфейса с s-машиной (☼вибровывод мобильного телефона, ввод с клавиатуры, манипулирование мышью☼);
- *фактурный* (служит для передачи сообщений путём изменения фактуры поверхности (☼плоская → волнистая или ребристая и т.д.☼) элементов устройств, предназначенных для интерфейса с s-машиной;
- *термический* (служит для передачи сообщений путём изменения температуры поверхности элементов устройств, предназначенных для интерфейса с s-машиной).

□**ЗАПАХОВЫЙ ВИД S-СИМВОЛОВ** (англ. Olfaction S-symbols) — совокупность s-символов, применимых для построения s-сообщений, содержащих запахи, рассчитанные на восприятие человеком посредством обоняния, а программами s-машин — посредством устройств распознавания запахов.

Обозначение в S-моделировании — *s-symbol[smell]*.

Специализация *s-symbol* по параметру *means* (средства):
s-symbol[smell] ≈ s-symbol[::means=olfaction].□

Значение *olfaction* из множества s-символов выделяет подмножество тех, которые рассчитаны на обонятельный аппарат человека и устройства распознавания запахов в составе s-машин.

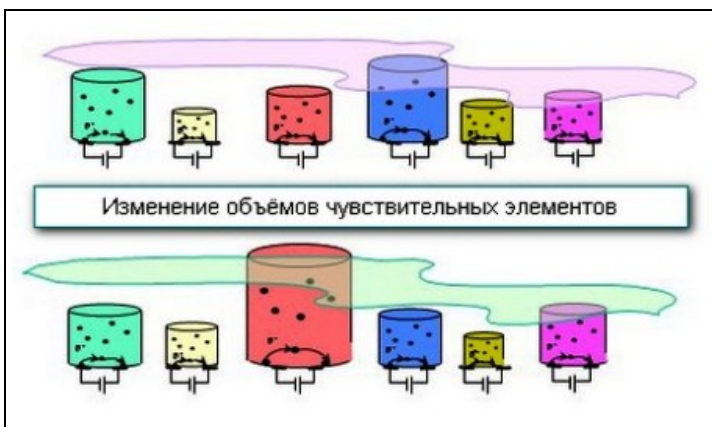
/ S-машинное распознавание запахов

Добавление запаховых s-символов в символьный арсенал современной s-среды – актуальная задача, решение которой сделает возможным продвижение по многим направлениям развития s-интерфейсов *человек-машина* и *машина-машина*.

Задача распознавания запахов – одна из давних задач анализа газовых смесей. К настоящему времени существует немало выпускаемых промышленностью газоанализаторов и распознавателей запахов (применяемых, в частности, для выявления токсичных примесей).

// Технология распознавания запахов (людьми и s-машинными распознавателями) включает:

1. получение исходного сообщения (рецепторами человека или сенсорами s-машинного распознавателя);
2. преобразование исходного сообщения в форму, принимаемую интерпретатором сообщений (для s-машинных распознавателей преобразование в цифровую форму);
3. интерпретация на моделях, хранимых в памяти распознавателя (сопоставление с образцами);
4. формирование результирующего сообщения, рассчитанного на получателя (человека или программу другой s-машины), и отправка.



/// На рисунке показан принцип действия распознавателя запахов «Электронный Нос», основанный на изменениях объёмов чувствительных элементов и вызванных этим изменениях величин электрических сопротивлений этих элементов.

Анализируемая газовая смесь взаимодействует с элементами набора чувствительных элементов. На рисунке изображены два состояния набора из шести элементов: первое (в верхней части рисунка) – в результате реакции со смесью, обозначенной розовым цветом; второе – со смесью, обозначенной светлозелёным. Элементы по разному реагируют на составляющие анализируемой смеси, по разному изменяя свой объём. С изменением объёма изменяется электрическое сопротивление каждого элемента набора, включённого в измерительную электрическую цепь.

Приборы типа «Электронный Нос» выпускаются промышленностью и используются для распознавания запахов и анализа газов (в частности, для выявления токсичных примесей).

А.3.1.2. Специализация и обобщение

□ Пусть задан тип $ts[A[at,op]]$. Специализацией $ts[B<A[at,op]]$ типа $ts[A[at,op]]$ будем называть подмножество $B<A[at,op]$ s-символов с набором атрибутов $at[B]$ ($at[B] > at[A]$) и семейством допустимых операций $op[B]$ ($op[B] > op[A]$).

Тип $ts[A[at,op]]$ будем называть *обобщением* типа $ts[B<A[at,op]]$. □

Текстовый, гипертекстовый (см. с.118) [Ильин В.Д. 2007, 2], *анимационный* и др. – специализации типа *графический*.

Тип *текстовый* служит для построения s-сообщений, содержащих текстовые фрагменты. Элементы множества s-символов типа *текстовый* – это графические изображения букв, знаков препинания и др., для которых задан набор ограничений (на размеры, взаимное размещение и др.).

Тип *числовой* – специализация типа *текстовый* (∅ в семействе операций типа *числовой* есть арифметические операции, которых нет в семействе операций типа *текстовый*). Служит для построения s-сообщений, содержащих числа.

/ Элементарный символ

— символ любого типа, удовлетворяющий условиям пороговой различимости.

☼ Точка на экране площадью в один пиксель – элементарный графический символ. ☼

/ Отношения

// Эквивалентность

Определяет взаимозаменяемость s-символов. Может быть задана для любого числа s-символов (не менее двух).



☼Текст <Энциклопедия информатики ИНФОПЕДИЯ> можно рассматривать как композиционный текстовый символ, который эквивалентен гипертекстовому (см. с.118) [Ильин В.Д. 2007, 2] символу, размещённому слева .

Вибровывоз мобильного телефона (s-сообщение, составленное из тактильных символов) эквивалентен любому из аудио вызовов (☼сообщений, составленных из музыкальных символов☼).☼

// Порядок

Определяет последовательность s-символов.

Отношение порядка на множестве числовых s-символов (натуральный ряд чисел) известно каждому с детских лет. Чтобы задать отношение порядка на любой совокупности символов любого типа, достаточно их перенумеровать (☼алфавит – последовательность текстовых символов☼).

// Принадлежность

Определяет принадлежность s-символа некоторому набору s-символов.

☼Символ □, используемый для выделения утверждений (определений, аксиом и др.), принадлежит набору символов TSM (см. с.11).☼

А.3.1.3. Типы символов

□ВИДЕО s-символ (англ. Video s-symbol) – s-символ, представленный подвижным изображением или совокупностью s-символов, содержащей подвижные изображения. Относится к видео типу s-символов, который принадлежит *визуальному* виду s-символов.

Обозначение в S-моделировании — *s-symbol[video]*.□

□ **ГРАФИЧЕСКИЙ *s*-символ** (англ. Graphic S-symbol) — *s*-символ, представленный неподвижным изображением точки, линии, рисунка, схемы, фотографии или др., применяемый для построения *s*-сообщений, содержащих неподвижные изображения. Относится к графическому типу *s*-символов, принадлежащему визуальному виду *s*-символов.

Обозначение в S-моделировании — *s-symbol[graphic]*.□

✧ Графический *s*-символ задан, если заданы размер матрицы пикселей и цвет каждого из них. ✧

□ **КИНЕТИЧЕСКИЙ *s*-символ** (англ. Kinetic S-symbol) — *s*-символ, представленный перемещением компоненты (✧ корпуса мобильного телефона при вибровывозе, клавиши клавиатуры или др. ✧) *s*-интерфейсного устройства *s*-машины. Принадлежит набору *s*-символов, предназначенному для формирования *s*-сообщений путём механических воздействий. Относится к кинетическому типу *s*-символов, принадлежащему тактильному виду *s*-символов.

Обозначение в S-моделировании — *s-symbol[kinetic]*.□

□ **ТЕРМИЧЕСКИЙ *s*-символ** (англ. Thermic S-symbol) — *s*-символ, представленный различимым наощупь изменением температуры поверхности компоненты *s*-интерфейсного устройства *s*-машины. Принадлежит набору *s*-символов, предназначенному для формирования *s*-сообщений путём изменений температуры поверхности компонент *s*-интерфейсных устройств. Относится к термическому типу *s*-символов, принадлежащему тактильному виду *s*-символов.

Обозначение в S-моделировании — *s-symbol[thermic]*.□

□ **ТЕКСТОВЫЙ *s*-символ** (англ. Textual S-symbol) — *s*-символ, представленный изображением, принадлежащим набору изображений (✧ русский алфавит — подмножество этого набора ✧), который используется для формирования *s*-сообщений, называемых текстовыми (или просто – текстами). Относится к текстовому типу *s*-символов, являющемуся специализацией графического типа *s*-символов, принадлежащего визуальному виду *s*-символов.

Обозначение в *S*-моделировании — *s-symbol*[*text*]. □

□ Упорядоченную совокупность текстовых символов, каждому из которых поставлен в соответствие числовой код, рассчитанный на *s*-машину, называем *текстовым набором s-символов*. □

Текстовый набор символов задаётся путём перечисления всех входящих в него символов. ✧ Текстовый набор в стандарте Unicode (Юникод) включает буквы национальных алфавитов и различные символы (математические, музыкальные и др.), используемые при создании текстовых сообщений с помощью *s*-машин. ✧

Различают рукописные и печатные текстовые символы. Текстовые символы могут быть графическими заменителями речевых символов (✧ буквы русского алфавита – заменители звуков русской речи ✧). Текстовыми наборами символов (включающих специальные символы) представлены алфавиты языков программирования, формальных языков математики и др.

Идея создания графической модели речи, воплощённая в средствах построения текстовых сообщений, сыграла выдающуюся роль в развитии образования, науки, производства и др. областей деятельности.

□ **ФАКТУРНЫЙ s-символ** (англ. Facture S-symbol) — s-символ, представленный различным наощупь изменением рельефа поверхности компоненты (✱волнистая, ребристая или др.✱) s-интерфейсного устройства s-машины. Принадлежит набору s-символов, предназначенному для формирования s-сообщений путём изменений рельефа поверхности компонент s-интерфейсных устройств. Относится к фактурному типу s-символов, принадлежащему *тактильному виду s-символов*.

Обозначение в S-моделировании — *s-symbol[facture]*.□

□ **РЕЧЕВОЙ s-символ** (англ. Speech S-symbol) — s-символ, представленный звуком, принадлежащим набору звуков, используемому для формирования разговорных s-сообщений на определённом языке взаимодействия. Относится к речевому типу s-символов, принадлежащему *аудио виду s-символов*.

Обозначение в S-моделировании — *s-symbol[speech]*.□

□ **МУЗЫКАЛЬНЫЙ s-символ** (англ. Musical S-symbol) — s-символ, представленный звуком, принадлежащим набору, элементы которого отнесены к музыкальным звукам. Применяется для формирования музыкальных s-сообщений. Относится к музыкальному типу s-символов, принадлежащему *аудио виду s-символов*.

Обозначение в S-моделировании — *s-symbol[music]*.□

А.3.2. S-коды

□ *S-КОД* (англ. S-code) – заменитель s-символа или s-сообщения, удовлетворяющий требованиям решения базовых задач *s-* (*представления, преобразования, распознавания, конструирования, интерпретации, обмена, сохранения, накопления, поиска и защиты*) (см. с.23) в *s-среде* ([см. с.93](#)).

Обозначение в S-моделировании — *s-code*.□

Исследователям и инженерам, имеющим дело с различными моделями, свойственно стремление представлять их в числовой форме. Это позволяет применять методы решения задач, которые можно представить в виде программ, рассчитанных на выполнение s-машинами.

/ Числовое кодирование

S-символам любого вида ставят во взаимно однозначное соответствие числа, которые можно представить в памяти s-машины.

✧ ASCII (American Standard Code for Information Interchange) – набор, состоящий из 128 текстовых символов (букв, цифр, знаков пунктуации и др.). Расширенный набор ASCII включает 256 текстовых символов. Каждому символу назначен уникальный номер, называемый кодом ASCII. В этих наборах строчные буквы и заглавные – это разные символы (буквы а строчная и А заглавная и т.д.), которые имеют разные коды.✧

// Представление в памяти s-машин (см. с.87)

Изобретая s-машины, выбирают систему счисления и число разрядов для представления чисел в памяти s-машин. При этом выбор направляется стремлением обеспечить наиболее

эффективное манипулирование числовыми кодами, которое выполняют s-машины.

Выбор ограничен рядом условий, среди которых физико-техническая реализуемость элементов, используемых для построения s-машин. Обычно основание системы счисления выбирают равным количеству устойчивых состояний, в которых могут находиться элементарные составляющие, из которых построена s-машина.

☼Транзисторы современных электронных s-машин имеют два устойчивых состояния. Поэтому в нынешних s-машинах используется двоичное представление числовых кодов программ и данных. ☼

/ Виды

Будем различать *аналоговые коды s-code [analog]* и *цифровые — s-code [digital]*.

Цифровые коды — результат числового кодирования s-символов.

/ Классы и типы

S-коды будем делить на *классы* (по виду s-сигнала, используемого для реализации s-кода): электрические s-коды, оптические и др.

Каждый класс — на *типы*. ☼Класс электрических s-кодов — на частотный тип и др.). ☼

А.3.3. S-сигналы

□*S-СИГНАЛ* (англ. S-signal) — физически реализованное (в виде композиций значений напряжения, частоты или др.) представление *s-кода*, рассчитанное на распознавание и интерпретацию аппаратным средством *s-машины* (микропроцессором, видео контроллером или др.).

Обозначение в S-моделировании — *s-signal*.□

☼В цифровой *s-машине* с двоичным представлением *s-кодов* машинных команд и данных *s-сигналы* — это композиции двух типов импульсов напряжения на выходах транзисторов. Один тип соответствует двоичному нулю, а другой — двоичной единице.☼

А.3.4. Символы, коды, сигналы и эффективность деятельности

Во многих областях деятельности применяемые системы символов имеют определяющее значение для эффективности.

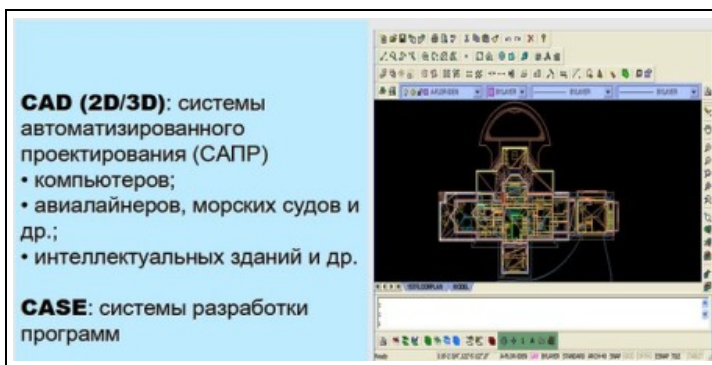
☼Полнота набора дорожных знаков (графических символов, истолкование которых дано в правилах дорожного движения), компактность и смысловая выразительность знаков — всё это влияет на безопасность дорожного движения.☼

Информатизация (см. с.130) [Ильин В.Д. 2008, 3] той или иной деятельности существенно зависит от применяемых систем символов, кодов и сигналов. От степени их соответствия уровню развития *s-среды* (см. с.93) зависят результаты информатизации.

/ Информатизация интеллектуальной деятельности

Арсеналы реализуемых в *s-среде* символов разных типов на каждом этапе во многом влияют на уровень информатизации интеллектуальной деятельности.

☼ Веб (см. с.121) [Ильин В.Д. 2006, 1], мобильная связь, электронная почта, САПРы [системы автоматизированного проектирования; в правой части следующего рисунка — окно IntelliCAD Pro Plus (Autodesk)] и мн. др. — в основу всего этого положено автоматизированное решение указанных базовых задач (см. с.23) S-моделирования. ☼



Примеров успешной информатизации [Ильин В.Д. 2008, 3] интеллектуальной деятельности так много, что из них непросто выбрать. Один из них связан с шахматами. Далее поясним этот поучительный пример.

// Информатизация шахматной игры

Существуют сотни шахматных программ, рассчитанных на установку в мобильных телефонах и коммуникаторах, персональных компьютерах и ноутбуках. Они позволяют пользователю выбрать уровень игры (соответствующий его подготовленности) и многое другое (вид шахматной доски и фигур, подсказки и др.). Существует большой выбор специальных шахматных компьютеров, шахматные сайты позволяют пользователям Интернета (см. с.106), [Ильин В.Д. 2008, 4])

проводить матчи и мн. др. Наряду с другими важными факторами (чёткие правила, немалое число энтузиастов-программистов и др.) результаты и темпы информатизации шахмат во многом зависят от применяемых символьных систем. Шахматисты давно используют компактную текстовую систему записи шахматных партий, где поля доски, фигуры и др. обозначены текстовыми символами. Для анализа позиций используются шахматные диаграммы.



✧ На шахматной диаграмме слева — позиция (которая могла бы возникнуть, если бы чёрные сыграли 5. ... K:e5) в партии Легаль – Сен-Бри, сыгранной в Париже в 18-м веке (1750 или 1787: в определении года истории шахмат расходятся). Вот запись этой партии (с комментарием пятого хода): 1. e4 e5 2. Cc4 d6 3. Kf3 Kc6 4. Kc3 Cg4

5. K:e5?! C:d1?? (опровергал авантюру белых ход 5. ... K:e5!)

6. C:f7+ Kpe7 7. Kd5x.

Рассчитанная на «жадность» чёрных игра в дебюте принесла Легалю победу на 7-м ходу и известность в шахматном мире. В шахматной литературе эту партию называют Мат Легалья. ✧

Компактная текстовая запись шахматных партий, представление позиций с помощью шахматных диаграмм во многом предопределили то, что уже на заре эпохи s-машин начали создавать шахматные программы. Прошло немного времени, и не только любители, но и шахматные профессионалы получили

возможность повышать уровень мастерства, пользуясь шахматными программами. А в эти дни, как известно, лучшие из шахматных программ выигрывают даже у чемпионов мира.

А.4. S-(СООБЩЕНИЕ, ДАННЫЕ, ИНФОРМАЦИЯ)



В этом разделе определены три ключевых понятия информатики⁴: *сообщение, данные и информация*.

Впервые конструктивные определения понятий *сообщение и информация* были предложены в [Ильин В.Д., Соколов И.А. 2006]. Введённые далее определения уточняют предложенные в 2006. Чтобы исключить путаницу, будем называть определённые нами понятия *s-сообщением, s-данными и s-информацией*. Префикс *s-* будем опускать везде, где очевидно, что рассуждения касаются только понятий *s-моделирования*.

А.4.1. S-сообщение

□ **S-СООБЩЕНИЕ** (англ. S-message) — конечная упорядоченная совокупность *s-символов*, рассчитанная на распознавание и интерпретацию получателем-человеком, или её *s-код* (см. с.43), удовлетворяющий требованиям решения базовых задач *s-* (*представления, преобразования, распознавания, конструирования, интерпретации, обмена, сохранения, накопления, поиска и защиты*) (см. с.23) в *s-среде*.

Обозначение в S-моделировании (см. с.18) — *s-message*.□

S-модели систем понятий (см. с.62) и систем знаний (см. с.67), в которых представлены результаты изучения некоторых сущностей (объектов исследований); программы, определяющие

4 □ Наука о символьном моделировании произвольных объектов в человеко-машинной среде (*s-среде*) [наука об *s-моделировании*].□

поведение *s-машин*; веб-страницы (см. с.121) и файлы документов – всё это *s-сообщения*.

В *s-среде* (см. с.93) люди с помощью *s-машин*:

- формируют сообщения, представляя их на языках запросов, программирования и др.;
- выполняют различные преобразования [\rightarrow из аналоговой формы в цифровую и обратно; из несжатой в сжатую и обратно; из одной формы представления документа в другую ($\rightarrow *.doc$ в $\rightarrow *.pdf \rightarrow$)];
- распознают, используют сообщения для конструирования новых сообщений (программ, документов и др.);
- интерпретируют на моделях систем понятий (которые хранятся в памяти интерпретатора также в форме сообщений);
- обмениваются сообщениями [используя при этом программно-аппаратно реализованные системы правил (*s-сетевые протоколы*)];
- сохраняют и накапливают сообщения (создавая электронные библиотеки, энциклопедии и др. *информационные ресурсы* (см. с.133)), занимаются решением задач поиска и защиты *s-сообщений*.

/ Типы сообщений

В *s-среде* реализован *s-интерфейс* двух типов:

- *человек-s-машина* $\approx interface[::sides=h \leftrightarrow m]$;
- *s-машина-s-машина* $\approx interface[::sides=m \leftrightarrow m]$.

// Типы сообщений для $interface[::sides=h \leftrightarrow m]$:

- $сообщение[sides=h \rightarrow m] \approx message[::sides=h \rightarrow m]$;
- $сообщение[sides=m \rightarrow h] \approx message[::sides=m \rightarrow h]$.

/// Для производства сообщений типа $message[::sides=h \rightarrow m]$ человек использует:

1. *органы речи* (голосовые и речевые команды и извещения); при этом порождаются сообщения типа

$message[::(sides=h \rightarrow m)::(means=audio)::audio\ means=speak]$,
являющегося специализацией типа

$message[::(sides=h \rightarrow m)::means=audio]$;

заметим, что $message[::sides=h \rightarrow m] > message[::(sides=h \rightarrow m)::means=audio] > message[::(sides=h \rightarrow m)::(means=audio)::audio\ means=speak]$;

2. *части тела, производящие различные движения* (жесты рук; движения пальцев рук, головы, лица, ног и др.); при этом порождаются сообщения типа

$message[::(sides=h \rightarrow m)::means=kinetic]$;

3. *глаза* (указание на расположение объекта, его оценка; реакция на полученное сообщение и др.); при этом порождаются сообщения типа

$message[::(sides=h \rightarrow m)::means=visual]$.

Для ввода в s-машину сообщений типа

$message[::sides=h \rightarrow m]$

применяются аппаратные средства $hardware[::sides=h \rightarrow m]$ (клавиатуры, микрофоны, фото- и видеокамеры и др.) и программные средства $software[::sides=h \rightarrow m]$, входящие в состав ОС и различных редакторов (текстовых, графических, аудио, видео и др.).

/// Для приёма сообщений типа

$message[::sides=m \rightarrow h]$ человек использует:

1. зрение,
2. слух,
3. осязание,
4. обоняние.

Для вывода сообщений из s-машины, рассчитанных на человека, применяются программные средства $software[::sides=m \rightarrow h]$ (входящие в состав ОС, различных редакторов и др.) и аппаратные средства $hardware[::sides=m \rightarrow h]$ (дисплеи мониторов, ноутбуков, коммуникаторов, фото- и видеокамер и др., вывод на которые реализуют видео контроллеры; наушники, звуковые колонки и др., вывод на которые осуществляют аудио контроллеры и др.).

// Тип сообщений для $interface[::sides=m \leftrightarrow m]$:

$сообщение[::sides=m \leftrightarrow m] \approx message[::sides=m \leftrightarrow m]$.

S-машины, обменивающиеся сообщениями типа

$message[::sides=m \leftrightarrow m]$,

как правило, связаны между собой проводными или беспроводными средствами в составе *s-сети* (локальной, *Интернет* (см. с.106) или др.), а передача и приём сообщений осуществляются в соответствии с одним из сетевых протоколов.

А.4.2. S-данные

□ *S-ДААННЫЕ* (англ. *S-data*) – *s-сообщение*, необходимое для решения некоторой *s-задачи* (см. с.71) и представленное в форме, рассчитанной на *s-*(распознавание, преобразование и интерпретацию) решателем этой задачи (программой *s-машины* (см. с.87) или взаимодействующим с ней человеком).

Специализация s-сообщения (s-message) по параметру получатель s-сообщения (s-recipient), значением которого является решатель s-задачи (s-solver):

$s\text{-}data \approx s\text{-}message[::s\text{-}recipient=s\text{-}solver]$.

Обозначение в S-моделировании — *s-data*. □

Человек воспринимает *s-data* в *символьной* форме (текст, числа, звук, изображения и др.), а программа *s-машины* – в *кодовой* (см. с.43).

☼ При съёмках цифровой фотокамерой сообщение, представляющее собой световой сигнал (см. с.45), воздействует на светочувствительную матрицу, распознаётся ею, а затем преобразуется в цифровой код. Этот код интерпретируется программой, улучшающей изображение. Полученный результат преобразуется и записывается (на встроенный накопитель или на карту памяти) как *графический файл* с выбранным пользователем расширением (*jpg*, *raw* или др.). ☼

◊ В *s-среде* (см. с.93) сообщение не относим к *s-data*, если не указано, для решения какой задачи сообщение предназначено, и

если оно не представлено в форме, рассчитанной на s-(распознавание, преобразование и интерпретацию) решателем задачи.♦

/ Типы элементов s-data (типы данных)

После распознавания программой s-машины выполняется поэлементное s-преобразование s-data в коды этих элементов [в соответствии с описаниями типов (числовых, текстовых и др.) этих элементов и форматов, заданных в программе].

□ *Тип s-data (тип данных)* определяет множество значений и совокупность допустимых операций. □

/ Структуры однотипных совокупностей элементов s-data (структуры данных)

Чтобы работу с s-data сделать более эффективной, связанные между собой однотипные элементы s-data представляют в виде различных структур s-data (массивов, списков и др.).

/ Форматы элементов s-data (форматы данных)

Ввод s-data и вывод результата решения задачи выполняются в соответствии с описаниями форматов, заданных для элементов s-data (числовых, текстовых, графич., аудио или видео).

На машинных носителях (жёстких дисках, картах памяти и др.) s-data хранятся в виде файлов [как правило, имеющих расширение, указывающее тип программы, на которую рассчитаны s-data (☼NEC-model.pdf - файл с расширением pdf, рассчитанный на любую из программ семейства Adobe Acrobat☼)].

Сжатие – это s-преобразование, выполняемое для уменьшения размера s-data.

Реализуется, чтобы уменьшить объём памяти для хранения файла, времени передачи по s-сетям [Ильин В.Д. 2009, 5].
Применяют различные алгоритмы сжатия. Одни из них обеспечивают сжатие без потери качества (✧*mp3* – для аудио✧), другие – с потерей (✧*jpg* – для неподвижных изображений✧).

А.4.3. S-информация

□**S-ИНФОРМАЦИЯ** (англ. S-information) – результат *s*-(интерпретации сообщения) на *s*-модели системы понятий.

Для извлечения информации из сообщения необходимо иметь:

1. принятое сообщение (представленное в форме, рассчитанной на *s*-распознавание и *s*-интерпретацию получателем сообщения);
2. хранящиеся в памяти *s*-модели систем понятий (см. с.62), среди которых – необходимая для интерпретации принятого сообщения;
3. механизмы поиска необходимой *s*-модели, интерпретации сообщения, представления результата интерпретации в виде *s*-сообщения и записи его в память.

Обозначение в S-моделировании — *s-information*.□

Схема извлечения информации из сообщения:



☼Запрос веб-клиента (см. с.121) – сообщение, интерпретируемое веб-сервером. Веб-страница, сформированная для отправки веб-клиенту, – информация, полученная в результате интерпретации на s-модели. Отправленная веб-сервером веб-страница – отправленное сообщение. Она же, принятая веб-клиентом, – принятое сообщение. Результат интерпретации принятого сообщения – экранное представление веб-страницы, рассчитанное на восприятие человеком [Ильин В.Д. 2006, 1].☼

◇Итак, информация – это то, что извлекается из сообщения путём интерпретации на s-модели системы понятий.

Человеку для этого нужно воспринять само сообщение (для этого нужно, как минимум, владеть языком, использованным при формировании сообщения), отыскать в памяти требуемую систему понятий (если она там есть) и, наконец – интерпретировать это сообщение на выбранной системе понятий.

☼Не знаешь языка – не извлечёшь информацию. Не располагаешь необходимыми системами понятий – не извлечёшь (☼для интерпретации дорожных знаков надо знать правила дорожного движения; чтобы решить геометрическую задачу, надо знать соответствующий раздел геометрии.☼).☼

Изменила ли извлечённая информация совокупность хранящихся в памяти систем понятий, повлияла ли на механизм поиска требуемой системы и механизм интерпретации – ответы на эти вопросы вынесены за рамки предложенного определения.◇

А.4.4. О понятийной неразберихе в «трёх соснах (сообщение, данные, информация)»

◊Продолжающаяся традиция неразличения *информации* и *сообщения*, возможно, где-то и безвредна, но не в информатике. *Информацию* продолжают передавать и получать, сохранять и накапливать, искать и защищать (с ней делают то же, что с *сообщением*). Для полноты неразберихи к этой паре добавляют и *данные*.◊

Возможно, для так называемого массового пользователя такое смешение понятий не имеет значения. Но здесь рассуждаем о *системе знаний информатики*.

◊Порядок в понятийном аппарате науки — одна из составляющих характеристики её развития.◊

✧Вот, что написано об *информации* в статье с одноимённым названием, опубликованной в Большой Российской Энциклопедии (бумажное издание, т. 11, 2008, с. 493):
«...сообщение или сигнал, совокупность данных, сведения, рассматриваемые в контексте их содержания, структурной организации, динамики (процессов создания, передачи, восприятия, использования, репрезентирования, анализа, хранения и т.п.).» [автор не указан; видимо, кто-то из философов].

На с. 494 того же тома истолкование понятия *информация в математике* предложил Ю.В. Прохоров. Его дефиниция в бессодержательности не уступает философской:

«...общее название понятий, играющих фундаментальную роль в информатике, информации теории, кибернетике, а также в математической статистике. В каждой из этих дисциплин интуитивное представление об И. относительно к.-л. величин или явлений требует своего уточнения и формализации.»

А вот что на 495 с. об *информации в праве* поведал И.Л. Бачило: «...сведения (сообщения, данные) независимо от формы их представления. И. как материальный и нематериальный объект является элементом любых отношений...»☼

Итак, для Ю.В. Прохорова *информация* — «...общее название понятий, играющих фундаментальную роль...», а для двух других истолкователей — «...сообщение или сигнал...»; «...сведения (сообщения, данные)...»⁵

/ Пояснение для неинформатиков: на примерах

☼Книга, кинофильм, музыкальное произведение, картина в художественной галерее — это *сообщения*. Бывает, что книгу прочитанную в юности, перечитывают в зрелом возрасте. То же бывает и применительно к кинофильмам, музыкальным произведениям и картинам. Нередко результаты интерпретации того же самого сообщения существенно отличаются.

Причина: изменились системы понятий (хранящиеся в памяти), на которых человек интерпретировал то же самое сообщение. То есть информация, извлечённая из сообщения, изменилась потому, что изменились системы понятий, используемые для интерпретации сообщения, и изменился механизм интерпретации.☼

☼Пытаясь помочь сыну или дочери решить школьную задачу по математике, родитель, успешно справлявшийся с такими задачами в свои школьные годы, не может решить задачу.

5 Предложение проф. В.Д. Ильина (научного консультанта по информатике редакции «Техника» Большой Российской Энциклопедии) написать статью *ИНФОРМАЦИЯ в информатике* не было поддержано. Объяснили сложившимся в БРЭ раскладом: статью ИНФОРМАЦИЯ пишут философы, математики и др., ...но не информатики.

Причина: системы понятий, нужные для интерпретации условия задачи, не отыскиваются в памяти.✧

A.4.5. Суть неудач машинного перевода

Человек, получив *сообщение* на языке *А*, сначала интерпретирует полученное сообщение на имеющихся в его памяти *моделях систем понятий*, чтобы понять смысл сообщения (извлечь информацию). После этого приступает к формированию сообщения на языке *Б*, стремясь передать тот же смысл.

✧Современные системы машинного перевода не содержат s-моделей систем понятий и механизмов интерпретации сообщений на этих моделях. Потому и...✧

Мало того, и занимающиеся задачей машинного перевода на исследовательском уровне продолжают разрабатывать подходы, в которых нет идеи интерпретации сообщений на s-моделях систем понятий.

Попробуйте, например, программу ПЕРЕВОДЧИК (Prompt Professional 8.0), о которой её разработчики в марте 2008 писали, как об одной из лучших на рынке программ машинного перевода. Вот образец её умения переводить с русского на английский:

"Результат перевода этого предложения с русского языка на английский будет использован для обратного перевода." Это она перевела так: "The result of transfer of this offer from Russian on English will be used for return transfer." А обратный перевод (своего же) выглядит вот так: "Результат передачи этого предложения от русского языка на английском языке будет использоваться для передачи возвращения."

/ Какой должна быть схема машинного перевода



A.5. S-МОДЕЛИ СИСТЕМ ПОНЯТИЙ И СИСТЕМ ЗНАНИЙ



□ **МОДЕЛЬ** в науке и технике (англ. Model in Science and Technology) — символьное или физическое представление изучаемого объекта, реализованное в выбранной среде моделирования. Выполняется при ограничениях, соответствующих цели исследований, условиям реализации моделирования и применения модели. Адекватность модели определяется степенью её соответствия задачам, для решения которых она создана, и точностью результатов, получаемых при решении этих задач.□

Физическая модель (ph-модель) – представление изучаемого объекта (*st-объекта*), реализованное в физической среде моделирования (*pt-среде*); *символьная модель (sm-модель)* – реализованное в символьной среде моделирования (*sm-среде*).

Как правило, построение символических моделей требует меньших затрат, чем построение физических.

◇ Развитие *s-среды* (см. с.93) увеличивает возможности *S-моделирования* (см. с.18), которое служит методологическим основанием этого развития. S-модели исследуемых объектов позволяют изучить множество различных состояний изучаемых объектов в заданных условиях. Продуктивность проектирования (в электронике, машиностроении, строительстве и др.) возрастает по мере роста адекватности s-моделей проектируемых объектов.◇

◇*S-среда* представляет собой специализацию *st-среды* по средствам моделирования: для реализации в *s-среде* каждой *s-модели* необходимо поставить в соответствие *s-код* (см. с.43), удовлетворяющий требованиям решения базовых задач (см. с.23) *S-моделирования* с помощью *s-машин*.◇

◇*S-моделирование* (см. с.18) – специализация *st-моделирования* по среде моделирования.◇

А.5.1. S-модель системы понятий

□*S-МОДЕЛЬ СИСТЕМЫ ПОНЯТИЙ* (англ. S-model of Conceptual System) — это пара

<память $mem[sc]$ модели sc системы понятий sC >,

<семейство $rel(mem[sc])$ связей, заданных на $mem[sc]$ >

(где $[sc]$ – помета). Здесь $mem[sc]$ и $rel(mem[sc])$ соответствуют совокупности понятий моделируемой системы sC и семейству связей, заданных на этой совокупности.

Результат *S-моделирования*.□

S-модель любого объекта представляет собой модель некоторой системы понятий.

В науке и технике особое внимание сосредоточено на *s-моделях*, где семейства связей $rel(mem[sc])$ представлены в форме разрешимых задач (задавая значения некоторого подмножества элементов памяти $mem[sc]$, можно вычислять значения других элементов).

✧Элементарным примером системы понятий с разрешимыми задачными связями между элементами памяти является система

понятий *треугольник* (в *s*-модели *tr* этой системы стороны *a*, *b*, *c*, периметр *p* являются элементами памяти; а связь $p = a + b + c$ — элементом семейства связей).☼

☼Методология конструирования разрешающих структур на задачных графах (вершины которых соответствуют элементам памяти системы понятий, а рёбра — связям между ними; см. с.82) стала теоретическим основанием одного из продуктивных подходов к автоматизации программирования:

[Ильин В.Д. 1989], [Ilyin V.D. 1995], [Ильин А.В. 2007].☼

S-модель системы понятий относится к символьным моделям, существование которых возможно только в виде *s*-сообщений (см. с.49), для которых средой реализации служит *s*-среда.

☼Сообщения, хранящиеся на носителях, с которых *s*-машины не могут считывать и на которые не могут записывать (☼память человека☼) не являются *s*-сообщениями.☼

☼Читаемая Вами (☼вероятнее всего, с помощью программы *Acrobat Reader*☼) гипермедийная книга представляет собой *s*-сообщение, сформированное в *s*-среде (на *s*-машинах авторов с помощью программ пакета OpenOffice.org, работающих под управлением операционной системы Windows Web Server 2008 R2). Это сообщение содержит гипермедийное описание *Основ теории s-моделирования*.☼

/ Тип системы понятий

□*Тип X системы sC понятий* ≈ множество *X* понятий системы *sC* понятий, для которой определена *s*-модель .

Может иметь *подтипы*, называемые *специализациями типа X*, и

надтипы, называемые обобщениями типа X .□

// Специализация типа понятий

□ *Специализация типа X системы sC понятий* – это порождение подтипа $X[::rule]$ (здесь сдвоенное двоеточие $::$ — *символ специализации*), соответствующего системе $sC[rule]$ понятий с семейством связей, расширенным добавлением связи $rule$.

Выделяет подмножество $X[::rule]$ множества X .

Специализацией называем и результат $X[::rule]$ этого порождения ($X > X[::rule]$).□

☼ $tr_L \approx tr[::angle=\pi/2]$. То есть тип tr_L системы *прямоугольный треугольник* – специализация типа tr системы *треугольник*, полученный путём добавления связи $angle=\pi/2$. Выделяет из множества треугольников подмножество тех, у которых величина одного из углов равна $\pi/2$ (здесь $angle=\pi/2$ используется и как имя связи).☼

/// Специализация типа системы понятий, заданная последовательностью добавляемых связей

$X[::(rule1)::rule2]$ – специализация типа $X[::rule1]$ по связи $rule2$. Число специализирующих связей в последовательности не ограничено. При этом имена связей, предшествующие последнему, заключены в круглые скобки, а перед открывающей скобкой каждой пары скобок – сдвоенное двоеточие.

☼ $message[::(interface=h->m)::means=touch]$ – специализация типа $message[::interface=h->m]$, определяющего множество сообщений, соответствующих интерфейсу *человек–s-машина*, по связи $means=touch$, выделяющей множество сообщений,

вводимых в s-машину посредством прикосновений (☼пальцев рук к клавишам клавиатуры или сенсорному экрану☼).☼

// Обобщение типа понятий

□*Обобщение типа Z* – это порождение его надтипа $Z[\#rule]$ путём ослабления (здесь # – символ ослабления) связи *rule* из семейства связей системы понятий, соответствующей типу *Z*. Исключение связи считаем её предельным ослаблением.□

☼Тип *выпуклые многоугольники с числом углов не более 6* можно рассматривать как обобщение типа *треугольник*, полученное ослаблением связи, ограничивающей число углов.☼

☼*modeling* \approx *symbol modeling*[#tools] – *моделирование (modeling)* можно рассматривать как обобщение типа *символьное моделирование (symbol modeling)*, полученное путём исключением связи *tools (средства моделирования)*.☼

A.5.2. S-знание

В обиходе почти каждый (кроме укоренённых агностиков) полагает для себя известным значение слова "знать". Знаю — и всё тут!

Наука — «почемучка» по определению. Недостаточно ей даже такого сильного аргумента как "и все тут!". Потому и в теории S-моделирования не может оставаться неопределённым важнейшее понятие ЗНАТЬ.

Определим его (в понятийной системе информатики)⁶.

□*ЗНАТЬ* — состояние, когда выходное *сообщение*, полученное в результате интерпретации входного, распознаётся, как уже известное.

6 Без философского тумана.

А потому не вызывает изменений в моделях понятийных систем, хранящихся в памяти.□

Знать «как решать задачу определенного типа» — готовность успешно интерпретировать сообщения, представляющие собой условия задачи этого типа. То есть, готовность отыскать в памяти алгоритм решения и применить его [или готовность получить такой алгоритм путём внутренних рассуждений (вывести алгоритм)].

□*S-ЗНАНИЕ* (англ. S-knowledge) — это s-(комплексное умение решать задачи (см. с.71) определённого класса, включающее умения выполнить:

1. приём и распознавание сообщений, содержащих условия задач этого класса;
2. приведение условия задачи к форме, принятой в решателе задач (s-машинной программе или решателе-человеке);
3. интерпретацию приведённого условия задачи на модели задачной области и запись результата в память (для последующего исследования на состоятельность);
4. проверку состоятельности результата путём использования его в условиях проверочных задач; если результат признан состоятельным, задача считается решённой).□

S-знание можно истолковывать и как комплексное умение извлекать информацию (см. с.55) из сообщений (см. с.49), содержащих условия задач определённого класса.

S-знание не предполагает никаких ограничений на класс задач: это могут быть задачи распознавания образов, перевода с одного языка на другой или иные классы задач.

◇S-машинный перевод сообщений (см. с.49), составленных на входном языке *A*, в сообщения на выходном языке *B* не может быть успешным, если метод перевода не включает интерпретацию сообщений на *s*-моделях систем понятий.◇

A.5.3. S-модель системы знаний

□S-МОДЕЛЬ СИСТЕМЫ ЗНАНИЙ (англ. S-model of Knowledge System) – это триада

$\langle sa \approx s\text{-модель системы } Sa \text{ понятий} \rangle$,

$\langle set[lng] \approx s\text{-модель совокупности языков сообщений, интерпретируемых на } sa \rangle$,

$\langle set[intr] \approx s\text{-модель совокупности интерпретаторов на } sa \text{ сообщений, составленных на языках из } set[lng] \rangle$.□

/ Интерпретация сообщения на модели *sa*:

1. построение выходного сообщения (извлечение информации (см. с.55)) по заданному входному;
2. анализ выходного сообщения (требуется ли изменения в модели *sa*);
3. если требуется, то – изменение модели *sa*; если нет – завершение.

/ Об s-модели языка сообщений

// Необходимым условием построения модели языка сообщений является существование моделей системы понятий, на которой предполагается интерпретировать сообщения, составленные на языке, и базовых типов символов, композиции которых предполагается использовать для построения системы символов языка. Эти модели играют роль исходных для построения языка.

// Построение модели языка сообщений включает разработку моделей:

1. композиции базовых типов *символов*;
2. системы символов языка, построенной на основе модели композиции базовых типов символов;
3. системы правил конструирования сообщений с использованием модели системы символов.

/ Об s-модели интерпретатора сообщений

// Необходимым условием построения модели интерпретатора сообщений является существование моделей входного и выходного языков, а также – системы понятий, на которой должны интерпретироваться сообщения, составленные на входном языке.

// Построение модели интерпретатора включает разработку моделей:

1. распознавания сообщений на принадлежность входному языку;
2. интерпретации распознанных сообщений на модели системы понятий;

-
3. представления результата интерпретации в виде сообщения на выходном языке.

А.6. ЗАДАЧНЫЕ КОНСТРУКТИВНЫЕ ОБЪЕКТЫ (S-ЗАДАЧИ) И СИСТЕМА ЗНАНИЙ ОБ S-ЗАДАЧАХ



Существует огромное число задач, запрограммированных на различных языках и реализованных в составе системных и прикладных комплексов. Какие это задачи, к каким классам отнесены? Где можно найти сведения об их формулировках, методах, алгоритмах и тестовых примерах? Как использовать этот арсенал при разработке программ?

Идея автоматизации программирования заключается в том, чтобы использовать ранее полученные результаты в последующих разработках. Как должны быть представлены эти ранее полученные результаты, чтобы служить основой автоматизированной разработки?

Уже на начальных этапах автоматизации программирования вместе с языками и трансляторами стали создавать и различные библиотеки программ. Что требуется для объединения библиотек и тех программ, которые успешно работают, но не входят ни в какие библиотеки? Какой должна быть система знаний о программируемых задачах, чтобы служить основанием для конструирования программных систем?

Один из возможных ответов на этот вопрос был предложен в [Ильин В.Д. 1989] применительно к порождению программ. В [Ильин В.Д. 1995] этот подход был применен в методологии конструирования параллельных программ.

А.6.1. Задачный конструктивный объект (s-задача)

□*S*-ЗАДАЧА (задачный конструктивный объект) [англ. *s*-problem (problem constructive object)] – это триада $\{Formul, Alg, Prog\}$, где *Formul* – постановка задачи, *Alg* – множество алгоритмов, поставленных в соответствие постановке задачи, *Prog* – объединение множеств программ, каждое из которых поставлено в соответствие алгоритму из *Alg*.

Постановка задачи *Formul* – это пара $\{Mem, rul\}$, где *Mem* – *s*-представление множества понятий задачи, на котором задано разбиение $Mem = Inp \cup Out$ ($Inp \cap Out = \emptyset$), а *rul* – правило, задающее на *Mem* бинарное отношение $Rel[rul(Mem)] \subset Inp \times Out$.

Назовем множество *Mem* *памятью задачи*, а *Inp* и *Out* – *входом* и *выходом задачи*, значения которых предполагается соответственно задавать и искать□.

Исходим из того, что одной формулировке могут соответствовать несколько алгоритмов, а каждому алгоритму – несколько программ. Различия между алгоритмами из множества *Alg* определяются задаваемым описанием применения (ограничениями на предельный размер задачи, точность полученного результата и др.), а различия между программами – выбранными языками программирования и операционными системами.

♦Каждая программа сопровождается обязательной ссылкой на набор тестовых примеров, что необходимо для проверки ее работоспособности [Ильин А.В. 2007].♦.

В общем случае множества *Prog* и *Alg* могут быть пустыми (число элементов этих множеств зависит от степени изученности задачи). Если ни один из алгоритмов множества *Alg* не

запрограммирован, $Prog = 0$. Если в соответствии постановке задачи не поставлен ни один алгоритм, $Alg = 0$ и $Prog = 0$.

□ *Спецификация $spec$ s -задачи* – это пара $(Formul, as)$, где as – описание применения. □

✧ *Формулировка задачи линейного программирования*. Вход задачи $Inp = \{ \text{матрица } a [i = 1...m, j = 1...n] \text{ коэффициентов ограничений, вектор } b [i = 1...m] \text{ правых частей ограничений, вектор } c [j = 1...n] \text{ коэффициентов целевой функции} \}$. Выход $Out = \{ \text{искомый вектор } x [max; j = 1...n] \}$. Правило rul максимизации по $x [j = 1...n]$ целевой функции $c [j = 1...n] * x [j = 1...n]$ при ограничениях $a [i = 1...m, j = 1...n] * x [j = 1...n] \leq b [i = 1...m]$ и $x [j = 1...n] \geq 0$ имеет следующий вид:

$max [x [j = 1...n]: a [i = 1...m, j = 1...n] * x [j = 1...n] \leq b [i = 1...m], x [j = 1...n] \geq 0] (c [j = 1...n] * x [j = 1...n])$. ✧

/ Связи по памяти между s -задачами

Связи по памяти между s -задачами определяются тремя типами функций, каждая из которых является функцией двух аргументов и позволяет поставить в соответствие паре s -задач некоторую третью s -задачу, образованную из этой пары.

□ *S -задача a связана с s -задачей b по памяти*, если существует хотя бы одна пара элементов $\{elem\ Mem[a], elem\ Mem[b]\}$, принадлежащих памяти $Mem[a]$ s -задачи a и памяти $Mem[b]$ s -задачи b , относительно которой определено общее означивание (элементы имеют одно и то же множество значений). Пусть S и H – множества s -задач и $D \leq S * S$. Если каждой паре $(s[i], s[j])$ элементов из D ставится в соответствие определенный элемент из H , то будем говорить, что задана *функция связи по памяти* $h = conn (s[i], s[j])$.

При этом D будем называть областью определения функции $conn$ и обозначать $D[conn]$. Множество $R = \{h: elem\ H; h = conn(s[i], s[j]); s[i]: elem\ D[conn], s[j]: elem\ D[conn]\}$ будем называть областью значений функции $conn$.

Тип связи зависит от содержимого пересечения по памяти: составлена ли связь из элементов выхода одной и входа другой задачи; из элементов выходов задач или из элементов их входов; или же связь получена путем комбинации предыдущих способов.

Будем обозначать функцию связи по памяти типа *вход-вход* через $conn[x]$, *выход-вход* – через $conn[ux]$ и *выход-выход* – через $conn[y]$.

/ Родовые связи между s -задачами

S -задача может быть *прообразом* некоторого непустого множества s -задач или *образом* некоторого прообраза; или быть одновременно и образом какой-то одной s -задачи, и прообразом некоторого множества других s -задач.

Предусмотрены следующие типы родовых связей между s -задачами:

s -(*специализация задачи*) — указание на s -задачу, частную по отношению к исходной;

s -(*обобщение задачи*) — указание на s -задачу, которая служит обобщением исходной.

/ Конструирование s -задачи

S -(*конструирование задачи*) реализуется посредством связи по памяти между задачами.

Элементарная задачная конструкция — это задачная пара.

Из задачных пар можно построить более сложную задачу конструкцию, если рассматривать их как задачные элементы. Любая задачная конструкция, в свою очередь, может быть использована как составляющая еще более сложной задачной конструкции.

/ Конкретизация s-задачи

□ *S*-(конкретизация задачи) — это переход (формулировка → алгоритм → программа).□

Для *s*-задач, имеющих пустое множество программ ($Prog = 0$), конкретизация сводится к выбору или разработке алгоритма. Если и $Alg = 0$, *s*-задача может быть использована в *s*-(конструировании задач), но не может быть конкретизирована.

/ Атомарная s-задача

S-задача называется *атомарной*, если её формулировка не представлена в виде структуры, заданной на некотором множестве формулировок других *s*-задач.

Будем также говорить об атомарной *s*-задаче как о *простой s-задаче*. *Простая задача* (с точки зрения строителя задачных конструкций) не наделена внутренней структурой и потому не подлежит декомпозиции. Простые задачи используются для создания конструкций. Каждая созданная задачная конструкция может быть объявлена некоторой новой задачей. В свою очередь, эти новые задачи вместе с атомарными могут быть применены при конструировании задач.

А.6.2. Система знаний об s-задачах

□ Система *pS* знаний о задачных конструктивных объектах (*p*-объектах, называемых также *s*-задачами) — это триада

$\langle pA, lng, intr \rangle$, где pA – задачная область, lng – язык спецификации p -объектов, $intr$ – интерпретатор спецификаций искомых p -объектов на pA \square .

/ Модель задачной области

Пусть P – множество всех p -объектов, а $A \subset P$ – его непустое подмножество. При этом в A (содержащем не менее двух элементов) не существует ни одного элемента, который не был бы связан по памяти хотя бы с одним элементом из A .

\square *S-модель pa задачной области pA* – это p -объект, который задаётся парой \langle память $mem[A]$ множества задач A задачной области pA \rangle , \langle семейство $rul(mem[A])$ связей, заданных на $mem[A]$ \rangle .

Непустое множество $mem[A]$ элементов памяти разбито на три подмножества: *входов $inp[A]$ задач*, *выходов $out[A]$ задач* и подмножество $or[A]$, каждый из элементов которого является и входом, и выходом некоторых задач. Любое одно из этих подмножеств может быть пустым; могут быть одновременно пустыми $inp[A]$ и $out[A]$ \square .

В отличие от памяти задачи, состоящей из входа и выхода, *память задачной области* содержит подмножество *or* элементов памяти, каждый из которых может быть или задан (как входной), или вычислен (как выходной). Будем называть такие элементы памяти *обратимыми*, а *or* – *подмножеством обратимых элементов*. Подмножество inp будем называть *подмножеством задаваемых*, а подмножество out *подмножеством вычисляемых элементов*.

Задачная область pa служит *s-моделью* [Ильин А.В. 2007], на которой интерпретируются спецификации искомых задач, составленные на языке lng .

// Разрешающая структура

◇Интерпретация заключается в постановке в соответствие некоторому подмножеству (или паре подмножеств) памяти $mem[A]$ некоторой подобласти задачной области pA , названной *разрешающей структурой*. Формально интерпретация спецификации искомого p -объекта на pA – это конструктивное доказательство существования разрешающей структуры.◇

/ Задачные графы

Задачный граф служит формальным представлением *задачной области*, рассчитанным на реализацию процесса *p-конструирования* и формализацию *задачных знаний*.

Множество вершин графа составлено из *задачных объектов*. Оно называется *задачным базисом графа* и обозначается *p-basis*. Ребро задачного графа — это пара вершин с непустым пересечением по памяти. Нагрузка ребра определяется множеством всех пар элементов памяти, входящих в это пересечение. Каждая вершина графа имеет память. *Память вершины* — это память задачи (или задачной области), которую представляет вершина.

□ *Составная задача comp* – это подобласть задачной области pA , которая содержит не менее двух элементов из множества задач A и на памяти которой задано разбиение: $mem[comp] = inp[comp] \cup out[comp]; inp[comp] \cap out[comp] = \emptyset$, определяющее вход $inp[comp]$ и выход $out[comp]$ составной задачи. Составной задаче поставлен в соответствие ориентированный граф, вершинами которого являются задачи. Каждая вершина помечена именем задачи. Ребра графа — это пары задач с непустыми пересечениями по памяти □.

◇Составная задача может быть построена путём последовательного применения функций связи по памяти.◇

// Типы задачных графов

В зависимости от состава вершин определим следующие *типы задачных графов*:

- *U-граф* имеет множество вершин только из простых задач;
- в *C-графе* хотя бы одна вершина представлена составной задачей и нет вершин, представляющих собой задачу область;
- в *G-графе* — не менее одной вершины представлено задачей областью (остальные могут быть простыми и составными задачами).

/// U-граф

□ Связный граф с непустым множеством ребер и задачным базисом $p\text{-basis}$, все элементы которого являются простыми задачами, называется *U-графом* и обозначается $U\text{-graph}$:

$U\text{-graph} = (p\text{-basis}, \text{set}[\text{ver}])$, где $\text{set}[\text{ver}]$ – множество ребер задачного графа. Объединение памяти задачных вершин, составляющих базис, называется *памятью U-графа* и обозначается $\text{mem}[U\text{-graph}]$.

На памяти *U-графа* задано разбиение: *подмножеством* $\text{Giv}[U\text{-graph}]$ *задаваемых элементов памяти* называется подмножество входных элементов; *подмножеством* $\text{Comput}[U\text{-graph}]$ *вычисляемых элементов памяти* называется подмножество выходных элементов; *подмножеством* Or *обратимых элементов памяти* называется разность $(\text{mem}[U\text{-graph}]) \setminus (\text{Giv}[U\text{-graph}] \cup \text{Comput}[U\text{-graph}])$ □.

/// С-граф

□ Связный граф с непустым множеством ребер и задачным базисом, в составе которого есть хотя бы одна составная задача и нет задачных областей, называется *С-графом* и обозначается *C-graph*. □

/// G-граф

□ Связный граф с непустым множеством ребер и задачным базисом, в составе которого есть хотя бы одна задачная область, называется *G-графом* и обозначается *G-graph*. Память *С-графа* и память *G-графа* обозначаются и определяются аналогично памяти *U-графа*. □

/ G-графы как средство формализации знаний о р-объектах

Система знаний об s-задачах обеспечивает процессы р-(специализации, конкретизации и конструирования).

◇Возможность существования в задачном графе одного или нескольких узлов, являющихся задачными областями, имеет принципиальное значение для формализации задачных знаний.◇

Конкретным воплощением задачной области может быть граф любого типа (*U*-, *C*- или *G*-граф). Тот факт, что *G*-граф может замещать задачный узел, открывает логически неограниченные возможности для усложнения задачной области. Она может быть представлена, в частности, посредством *G*-графа, базис которого состоит только из вершин, представленных *G*-графами [Ильин А.В. 2007].

А.7. ИСЧИСЛЕНИЕ S-ЗАДАЧ И КОНСТРУИРОВАНИЕ РАЗРЕШАЮЩИХ СТРУКТУР



Объединение множеств базовых задачных объектов и сконструированных объектов обозначается через P и называется *множеством p -объектов*.

Совокупность пространств, построенных на подмножествах множества P , называется *миром p -объектов* (или *миром задач*).

Процесс работы с задачными объектами реализуется по принципу «от общего к частному». В системе задачного конструирования существуют пространства задачных конструктивных объектов, одни из которых содержат представленные в самом общем виде объекты, другие – объекты, полученные путем специализации объектов-прообразов.

Множество пространств p -объектов (*мир p -объектов*) не имеет логических ограничений на расширение.

А.7.1. Исчисление задачных конструктивных объектов (исчисление s -задач)

□ *Исчисление задачных конструктивных объектов I_{gen}* (которое для краткости называем также *исчислением s -задач*) — это построение некоторого множества T конструктивных объектов (t -объектов, обозначаемых через t) на основе множества B базовых конструктивных объектов (b -объектов, обозначаемых через b) по правилам R построения t -объектов из b -объектов и ранее построенных t -объектов.

Формально $Igen$ — это четвёрка (Ing, B, T, R) , включающая язык Ing описания объектов и конструкций из объектов, непустое множество B базовых объектов, порождаемое множество T ($B \wedge T = O$; до начала процесса задачного конструирования $T = O$) и множество R правил построения объектов \square .

Символы $conn[x]$, $conn[yx]$, $conn[y]$ обозначают функции трёх типов, применение которых позволяет построить t -объект из пары уже существующих объектов или конструкций из объектов. Символ pre — обозначение прообраза некоторого непустого множества объектов. Этот символ используется в объявлениях и в функциях. Символ im — обозначение образа некоторого объекта (рассматриваемого как прообраз). Этот символ употребляется (так же, как и символ pre) в объявлениях и функциях.

Множество B состоит из базовых объектов (b -объектов), являющихся объектами первого поколения. Иначе говоря, любой b -объект не имеет прообраза, но может иметь образы. Любому b -объекту может быть поставлено в соответствие некоторое непустое множество объектов-образов $set(t[im; p(b)]) = im(b)$ (в левой части символ im является верхней пометой, а $p(b)$ — нижней, обозначающей указатель на объект-прообраз b).

Каждый объект-образ $t[im; p(b)]$ имеет единственный объект-прообраз. Любой объект-образ может, в свою очередь, иметь образы. В этом случае он является прообразом своих образов и образом некоторого прообраза.

Объект, множество образов которого пусто, будем называть *конечным объектом*.

Объекты, имеющие прообраз и непустое множество образов, будем называть *промежуточными объектами*.

Множество T строится на базисе B по правилам R . В T могут существовать конструкции, построенные из объектов, принадлежащих B , конструкции из b -объектов и ранее построенных t -объектов, а также конструкции из t -объектов.

Определены правила построения t -объектов, в соответствии с которыми работает механизм задачного конструирования при создании конструкций из b -объектов, b - и t -объектов, а также из t -объектов.

/ Правила построения t -объектов

rul[1]: $t \mid p(b) = b$. Правило устанавливает, что объект $t \mid p(b)$ множества T может быть построен из одного объекта b , принадлежащего множеству B . Слева от вертикальной черты « \mid » стоит буква t , обозначающая некоторое имя объекта из T ; после вертикальной черты размещён указатель на объект, из которого построена конструкция; буква p – символ указателя, а в скобках – имя объекта.

rul[2]: $t \mid p(im(t)) = im(t)$. Объект $t \mid p(im(t))$ множества T , являющийся образом объекта t , может быть получен p -специализацией t .

rul[3]: $t \mid p(pre(t)) = pre(t)$. Объект $t \mid p(pre(t))$ множества T , являющийся прообразом объекта t , может быть получен p -обобщением t .

rul[4]: $t \mid p(conc(t)) = conc(t)$. Любой объект, полученный путем p -конкретизации, может быть t -объектом.

rul[5]: $t \mid p(t[] \mid p(f,s)) = t[*] \mid p(f,s)$.* Правило устанавливает, что t -объектом может быть любая допустимая конструкция $t[*] \mid p(f,s)$. Конструкцию $t[*] \mid p(f,s)$ получаем, применяя функцию связи по памяти $conn[*](f,s)$, т. е. $t[*] \mid p(f,s) = conn[*](f,s)$. Помета $*$ принимает одно из трёх значений (x, ux, y), каждое из которых

обозначает определённый тип функции $conn[*]$ (имена объектов-аргументов указываются в круглых скобках сразу после символа функции $conn[*]$). Конструкция $t[*] / p(f, s)$ получена путём применения функции $conn[*]$, аргументами которой служит пара (f, s) , для которой допустимы следующие значения:

$(f: elem(B), s: elem(B));$

$(f: elem(B), s: elem(T));$

$(f: elem(T), s: elem(B));$

$(f: elem(T), s: elem(T)).$

А.7.2. Разрешающие структуры на задачных графах

Рассмотрим принцип действия механизма конструирования на задачном графе. Искомая конструкция задаётся спецификацией задачи, содержащей описание её памяти, ограничений на число задачных узлов (и, если необходимо – ограничений, связанных с размером задачи, точностью результата и др.). Заданное описание интерпретируется на задачном графе, который служит представлением интересующей конструктора задачной области. Средством интерпретации спецификаций задач служит механизм конструирования на задачном графе.

Интерпретация на U-графе в процессе задачного конструирования заключается в постановке в соответствие подмножеству (или паре подмножеств) элементов его памяти такого подграфа, память которого находилась бы в заданном отношении к введённому подмножеству (или паре подмножеств). Интерпретации на C-графе и G-графе аналогичны интерпретации на U-графе.

□ *Задача t представима на задачном графе $graph$, если вход $inp[t]$ задачи содержится в подмножестве $Giv[graph] \cup Or[graph]$, а выход $out[t]$ – в подмножестве $Comput[graph] \cup Or[graph]$ памяти задачного графа; при этом существует не менее одной задачи из базиса этого графа, вход которой содержится в $inp[t]$ или совпадает с ним.* □

□ *Разрешающей структурой $solv[t]$ на графе $graph$, поставленной в соответствие некоторой задаче t , называется подграф с минимальным числом задачных вершин, на котором задача t представима.* □

Интерпретация задачного узла U -графа (или C -графа) в процессе поиска разрешающей структуры заключается в соотнесении означенности входа и выхода.

Смысл этого определения поясняют *правила интерпретации задачного узла*:

- *если полностью означен вход, то полностью означен и выход;*
- *если означенным полагается хотя бы один элемент выхода, то означенным полагается полностью вход.*

Механизм построения разрешающих структур ставит в соответствие спецификации исходной задачи подграф на задачном графе путём реализации трёх типов поведения в соответствии с тремя типами запросов на конструирование.

Для каждого из трёх типов запросов в [Ильин В.Д. 1989] было получено конструктивное доказательство существования разрешающей структуры соответствующего типа. Там эти доказательства были получены применительно к задачным

сетям. Схемы доказательств существования разрешающих структур на задачных графах аналогичны.

После того как найдена разрешающая структура, начинается процесс ее конкретизации в соответствии со спецификацией условий применения исходной задачи.

/ О наборах тестовых примеров в системе знаний о задачах

Применительно к конструированию разрешающих структур на задачных графах существование набора тестовых примеров для каждого р-объекта – необходимое условие реализации предложенной технологии. Заметим, что тестирование полученной разрешающей структуры входит в конструктивное доказательство ее существования.

♦Нынешние библиотеки программ различного назначения (в том числе – динамически присоединяемые *dll*) не удовлетворяют требованиям, предъявляемым к s-модели задачного конструктивного объекта. В частности, они не содержат ссылок на наборы тестовых примеров. Нет возможности протестировать составляющие нынешних операционных систем и работающих под их управлением приложений.

Правило – продавать программные продукты с заранее оговоренной возможностью доступа к наборам тестовых примеров (выложенных на сайте производителя) – заметно уменьшит число ошибок и уязвимостей на этапе разработки и увеличит вероятность их выявления после начала эксплуатации.

Известно, что составление наборов тестовых примеров требует понимания не только сути задачи, выполняемой тестируемой программой, но и допустимых условий ее применения. Если правило – предъявлять потребителю наборы тестовых примеров – станет обязательным, изменится отношение разработчиков и к

изготовлению таких наборов, и к производству программных продуктов.◊

/ Значение

Конструирование разрешающих структур на задачных графах позволяет путем интерпретации интерактивно формируемой спецификации искомой задачи получить структуру задач с известными алгоритмами.

Построение множества задачных конструктивных объектов в составе системы знаний об *s*-задачах основано на исчислении *p*-объектов.

Важными нововведениями по сравнению с [Ильин В.Д. 1989] является то, что задачные сети заменены задачными графами и в *s*-модель *p*-объекта введена гиперссылка на набор тестовых примеров.

Примером применения метода построения разрешающих структур на задачных графах может служить интерактивный преобразователь ресурсов с изменяемыми правилами поведения [Ильин А.В., Ильин В.Д. 2004], [Ильин А.В., Ильин В.Д. 2005].

Б. МЕТОДОЛОГИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ПРОЕКТИРОВАНИЯ S-СРЕДЫ И ИНФОРМАТИЗАЦИИ



Успешное проектирование s-машин и s-среды, изобретение информационных технологий и сервисов (см. с.117) предполагает непрерывное совершенствование методологического обеспечения этих видов научно-технической деятельности.

Теория s-моделирования рассматривается как платформа для создания методологического обеспечения проектирования s-среды и технологий информатизации (см. с.130) различных видов деятельности.

Без такой платформы развитие s-среды осуществляется почти рефлексивно (отражая коммерческие устремления участников её строительства). В частности, по коммерческим соображениям продлевается жизнь устаревших решений (аппаратных, программных, сервисных и др.), многие из которых родились на заре компьютерной эпохи.

Неразвитость методологического обеспечения информатизации [Ильин В.Д. 2008, 3] в соединении с ИТ-неподготовленностью заказчиков приводит к тому, что вместо средства повышения продуктивности информатизация нередко становится дорогостоящим маскарадом.

Б.1. S-МАШИНЫ



□ **S-МАШИНА** (англ. S-machine) — программно-аппаратно реализованное сооружение для решения задач *S-моделирования*. Элемент *s-среды* (см. с.93).□

☼ Суперкомпьютеры, мейнфреймы, персональные компьютеры, ноутбуки, коммуникаторы, мобильные телефоны, GPS-навигаторы, цифровые фото- и видеокамеры — всё это *s-машины*.☼

Клавиатуры, мыши, трекболы, тачпады и др. устройства ввода жестовых символов — это составляющие *s-машин*, выполняющие *s-преобразования* в коды (см. с.43), воспринимаемые драйверами⁷ соответствующих устройств. Мониторы персональных компьютеров, дисплеи ноутбуков, коммуникаторов и др. мобильных *s-машин* — это составляющие, которые выполняют *s-преобразования* кодов, направляемых видеоконтроллерами *s-машин*, в символьные композиции, рассчитанные на зрительный канал человека.

Задачи *s-интерпретации* решают человек (☼ разглядывающий изображение на мониторе ☼), программы (☼ драйверы,

⁷ **S-ДРАЙВЕР** (англ. S-driver) — программа, по запросам других программ управляющая определённой аппаратной составляющей *s-машины* (мышью, принтером или др.). Обозначение в *s-моделировании* — *s-driver*.□

Обычно реализуется как аппаратно-зависимое расширение исполнительной системы ОС.

компоненты ОС, различные редакторы и др.☼), аппаратно реализованные интерпретаторы (☼микропроцессоры и др. ☼).

/ О форме представления сообщений, подлежащих интерпретации:

человеку – символьная, программам – кодовая, аппаратно реализованным частям s-машин – сигнальная.

/ Люди в s-среде (по определению) отнесены к masters (хозяевам), а s-машины к slaves (рабам)

В проекте s-среды предполагается следующее распределение ролей:

1. Цели выбирают люди.
2. Люди делают «развёртку» целей в комплексы задач.
3. Часть достаточно изученных задач люди относят к числу решаемых с помощью s-машин. Остальные продолжают изучать (используя инструментальные возможности s-среды).
4. Задачи, отнесенные к числу решаемых с помощью s-машин, специфицируются и программируются людьми (с помощью средств s-среды). В итоге (после тестирования и т.д.), программы решения изученных задач пополняют арсенал s-среды.
5. Люди-пользователи выбирают те средства s-среды, которые помогают им в их деятельности (☼профессиональной, бытовой и др.☼)

/ Видеоклип (с некоторыми фактами из истории развития s-машин)

S-интерфейс

□ *S-ИНТЕРФЕЙС* (англ. S-interface) — способ и средства взаимодействия пользователя с программами s-машин, программ между собой или с аппаратными средствами s-машин, а также аппаратных средств между собой. Обозначение в *s-моделировании* — *s-interface*. □

/ Интерфейс пользователя

Определяет взаимодействие человека с операционной системой (ОС) и прикладными программами (приложениями), работающими под её управлением.

Наиболее распространёнными аппаратными средствами реализации данного вида интерфейса являются: клавиатура, мышь, джойстик, экран монитора персонального компьютера, ноутбука, коммуникатора и др.

// Графический интерфейс

В большинстве ОС применяется графический интерфейс пользователя (graphical user interface, GUI) в котором для экранного отображения ввода команд пользователя и ввода/вывода данных используются окна (windows) – области экрана, каждая из которых относится к одной из работающих программ. Элементы управления программой отображаются внутри окон в виде меню, графических символов, полей ввода и др. Выбор и активация одного из элементов обычно осуществляются с помощью мыши, клавиатуры, джойстика или прикосновением к сенсорному экрану. Программы могут осуществлять вывод данных (✧ на экран ✧) в виде текста, картинок, таблиц и др.

Основы стандартов графического интерфейса пользователя были заложены компанией Apple (США), выпустившей в 1984 персональный компьютер Macintosh с установленной на нём ОС MacOS. Применение стандартных графических элементов управления в различных программах облегчает пользователю освоение новых программ.

// Интерфейс командной строки и голосовой интерфейс

Другими видами интерфейса пользователя являются интерфейс командной строки (текстовые команды вводятся пользователем с клавиатуры) и голосовой (голосовые команды вводятся с помощью микрофона и при успешном распознавании выполняются программой). В одной программе могут быть реализованы различные виды интерфейса пользователя.

// Жестовый интерфейс



В 2007 компания Microsoft (США) анонсировала программно-аппаратное решение Microsoft Surface (изображение-гиперссылка), позволяющее управлять

программами s-машин с помощью движений рук перед сенсорной поверхностью, за которой находятся несколько цифровых видеокамер, фиксирующих движение, и цифровой проектор, создающий изображение на поверхности. Такой интерфейс позволяет нескольким пользователям работать одновременно без применения мыши и клавиатуры, а также переносить в память s-машины данные с помещённых на эту поверхность объектов.

/ Интерфейс программ

Способ и средства взаимодействия программных объектов (приложений, библиотек программ, компонентов ОС) называют *интерфейсом программ*.

Описание функций и структур *данных* (см. с.53) программных объектов, выполненное на языке программирования или языке описания интерфейсов, называют *интерфейсом программирования приложений* (англ. application programming interface, сокр. API). Такое описание позволяет разработчикам вставлять в коды создаваемых программных объектов вызовы исполняемых кодов других (ранее разработанных) объектов. Это дает возможность повторно использовать программные объекты, созданные разными разработчиками.

// Стандартизация интерфейсов программ

Позволяет программам, написанным на различных языках программирования, обмениваться командами и данными с помощью *сообщений* определенного *формата*⁸. При этом взаимодействующие программы могут работать как на одной s-машине, так и на разных (являющихся узлами *сети s-машин*).

/ Интерфейс приложений с аппаратными средствами

Это интерфейс реализует ОС. Она обеспечивает работу исполняемых *кодов* (см. с.43) программных объектов (в т.ч., *драйверов* устройств s-машин), передавая необходимые команды центральному процессору s-машины.

- 8 □ *S-ФОРМАТ* (англ. S-format) — спецификация s-представления сообщения, необходимая для распознавания и интерпретации этого сообщения получателем [человеком (при этом сообщение имеет символическое представление) или s-машиной (сообщение представлено s-кодом)]. Обозначение в s-моделировании — *s-format*.□

/ Интерфейс аппаратных средств s-машин

Осуществляется посредством шин, разъёмов, кабелей, а также средств беспроводного взаимодействия, реализующих различные технологии радио интерфейса. [☼Bluetooth (от англ. Bluetooth – синий зуб) – для взаимодействия (обычно на небольших расстояниях) Bluetooth-мыши с ноутбуком, для обмена *файлами*⁹ между мобильным телефоном и ноутбуком или др.☼].

Разъёмы s-машин, к которым подсоединяются устройства или др. s-машины, называют *портами*.

USB-порты (universal serial bus port – порт универсальной последовательной шины) используются для подключения таких устройств с USB-интерфейсом, как фото- и видеокамеры, флэш-накопители, мыши, клавиатуры, принтеры и др.☼

/ Стандартизация интерфейса

Актуальной научно-технической задачей является стандартизация интерфейсов всех видов. От её решения зависят эффективность проектирования, изготовления и применения программных и аппаратных средств s-машин, а в итоге – производительность и надёжность функционирования s-среды [Ильин А.В. 2008, 2] .

9 ☐S-ФАЙЛ (англ. S-file) — поименованная единица хранения s-кода сообщения (данных или программы) на накопителе s-машины. Обозначение в s-моделировании — *s-file*.☐

Б.2. S-СРЕДА



□ *S-СРЕДА* (англ. *S-environment*) — объединение взаимодействующих *s-сетей* и отдельных *s-машин* (см. с.87), используемых для решения задач *s-моделирования* (см. с.18) и применения полученных результатов. Средство информатизации (см. с.130) различных видов деятельности.

Обозначение в *s-моделировании* — *s-environment*. □

Современным воплощением ядра *s-среды* является *Интернет* (см. с.106) [Ильин В.Д. 2008, 4].

Каждая *s-модель*, хранящаяся в *s-среде* — это некоторое сообщение, рассчитанное на интерпретацию получателем определенного типа [Ильин В.Д., Соколов И.А. 2006].

Спецификации программируемых задач интерпретируют программисты.

Исходные тексты написанных ими программ — программы-трансляторы (*компиляторы*, интерпретаторы, ассемблеры).

Инструкции, из которых состоят исполняемые программы, интерпретируют микропроцессоры *s-машин*.

Сообщения, представленные в форме аудио- и видеофайлов — соответствующие программы-плееры.

Сообщения, поступающие от *веб*-серверов в ответ на запросы пользователей и представляющие собой веб-страницы, интерпретируют программы-браузеры.

Графические (текст, неподвижные и подвижные изображения), аудио и механические сообщения, выводимые соответственно на экраны мониторов, колонки аудиосистем (или наушники) и корпуса, напр., мобильных устройств (вибровыводы) – такие сообщения интерпретируют люди.

Книга, созданная писателем, музыкальная композиция, произведение художника и т.д. – всё это сообщения, рассчитанные на получателей различных типов. Успешная интерпретация сообщений позволяет извлечь информацию (см. с.55).

/ Изобретатели и строители s-среды

Изобретатели систем *символов* и систем кодов (см. с.43), систем машинных команд и языков программирования, трансляторов, сетевых архитектур и протоколов, сервис-ориентированных архитектур, чипсетов и т.д. – это творцы, деятельность которых осуществляется в s-среде и направлена на методологическое обеспечение проектов ее непрерывно продолжающегося строительства.

Их продукция служит основанием для творчества программистов, разработчиков аппаратных составляющих s-машин и др. (строителей s-среды). Все вместе они изобретают и строят s-среду.

/ Пользователи s-среды

Ими являются все, кто применяет средства s-среды. В их число входят изобретатели и строители s-среды. Программисты и проектировщики, использующие САПРы (системы

автоматизированного проектирования) различного назначения; дизайнеры, композиторы, художники, литераторы, создающие свои произведения с использованием различных редакторов, установленных на их s-машинах (ноутбуках, персональных компьютерах и др.) – все они пользователи s-среды.

Изобретатели, строители и другие пользователи s-среды неразрывно связаны ею.

Их деятельность непрерывно увеличивает число s-моделей систем понятий и систем знаний, хранящихся и использующихся в s-среде. Вместе они увеличивают потенциал s-среды, который, в свою очередь, позволяет увеличить потенциал каждого из них.

/ Электронная и бумажная формы сообщений

В эти дни преобладают две формы документального представления сообщений: в s-среде – электронная (файлы на CD или DVD, веб-сайты и др.), вне её – бумажная (книги, брошюры и др.).

При этом большинство бумажных документов получают путём распечатки соответствующих им электронных. В частности, бумажные книги выпускают, используя т.н. оригиналы-макеты, представляющие собой электронные документы. Бумажная форма часто используется для дублирования электронной. На начальном этапе создания продукции интеллектуальной деятельности современные авторы пользуются различными редакторами для построения сообщений, установленными на s-машинах. Поэтому рождаются сообщения в электронной форме и сохраняются в виде файлов.

Пока еще сохраняющаяся потребность в бумажных формах связана с тем, что:

-
- часть населения не пользуется s-машинами (по разным причинам);
 - среди пользующихся s-машинами есть те, кто иногда или постоянно предпочитает бумажные формы для того, чтобы читать произведения.

// Возможности: изобразительные, навигационные и др.

По богатству изобразительных, навигационных и др. возможностей (в частности, связанных с распространением произведений) электронные гипермедийные [Ильин В.Д. 2007, 2] сообщения, используемые, напр., в веб-сервисах [Ильин В.Д. 2006, 1], нелепо сравнивать с бумажными. Сочетание в них текста, картинок, аудио- и видео- составляющих и наличие удобных навигационных средств (гиперссылок, позволяющих вызывать различные сервисы (см. с.117) (почтовый, поисковый и др.) или произвольно перемещаться между документами; перекрёстных ссылок, предназначенных для произвольного перемещения внутри документа) – всё это несопоставимо с тем, что может дать самая роскошная книга с цветными иллюстрациями.

☀Читая (интерпретируя) гипермедийную книгу, можно быстро выяснить значение непонятого слова (перейдя к электронному словарю или энциклопедии) и снова вернуться; можно посмотреть видеоклип, послушать аудиозапись (напр., с комментарием автора). Таких возможностей довольно много и с каждым месяцем становится всё больше.

Гипермедийная библиотека объемом, превышающим в разы бумажные библиотеки ярых библиофилов, легко уместится на нескольких жёстких дисках. Отправляясь в отпуск, командировку или еще куда-то, можно закачать избранные вещи на карту

памяти своего коммуникатора или на жёсткий диск
субноутбука.☼

// Обновления

Электронная форма даёт возможность относительно легкого обновления хранящегося сообщения (дополнения, изъятия каких-то фрагментов; изменений в оформлении, исправления ошибок и др.). Если сообщение (напр., научная статья) размещено на веб-сайте, то процесс обновления выполняется довольно просто. Сразу после его завершения пользователи имеют возможность ознакомиться с обновленным произведением (с точным указанием содержания выполненного обновления).

// Комментарии, форумы, рецензии

Существование произведения в форме электронного сообщения позволяет автору выбрать приемлемые для него дисциплины комментирования, форумов и рецензирования (если он выкладывает произведения на своём сайте). Если же – на сайте творческого сообщества, то всё это должно соответствовать правилам сообщества.

// Авторское право, плагиат

☼Выложил автор произведение на сайте своего творческого сообщества. Поступившее сообщение автоматически зарегистрировано (на него заведена электронная карточка). Любое обновление регистрируется в этой карточке. Скрыть улики хищения значительно труднее, чем в случае с бумажной формой.

// Писатель, художник, композитор, исполнитель: в одном лице

Вспомните рисунки А.С. Пушкина и М.Ю. Лермонтова. Многие творцы научных, художественных и др. произведений имеют склонность и способности делать неподвижные, а в наши дни и

подвижные изображения (анимации, видеоклипы и др.), дополняющие тексты их книг. То же можно сказать и о художниках, скульпторах и музыкальных композиторах: редко, кто из них откажется от авторского комментария.

Более того, немало авторов имеют склонность к авторскому исполнению своих произведений. S-среда служит инструментарием, позволяющим творцам использовать при создании своих произведений сочетания выбранных ими символьных систем и непрерывно прирастающий арсенал сообщений, хранящихся в ней.

◊Ближайшее будущее s-среды определено интенсивным развитием разнообразных сервисов (образовательных, коммерческих, развлекательных и др.).◊

Б.3. S-СЕТЬ



□ *S-СЕТЬ* (англ. Network) — система, состоящая из *s-машин* (см. с.87), взаимодействующих по единым правилам, определённым сетевыми протоколами.

Предназначена для совместного пользования различными *s-сервисами* (см. с.117) (электронной почтой, поисковыми системами и др.), *информационными ресурсами*, программами (☼ программами серверов приложений ☼) и аппаратными средствами (жёсткими дисками, принтерами и др.) *s-машин*. Служит конструктивным элементом при построении *s-среды* (см. с.93).

Обозначение в *s-моделировании* — *network*.□

/ Основы построения и функционирования

Для объединения *s-машин* в *s-сеть* необходимы сетевые аппаратные средства (сетевые адаптеры, установленные на компьютерах; коммутаторы, маршрутизаторы, кабели и др.) и программные средства, реализующие правила взаимодействия программных и аппаратных компонент *s-сети*, которые определены сетевыми протоколами.

Сетевые протоколы соответствуют действующим в *s-сети* сетевым технологиям (комплексам программно-аппаратно реализованных методов, определяющих функционирование *s-сети*).

// По назначению программные и аппаратные составляющие s-сети разделяют на три уровня:

- на верхнем – прикладные программы (серверы и клиенты электронной почты, веб-серверы и браузеры [Ильин В.Д. 2006, 1] и др.);
- на среднем – программные средства, реализующие сетевые протоколы;
- на нижнем – сетевые аппаратные средства.

// Узлы s-сети, сетевая адресация и обмен сообщениями между узлами

S-машины в составе s-сети, имеют уникальные (в её пределах) сетевые адреса (принадлежащие единому сетевому адресному пространству) и называются *узлами s-сети*.

Формат и интерпретация сетевого адреса определяются применяемым сетевым протоколом (☼если применяется IP-протокол, то узлы получают IP-адреса [Ильин В.Д. 2008, 4]☼).

Взаимодействие между узлами осуществляется путём обмена *сообщениями*, автоматически разбитыми на части определённого формата, называемые сетевыми пакетами. Разбивку на пакеты, их сжатие (при необходимости) и др. осуществляют программы, реализующие сетевые протоколы.

Оптические или электрические сигналы (см. с.45), соответствующие значениям битов в составе сетевых пакетов, передаются по медным или оптоволоконным кабелям, а также – с использованием радио или др. видов беспроводной связи.

Проводные и беспроводные среды передачи сигналов s-сети называют *сетевой средой передачи*. Её свойства определяют

скорость прохождения сигналов и допустимые расстояния их эффективной передачи.

Наибольшая скорость достигается при передаче по оптоволоконным кабелям, а наименьшая – в беспроводной среде (применяемой для подключения перемещаемых компьютеров, а также когда прокладка кабелей нецелесообразна или невозможна).

// По размеру территории, охватываемой s-сетью, различают:

- локальная s-сеть \approx LAN (Local Area Network); обычно размещается в помещениях здания или группы зданий;
- s-сеть, охватывающая территорию региона, страны или др. \approx WAN (Wide Area NetWorks);
- глобальная s-сеть \approx GAN (Global Area Networks) ✨самой крупной глобальной сетью является Интернет (см. с.106) [Ильин В.Д. 2008, 4] ✨.

/ Сетевые протоколы

Эффективность функционирования s-сети во многом определяется применяемыми сетевыми протоколами.

7	Прикладной
6	Представление данных
5	Сеансовый
4	Транспортный
3	Сетевой
2	Канальный
1	Физический

S-сети, построенные с использованием аппаратных и программных средств разных производителей, успешно взаимодействуют, если они соответствуют стандарту OSI (англ. Open System Interconnection – взаимодействие открытых систем), разработанному Международной организацией стандартизации (International Standards Organization – ISO).

Концепцию этого стандарта обычно поясняют, пользуясь т.н. семиуровневой моделью OSI взаимодействия открытых систем.

Модель OSI представляет собой иерархическую систему стандартизованных на международном уровне сетевых протоколов, где множество протоколов разбито по функциональному назначению на семь взаимодействующих подмножеств (т.н. уровней абстракции).

Протоколы каждого уровня решают строго определенный комплекс задач и взаимодействуют только с протоколами смежных уровней (протоколы уровня 1 предоставляют сервисы протоколам уровня 2, те – протоколам уровня 3 и т.д.). Протоколы уровня 7 пользуются сервисами предоставляемыми протоколами уровня 6, но не предоставляют никаких сервисов протоколам других уровней; протоколы уровня 1 не пользуются сервисами других уровней.

В узле-отправителе программы, реализующие протоколы, включаются в работу, начиная с уровня 7, который отвечает за обмен сообщениями между приложениями (✱браузером и веб-сервером [Ильин В.Д. 2006, 1]✱). На этом уровне находятся протоколы DNS [Ильин В.Д. 2007, 1] системы отображения доменных имен в IP-адреса, HTTP транспортировки гипертекстовых [Ильин В.Д. 2007, 2] документов и др. В узле-получателе протоколы реализуются в обратном порядке, начиная с уровня 1, отвечающем за доставку потоков битов, согласование сигналов и др. (на этом уровне действуют протоколы *Ethernet*, *Token Ring* и др.).

/ Сетевые архитектуры

Сетевая архитектура определяет функциональное назначение узлов s-сети и методы их взаимодействия. Выбор сетевой

архитектуры при создании s-сети определяется требованиями к её быстродействию, надёжности и др.

В настоящее время наиболее распространены сетевые архитектуры *клиент-сервер* (англ. Client/Server network) и *точка-точка* (англ. peer-to-peer, сокр. P2P).

// Клиент-сервер

В s-сети с архитектурой *клиент-сервер* узлы разделены по функциональному назначению на клиенты и серверы. Клиенты отправляют запросы только серверам и не принимают запросы. Серверы обрабатывают запросы и отправляют клиентам запрошенное (веб-страницу [Ильин В.Д. 2006, 1], сообщение электронной почты или др.). Серверы могут отправлять запросы друг другу (☼ при поиске *файла*☼).

// Точка-точка

В s-сети с архитектурой *точка-точка* (англ. peer-to-peer, сокр. P2P), называемых также пиринговыми, реализован принцип ролевой эквивалентности узлов, каждый из которых является одновременно и сервером, и клиентом.

С ростом числа узлов пиринговой s-сети растут её эффективность и надёжность. Пиринговая s-сеть сохраняет работоспособность даже при небольшом числе функционирующих узлов.

☼ Архитектуру *точка-точка* имеют, файлообменные сети (для быстрого и надёжного обмена файлами). ☼ Пиринговые сети эффективны также для распределённых вычислений, применяемых при решении сложных задач, решение которых даже на суперкомпьютере выполнялось бы недопустимо долго.

// Гибридные сети

В гибридных s-сетях, сочетающих быстродействие клиент-серверных и надёжность пиринговых, реализованы элементы обеих архитектур.

/ Топологии s-сетей

Топология определяет конфигурацию связей между узлами s-сети.

Различают топологии s-сети:

- *точка-точка* (англ. point-to-point),
- *шина* (bus),
- *звезда* (star),
- *кольцо* (ring),
- *решётка* (mesh).

В s-сети с топологией *точка-точка* каждые два узла соединены между собой непосредственно.

В s-сети, имеющей топологию *шина*, все узлы имеют общее соединение (при этом каждый узел получает не только сетевые пакеты, адресованные ему, но и – другим узлам).

В s-сети с топологией *звезда* все узлы взаимодействуют через один общий узел (центр звезды), который может быть пассивным (не сортирующим проходящие через него пакеты) или активным (направляющим каждому узлу только адресованные ему пакеты).

В s-сети, имеющей топологию *кольцо*, узлы соединены в кольцо (при этом через каждый узел проходят пакеты, направляемые из узлов, между которыми он расположен).

В s-сети с топологией решётка каждый узел способен самостоятельно взаимодействовать с любым другим узлом.

Топологию *шина*, *звезда* или *кольцо* обычно имеют локальные s-сети (LANs).

Другие топологии применяются в s-сетях со значительной территориальной распределённостью (WANs).

/ Проблемы и тенденции развития

На современном этапе развития s-сетей необходимость разработок новых и совершенствования существующих сетевых технологий определена интенсивным ростом разнообразных сетевых *сервисов* (см. с.117) (образовательных, торговых, развлекательных и др.), реализация которых требует значительного увеличения пропускной способности, надёжности и информационной безопасности s-сетей. При этом непрерывно растёт число сетевых узлов, представленных мобильными *s-машинами* (ноутбуками, коммуникаторами и др.), что требует поиска новых и развития действующих беспроводных сетевых технологий.

Неубывающую актуальность сохраняет комплекс проблем информационной безопасности сетевых технологий [включая защиту от несанкционированного доступа к информационным ресурсам, программным и аппаратным средствам, от вредоносных воздействий (компьютерных вирусов [Ильин В.Д. 2006, 2], спама и др.). Часть проблем информационной безопасности и эффективности применения сетевых технологий связана с недостаточно квалифицированным администрированием многих s-сетей и неосмотрительным поведением сетевых пользователей [Ильин В.Д. 2009, 5].

Б.4. ИНТЕРНЕТ



□ **ИНТЕРНЕТ** (англ. Internet: INTERconnected NETworks) – всемирная сеть *s-машин* (см. с.87), множество узлов которой составляют взаимодействующие по единым правилам *s-машины*, работающие в составе независимых пакетных *s-сетей* с различными архитектурами, техническими характеристиками и территориальным размещением. Правила обмена *сообщениями* между узлами Интернета определены семейством протоколов TCP/IP (сокр. англ. Transmission Control Protocol / IP – протокол управления передачей / Интернет протокол) и его последующими расширениями и продолжениями.

Обозначение в *s-моделировании* — *Internet*.□

Являясь основанием *s-среды* (см. с.93), И. служит коммуникационной инфраструктурой электронной почты, *Веба* (см. с.121) [Ильин В.Д. 2006, 1], поисковых систем, IP-телефонии и др. *s-сервисов*, широко применяемых при *информатизации* (см. с.130) образования, науки, экономики, гос. управления и др. видов деятельности.

Интернет – существенный фактор глобализации, содействующий добровольному объединению людей разных стран [Ильин В.Д. 2008, 4].

Интернет является одним из выдающихся достижений науки и техники 20 века.

/ Основы построения и функционирования

Присоединение сети или отдельных s-машин к Интернету не ограничивает их функциональные возможности. Для соединяемых Интернетом сетей нет ограничений по масштабу, местам размещения, характеристикам сетевой среды передачи (проводной или беспроводной), применяемым программно-аппаратным средствам и др.

// Для подключения к Интернету и нормального функционирования в его составе должны выполняться следующие требования.

1. Сети, соединяемые Интернетом, должны иметь стандартные архитектуры пакетных сетей (англ. packet-switching network).
2. Программы s-машин, обменивающиеся сообщениями в Интернете, должны поддерживать семейство протоколов TCP/IP, его последующие расширения и продолжения.
3. Проектировщики, интернет-провайдеры и др. должны строго следовать открытым протоколам и процедурам, определённым действующими интернет-стандартами.

// Пакетная коммутация

По технологии пакетной коммутации отправляемые сообщения (веб-страницы [Ильин В.Д. 2006, 1], электронные письма и др.) автоматически разбиваются на части определённого формата, называемые пакетами. Каждый пакет имеет заголовок с адресами узла-получателя и узла-отправителя.

При обмене сообщениями между узлами различных сетей пакеты одного и того же сообщения могут доставляться по разным маршрутам. Выбор маршрутов и доставку пакетов по адресам осуществляют специальные s-машины –

IP-маршрутизаторы. При обнаружении повреждённых или недостающих пакетов с узла-получателя автоматически посылается запрос на повторную передачу требуемых пакетов.

Когда все пакеты получены и признаны правильными, сообщение автоматически собирается и становится доступным прикладной программе, работающей с сообщениями данного типа (✱браузеру, если сообщение – веб-страница [Ильин В.Д. 2006, 1]✱).

Для межсетевых соединений в Интернете применяются проводные и беспроводные (радио, в т.ч. спутниковые) каналы связи, а межсетевое взаимодействие при обмене сообщениями обеспечивается с помощью IP-маршрутизаторов и др. оборудования.

// Правила функционирования: TCP/IP

Правила функционирования Интернета определены десятками протоколов, входящих в семейство TCP/IP, которое получило свое название по именам основных протоколов [TCP (транспортировки сообщений) и IP (обработки адресов и выбора маршрутов доставки пакетов)].

В соответствии с протоколом IP каждый узел Интернета получает двоичный IP-адрес (Internet Protocol Address), принадлежащий пространству глобальных уникальных адресов. По версии IPv4 протокола IP длина IP-адреса составляет 32 бита, а по версии IPv6 протокола IP следующего поколения – 128. Максимальное число IP-адресов на каждом этапе развития Интернета определяется действующей версией IP-протокола: версия IPv4 поддерживает 2^{32} (4294967296) IP-адресов, а IPv6 – 2^{128} .

С ростом популярности Интернета растёт потребность в IP-адресах, поскольку сервисы (см. с.117) Интернета используются с применением стационарных и мобильных

s-машин (компьютеров, коммуникаторов, различных компьютерных устройств, встроенных в транспортные средства, бытовую технику и др.).

Адреса IPv4 обычно записывают в виде четырёх десятичных чисел, разделённых точками (*193.27.41.108*); адреса IPv6 – в виде восьми шестнадцатеричных чисел, разделённых двоеточиями (*1080:0:0:0:8:800:200C:417A*) или в др. виде.

При записи адресов ресурсов, размещённых на веб-серверах и серверах других сервисов Интернета, предусмотрена возможность вместо числовых адресов применять буквенно-цифровые имена узлов Интернета. Такие имена строятся по правилам системы доменных имен (DNS – Domain Name System), представляющей собой распределённый механизм отображения доменных имен в IP-адреса.

Применение TCP/IP обеспечивает достаточно эффективную передачу сообщений между узлами Интернета, как угодно удалёнными друг от друга. Качество функционирования Интернета существенно зависит от производительности IP-маршрутизаторов, пропускной способности (измеряемой бит/сек) соединяющих их каналов связи и производительности др. оборудования, применяемого для осуществления межсетевого взаимодействия.

/ Разработка и реализация интернет-стандартов

Нормальное функционирование и развитие Интернета возможно при условии, что проектировщики, интернет-провайдеры и др. строго следуют открытым протоколам и процедурам, определённым действующими интернет-стандартами.

Их разработкой занимается открытое международное сообщество профессионалов IETF (Internet Engineering Task Force –

Объединение групп интернет-проектирования). IETF на добровольных началах объединяет проектировщиков, исследователей, интернет-провайдеров и др. IETF входит в международную некоммерческую профессиональную организацию ISOC (The Internet Society – Общество Интернета). Под управлением ISOC работает Internet Architecture Board (IAB) (Совет по Интернет-архитектуре), координирующий работы по развитию архитектуры Интернета (включая разработки, направленные на совершенствование стека TCP/IP). IAB является конечной инстанцией при определении новых интернет-стандартов.

/ Подключение к Интернету

Чтобы пользоваться сервисами Интернет, необходимо на договорной основе подключиться к сети одного из *интернет-провайдеров* (Internet Service Provider).

Среди предоставляемых видов доступа наиболее распространены доступ по выделенному каналу (кабельному или радио), спутниковый доступ и доступ по коммутируемому телефонному каналу с использованием модема (теряет применимость). Пользователи, имеющие доступ в Интернет, выбирают, когда и какие сервисы Интернета использовать, а провайдеры – когда и какие предоставлять.

/ Создание и развитие

Идея построения сети s-машин, подобной современному Интернету, была предложена в 1962 Дж. Ликлайдером (США) в заметках, посвящённых концепции построения т.н. «галактической сети» («Galactic Network»).

Теоретическое обоснование пакетной коммутации при передаче сообщений в компьютерных сетях было дано в статье Л.Клейнрока (США), опубликованной в 1961.

Первый практический шаг по пути создания Интернета был сделан Оборонным департаментом передовых исследовательских проектов ARPA (США), который в 1967 представил план построения пакетной сети ARPANET. В 1969 четыре компьютера были соединены в первоначальную конфигурацию ARPANET, ставшую зародышем Интернета. В последующие годы число узлов ARPANET быстро росло путём подключения компьютеров различных гос. организаций и университетов США.

В 1971 Р. Томлинсон (США) изобрёл электронную почту, а в 1972 этот первый из наиболее популярных сервисов Интернета был реализован.

Ключевым событием в истории развития Интернета стало изобретение Р. Каном (США) сетевого протокола, на базе которого в 1973 было разработано семейство протоколов TCP/IP, созданных Р. Каном совместно с В. Серфом (США). TCP/IP был стандартизирован в 1983.

В 1989 был реализован проект Веба, разработанный Т. Бернерс-Ли (Великобритания). Огромная популярность Веба и электронной почты определили стремительный рост Интернета.

Высокому темпу развития Интернета способствует то, что со времени его зарождения профессионалы, занимающиеся научно-техническими задачами построения Интернета, без задержек обмениваются идеями и решениями, используя возможности Интернета.

Созданные на основе Интернета *сервисы* сделали возможным существование разнообразных коммерческих и некоммерческих интернет-образований: интернет-магазинов, социальных сетей, поисковых систем, энциклопедических изданий (включая Энциклопедию информатики ИНФОПЕДИЯ), виртуальных

исследовательских лабораторий, корпоративных и гос. информационных порталов и др.

/ Перспективы и проблемы

С ростом числа узлов и сервисов Интернета возрастает актуальность связанных между собой проблем: пропускной способности каналов, пресечения распространения спама, вредоносных программ [Ильин В.Д. 2006, 2] и др. вредоносных воздействий на Интернет. На обострение проблемы пропускной способности существенно влияет рост популярности сервисов, основанных на технологиях обмена видео-сообщениями высокого разрешения в режиме реального времени; проведение видеоконференций с эффектом телеприсутствия и др. Попытки кардинального решения этих и ряда др. проблем предпринимаются в рамках начатых в США проектов «Интернет2» (Internet2, 1996) и «Интернет следующего поколения» (NGI – Next Generation Internet, 1997).

/ Правовые вопросы

Как средство глобального обмена сообщениями в s-среде Интернет не является чьей-то собственностью.

Однако объекты, связанные с Интернетом [оборудование инфраструктуры Интернета (кабели, маршрутизаторы и др.), доменные имена и сервисы Интернета, аппаратные и программные средства интернет-провайдеров и пользователей, электронные документы и др.], являются частной, корпоративной, гос. или общественной собственностью.

Перегрузка оборудования Интернета, аппаратных средств провайдеров и пользователей [✱массовой несанкционированной рассылкой почтовых сообщений (спама)✱]; несанкционированное влияние на работу аппаратных средств и программ, повреждение файлов документов (✱путём рассылки

вредоносных программ [Ильин В.Д. 2006, 2]✱), несанкционированное использование сервисов Интернета и электронных документов – всё это относится к нарушениям действующих в развитых странах законов.

Специфика задач нормативно-правового регулирования использования Интернета определяется открытостью Интернета и международным характером применения сервисов электронной коммерции, дистанционного обучения и др.

Среди актуальных юридических задач: охрана авторских и иных исключительных прав на объекты интеллектуальной собственности, связанные с Интернетом; определение юридического статуса различных типов электронных документов и сделок, совершаемых с использованием Интернета; нормативное регулирование электронной коммерции, электронных платежей и др. [Ильин В.Д. 2008, 4].

Б.5. S-ДОМЕН



□ *S-ДОМЕН* (англ. S-domain) – поименованная совокупность элементов некоторой системы (напр., IP-адресов узлов *Интернета* (см. с.106)), относящейся к *s-моделированию* (см. с.18), для которой определены единые правила управления и применения.

Специализация *домена (domain)* по параметру область применимости (*applicability*). Для информатики значение этого параметра – *s-моделирование (s-modeling)*:

$s\text{-domain} \approx \text{domain}[:,\text{applicability}=s\text{-modeling}]$.

Обозначение в *s-моделировании* — *s-domain*.□

/ S-домен Интернета (*inet-domain*)

S-домен Интернета – специализация *s-domain* по параметру *s-среда (s-environment)*: $\text{inet-domain} \approx s\text{-domain}[:,\text{s-environment} = \text{Internet}]$.

В *Интернете* осуществляется одновременный обмен *сообщениями* между миллионами его узлов (представленных разнообразными *s-машинами* (см. с.87): ноутбуками, коммуникаторами и др.), каждый из которых имеет уникальный IP-адрес [32-битовое число (для IPv4) или 128-битовое число (для Ipv6)].

/ Доставка сообщений

Осуществляется по IP-адресам, организованным в иерархическую древовидную систему со структурной единицей – inet-domain. Он включает в себя IP-адреса, которые имеют общую часть (т.е. у них совпадают либо одно, либо два, либо три первых числа). На вершине иерархического дерева находится inet-domain нулевого уровня (корневой inet-domain), включающий все inet-domain первого уровня, в каждый из которых могут входить один или несколько inet-domain второго уровня и т.д. К inet-domain первого уровня относятся inet-domain стран (напр., ru – домен России), коммерческих (com) или сетевых организаций (net), образовательных учреждений (edu) и ряд др.

/ Запись адресов

Для удобства записи адресов ресурсов (размещенных на веб-серверах и др.) пользователям Интернета предоставлена возможность вместо IP-адресов применять буквенно-цифровые имена (*доменные имена*). Доменное имя состоит из разделённых точкой имени узла и полного имени домена (состоящего, в свою очередь, из разделённых точками имен inet-domain, в которые входит этот узел). Последним в полном имени любого inet-domain стоит имя одного из inet-domain первого уровня.

☼ serv1.mydomain.ourdom.com – доменное имя узла serv1, входящего в inet-domain mydomain.ourdom.com третьего уровня, который входит в inet-domainourdom.com второго уровня, принадлежащий inet-domain com первого уровня.☼

/ Перевод доменных имен в IP-адреса

Выполняют DNS-серверы глобально распределенной иерархической системы преобразования доменных имен (Domain Name System), которые направляют IP-адреса узлов-получателей узлам, отправляющим сообщения.

DNS-серверы функционируют, используя распределенную базу данных с обновляемыми записями соответствия IP-адресов и доменных имен. В процессе перевода доменных имён серверы, обслуживающие корневой *inet-domain* взаимодействуют с серверами, обслуживающими *inet-domain* первого уровня и т.д. по дереву иерархии.

Распределением пространства IP-адресов и регистрацией *inet-domain* первого уровня занимается международная некоммерческая корпорация *ICANN* (англ. Internet Corporation for Assigned Names and Numbers). *Inet-domain* второго уровня регистрируют коммерческие компании, аккредитованные *ICCAN*. Владельцами *inet-domain* могут быть страны, юридические или физические лица.

/ S-домен локальной сети (*LAN-domain*)

LAN-domain – специализация *s-domain* по параметру *s-environment*: $LAN-domain \approx s-domain[::s-environment = LAN]$.

От *inet-domain* отличается методами построения и управления, а также назначением. Деление локальной сети на *LAN-domain* выполняет системный администратор, используя средства серверной операционной системы. При этом для каждого *LAN-domain* он определяет единую систему безопасности, права доступа пользователей к совместно используемым программам, файлам и др. Обычно деление локальной сети на *LAN-domain* производится по признаку общей деятельности пользователей *s-машин* *LAN-domain* (✱*LAN-domain* бухгалтерии, *LAN-domain* отдела кадров и т.п.✱). [Ильин В.Д. 2007, 1]

Б.6. S-СЕРВИС



□ *S-СЕРВИС* (англ. Online Service) – комплекс услуг определённого назначения (электронная почта, поиск и др.), реализованный в *s-среде*. Обозначение в *s-моделировании* — *s-service*.□

S-сервисы – воплощение *информатизации* (см. с.130) «под ключ» различных видов деятельности [образовательной (дистанционное образование и др.), экономической (НЭк-модель [Ильин В.Д. 2009, 2], [Ильин В.Д. 2009, 3] и *s-моделирование* имущественного обмена [Ильин В.Д. 2008, 1]).

Пользователь *s-сервиса*, заплатив сервис-провайдеру, получает интересующие его услуги, освободившись от необходимости разбираться в аппаратных и программных средствах *s-машин* (см. с.87), заботиться об апгрейдах и апдейтах.

☼ Как в своё время были введены стандарты на резьбовые соединения, то же происходит и во всех составляющих обеспечения, производства и потребления *s-сервисов*.☼

Б.7. ГИПЕРТЕКСТ



□ ГИПЕРТЕКСТ (англ. Hypertext) —

1. совокупность реализованных в *s-среде* (см. с.93) электронных документов, связанных между собой гиперссылками (для быстрого перехода от одного документа в заданное место другого и произвольных перемещений внутри документов);

2. технология построения совокупностей связанных гиперссылками документов, являющаяся технологическим основанием *Веба* (см. с.121) (применяется при разработке веб-сайтов [Ильин В.Д. 2006, 1], электронных энциклопедий, словарей, справочных систем и др).□

Принцип автоматизации работ с совокупностью связанных между собой документов сформулировал в 1945 В. Буш. Термин *гипертекст* ввёл в 1965 амер. Т. Нелсон, который вместе с Э. Ван Даму реализовал в 1967 первую гипертекстовую систему Hypertext Editing System.

Тип гипертекстовый *s-символ* — это специализация типа графический *s-символ* по параметру связь между документами (*link*): *hypertext symbol* \approx *graphic symbol* [*:link=hyperlink*].

Гипертекстовые документы представляют собой сообщения, которые обычно изготавливают с помощью специальных редакторов (☼ при подготовке статьи [Ильин В.Д. 2007, 2] её автор использовал MS Office FrontPage 2003☼). Такие редакторы имеют

встроенные интерпретаторы языков разметки (HTML и др.). Язык разметки позволяет описать тип документа, его структуру и составляющие (текст, изображения и др.), задать гиперссылки и др. Быстрому распространению гипертекстовых (см. с.118) технологий способствовала разработка языка SGML (Standard Generalized Markup Language – стандартный обобщенный язык разметки), который в 1986 был принят Международной организацией по стандартизации (ISO – International Organization for Standardization).

Важным событием в развитии гипертекстовых технологий стало создание британ. физиком Т. Бернерс-Ли [изобретателем Веба] языка HTML (англ. HyperText Markup Language – язык разметки гипертекстовых документов) на базе SGML. В 1994 был опубликован первый стандарт HTML 2.0, утверждённый IETF (Internet Engineering Task Force). В 1990-е этот язык стал основным средством разработки веб-сайтов.

По мере развития Веба в гипертекстовые документы стали включать не только текстовые, но и графические, аудио- и видео составляющие. Такие документы и технологии их разработки получили название гипермедийных (от англ. hypermedia). Первые гипермедийные системы были реализованы в США: ZOG (1975); The Aspen Myvie Map (виртуальное путешествие по г. Аспен) на видеодиске (1978). Позднее гипермедийные технологии обогатились и методами реализации гиперссылок внутри анимационных и видео составляющих.

Гипермедийные технологии располагают мощным инструментарием решения задач s-представления различных сообщений. Они широко применяются при разработке веб-сайтов, виртуальных лабораторий, средств дистанционного обучения и др. Развитие гипермедийных технологий предполагает создание новых языков и систем разработки

документов, совершенствование аппаратных средств s-машин (см. с.87), используемых для воспроизведения гипермедийных документов в формах, рассчитанных на человека [Ильин В.Д. 2007, 2].

Б.8. ВЕБ



□ **ВЕБ** (Всемирная паутина) [англ. Web (World Wide Web)] – распределенная неоднородная *s-машинная* (см. с.87) система коллективного пользования гипермедийными документами, действующая на базе сети *Интернет* (см. с.106). Один из наиболее популярных сервисов, реализованных в Интернете (наряду с электронной почтой, поисковыми системами и др.). Обозначение в *s-моделировании* — *Web*.□

Гипермедийные документы Веба, называемые веб-страницами, являются развитием гипертекстовых (см. с.118).

Они могут содержать текст, изображения, аудио, видео и др. составляющие. Каждая веб-страница имеет уникальный адрес – URL (сокр. Universal Resource Locator – унифицированный указатель ресурса), по которому её можно найти.

☼ www.webopedia.com – URL главной веб-страницы электронного компьютерного словаря.☼

Любая совокупность смежных символов веб-страницы может быть гиперссылкой на др. веб-страницу или др. ресурс Интернета. Связанные гиперссылками документы образуют сложную разветвлённую структуру, образно названную «паутиной» (web). Гиперссылки могут указывать не только на документы Веба, но и на другие *информационные ресурсы* (см.

с.133), предоставляемые сервисами (см. с.117), реализованными в Интернете.

□ Совокупность связанных гиперссылками веб-страниц, которые имеют общую часть URL (и, как правило, объединены тематически), называют веб-сайтом. □

☼ Сайт энциклопедии информатики ИНФОПЕДИЯ имеет адрес <http://infopedia2008.wordpress.com/>, а веб-страница статьи ВЕБ — <http://infopedia2008.wordpress.com/2009/08/03/□веб-всемирная-паутина/>. Общей частью адреса всех страниц сайта является адрес сайта. ☼

Веб-сайты хранятся на веб-серверах (s-машинах со специальным программным обеспечением).

Веб-серверами могут быть s-машины разл. архитектуры (удовлетворяющие требованиям надёжности и производительности), они могут работать под управлением разных операционных систем и использовать разные серверные программы. Поэтому Веб является неоднородной s-машинной системой.

Для получения веб-страниц применяют браузер (программу веб-клиент).

С помощью браузера пользователь формирует и отправляет запросы на получение интересующих его информационных ресурсов. Браузер интерпретирует полученные от веб-серверов ответные сообщения и отображает результаты интерпретации на устройстве в составе s-машины пользователя (дисплее компьютера, коммуникатора или др.).

С ростом числа различных веб-приложений (веб-почты, поиска, различных текстовых и графических редакторов и др. программ) браузер стал платформой для веб-приложений. С ростом числа

веб-приложений браузер берёт на себя всё большую долю задач операционной системы (☼ см. [беседу с разработчиком браузера Google Chrome и основанной на нём Google Chrome OS](#)☼).

Интерпретацию сообщений, поступающих от браузеров, выполняют серверные программы, установленные на веб-серверах. Обмен сообщениями между веб-клиентами и веб-серверами реализуется по правилам (протоколам), соответствующим архитектуре клиент–сервер. В Вебе действует протокол передачи гипертекста — HTTP (Hypertext transfer protocol).

Глобальная распределённость Веба выражается в том, что веб-серверы могут находиться в любой точке мира, где есть возможность подключения к Интернету.

Изобретатель Веба Тим Бернерс-Ли (Tim Berners-Lee) в 1989 разработал и реализовал проект Веба [британский физик, в то время сотрудник Европейской организации ядерных исследований CERN (Женева, Швейцария)]. В наст. время изобретатель Веба является директором международного консорциума World Wide Web Consortium ([W3C](#)). Цель W3C – увеличение потенциала Веба (путём разработки более совершенных протоколов и технологий) и обеспечение его непрерывного устойчивого роста.

С кон. 1993 (после создания первого браузера Mosaic с графическим интерфейсом) начался быстрый рост популярности Веба. К нач. 21в. Веб стал самым популярным и наиболее динамично развивающимся сервисом Интернета. Простота взаимодействия с Вебом и тематич. разнообразие веб-ресурсов определили масштаб его применения в различных областях человеческой деятельности (дистанционное обучение,

электронная торговля, электронные публикации, виртуальные научные эксперименты и др.) [Ильин В.Д. 2006, 1].

Б.9. О ПРОБЛЕМЕ S-ЗАЩИТЫ



Размер и сложность этой проблемы тем больше, чем неудачнее архитектурные решения, реализованные строителями s-машин и s-среды.

Следствием коммерческих гонок производителей аппаратных и программных средств s-среды является не только её динамичное, но и почти стихийное развитие. Устаревшие архитектурные решения и новые решения, не приведённые в соответствие требованиям развития s-среды, постоянно увеличивают сложность задач s-защиты.

Известно, что наиболее тяжёлые последствия имеют уязвимости, родившиеся на стадии проектирования. А проектирование, не основанное на стандартах, неизбежно служит источником уязвимостей. Несанкционированный доступ к документам, аппаратным и программным средствам, спам и вредоносные программы [Ильин В.Д. 2006, 2] — широко известные составляющие «зла», пристроившегося к «благу» быстро прирастающих полезных сервисов стихийно развивающейся s-среды.

Далее коснёмся только одной из «популярных» составляющих упомянутого «зла» (вредоносных программ).

/ Вредоносные программы

□ **ВРЕДОНОСНАЯ ПРОГРАММА** (англ. Malware) – программа, созданная для выполнения воздействий, нарушающих

нормальное функционирование *s-машин* (без ведома их владельцев). Обозначение в *s-моделировании* — *malware*.□

Вредоносные программы условно делят на группы, среди которых наиболее известны *сетевые черви* (англ. network worms), *классические вирусы* (англ. classic viruses) и *трояны* (англ. trojan programs).

Чаще всего вредоносные программы внедряют в машину-жертву (англ. victim machine) под видом прикладных программ или компонентов операционной системы. Как правило, они попадают в *s-машину* в составе *файлов*, вложенных в *сообщения* электронной почты, в составе *веб-страниц*, полученных с инфицированных веб-сайтов, и с файлами, доставленными по локальной *сети* (см. с.99) или скопированными с внешних носителей (флэш-накопителей и др.).

Наиболее применяемые способы распространения вредоносных программ: вставка в исполняемый файл какой-либо программы; создание копии исполняемого файла программы с добавленной вредоносной программой. В инфицированной *s-машине* вредоносные программы могут находиться в файловых системах, загрузочных секторах, в различных макросах и скриптах.

Классический вирус может размножаться, присоединяться к файлам, удалять или повреждать системные файлы, расходовать системные ресурсы (объём основной памяти, время процессора).

Наиболее опасными считаются сетевые черви, которые способны самостоятельно распространяться по компьютерной сети путём саморассылки сообщений со своей копией.

Попав в *s-машину*, червь рассылает свои копии по адресам электронной почты, которые находит в адресной книге. В каждой *s-машине*, куда отправлено сообщение с копией червя,

происходит то же самое. Лавинообразно нарастающий процесс рассылок червивых сообщений катастрофически перегружает почтовые серверы, снижает полезную пропускную способность сети. Кроме того, черви могут обеспечить возможность удалённого управления машиной-жертвой.

Трояны попадают в s-машину при установке пользователем какой-либо (по описанию будто бы полезной) программы.

Трояны способны выполнять различные действия, повреждающие программы и данные, а также собирать конфиденциальные сведения и затем передавать их по сети своим хозяевам (выступившим в роли дарителей троянских коней). Трояны могут использовать машину-жертву для рассылки спама или включить её в число участвующих в атаке какого-нибудь веб-сайта.

Нередко создаются сетевые черви с функциями и троянов, и классических вирусов.

Способы проникновения вредоносных программ в машины-жертвы спроектированы в расчёте на распространённые нарушения правил осмотровительного пользования.

Вредоносная программа создаётся с учётом результатов изучения уязвимостей в архитектуре, программных и аппаратных средствах s-машин, рассматриваемых как потенциальные объекты инфицирования.

Своевременное обновление операционной системы, прикладных программ и драйверов позволяет уменьшить уязвимость s-машины.

Средства защиты от вредоносных программ включают: программно или программно-аппаратно реализованные сетевые

экраны (называемые также брандмауэрами; от англ. firewalls); программные средства для обнаружения и удаления вредоносных программ [антивирусные (англ. antivirus software), антишпионские (англ. antispyware) и др.] и др.
[Ильин В.Д. 2006, 2].

Б.10. О МЕТОДОЛОГИИ ИНФОРМАТИЗАЦИИ



Информатизация (см. с.130) как способ совершенствования различных видов деятельности — целевой этап, на котором апробируются результаты, полученные в *s-моделировании* и построении *s-среды*.

Повышение продуктивности информатизируемой системы достижимо, если проект информатизации [Ильин В.Д. 2008, 3] реализован на основе апробированного методологического обеспечения.

Широко распространившаяся практика «компьютеризации и интернетизации» чаще всего приводит к бесполезным затратам. Мало того: сотрудники отвлекаются на игры, поиск товаров, пребывание в социальных сетях и прочие «шалости». Без ясной цели информатизации, развёрнутой в детализирующие комплексы задач, без профессионального изучения объекта информатизации всё неизбежно заканчивается «компьютеризацией и интернетизацией».

У «компьютеризации и интернетизации» есть трудно устранимая пара типовых причин:

1. руководители информатизируемых систем, как правило, слабо ориентируются во всём, что связано с информатизацией;

2. те, кто «информатизирует», пользуясь слабой подготовкой заказчика, делают не то, что «надо бы», а то, что выгодно.

Б.10.1. Информатизация

□ **ИНФОРМАТИЗАЦИЯ** (англ. Informatization) – способ повышения продуктивности организационно-технической системы путём применения взаимоувязанных технологий электронного документооборота, кооперативного формирования и использования информационных ресурсов, решения комплексов задач управления, учёта, проектирования и др.

Обозначение в s-моделировании — *informatization*. □

Объектами информатизации служат образовательные учреждения, научные и проектные организации, производственные и торговые предприятия, органы государственного управления и др.

Информатизация, как правило, сопровождается совершенствованием организационно-технической структуры объекта.

Информатизации предшествует изучение существующей на объекте технологии информационного взаимодействия, состава задач, методов и средств, применяемых для их решения.

Выделяются те задачи, решение которых целесообразно выполнять с помощью s-машин (см. с.87).

В зависимости от целей и финансовых возможностей выбирают разные подходы к информатизации разл. по масштабу организационных систем (от индивидуальных и малых предприятий до систем регионального и гос. управления).

Наиболее распространён т.н. унифицированный подход, который опирается на применение типовых проектных решений, реализуемых относительно просто и в довольно короткие сроки.

Другой подход предполагает разработку индивидуального проекта информатизации и ориентирован на качественное улучшение решения основных задач системы, являющейся объектом информатизации [Ильин В.Д. 1996].

На практике наибольшее распространение получили унифицированные проектные решения первого типа, реализованные на основе локальных сетей, рассчитанных на совместное использование данных, программ и аппаратных средств *s-машин* (принтеров, факсов и др.). Узлами таких сетей обычно служат: почтовый сервер, сервер приложений или др. (состав и назначение серверов зависят от решаемых задач), а также персональные компьютеры и ноутбуки руководителей и сотрудников. Как правило, часть *s-машин* такой сети подключена к *Интернету* (см. с.106). На рабочих местах используются пакеты прикладных программ для работы с документами (включая электронные таблицы), а также электронной почтой, поисковыми системами и др. сервисами Интернета. В зависимости от вида деятельности решаются задачи автоматизированного проектирования, бухгалтерского учёта, проектирования бюджета и др.

К числу проектов второго типа относятся проекты информатизации органов государственного управления и др. Уникальность проектов информатизации механизмов государственного управления объясняется различиями в системах государственного управления разных стран, нормативно-правовых оснований для реализации управляющих воздействий и др.

/ Информатизация систем ситуационного управления

[Ильин В.Д. 2005]

Сложные конкурирующие организационно-технические системы (крупные корпорации, системы государственного управления) связаны изменяющимися во времени отношениями подчинения и координации [Ильин В.Д. 1996]. Они функционируют в условиях, когда недостаточно известны не только планы и действия конкурентов, но и изменяющиеся характеристики собственных систем. Попытки расширить методический арсенал решения задач управления такими системами предпринимаются тем активнее, чем производительнее становятся средства s-среды.

Методология ситуационного управления такими системами и их информатизации, предложенная в [Ильин В.Д. 1996], представлена концепцией [Ильин В.Д. 1995], постановками и методами решения задач формирования портретов ситуаций (на основании наблюдений пространств состояний системы), проектирования ресурсно-обоснованных управляющих воздействий, принятия решений и контроля их исполнения. Управляющие воздействия проектируются так, чтобы при заданных ресурсных ограничениях сделать возможным перемещение из отправной ситуации в заданную окрестность спроектированной целевой ситуации.

На рубеже 20-го и 21-го веков информатизация (см. с.130) приобрела решающее значение для развития образования, науки, экономики и др. Современные тенденции в информатизации характеризуются повышением роли беспроводных сетевых технологий взаимодействия и мобильных s-машин (ноутбуков, коммуникаторов и др. перемещаемых устройств) [Ильин В.Д. 2008, 3].

Б.10.2. Информационные ресурсы

□ *ИНФОРМАЦИОННЫЕ РЕСУРСЫ* (англ. Information Resources) – *сообщения*, представленные в форме, рассчитанной на хранение (☼электронные и бумажные документы, фильмы и др.☼), накопление и интерпретацию пользователями, деятельность которых связана с построением и применением знаний (в науке, образовании, экономике и др.).

Обозначение в *s-моделировании* — *information resources*.□

Информационные ресурсы служат источником *информации* (см. с.55) и предполагают их неоднократное использование без участия тех, кто их создал. Являясь средством сохранения *знаний*, информационные ресурсы служат для построения новых знаний, которые, в свою очередь, могут быть представлены в форме информационных ресурсов.

Информационные ресурсы существуют в форме *s-сообщений* (☼гипермедийные документы *Веба* (см. с.121), *файлы* электронных документов☼) и иных формах (☼каталогизированные тексты и изображения на бумажных носителях, кинофильмы на киноплёнке и др.☼).

Использование информационных ресурсов в форме *s-сообщений* предполагает возможность их интерпретации программами *s-машин*.

Информационные ресурсы хранятся в гос. и др. архивах документов, библиотеках (в т.ч. электронных), на веб-серверах и др. Каждый информационный ресурс сопровождается спецификацией, содержащей *данные*, необходимые для его поиска, определения области применимости и др.

В сер. 20 века в библиотеках подобные спецификации были представлены в виде карточек, которые хранились в каталожных

ящиках. В совр. библиотеках, архивах и др., как правило, применяют электронные каталоги (где каждый информационный ресурс имеет электронную спецификацию) и поисковые системы.

Электронные информационные ресурсы используются с помощью различных s-машин (персональных компьютеров, ноутбуков и др.). Компактность накопителей s-машин, возможность хранить на них большие объёмы разнотипных s-сообщений, быстродействие s-машин и удобство работы с ними определили стремление перевести в электронную форму разл. информационные ресурсы (сканированием бумажных документов и книг, видеосъёмкой с помощью цифровых фото- и видеокамер и др.).

Электронные информационные ресурсы (электронные энциклопедии, электронные книги и др.) интенсивно вытесняют традиционные (неэлектронные).

В кон. 20 – нач. 21 века информационные ресурсы по значимости не уступают трудовым, энергетическим и др. ресурсам государств, и рассматриваются как национальное достояние [Ильин В.Д. 2008, 2].

Б.10.3. О методологии ситуационной информатизации

В методологии ситуационной информатизации государственного управления [Ильин В.Д. 1996] была предложена s-модель системы государственного управления (как объекта информатизации). В соответствие процессам выработки и исполнения государственных решений были поставлены комплексы задач, предложен подход к проектированию государственных решений, формализованному описанию и анализу государственных ситуаций.

Разработанный тогда же и программно реализованный комплекс решения задач проектирования ресурсно-обоснованных решений позднее был методически усовершенствован в [Ильин А.В., Ильин В.Д. 2004].



На примере этой методологии рассмотрим общезначимые вещи, которые могут быть полезны при разработке методологий информатизации [Ильин В.Д. 2008, 3] других организационно-технических систем.

/ Характеристика понятийных систем методологии

Для чего нужна методология? Если отвечать на этот вопрос кратко, то — для прокладывания научно обоснованного пути от намерений к реализации.

Когда сформировались намерения усовершенствовать процесс информационного обслуживания государственного механизма,

необходимо было отразить их содержание в концепции, отражающей замысел ситуационной информатизации и конструктивные принципы его реализации [Ильин В.Д. 1995]. Результаты обсуждения концепции стали основанием для продолжения работы над методологией ситуационной информатизации.

Предстояло конкретизировать положения концепции, построив понятийный каркас, который бы связал воедино основные (в терминах государственного управления) понятия концепции и понятия информатики и математики, используемые для формализации описаний. Без такой связи никакие намерения что-то реализовать в s-среде не имеют оснований.

Отсутствие понятийной связи (выраженной в форме спецификаций) между теми, кто работает на объекте информатизации (и станет пользователями) и разработчиками проекта информатизации приводит нередко к появлению и даже продолжительному существованию бесполезных систем. Обычно такие системы играют роль «ненужной мебели, оправдывающей прогрессивность ее приобретателей».

// Система основных понятий

К числу основных отнесены введенные в [Ильин В.Д. 1996] понятия:

- *госпространства,*
- *государственной ситуации,*
- *портрета государственной ситуации,*
- *государственного управляющего воздействия,*
- *средств государственного управляющего воздействия,*

- государственных ресурсов,
- объекта государственного управляющего воздействия,
- ресурсно-обоснованного государственного решения.



// Система реализационных понятий

Эта понятийная система выполняет роль моста между системой основных понятий и системой формализующих понятий (формализующей системой), включающей понятия информатики и математики.

К системе реализационных понятий относятся такие понятия, как *прототип государственного воздействия, шаблон описания ситуации, правила анализа ситуации, специализация правила, обобщение правила* и ряд других.

// Система целевых понятий

Она представляет собой понятийную систему, построенную на основе *системы основных понятий* и *системы реализационных понятий*. В системе целевых понятий (в целевой системе) определена совокупность связей на множестве понятий, являющемся объединением множеств основных и реализационных понятий.

Целевая система понятий представляет собой формализуемую систему понятий¹⁰.

10 Служит предметом формализации.

// Формализующая система

Нужна для того, чтобы:

- компактно и однозначно описать системы основных, реализационных и целевых понятий;
- разработать спецификации s-задач (для программного воплощения системы реализационных понятий).

/// Требования к формализации

Необходимо представить синтаксис и семантику средств описания состояния системы государственного управления, анализа ситуаций и выработки решений. Это нужно для восприятия этих средств заказчиками проекта информатизации и — для последующей программной реализации.

Разработка метода решения любой задачи, рассчитанного на реализацию в s-среде, предполагает его описание в составе спецификации s-задачи, удовлетворяющей определенным требованиям.

Даже если сложность задачи такова, что алгоритм её решения можно сразу записать на одном из языков программирования, то и в этом случае спецификация задачи не теряет своего значения. Она создает благоприятные условия для обмена профессиональными знаниями, облегчая восприятие метода другими специалистами данной предметной области. И самое важное: спецификация задачи выполняет роль «моста для анализа соответствия» между постановкой и программным воплощением задачи.

ЛИТЕРАТУРА

[Ильин А.В. 2008, 1]

Ильин А.В. *Интерактивный режим*. Большая Российская Энциклопедия, т. 11, 2008, с.435

[Ильин А.В. 2008, 2]

Ильин А.В. *Интерфейс в информатике*. Большая Российская Энциклопедия, т. 11, 2008, с.457-458

[Ильин А.В. 2007]

Ильин А.В. *Конструирование разрешающих структур на задачных графах системы знаний о программируемых задачах* // Информационные технологии и вычислительные системы, №3, 2007, с.30-36

[Ильин А.В., Ильин В.Д. 2009]

Ильин А.В., Ильин В.Д. *TSM – комплекс средств формализации гипермедийных описаний s-моделей*. Энциклопедия информатики ИНФОПЕДИЯ, 28.07.2009
<http://infopedia2008.wordpress.com/2009/07/28/tsm-комплекс-средств-формализации-текст/>

[Ильин А.В., Ильин В.Д. 2005]

Ильин А.В., Ильин В.Д. *Распределение ресурсов по обязательным и ориентирующим правилам: сравнительная эффективность алгоритмов. Системы и средства информатики*, вып. 15. М: Наука, 2005, с.123-159

[Ильин А.В., Ильин В.Д. 2004]

Ильин А.В., Ильин В.Д. *Интерактивный преобразователь ресурсов с изменяемыми правилами поведения //Информационные технологии и вычислительные системы*, №2, 2004, с.67-77

[Ильин В.Д. 2009, 1]

Ильин В.Д. *СИНФ: система знаний информатики. Энциклопедия информатики ИНФОПЕДИЯ*. 25.08.2009
<http://infopedia2008.wordpress.com/2009/08/25/синф-система-знаний-информатики/>

[Ильин В.Д. 2009, 2]

Ильин В.Д. *Модель нормализованной экономики*. М.: ИПИ РАН, 2009, 125с.

[Ильин В.Д. 2009, 3]

Ильин В.Д. *Модель нормализованной экономики (НЭК-модель): основы концепции // Управление большими системами*, 2009, вып. 25, с.116-138

[Ильин В.Д. 2009, 4]

Ильин В.Д. *Компьютерное моделирование*. Большая Российская энциклопедия, т. 14, 2009, с.712

[Ильин В.Д. 2009, 5]

Ильин В.Д. *Компьютерная сеть*. Большая Российская энциклопедия, т. 14, 2009, с.711-712

[Ильин В.Д. 2008, 1]

Ильин В.Д. *S-моделирование имущественного обмена*. М.: ИПИ РАН, 2008, 80с.

[Ильин В.Д. 2008, 2]

Ильин В.Д. *Информационные ресурсы*. Большая Российская Энциклопедия, т. 11, 2008, с.492-493

[Ильин В.Д. 2008, 3]

Ильин В.Д. *Информатизация*. Большая Российская Энциклопедия, т. 11, 2008, с.480-481

[Ильин В.Д. 2008, 4]

Ильин В.Д. *Интернет*. Большая Российская Энциклопедия, т. 11, 2008, с.451-452

[Ильин В.Д. 2007, 1]

Ильин В.Д. *Домен в информатике*. Большая Российская Энциклопедия, т. 9, 2007, с.232-233

[Ильин В.Д. 2007, 2]

Ильин В.Д. *Гипертекст*. Большая Российская Энциклопедия, т. 7, 2007, с.159

[Ильин В.Д. 2006, 1]

Ильин В.Д. *Всемирная паутина*. Большая Российская энциклопедия, т. 6, 2006, с.45

[Ильин В.Д. 2006, 2]

Ильин В.Д. *Вирус в информатике*. Большая Российская энциклопедия, т. 5, 2006, с.374

[Ильин В.Д. 2005]

Ильин В.Д. *Информатизация ситуационного управления*. Учебное пособие. М.: МИРЭА, 2005, 104с.

[Ильин В.Д. 1996]

Ильин В.Д. *Основания ситуационной информатизации*. М.: Наука, 1996, 180с.

[Ильин В.Д. 1995]

Ильин В.Д. *Концепция ситуационной информатизации государственного управления России*. М.: ИПИ РАН, 1995, 37с.

[Ильин В.Д. 1989]

Ильин В.Д. Система порождения программ. М.: Наука, 1989, 264с.

[Ильин В.Д., Соколов И.А. 2008]

Ильин В.Д., Соколов И.А. *Символьное моделирование в человеко-машинной среде: основы концепции / Информационные технологии и вычислительные системы*, 2008, №1, с.51-60

[Ильин В.Д., Соколов И.А. 2007]

Ильин В.Д., Соколов И.А. *Символьная модель системы знаний информатики в человеко-автоматной среде* //Информатика и её применения, т. 1, вып. 1, 2007, с.66-82

[Ильин В.Д., Соколов И.А. 2006]

Ильин В.Д., Соколов И.А. *Информация как результат интерпретации сообщений на символьных моделях систем понятий* //Информационные технологии и вычислительные системы, №4, 2006, с.74-82

[Ilyin V.D. 1995]

Ilyin V.D. *A Metodology for Knowledge Based Engineering of Parallel Program Systems* //Proc. Of the Eghth Int. Conf. "Industrial and Engineering Applications of Artifisial Intelligence and Expert Systems", Melbourne, Australia, June 6-8, 1995. p.805-809

Научное издание

Ильин Александр Владимирович, Ильин Владимир Дмитриевич

ОСНОВЫ ТЕОРИИ S-МОДЕЛИРОВАНИЯ

Электронная книга изготовлена авторами

Утверждено к изданию

Институтом проблем информатики Российской академии наук

18.12.2009

www.ipiran.ru

119333, Москва, ул. Вавилова, д. 44, корп. 2