

Кафедра «Информационные технологии»

Введение в специальность

Курс лекций по дисциплине
«Введение в специальность»
для специальности направления
1-40 05 01-01 «Информационные системы и
технологии
(в проектировании и производстве)»

Автор-составитель

Е.Г. Стародубцев, доцент, канд. физ.-мат. наук

Лекции 6, 7

Искусственный интеллект

Понятие искусственного интеллекта.

Экспертные системы. Базы знаний.

Оболочки экспертных систем.

Языки логического программирования.

Нейронные сети.

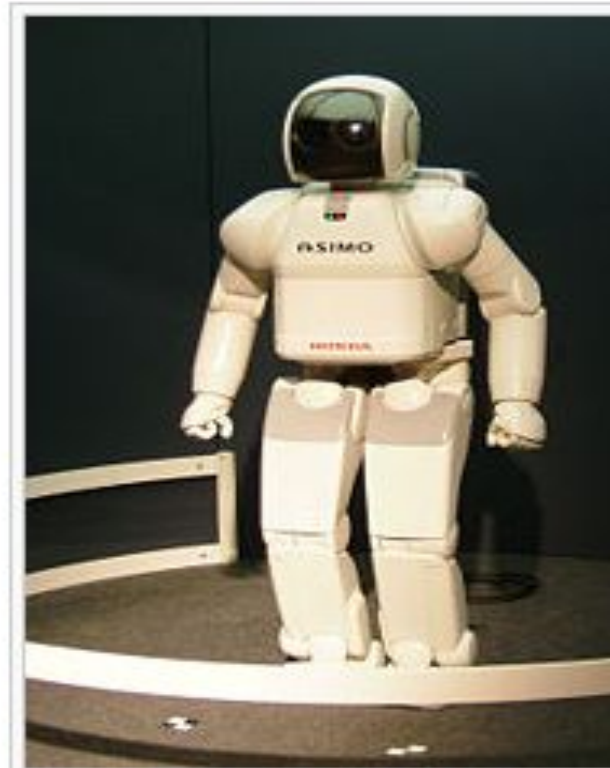
Распознавание образов.

Понятие искусственного интеллекта

Искусственный интеллект (англ. *Artificial intelligence, AI*) — раздел информатики, изучающий возможность обеспечения разумных рассуждений и действий с помощью вычислительных систем и иных искусственных устройств. При этом в большинстве случаев заранее неизвестен алгоритм решения задачи.

Согласия в определении того, что именно считать необходимыми и достаточными условиями достижения интеллектуальности не достигнуто. Существует точка зрения, которую, в частности, в явном виде разделяли **Норберт Винер**, **фон Нейман**, академики **Велихов** и **Раушенбах**, что искусственный интеллект это математическое моделирование всех функций высшей нервной деятельности (в СССР) или всех функций коры головного мозга (в Западных исследованиях) ^[1]. Другие специалисты принимают иные гипотезы, например, **тест Тьюринга** или **гипотеза Ньюэлла — Саймона**. Обычно к реализации интеллектуальных систем подходят именно с точки зрения моделирования человеческой интеллектуальности. Таким образом, в рамках искусственного интеллекта различают два основных направления:

- **символьное (семиотическое, нисходящее)** основано на моделировании высокоуровневых процессов мышления человека, на представлении и использовании знаний;
- **нейрокибернетическое (нейросетевое, восходящее)** основано на моделировании отдельных низкоуровневых структур мозга (**нейронов**).



ASIMO — Интеллектуальный гуманоидный робот от Honda

Понятие ИИ – другие определения

Восхищающие новые усилия заставить компьютеры думать... машины с разумом в полном и языковом смысле (J. Haugeland, 1955 г.)

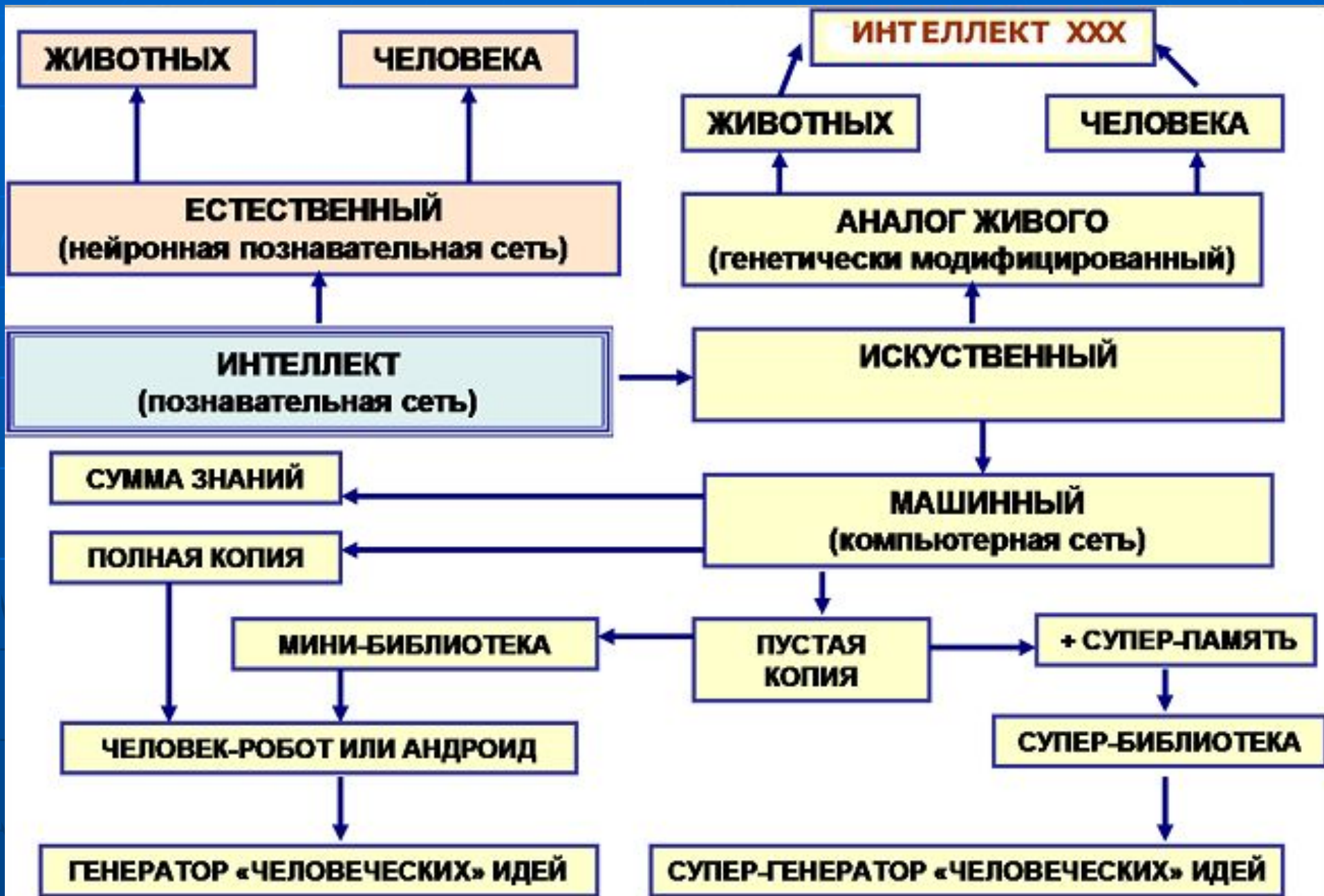
Автоматизация деятельности, которую мы ассоциируем с человеческим мышлением, такой как принятие решений, решение задач, усвоение знаний (R. E. Bellman, 1978 г.)

Исследование умственных способностей с помощью использования вычислительных моделей (E. Charniak, O. McDermot, 1985 г.)

Теория эвристического поиска и вопросы создания решателей задач, относящихся к разряду творческих или интеллектуальных (Г. С. Поспелов, 1986 г.)

Искусство создания машин, способных выполнять действия, которые потребовали бы интеллекта от людей, если бы они эти действия выполняли (R. Kurzweil, 1990 г.).

Возможные классификации систем ИИ



Основные классы задач, решаемые с помощью систем ИИ

Перед ИИ с момента его зарождения как научного направления (с середины 1950-х годов) ставились следующие трудно формализуемые задачи:

важные для задач робототехники:

доказательство теорем,
управление роботами,
распознавание изображений,
машинный перевод и понимание текстов на
естественном языке,
игровые программы,
машинное творчество (синтез музыки,
стихотворений, текстов);

**создание интеллектуальных обучающих
систем;**

создание экспертных систем;

• • •

Терминология

Слово «**робот**» появилось в 1920-х годах, как производное от чешского «робота» — тяжелой грязной работы. Его автор — чешский писатель Карел Чапек, описавший роботов в своем рассказе «R.U.R».

Термин «**искусственный интеллект**» — ИИ — (AI — **artificial intelligence**) был предложен в 1956 г. на семинаре в Дартмутском колледже (США). Семинар был посвящен разработке методов решения логических, а не вычислительных задач. В английском языке данное словосочетание не имеет той слегка фантастической антропоморфной окраски, которую оно приобрело в довольно неудачном русском переводе. Слово **intelligence** означает «умение рассуждать разумно», а вовсе не «интеллект», для которого есть термин **intellect**.

Терминология

Теоретические основы современной робототехники были заложены еще в 1950-60-е годы, но их реализация сдерживалась отсутствием технологий, материалов, ресурсов вычислительных систем.

В это же время писатель-фантаст Айзек Азимов предлагает термин **robotics** (**робототехника**) и впервые формулирует

Три закона робототехники

- Робот не может причинить вред человеку или своим бездействием допустить, чтобы человеку был причинен вред.
- Робот должен подчиняться командам человека, если эти команды не противоречат первому закону.
- Робот должен заботиться о своей безопасности, пока это не противоречит первому и второму закону.

Эти законы до сегодняшнего дня остаются стандартами при проектировании и разработке роботов.

«Поколения» роботов

Роботы - это электротехнические устройства для автоматизации человеческого труда.

Условно в истории робототехники выделяют:

I поколение. Роботы с жесткой схемой управления. Практически все современные промышленные роботы принадлежат к первому поколению. Фактически это программируемые манипуляторы.

II поколение. Адаптивные роботы с сенсорными устройствами. Есть образцы таких роботов, но в промышленности они пока используются мало.

«Поколения» роботов

III поколение. Самоорганизующиеся или интеллектуальные роботы. Это - конечная цель развития робототехники. Основные нерешенные проблемы при создании таких роботов - проблема машинного зрения и адекватного хранения и обработки 3D визуальной информации.

В настоящее время в мире изготавливается более **60 000 роботов в год.**

Фактически робототехника сегодня — это инженерная наука, не отвергающая технологий ИИ, но не готовая пока к их широкому внедрению по разным причинам.

Роботы Японии остановят радиацию вместе

22.02.2013, 17:11 » Телекоммуникации (мировые новости)

ntd.tv.ru

Японские производители роботов показали свои последние разработки. Главный акцент они сделали на способах остановить кризис на АЭС "Фукусима".



Вот это, например, специальный костюм, который позволяет без труда поднять предмет весом до 60 килограммов. Его разработала компания Cyberdyne. Изначально он был предназначен для помощи в передвижении быстро стареющему населению Японии. Но инцидент на "Фукусиме" открыл новые пути использования этого приспособления.

Масахиро Такама, разработчик: *"Костюм разработан для носки внутри АЭС, а также в других зонах, где произошло бедствие"*.

На костюме имеются пластины, которые защитят от гамма-излучения, а также мониторы, фиксирующие жизненные показатели человека. Внутри также имеется система кондиционирования воздуха, чтобы при работе внутри реактора, в условиях раскаленного воздуха, не получить тепловой удар.

Университет Тибы и компания Mitsubishi показали свое совместное творение: маленького и мобильного робота по имени "Сакура". При помощи своей руки он смог аккуратно открыть клапан на трубе с высоким давлением и предоставил видеообзор с разных сторон.

Такаюки Фурута, исследователь Университета Тибы: *"Это беспрецедентный случай, когда разработчики роботов сотрудничают вместе ради единой цели. Мы понимаем, что в этой ситуации материальная выгода - это не самое главное"*.

А вот робот-амфибия, разработанный совместно с Toshiba. Он может проводить поисковые и разведывательные операции внутри реакторов, заполненных водой.

В целом, здесь можно увидеть разработки девяти компаний, которые объединились в рамках проекта "Организация развития новой энергии и промышленных технологий".

Содзи Кукита, глава проекта: *"Японские исследователи не фокусировались на создании роботов, которые можно использовать в ядерных реакторах. Но авария на "Фукусиме" предоставила им возможность охватить эту новую область"*.

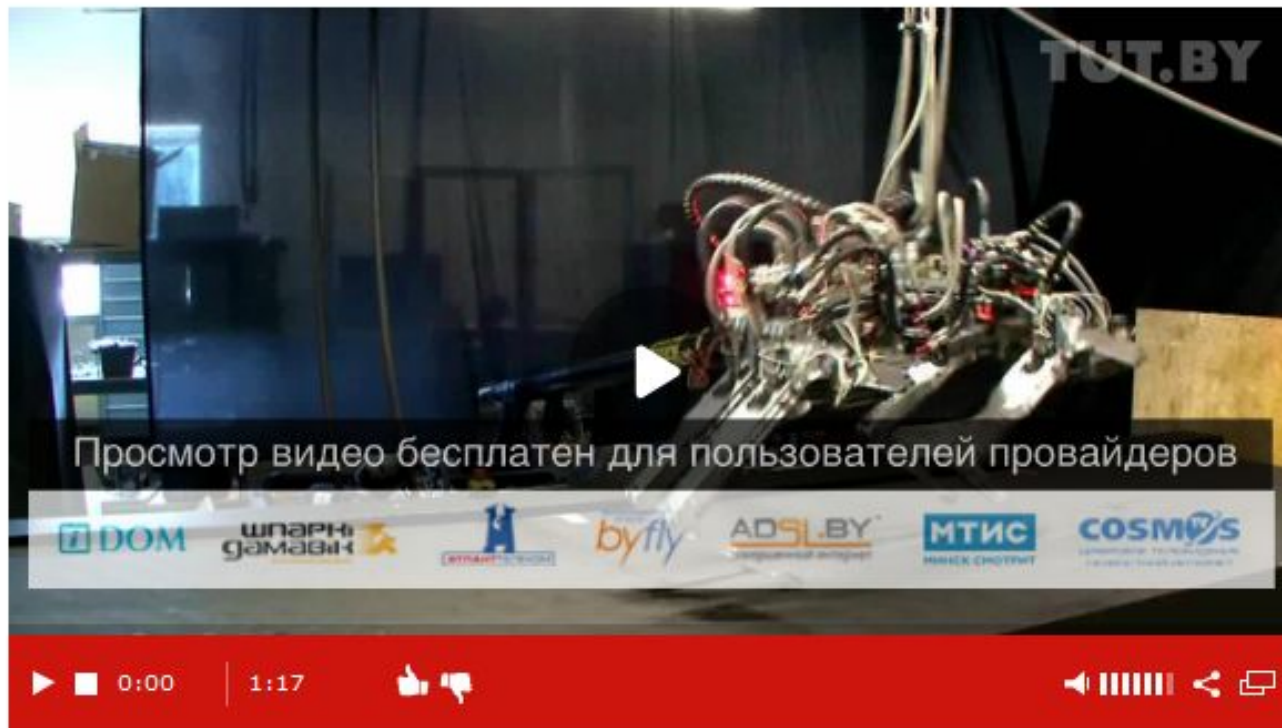
Главная цель исследователей - достичь того, чтобы в аварийных реакторах людям вообще работать не приходилось.

Робот впервые побежал быстрее человека

06.09.2012, 13:22 » Техника (мировые новости)

Компьюлента

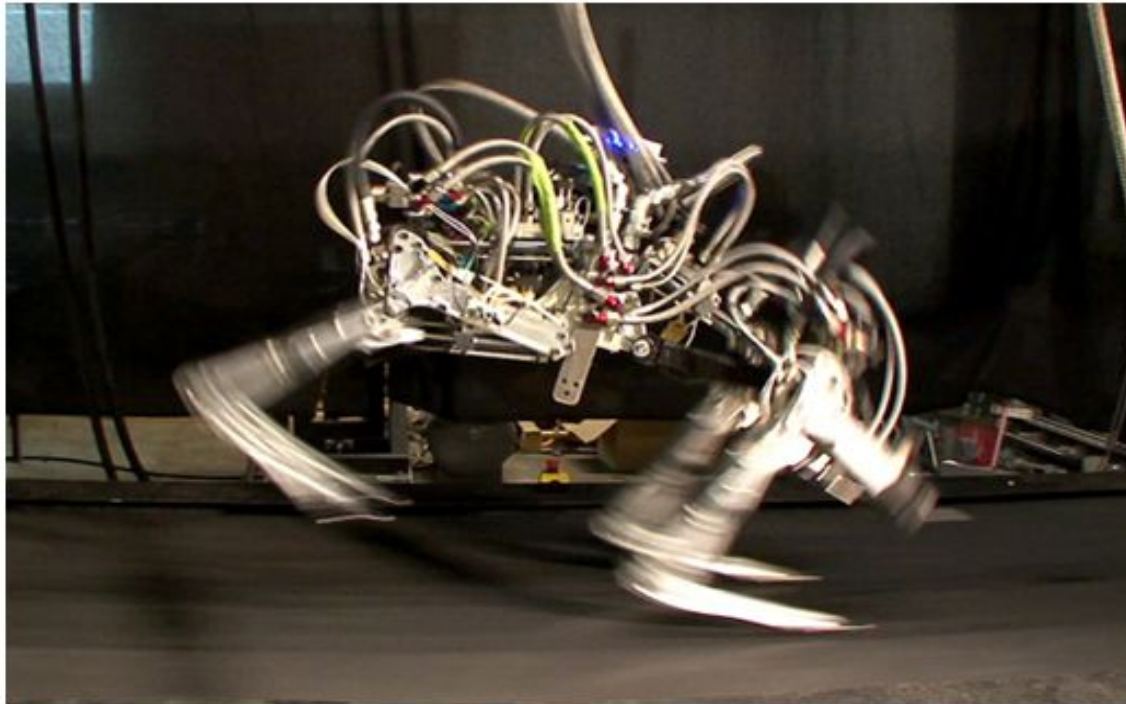
Шагающий робот Cheetah ("Гепард"), разрабатываемый американской компанией Boston Dynamics, впервые в истории робототехники достиг скорости в 45,5 км/ч, превысив тем самым лучший человеческий результат — рекорд Усэйна Сент-Лео Болта, установленный им в 2009 году. Человек-ракета с Ямайки, напомним, пронесся 100 м за 9,58 с (37,6 км/ч).



[Открыть/скачать видео \(25.76 МБ\)](#)

А ведь совсем недавно тот же робот показывал лишь 30 км/ч и на соперничество с г-ном Болтом, совсем не претендовал. Новый результат достигнут в основном за счет изменения алгоритмов работы "лап", сохранивших ту же конструкцию:

Однако все эти фанфары пока далеки от главной цели разработчиков. Они хотят бегать со скоростью 80 км/ч, причем не на привязи, а перемещаясь по открытой местности. И всё же продемонстрированное заставляет снять шляпу — скорость робогепарда выросла в полтора раза (с 30 до 45 км/ч) менее чем за полгода.



"Гепард" на привязи уже обскакал лучшего человека-бегуна. Даже на самых быстрых 20 м стометровки скорость Усэйна Болта была на 10% ниже, чем у Cheetah

Как объясняют конструкторы, их робот "учится летать" подобно молодым гепардам, которые без обучения (в зоопарке, например) бегают не очень хорошо, не быстрее 50 км/ч, притом что опытные "живые" животные разгоняются до 110 км/ч.

Автономный AlphaDog демонстрирует на испытаниях возможности боевых шагающих машин. Если робот будет столь же быстрым, как "Гепард", то на пересеченной местности ему не будет равных среди построенных людьми аппаратов.

Тест Тьюринга

Алан Тьюринг, 1950 г., статья
«Вычислительные машины и разум»;
проверка, разумен ли компьютер в
человеческом смысле слова.

Тьюринг предложил тест, для замены
бессмысленного, по его мнению, вопроса
«может ли машина мыслить?» на более
определенный.

Тест Тьюринга

Проведение теста

Судья (человек) переписывается на естественном языке с двумя «собеседниками»: человеком и компьютером. Если судья не может определить, кто есть кто, - компьютер прошел тест.

Каждый из «собеседников» стремится, чтобы человеком признали его. Для простоты и универсальности теста переписка сводится к обмену текстовыми сообщениями и выполняется через одинаковые промежутки времени, чтобы судья не мог делать выводы исходя из скорости ответов. (Во времена Тьюринга компьютеры реагировали медленнее человека. Сейчас это правило необходимо, т.к. они реагируют быстрее, чем человек.)

Тьюринг считал:

- к 2000 году компьютер с памятью 1 миллиард бит (119 МБ) в ходе 5-минутного теста «обманет» судей в 30 % случаев - не сбылось;
- обучение компьютеров будет играть важную роль в создании мощных компьютеров (сбылось).

Пока - ни одна программа и близко не подошла к прохождению теста. Такие программы, как Элиза (*ELIZA*), иногда заставляли людей верить, что они говорят с человеком (пример - неформальный эксперимент *AOLiza*).

Но такие «успехи» не являются прохождением теста Тьюринга по ряду причин («нечистый» эксперимент).

Ежегодно - соревнование между «разговаривающими» программами и наиболее человекоподобной, по мнению судей, присуждается приз Лёбнера.

Есть также приз для программы, прошедшей тест Тьюринга. Этот приз еще не присуждался.

Лучший результат - программа A.L.I.C.E., выиграла приз Лёбнера 3 раза (в 2000, 2001 и 2004).

Суперкомпьютер IBM Watson победил в телевикторине Jeopardy

16.01.2011 16:00 | просмотров: 100

Необычное шоу ожидает в ближайшее время американских телезрителей. В США идет передача Jeopardy, представляющая собой телевикторину, в которой игроки отвечают на самые разные вопросы по самым разным тематикам. Российским телезрителям такой же проект известен под названием "Своя игра". В отличие от других игр Jeopardy, когда за главный приз в миллион долларов соревнуются трое знатоков, здесь место третьего знатока занял не человек, а машина - суперкомпьютер IBM Watson, представляющий собой уменьшенный вариант суперкомпьютера Blue Gene.



Здесь оборудование разместились в 10 серверных закрытых шкафах, объединенных между собой высокоскоростными сетевыми соединениями, а внутри Watson разместился объем энциклопедической информации, эквивалентный 200 млн печатных страниц текста. Система Watson была названа в честь основателя корпорации IBM Томаса Уотсона.

Задача компьютера является с одной стороны на удивление простой, а с другой невероятно трудной. Перед инженерами IBM стояла задача создать компьютерную систему, способную на уровне человека отвечать на вопросы, изложенные на естественном языке, причем быстро, точно и достоверно. Формат викторины Jeopardy является для компьютера исключительно трудным, поскольку предлагаемые участникам подсказки вынуждают их анализировать тонкие смысловые оттенки, учитывать иронию, разгадывать загадки и преодолевать другие сложности, т.е. заниматься теми видами деятельности, которые люди традиционно выполняют лучше, чем компьютеры.

Соревновались с системой Watson два самых знаменитых игрока, когда-либо участвовавших в викторине Jeopardy. Кен Дженнингс побил рекорд Jeopardy по максимальному количеству сыгранных друг за другом игр - на протяжении сезонов 2004-2005 г.г. он выиграл 74 игры подряд, что обеспечило ему выигрыш более 2,5 млн. долл. Бред Руттер выиграл максимальное совокупное количество игр для одного участника Jeopardy, что принесло ему 3 255 102 долл. Эта сумма - результат первого появления Руттера в Jeopardy, состоявшееся в 2000 г., плюс три победы в турнирах «Jeopardy»: «Tournament of Champions» (2001 г.), «Million Dollar Masters Tournament» (2002 г.) и «Ultimate Tournament of Champions» (2005 г.).

«Большой приз» состязания составил 1 млн. долл., а призы за второе и третье места - 300 тыс. долл. и 200 тыс. долл., соответственно. Руттер и Дженнингс пожертвуют по 50% своего выигрыша на благотворительные цели, а корпорация IBM полностью жертвует свой выигрыш на благотворительные цели.

«После четырех лет работы наша научная группа уверена в том, что система Watson готова к этому состязанию. Она способна быстро осознавать, что означает подсказка «Jeopardy», анализировать доступную информацию и быстро находить точные и достоверные ответы, - говорит д-р Дэвид Ферруччи (David Ferrucci), ученый, возглавляющий группу из IBM Research, которая создала систему Watson».

Прошлой осенью система Watson провела более 50 спарринг-игр против прежних участников турнира «Tournament of Champ Jeopardy» на заключительном этапе подготовки к своему телевизионному дебюту. Кроме того, система Watson успешно прошла тот же тест для претендентов, который проходят люди при первичном отборе на участие в шоу Jeopardy.

На протяжении практически всего соревнования лидировала именно Watson, определяя тон, тематику и смысл вопросов, и давая на них ответы с исключительной скоростью. В рамках соревнования компьютер успешно справился практически со всеми вопросами, не уловив тематики лишь некоторых из них и пропустив других участников викторины вперед. По итогам викторины Watson опередил других участников.

Гипотеза Ньюэлла — Саймона

Эллен Ньюэлл и Герберт Саймон следующим образом сформулировали гипотезу физической символической системы (Physical Symbol System Hypothesis), 1976:

Физическая символическая система имеет необходимые и достаточные средства для того, чтобы производить осмысленные действия.

Т.е., без символических вычислений невозможно выполнять осмысленные действия, а способность выполнять символические вычисления вполне достаточна для выполнения осмысленных действий. Таким образом, если мы полагаем, что животное, человек, или машина действуют осмысленно, то они каким-то образом выполняют символические вычисления.

Независимо от справедливости гипотезы, символические вычисления стали реальными и полезными для разных областей программирования и обработки данных.

ИИ – некоторые основные понятия

Предмет информатики - обработка информации по известным законам.

Предмет ИИ - изучение интеллектуальной деятельности человека, подчиняющейся заранее неизвестным законам.

ИИ - все то, что не обрабатывается с помощью алгоритмических методов.

Система - множество элементов, находящихся в отношениях друг с другом и образующих причинно-следственную связь.

ИИ – некоторые основные понятия

Адаптивная система - система, сохраняющая работоспособность при непредвиденных изменениях свойств управляемого объекта, целей управления или окружающей среды путем смены алгоритма функционирования, программы поведения или поиска оптимальных (эффективных) решений и состояний.

Традиционно, по способу адаптации различают **самонастраивающиеся**, **самообучающиеся** и **самоорганизующиеся** системы.

Интеллектуальная система (ИС) - адаптивная система для создания программ целесообразной деятельности по решению поставленной задачи для конкретной ситуации, складывающейся на данный момент в окружающей среде.

Дополнение 1 к данному определению ИС:

К сфере решаемых ИС задач относятся задачи, обладающие следующими особенностями:

- неизвестен алгоритм решения задач (**интеллектуальные задачи**);
- используется помимо традиционных числовых данных информация в виде рисунков, знаков, букв, слов, звуков;
- наличие выбора (нет алгоритма => нужно выбрать между многими вариантами решения в условиях неопределенности); «свобода действий».

Интеллектуальная система (ИС) - адаптивная система для создания программ целесообразной деятельности по решению поставленной задачи для конкретной ситуации, складывающейся на данный момент в окружающей среде.

Дополнение 2 к данному определению ИС:
Интеллектуальные робототехнические системы (ИРС) содержат 2 модели:

- 1) переменную, настраиваемую модель внешнего мира;
- 2) модель реальной исполнительный системы с объектом управления.

Цель и управляющие воздействия формируются в ИРС на основе знаний о внешней среде, объекте управления и на основе моделирования ситуаций в ₂₆ реальной системе.

Понятие **знания** - центральное понятие в ИС

Несколько определений:

1) Знания - результат, полученный познанием окружающего мира и его объектов.

2) Знания - система суждений с принципиальной и единой организацией, основанная на объективной закономерности.

3) Знания - формализованная информация, на которую ссылаются или которую используют в процессе логического вывода.

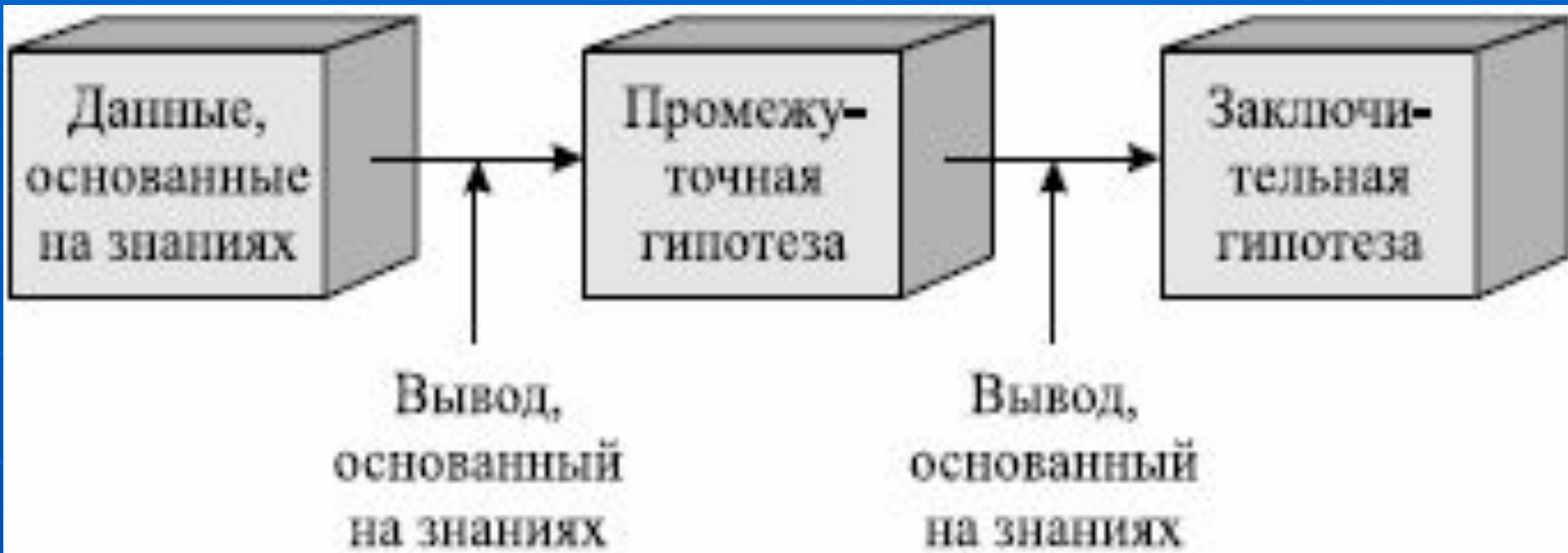
4) Знания - совокупность фактов и правил.

Правило, представляющее **фрагмент знаний**, имеет вид:

если <условие> то <действие>

Например: если выражения (условия) **X** и **Y** истинны, то выражение (условие) **Z** истинно с вероятностью **P**.

В ИС принято использовать определение 3 для определения знаний. Определение 4 есть частный случай определения 3.



Процесс логического вывода в ИС

Статические знания - знания, введенные в ИС на этапе проектирования.

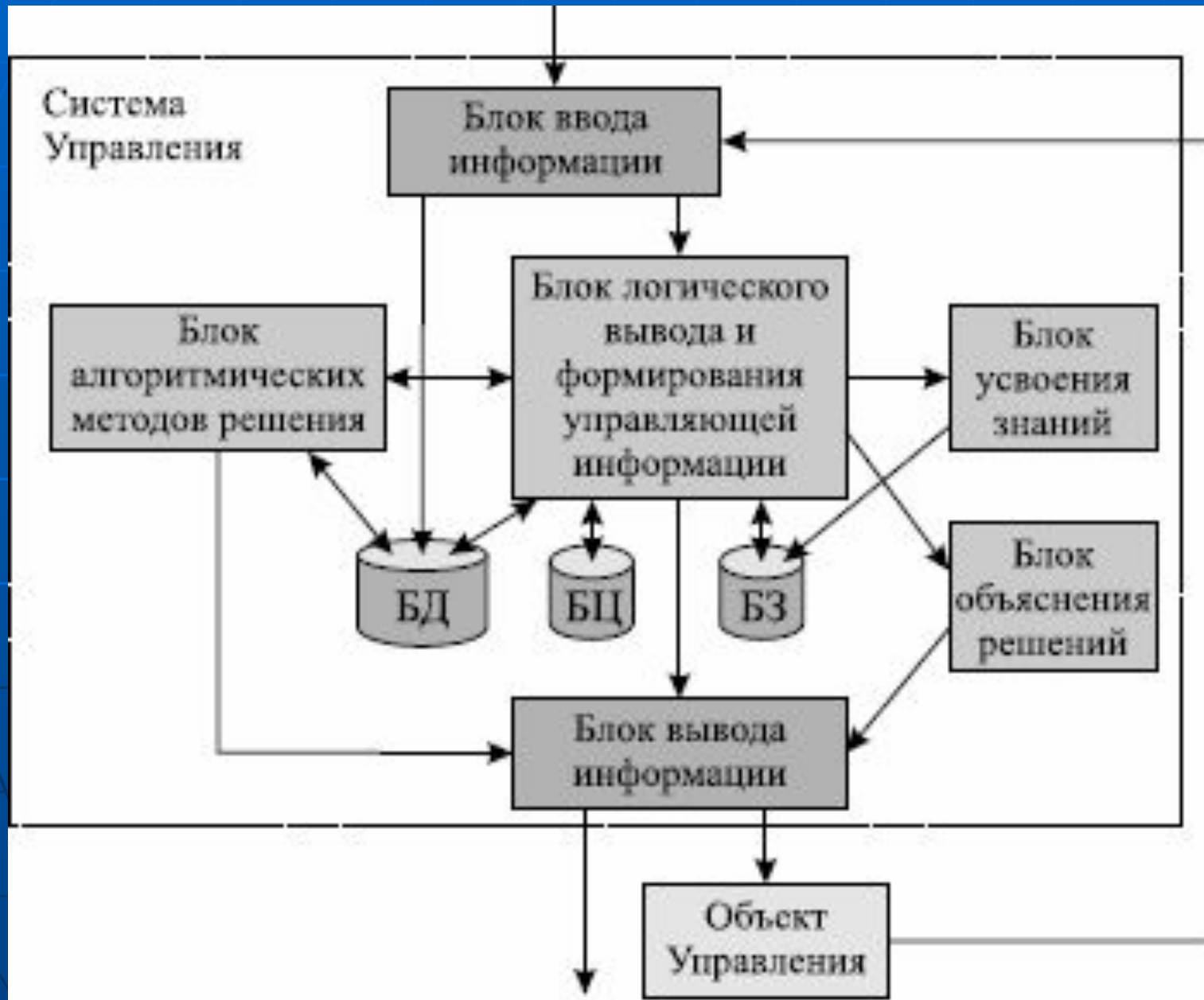
Динамические знания (опыт) - знания, полученные ИС в процессе функционирования или эксплуатации в реальном времени.

Знания делятся на **факты** и **правила**. Под фактами подразумеваются знания типа «А это А», они характерны для баз данных.

Под **правилами (продукциями)** понимаются знания вида «ЕСЛИ-ТО».

Кроме этих знаний существуют **метазнания** («знания о знаниях») – информация о том, как «устроены» правила, факты и соответствующие структуры данных.

Пример структуры ИС – обобщенная структура ИРС



БД –
база
данных

БЦ –
база
целей

БЗ –
база
знаний

ЭЛЕМЕНТЫ ИРС

Блок логического вывода и формирования управляющей информации – выполняет функции:

- решение нечетко формализованных задач;
- планирование действий;
- формирование управляющей информации для пользователя или объекта управления на основе Базы Знаний (БЗ), Базы данных (БД), Базы Целей (БЦ) и Блока Алгоритмических Методов Решений (БАМР).

БЗ - совокупность знаний (например, система продукционных правил) о закономерностях предметной области.

БЦ - множество локальных целей системы, представляющих собой совокупность знаний, активизированных в конкретный момент и в конкретной ситуации для достижения глобальной цели.

ЭЛЕМЕНТЫ ИРС

БАМР содержит программные модули решения задач предметной области по жестким алгоритмам.

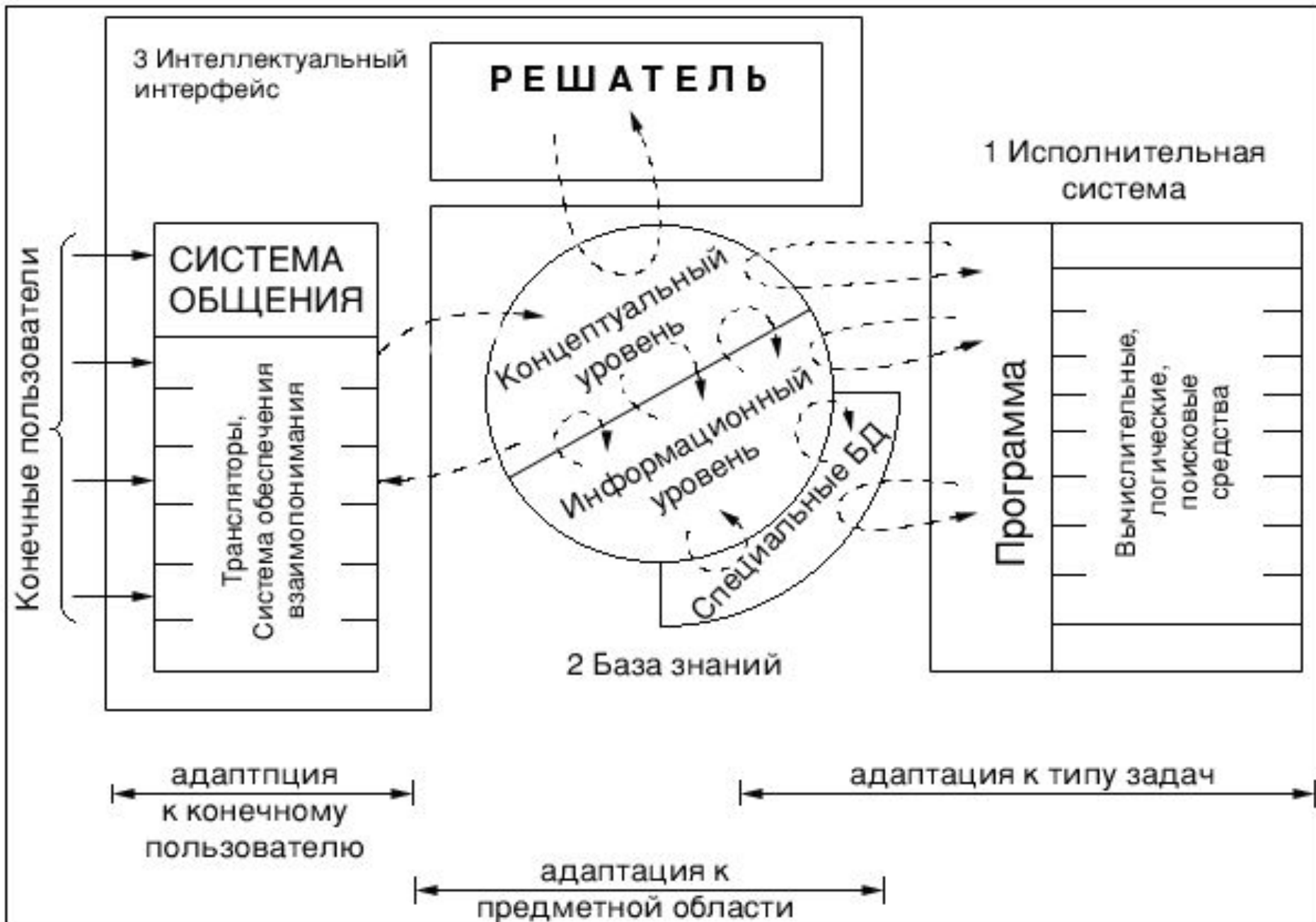
Блок усвоения знаний (БУЗ) анализирует динамические знания с целью их усвоения и сохранения в БЗ.

Блок объяснения решений (БОР) интерпретирует пользователю последовательность логического вывода, примененную для достижения текущего результата.

Блок вывода информации выводит данные, текст, речь, изображения и другие результаты логического вывода пользователю и/или **Объекту Управления (ОУ)**.

Контур обратной связи обеспечивает адаптивность и обучение ИС. На этапе проектирования ИС эксперты наполняют БЗ и БЦ, а программисты разрабатывают программы алгоритмических методов решений. БД создается и пополняется при эксплуатации ИС.

Схема структуры и работы систем ИИ



Подходы к изучению (разработке) ИИ

Сейчас - 4 основных достаточно различных подхода:

1. Логический подход

Основа - Булева алгебра, расширенная исчислением предикатов (введены предметные символы, отношения между ними, кванторы существования и всеобщности).

Особенности систем ИИ на логическом принципе:

- «Машина доказательства теорем» - исходные данные - аксиомы, правила вывода - отношения между аксиомами.
- Наличие блока генерации цели; система вывода пытается доказать данную цель как теорему. Если цель доказана, то трассировка примененных правил дает цепочку действий по реализации поставленной цели (пример таких систем - экспертные системы).
- Мощность системы определяется возможностями генератора целей и машиной доказательства теорем.

1. Логический подход к ИИ

Добиться большей выразительности логическому подходу позволяет сравнительно новое направление - нечеткая логика и теория нечётких множеств - раздел математики, являющийся обобщением классической логики и теории множеств. Понятие нечеткой логики было введено профессором Лотфи Заде в 1965 г.

Основное отличие - правдивость высказывания может принимать кроме **Да/Нет (1/0)** еще и промежуточные значения , например:

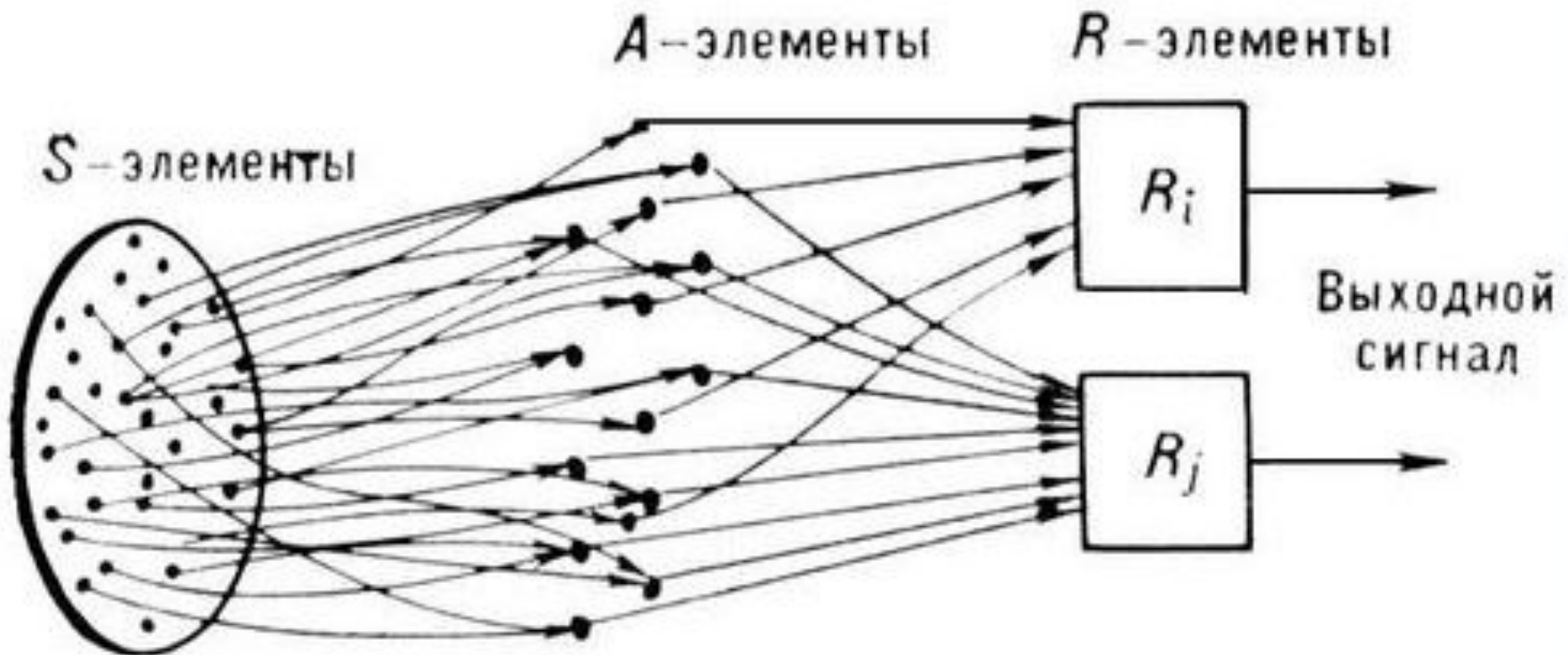
- не знаю (0.5),
- пациент скорее жив, чем мертв (0.75),
- пациент скорее мертв, чем жив (0.25).

Данный подход: больше похож на мышление человека, который на вопросы редко отвечает только Да или Нет; позволяет учитывать субъективные мнения отдельных экспертов при разработке экспертных систем.

2. Структурный подход к ИИ

Попытки построения ИИ путем моделирования структуры человеческого мозга.

Одна из первых таких попыток - **перцептрон** Френка Розенблатта (1958), модель электронного устройства, имитирующего процессы человеческого мышления.



Перцептрон должен был передавать сигналы от «глаза», составленного из фотоэлементов, в блоки электромеханических ячеек памяти, которые оценивали относительную величину электрических сигналов. Эти ячейки соединялись между собой случайным образом.

Два года спустя была продемонстрирована первая действующая машина («Марк-1»), которая могла «научиться» распознавать некоторые из букв, написанных на карточках, которые подносили к его «глазам», напоминая кинокамеры.

2. Структурный подход к ИИ

Основной моделируемой структурной единицей в перцептронах (и в большинстве других вариантов моделирования мозга) является **нейрон – аналог клетки мозга человека** (человеческий мозг – до 10^{11} нейронов).

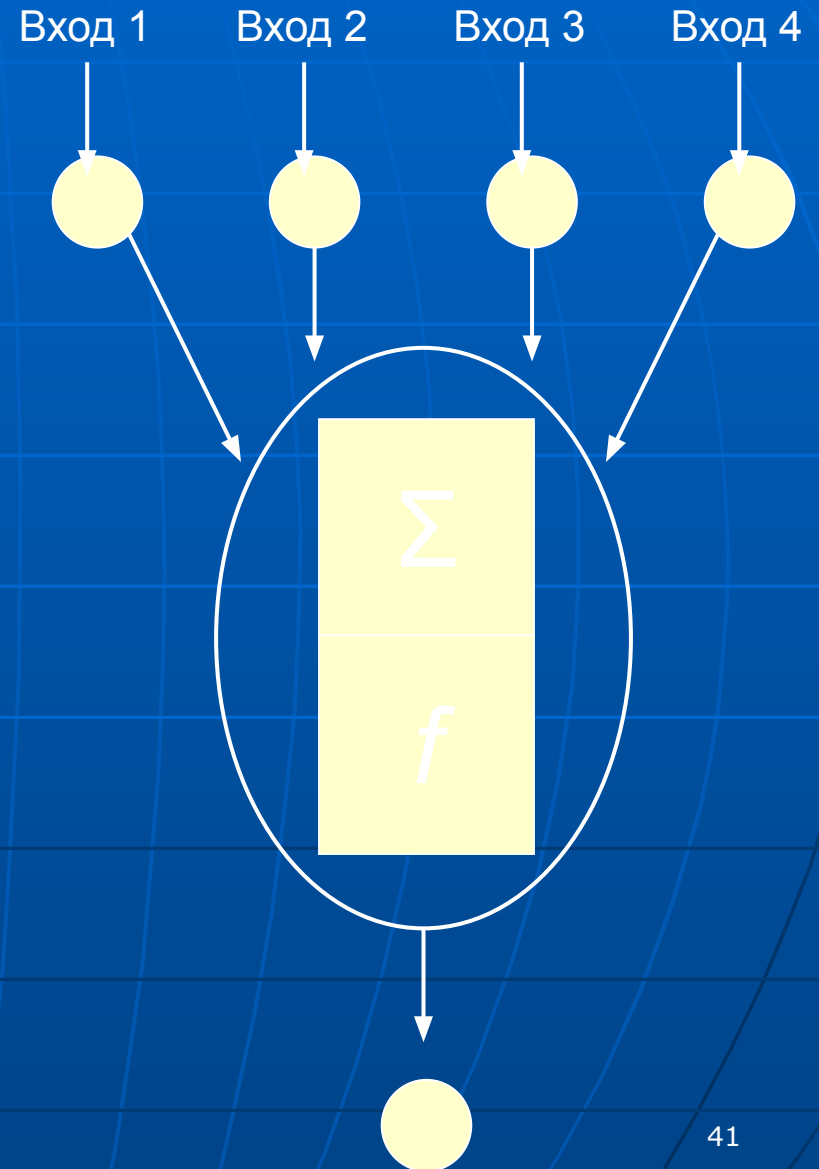
Позднее возникли и другие модели, известные под термином **нейронные сети (НС)**. Эти модели различаются по строению отдельных нейронов, по топологии связей между ними и по алгоритмам обучения.

2. Структурный подход к ИИ

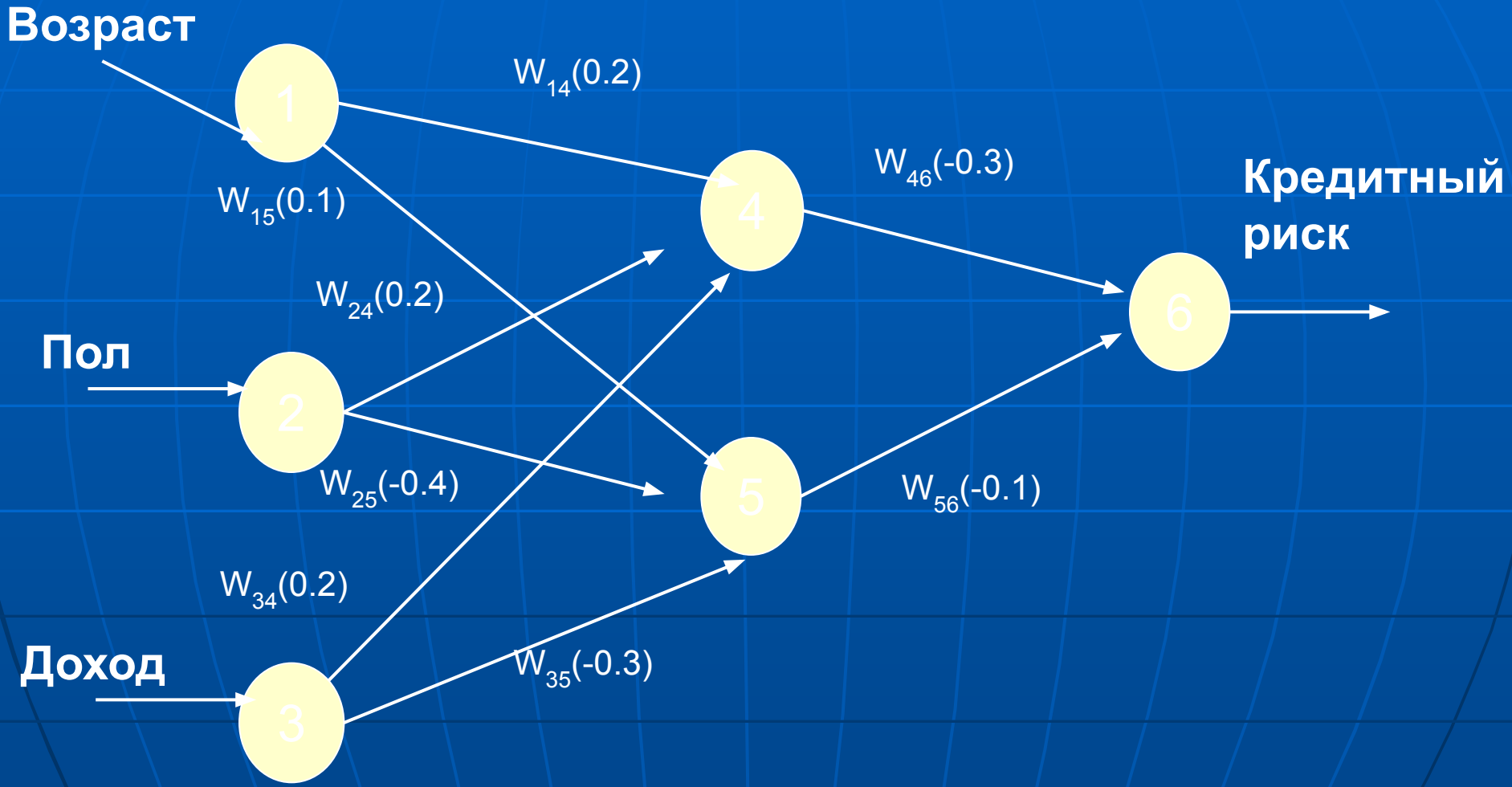
В более широком смысле такой подход известен как **коннективизм** - моделирование мыслительных или поведенческих явлений процессами становления в сетях из связанных между собой простых элементов.

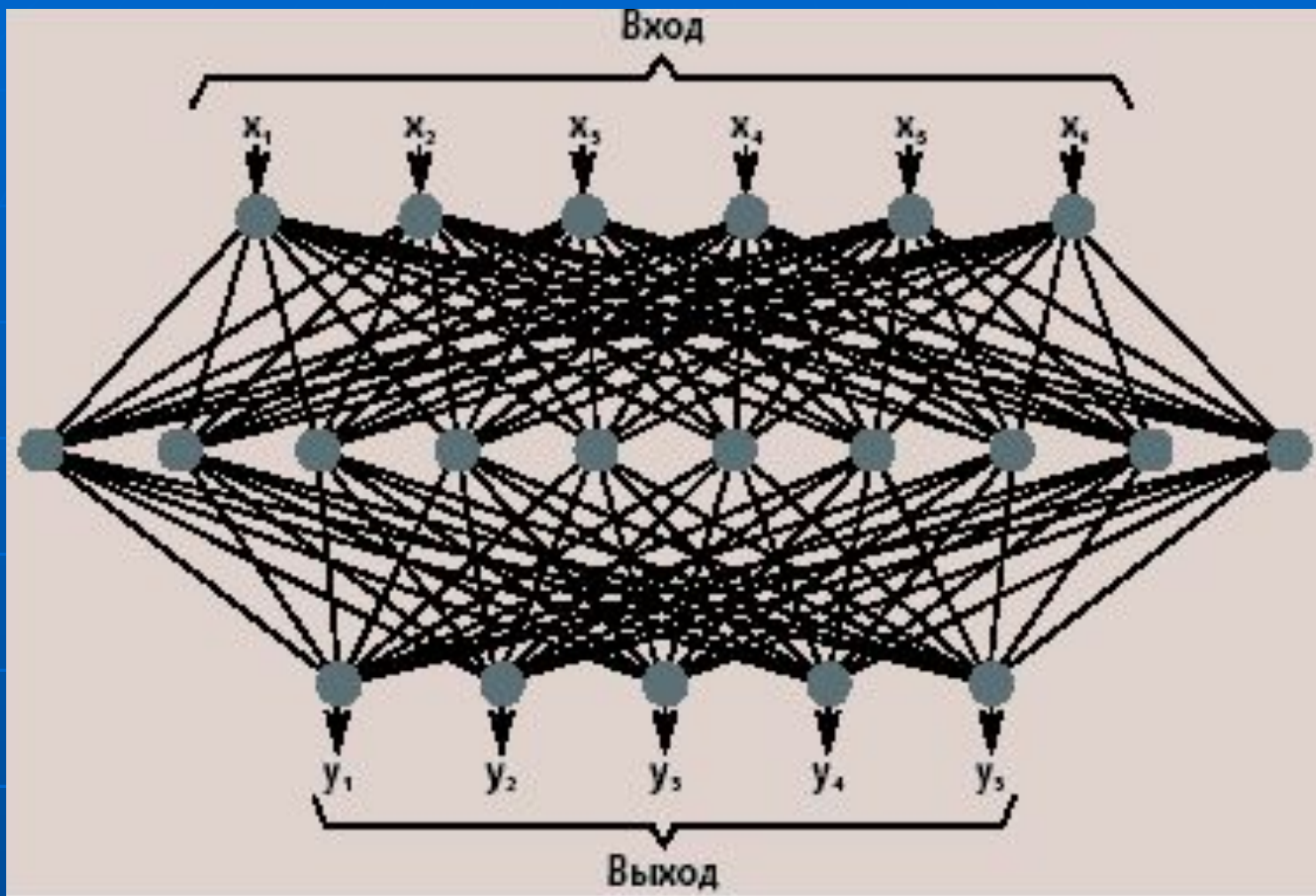
Существуют различные формы коннективизма, но наиболее общие используют НС модели.

Как устроен нейрон ?



Примеры «работы» нейронной сети

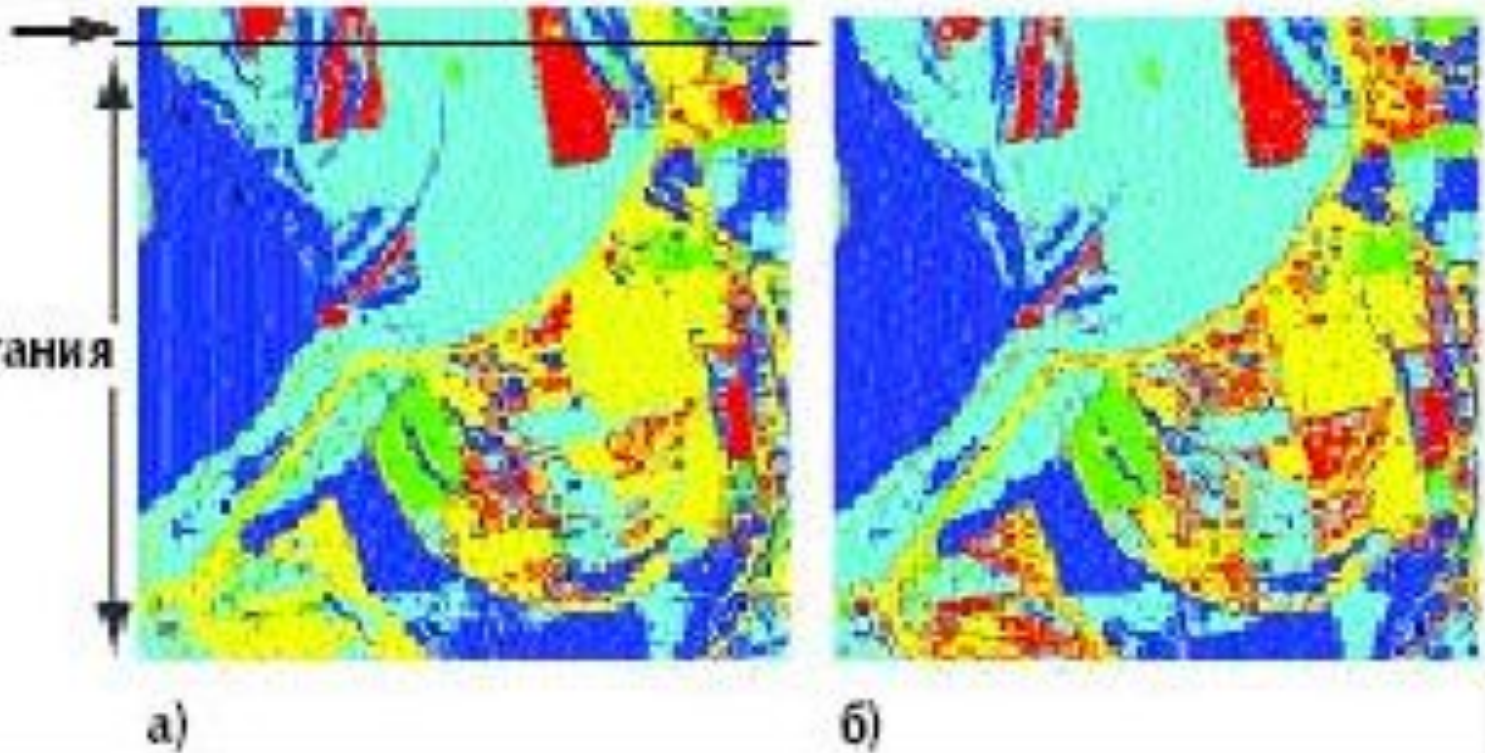




Многоуровневый персептрон. Трехуровневая сеть принимает входные данные со снимка, сделанного в 6 волновых диапазонах, и формирует на выходе изображения, распределяющие землю по 5 категориям землепользования.

Обучение и
испытания

Испытания



(а) Реальная карта землепользования и **(б)** карта землепользования, сформированная с помощью нейронной сети. Условными цветами обозначены: темно-синий — вода; голубой — болота; зеленый — деревья; красный — возделываемая почва; желтый — камни; черный — участки, назначение которых неизвестно.

3. Эволюционный подход к ИИ

При построении систем ИИ по данному подходу основное внимание уделяется построению начальной модели, и правилам, по которым она может изменяться (эволюционировать). Причем модель может быть составлена по самым различным методам, это может быть и НС, и набор логических правил, и любая другая модель.

После этого компьютер на основании проверки моделей отбирает самые лучшие из них, на основании которых по самым различным правилам генерируются новые модели. Среди эволюционных алгоритмов классическим считается генетический алгоритм.

3. Эволюционный подход к ИИ

Генетический алгоритм (Джон Холланд) – алгоритм поиска, используемый для решения задач оптимизации и моделирования путем последовательного подбора, комбинирования и вариации искомых параметров методами, напоминающими биологическую эволюцию.

Отличительной особенностью генетического алгоритма является акцент на использование оператора «скрещивания», который производит операцию рекомбинации решений-кандидатов, роль которой аналогична роли скрещивания в живой природе.

Генетический алгоритм

1) Задача кодируется так, чтобы её решение описывалось вектором («**хромосома**»).

2) Случайным образом создаются начальные вектора («**начальная популяция**»). Они оцениваются «**функцией приспособленности**» - каждому вектору присваивается определённая «приспособленность» (вероятность выживания «организма», представленного данным вектором).

3) По полученным приспособленностям выбираются вектора (**селекция**) для «**скрещивания**». К этим векторам применяются «генетические операторы» («**скрещивание**», «**мутация**»), создавая следующее «поколение».

4) Следующее поколение также оценивается, затем - селекция, применение генетических операторов и т.

Генетический алгоритм

Так моделируется «**эволюционный процесс**» в течение нескольких жизненных циклов (**поколений**), пока не будет выполнен критерий останова алгоритма, например, нахождение оптимального решения; исчерпание числа поколений или времени, отпущенных на эволюцию.

Схема работы генетического алгоритма



Начальная популяция формируется случайным образом, ее размер (количество особей N) фиксируется: $N = const$ в течение работы алгоритма. Каждая особь генерируется как случайная L -битная строка, где L — длина кодировки особи, она тоже фиксирована и для всех особей одинакова. Каждая особь - одно из решений задачи. Более приспособленные особи - это более подходящие ответы. Этим ГА отличается от др. алгоритмов оптимизации, которые оперируют лишь с одним решением, улучшая его.

Генетические алгоритмы служат для поиска решений в очень больших и сложных пространствах поиска.

Шаги генетического алгоритма:

- 1) Создание начальной популяции
- 2) Вычисление функций приспособленности для особей популяции (оценивание)
- 3) Выбор индивидов из текущей популяции (селекция)
- 4) Скрещивание и\или мутация
- 5) Вычисление функций приспособленности для всех особей
- 6) Формирование нового поколения

Если выполняются условия останова, то

(конец цикла), иначе (начало цикла)

4. Имитационный подход

Данный подход - классический для кибернетики с одним из ее базовых понятий **черный ящик**. Объект, поведение которого имитируется, представляет собой «черный ящик». Нам не важно, что у него и у модели внутри, и как он функционирует, главное, чтобы модель в аналогичных ситуациях вела себя точно так же.

Так моделируется другое свойство человека - способность копировать то, что делают другие, не вдаваясь в подробности, зачем это нужно. Зачастую эта способность экономит человеку массу времени, особенно в начале его жизни.

Методы построения экспертных систем

Экспертная система (ЭС) - программа, которая использует знания специалистов (экспертов) о некоторой предметной области и в пределах этой области способна принимать решения на уровне эксперта-профессионала.

ЭС (основанные на «знаниях» системы) - одно из наиболее значительных достижений ИИ.

При решении современных задач управления сложными системами, процессами нужно решать неформализуемые задачи.

Области использования экспертных систем

Области: авиация, космос, оборона, химия, энергетика, металлургия, транспорт, медицина, административное управление, прогнозирование и мониторинг,

Достижения: создание систем, которые ставят диагноз заболевания, предсказывают месторождения полезных ископаемых, помогают в проектировании электронных устройств, машин и механизмов,

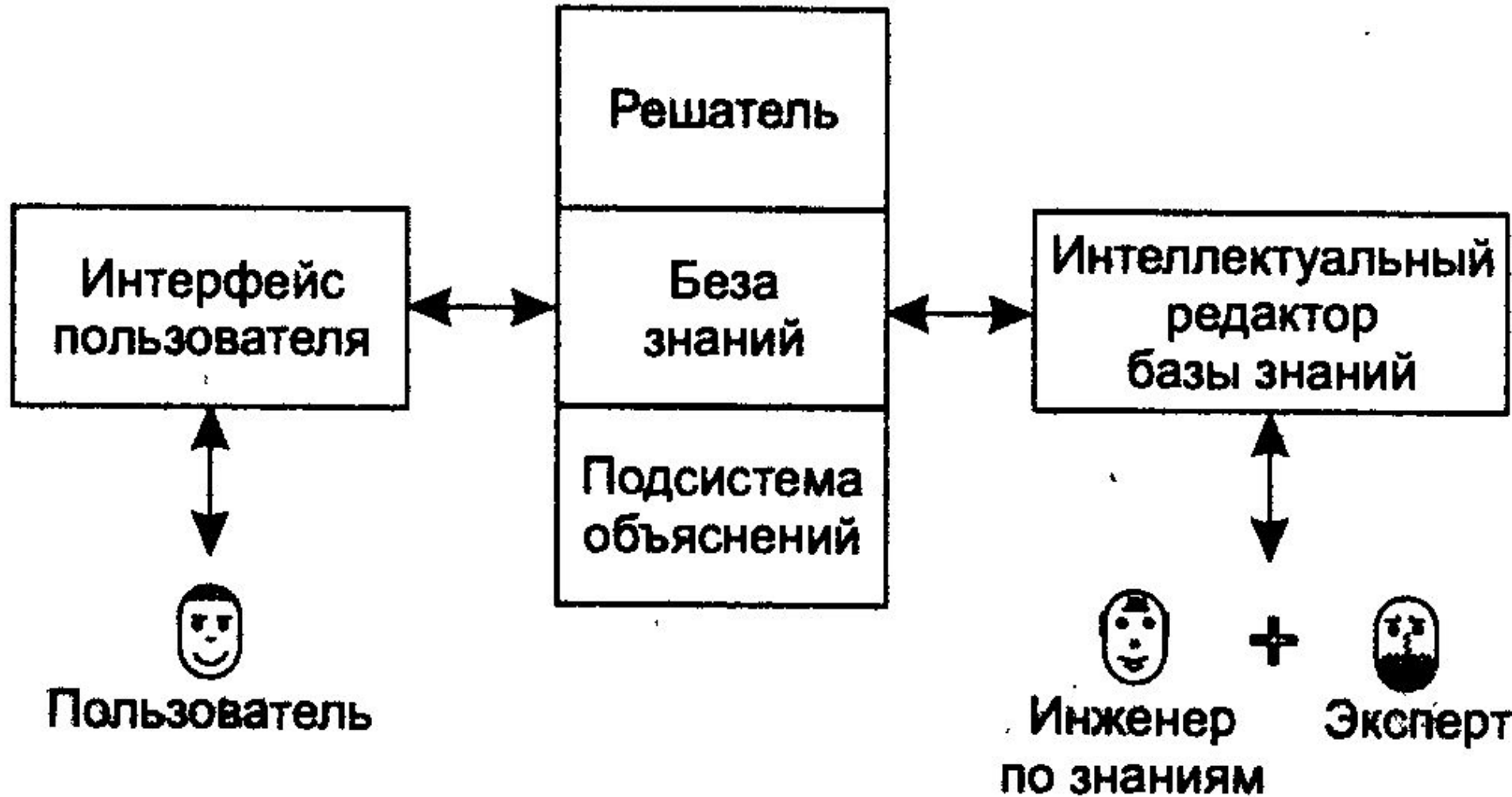
Основу успеха ЭС составили свойства:

- в ЭС знания отделены от данных, и мощность ЭС обусловлена в 1-ю очередь мощностью базы знаний и только во 2-ю очередь используемыми методами решения задач;
- решаемые ЭС задачи – слабоформализованные; используются экспериментальные, субъективные знания экспертов в определенной области.

Основные категории решаемых ЭС задач:

диагностика, управление (в т. ч. технологическими процессами), ***интерпретация, обучение, прогнозирование, проектирование, отладка и ремонт, планирование, мониторинг.***

Структура ЭС



Структура ЭС

Пользователь – специалист предметной области (обычно не самой высокой квалификации, т.е. не эксперт), для которого предназначена ЭС; нуждается в поддержке своей деятельности со стороны ЭС.

Инженер по знаниям – специалист в области ИИ, «промежуточный буфер» между экспертом и базой знаний (*когнитолог, аналитик*).

Структура ЭС

База знаний – ядро ЭС, набор знаний из предметной области, записанный на машинный носитель в форме, понятной эксперту и пользователю (некоторый язык, приближенный к естественному).

Кроме этого, есть «машинное» представление БЗ.

Решатель – программа, моделирующая рассуждения эксперта на основании знаний из БЗ (или: дедуктивная машина, машина вывода, блок логического вывода).

Структура ЭС

Подсистема объяснений – программа, позволяющая пользователю получить ответы на вопросы: *«Как получена данная рекомендация?»*, *«Почему система приняла такое решение?»*

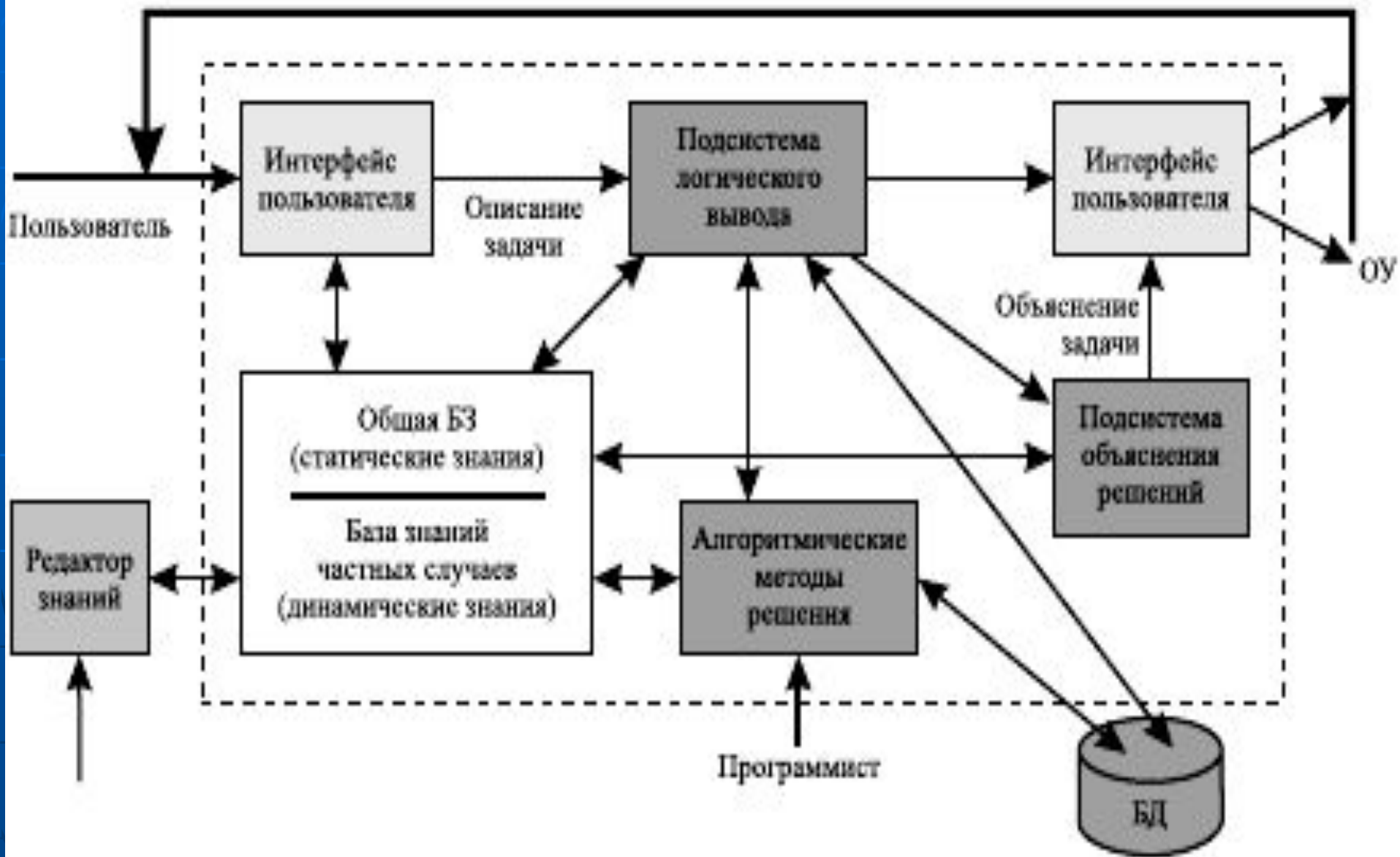
Ответ на вопрос «как» - это трассировка всего процесса получения решения с указанием использованных фрагментов БЗ (всех шагов цепи умозаключений).

Ответ на вопрос «почему» - ссылка на умозаключение, предшествующее полученному решению, т.е. отход на 1 шаг назад. Возможны и другие типы вопросов.

Структура ЭС

Интеллектуальный редактор БЗ – программа, представляющая инженеру по знаниям возможность создавать БЗ в диалоговом режиме, имеет различные сервисные средства, облегчает работу с БЗ.

Структура ЭС



Основа ЭС - **подсистема логического вывода** - использует информацию из базы знаний, генерирует рекомендации по решению задачи. Чаще всего для представления знаний в ЭС используются **системы продукций** и **семантические сети**.

Обязательными частями любой ЭС являются также

- **модуль приобретения знаний**
- **модуль отображения и объяснения решений**

Реальные ЭС используют также БД. Одновременная работа со знаниями и большими объемами информации из БД позволяет ЭС получить неординарные результаты, например:

- ***поставить сложный диагноз*** (медицинский или технический),
- ***открыть месторождение полезных ископаемых,***
- ***управлять ядерным реактором в реальном времени.***

Инструментальные средства создания ЭС

Языки программирования:

LISP, PROLOG, ...

Экспертные системы-оболочки:

**KEE, CENTAUR, G2, GDA, CLIPS,
AT_ТЕХНОЛОГИЯ, ...**

- предоставляют в распоряжение разработчика
- **инженера по знаниям** широкий набор для комбинирования систем представления знаний, языков программирования, объектов и процедур.

Prolog — «несбывшаяся мечта» об ЭВМ V поколения (уже обсуждали)

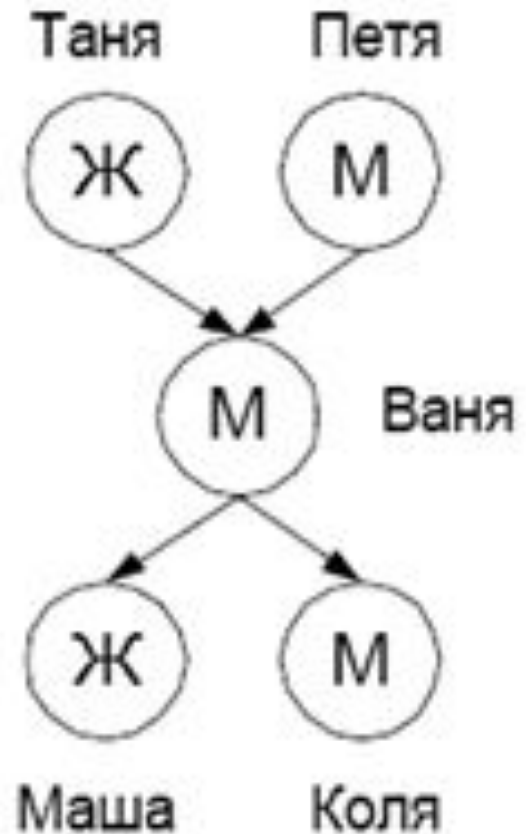
Язык **Prolog** (**PRO**gramming **for** **LOGIC** — программирование для логики) был создан в Европе в начале 1970-х годов. В отличие от всех предыдущих языков, включая Лисп, Пролог не является алгоритмическим языком, то есть он не описывает, **как** нужно получить результат. Он является декларативным языком, то есть показывает, **что** дано и **что** нужно получить.

Программа на Прологе состоит из фактов некоторой предметной области и правил. Интерпретирующая система сама делает логический вывод - применяет правила к фактам, используя аппарат математической логики.

Примеры программ на языке Prolog

Предметная область - родственные отношения между 5 членами семьи, описываемая фактами:

```
муж (петя) ,  
муж (ваня) ,  
муж (коля) ,  
жен (таня) ,  
жен (маша) ,  
мать (ваня, таня) ,  
отец (ваня, петя) ,  
отец (маша, ваня) ,  
отец (коля, ваня) .
```



Зададим следующие очевидные правила родства, используя синтаксис Пролога. Знак «:—» символизирует «есть по определению», запятая означает одновременное выполнение нескольких высказываний, а $\langle \rangle$ обозначает знак неравенства.

```
родитель (X, Y) :- отец (X, Y)
родитель (X, Y) :- мать (X, Y)
дед (X, Y) :- родитель (X, Z), отец (Z, Y)
брат (X, Y) :- муж (Y), родитель (X, Z),
родитель (Y, Z), X<>Y
```

Введя текст фактов и правил в Пролог-систему, можно задавать ей вопросы. Диалог пользователя с системой выглядит следующим образом (справа от вопроса на Прологе приведена его расшифровка на естественном языке):

```
GOAL> дед (коля, X) Кто дед Коли?
X = Петя
GOAL> брат (маша, X) Кто брат Маши?
X = Коля
```

Примеры ЭС

- ЭС **DENDRAL** (1960-е, Стэнфордский университет) расшифровка данные спектрографического анализа.
- ЭС **MYCIN** (1970-е, Стэнф. университет) ставила диагноз при инфекционных заболеваниях крови.
- ЭС **PROSPECTOR** (1974-1983, Стэнф. университет) обнаруживала полезные ископаемые.
- ЭС **SOPHIE** обучала диагностированию неисправностей в электрических цепях.
- ЭС **XCON** помогала конфигурировать оборудование для систем VAX фирмы DEC,
- ЭС **PALLADIO** помогала проектировать и тестировать СБИС-схемы.

Примеры ЭС (продолжение)

- ЭС **JUDITH** помогает специалистам по гражданским делам и вместе с юристом (с его слов) усваивает юридические предпосылки дела, а затем предлагает различные подходы к разрешению дела.
- ЭС **LRS** -помощь в подборе/анализе информации о судебных решениях и правовых актах в области кредитно-денежного законодательства, связанного с использованием векселей и чеков.
- ЭС **Ущерб** на основе российского трудового законодательства обеспечивает юридический анализ ситуации привлечения рабочих и служащих к материальной ответственности при нанесении предприятию материального ущерба действием⁶⁸ или бездействием.

Классификации ЭС

По назначению ЭС делятся на:

1) ЭС общего назначения

2) Специализированные ЭС:

- **проблемно-ориентированные** - для задач диагностики, проектирования, прогнозирования;
- **предметно-ориентированные** - для специфических задач, например, контроля ситуаций на атомных электростанциях.

Классификации ЭС

По степени зависимости от внешней среды выделяют:

- 1) Статические ЭС, не зависящие от внешней среды.**
- 2) Динамические ЭС, учитывающие динамику внешней среды и предназначенные для решения задач в реальном времени (время реакции таких систем – миллисекунды), эти системы реализуются, как правило, на языке C++.**

Классификации ЭС

По типу использования выделяют:

- 1) Изолированные ЭС.**
- 2) ЭС на входе/выходе других систем.**
- 3) Гибридные ЭС – ЭС, интегрированные с базами данных и другими программными продуктами (приложениями).**

Классификации ЭС

По сложности решаемых задач различают:

- 1) Простые ЭС - до 1000 простых правил.**
- 2) Средние ЭС - от 1000 до 10 000 структурированных правил.**
- 3) Сложные ЭС - более 10 000 структурированных правил.**

Классификации ЭС

По стадии создания выделяют:

- 1) Исследовательский образец ЭС,**
разработанный за 1-2 месяца с
минимальной БЗ.
- 2) Демонстрационный образец ЭС,**
разработанный за 2-4 месяца, например,
на языке типа *LISP, PROLOG, CLIPS*.

Классификации ЭС

По стадии создания выделяют:

- 3) Промышленный образец ЭС,**
разработанный за 4-8 месяцев,
например, на языке типа CLIPS с полной
БЗ.
- 4) Коммерческий образец ЭС,**
разработанный за 1,5-2 года, например,
на языке типа C++, Java с полной БЗ.

Модели представления данных в ЭС

Существует много моделей (или языков) представления знаний для ЭС. Большинство из них сводится к следующим классам:

- **Продукционные модели**
- **Семантические модели**
- **Фреймы**
- **Формальные логические модели**

Продукционная модель

Или модель, основанная на правилах, позволяет представить знания в виде предложений типа

«Если (условие), то (действие)»

Модель часто применяется в промышленных ЭС, т.к. имеет высокую наглядность и достаточно проста в использовании.

ПО на основе этой модели:

Язык **OPS 5**; «оболочки» или пустые ЭС – **EXSYS Professional, Каппа, ЭКСПЕРТ, ЭКО** и др.

Семантические сети

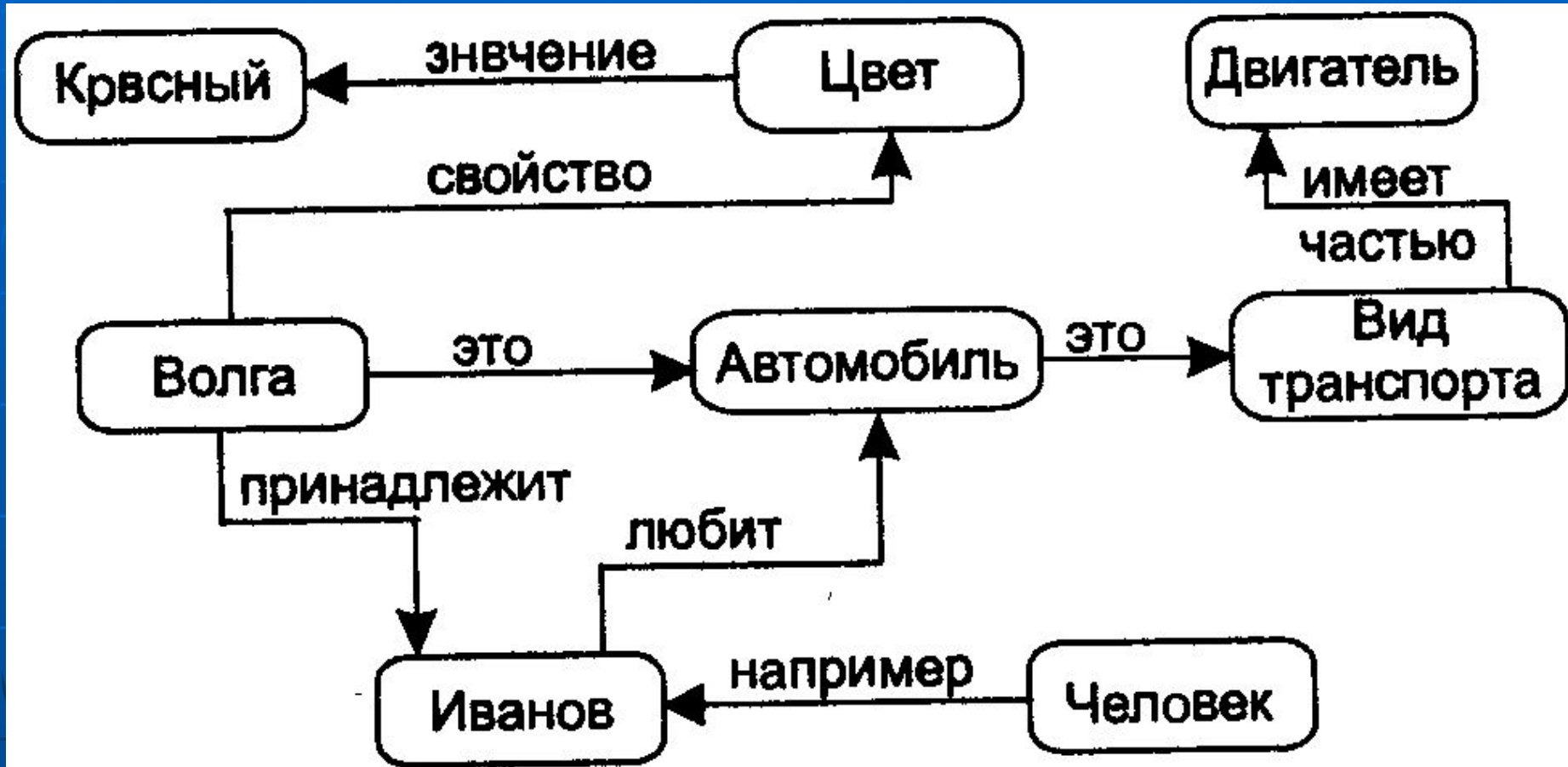
«Семантическая» означает «смысловая» (семантика – наука, устанавливающая отношения между символами и объектами, которые они обозначают, т.е. наука, определяющая смысл знаков).

Семантическая сеть – ориентированный граф, вершины которого – **понятия**, а дуги – **отношения** между понятиями.

В качестве понятий обычно выступают различные объекты, а отношения – это связи типа: «это», «имеет частью», «принадлежит». Особенность семантических сетей – обязательное наличие 3-х типов отношений:

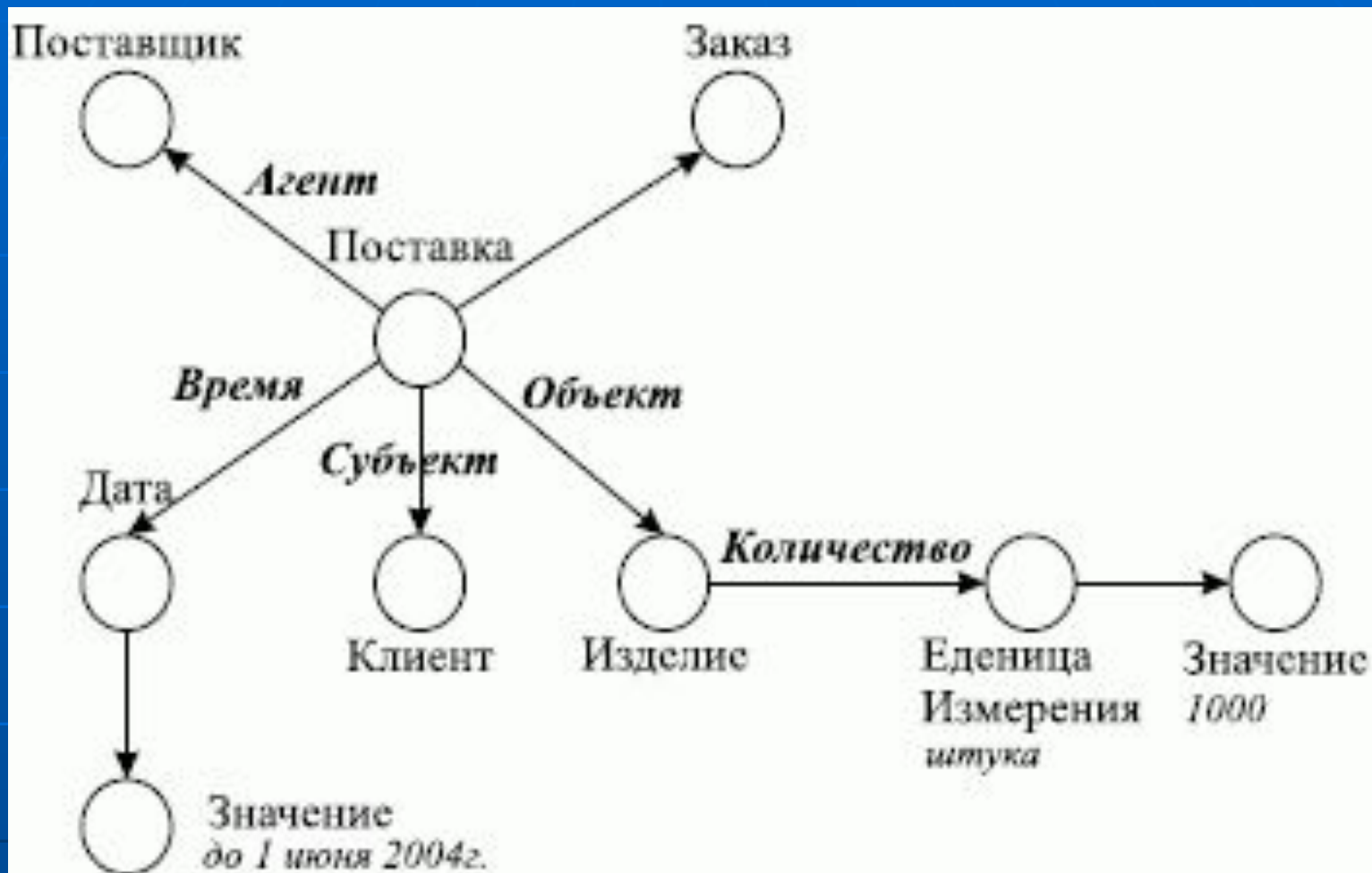
класс – элемент класса (цветок – роза);
свойство – значение (цвет – желтый);
пример элемента класса (роза – чайная).

Пример 1 - семантическая сеть



В качестве вершин выступают понятия «человек», «Иванов», «Волга», «автомобиль», «вид транспорта», «двигатель».

Пример 2 - семантическая сеть



Пример семантической сети для предложения "Поставщик осуществил поставку 1000 изделий по заказу клиента до 1 июня 2004 года". Между объектами Поставщик и Поставка определено отношение "агент", между объектами Изделие и Поставка определено отношение "объект" и т.д.

Фреймы

Термин **фрейм** (от англ. frame – «каркас», «рамка») предложен в 1970-е гг. для обозначения структуры знаний для восприятия пространственных сцен.

Фрейм – это абстрактный образ для представления некоего стереотипа восприятия. Фреймом также называется и формализованная модель для отображения образа.

Различают **фреймы-образцы**, или **прототипы**, хранящиеся в БЗ, и **фреймы-экземпляры**, создаваемые для отображения реальных фактических ситуаций на основе поступающих данных.

Фреймы

Модель фрейма позволяет отобразить все многообразие знаний о мире через:

фреймы-структуры – используются для обозначения объектов и понятий (заем, залог, вексель);

фреймы-роли (менеджер, кассир, клиент);

фреймы-сценарии (банкротство, собрание акционеров);

фреймы-ситуации (тревога, авария, рабочий режим устройства) и др.

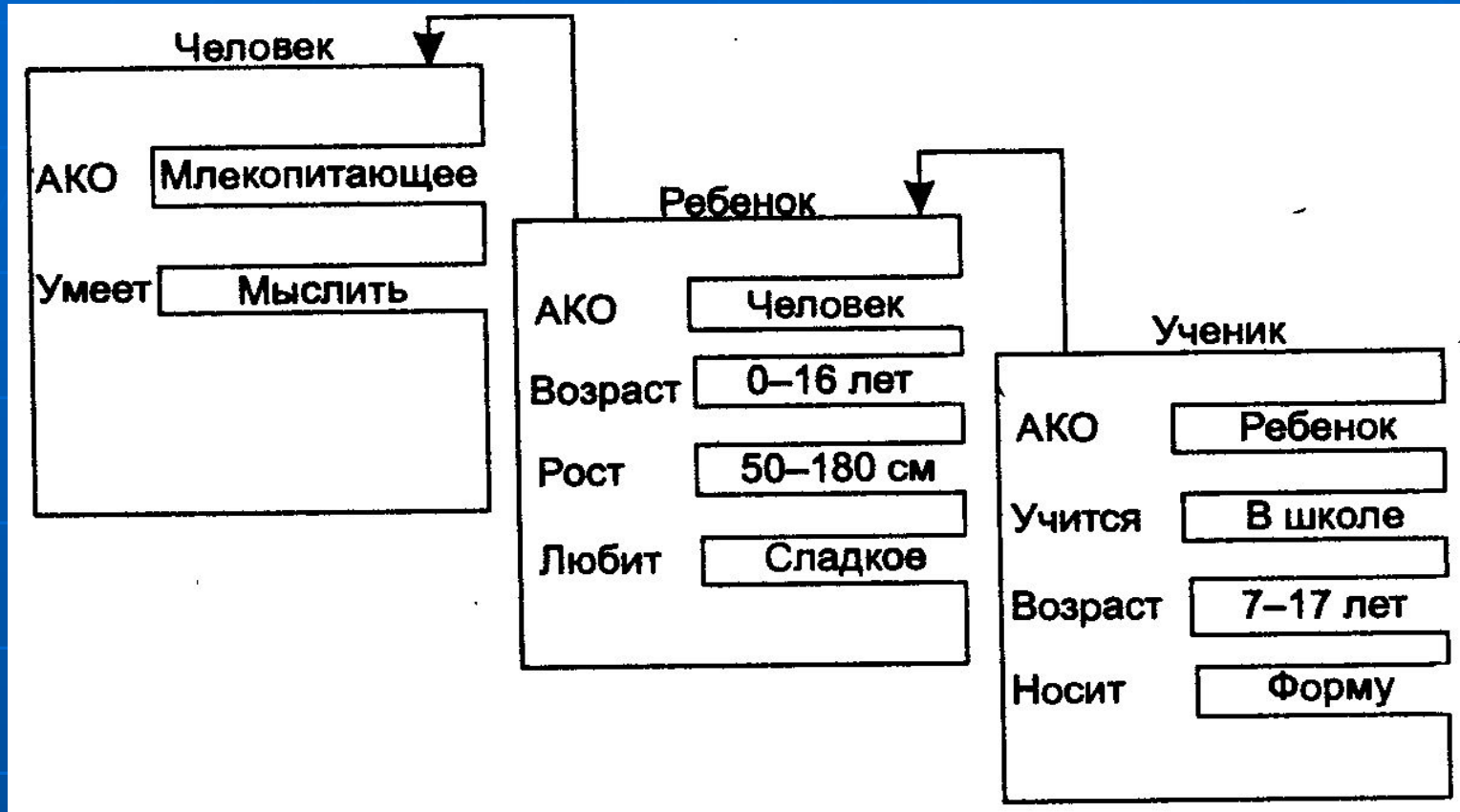
Фреймы

Структуру фрейма можно представить в виде таблицы:

Имя фрейма			
Имя слота	Значение слота	Способ получения значения	Присоединенная процедура

Важнейшее свойство теории фреймов взято из теории семантических сетей, это – **наследование свойств**. И во фреймах, и в семантических сетях наследование происходит по **АКО-связям** (A-Kind-Of = это).

Пример сети фреймов



Понятие «ученик» наследует свойства фреймов «ребенок» и «человек», находящихся на более высоком уровне иерархии. Тогда на вопрос «Любят ли ученики сладкое?» следует ответ «Да», т.к. этим свойством обладают все дети, что указано во фрейме «ребенок». Наследование свойств может быть частичным, так как возраст учеников не наследуется из фрейма «ребенок», т.к. явно указан в своем собственном фрейме.

Фреймы

Специальные языки представления знаний в сетях фреймов: FPL (Frame Representation Language), KPL (Knowledge Representation Language), фреймовая оболочка Карра и др.

Известные фрейм-ориентированные экспертные системы: ANALYST, МОДИС, TRISTAN, ALTERID.

Формальные логические модели

Традиционно в представлении знаний выделяют формальные логические модели, основанные на классическом исчислении предикатов 1-го порядка, когда предметная область или задача описывается в виде набора аксиом. Но в промышленных ЭС такие модели напрямую не используются, а применяются различные их «усложнения» и «расширения».

Классификации ЭС – еще одна

Экспертные системы

По задаче

Интерпретация данных

Диагностика ✓

Проектирование

Прогнозирование

Планирование

Обучение

По связям с реальным временем

Статическое

Квазидинамические

Динамические ✓

По типу ЭВМ

На супер-ЭВМ

На ЭВМ средней производительности

На символьных процессорах

На рабочих станциях ↗

на ПЭВМ

По степени интеграции

Автономные

✓ Гибридные (интегрированные)

Интерпретация данных - одна из традиционных задач для ЭС - процесс определения смысла данных, результаты которого должны быть согласованными и корректными.

Обычно - ***многовариантный анализ данных***, например:

- обнаружение и идентификация различных типов океанских судов по результатам аэрокосмического сканирования - ЭС **SIAP**;
- определение основных свойств личности по результатам психодиагностического тестирования в системах **АВТАНТЕСТ**, **МИКРОЛЮШЕР** и др.

Диагностика - соотнесение объекта с некоторым классом объектов и/или обнаружение неисправности.

Неисправность - это отклонение от нормы. Такая трактовка позволяет с единых теоретических позиций рассматривать и неисправность оборудования в технических системах, и заболевания живых организмов, и всевозможные природные аномалии.

Например, диагностика и терапия сужения коронарных сосудов – ЭС **ANGY**; диагностика ошибок в аппаратуре и математическом обеспечении ЭВМ - система **CRIB** и др.

Мониторинг. Основная задача- непрерывная интерпретация данных в реальном масштабе времени и сигнализация о выходе тех или иных параметров за допустимые пределы.

Главные проблемы — **«пропуск» тревожной ситуации** и **задача «ложного» срабатывания**. Сложность этих проблем - в размытости симптомов тревожных ситуаций и необходимость учета временного контекста.

Примеры ЭС: контроль за работой электростанций – ЭС **СПРИНТ**, помощь диспетчерам атомного реактора - ЭС **REACTOR**; контроль аварийных датчиков на химическом заводе – ЭС **FALCON**.

Проектирование - подготовка спецификаций (чертёж, пояснительная записка, ...) на создание «объектов» с заранее определенными свойствами.

Основные проблемы - получение четкого описания знаний об объекте и **проблема «следа»**: для организации эффективного проектирования необходимо формировать не только сами проектные решения, но и мотивы их принятия.

В задачах проектирования тесно связываются два основных процесса, выполняемых в рамках ЭС:

- 1) процесс вывода решения;*
- 2) процесс объяснения.*

Примеры ЭС:

проектирование конфигураций ЭВМ
VAX-11/780 в ЭС **XCON**,

проектирование БИС - **CADHELP**;

синтез электрических цепей — **SYN**.

Прогнозирование - предсказание следствий некоторых событий или явлений на основании анализа имеющихся данных.

Прогнозирующие ЭС логически выводят вероятные следствия из заданных ситуаций.

При этом обычно используется параметрическая динамическая модель, в которой значения параметров «подгоняются» под заданную ситуацию. Выводимые из этой модели следствия составляют основу для прогнозов с вероятностными оценками.

Примеры ЭС: предсказание погоды - ЭС **WILLARD**; оценки будущего урожая — **PLANT**;
прогнозы в экономике — **ECON**.

Планирование - нахождение планов действий, относящихся к объектам, способным выполнять некоторые функции.

В таких ЭС используются модели поведения реальных объектов, позволяющие логически выводить последствия планируемой деятельности.

Примеры ЭС: планирование промышленных заказов - **ISIS**; планирование эксперимента - **MOLGEN**.

Обучение

ЭС для обучения:

- диагностируют ошибки при изучении какой-либо дисциплины с помощью ЭВМ и подсказывают правильные решения
- аккумулируют знания о гипотетическом «ученике» и его характерных ошибках
- находят слабости в знаниях обучаемых и соответствующие средства для их ликвидации

Обучение

ЭС для обучения:

- планируют общение с учеником в зависимости от успехов ученика с целью передачи знаний.

Примеры ЭС:

- обучение языку программирования ЛИСП в ЭС «Учитель ЛИСПа»
- ЭС **PROUST** — обучение языку Паскаль.

Управление - поддерживают определенный режим деятельности, управляют поведением сложных систем в соответствии с заданными требованиями.

Примеры ЭС:

- помощь в управлении газовой котельной - **GAS**;
- управление системой календарного планирования **Project Assistant**.

Поддержка принятия решений -

совокупность процедур, обеспечивающая лицо, принимающее решения, необходимой информацией и рекомендациями, облегчающими процесс принятия решения.

Эти ЭС помогают специалистам выбрать и/или сформировать нужную альтернативу среди множества выборов при принятии ответственных решений.

Основы теории нейронных сетей

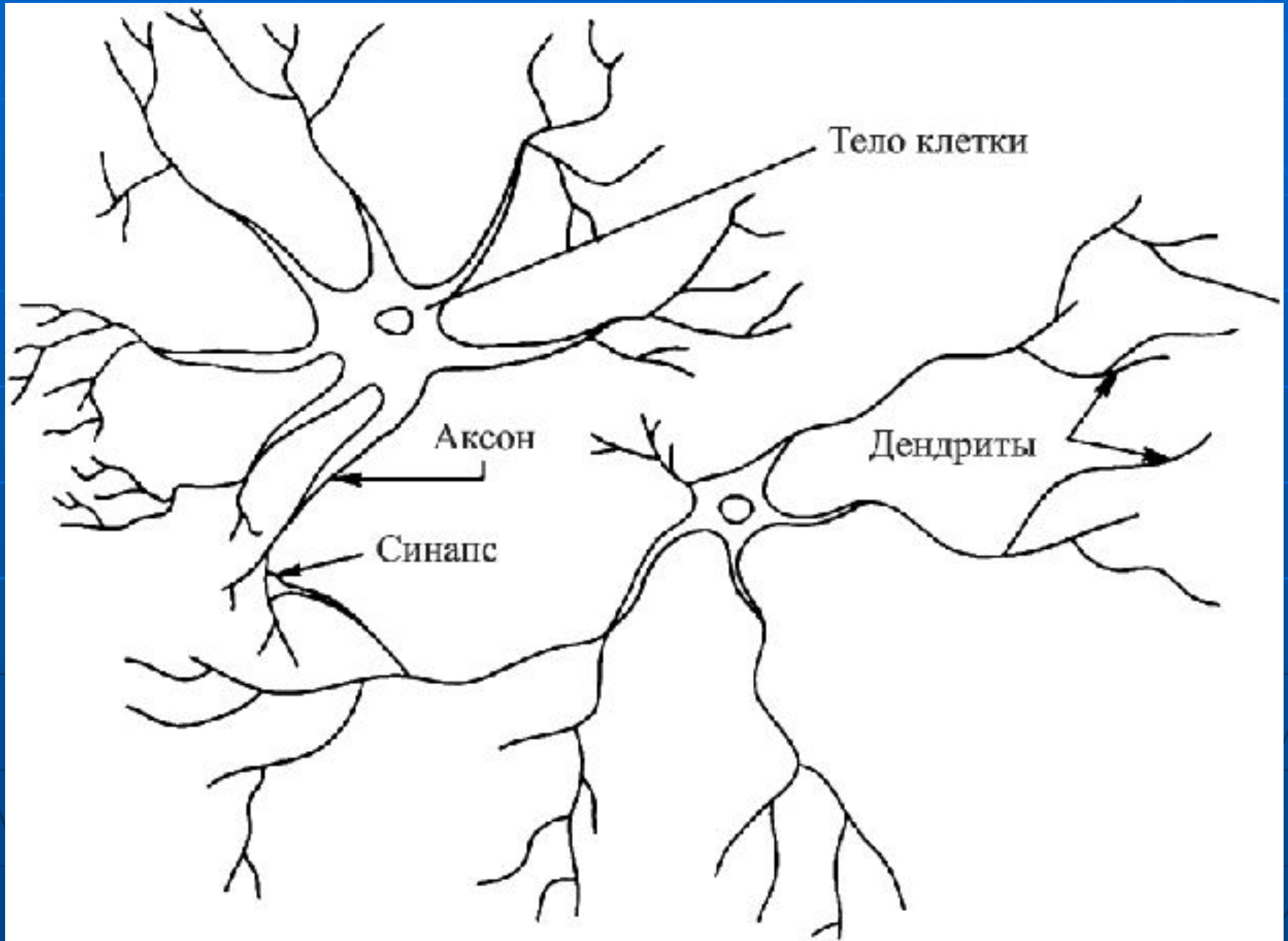
Нервная система человека, построенная из элементов, называемых **нейронами**, имеет ошеломляющую сложность.

Около **10^{11}** нейронов участвуют в примерно **10^{15}** передающих связях, имеющих длину метр и более.

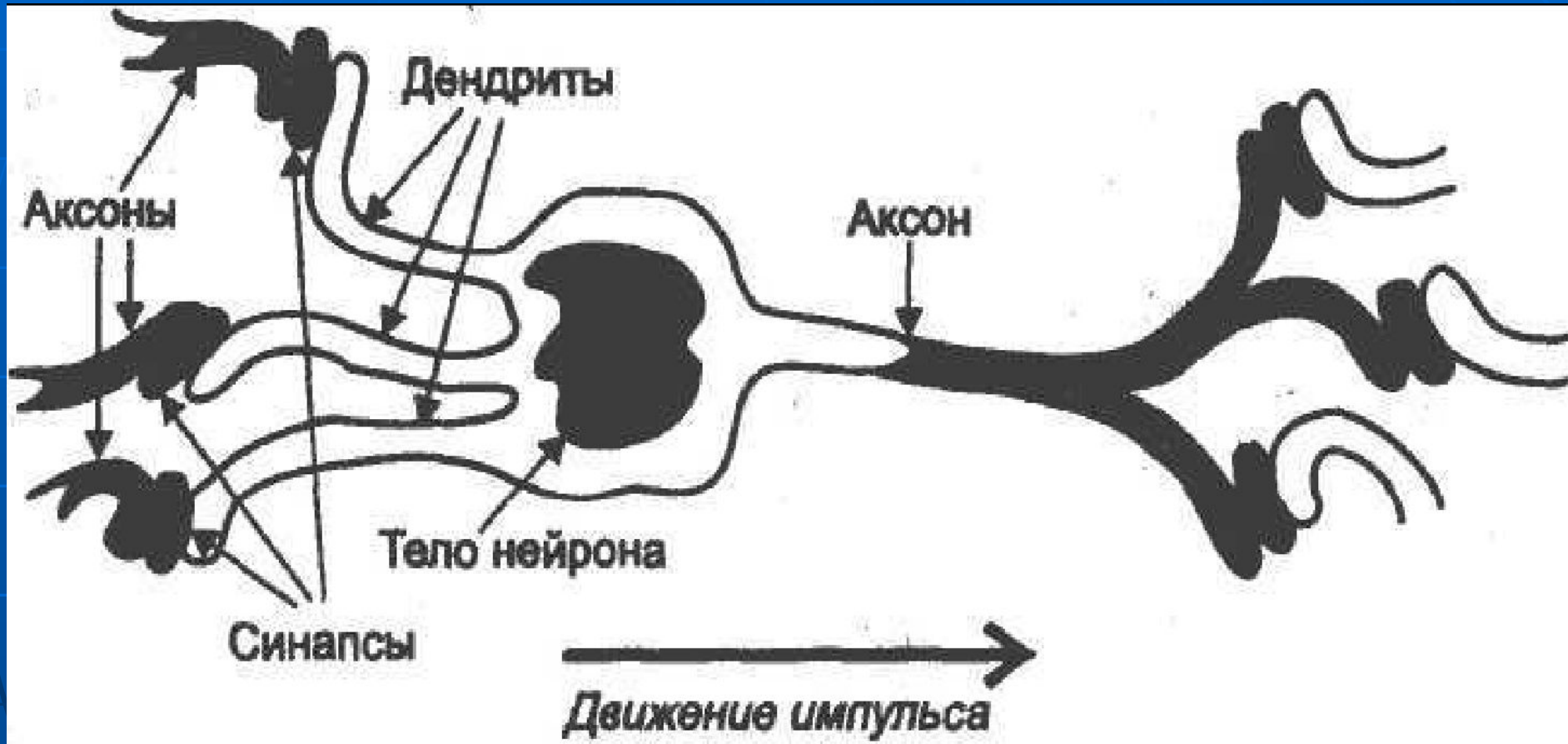
Основы теории нейронных сетей

Каждый нейрон обладает многими свойствами, общими с другими органами тела, но ему присущи абсолютно уникальные способности: принимать, обрабатывать и передавать электрохимические сигналы по нервным путям, которые образуют коммуникационную систему мозга.

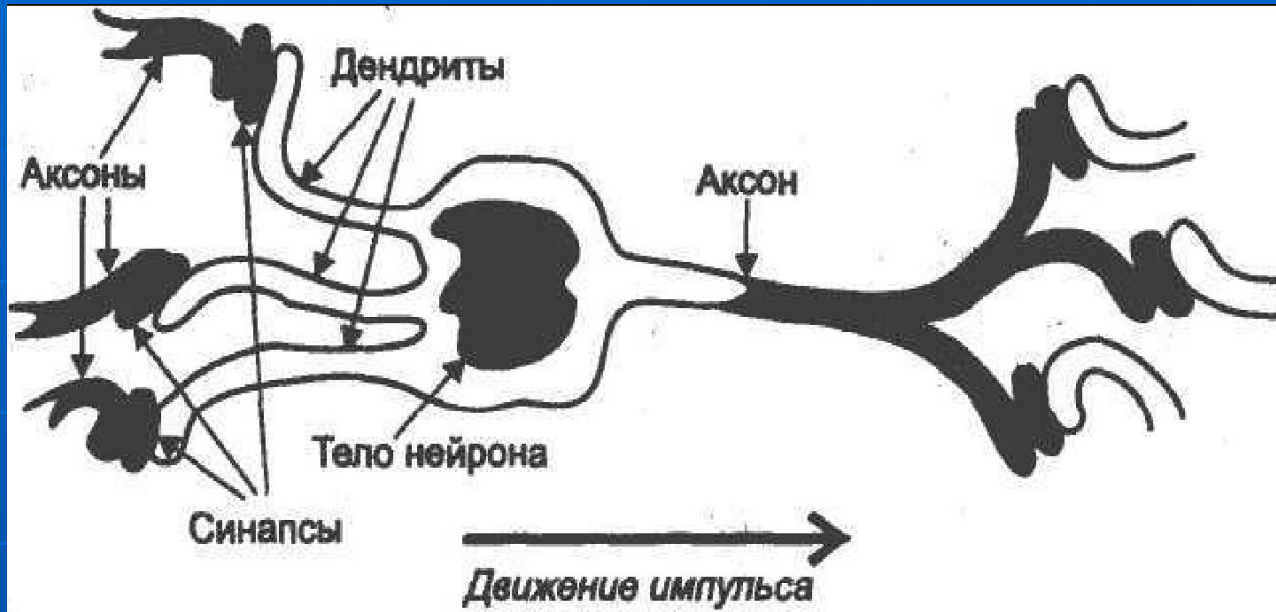
Нейроны человека



Нейроны человека

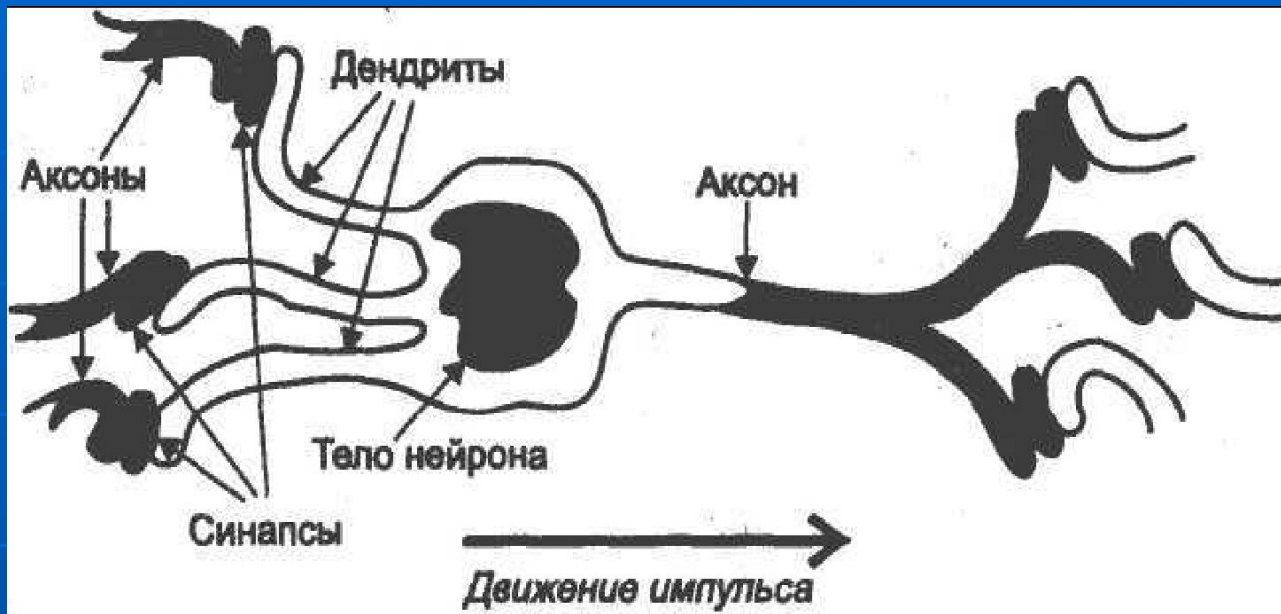


Нейроны человека



Дендриты идут от тела нервной клетки к другим нейронам, где они принимают сигналы в точках соединения, называемых **синапсами**. Принятые синапсом входные сигналы передаются к телу нейрона. Здесь они суммируются, причем одни входы стремятся возбудить нейрон, другие - воспрепятствовать его возбуждению.

Нейроны человека



Когда суммарное возбуждение в теле нейрона превышает некоторый порог, нейрон *возбуждается*, посылая по **аксону** сигнал другим нейронам. У этой основной функциональной схемы много усложнений и исключений, но большинство искусственных нейронных сетей моделируют лишь эти простые свойства.

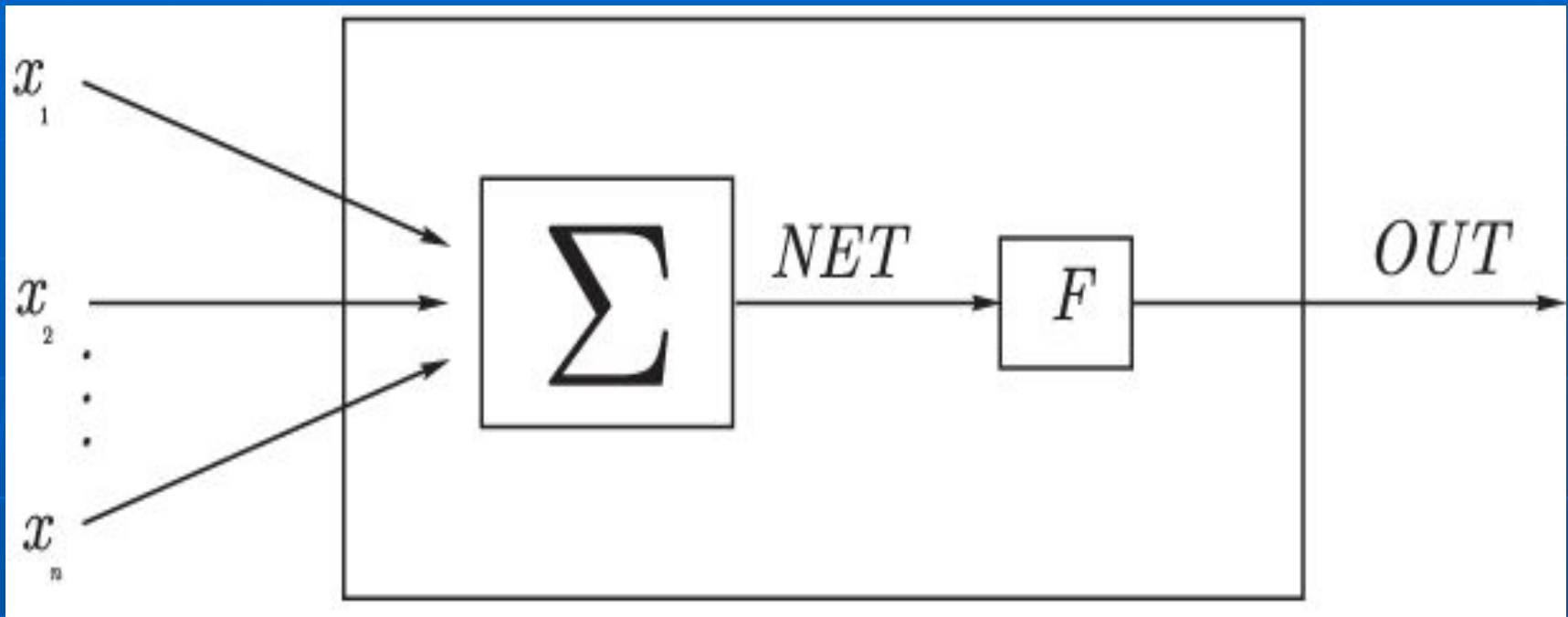
На сегодняшний день - две взаимно обогащающие друг друга цели нейронного моделирования:

1) понять функционирование нервной системы человека на уровне физиологии и психологии;

2) создать вычислительные системы (искусственные нейронные сети - ИС), выполняющие функции, сходные с функциями мозга.

Будем обсуждать цель **2**.

Искусственный нейрон



Искусственный нейрон имитирует свойства нейрона биологического: на вход поступает много сигналов, каждый из которых является выходом другого нейрона. Каждый вход умножается на некоторый **вес** - аналог **синаптической силы**, и все произведения суммируются, определяя **уровень активации нейрона**.

Искусственный нейрон

Множество входных сигналов, описываемых вектором

$$X = (x_1, x_2, \dots, x_n),$$

поступает на искусственный нейрон. Эти входные сигналы соответствуют сигналам, приходящим в синапсы биологического нейрона. Каждый входной сигнал умножается на свой **вес** – элемент вектора

$$W = (w_1, w_2, \dots, w_n),$$

и поступает на **суммирующий блок Σ** (аналог тела биологического нейрона). Каждый вес - аналог "силы" одной биологической **синаптической связи**.

Суммирующий блок складывает взвешенные входы, создавая выход, который обозначим **NET**. В векторных обозначениях:

$$NET = X W = x_1 w_1 + \dots + x_n w_n.$$

Искусственный нейрон

Сигнал **NET** далее, как правило, преобразуется **активационной функцией F** и дает **выходной нейронный сигнал OUT**:

$$OUT = F(NET).$$

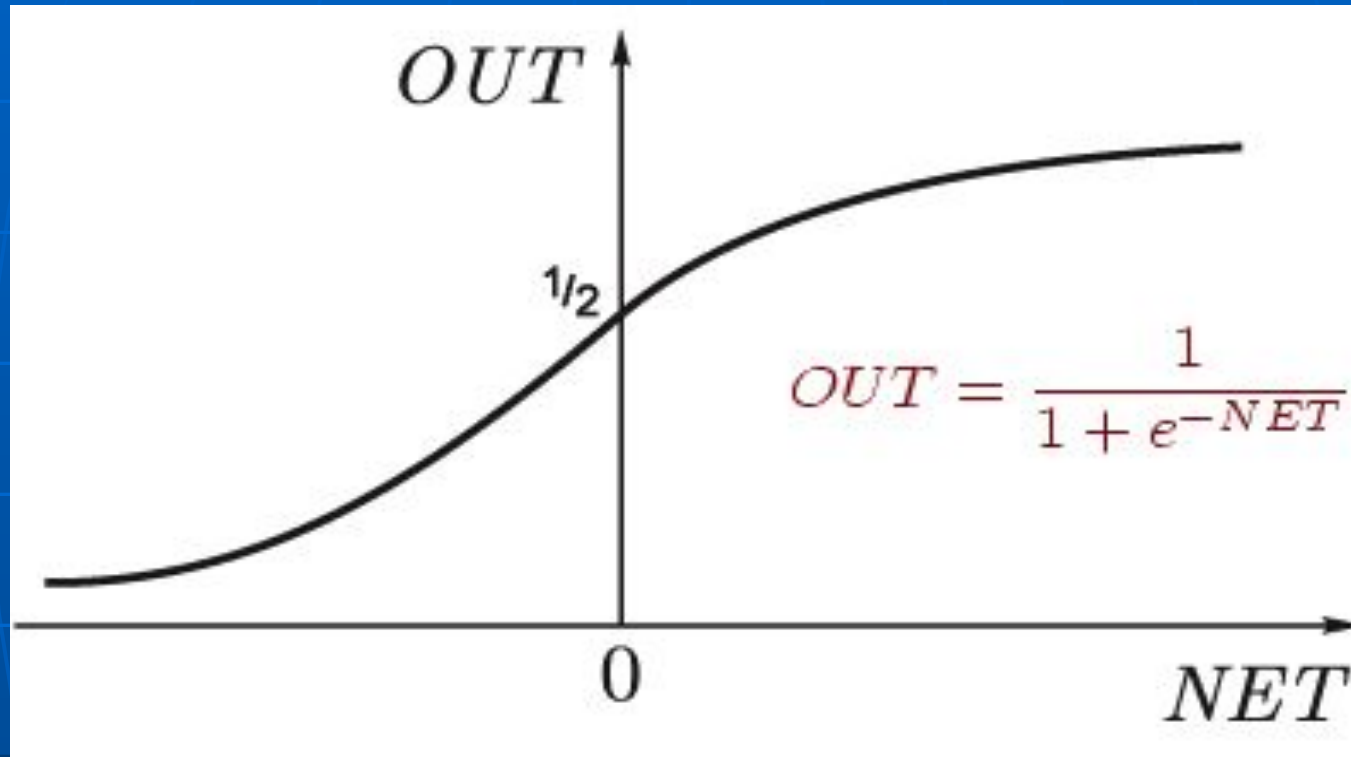
Активационная функция может быть:

- 1) **линейной функцией;**
- 2) **пороговой функцией** вида

$$OUT = \begin{cases} 1, & \text{если } NET > T; \\ 0, & \text{если } NET \leq T \end{cases}$$

где **T** - некоторая постоянная пороговая величина;

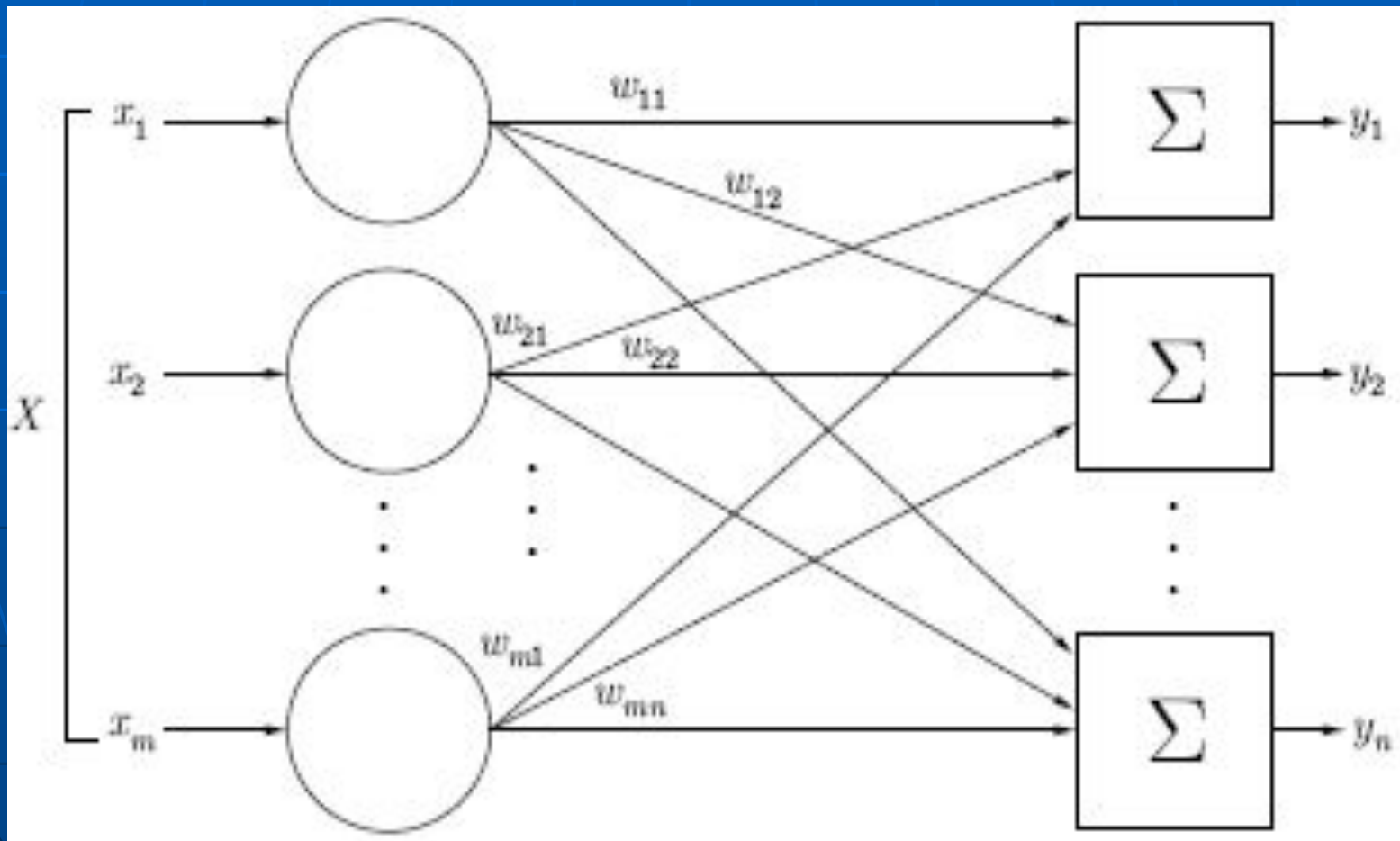
3) S-образной функцией, точнее моделирующей нелинейную передаточную характеристику нейрона и дающей больше возможностей настройки НС:



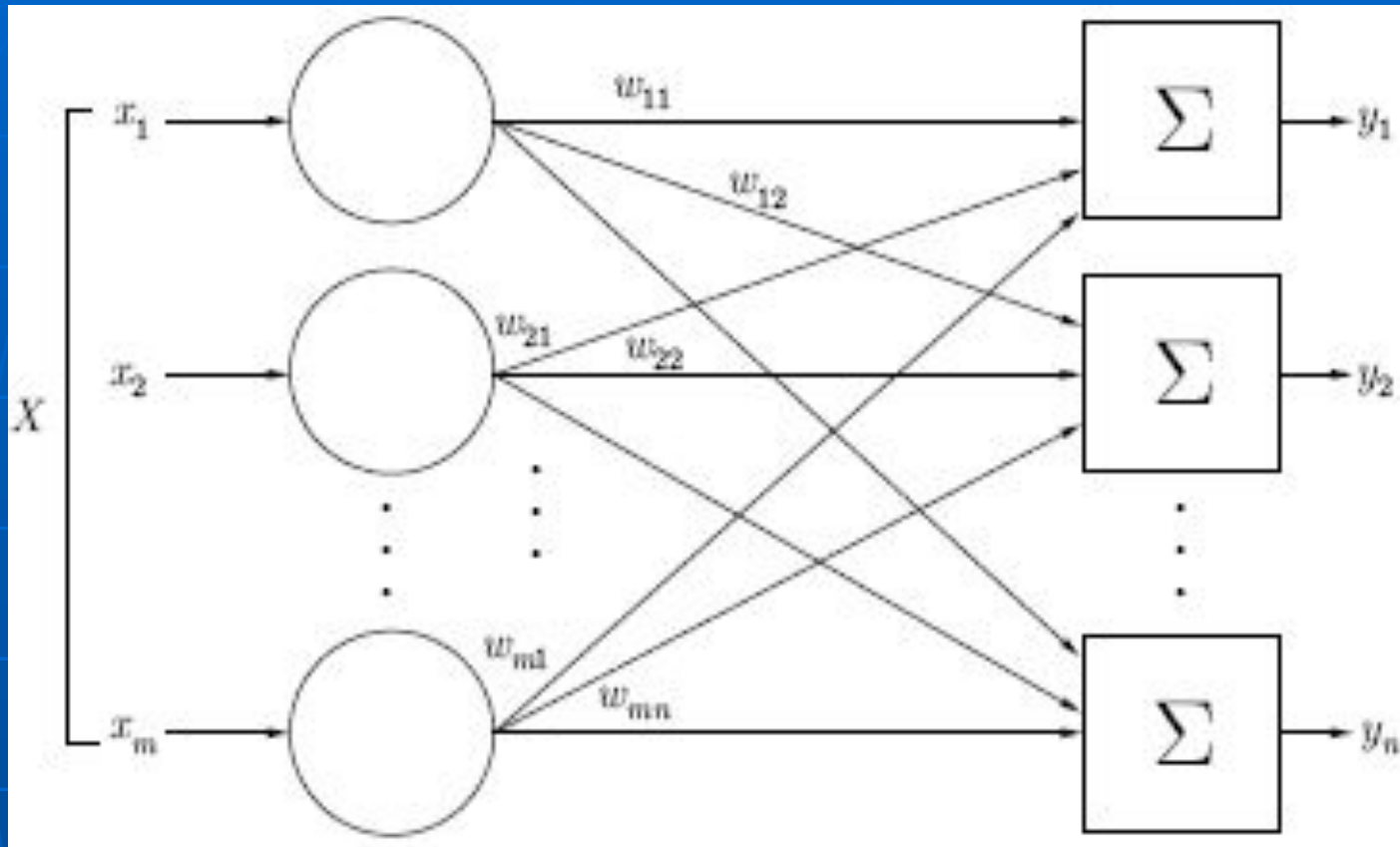
Если блок F сужает диапазон изменения величины NET так, что при любых значениях NET значения OUT принадлежат некоторому конечному интервалу, то функция F называется «сжимающей» функцией.

Однослойные искусственные нейронные сети

Один нейрон способен выполнять только простейшие вычисления, поэтому для более сложных нейронных вычислений нейроны соединяют в сети. Простейшая сеть состоит из группы нейронов, образующих **слой**:



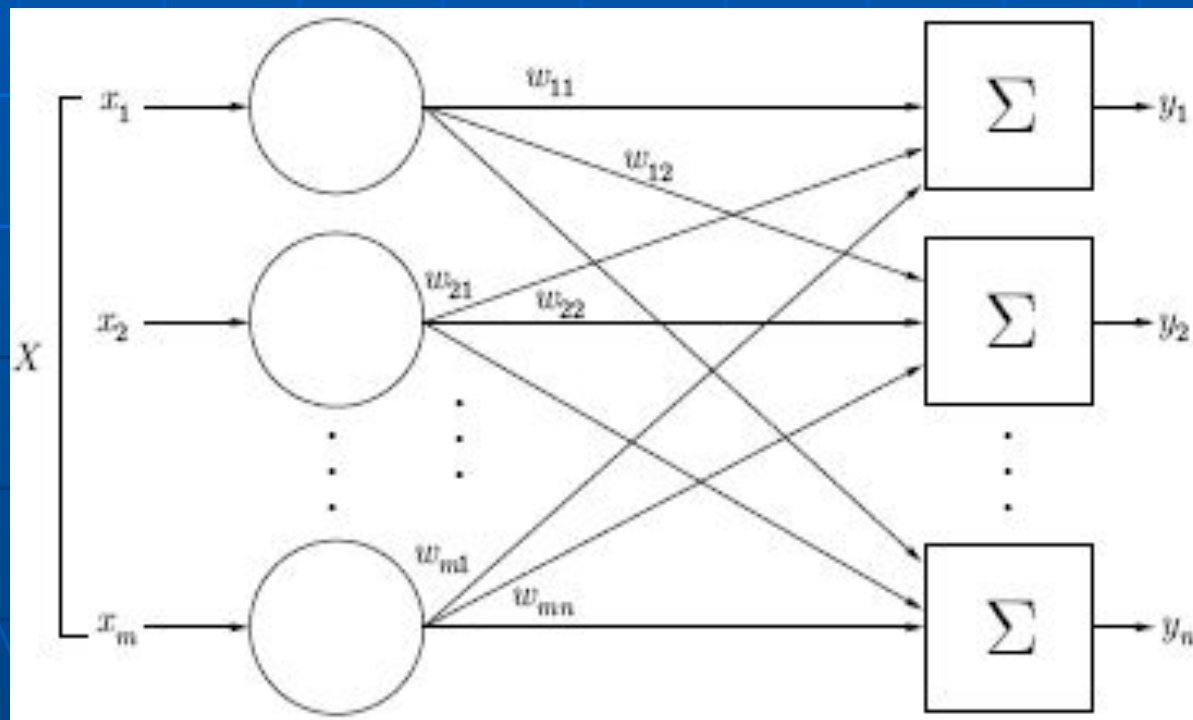
Однослойные искусственные нейронные сети



Вершины-круги слева служат для распределения входных сигналов, не выполняют вычислений и не считаются слоем. Вычисляющие нейроны (справа) обозначены квадратами.

Однослойные искусственные нейронные сети

Каждый элемент из множества входов X отдельным весом соединен с каждым искусственным нейроном, выдающим взвешенную сумму входов в сеть. В сетях многие соединения могут отсутствовать и возможны соединения между выходами и входами элементов в слое.



Однослойные искусственные нейронные сети

Удобно считать веса элементами матрицы W (m строк \times n столбцов, где m – число входов, n – число нейронов). Например, W_{23} – вес, связывающий 3-й вход со 2-м нейроном.

Тогда выходной вектор N , компонентами которого являются выходы OUT отдельных нейронов, определяется по формуле:

$$N = XW,$$

где N, X – векторы-строки.

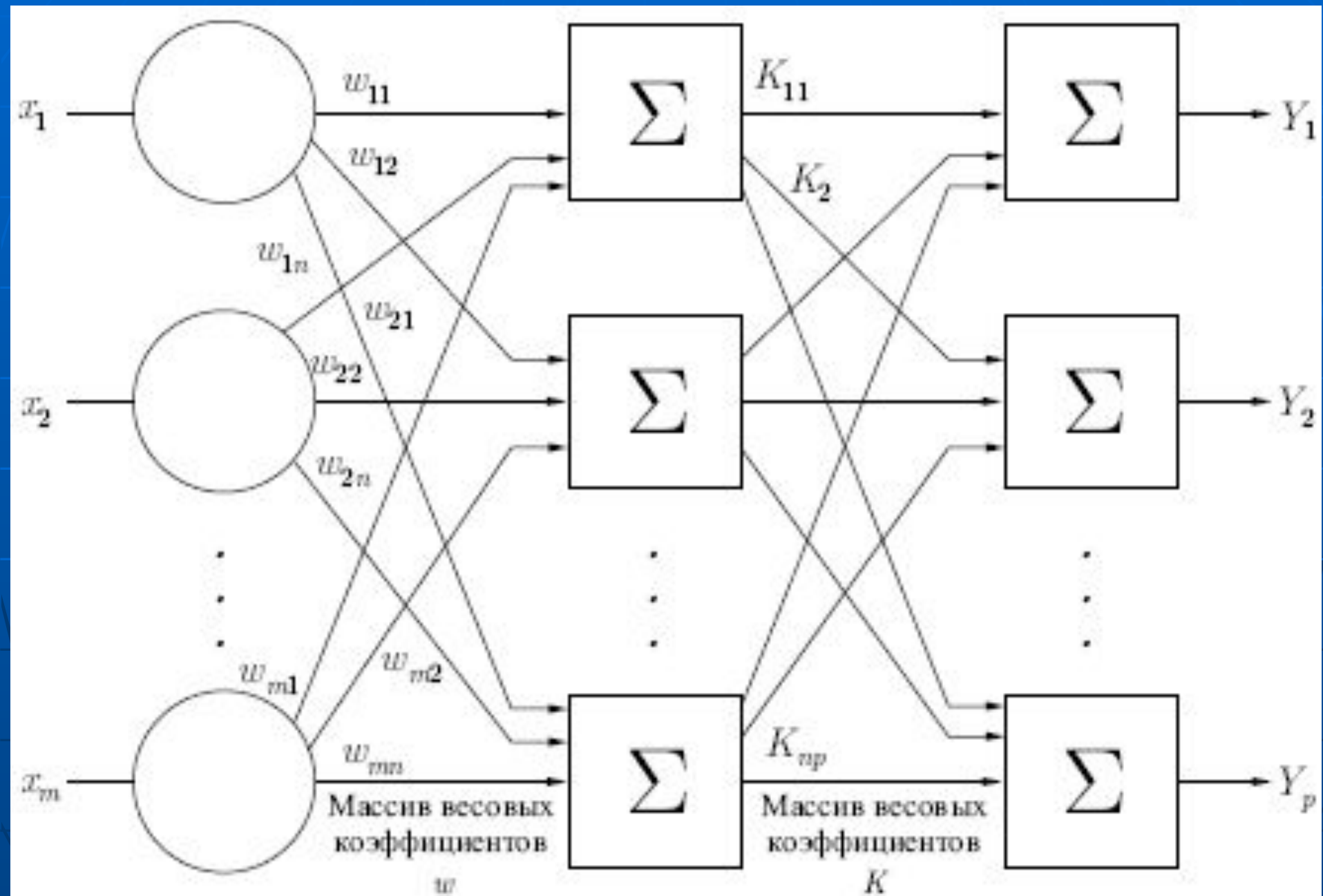
Многослойные искусственные НС

Более сложные НС обладают, как правило, и большими вычислительными возможностями. Созданы сети самых разных конфигураций.

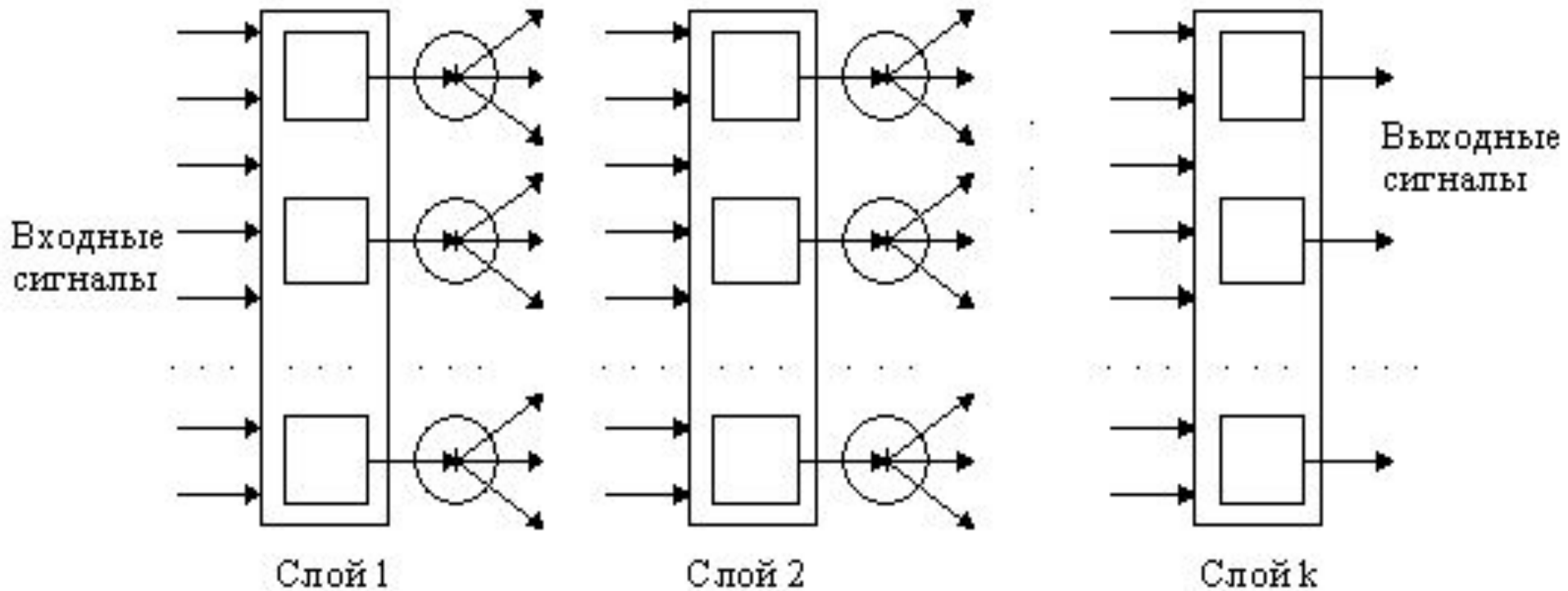
Послойная организация нейронов копирует слоистые структуры определенных отделов мозга. Многослойные сети могут строиться из каскадов слоев. Выход одного слоя является входом для последующего слоя.

Многослойные сети не могут привести к увеличению вычислительной мощности по сравнению с однослойной сетью, если активационная функция между слоями линейна.

Многослойные искусственные нейронные сети



Многослойные искусственные нейронные сети



Виды НС

Сети без обратных связей или **сети прямого распространения** - сети без соединений, идущих от выходов некоторого слоя к входам этого же слоя или предшествующих слоев. У таких сетей «нет памяти», их выход полностью определяется текущими входами и значениями весов.

Виды НС

Сети с обратными связями - сети более общего вида, имеют соединения от выходов к входам. В некоторых конфигурациях сетей с обратными связями предыдущие значения выходов возвращаются на входы; выход, следовательно, определяется как текущим входом, так и предыдущими выходами.

Поэтому такие сети могут обладать свойствами, сходными с кратковременной «памятью», где сетевые выходы тоже частично зависят от предыдущих входов.

Обучение искусственных нейронных сетей

Сеть обучается, чтобы для некоторого множества входов давать желаемое (или близкое к желаемому) множество выходов. Каждое такое входное (или выходное) множество рассматривается как вектор.

Обучение искусственных нейронных сетей

Обучение - путем последовательного предъявления *входных векторов* с одновременной подстройкой весов в соответствии с определенной процедурой.

В процессе обучения веса сети постепенно становятся такими, чтобы каждый входной вектор вырабатывал требуемый выходной вектор.

Различают алгоритмы обучения **с учителем** и **без учителя.**

Обучение с учителем

Предполагается, что для каждого входного вектора существует **целевой вектор**, представляющий собой требуемый выход. Вместе они называются **обучающей парой**. Обычно сеть обучается на некотором числе таких обучающих пар.

Обучение с учителем

Предъявляется входной вектор, вычисляется выход сети и сравнивается с соответствующим целевым вектором, разность (ошибка) с помощью обратной связи подается в сеть, и веса изменяются в соответствии с алгоритмом, стремящимся минимизировать ошибку.

Входные векторы обучающего множества предъявляются последовательно, ошибки вычисляются и веса подстраиваются для каждого вектора до тех пор, пока ошибка по всему обучающему массиву не достигнет приемлемо низкого уровня.

Обучение без учителя

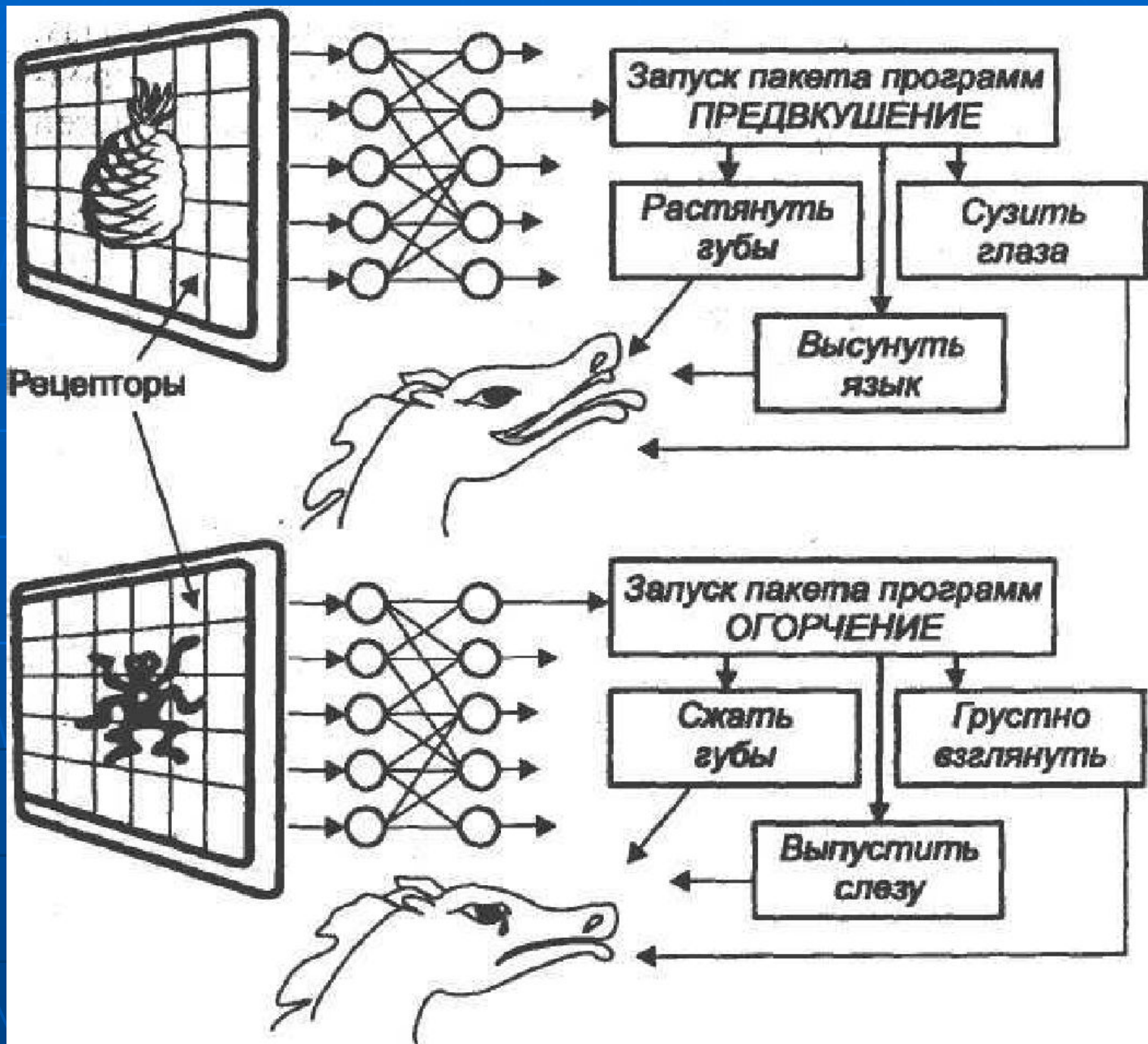
Является более правдоподобной моделью обучения для биологической системы. Модель развита Кохоненом и др., не нуждается в целевом векторе для выходов => нет сравнения с predetermined идеальными ответами.

Обучающее множество состоит лишь из входных векторов. Обучающий алгоритм подстраивает веса сети так, чтобы получались согласованные выходные векторы, т. е. чтобы предъявление достаточно близких входных векторов давало одинаковые выходы.

Обучение без учителя

Обучение выделяет статистические свойства обучающего множества и группирует сходные векторы в **классы**. Предъявление на вход вектора из данного класса даст определенный выходной вектор, но до обучения невозможно предсказать, какой выход даст данный класс входных векторов.

Приложения НС – распознавание образов



Приложения НС – распознавание образов

