

# Интеллектуальные информационные системы (ИИС) Лекция 6

Производственные системы.

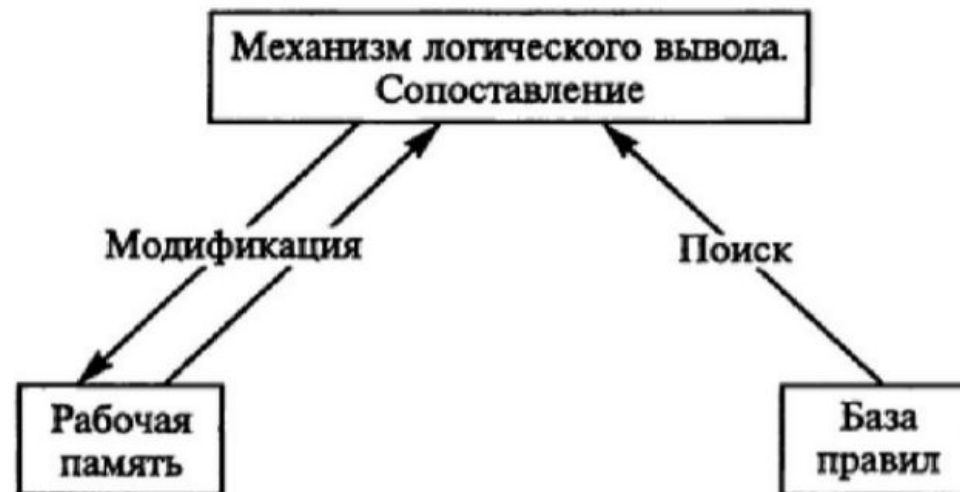
# Продукционная модель знаний

Продукционная модель знания — модель, основанная на правилах, позволяет представить знание в виде предложений типа «Если (условие), то (действие)»

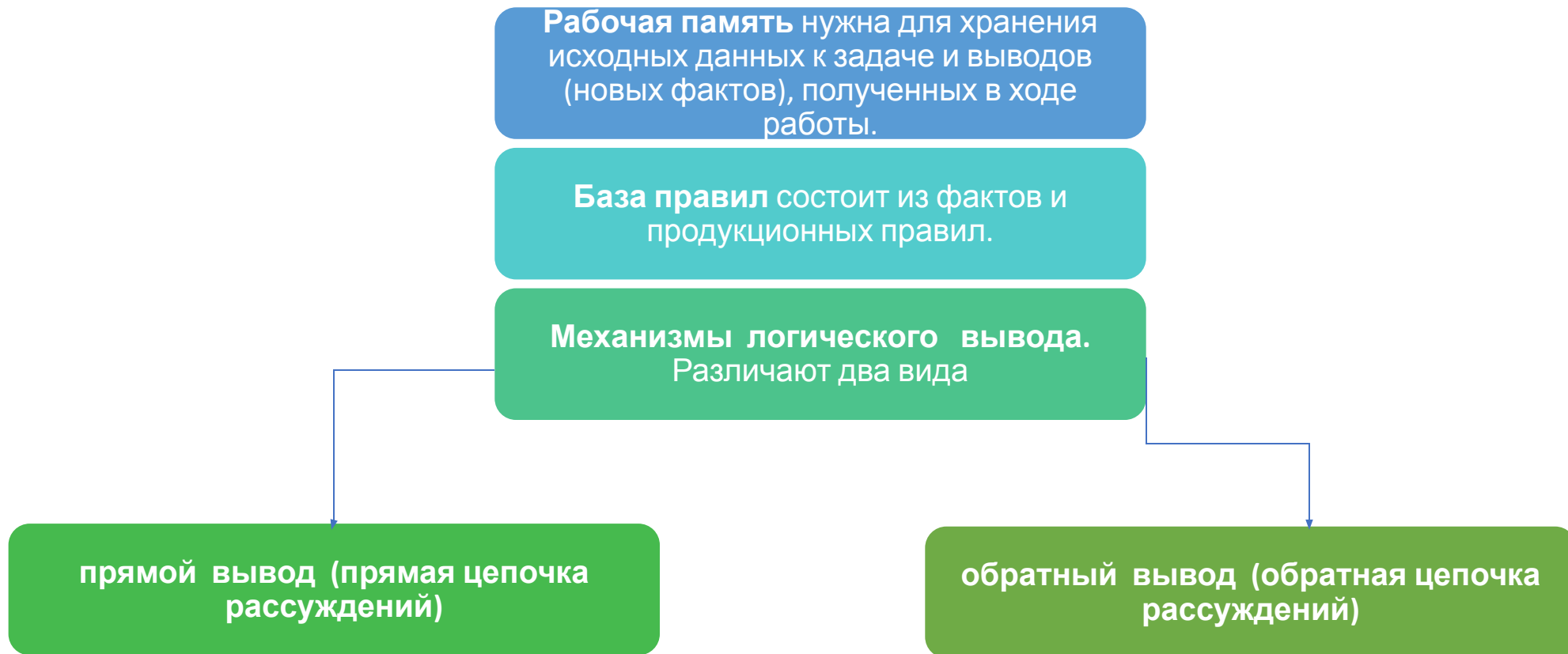
**Факт** – это есть предикат, который является истинным. Он фиксирует (определяет) некоторое отношение между объектами.

Продукционные системы основаны на использовании базы правил и механизмов логического вывода.

Блок- схема продукционной системы



# Элементы продукционной системы



# Пример

БАЗА ЗНАНИЙ:

Факт: это некоторое утверждение  $X$  есть  $Y$ , обозначают  $Y(X)$ .

«Сократ есть человек, Платон тоже человек»:

Человек (Сократ), Человек (Платон)

Продукционное правило:

ЕСЛИ  $A$ , ТО  $B$  ; Обозначают:  $B :- A$  (Если  $A$  истина, то  $B$  тоже истина)

Примеры:

«Если «некто» – человек, то он смертен».

(Или «Смертен «некто», если он человек»)

Пусть  $Y$  – «некто», тогда продукционное правило выглядит так:

Смертен ( $Y$ ):-Человек ( $Y$ ).

# Пример запросов

## Запрос 1:

Человек (X)?

## Вывод продукционной системы:

X = Сократ

X = Платон

Yes (да) – выдаст система в конце.

## Запрос 2:

Смертен (Сократ)?

Вывод: Yes (да).

## БАЗА ЗНАНИЙ

- Человек (Сократ).
- Человек (Платон).
- Смертен (Y):-Человек (Y).

## Запрос 3:

Смертен (кто)?

Результат:

кто = Сократ

кто = Платон

Yes (да).

# Преимущества и недостатки

## Преимущества :

- наглядность;
- легкость модификации;
- простой механизм логического вывода;
- универсальность, как метода программирования

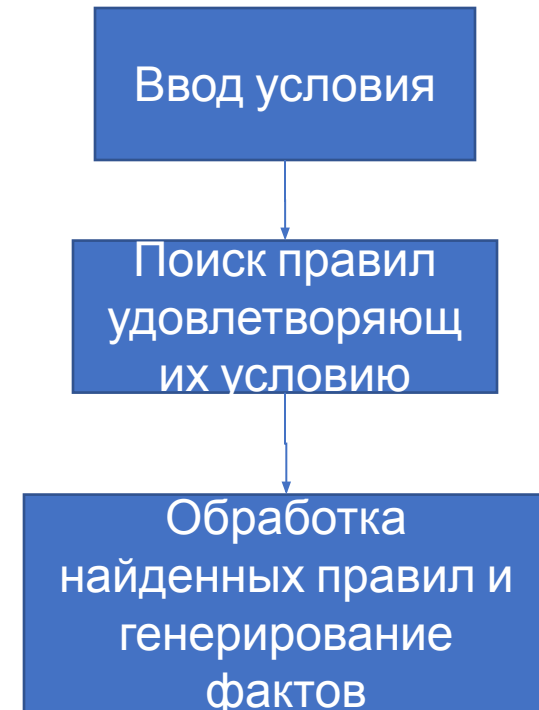
## Недостатки:

- противоречивость при большом количестве продукций;
- сложность контроля правильности программ продукционных систем

# ПРЯМОЙ ВЫВОД – ВЫВОД ОТ ДАННЫХ К ПОИСКУ ЦЕЛИ

Суть: формирование цепочки вывода заключается в многократном повторении элементарных шагов "сопоставить – выполнить".

Механизм вывода начинает сопоставлять образцы из условных частей правил с образцами, хранимыми в рабочей памяти. Если образцы из условной части имеются в рабочей памяти, то условная часть считается истинной, в противном случае – ложной.



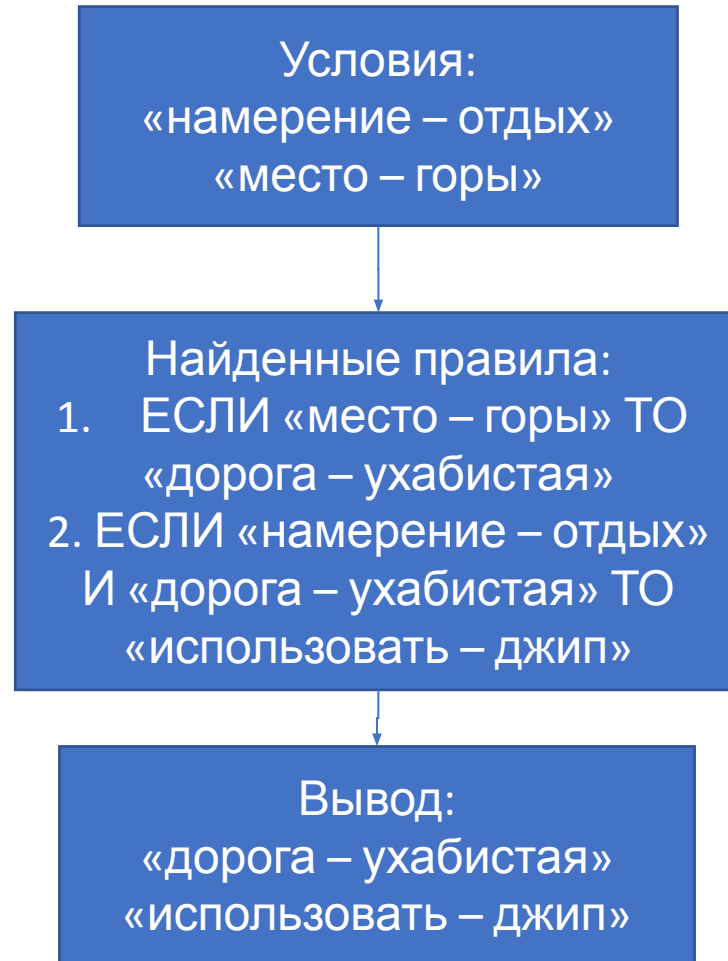
# Алгоритм прямого вывода

Алгоритм прямого вывода обычно основан на *стратегии поиска в ширину*. Этот процесс предусматривает следующее:

1. Изначально система содержит описание ряда ситуаций.
2. Для каждой ситуации система ищет в базе знаний правила, в условной части которых содержится соответствующее условие.
3. В соответствии с консеквентом (частью ТО) каждое правило может генерировать новые факты, которые добавляются к уже имеющимся в рабочей памяти.
4. Система обрабатывает каждый вновь сгенерированный факт. При наличии хотя бы одного правила, в антецеденте (части ЕСЛИ) которого присутствует данный факт выполняются действия, начиная с пункта 2.
5. Рассуждения заканчиваются, когда больше нет необработанных фактов и правил.



# Пример

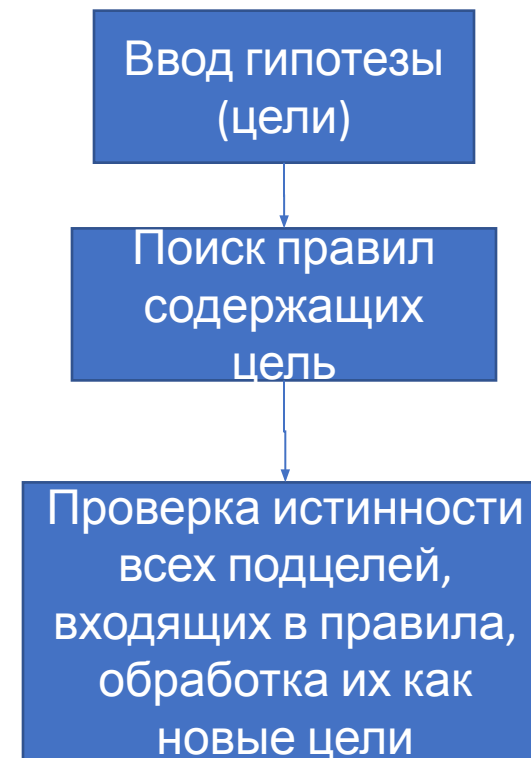


# Обратная цепочка рассуждений

От цели, для ее подтверждения, к данным

Суть: "Что нужно, чтобы правая часть данного правила была справедлива, и есть ли необходимые суждения в рабочей памяти?"

Исходя из изложенного, применяется для того, чтобы по известному результату найти причины, которые его вызвали

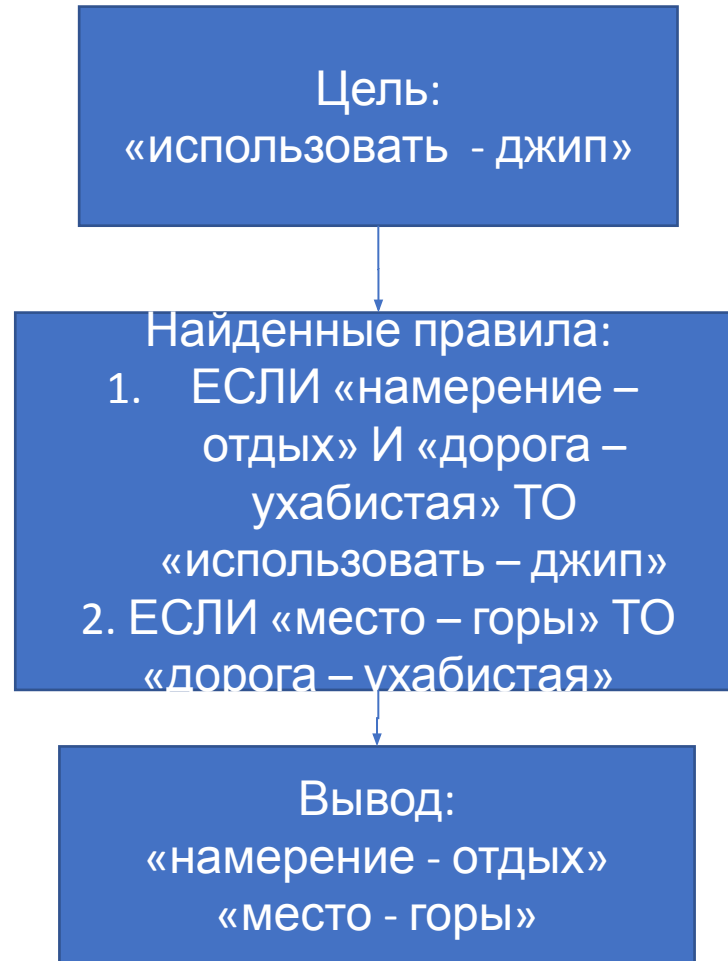


# Алгоритм обратного вывода

Алгоритм прямого вывода обычно основан на *стратегии поиска в глубину*. Этот процесс предусматривает следующие шаги:

1. Определить цель для логического вывода и выбрать ее в качестве текущей подцели.
2. В списке правил найти первое вхождение этой подцели. Если правило найдено, перейти к рассмотрению условной части найденного правила. Если правило не найдено, сообщить пользователю, что ответ найти невозможно.
3. Выбрать в качестве подцелей факты из условия из данного правила.
4. Если в списке подцелей имеются факты, истинность или значение которых могут быть запрошены у пользователя, то задать пользователю соответствующие вопросы.
5. Если очередная подцель выведена, то перейти к шагу 2.
6. Если очередная подцель не может быть выведена или запрошена у пользователя, сообщить, что ответ получить невозможно.
7. Если все подцели подтверждены, то сообщить пользователю окончательный вывод.

# Пример



# Механизм выбора

Механизм выбора – это механизм который управляет перебором правил и позволяет формировать логические выводы.

# Функция вывода

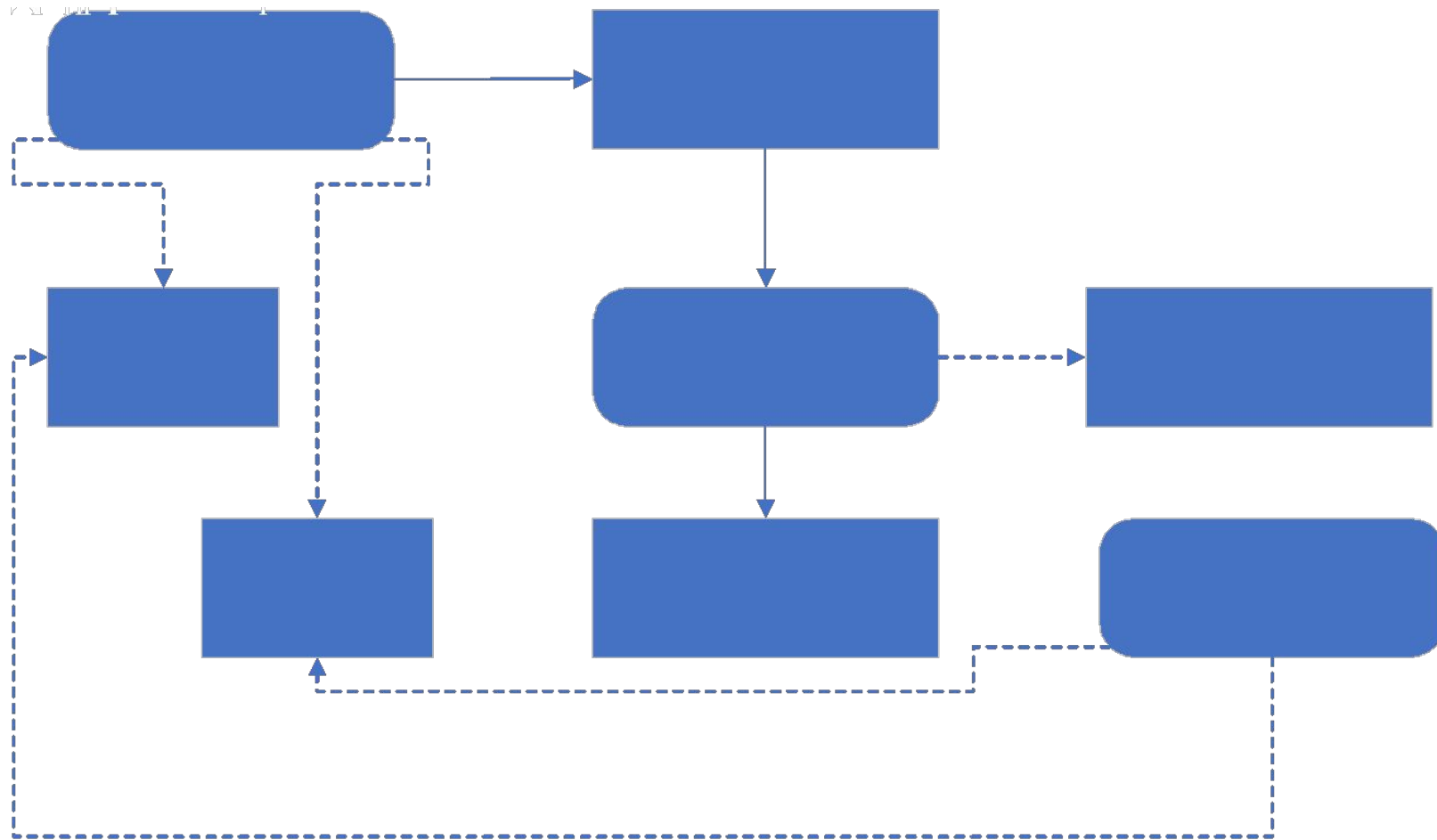
Функция вывода обеспечивает просмотр правил и/или существующих фактов из рабочей памяти и сопоставление этих фактов с базой правил или добавление фактов по мере необходимости.

# Функция управления

Функция управления обеспечивает определение порядка просмотра и применения правил и выполняет следующие функции:

1. Сопоставление образца правила с имеющимися фактами.
2. Выбор наиболее подходящего правила, если их несколько.
3. Срабатывание правила при совпадении образца с фактами.
4. Выполнение правила и запись результата.

# Структурная схема механизма выбора





# Заключение

Фреймовые модели используются в искусственном интеллекте и информационных системах для структурирования и организации знаний.

Фреймы позволяют описывать не только взаимосвязи, но и их структуру.