

## Лекция 8. Вывод на знаниях

**Цель:** Изучение способов вывода на знаниях

**План:**

1. Машина вывода (интерпретатор правил)
2. Управляющий компонент вывода
3. Управление выводом
4. Прямой и обратный выводы

Наибольшее распространение получила продукционная модель представления знаний. При ее использовании база знаний состоит из набора правил, а программа, управляющая перебором правил, называется *машиной вывода*.

### *Определение*

**Машина вывода (интерпретатор правил) — это программа, имитирующая логический вывод эксперта, пользующегося данной продукционной базой знаний для интерпретации поступивших в систему данных.**

Обычно она выполняет две функции:

- просмотр существующих данных (фактов) из рабочей памяти (базы данных) и правил из базы знаний и добавление (по мере возможности) в рабочую память новых фактов;

- определение порядка просмотра и применения правил. Этот механизм управляет процессом консультации, сохраняя для пользователя информацию о полученных заключениях, и запрашивает у него информацию, когда для срабатывания очередного правила в рабочей памяти оказывается недостаточно данных [Осуга, Саэки, 1990].

В подавляющем большинстве систем, основанных на знаниях, механизм вывода представляет собой небольшую по объему программу и

включает два компонента — один реализует собственно вывод, другой управляет этим процессом.

Действие *компонента вывода* основано на применении правила, называемого *modus ponens*: "Если известно, что истинно утверждение А, и существует правило вида "ЕСЛИ А, ТО В", тогда утверждение В также истинно".

Таким образом, правила срабатывают, когда находятся факты, удовлетворяющие их левой части: если истинна посылка, то должно быть истинно и заключение.

Компонент вывода должен функционировать даже при недостатке информации. Полученное решение может и не быть точным, однако система не должна останавливаться из-за того, что отсутствует какая-либо часть входной информации.

*Управляющий компонент* определяет порядок применения правил и выполняет четыре функции:

1. *Сопоставление* — образец правила сопоставляется с имеющимися фактами.

2. *Выбор* — если в конкретной ситуации могут быть применены сразу несколько правил, то из них выбирается одно, наиболее подходящее по заданному критерию (разрешение конфликта).

3. *Срабатывание* — если образец правила при сопоставлении совпал с какими-либо фактами из рабочей памяти, то правило срабатывает.

4. *Действие* — рабочая память подвергается изменению путем добавления в нее заключения сработавшего правила. Если в правой части правила содержится указание на какое-либо действие, то оно выполняется (как, например, в системах обеспечения безопасности информации).

Интерпретатор продукций работает циклически. В каждом цикле он просматривает все правила, чтобы выявить те, посылки которых совпадают с известными на данный момент фактами из рабочей памяти. После выбора

правило срабатывает, его заключение заносится в рабочую память, и затем цикл повторяется сначала.

В одном цикле может сработать только одно правило. Если несколько правил успешно сопоставлены с фактами, то интерпретатор производит выбор по определенному критерию единственного правила, которое срабатывает в данном цикле. Цикл работы интерпретатора схематически представлен на рис. 4.

Информация из рабочей памяти последовательно сопоставляется с посылками правил для выявления успешного сопоставления. Совокупность отобранных правил составляет так называемое *конфликтное множество*. Для разрешения конфликта интерпретатор имеет критерий, с помощью которого он выбирает единственное правило, после чего оно срабатывает. Это выражается в занесении фактов, образующих заключение правила, в рабочую память или в изменении критерия выбора конфликтующих правил. Если же в заключение правила входит название какого-нибудь действия, то оно выполняется.

Работа машины вывода зависит только от состояния рабочей памяти и от состава базы знаний. На практике обычно учитывается история работы, т. е. поведение механизма вывода в предшествующих циклах. Информация о поведении механизма вывода запоминается в памяти состояний (рис. 5). Обычно память состояний содержит протокол системы.

### **Управление выводом**

От выбранного метода поиска, т. е. стратегии вывода, будет зависеть порядок применения и срабатывания правил. Процедура выбора сводится к определению направления поиска и способа его осуществления. Процедуры, реализующие поиск, обычно "зашиты" в механизм вывода, поэтому в большинстве систем инженеры знаний не имеют к ним доступа и, следовательно, не могут в них ничего изменять по своему желанию.

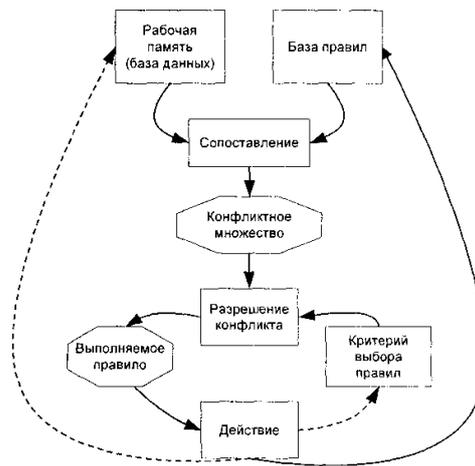


Рис. 4. Цикл работы интерпретатора

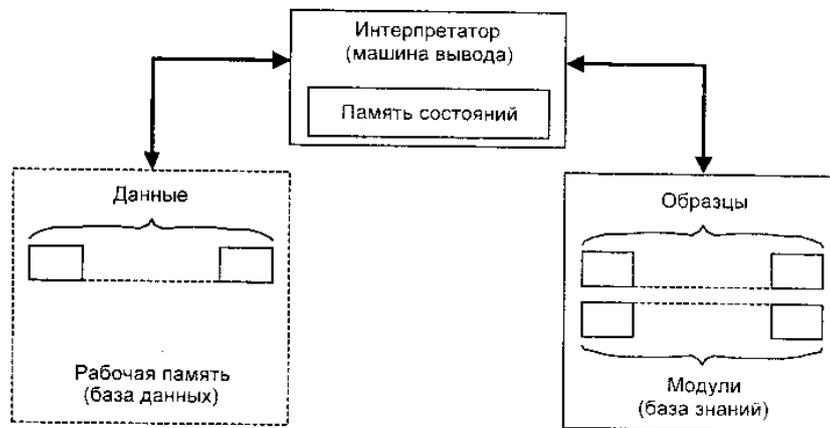


Рис.5.Схема функционирования интерпретатора

При разработке стратегии управления выводом важны:

- исходная точка в пространстве состояний. От выбора этой точки зависит и метод осуществления поиска — в прямом или в обратном направлении.
- метод и стратегия перебора — в глубину, в ширину, по подзадачам или иначе.

При *обратном* порядке вывода вначале выдвигается некоторая гипотеза, а затем механизм вывода как бы возвращается назад, переходя к фактам, пытаясь найти те, которые подтверждают гипотезу (рис. 6, правая часть). Если она оказалась правильной, то выбирается следующая гипотеза, детализирующая первую и являющаяся по отношению к ней подцелью. Далее отыскиваются факты, подтверждающие истинность подчиненной гипотезы.

Вывод такого типа называется управляемым целями, или управляемым консеквентами. Обратный поиск применяется в тех случаях, когда цели известны и их сравнительно немного.

В системах с *прямым выводом* по известным фактам отыскивается заключение, которое из этих фактов следует (см. рис. 6, левая часть). Если такое заключение удастся найти, то оно заносится в рабочую память. Прямой вывод часто называют выводом, управляемым данными, или выводом, управляемым антецедентами.

Существуют системы, в которых вывод основывается на сочетании упомянутых выше методов — обратного и ограниченного прямого. Такой комбинированный метод получил название циклического.

Пусть имеется фрагмент базы знаний из двух правил:

- П1: Если "отдых — летом" и "человек — активный", то "ехать в горы".
- П2: Если "любит солнце", то "отдых летом".

Предположим, в систему поступили факты — "человек активный" и "любит солнце".

*ПРЯМОЙ ВЫВОД*— исходя из фактических данных, получить рекомендацию.

**- 1-й проход.**

- *Шаг 1.* Пробуем /7/, не работает (не хватает данных "отдых — летом").
- *Шаг 2.* Пробуем /72, работает, в базу поступает факт "отдых — летом".

**- 2-й проход.**

- *Шаг 3.* Пробуем Я/, работает, активизируется цель "ехать в горы", которая и выступает как совет, который дает ЭС.

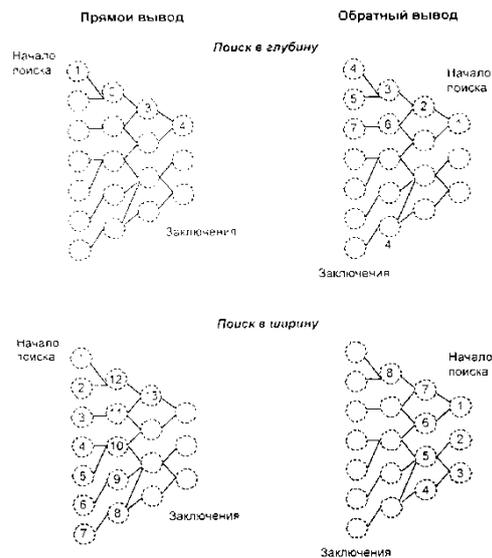


Рис. 1.7. Стратегии вывода

## Рис. 6. Стратегии вывода

**ОБРАТНЫЙ ВЫВОД**— подтвердить выбранную цель при помощи имеющихся правил и данных.

### - 1-й проход.

- Шаг 1. Цель — "ехать в горы": пробуем  $П1$  — данных "отдых — летом"

нет, они становятся новой целью и ищется правило, где она в левой части.

- Шаг 2. Цель "отдых — летом": правило  $П2$  подтверждает цель и активирует ее.

визурует ее.

### - 2-й проход.

- Шаг 3. Пробуем  $П1$ , подтверждается искомая цель.

### Контрольные вопросы:

1. Дайте определение термина «машина вывода»
2. Какие функции выполняет управляющий компонент?
3. Опишите цикл работы интерпретатора.

### Литература:

1. А. П. Частиков Т. А. Гаврилова Д. Л.Белов. РАЗРАБОТКА  
ЭКСПЕРТНЫХ СИСТЕМ. СРЕДА CLIPS. СПб: БХВ-  
Петербург, 2003

**Ключевые слова:**

Машина вывода, интерпретатор правил, управление выводом.