Production Systems

Notes from Bruce R. Maxim
UM-Dearborn

Rule-Based Systems

- Also known as "production systems" or "expert systems"
- Rule-based systems are one of the most successful AI paradigms
- Used for synthesis (construction) type systems
- Also used for analysis (diagnostic or classification) type systems

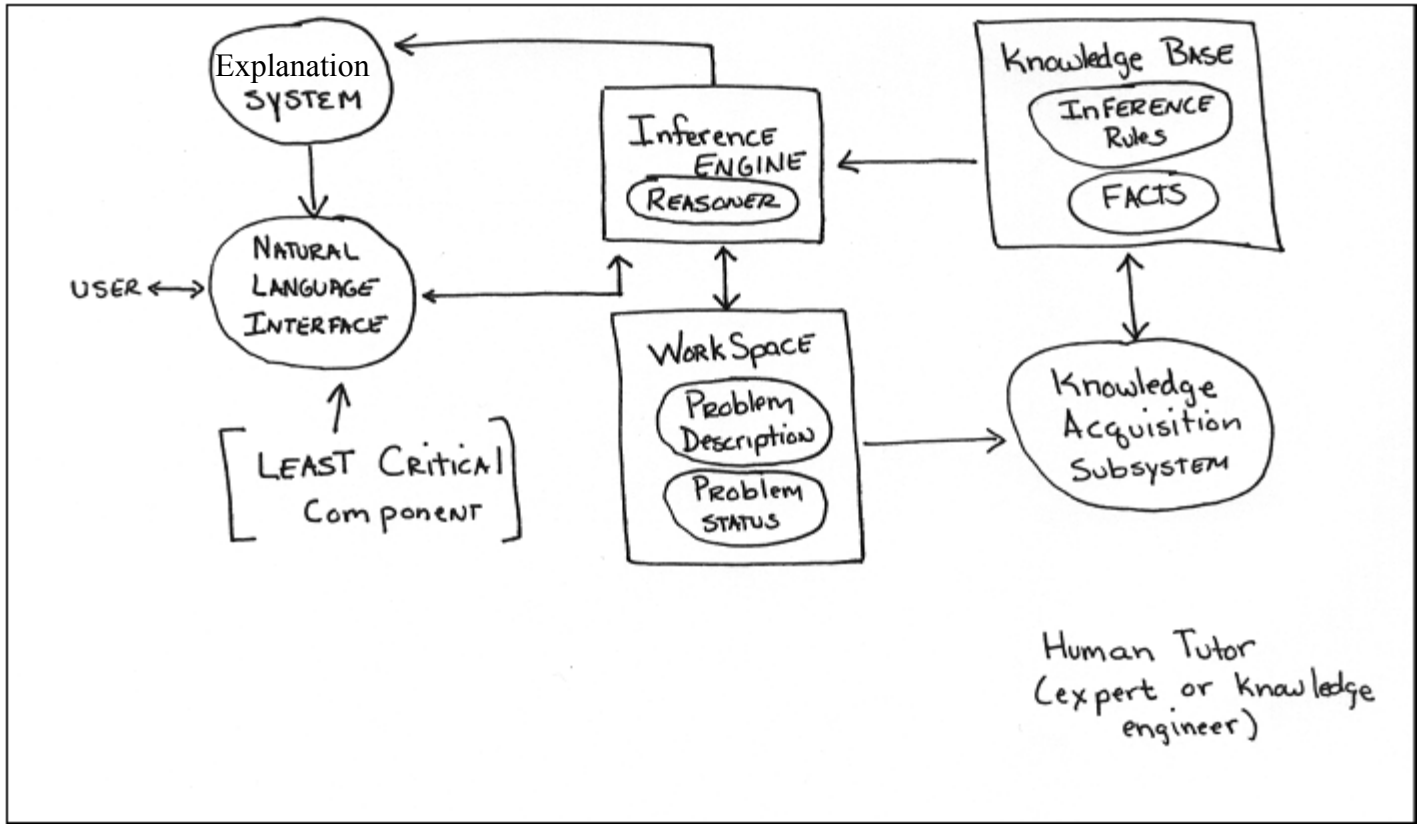
Rule Format

Label R_n if *condition 1*
condition2
…
then
action 1
action2
…

Generic System Components

- Global Database
  - content of working memory (WM)
- Production Rules
  - knowledge-base for the system
- Inference Engine
  - rule interpreter and control subsystem

Expert System Architecture



Forward Chaining Procedure

- Do until problem is solved or no antecedents match

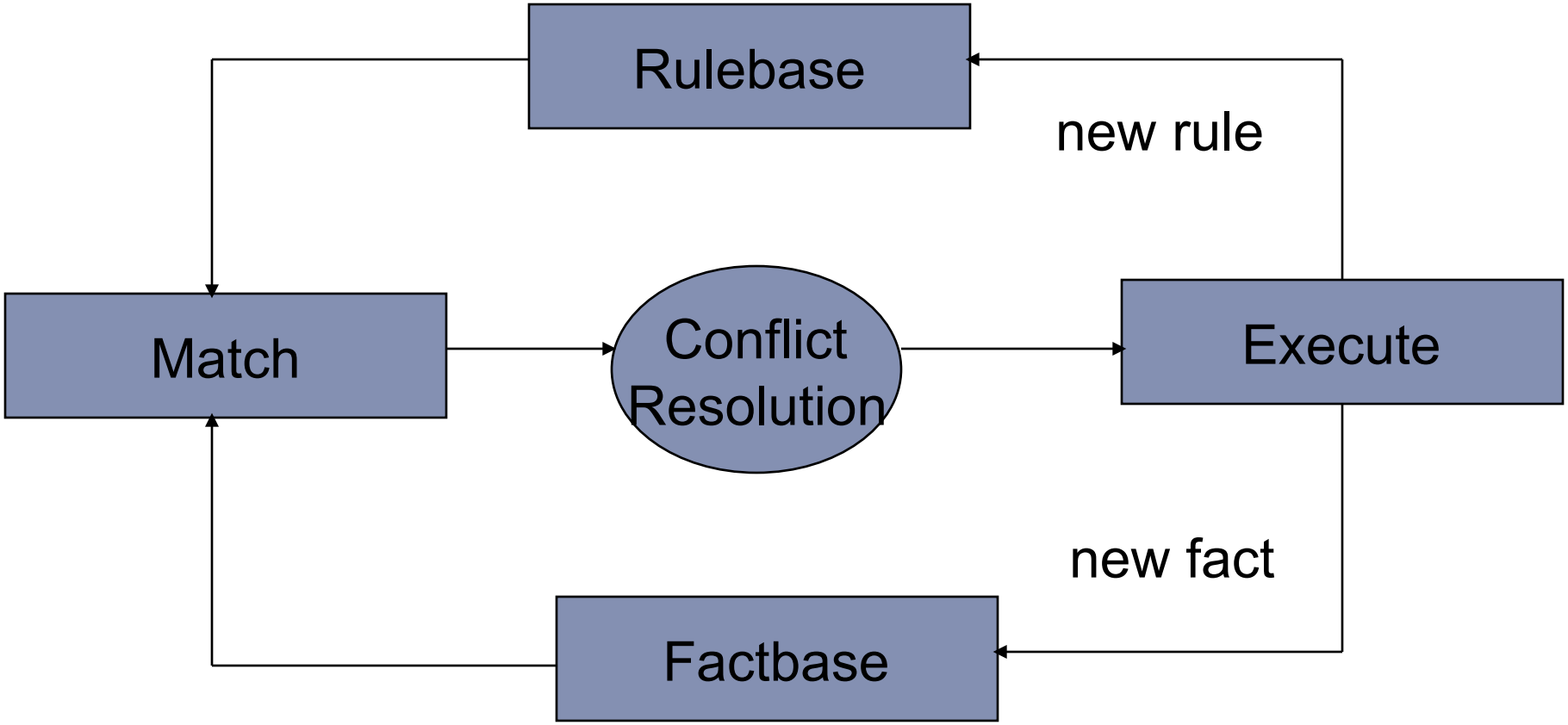
  Collect the rules whose antecedents are found in WM.

  If more than one rule matches

    use conflict resolution strategy to eliminate all but one

  Do actions indicated in by rule "fired"

Inference Engine



Conflict Resolution Strategies

- Specificity or Maximum Specificity
  - based on number of antecedents matching
  - choose the one with the most matches
- Physically order the rules
  - hard to add rules to these systems
- Data ordering
  - arrange problem elements in priority queue
  - use rule dealing with highest priority elements
- Recency Ordering
  - Data (based on order facts added to WM)
  - Rules (based on rule firings)

Bagger

An expert system to bag groceries

1. Check order to see if customer has forgotten something.
2. Bag large items with special attention to bagging big bottles first.
3. Bag medium items with special handling of frozen foods.
4. Bag small items putting them wherever there is room.

Bagger

- For set of rules see the handout
- The conflict resolution strategy
  - Maximum specificity

    (can be simulated by careful rule ordering)
  - Context Limiting

    (needs to set and evaluate context variable)

- **Rule B1**
  IF step is check-order
  there is bag of potato chips
  there is no soft drink bottle
  THEN add one bottle of Pepsi to order
- **Rule B2**
  IF step is check-order
  THEN discontinue check-order-step
  start bag-large-items step
- **Rule B3**
  IF step is bag-large-items
  there is large item to be bagged
  there is large bottle to be bagged
  there is bag with less than 6 large items
  THEN put large item in bag

match: B1 & B2
select: B1 (specificity)
act: fire B1

If B2 matches and not B1, we only fire B2, which jumps us to "bag-large-item"

- **Rule B4**

  IF step is bag-large-items

  there is large item to be bagged

  there is bag with less than 6 large items

  THEN put large item in bag

- **Rule B5**

  IF step is bag-large-items

  there is large item to be bagged

  THEN start fresh bag

- **Rule B6**

  IF step is bag-large-items

  THEN discontinue bag-large-items

  start bag-medium-items step

match: B4 & B5 & B6
select: B4 (specificity)
act: fire B5

May fire multiple times, for all these kinds of bags

Only fire B5 is there are no more B4 stuff to do

Only fire B6 is there are no more bag-large-item things to do

- Rule B7

  IF step is bag-medium-items
  there is medium item to be bagged
  there is empty bag or bag with medium items
  bag is not yet full
  medium item is frozen
  medium item is not in freezer bag
  THEN put medium item in freezer bag

- Rule B8

  IF step is bag-medium-items
  there is medium item to be bagged
  there is empty bag or bag with medium items
  bag is not yet full
  THEN put medium item in bag

- Rule B9

  IF step is bag-medium-items

  there is medium item to be bagged

  THEN start fresh bag

- Rule B10

  IF step is bag-medium-items

  THEN discontinue bag-medium-items

- Rule B11

  IF step is bag-small-items

  there is small item to be bagged

  there is bag that is not yet full

  bag does not contain bottles

  THEN put small item in bag

- Rule B12

  IF step is bag-small-items

  there is small item to be bagged

  there is bag that is not yet full

  THEN put small item in bag
- Rule B13

  IF step is bag-small-items

  there is small item to be bagged

  THEN start fresh bag
- Rule B14

  IF step is bag-small-items

  THEN discontinue bag-small-items

  stop

R1/XCON

- Rule-based system developed by DEC and CMU to configure Vax computers
- Input is customer order
- Output is corrected order with diagrams showing component layout and wiring suggestions
- Does in minutes what used to take humans days and has a much lower error rate

R1/XCON

- Similar to Bagger in that it is a forward chaining expert system
- Makes use of the maximum specificity and the context limiting conflict resolution strategies
- Rules written using OPS5 a rule-based language developed for this project

R1/XCON Stages

1. Check order for missing/ mismatched pieces
2. Layout processor cabinets
3. Put boxes in input/output cabinets and put components in boxes
4. Put panels in input/output cabinets
5. Layout floor plan
6. Indicate cabling

R1/XCON Rule (Pseudocode)

X1 if context is layout and
you are assigning power supply
then
add appropriate power supply

Advantages of rules

- Incremental specification development
  - New rule? Just throw it in
- Supports end-user programming
  - Simple definition of support environments
    - E.g. look for condition facts conditions, never made by actions
- Exception-driven programming
  - One more case? Just add it
- How to jump control, anytime, for special events
  - If rule triggering favors most recent assertions…
  - … then at anytime, assert an error condition …
  - … and watch the special case rules fire
- Adaptive systems
  - Rules= simple regular structures
  - Excellent candidates for learning algorithms

For more, see “Jess” or “Clips”