PROBLEMS WITH XCON

- MATCH:
  Each rule must include the full description of its triggering situation.

  1989
  Avg # of conditions per rule = 6
  Avg # of attributes/condition = 5
  ie. 30 tests for applicability

- A problem:
  How do we know, when we add a rule, if that situation is one already described, or if it is completely specified?
• No clear guidelines.

• When writing a rule how is it "bounded" to relevant situations?

• When faced with more than one relevant rule, how should the "right" relevant rule be selected?

• Apart from contexts, the approach to rule writing varies — there are many authors.
PROPOSED SOLUTION

• Give more structure by providing a set of roles for knowledge.

  eg.
  • Initialize state
  • Recognize success
  • Recognize failure
  • Propose operator
  • Reject operator
  • Vote for operator
  • Apply operator

• By adding this structure to contexts (problem spaces) it provides more direction to those adding knowledge.
OTHER WORK & IDEAS

• R1-Soar: An implementation of a portion of R1 using Soar, a general p-s architecture that uses heuristic search in problem-spaces. 1984

• Proto-R2: Uses Evaluate-goal, Propose-operator, Evaluate-operator, Apply-operator in each problem-space. 1985

• SEAR: A knowledge acquisition tool for constructive systems. Produces knowledge in Proto-R2 form. 1985

• RIME: A software engineering methodology for writing XCON. 1986?
Rule Name: R1a-unmounted-ubx-options

IF

C1 If the current step in the configuration process involves mounting options in containers;

C2 and the system being configured is not a vax11/780, vax11/782, vax11/785, vax6650, or vax6600;

C3 and there is no unconfigured disk which sits on the IDC bus;

C4 and there is an unconfigured r102-type disk which needs to mount inside a cabinet and whose pre-assigned controller sits on a unibus and it is the first disk assigned;

C5 and there is no unconfigured r102-type disk assigned to a controller that is placed closer to the cpu than the controller assigned to the aforementioned disk;

C6 and there is a requirement to cable the disk to be configured to a controller;

C7 and there has been no connection made between the disk to be configured and anything else;

C8 and there is a controller to which the disk to be configured has been pre-assigned and which sits on a unibus;

C9 and there is a requirement to cable the controller to a disk whose type and quantity of cable match one of the possibilities specified for the disk;

C10 and there has been no connection created yet to this controller from any disk;

C11 and there are no unused disk spaces in any unibus cabinet;

C12 and there is a description for the capacity of a disk cabinet, whose name is not "h9643";

C13 and there is an unconfigured disk cabinet;

C14 and the top space available for disk placement is unused;

THEN

A1 mark the disk configured;

A2 and update the top space in the cabinet to be used;

A3 and create a connection relationship between the disk and its controller, fully specifying the identifying information for the disk, controller, cabinet, and the type and quantity of the cable to be used for the connection;

A4 and create a containing relationship between the disk and the cabinet, specifying the identifying information for the disk and cabinet as well as the location of the placement;

A5 within the cabinet for both the skyline view of the cabinet layout and the detailed view of the particular cabinet.
XCON was reimplemented using the RIME approach.
In production use since Jan. 1988.

Specifying Algorithmic Control
- use rules in fixed sequence
- specify pre-defined algorithm at development time.
  Made up of steps.
  Made up of rules.
  All satisfied rules in a step are used, before going to next step.
- reduces LHS
Explicit Choice among Alternatives

- Break out types of tasks.
  - eg. selecting device
    selecting container
    selecting location
  - Explicit control of these via rules

- i.e. task specific control,
  not task independent control (as provided by OPS5)
Rule Name: Configure-device:propose:200a:select-device

IF

C1 The current step of the process of configuring devices involves proposing alternatives;

C2 and no device has yet been chosen;

C3 and there is at least one unconfigured device;

C4 and the process of selecting a device has not yet been proposed;

C5 and no problem has been identified concerning selecting a device;

THEN

A1 Propose to go through the process of selecting a device.
Rule Name: Configure-device:eliminate:200n:prefer-select-container

IF

C1 The current step of the process of configuring devices involves arbitrating between alternatives by pairwise elimination;

C2 and the process of selecting a container has been proposed as an alternative;

C3 and the process of selecting a device has been proposed as an alternative;

C4 and the current goal of the process of configuring devices is to fill a container with devices;

THEN

A1 Eliminate selecting a device from consideration.
- In XCON it had become harder & harder to put related rules next to each other.

- RIME rules are less complex, and relate to fewer other rules.

- Grouping criteria are recorded in "subgroup schemas".
  - Grouped according to $f^n$.
  - Rule names include number that refers to part of schema.
  - Rule-naming rules enforced.
RIME - BENEFITS

- Smaller rules
- About same number as now many more general rules.
- Easier to add new rules.
- Number of rule firings increased slightly.
- Processing time didn't increase much, as RETE network less complex.
- Time/rule was about 10 times less.
- Avg cpu time per config on VAX 8800 with 128 MB in < 60 seconds.