Moving the condition and operator attributes to the level of the case step

The choices (decision options) of a choice step are usually not defined for a single variable (e.g., one variable can be renal_failure, another can be diabetes, etc.). So a question comes into mind whether the condition (of type expression) attribute of Decision Step, and also the operator attribute, should be moved to the Case_Step.

Following are Samson's arguments:

1) Type-mismatch problem: A Choice_Step has a set of options (of class Decision_Option). A Decision_Option has a condition_value, which can be a RuleInChoice, which has a rule-in criteria associated with it. The condition expression and operator won't work with RuleInChoice. They only work with Decision_Condition of Case_Destination (evaluating a comparison like "condition operator case_value").

My comment: A rule-in is a criterion that is evaluated to true, false, or unknown. In the example that I used, the types do match, since the expression is assigned with a criterion that also is evaluated to true, false, or unknown. But, it was a special case that I used, where the choices were mutually exclusive, so only one choice evaluates to true at any given circumstances. Comparing a choice step's condition to a rule-in forces us to have a certain criterion type for the condition.

Another comment I have is that Samson convinced me that in the example that I used, the top level decision should be modeled as a case step and not as a choice step. Ronilda also suggested that, but I argued that since the details of the top-level decision include a choice, then the top-level decision should be modeled as a choice and not as a case. Samson argued that after the subguideline (the details of the top-level decision) is executed, then the decision is automatic and should therefore be modeled as a case step and not as a choice step.

2) The rule-in criterion in the RuleInChoice can still use whatever variables whose value are defined in the decision_detail subguideline. I don't think you should need to have condition and operator attributes to referece these variables. Using condition and operator restricts the kind of comparison you can make, and I think that's not appropriate for Choice_Step.

My comment: I agree. Like I said above, having the condition as an attribute of a decision step forces the type of the condition to be a certain criterion.

I also agree that we shouldn't need to have condition and operator in order to reference variables of subguidelines and vice versa. Qing will present an example of local parameters and parameter passing between a guideline and a subguideline in the meeting of May 30.