

P1450.4 meeting minutes - 10/22/03

Attendees: Jim O'Reilly, Doug Sprague, Don Organ, Eric Nguyen, Jim Mosely, Jose Santiago, Ernie Wahl,
Not present: Yuhai Ma, Dave Dowding

Agenda: Review Dave's latest diagrams (sent out 10/10/03), and Yuhai's email. Get closure on concepts from flow diagrams.

Continued discussion on the concepts of flow-nodes, subflows, tests, and flows, and how each of these entities is related to each of the other entities.

Ernie mentioned that many (most?) of his concepts of flow-nodes, flows, tests, subflows, etc., are constructed around what his view of what tester GUI languages are capable of today.

Tentative consensus was (I think) reached on the following points:

- A subflow will have one input and one exit path. This will allow a subflow to substitute for a test in the body (module) of a flow-node. (An alternative was to allow subflows to have more than one exit path - but if one wants to use a subflow as a substitute for a test, then that subflow is constrained to have only one exit path).
- One implication of this decision is that a subflow **MUST** be contained within a flow-node in order to branch to multiple places, depending on what happens inside a subflow (i.e., the execution path that was taken through a subflow). The flow-node post-actions and arbiter will have the responsibility of determining which of the multiple exit-actions/exit paths from a flow-node will be taken. This decision will be based on information such as subflow pass/fail result, the state of other (as yet-unspecified) STIL variables, and/or ???
- While exit from a subflow will only be from one point, the exit paths of any of the flow-nodes which make up the subflow can go to either the input to any another subflow flow-node, or directly to the subflow exit path. Thus, branching within a subflow is allowed, as long as all branches reconverge prior to exiting a subflow.
- We discussed Yuhai's email, outlining his thoughts about multiple exit-paths from flow nodes. Yuhai seems to be arguing for a single exit-path from a flow-node, with the decision about where to go next under various conditions (i.e., pass/fail, etc.) being handled elsewhere. After all, in order for a flow to branch (which I think we all agree is essential!), it must be possible, upon exiting from a flow-node, to have a choice about where to go next. This idea has been embodied in the concept of the arbiter and multiple exit paths from a flow-node. In Yuhai's parlance, the common post-actions are the exit door to the aircraft, and the choice of what to do and where to go next (baggage claim, rental car, parking lot) is the arbiter and multiple exit paths.

I think we probably need to discuss this some more, but to me at least, it seemed that the concept of a flow-node having multiple exit paths was fairly well established and accepted.

- I also felt that we had tentative agreement that the flow would end when it reached a terminal flow-node, rather than having "stop executing the flow" as one of the possible exit actions. In this scenario, a terminal flow-node is a special type of flow-node having only an input and no output (just as a predefined flow entry point - such as BEGIN, RESET, INITIALIZE, etc.) is a special type of flow-node having only an output but no input). However, I think that my feeling that we had tentative agreement on this point is less solid than some of the above points - so this point may be revisited.

That's about all I can remember for now - let's pick this up again on Wednesday.

Jim