P1450.4 meeting minutes - 11/05/03

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

Agenda:

- Quick review of meeting notes.
- Review emails from Ernie, Jim describing proposed sets of actions (pre/post/exit)
- Review Dave's proposed definitions of flow-node elements.
- From Summary section of last week's meeting, can we agree on how the flow elements will be organized (if so, then diagrams will need to be updated).
- At end of meeting, identify areas/issue on which a decision has been reached (closed issues) and areas which are still open.
- Set agenda topics for next meeting.

SUMMARY: Continued discussion about the interrelationship of flow-nodes, flow-node components, tests, test methods, subflows, single vs. multiple exit paths, and the types of actions allowed in pre-, post-, and exit-action blocks.

Three major issues were the focus of discussion in today's call:

1. Hierarchy of flow components (flow-node, test, and test-method/subflow). I'm not sure we still have definitive agreement on the hierarchy of flow components.
   Is it: flow-node -> flow-node module (body)->test (pointed to by flow-node body)->test method/ subflow? (i.e., is the body of a flow-node a test, which can be either a test method or a subflow?).
   Or is it: flow-node -> flow-node module (body)->test method/subflow?(i.e., the body of a flow-node directly calls a previously-defined test method or a subflow.

2. Should a subflow and a test have their own explicit pre/post actions, arbiter, and exit actions? Or should those components rely on the pre/post actions, arbiter, and exit-actions of their parent flow-node? Note that should a test method (being called from a test, which is called from a flow-node module) encapsulate any needed pre/post/exit actions and arbiter?

3. A subflow can be used as the body of a flow-node. What happens if that subflow contains a terminal node (either as one of the exit-actions from any of the subflow's flow-nodes, or as a terminal flow-node). This issue dealt with the issue of how one exits from a subflow - can a subflow contain an immediate stop, or must a subflow exit via its post-actions/arbiter/exit-actions and the post-actions/arbiter/exit-actions of it's parent flow-node? We have not yet defined (and perhaps don't need to define, for a standard which describes the flow, rather than an implementation of that standard) what happens at a terminal node. It doesn't matter whether the terminal point is a particular type of flow-node (i.e., a flow-node with an input and no output) or a "stop" action in the action list of an exit-action block. But in any case, the question remains - when you reach a stop node or action, is it an immediate stop (i.e., via a jump - a longjmp? - to some exit point outside the boundaries of the flow).

For #3, it's clear to me that me MUST allow for the possibility of a subflow containing an immediate stop (regardless of whether we define it in our syntax as a stop-action in an exit-action list, or as a terminal flow-node). I (and I think others in the group) see it as too restrictive to force a subflow to return through its post-actions, arbiter, and exit-actions before terminating the flow. Are we in agreement on this point?

For #2, I think that there are two views of what goes into the body of a flow-node. The first is that the body of a flow-node is an entity called "test" (what's labelled "TestMethod Module" in the figure below), which has its own pre/post/exit actions and arbiter, and whose body can be a testmethod call (a test method of type subflow, in this case). In this figure, the "TestMethod Module" forms the body of a flow-node. When executing a flow-node, the execution path goes through the following:

Flow-node pre-actions
TestMethod Module pre-actions
TestMethod
TestMethod Post-actions
TestMethod arbiter
TestMethod exit (pass or fail) actions
Flow-node post-actions
Flow-node arbiter
Flow-node exit actions.

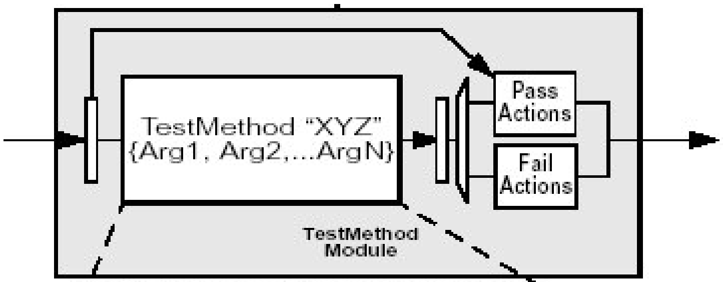

Figure 2A: TestMethod Module

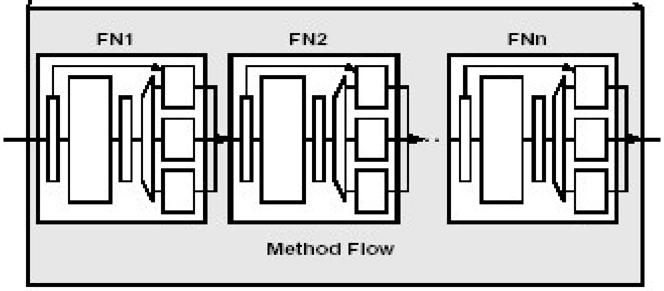

Figure 2B: Method Flow Example

And if the TestMethod itself is a subflow, then the test method will go through a similar sequence all over again!!

The second view is that the body of a flow-node directly calls a test method or subflow, without the intermediate layer of "test" (or "TestMethod Module"). See the "other" Fig. 2 on the next page. In this case, the execution path goes through the following:

Flow-node pre-actions
TestMethod
Flow-node post-actions
Flow-node arbiter
Flow-node exit actions.

As in the previous case, if the TestMethod itself is a subflow, then the test method will go through a similar sequence - but there are less overall steps!

So, the question is, which of these two models makes more sense? The second approach is conceptually a bit simpler - and if one WANTED to have pre- and post-actions, etc. for the test method, one could simply build them into the test method.

#1 relates to what I described above - we need to decide what our component hierarchy will be. Will the flow-node body reference a test, which then references a test method or subflow? Or will the flow-node body directly reference a test method or subflow?

So, for Wednesday, I propose the following:

- Decide on our flow-component hierarchy (do we want the extra layer of "test" or "TestMethod Module"?) Please give this some thought before the meeting!
- Let's also give some thought to the definitions of the various components (see Dave Dowding's email from 11/05/03).

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

Jim

Figure 2: Variations on What a FlowNode Module Is

Figure 2A: TestMethod Module

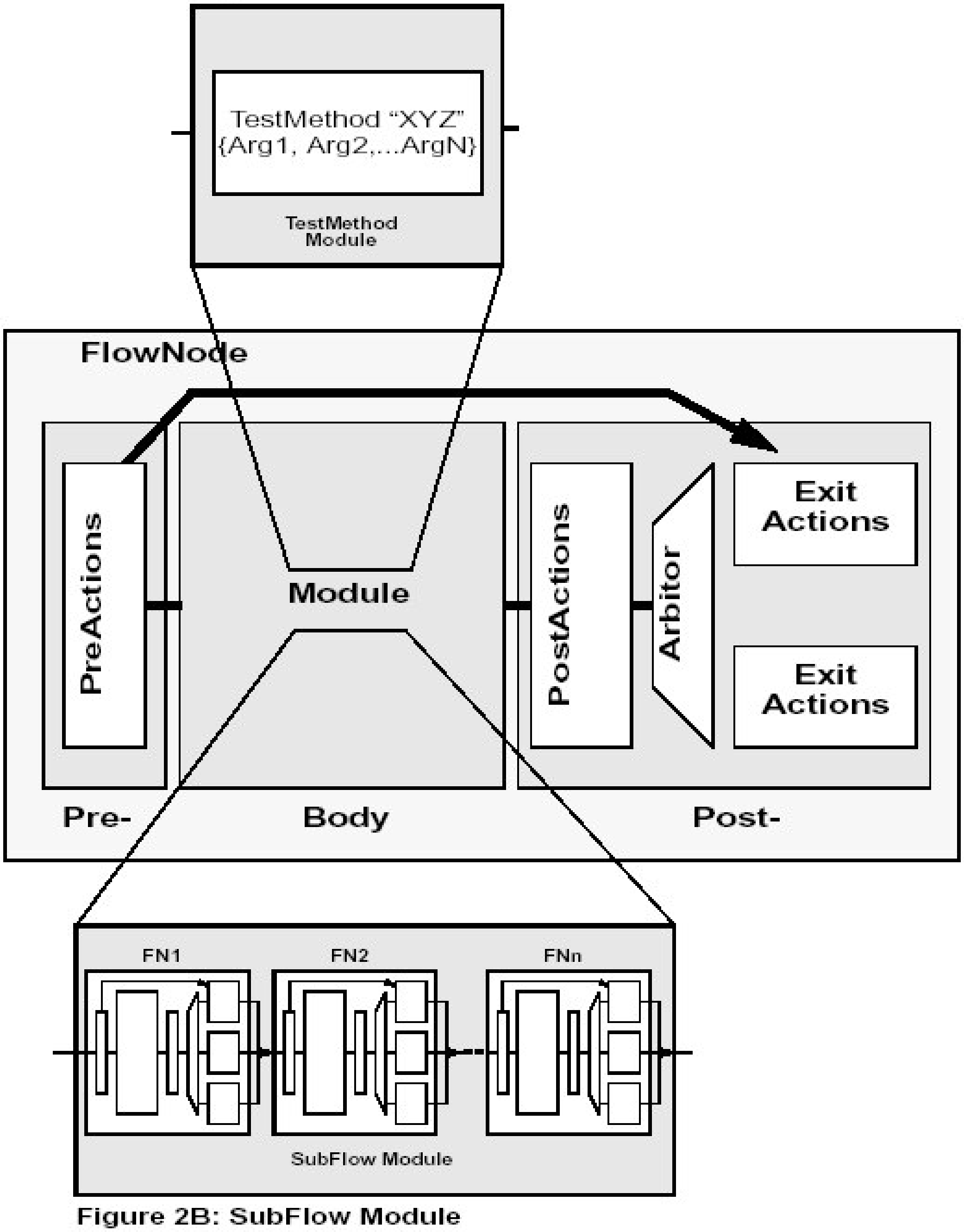

Figure 2B: SubFlow Module