P1450.4 meeting minutes - 02/11/04

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

Agenda:

- Review latest flow diagram documents
- Face-to-face meeting
- STC/STIL collaboration update
- Follow-up questions/discussion from Stylus demo
- Progress report on STC/STIL .4 collaboration

Progress on STC/STIL.4 collaboration
Update from Tom Micek: No response yet (via Paul Roddy, STC chair) from STC legal folks on the collaboration. Expect no problems in getting approval, however.

PAR status:
Update from Tony: PAR P1450.4 resubmitted, approved by Rochit Kapur (TTTC chair), and forwarded to IEEE. Approval of PAR is expected at the April (?) IEEE meeting - no difficulties are expected.

Face-to-face meeting:
Tentative date for a face-to-face is March 16-17, 2004. Location to be determined (will be in Bay Area).

Follow-up questions/discussion from Stylus demo:
Did we get to this? I think Don mentioned that it would take him several weeks to put together syntax documents - and it's not one of his top priorities right now. As I recall, most everyone agreed that we'd be well served by using Stylus as a starting point, or at least incorporating into STIL .4 many of its concepts and ideas. I'm not sure how much farther than that the discussion went, though.

Review latest flow diagram documents:
Review Dave's latest flow document (TestProgramFlowWorkingDiagramsB.pdf, dated 2/11/2004) - emailed to WG members on that date (not yet posted on the web site).

Comments: Page 1: Diagram entitled "Diagram of flow control constructs within STIL": In addition to 1450.0 constructs/defs, need to include 1450.2 constructs/defs, as well as any other STIL extensions whose contents we refer to (i.e., .6, .1).

Page 1: Some discussion of use of Environment block to wrap Test Program block. Not sure if this will prove to be necessary - expect additional discussion about that in upcoming meetings.

Page 2: General agreement that this diagram is an accurate representation of the Test Program construct. The only comment I (Jim) have after the fact is that the diagram seems to indicate that the Test Program will contain various flow definitions - which the diagram indicates (implicitly) are invoked by the entry points. However, as I see it, the flows will likely be defined outside the Test Program construct - and per the definitions of terms on page 3, entry points don't call flows, they call TestModules, which call the flows.

Page 3. Complete reworking of Terms definitions. General agreement on both the list of terms (which will, of course, continue to grow) and the wording of each term's definitions. In particular, the addition of flow node types task-node and decision-node (further subdividing flow-node types into more specialized types) was welcomed.

Page 4 - no comments - diagram unchanged.

Page 5. Fig. 2. The "Flow-node moduleRef referencing a TestModule" diagram (where the Test Module calls a VOH test in this example), is now seen as correct. Note the instantiation of test methods has been expanded to include in-line (unnamed) instantiation as well as define-before-use (named) instantiation. Of course, to allow a test method to be used by more than one flow-node, it must be defined before use (i.e., named).

Page 6: Figure 3: Two “out-flow” Configurations for TestModules". Removed Fig. 3c, which allowed multiple exit paths from a test module, which we've decided not to allow (a flow-node is the construct to use for branching). (See flow diagrams from Jan 14, 2004 (Updated flow diagrams.) p5, Fig. 3 to see the variation which has been eliminated)

General agreement on Fig. 4's representation of a TestModule which contains a subflow instead of a test method. Figure 3: Two “out-flow” Configurations for TestModules".