P1450.4 meeting minutes - 05/09/07

Attendees: Jim O'Reilly, Bruce Parnas, Ernie Wahl

Not present: Doug Sprague, Ajay Koche, Greg Maston, Brian Johnson, Tony Taylor, Carol Dowding, Daniel Fan, Yuhai Ma, Bob Roberts, Oscar Rodrigues, Jim Mosley, SB Thum, Jose Santiago

Agenda/Summary:

- **Preamble:**
  - Record Meeting (*2)
    - To listen to the meeting recording, do the following:
      - Call the (US) dial-in numbers 1-877-421-0003 (toll free) or 1-770-615-1374 (toll)
      - Enter the passcode code 747464
      - Once dialed in with the proper access code, enter *3 (star 3)
      - Then enter the file number 99087701 for this conference (this number will change each week).
      - Press 1 to listen to the conference.
  - IEEE Meeting Preamble (No discussion of proprietary information).
  - Review decision (and rationale for decision) to allow specification of flow nodes in both flow types and flows (instances of flow types).
    - Flow nodes will likely vary on a per-instance basis.
    - There may be instances (using a flow as a test, for example) where specifying the sequence of flow nodes in the type definition is useful.
    - Useful for test program generators, as well as for generating simple programs simply.
  - Review decision to define a .4 standard flow type; instantiation of flows using this type will NOT require a flow type specification. The benefit of this is that a "standard" flow can be specified syntactically without the overhead of requiring the user to specify the type first, then to instantiate a flow of that type. If the decision holds, we need to add the .4 standard flow definition to the document (probably in the same area where we include the .4 standard TestBase definition).
    - The standard will define a standard flow type (named FlowBase, for instance). It will be illegal for the user to provide a flow type with the same name as the .4-provided standard flow type (this will apply to any STIL-provided test type or flow type).
    - Need to package TestBase, standard flow type definition, default flow node definition into a "defaults" container. This is different from the defaults provided by parameters of user- or system-defined types, and also different from the idea of being able to modify TestBase for the requirements of a particular vendor's SW.
    - Useful for test program generators, as well as for generating simple programs simply.
  - Discuss whether or not to allow specification of preactions, postactions, pass-actions and fail-actions in both flows and tests (instances of flow types and test types) as well as in flow types and test types. If allowed, specification of any of these types of actions in the instance will override and completely replace the actions specified in the type. However, each set of actions is treated independently of all others. For example, in the instantiation of a test, the user might specify the fail-actions in the instance. In this case, those fail actions override and completely replace the fail-actions specified in the type. However, the preactions, postactions, and pass-actions, NOT being specified in the test (instance), will be as specified in the type definition (or from TestBase, if the type definition does not include any of the above action types).
    - This is seen as a useful feature; it will follow the same rules as inheritance.

Actions:
For next meeting, update syntax document based on ideas from recent meetings, and provide some code examples.

Next meeting:

- Next meeting 05/23/2007 (skipping next week's meeting on 5/16/07).

For reference STIL .4 information can be found at the IEEE STIL website:
http://grouper.ieee.org/groups/1450/ (select the P1450.4 link from the table) or use the direct link
http://grouper.ieee.org/groups/1450/dot4/index.html