P1450.4 meeting minutes - 01/03/06

Attendees: Doug Sprague, Jim O'Reilly, Ernie Wahl, Ajay Koche, Tony Taylor

Not present: Brian Johnson, Bruce Parnas, Jose Santiago, Carol Dowding, Greg Maston, Daniel Fan, Yuhai Ma, Bob Roberts, Oscar Rodrigues, Jim Mosley, SB Thum

Agenda/Summary:

- P**reamble:**
  - 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 06728801 for this conference (this number will change each week).
      - Press 1 to listen to the conference.
  - IEEE Meeting Preamble (No discussion of proprietary information)

- **Status on new members for WG (Jim)**
  - Contacted Inovys. They're interested in having someone participate, but don't yet know who it will be. Expect to have individual(s) identified after Jan. 1, 2007.

- **TestBase, TestMethods, and Inheritance Discussion**
  - More discussion about the proposals for providing a common set of elements for all TestMethods. At the previous meeting, we had tentative agreement on the following:
    - There are two ways to provide the common framework
      - Via inheritance. The common framework TestBase is defined syntactically as an abstract TestMethod. All other TestMethods are directly or indirectly derived from TestBase.
      - Explicitly. If the user chooses NOT to use inheritance, ALL TestMethods must explicitly include the full set of elements which would have been provided by TestBase. Further, if TestBase is updated, those TestMethods must then also be updated as well.
  - However, the above proposal has a limitation – if a user chooses NOT to use inheritance to provide the common set of elements to TestMethods, those elements must be provided **explicitly** in the definition of every TestMethod. It was requested that we develop an approach which would provide defaults automatically, even if inheritance is not used. The reason for the request is to avoid biasing the language of the standard toward requiring the use of an OO approach (that is, maintain a separation of the language of the standard from the implementation or usage of that language.
    - Key issue: Is it our intent to define the standard in an OO approach? Or do we just want to support an OO approach?
  - Inheritance from TestBase vs. defaults.
    - Benefits of defaults – never have a case where the tools will generate an error because something missing.
    - Con – you'll always have something – even if it's not what you want.
    - Changing defaults – can only change values, not types or directions.
    - Requirement – must allow user to define user-defined types
  - Proposal: if not using inheritance, all required attributes for TestMethods must be specified in the TestMethod definition. What if, instead, we define a set of defaults that would apply if inheritance were not used?
    - Why would you want to use inheritance?

        - If not, can't count on expected behavior of test (don't know if common expected elements are present).
        - Tools can be more complicated if Inherit is not explicit.
    - Treating TestFlow as a TestMethod (i.e., TestFlow is the one integral TestMethod defined in P1450.4)
        - Driving reason for TestFlow being a TestMethod is to allow user to create user-defined TestMethods. VOH – test functionally vs. parametrically (where the functional test is a single test, while the parametric test is a sequence of tests).
        - Need further discussion on this.

Next meeting:

- Next Meeting 01/10/2007.

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