1450.4 meeting minutes – 01/28/11

Attendees: Ernie Wahl, Paul Reuter
Not present: Oleg Erlich, Markus Seuring, Jim O'Reilly, Ajay Khoche

Agenda:
- IEEE Meeting Preamble (No discussion of proprietary information).
- Continue discussion of unresolved areas in draft ballot document. Focus on Parameter directionality Semantics (table 8).
- Open issues - are there other open issues that should be considered? A review of the open issues list can guide us here.
  - Issues list:
    http://spreadsheets.google.com/ccc?key=0AoKiPr1I9LY9dF95dkhSTVVqOU5GbWJyWFNhY0JPX0E&hl=en

- Reference documents (If logged into your google account, can edit. If not, can only view.)
  - Namespace resolution examples document:
    http://docs.google.com/Doc?docid=0AYKiPr1I9LY9ZGY4dmNjNTNfMGZkOGJ2bmZy&hl=en
  - Scratchpad spreadsheet: https://spreadsheets0.google.com/ccc?key=tQ93VDnAZ-Cl9RFKpPrPDzw&authkey=COzyro8K&hl=en&authkey=COzyro8K#gid=0
  - Scratchpad "Word" doc: https://docs1.google.com/document/d/1zVu2M8nTJsrm0nFbBhiuM8-YRt4ErYqdy_uSa3x3_T4/edit?authkey=CLrgwrsG#

Summary:
- Worked out from Table 8 (Parameter Directionality Semantics) following up on the recently ratified sentence "Pass by reference/value is determined by data-type". This took us to Table 4 (Data Structure Parameter Types) where we added type PatLocation and recommended the use of several user-settable constraints which will appear in blue in the next syntax doc version.
- Based on Paul's input (to solve a problem which requires branching on evaluation of fail information across one or more signals and one or more vectors), proposed adding type VecWindow. This data type include a SignalGroup and a start and stop vector location. A standard test taking the parameters that make up VecWindow may be sufficient but the VecWindow proposal is there in red to remind us to deal with the issue. We would like the URL of the Google Issues and Resolutions document so we might edit that instead.
- Agreed to move Limits and PatLocation from Table 4 (Data Structure Parameter Types) to Table 5 (New title: Integral and Small Structure Variable and Parameter Types). Then we can describe Table 4 as Parameter types that can be passed via reference (InOut) only. Table 5 then shows types that, as Parameters, can be passed any way we like, and may also be defined as Variables, i.e., Limits and PatLocation can now also be defined as Variables.
- Followed out user-settable constraints, specifically BinSpec's BinType (Pass|Fail)(,Group|Axis|Bin), and agreed that in order for these to be of use, we had to allow for the refinement of constraints at inheritance (contradictions are illegal), as shown in the following example:

```
// =============================================================================
// Narrowed TestBase definition for illustrative purposes
TestBase {
    Parameters {
        InOut BinSpec failBin = None { BinType Fail; }
    }
}
// =============================================================================
// Legal: substitutes new default value as per existing syntax
TestType MyType0 {
    Inherit TestBase {
        failBin = Failed;
    }
}
```

```
// --------------------------------------------------------------------------
// Legal: explicit specification narrows constraint from Fail to Fail Group
TestType MyType1 {
    Inherit TestBase {
        failBin = None { BinType Fail, Group; }
    }
}
// --------------------------------------------------------------------------
// Legal: narrows constraint from Fail to Fail Group and accepts TestBase default
initialization value
TestType MyType2 {
    Inherit TestBase {
        failBin { BinType Fail, Group; }
    }
}
// --------------------------------------------------------------------------
// Legal: narrows constraint from Fail to Fail Group
TestType MyType3 {
    Inherit TestBase {
        failBin = Failed { BinType Group; }
    }
}
// --------------------------------------------------------------------------
// Error: Pass contradicts TestBase def
TestType MyType4 {
    Inherit TestBase {
        failBin = None { BinType Pass, Group; }
    }
}
// --------------------------------------------------------------------------
```

Next meeting: 02/4/11

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