

## Proposal to add Unit\_Model\_Id to IEEE 1212 Config Rom

**Tarl Neustaedter, Sun Microsystems.**

Unit\_Model\_Id problem statement:

OpenFirmware (IEEE 1275, also known as OpenBoot, the firmware used by Sun and several other vendors in place of a PC BIOS), needs to determine the types of devices with enough detail to specify the device driver that the host operating system will later load.

The firmware which does this must be able to fit in the on-board PROMs of PCI plug-in cards, which can be quite small, so we do not wish to specify implementing SBP-2 as a necessary part of probing attached hardware.

The only existing unit-specific identifiers in the config ROM are the Unit\_Spec\_Id and Unit\_Sw\_Version. SBP-2 requires that these two entries be set to 609E/10483, leaving no available generic unit-specific identifiers

SBP-2 does define a command\_set\_spec\_id and command\_set, which further identify the unit, but these are defined among the vendor-specific entries, so cannot be parsed without specific knowledge of the protocol selected by the Unit\_Spec\_Id and Unit\_SW\_Version. Trying to use these values would require specifying something equivalent to:

- Produce a string of the form SSSSS,VVVVV, where SSSSS is the unit\_spec\_id and SSSSS is the unit\_sw\_version.
- Unless SSSSS is equal to 609E and VVVVV is equal to 10483, then produce a string of the form SSSSS,VVVVV,CCCCC,cccc, where CCCCC is the command\_set\_spec\_id and cccc is the command set.
- Unless SSSSS is equal to <xxxx> and VVVVVV is equal to <yyyy>, then produce a string of the form SSSSS,VVVVV,qqqqq,iiiiii, where qqqqq and iiiii are the protocol specific identifiers for the <xxxx><yyyy> protocols, encoded in their own specific entries.
- Unless ... (repeat for each protocol defined)

By defining a 1394-generic (or 1212-generic) unit directory entry, the full description of how to produce a string identifying the unit becomes: Produce a string from the first of the following pairs to be found in the config ROM:

1. Module\_Vendor\_Id, Unit\_Model\_Id
2. Unit\_Spec\_Id, Unit\_SW\_Version
3. Node\_Spec\_Id, Node\_SW\_Version,
4. Node\_Vendor\_Id, Node\_Hw\_Version
5. Module\_Spec\_Id, Module\_SW\_Version
6. Module\_Vendor\_Id, Module\_HW\_Version

## Proposal to enhance ANSI/IEEE Std 1212-1994

### Add paragraph 8.5.6, Unit\_Model\_Id:

#### 8.5.6 Unit\_Model\_Id

(optional)

Identifies the hardware and software model for the unit.

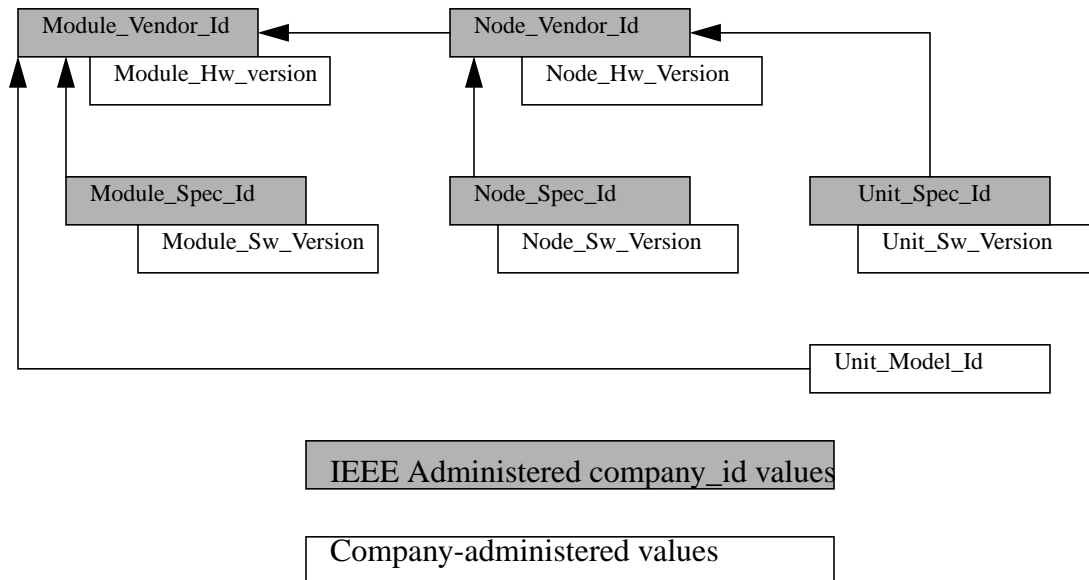
The 24-bit immediate Unit\_Model\_Id value provides a vendor-dependent model number. This Unit\_Model\_Id, when prepended with the Module\_Vendor\_Id, is expected to uniquely identify the unit function, and therefore the appropriate I/O driver software for the unit.

Note that the Module\_Vendor\_Id is normally in the root directory, but if the vendor ID of the entire module is different than the vendor ID of the particular unit, the unit directory may also contain a Module\_Vendor\_Id entry to supersede the entry in the root directory.

### Modify table 35, add entry at bottom of table:

Entry Name	key_types(s)	key_value
Unit_Module_Id	immediate	17

### Modify figure 53, Software Identifier Hierarchy:



## Proposal to modify T10/1155D (SBP-2)

### Modify first paragraph in 7.3 Unit Directory:

Configuration ROM for targets shall contain a unit directory in the format specified by this standard. The unit directory shall contain Unit\_Spec\_ID and Unit\_SW\_Version entries as specified by ISO/IEC 13213;1994, and Management\_Agent and Unit\_Model\_Id entries, as specified by this standard.

### Add paragraph between 7.3.2 Unit\_SW\_Version and 7.3.3 Command\_Set\_Spec\_ID

#### 7.3.2a Unit\_Model\_Id entry

The Unit\_Model\_Id entry is an immediate entry in the unit directory that, in combination with the Module\_Vendor\_Id, specifies the unit function in enough detail to determine the appropriate driver software.



### Add line to sample configuration ROM in appendix B, add inside the unit directory:

