

Subject: P1394> Meeting Minutes from Monterey - July  
=====

#### Meeting Attendees

Takashi Isoda Canon  
Osamu Hirata Canon  
Akihiro Shimura Canon  
Lee Farrell Canon Information Systems  
Greg LeClair Epson (1394 PWG Chairman)  
Fumio Nagasaka Seiko Epson  
Atsushi Uchino Seiko Epson  
Brian Batchelder Hewlett Packard  
Alan Berkema Hewlett Packard  
Scott Bonar Hewlett Packard  
Eric Clement Hewlett Packard  
Laurie Lasslo Hewlett Packard  
Greg Shue Hewlett Packard  
Kwang Kim Hitachi  
Yuji Sasaki Japan Computer Industry  
Brian Nagy Kodak  
Jerry Thrasher Lexmark  
Don Wright Lexmark  
John Fuller Microsoft  
Frank Zhao Panasonic  
Randy Turner Sharp  
Bob Morford SIS Microelectronics  
Anthony Fung ST Micro Electronics  
Peter Groz ST Micro Electronics  
Harry Hvostov ST Micro Electronics

#### Administrivia

Don Wright gave the next PWG meeting details:  
August 17-21  
Toronto Marriott at Eaton Centre  
525 Bay Street  
Toronto, Ontario, Canada M5G 2L2  
\$175CN (about \$120 US)  
Registration deadline is July 24, 1998

Future PWG weekly schedules will be as follows:  
1394 PWG on Monday and Tuesday  
MIB meetings on Tuesday evening (if needed)  
PWG Plenary meetings on Wednesday morning  
IPP on Wednesday afternoon and Thursday morning  
SDP on Thursday afternoon  
UPD on Friday

Greg LeClair presented the 1394 PWG meeting goals and proposed agenda topics:

- OUI Issues
- Transport Command Set
- Config ROM Review
- Profile Review
- Updates

#### OUI Issues

Greg LeClair asked if we need to have an OUI for the 1394 PWG Profile?

The group did not resolve this issue, but a few attendees suggested that the IEEE 1212 group should maintain the list of Function Classes. Greg

said that this topic was raised at the previous 1212 meeting, but no conclusion was reached at that meeting. Further discussions were deferred.

#### Error Recovery Model

Isoda-san gave a presentation on his ideas for supporting error recovery. His main requirements are that the Initiator shall keep the contents of the data buffers associated with the ORBs in the linked list, and maintain the correspondence of the data buffers to a Sequence Identifier. The Target must guarantee that it does not execute any data or command twice.

Greg Shue asked if the proposal addresses both writing from the Initiator as well as writing from the Target. Isoda-san suggested that it does address both, but he is more confident about the Initiator writing recovery.

Isoda-san claims that we should maintain a Sequence Id for the ORBs. As part of the proposal, he explained that it is not necessary for every ORB to have a status notification sent. It is adequate to send status for a subset of the ORBs. Specifically, if a Sequence Id can be a maximum of  $2n-1$ , it is necessary to have at least one status sent for every group of  $n$  ORBs that have been sent since the previous status.

He showed that, based on the requeuing of ORBs, the Target can determine if the Initiator has received the most recently sent status block. He also said that if a Sequence Id is not used, the Target cannot determine if the Initiator has received the status block or not.

A few people commented that the idea of a "sliding window" of ORB sequences was a good one.

During discussion of the proposal, Randy Turner said that he believes we should keep negotiation of buffer size.

Brian Batchelder asked if anyone could evaluate the benefit (in terms of performance efficiency) gained by reducing the number of notification status blocks sent. The only comment was that "it really depends on the size of the data buffers."

#### Transport Command Set

Greg Shue referenced his proposal document, PWG 1394 Transport Command Set Proposal Revision 0c. (This document is available at <ftp://ftp.pwg.org/pub/pwg/pl394/mtg070698/pwgcmd0c.pdf>.) He provided a quick overview of the document and each of the open issues including those items in the document that are identified as "to be determined." The major discussion items and any conclusions reached are given below. (The discussions were not sequential as the paragraphs below might imply.)

[NOTE: Greg LeClair captured additional issues that were raised. His list should be included in the minutes.]

#### Queues

Should the usage of each queue be specified (e.g. Read and Write) or left arbitrary? Should we allow a vendor to extend the number of queues within a single Login?

Brian Batchelder suggested that if an implementer would like to extend the number of queues, it should be accomplished by additional Logins. He

suggests that we should limit a given Login to a (simple) bi-directional communication "pipe." (The idea of two independent queues was originally proposed because SBP-2 doesn't inherently support this capability.)

The group agreed to limit the number of queues to 2; one for Initiator-to-Target transfer and one for Target-to-Initiator transfer. It is intended that the group will focus on providing the equivalent of a bi-directional transport.

Should we eliminate the queue (q) bit, and just use the direction (d) bit only? Is it a reasonable assumption that the direction is equivalent to the queue id? It was decided that this issue is dependent on error recovery, and it was deferred.

#### Notification

There was a long discussion about the problems that can occur if a status notification (or its acknowledgment) can be lost. There was also some doubt expressed about the frequency of this condition ever happening. If it is an extremely unlikely event, perhaps it is better to simplify the recovery process as much as possible.

Should we set the notify bit to one (1) for every ORB? Do we need to always notify completion of every ORB? Greg Shue points out that we are using the unordered model of the SBP-2 specification. A direct implication of this is that every ORB must have completion notification. A "well-partitioned," layered SBP-2 driver does not have sufficient knowledge about the ORBs in the queue when using an unordered model. If the driver did have this knowledge (by examining the q bit, for example), it would be considered "munging" (combining) the different layers.

After looking at the latest SBP-2 specification, this issue is not (apparently) explicitly addressed. There is no clear statement that says the Initiator must set the notify bit when using the unordered model. Greg Shue would like to make an explicit statement about this in the PWG Profile document. He identified three possible alternatives:

1. Require an Initiator to set the notify bit on every ORB
2. Require the Target to always give completion notification
3. Allow for the possibility of "monolithic" (multi-layered)

SBP-drivers to be implemented without saying anything additional beyond what is already specified in the SBP-2 specification

After more discussion, the group agreed that the PWG Profile document would not make any additional, explicit statements on this issue other than to follow the SBP-2 specification.

#### Extensibility

To what degree should we provide extensibility? Do we really need to support a mechanism for vendor-specific transport commands?

As stated above, the group agreed to limit the number of queues to 2. The number of queues is not extensible.

Shimura-san suggested that vendor-specific commands could be useful as a method to fix bugs. Most of the other attendees felt that this was not a good enough reason to include support for vendor-specific commands. It was suggested that vendor feature differentiation could (and should?) be achieved at higher levels.

The group agreed that a Command OUI is not necessary (at least for this version of the Profile.) No vendor-specific commands or parameters will be supported. Any extensions will only be possible through the PWG

standards group by revision of the document.

## Parameters

When can parameters be set and/or changed?

Because the SBP-2 Login is not extensible, Randy Turner suggests that the PWG Profile should include a Login Response, followed by an exchange of additional parameters. He also pointed out that both the Target and the Initiator should know each other's capabilities -- at least to help resource allocation efficiency. It was suggested that the commands Get Param and Set Param List should be executed before data transfer, similar to a "connect" process. This suggestion was later expanded to an Open command that could establish other items that would be necessary for managing a connection.

ACTION: Randy volunteered to write up an example that would explain this concept.

## Error Recovery

Should we include a Sequence Id for the ORBs? What is the scope of the sequence number?

After a fairly brief discussion, the group agreed to use a Sequence Id for error recovery purposes.

Brian Batchelder noted that the Target should not process more than  $2n$  ORBs ahead of the last ORB for which the Target received an acknowledgment of completion status from the Initiator (where 'n' is the bit width of the Sequence Id.) Otherwise, it is possible that the Target could lose track of the error recovery information provided by the Sequence Id.

A long discussion occurred about pre-fetching ORBs and the reliability of ORB pointers after a bus reset.

Greg Shue noted that the group needs to define a unit attention status model. However, he suggested that the topic is probably too large to adequately address at this meeting. It was suggested that the topic be discussed via e-mail.

A few individuals discussed the processes involved in error recovery, based on the assumptions of the proposal given by Isoda-san. After the discussion, John Fuller proposed that the Sequence Id should be eight bits wide. Nagasaka-san wanted it to be 16 bits wide. After discussing the need to have the Sequence Id be sufficiently larger than the Maximum Task Set, the group agreed to 16 bits for the Sequence Id and 14 bits for the Maximum Task Set size.

What should be the initial value for the Sequence Id? Greg Shue suggested that the initial value could be inferred based on the first ORB on each queue after a Login or an abort task. Randy preferred that some "initial process" be defined to ensure a specific value for the first Sequence Id. The topic was deferred for discussion via e-mail.

John Fuller noted that the ORB Sequence Ids might have "gaps," because of the use of Abort Task and Abort Task Set commands. This caused people to wonder if it is necessary to generate a notification for each ORB to handle error recovery properly. However, it was decided that Greg Shue (and others) would spend time after the meeting to examine this question more closely.

ACTION: Greg will attempt to write up a statement for discussion via

e-mail.

Randy is concerned that without a timeout value (for an ORB completion notification), there is no way a host can know the lifetime of an ORB. How long should an Initiator keep an ORB allocated? The group suggested that if the Initiator wants to free the resources associated with an ORB before the completion notification has been received, an abort task command should be issued.

What should the timeouts be for the following items?

- abort task
- abort task set
- logical unit reset
- Target reset

The discussion of timeout values was deferred.

#### Open/Close Connection

Several items were put into a category of "connection meta-data" that should be communicated and established. The items include:

- initial values for Sequence Id numbers
- connection type
- completion policy
- maximum record size

How can a Target close a connection (i.e. unsolicited status)?

This entire topic was deferred for e-mail discussion.

#### Other Deferred Topics/Issues/Questions

The following items were raised during the discussions, but not directly resolved:

- When should a Target complete the Store Data Command Response?
  - Does it make sense to use the names "Store Data" and "Fetch Data" for commands?
  - Would it be better to just have a single "Transfer" command, with the direction (d) bit used to indicate which way the data should be transferred?
  - Should the q bit be placed in the status block?
  - Should we keep the Max Data Size Per Fetch ORB parameter?
  - Should we keep the Max Data Size Per Store ORB parameter?
- What should happen to Fetch Data and Store Data commands that are in or added to the active task set when a data flow direction is disabled?
  - Maybe we could put "last acknowledged Sequence Id" in each status block?
  - Perhaps we should add some attributes and features to the Unit Directory? This might be a mechanism (for example) to communicate the maximum buffer size of the Target.

#### Summary of Commands Discussions

Greg Shue attempted to summarize the decisions reached:

- 2 queues only
- No vendor-unique extensions (for this revision)
  - no CMD\_OUI
  - no Parameter\_OUI
- Provide only equivalent bi-directional transport
  - MSG (packet)
  - Stream
- Error recovery
  - 16 bit increasing Sequence Ids

14 bit size for Max Task Set

Remaining issues and assignments for write-ups:

- sequence # [Greg Shue]
  - seeding, abort task, interactions
  - notify bits and gaps in sequence
- Open/Close CMDs [Randy Turner and Greg Shue]
  - parameters and encodings
  - negotiated buffer sizes
  - mode and completion policy [Bob Morford]
  - management agent CMD support
- How will Target initiate close (unsolicited status)?
- How to communicate/clear UNIT ATTENTION?
- Timeouts [Alan Berkema]
- Is queue implied by direction bit?
- What initiates error recovery on Microsoft stack? What steps are taken? [John Fuller]

Config ROM Review

Because Greg LeClair could not be present the second day of the meeting, Alan Berkema attempted to lead a discussion of the CSR and Config ROM document. (This document is available at [ftp://ftp.pwg.org/pub/pwg/p1394/mtg070698/pwg\\_cfgrom.pdf](ftp://ftp.pwg.org/pub/pwg/p1394/mtg070698/pwg_cfgrom.pdf).) However, it did not seem that many people read the document. There was very little discussion.

Imaging Device Profile Review

Alan Berkema decided that the Profile document will continue to contain Issues and their resolutions. (This document is available at [ftp://ftp.pwg.org/pub/pwg/p1394/mtg070698/pwg\\_cfgrom.pdf](ftp://ftp.pwg.org/pub/pwg/p1394/mtg070698/pwg_cfgrom.pdf).)

Meeting adjourned.