# SpaceWire-RT: 10th SpaceWire Working Group Meeting, ESTEC, 20th February 2008

## A) Summary of Open Issues in the SpaceWire-RT Requirements

The following summary describes the key discussion points (open issues only) raised during the presentation of the SpaceWire-RT requirements document by Steve Parkes.

1. Should there be a requirement on the best-effort and resource-reserved services to deliver packets to the user in sequence?
    - For:
        - SOIS defines in-sequence delivery
        - SOIS aims to present a consistent interface for all sub-network standards, other standards (e.g. MIL STD-1553) always deliver packets in order
        - If an application requires in-sequence delivery and it uses a best-effort service without in-sequence guarantees it requires more complexity
    - Against:
        - In-sequence delivery requires extra complexity in the SpaceWire-RT stack
        - If best-effort is required to guarantee in-sequence delivery then SpaceWire-RT will not be able to provide a basic service equivalent to simple SpaceWire packets
        - If an application requires in-sequence delivery it should use the assured service
        - If an application requires in-sequence delivery it will probably require other characteristics provided by the assured service

2. Should there be an explicit requirement on the best-effort and resource-reserved services to ensure that no duplicate packets are delivered to the user?
    - For:
        - SOIS defines that no duplicates are delivered
        - SOIS aims to present a consistent interface for all sub-network standards, other standards (e.g. MIL STD-1553) never deliver duplicate packets
        - If an application requires no-duplicates and it uses a best-effort service without no-duplicate guarantees it requires more complexity
    - Against:
        - No-duplicate delivery requires extra complexity in the SpaceWire-RT stack
        - Under normal conditions SpaceWire does never transmit duplicate packets

3. Should there be a requirement for best-effort and resource-reserved services to carry out error checking?
    - Similar to above, should the best-effort and resource-reserved services be a thin a layer as possible over SpaceWire?

4. Should SpaceWire-RT define a maximum packet length? If so, how long should it be?
    - Can this be implementation defined?
        - Would be difficult to provide guarantees of properties such as timeliness under these conditions
    - Would it be possible to give a list of supported packet lengths?
    - Should the requirements specify where the segmentation/de-segmentation should take place?

5. The current SpaceWire-RT requirements document exposes a single global address space to the user which is translated into SpaceWire addresses, it is proposed that these are 16-bit.
    - Does this have to be fixed, can it be variable length?

- Is 16-bits enough?
- SOIS does not define address size.

6. <mark>The current set of requirements define the timeliness of SpaceWire-RT networks by giving the case of a 1ms latency across a network composed of three routers.</mark>
   - Is this definition necessary?
   - Is 1ms correct?

7. <mark>Better use cases are required for testing and analysing proposed implementations of these requirements.</mark>
   - Use cases gratefully received.

Note:

- Where these notes state "Suggest:", the following comment was made by the speaker
- Where these notes state "Query:", the following comment/question was made by a member of the working group who was not the speaker.

# B) Detailed discussion flow:

## B.1 SpaceWire-RT Requirements (S. Parkes presentation)

- Concentrate on requirements, not on protocols
- RT = real-time or reliable and timely
- Best effort
  - Single attempt
  - In sequence, no errors, no duplications
  - Permit priorities
- Assured
  - Ensures delivery
  - If no delivery possible, sender informed
  - In sequence, no errors, no duplications
- Resource reserved
  - Single delivery attempt
  - Not assured
  - In sequence, no errors, no duplications
  - All within a channel (channel defines resources used to transmit SDU)
  - Priority within a channel
- Guaranteed
  - Combines assured and resource-reserved
- Suggest: Best effort and assured should not require in-sequence, can cause unnecessary lost packets

- Suggest: No duplicates, should implement as don't send duplicates rather than check for duplicates
- Packet length is defined by SpW-RT, a lot of analysis depends on maximum packet length
  - Query: should it be defined on a project-basis?  Suggest defined once-and for all
  - Query: give a predefined list of maxima, project can choose? Suggest: possibly
  - Query: where should packets be chopped up?  Should this be specified in the requirements?
- SpaceWire shall see SpaceWire addresses
- Redundancy defined as a primary route plus alternative route(s)
  - Query: are these routes to the same destination? Suggest: yes, always
- Channels have channel identifiers and are used when timeliness is required
- May use time-division
- May use bandwidth-reservation
- Suggest remove requirement on no duplication
  - Query: but SOIS specifies this, and SOIS specification should not be driven by SpaceWire
  - Query: If SOIS is maybe wrong then we need use cases
  - Query: Should best effort include error checking? Suggest: Needs discussion
- Discussion:
  - Best effort should be as simple as possible
  - If you permit out of sequence then it puts complexity into application
- Channel resources are shared between guaranteed and assured
- Addressing discussion:
  - Global address number
  - Should this be limited?
  - SOIS says that this should not be limited
  - Suggest: 16-bit
  - Suggest: global address space that can be mapped onto SpaceWire local addresses
  - Query: shouldn't use addresses for functions with SpaceWire nodes
  - Query: Can't 223 addresses (LAs) because need to allocate both duplicate LAs and LAs for EGSE
- Retry and redundancy strategies:
  - Query: You should state that all strategies are permitted concurrently
  - Query: Do you have requirements on statistic collection?  Suggest: This should be added as a requirement
- Query: What is 1ms requirement based on?  Surely packet size and link speeds should be specified?  Suggest: that links speeds are added to the requirement.
- Query: Should the requirements have a number for timeliness in it at all?  Suggest: yes.
- Query: Wish to see an additional requirement stating that SpW-RT should not affect existing standards.
- Query: Should this include time-synchronisation?  Left open.
- Query: With TDM use case, what about multiple PDUs per slot?  Are the requirements restricted to one PDU per slot?  Suggest: Not in requirements
- Query: Can synchronise OS scheduling to TD slots which will improve efficiency

- Query: On what basis do you calculate the bandwidth usage?  Measure bandwidth using a similar resolution to TDM
- Query: Do you need to synchronise the periods over which the bandwidth is measured? Suggest: yes
- Query: Do acknowledgements have higher priority than the packets that they are acknowledging?  Suggest: yes.
- Query: Use cases are too simplistic for testing?  Suggest: Yes, can you provide better use cases?
- Philippe: Issues will be collated, and distributed in May, ready for discussion in June WG meeting.

## B.2 Use Case Study for SpaceWire-RT (T. Yamada)

- JAXA onboard system architecture specification is written but only available in Japanese
- Use intelligent nodes to manage non-intelligent nodes
- Suggest/assume the SpaceWire-RT is carried over RMAP
- Clock synchronization is done using RMAP writes
    - Synchronisation is done using time-codes
    - RMAP writes provide precise clock value at which time-code was/will be transmitted
- Many use cases require TDM
- All use cases could use assured with priority
- Suggest: TDM is the simplest way of meeting latency and multiplexing requirements
- Suggest: May need to mix BWR and TDM (BW reservation in TDM slots)
- Conclusion: Most networks will use Resource reserved using TDM (possibly with priority, possibly using BWR in TDM slots)
- Query: Is RMAP in the correct position in the stack
- Query: Do you have requirements to do "remote terminal" to "remote terminal" transfers? (Based on 1553) No, everything goes via intelligent node (used as bus controller).