Four address support by RPR MAC service primitives

Marc Holness, Nortel Networks

IEEE 802.17 WG — 802.17b SG

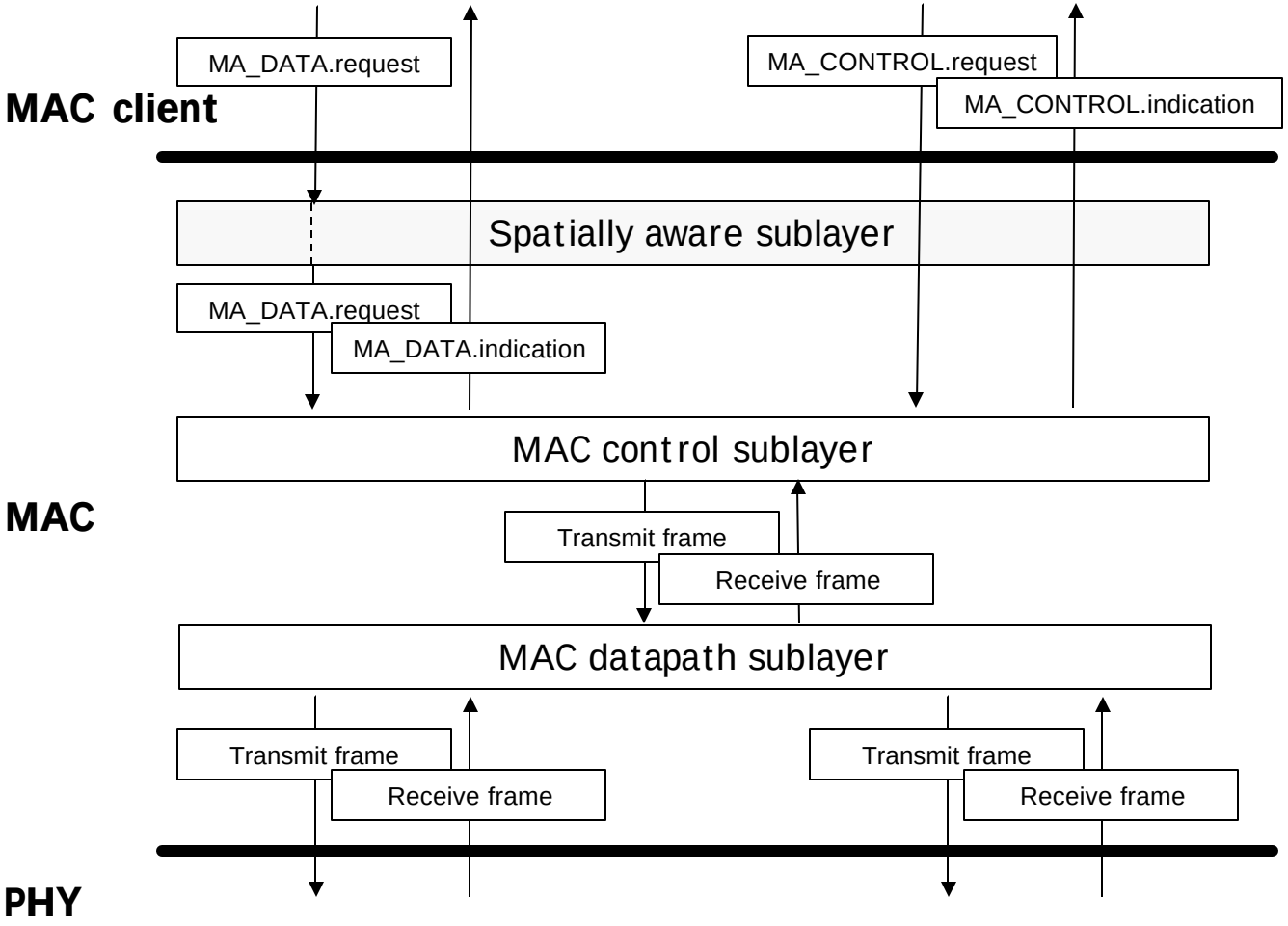
Atlanta, Georgia

March, 2004

Objectives

- Illustrate how client MAC addresses and RPR MAC address can be supported by the RPR service primitives

Service primitives



MA_DATA primitives

```
MA_DATA.request
(
      destination_address,
      source_address,                  // optional
      mac_service_data_unit,
      frame_check_sequence,            // optional
      service_class,
      ringlet_id,                      // optional
      mac_protection,                  // optional
      mark_fe,                         // optional
      strict_order,                    // optional
      destination_address_extended,    // optional
      source_address_extended,         // optional
      flooding_form                    // optional
)
```

```
MA_DATA.indication
(
      destination_address,
      source_address,
      mac_service_data_unit,
      frame_check_sequence,
      reception_status,
      service_class,
      ringlet_id,
      fairness_eligible,
      strict_order,
      extended_frame,
      destination_address_extended,
      source_address_extended
)
```

MAC client provides addresses

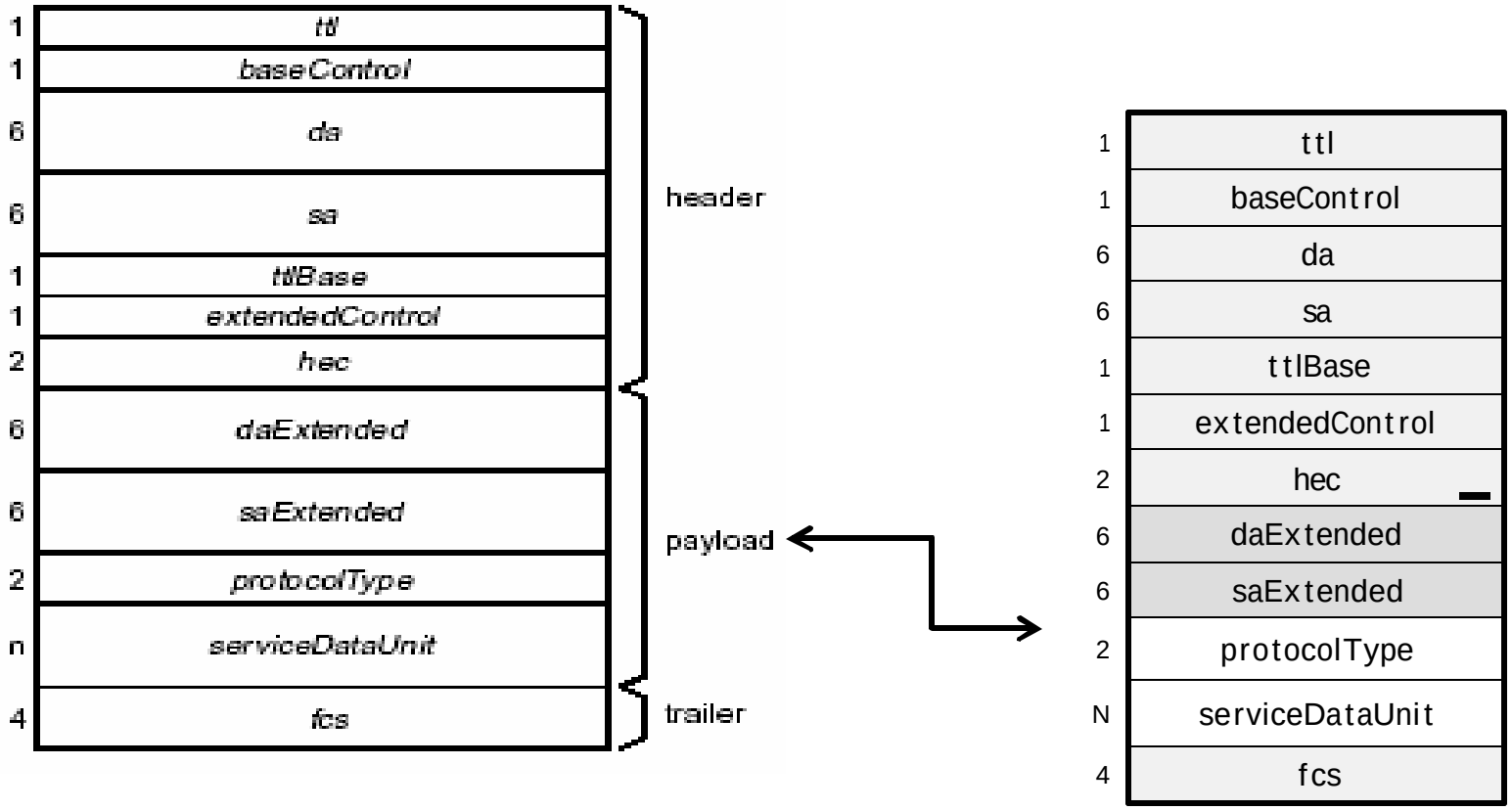
- Two methods that client can provide all 4 address to the MAC

1. Utilization of extended address parameters
   - Client passes additional addresses in the `destination_addres_extended` and `souce_address_extended` parameters of the MA_DATA request primitive
2. Embedding additional addresses in the `mac_service_data_unit`

NOTE: When a client uses the extended address parameters in the MA_DATA requrie primitive, the resulting RPR frame format is an extended frame format.

Extended addressing frame structure

Extended RPR frame format



NOTE: RPR MAC needs to process the daExtended and saExtended fields.

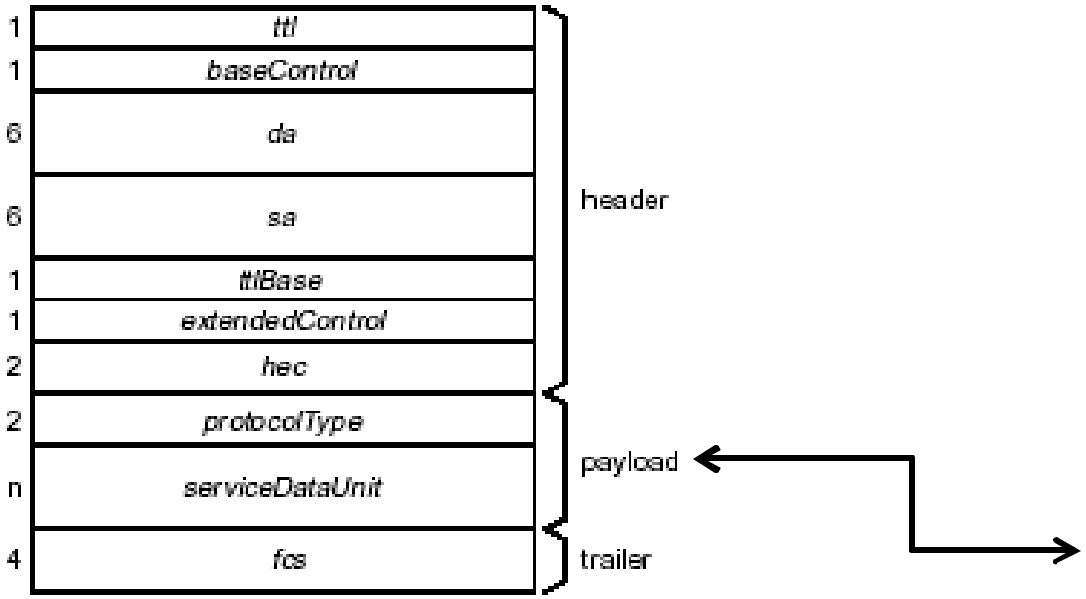
Embedding addition addresses in MSDU

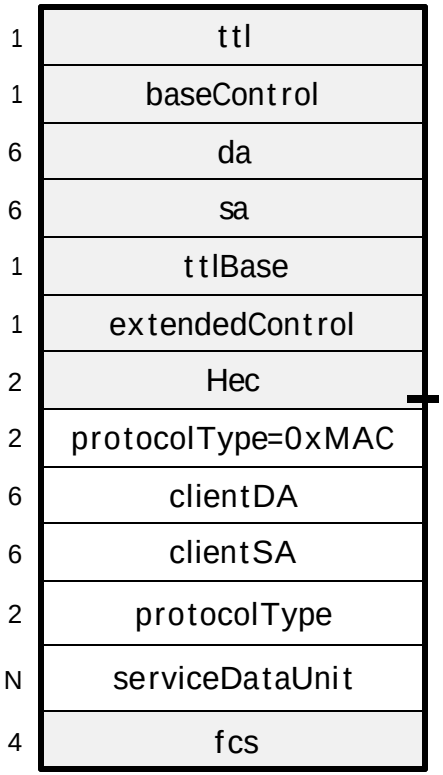
- MSDU parameter passed by client to MAC begins with a protocol type indicating next 12-bytes are MAC addresses (i.e., client destination address and client source address)
- A new protocol type needs to be defined and can be interpreted by RPR MAC clients
- Additional (i.e., client) MAC addresses are transparent to the RPR MAC

NOTE: Similar technique to support MAC encapsulation being being considered by IEEE P802.1ah.

Embedded addressing frame structure

Basic RPR frame format



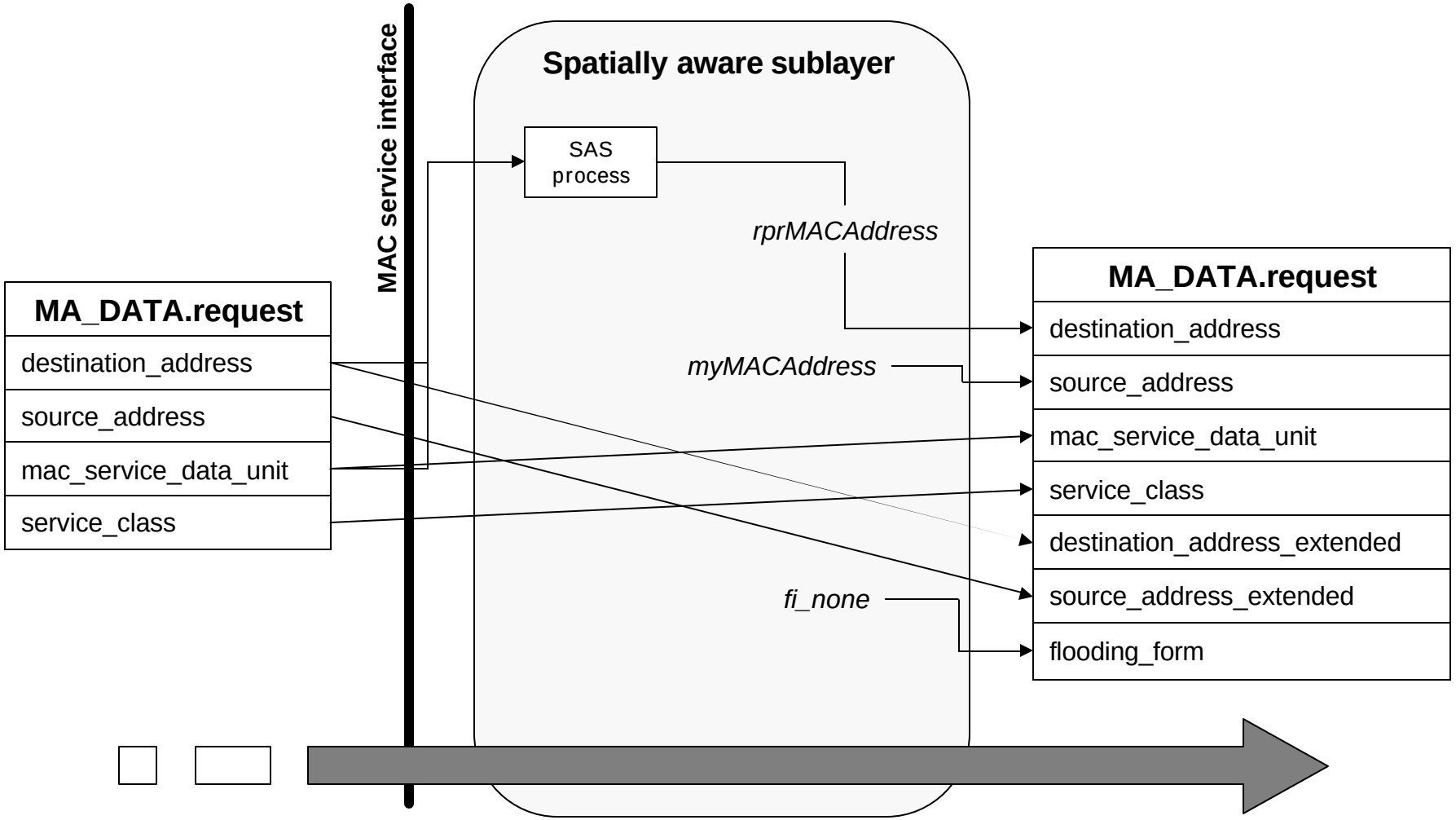


NOTE: The clientDA and clientSA fields are transparent to the PR MAC.

SAS provides addresses

- SAS utilizes the extended address parameters in the MA_DATA request (and indication) primitives
- Since SAS is apart of the RPR MAC, it makes sense that the RPR MAC processes the `destination_addres_extended` and `souce_address_extended` parameters, unlike the case when the MAC client passes additional addresses

Service data request primitive



Assumptions: The destination_address parameter is remote. Directed transmission is used since RPR destination address found in SAS FDB.

Summary

- Two techniques generally available to transport additional addresses by RPR MAC
  1. Utilization of the extended address parameters
  2. Embedding in additional address in the MSDU

Recommendation

- RPR MAC clients that wish to send addition MAC addresses should use the embedding in the MSDU technique
- SAS (which is apart of the RPR MAC) should use the extended address parameters
- If/when SAS moves to a client, it should use the embedding in the MSDU technique
  - This is the technique being proposed by IEEE P802.1ah which has a similar problem of client MAC encapsulation
  - No 802.1 service interface changes would be required