

On Scalability of In-Situ OAM

draft-song-ippm-ioam-scalability-01

Haoyu Song, Tianran Zhou, James Guichard

Scalable Data Type Extension (1)

- **Issue**

- The original draft supports at most 16 types of telemetry data. Currently 14 types were defined.
- Include a user defined data type, which is proprietary and cannot be standardized.
- No way to extend the data type if more data types need to be standardized

- **Solution**

- Use extended data-type bitmap to support more data types
- The last bit of the data-type bitmap indicates the presence of the next data-type bitmap

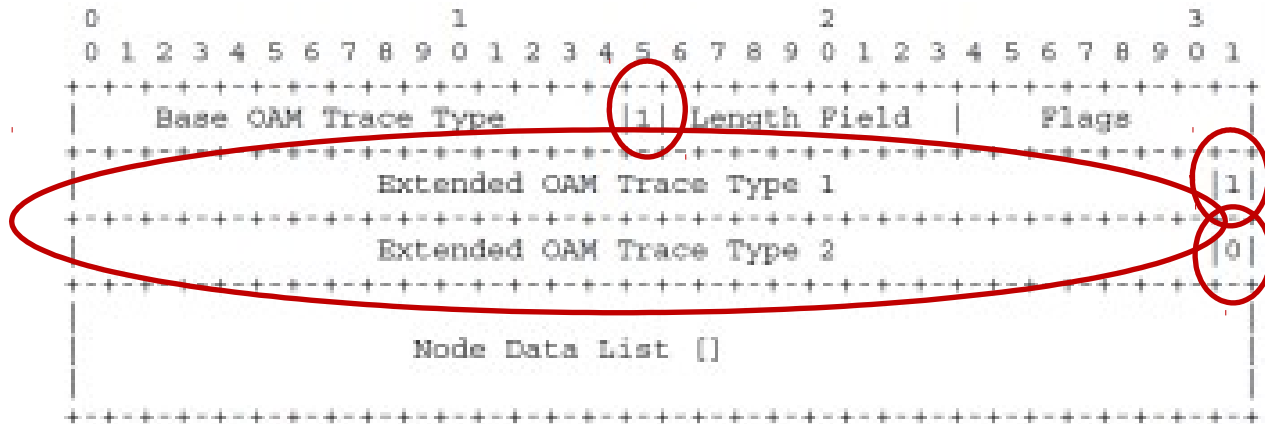


Figure 1: Extended OAM Trace Type Header Format

Scalable Data Type Extension (2)

- **Use Cases :**

- When flow packets pass a service chain, the original flow identity may be altered (e.g., NAT, load balancer), but the OAM system needs to map each packet back to its original flow.
 - Save the original flow identifier (e.g., 5-tuple) in the iOAM header
- Wireless, mobile, and optical networks need to collect some physical data associated with a flow
 - Power, temperature, signal strength, GPS location, wavelength, etc.
- Other possible data types
 - Metered flow bandwidth
 - Time gap between two consecutive flow packets
 - Buffer occupancy of the node
 - Remaining time to the packet delivery deadline

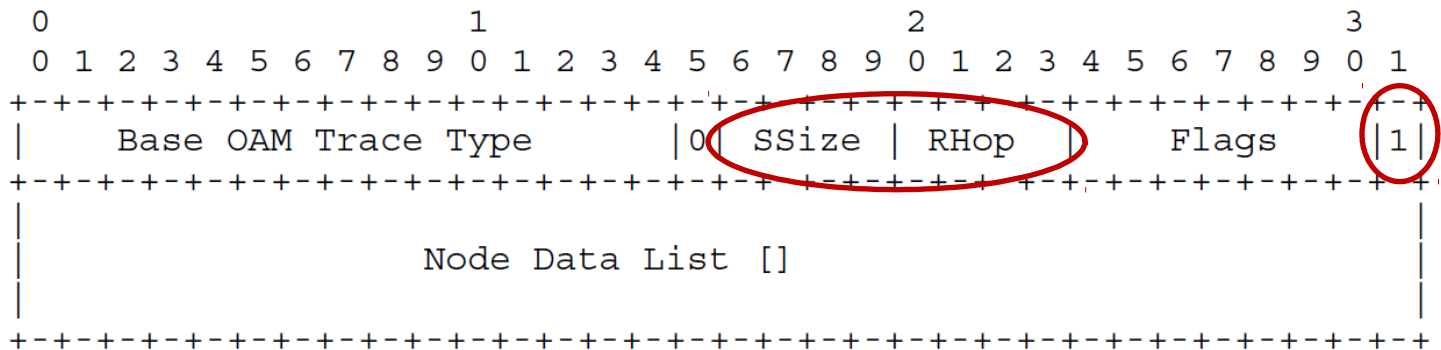
Segment iOAM – Cope with Packet Size Limitation (1)

- **Issue**

- Path length and MTU limit may leave no space for iOAM data in packets
- iOAM data overhead may be too large and consume too much network bandwidth
- Packet drop causes the loss of all iOAM data

- **Solution**

- Partition path to fix sized segments and collect iOAM data at every segment end, which limit the iOAM header overhead



Segment iOAM – Cope with Packet Size Limitation (2)

- **Use cases**

- Identify the drop point of flow packets and retain iOAM data up to the drop point
- Determine the segment size based on the packet size, the MTU limit, and the data collected per node
- Determine the segment size when given the fixed iOAM data overhead and the data collected per node
- On a very long path, segment is needed to avoid a super long iOAM header, which may be beyond the node processing capability

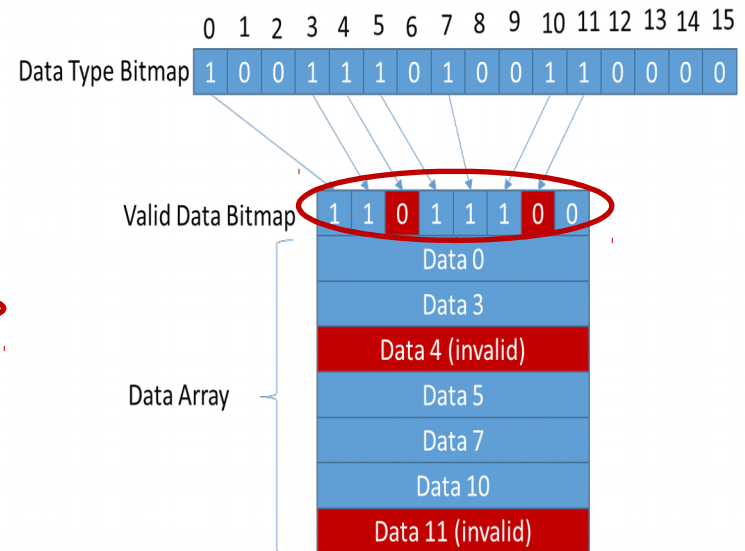
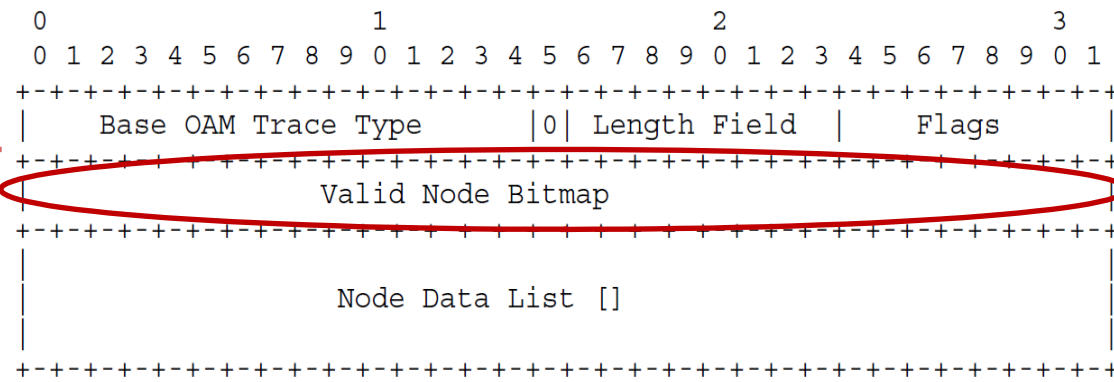
Valid Data Bitmap – Adapt to Node Processing Capability (1)

- **Issue**

- Due to privacy, security, or capability reasons, some node may not be able to fulfill a part or all iOAM data request

- **Solution**

- Valid Node Bitmap: if a node cannot provide iOAM data, its corresponding bit is set to 0
- Valid Data Bitmap per node data: if a node cannot provide some iOAM data, its corresponding bit is set to 0



Valid Data Bitmap – Adapt to Node Processing Capability (2)

- **Use cases**

- A node is temporarily under heavy traffic load and is in danger of dropping packets if it tries to satisfy all the iOAM data request. In this case, it stops adding iOAM data in the node and only set the corresponding bits in the valid data bitmap to 0
- In a heterogeneous data plane, some node support to provide data type x but the other nodes cannot. In this case, the head node can still request data type x but only those nodes who can support it will add the data
- For security reason, a node decides to not respond to the iOAM data request, but it can still handle and pass the packets with iOAM header. The node only needs to set the corresponding bit in the valid node bitmap to 0.

Other Issues and Proposed Solutions

- iOAM header overhead is large
 - More efficient data packing in iOAM data node
 - Allow data size to be multiple of 2 bytes
 - Head node can pre-calculate the offset of each data type and carry it with the iOAM header, or
 - Each node calculates the data type offset by itself, or
 - Each node adds data sequentially in the order that the data type appears in the bitmap
- Processing all flow packets and adding a large amount of iOAM data may overload node device, and
- Some data is redundant and repetitive, adding unnecessary burden to node device
 - Flexible iOAM configuration at head node
 - Allow to pick arbitrary flows or flow packets to add iOAM header
 - Allow different flow packets to carry different iOAM headers (i.e., w/ different data requests)
 - Allow add iOAM header to sampled flow packet only (based on probability or interval)

Thank you!