



draft-ietf-6tisch-msf

Presenter: Yasuyuki Tanaka

Tengfei Chang,
Malisa Vucinic,
Xavier Vilajosana,
Simon Duquennoy,
Diego Dujovne.

Updates since IETF 104 at Prague

- Three versions of MSF are iterated
 - draft-ietf-6tisch-msf-02 (presented at IETF04 at Prague)
 - draft-ietf-6tisch-msf-03 (April 8th)
 - draft-ietf-6tisch-msf-04 (July 2nd)
 - draft-ietf-6tisch-msf-05 (July 8th)
- Main Changes
 - Usage of Autonomous cell
 - Downward traffic adaptation
 - Resolve comments from Fabrice, Atis, Yatch, Toshio, Thomas, Pascal (available on mailing list)

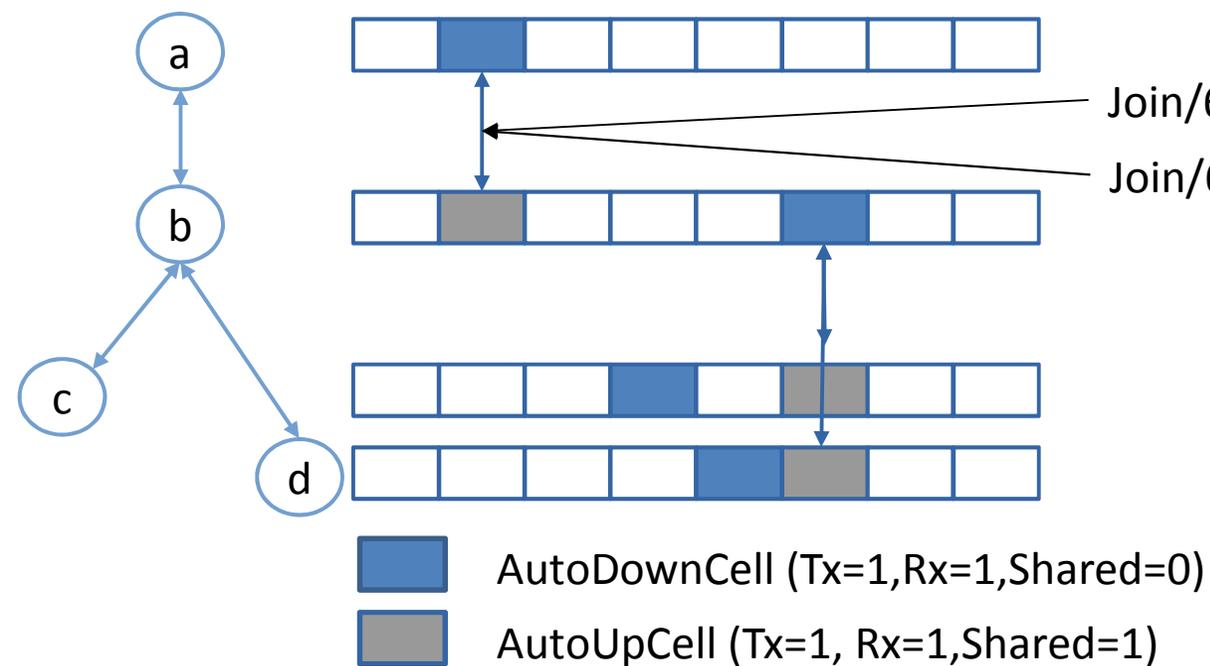
Main changes

- Usage of Autonomous cell
 - (MSF-02, MSF-03) AutoUpCells and AutoDownCells
 - AutoUpCells, (slotOffset, channelOffset) hashed with **parent EUI64** address
 - (Tx=1, Rx=1, Shared=1)
 - AutoDownCells, (slotOffset, channelOffset) hashed with **node itself EUI64** address
 - (Tx=1, Rx=1, Shared=0)
 - **AutoCells are installed permanently**
 - (MSF-04, MSF-05) AutoTxCells and AutoRxCells
 - AutoTxCells, (slotOffset, channelOffset) hashed with *l2 destination address of packet* to send
 - (Tx=1, Rx=0, Shared=1)
 - AutoRxCells, (slotOffset, channelOffset) hashed with **node itself EUI64** address
 - (Tx=0, Rx=1, Shared=0)
 - *AutoRxCells are installed permanently, AutoTxCells are installed on-demand.*

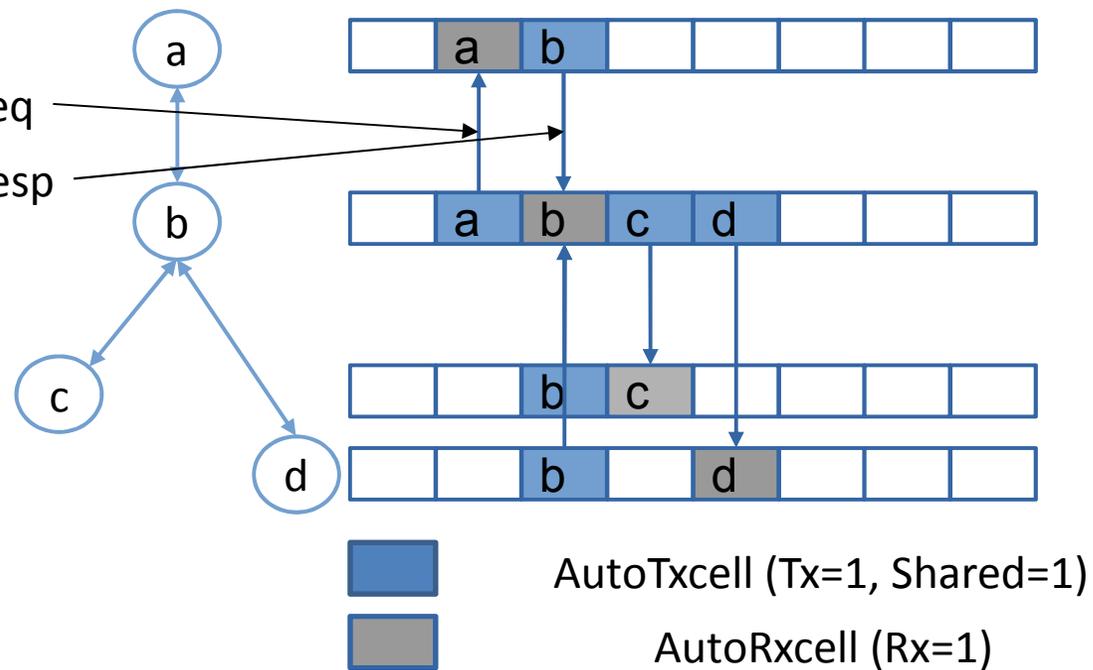
Main changes

- Usage of Autonomous cell
 - (MSF-02, MSF-03) AutoUpCells and AutoDownCells
 - AutoUpCells, (slotOffset, channelOffset) hashed with *parent EUI64* address
 - (Tx=1, Rx=1, Shared=1)
 - AutoDownCells, (slotOffset, channelOffset) hashed with *node itself EUI64* address
 - (Tx=1, Rx=1, Shared=0)
 - ***AutoCells are installed permanently***
 - (MSF-04, MSF-05) AutoTxCells and AutoRxCells
 - AutoTxCells, (slotOffset, channelOffset) hashed with ***12 destination address of packet*** to send
 - (Tx=1, Rx=0, Shared=1)
 - AutoRxCells, (slotOffset, channelOffset) hashed with *node itself EUI64* address
 - (Tx=0, Rx=1, Shared=0)
 - ***AutoRxCells are installed permanently, AutoTxCells are installed on-demand.***

Main changes



MSF-02, 03



MSF-04, 05

Main changes

- Downward traffic adaptation
 - Issue a 6P request to add a Rx to parent
 - NUM_CELLELAPSED and NUM_CELLUSED for Rx
 - CELLELAPSED++ when the current cell is a Rx cell to parent
 - CELLUSED++ when received something at a Rx Cell from parent
 - When CELLELAPSED == MAX_NUMCELL:
 - If CELLUSED / CELLELAPSED > HIGH_THRESHOLD:
 - Trigger a 6P Request to add one cell with celloption Rx=1 to parent
 - If CELLUSED / CELLELAPSED < LOW_THRESHOLD:
 - Trigger a 6P Request to delete one cell with celloption Rx=1 to parent

Main changes

- Comments Resolved

- Formula to calculate the channelOffset should use MAX_NUMCHANNEL rather than 16
- Recommended/suggestions value for MAX_NUMCELLS
- Looking for “frame is used for...” is implementation-specific. Instead using “MUST”, use “SHOULD”
- Unprotected frames will never send on negotiated cell, so no need to state only security packet sent on negotiated cell.
- Length of CellList is implementation specific. 5 should just be a recommended value
- CellOptions in 6P ADD/DELETE request are not specified.
- Items 1 and 2 for “parent switching” in Section 5.2 are not necessary.
- For start State, there are other process may undergoing, e.g. the 6LoWPAN ND, describe that phase and add a reference for it.
- Layer 2 source address - > Layer 2 destination address
- The node should not synchronize before listening expected EB and should listen for a certain mount of duration. (this is stated in the RFC8180, will add a reference from there)

- To be discussed

- Rules for CellList
- Downward traffic adaptation

Discussion

- Rules for CellList
 - Add rules to listen to the cells for a few slotframes to ensure that they are not used by neighbors.
 - Answers:
 - This situation is handled by “locked” feature in RFC8480 at section 3.4.3.
 - > In this case, the cells involved in an ongoing 6P Transaction MUST be "locked" until the
 - > transaction finishes....If the requested cells are locked, it MUST reply to that request with a
 - > 6P Response with return code RC_ERR_LOCKED (as per Figure 38). The node receiving
 - > RC_ERR_BUSY or RC_ERR_LOCKED MAY implement a retry mechanism as defined by
 - > the SF.

(offline)Discussion

- The issue of current downstream traffic adaptation
 - Each node installs one Rx negotiated cell at beginning, which is one Tx cell from its parent side. When a node has too many children, the Tx cells to children will occupy mostly of the schedule without being used frequently.
- Solution after discussed with Malisa and Thomas:
 - For Node A:
 - For upstream adaptation, nothing changes. The node A starts to adapt traffic when the first 6P negotiated Tx cell is installed
 - For downstream adaptation, the node A starts to adapt when the autoRx cell is installed. And we only count the NUMCELL_USED for packet from the parent of node A.
 - The downstream adaptation works in the fact that the children of node A will transmit on the 6P negotiated Tx cell, which won't conflict with the traffic from node A's parent.
- Will be in the next version of MSF

Conclusion

- Publish the new version after this meeting according to the discussion
 - Changes:
 - Apply the new downward traffic adaptation changes
 - Apply the new changes according to the discussion