

Forwarding Information Base for MANET Multicast

Lessons Learned from SMF

96th IETF MANET

Background - SMF

Simplified Multicast Forwarding (SMF)

- MANET Multicast (broadcast) protocol.
- Packets are disseminated to the entire MANET.
 - Duplicates are eliminated (no loops).
 - Optimized through use of Connected Dominating Set (CDS).
- Very efficient in small, highly dynamic mobile networks.
- Not intended to scale to large networks.

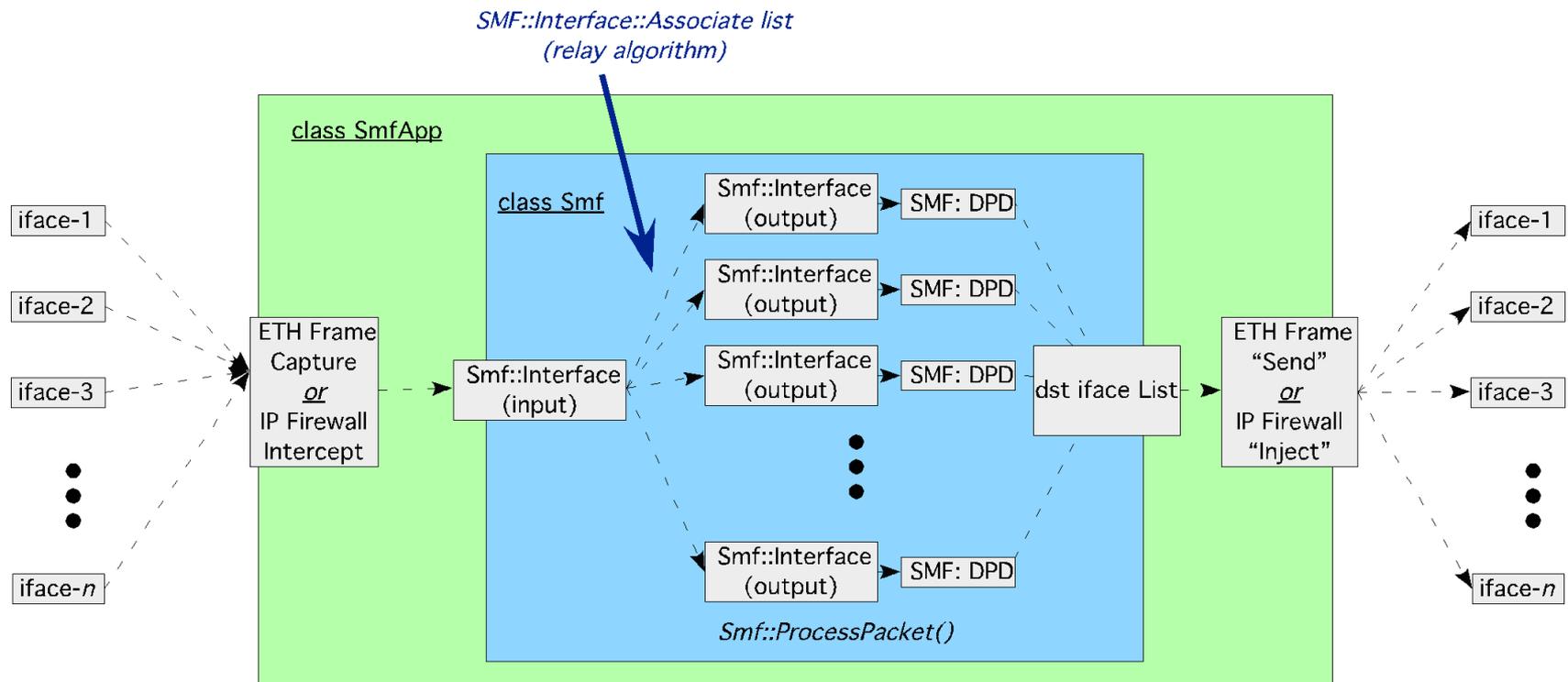
Background – SMF Shortcomings

- Design results in implementations in user space = poor performance
- Packets are disseminated to the entire MANET
- No knowledge of group membership
- CDS algorithms do not support multiple interfaces well
 - Appendix algorithms are all/none forwarding on all interfaces
- Forwarding rules are not well defined for multiple interfaces
 - DPD per interface? Incoming/outgoing?
 - DPD method managed manually
 - Hash based vs ID based

nrlsmf Functions

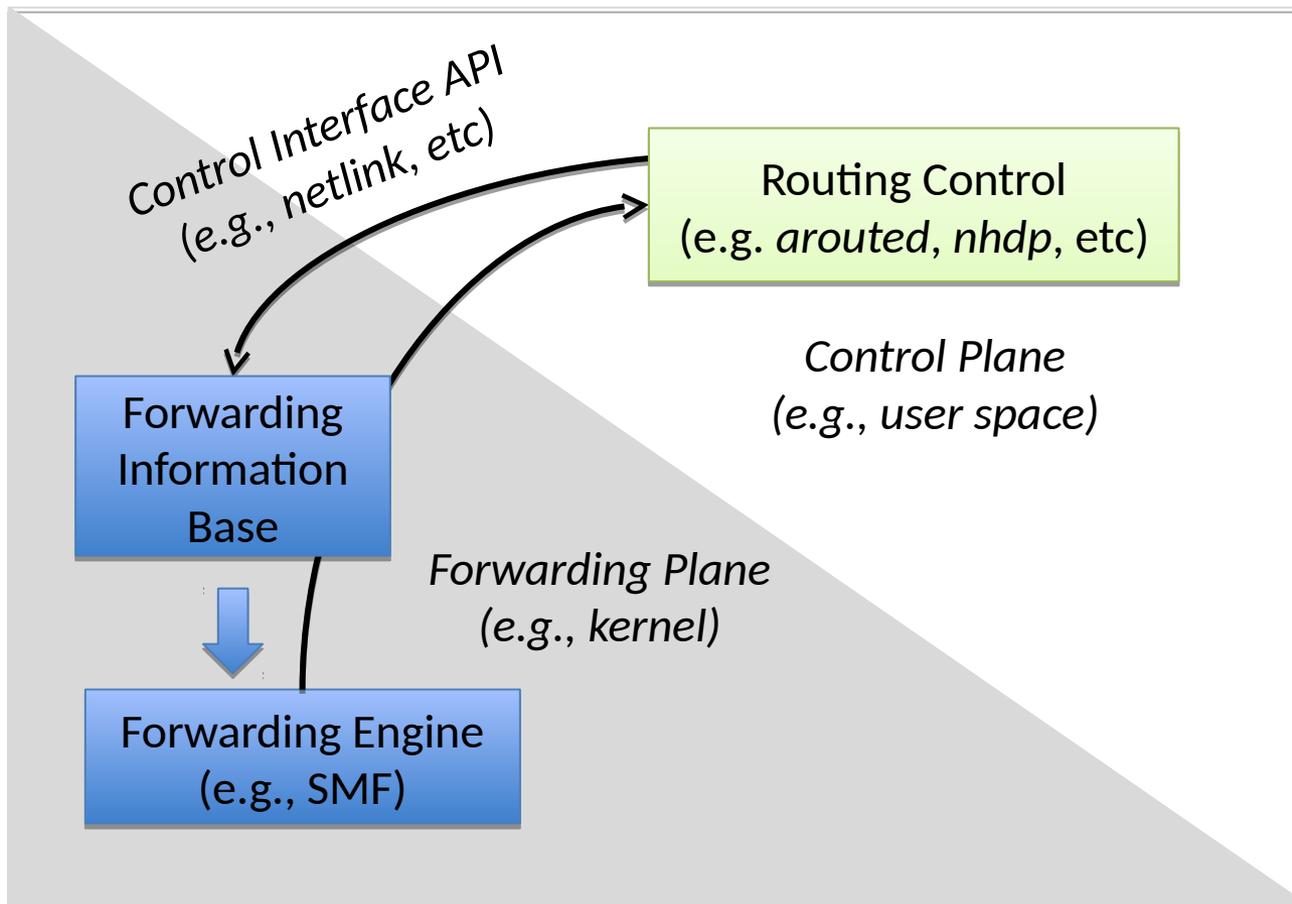
- MANET interface support
 - Duplicate packet detection
 - Supports S-MPR/ECDS/Classical forwarding
 - Multi-interface support (limited algorithmic control)
- Gateway support
 - Forced relaying of multicast packets across and among multiple interfaces.
 - Resequencing or packet marking for external flows injected into MANET/SMF areas
- “Remote control” interface allows external processes to control *nrlsmf* forwarding.
- Packet marking and resequencing for source hosts.

The Current nrlsmf Architecture

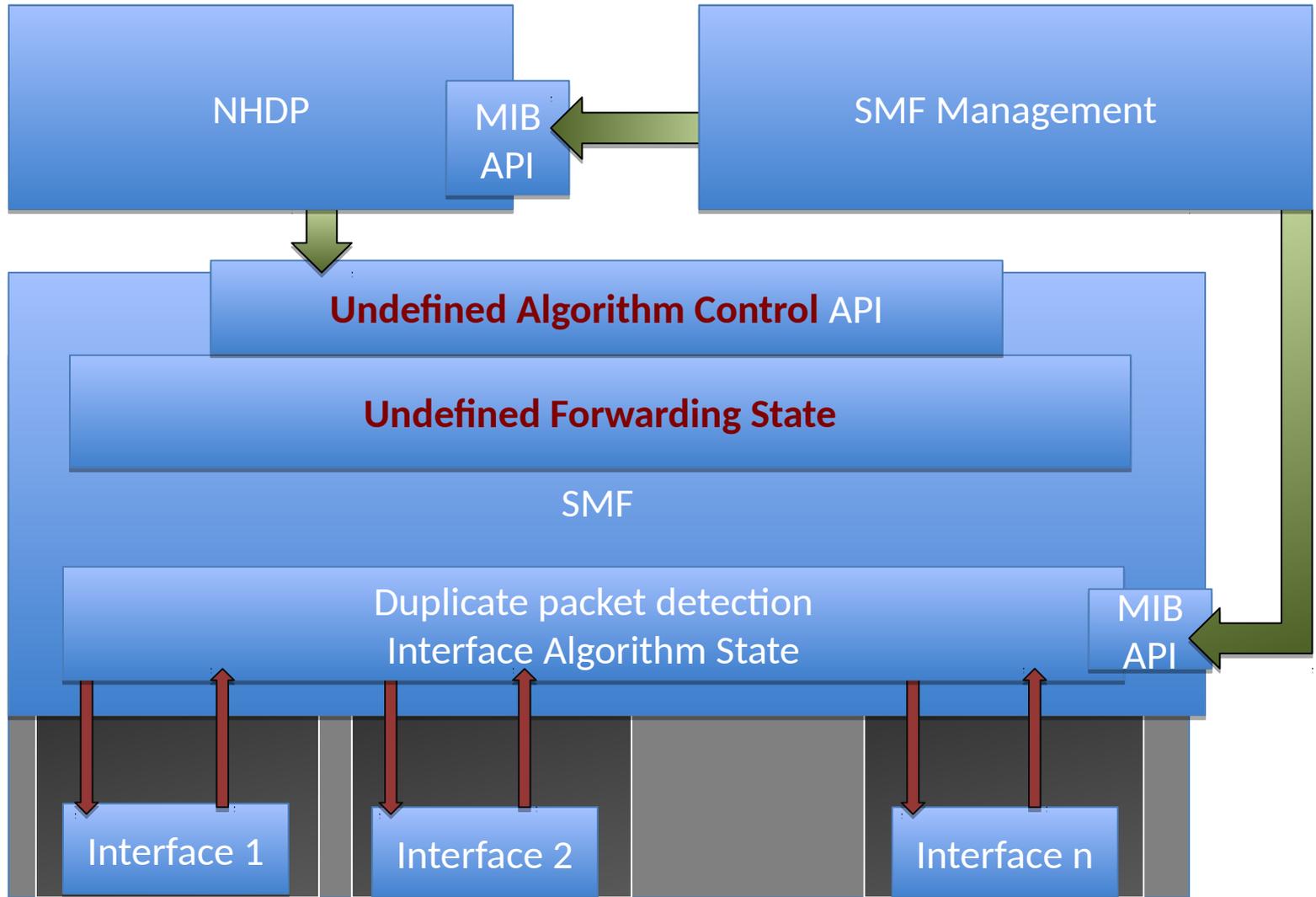


Note: A "Smf::Interface" may "associate" with itself for MANET interface forwarding

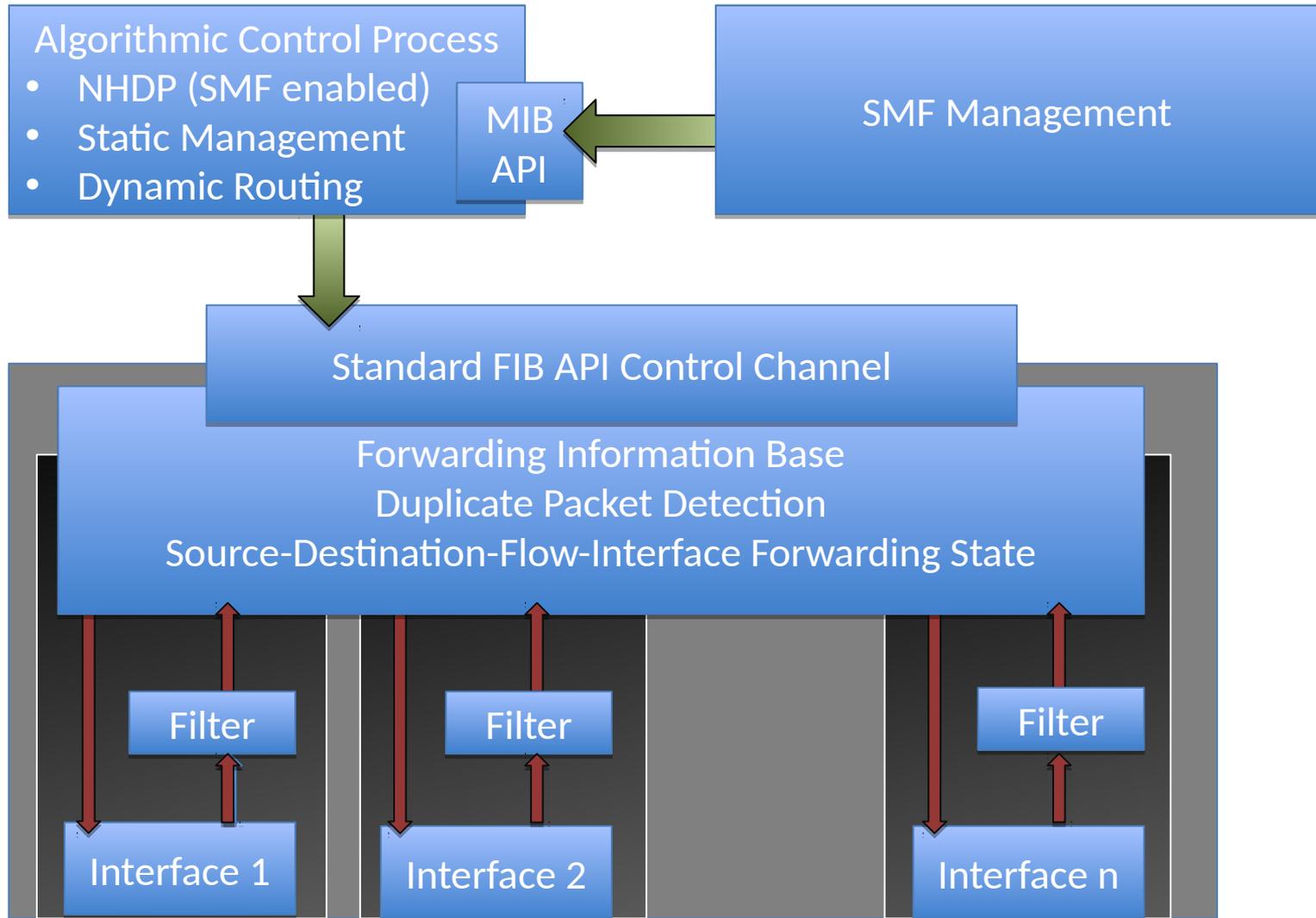
Separation of Forwarding/Control Plan in the Design



Current SMF Design



Next Generation MANET Multicast FIB Based Design: a cleaner approach

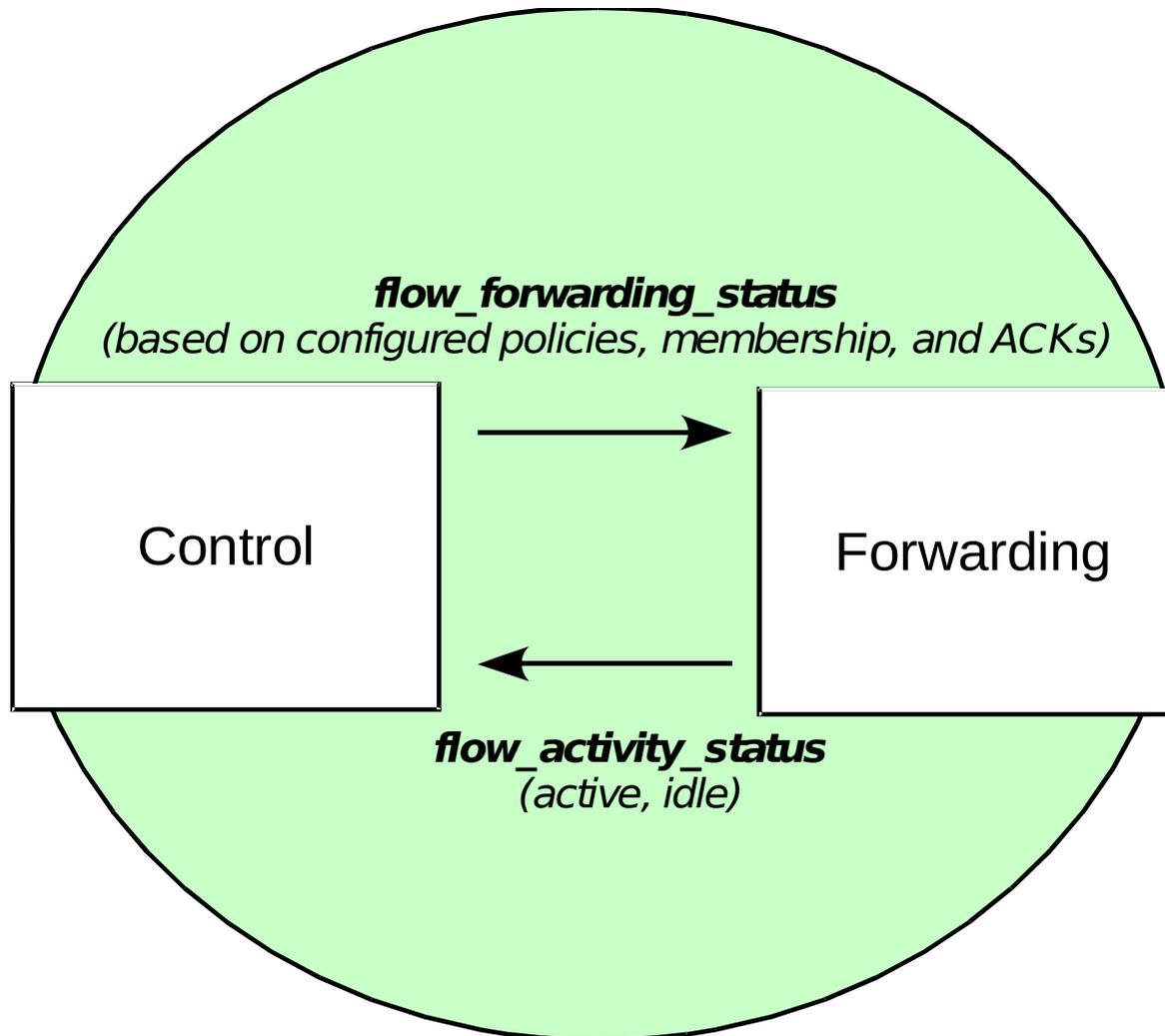


Future - Elastic Multicast

FIB Design should support

- Dynamic pruning of SMF relays for specific **group memberships**
- Converges to **single path multicast trees in stable networks**
- Expands to full network flood in highly dynamic networks
 - **Dynamic expansion/reduction in forwarding**
- Maintains no topology or global membership information
- Relay set reduction applied to higher bandwidth traffic
 - **Flow based reductions**
- Lower bandwidth traffic still flooded to the entire MANET

Elastic Multicast Control/Forwarding Interaction



Flow vs. Generalized Forwarding

- What do I mean by Flow?
 - Packets identified by
 - Destination
 - Source
 - Port
 - Protocol
 - Incoming interface
 - Forwarding rules based on largest match
 - Supports pruning
 - Supports group based joins
 - Supports general dissemination rules
 - Can reduce redundant rebroadcasts
- Current SMF algorithms only specify On/Off forwarding on all interfaces for ALL multicast

Just Forwarding?

- Current *nrlsmf* implementation doesn't just do forwarding on matching packets....
 - Forward
 - Limited (rate limited)
 - Hybrid (fixed forward then drop)
 - Drop
 - Queue
 - Error

FIB API should NOT be unidirectional

- Algorithmic controllers should be able to be notified of new flows
 - Perform algorithmic duties (i.e. send control messages etc.)
 - Update the FIB

MANET FIB for just Multicast?

- A “MANET Multicast FIB” useful for supporting multiple multicast approaches
 - Elastic Multicast
 - On Demand MANET Routing Protocol
- A “MANET FIB” **could** be part of a more generalized MANET approach in support other protocols
 - On-demand protocols
 - Network Coding
- A “MANET Multicast solution” would need to consider additional things including group management and gateway to existing infrastructure network multicast protocols

mroute.h

```

193 struct mfc_cache {
194     struct mfc_cache *next;                /* Next entry on cache line */
195 #ifdef CONFIG_NET_NS
196     struct net *mfc_net;
197 #endif
198     __be32 mfc_mcastgrp;                  /* Group the entry belongs to */
199     __be32 mfc_origin;                    /* Source of packet */
200     vifi_t mfc_parent;                    /* Source interface */
201     int mfc_flags;                        /* Flags on line */
202
203     union {
204         struct {
205             unsigned long expires;
206             struct sk_buff_head unresolved; /* Unresolved buffers */
207         } unres;
208         struct {
209             unsigned long last_assert;
210             int minvif;
211             int maxvif;
212             unsigned long bytes;
213             unsigned long pkt;
214             unsigned long wrong_if;
215             unsigned char ttls[MAXVIFS]; /* TTL thresholds */
216         } res;
217     } mfc_un;
218 };

```

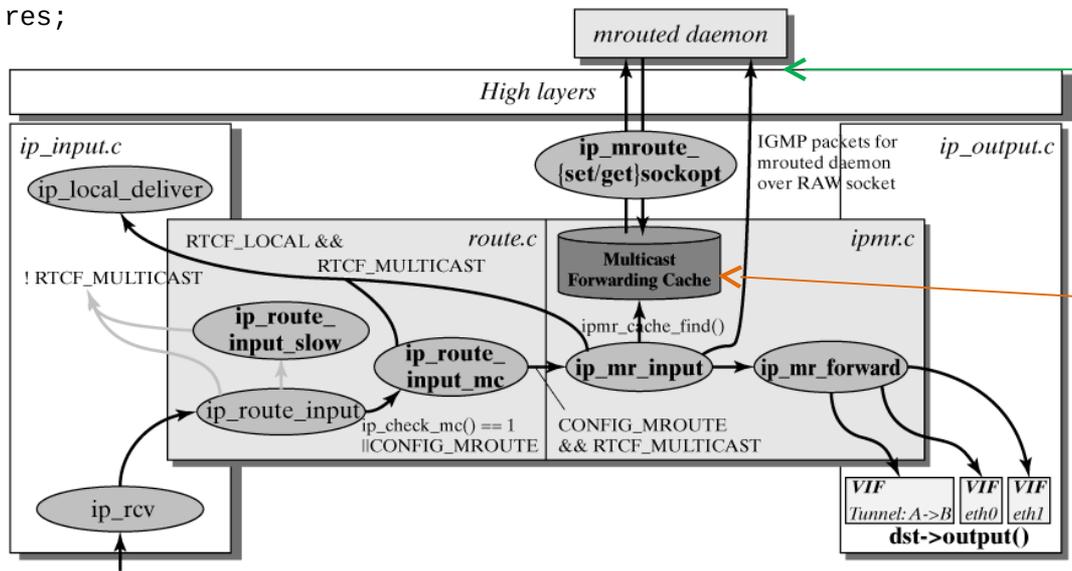
A possible implementation approach

DPD?

Other stuff

SMF / EM

Multiple tables



← (Red arrow pointing from 'DPD?' to the 'unresolved' field in the code)

← (Blue arrow pointing from 'Other stuff' to the 'res' union in the code)

← (Green arrow pointing from 'SMF / EM' to the 'mroute daemon' box)

← (Orange arrow pointing from 'Multiple tables' to the 'Multicast Forwarding Cache' box)