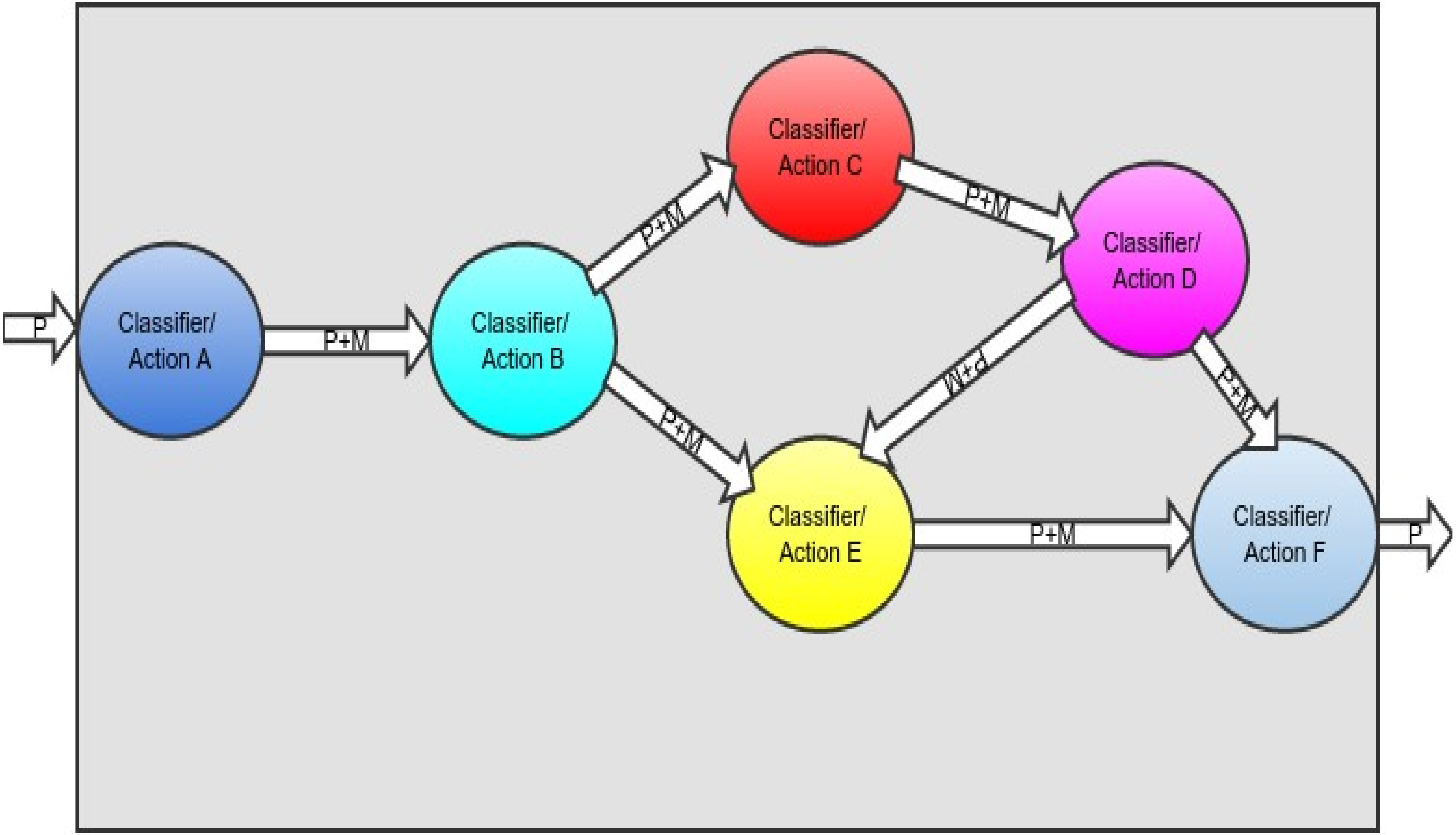
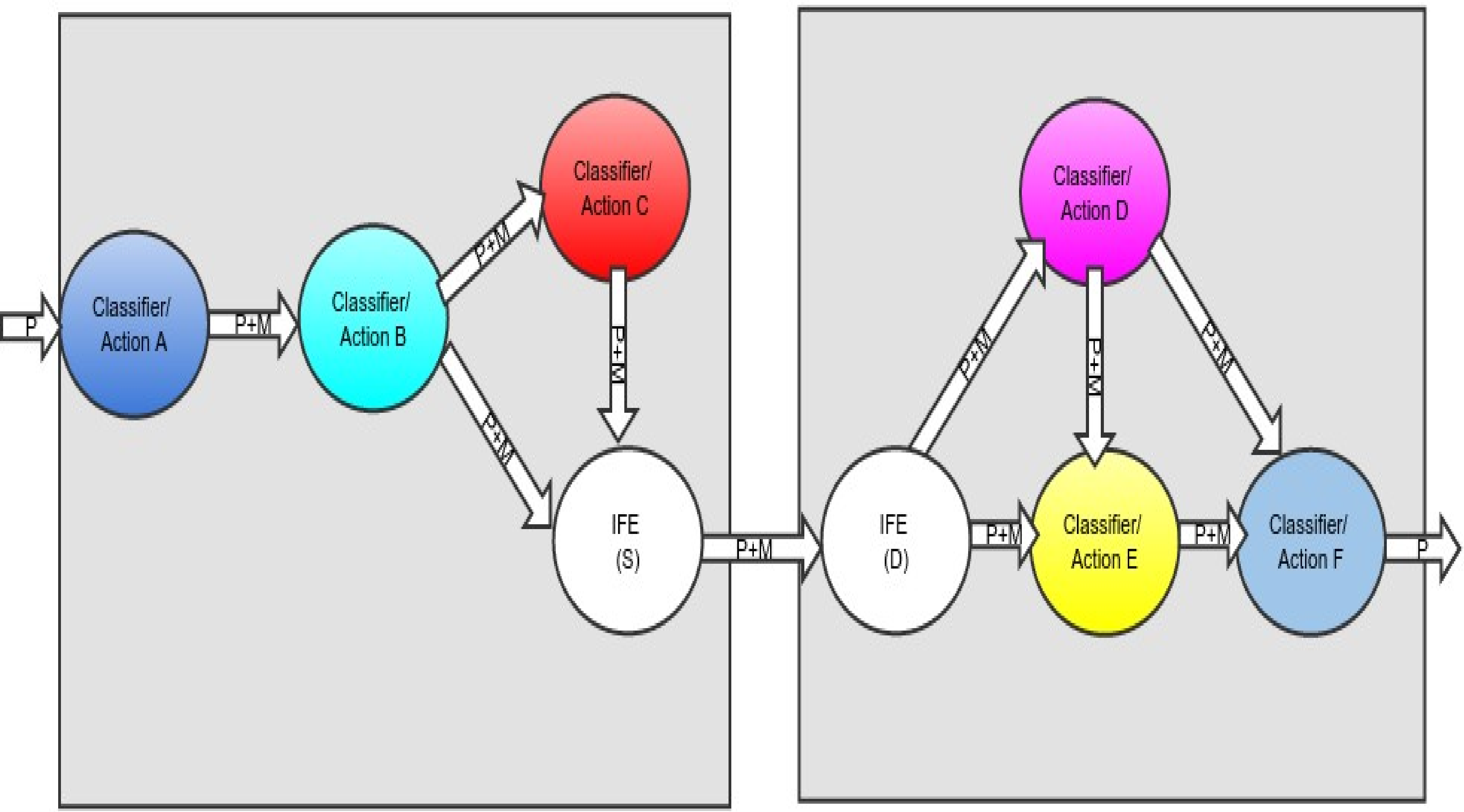
Distributing TC Action

Jamal Hadi Salim
Damascene M. Joachimpillai

Basic TC Classifier Action Graph



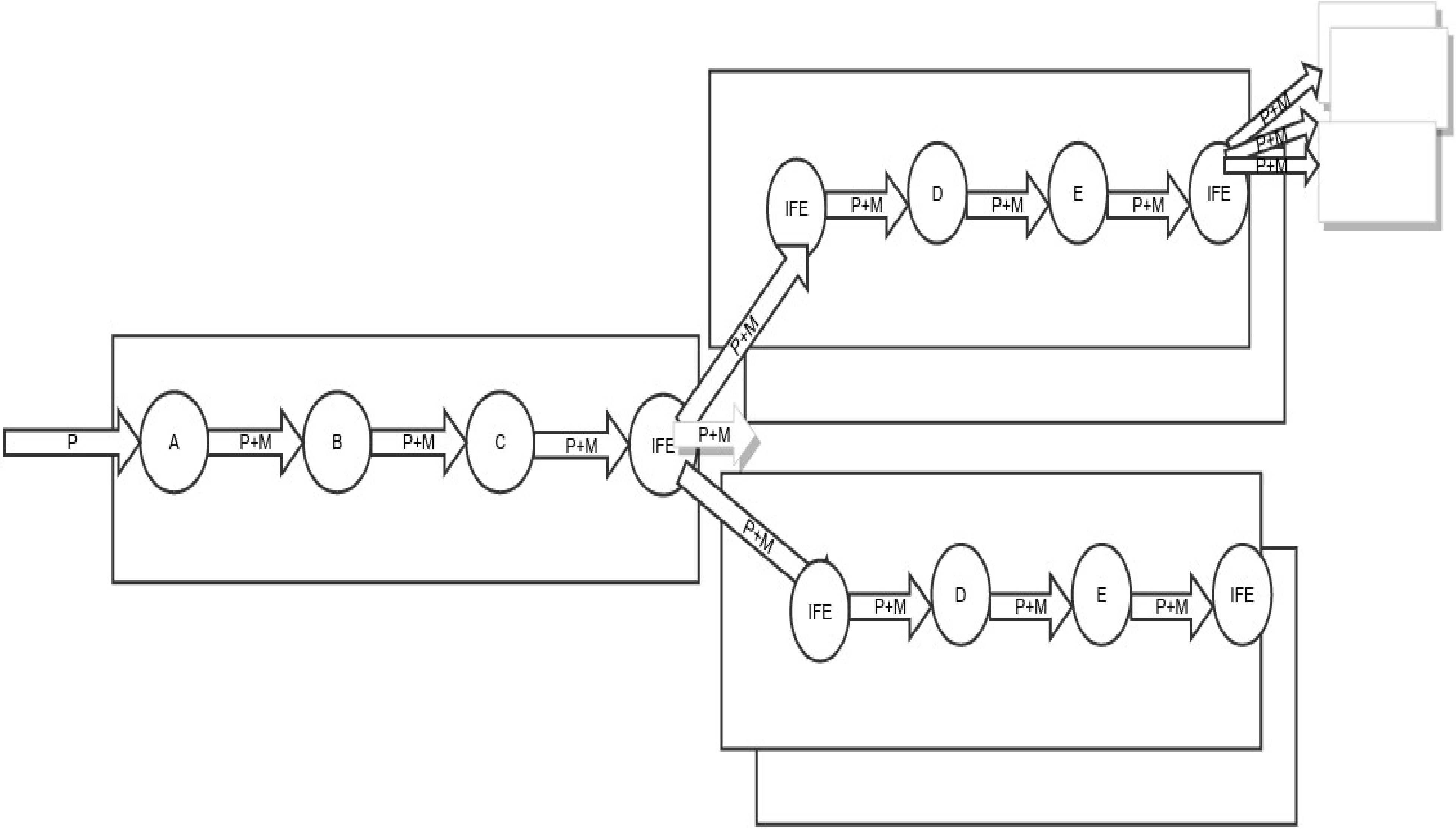
Distribute It...



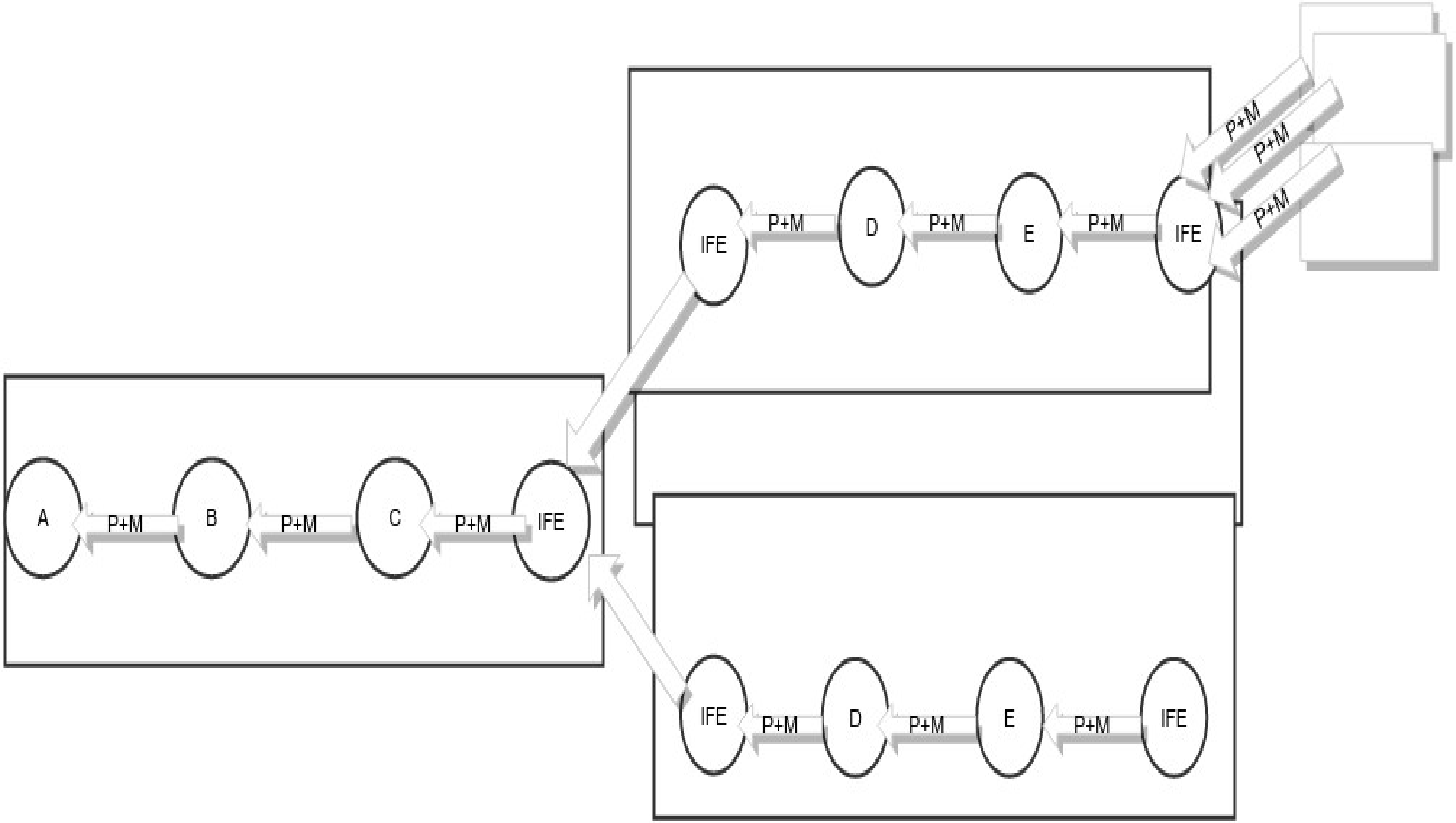
Basic Requirements

- Need to be able to distribute metadata
- The graph node vertices and number of edges remain unchanged

Use Case: Systolic Processing



Use Case: Systolic Processing



Use Cases

- OAM info encoding
- Exception service handling
- Authentication and Authorization info
- Versioning info
- Compliance info
- etc

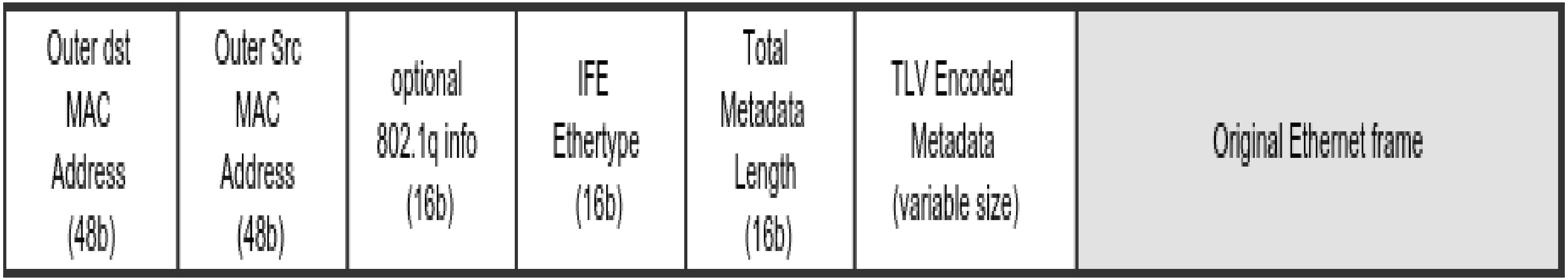
Egress Side

```
tc filter add dev $ETH parent 1: protocol ip prio 10 \
u32 match ip protocol 1 0xff flowid 1:2 \
action ... action ... \
action ife encode type 0xDEAD \
allow mark use hash 10 use qmap 17 \
use mystring "foobar" \
dst 02:00:00:22:01:01 src 52:54:00:c3:4b:c5 \
action ...
```

Ingress Side

```
tc filter add dev $ETH parent ffff: prio 2 \
protocol 0xdead \
u32 match u32 0 0 flowid 1:1 \
action ife decode allow mark reclassify
#
tc filter add dev $ETH parent ffff: prio 5 protocol ip \
handle 0x11 fw flowid 1:1 action .....
```

Wire Format



Outer dst MAC Address (48b)	Outer Src MAC Address (48b)	optional 802.1q info (16b)	IFE Ethertype (16b)	Total Metadata Length (16b)	TLV Encoded Metadata (variable size)	Original Ethernet frame

- Encode on Egress port
- Decode on Ingress

Ethernet Challenges

- MTU
- Ethernet type

Metadata ID Challenges

- Standardize vs proprietary
- Discover vs static

Adding New Metadata Extensions

- Kernel module struct tcf_meta_ops
  - metaid
  - name
  - ops: encode(), decode(), alloc(), release(), get()
- Write a small extension to tc

Future Plans

- Performance measurement
- Discoverability
- Hardware offload
  - Ingress parse and populate dma descriptor