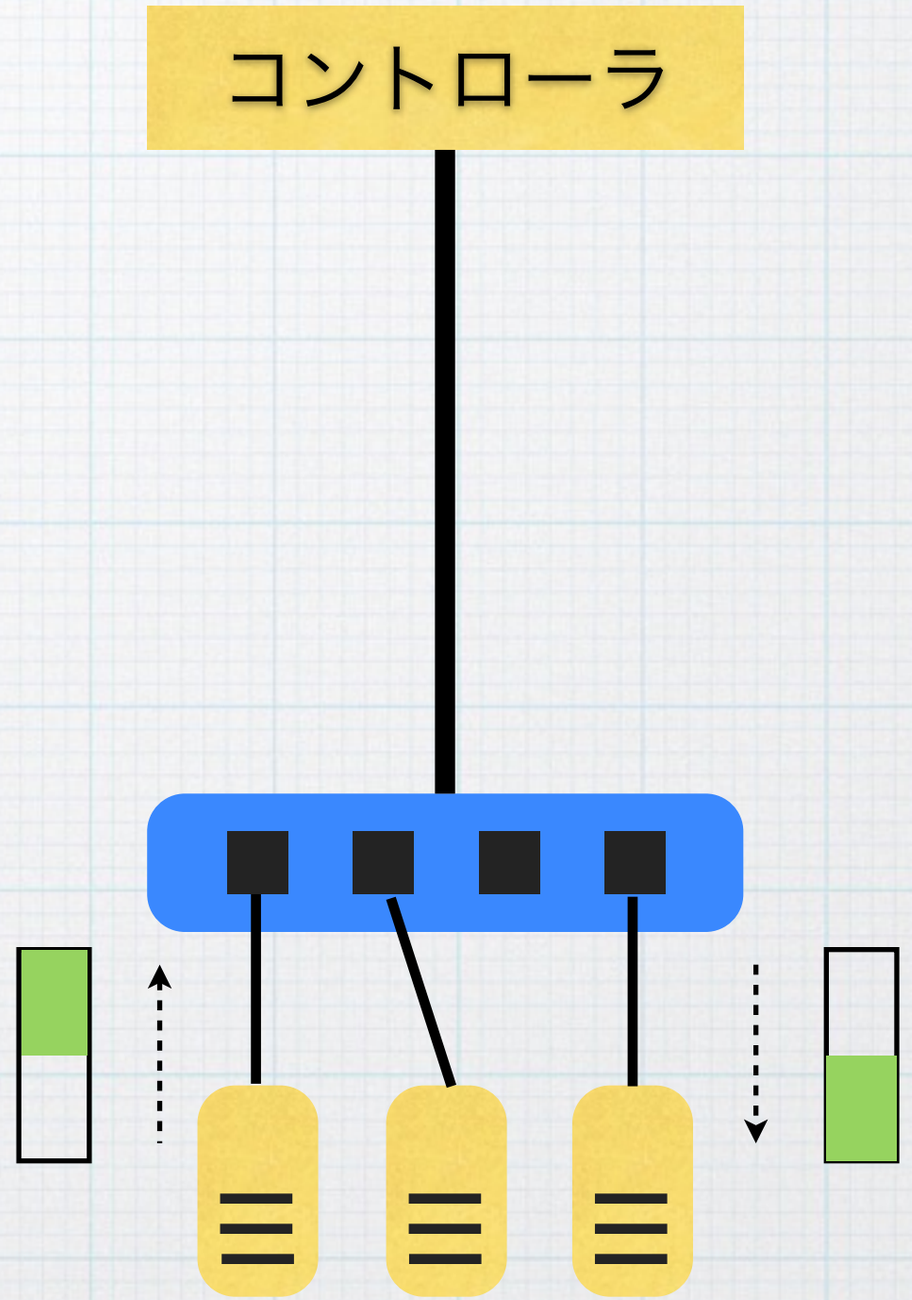
スイッチを作ろう
OpenFlow1.3編



仮想NW

ルーティング
スイッチ

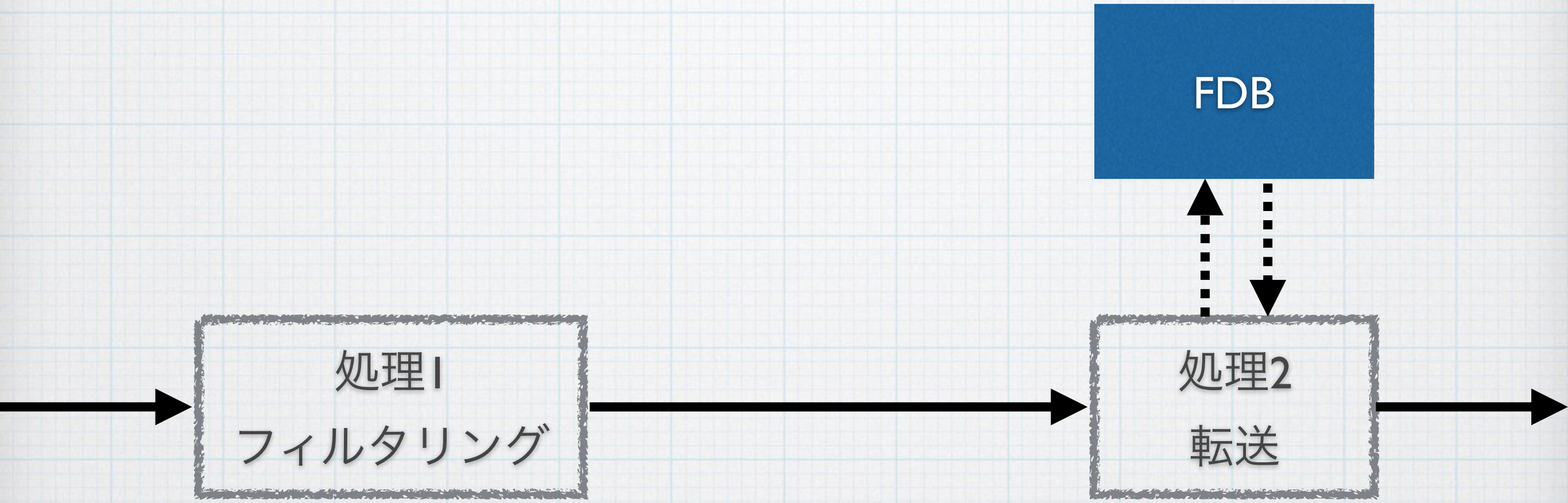
トポロジ
ディスカバリ

スイッチ

Hello
World

おさらい
OpenFlow1.0でのパケット処理

ラーニングスイッチ



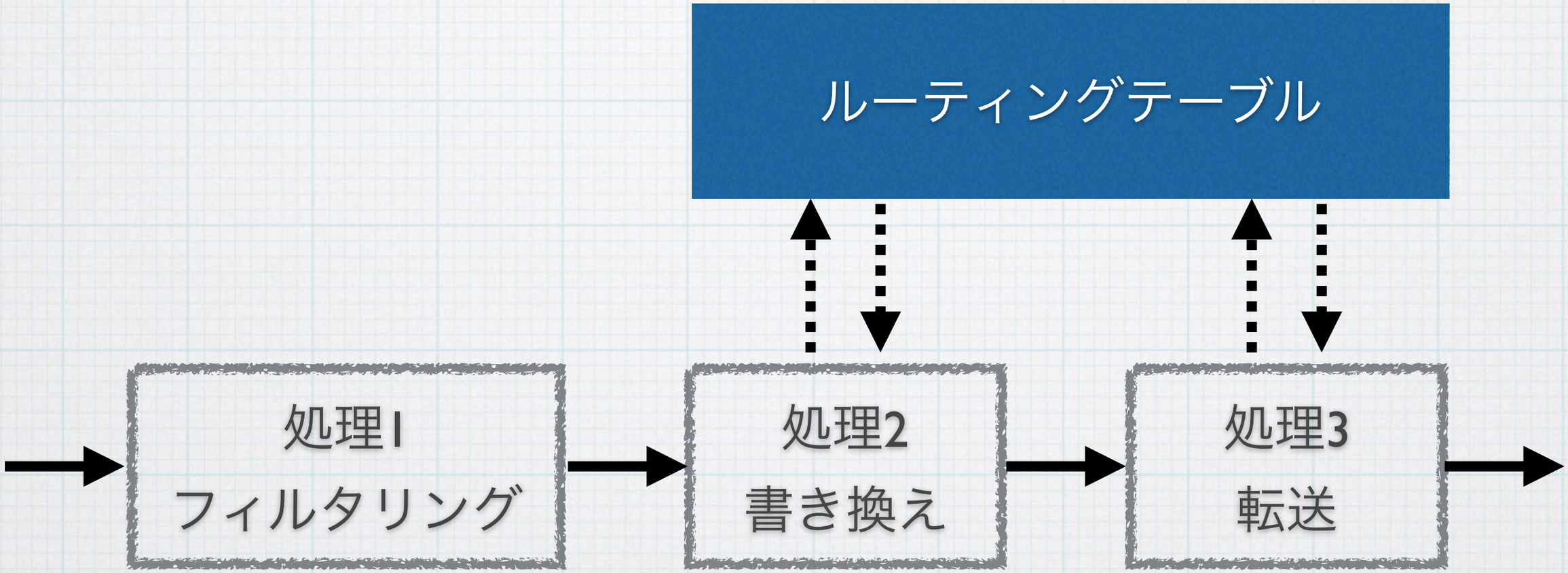
宛先 MAC =

- 802.1D/802.1Q reserved MAC
- マルチキャスト

をドロップ

ポートn番へ
or
FLOODING

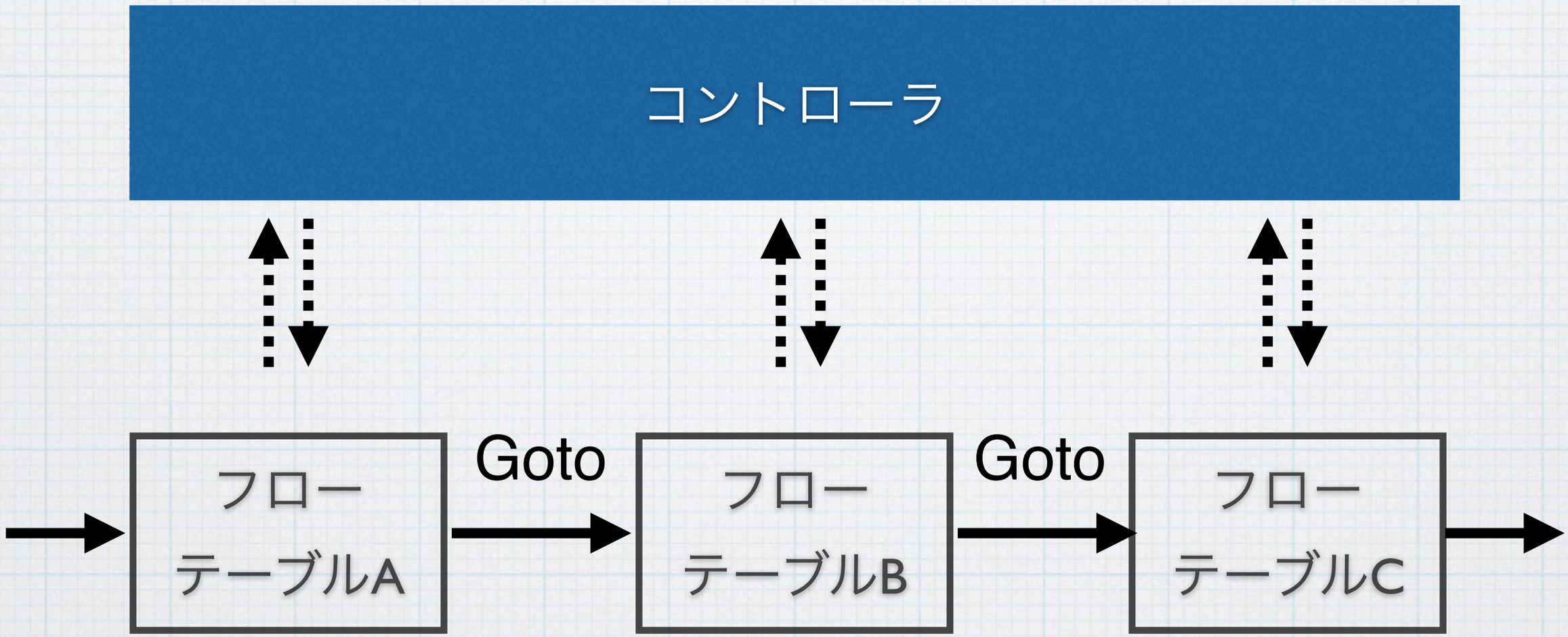
例: ルータ



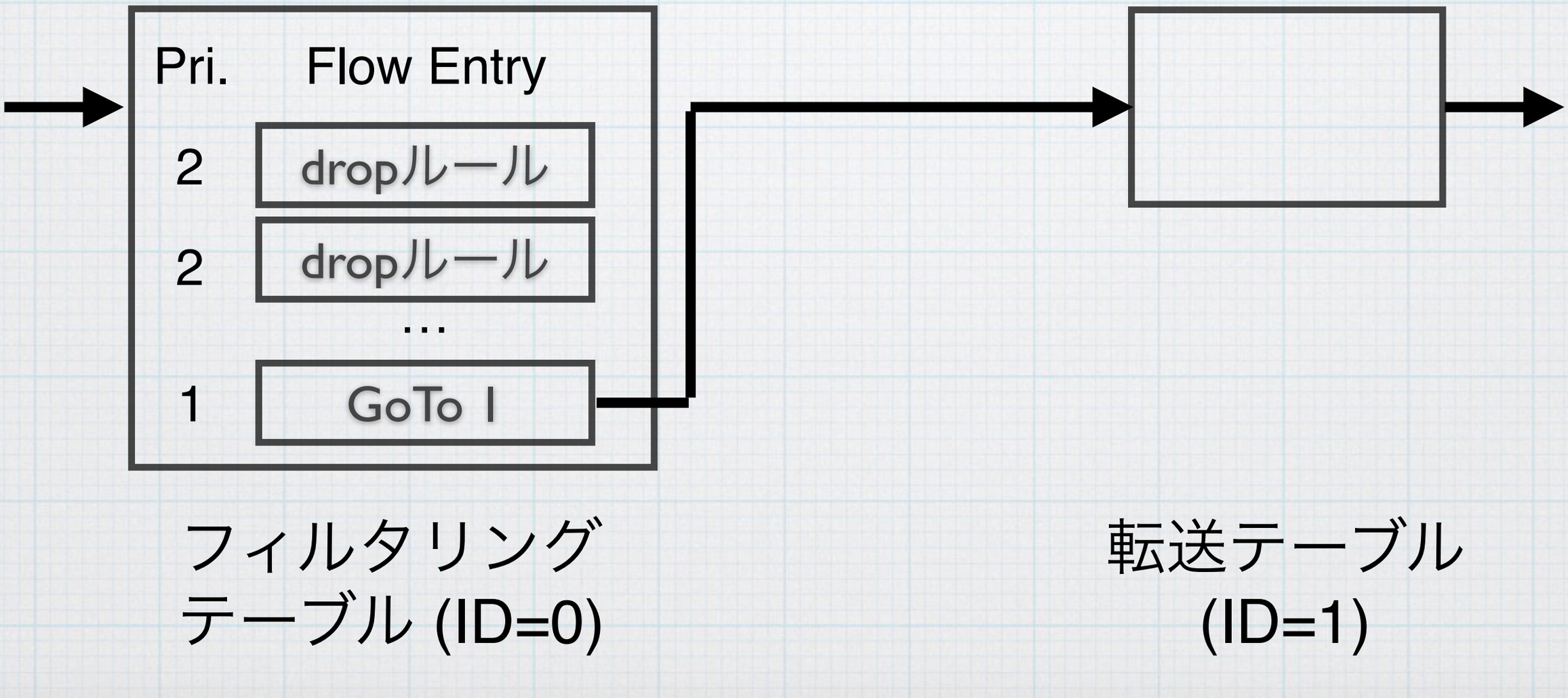
送信元/宛先
MACアドレス

OpenFlow1.3でのパケット処理

マルチプルテーブル



ラーニングスイッチ



フィルタリング
テーブル (ID=0)

転送テーブル
(ID=1)

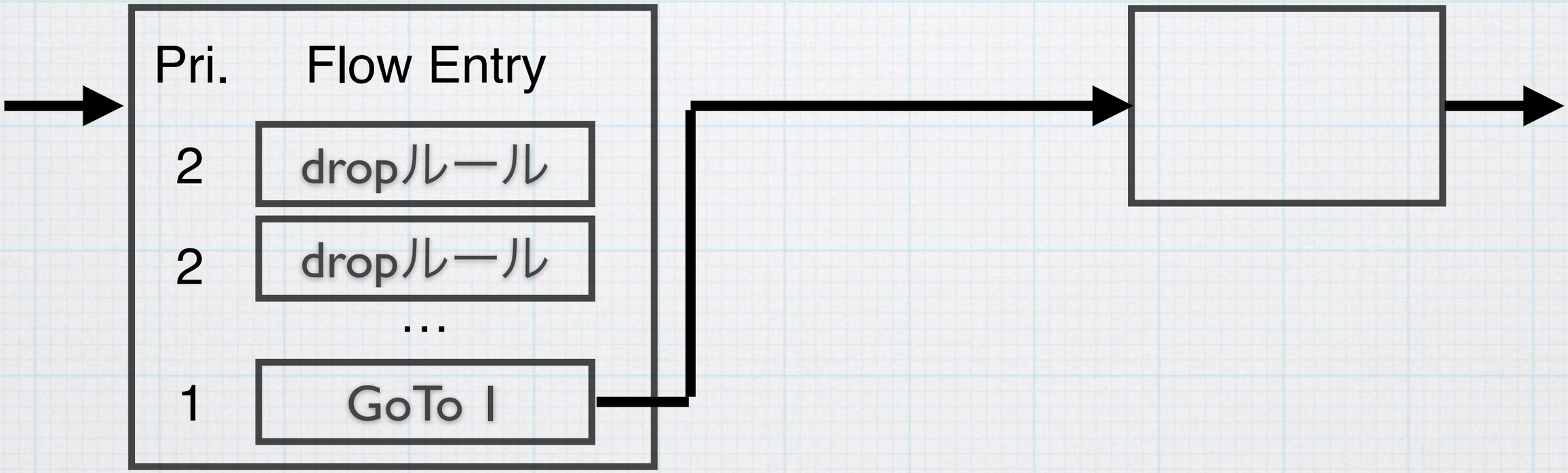
```
send_flow_mod_add(
      datapath_id,
      table_id: 0,
      idle_timeout: 0,
      priority: 2,
      match: Match.new(ether_destination_address: '01:00:5e:00:00:00',
                       ether_destination_address_mask: 'ff:ff:ff:00:00:00')
)
```

・マルチキャストを落とす
・最初のテーブル（ID=0）で処

```
send_flow_mod_add(
      datapath_id,
      table_id: 0,
      idle_timeout: 0,
      priority: 1,
      match: Match.new,
      instructions: GotoTable.new(1)
)
```

- フィルタリング（pri=2）の後にこのルール（pri=1）にヒット
- 処理をテーブル1へ移行

ラーニングスイッチ(再掲)



フィルタリング
テーブル (ID=0)

転送テーブル
(ID=1)

コントローラ

PacketIn

Pri.

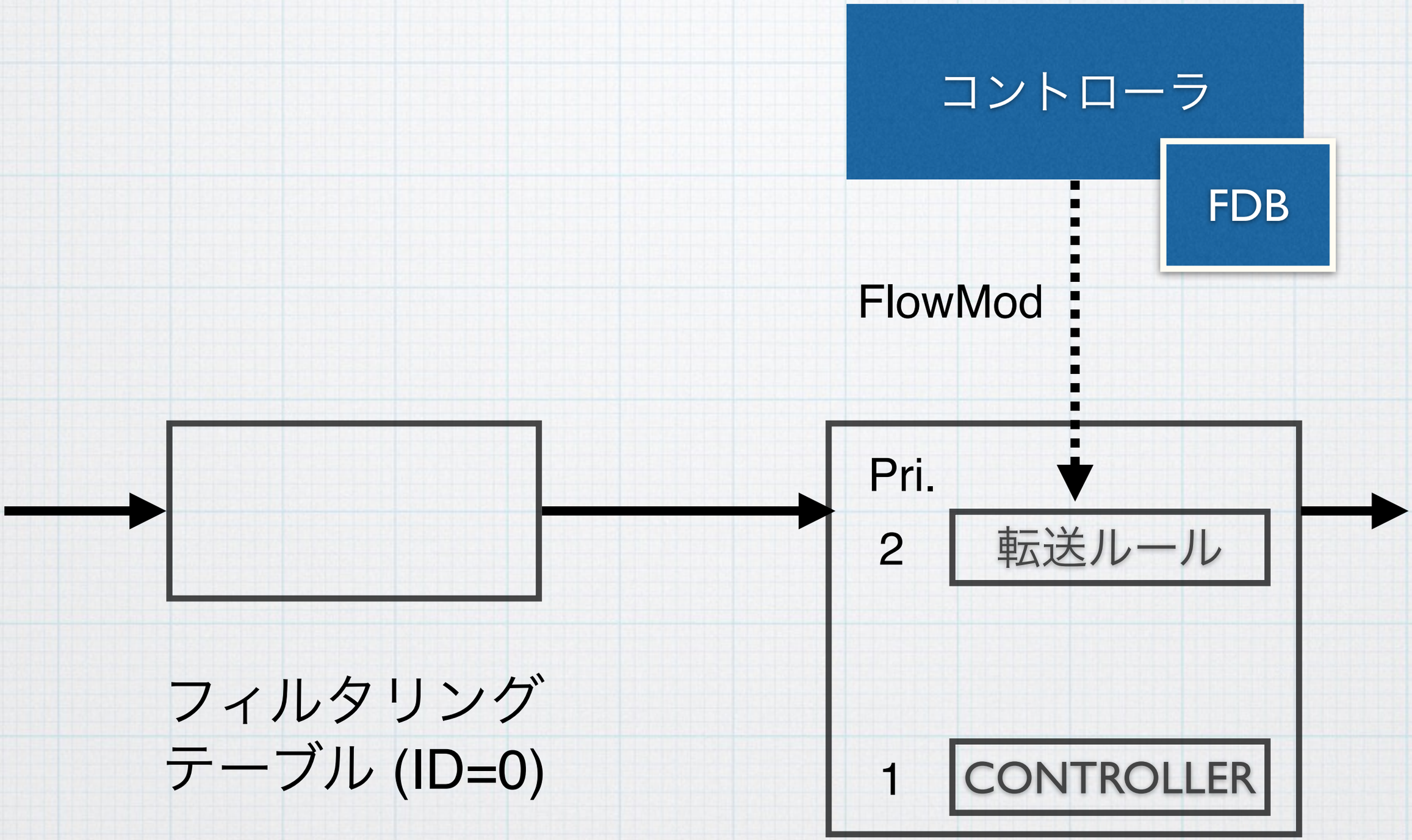
1 CONTROLLER

フィルタリング
テーブル (ID=0)

転送テーブル
(ID=1)

```
send_flow_mod_add(
  datapath_id,
  table_id: 1,
  idle_timeout: 0,
  priority: 1,
  match: Match.new,
  instructions: Apply.new(SendOutPort.new(:controller))
)
```

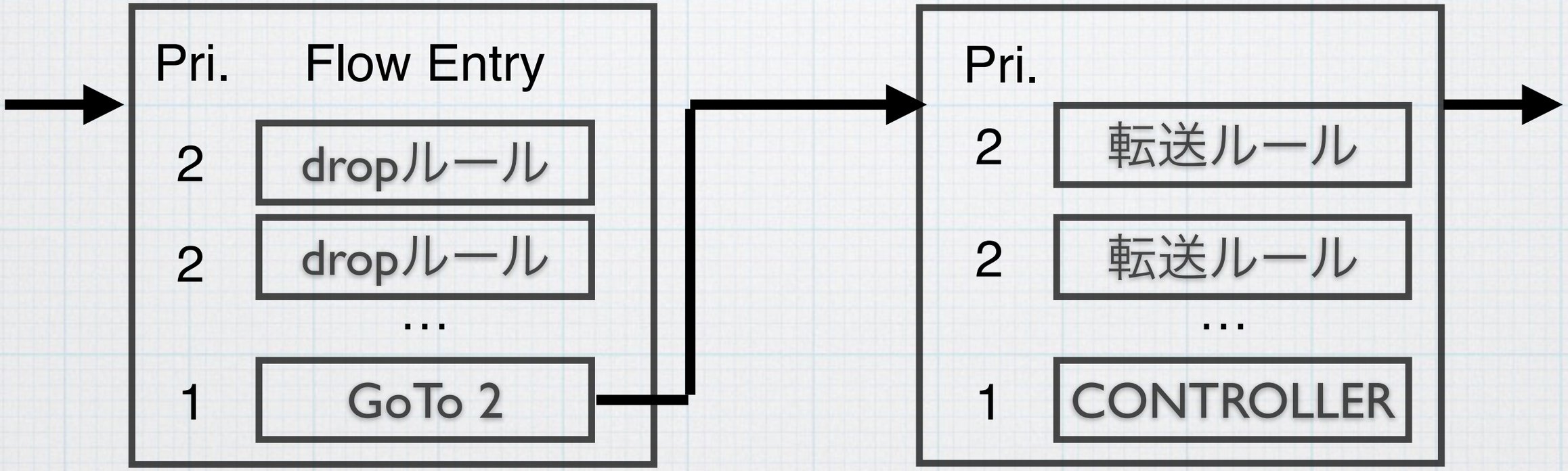
・明示的にPacketInを起こす


コントローラ
FDB
FlowMod
Pri.
2
転送ルール
1
CONTROLLER
フィルタリング
テーブル (ID=0)
転送テーブル
(ID=1)

```
send_flow_mod_add(
      datapath_id,
      table_id: FORWARDING_TABLE_ID,
      idle_timeout: AGING_TIME,
      priority: 2,
      match: Match.new(in_port: packet_in.in_port,
                       ether_destination_address: packet_in.destination_mac,
                       ether_source_address: packet_in.source_mac),
    instructions: Apply.new(SendOutPort.new(port_no))
  )
```

- packet_inの送信元+ポートをフローエントリとして追加
- 優先度はPacketInより高い

コントローラ



フィルタリング
テーブル (ID=0)

転送テーブル
(ID=1)