

OpenFlow1.3の利点

OpenFlow 1.0の問題点 I

Flow Table

dropルール1

dropルール2

dropルール3

..

書き換えルール1

書き換えルール2

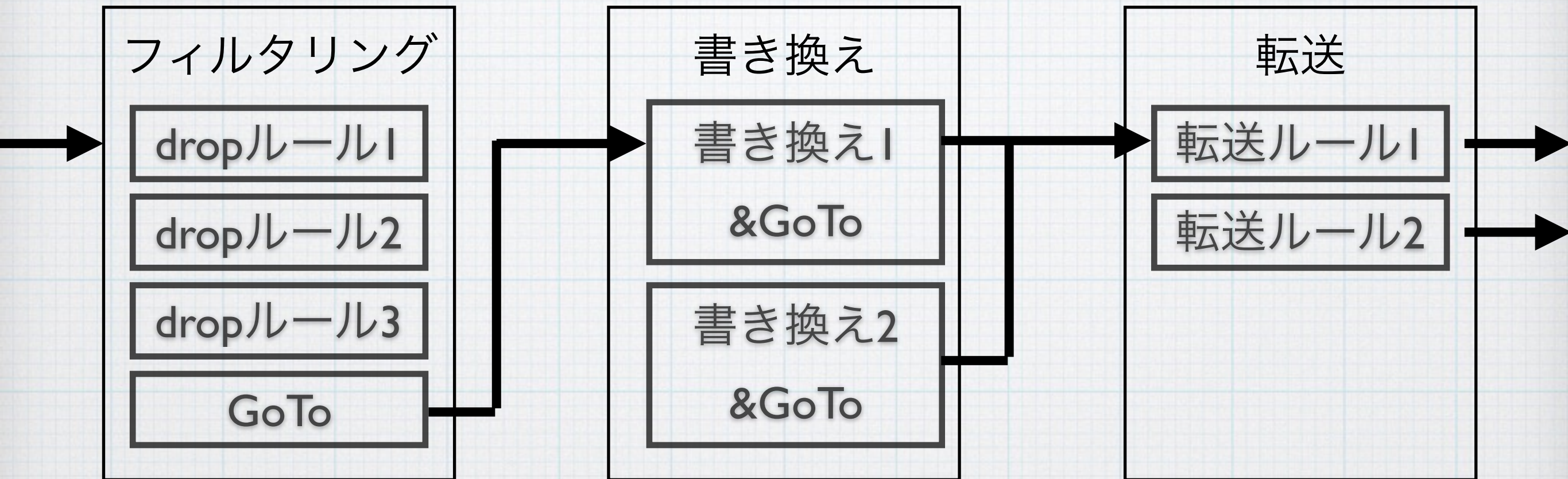
..

転送ルール1

転送ルール2

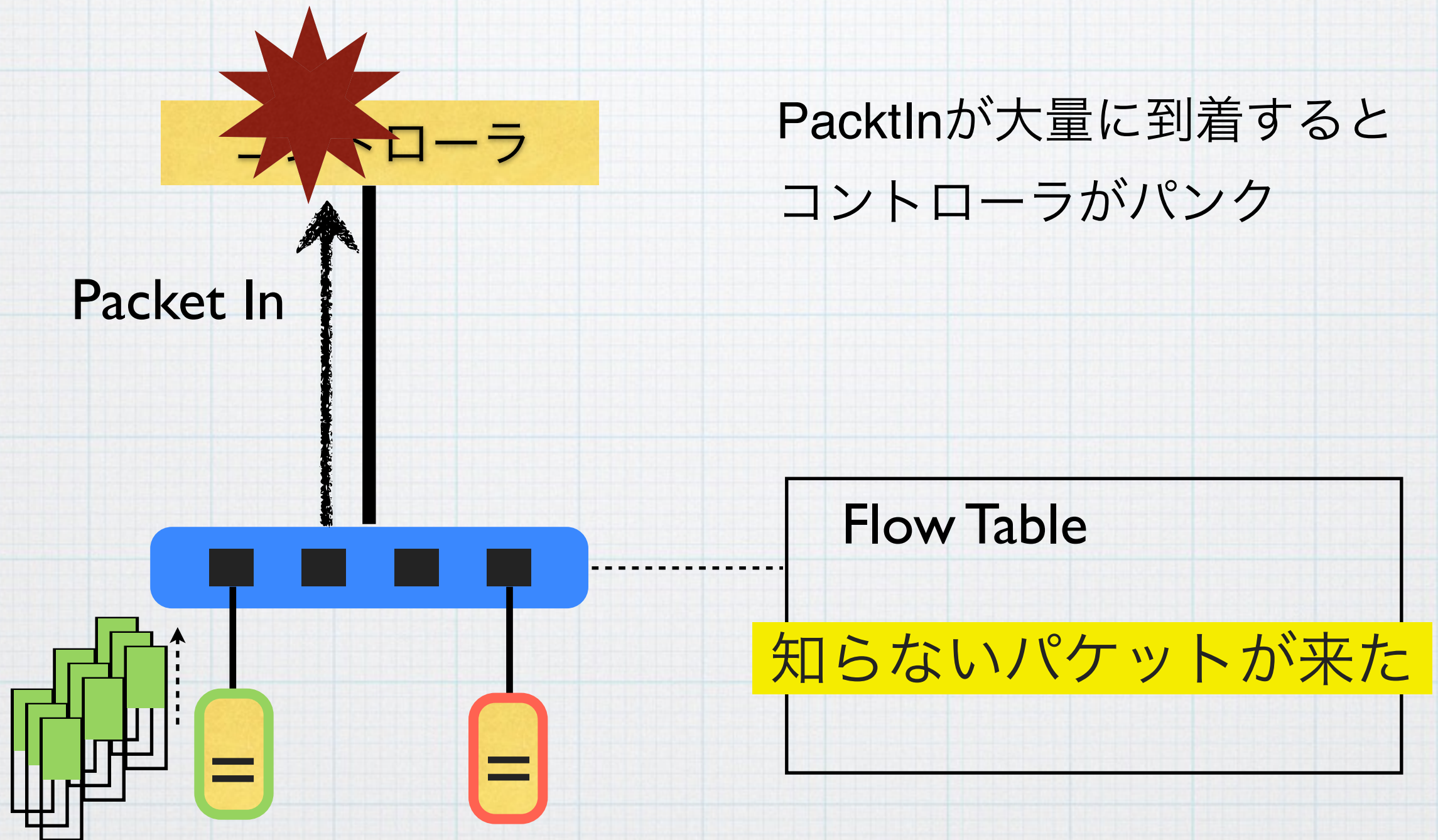
- OpenFlow 1.0 では、フローテーブルは一つだけ
- 役割の違うエントリが混在 & 混乱しがち

OpenFlow 1.3では

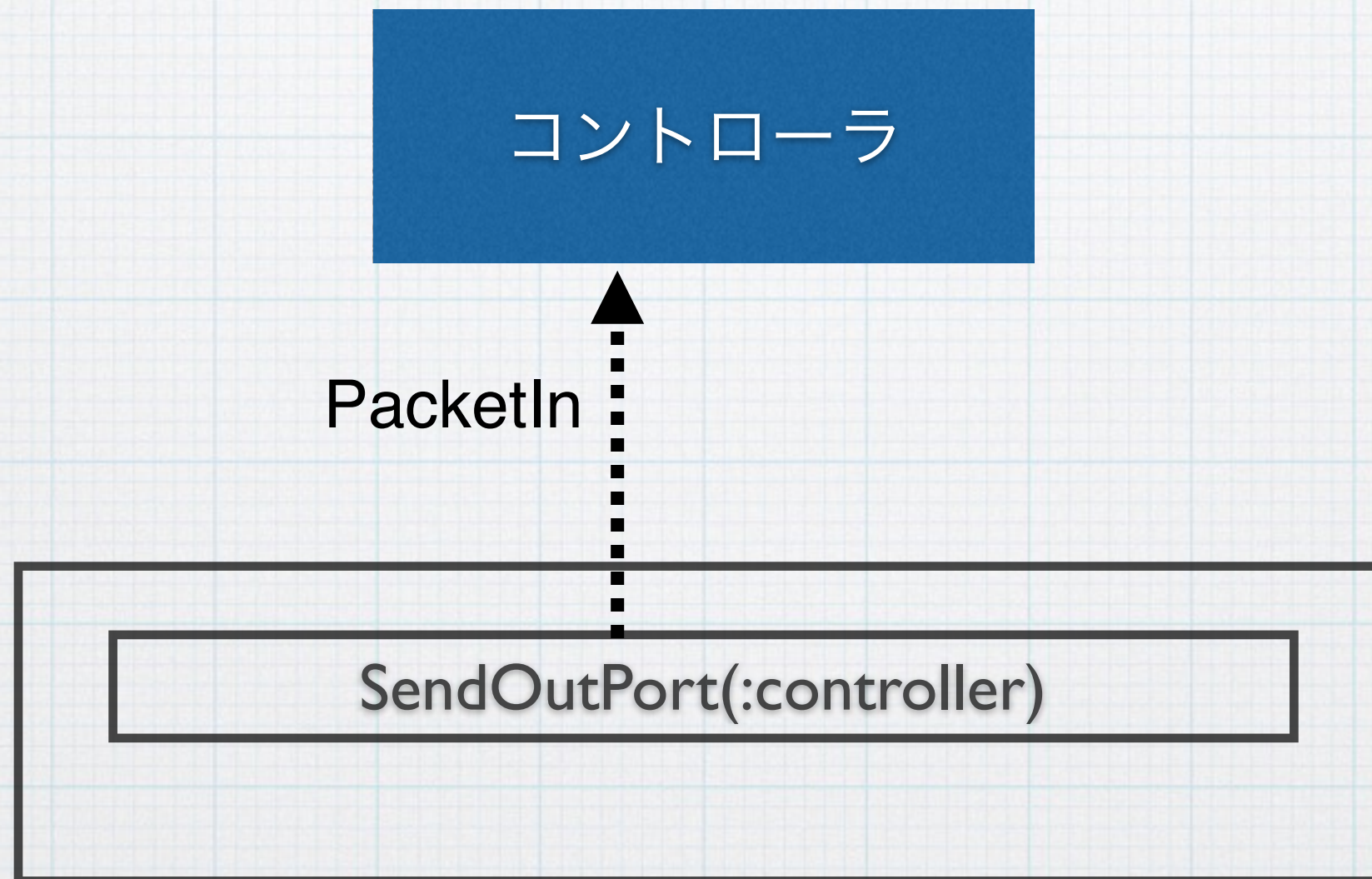


- 分けてスッキリ!
- 処理をパイプライン化できるので性能が上がる

OpenFlow 1.0の問題点 2



OpenFlow 1.3では



- 明示的に指定しない限り、PacketInは起こらない

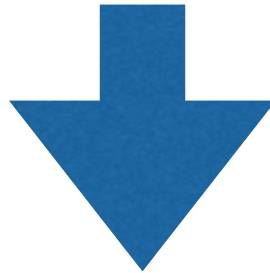
アクションと インストラクション

- アクション
 - パケットの書き換えと転送 (OFI.0と同じ)
 - OFI.3で種類が増えました
- インストラクション
 - テーブルの移動 (GoTo)
 - アクションの実行方法の指定

```
send_flow_mod_add(  
    datapath_id,  
    table_id: 1,  
    idle_timeout: 0,  
    priority: 1,  
    match: Match.new,  
    instructions: GotoTable.new(2)  
)
```

- ・ 処理をテーブル2へ移行


```
send_flow_mod_add(  
    datapath_id,  
    ...  
    actions: SendOutPort.new(1)  
)
```



```
send_flow_mod_add(  
    datapath_id,  
    ...  
    instructions: Apply.new(SendOutPort.new(1))  
)
```

アクションセット

[]

[A1]

[A1,A2]



- パケットに関連付いたアクション集合
- WriteActionsでアクションを追加
- GoToを含まないエントリにマッチしたら実行

まとめ

- OpenFlow 1.3 版スイッチの仕組み
- マルチプルテーブルの使い方
- アクションとインストラクション

レポート課題

- OpenFlow1.3 版スイッチの動作について、
`trema dump_flows` の出力を混じえながら
各ステップごとに説明しよう
- 実行のしかた

```
$ trema run lib/learning_switch13.rb  
  --openflow13 -c trema.conf
```