

UNIVERSIDADE FEDERAL DE SÃO CARLOS

PROXY ROUTEFLOW BASEADO EM JAVA

Fabiano Silva Mathilde

Orientador: Prof. Dr. Cesar Augusto Cavalheiro Marcondes

Coorientador: Dr. Christian Esteve Rothenberg

Agenda

- Introdução
- Redes Definidas por Software
- Arquitetura Básica do RouteFlow
- Proxy RouteFlow em Java
- Resultados
- Conclusão

Agenda

- **Introdução**
- Redes Definidas por Software
- Arquitetura Básica do RouteFlow
- Proxy RouteFlow em Java
- Resultados
- Conclusão

Introdução

- Objetivo Principal
 - Suporte ao software de controle Floodlight

- Objetivos Secundários
 - Integração com os usuários do Floodlight
 - Testes de desempenho

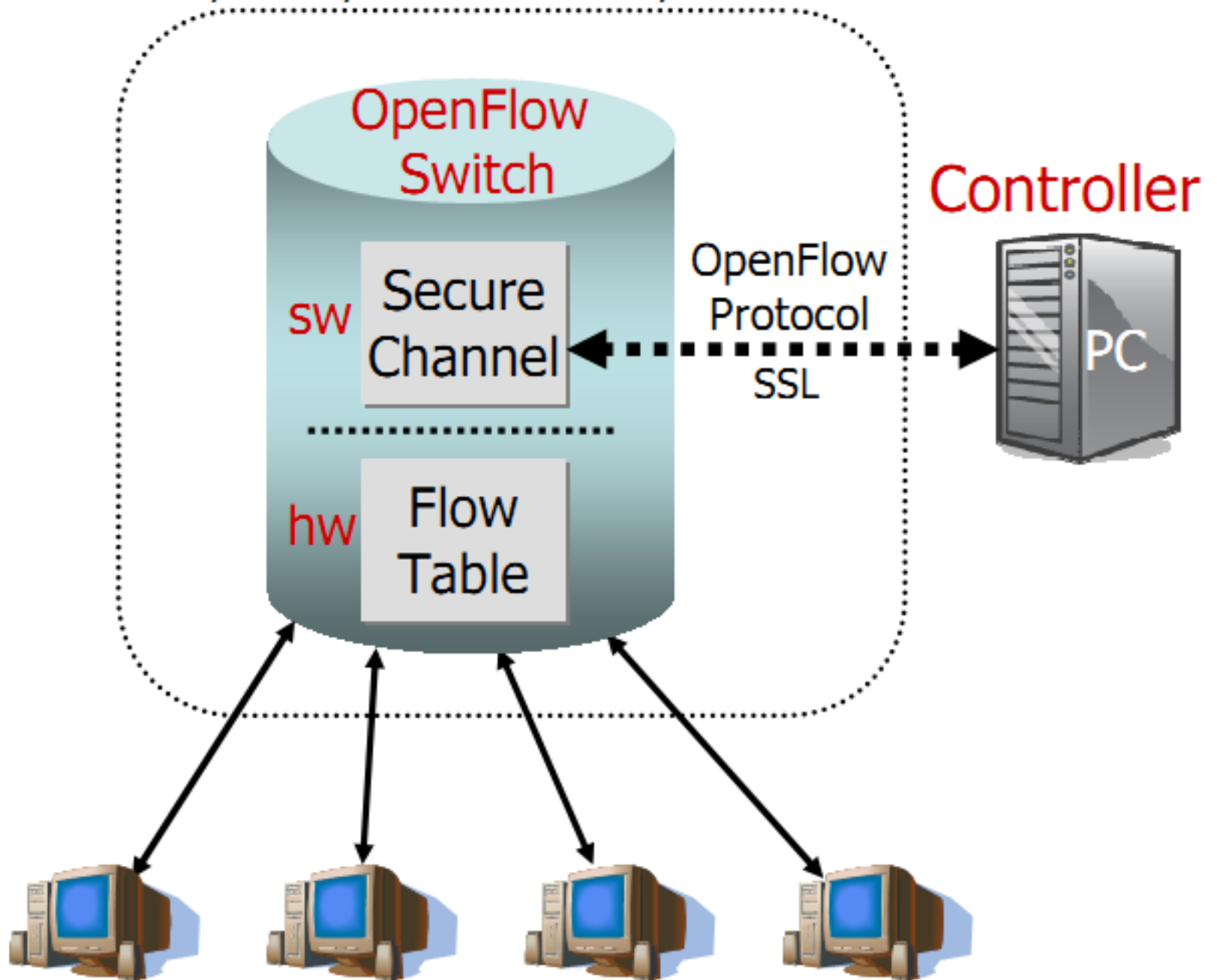
Agenda

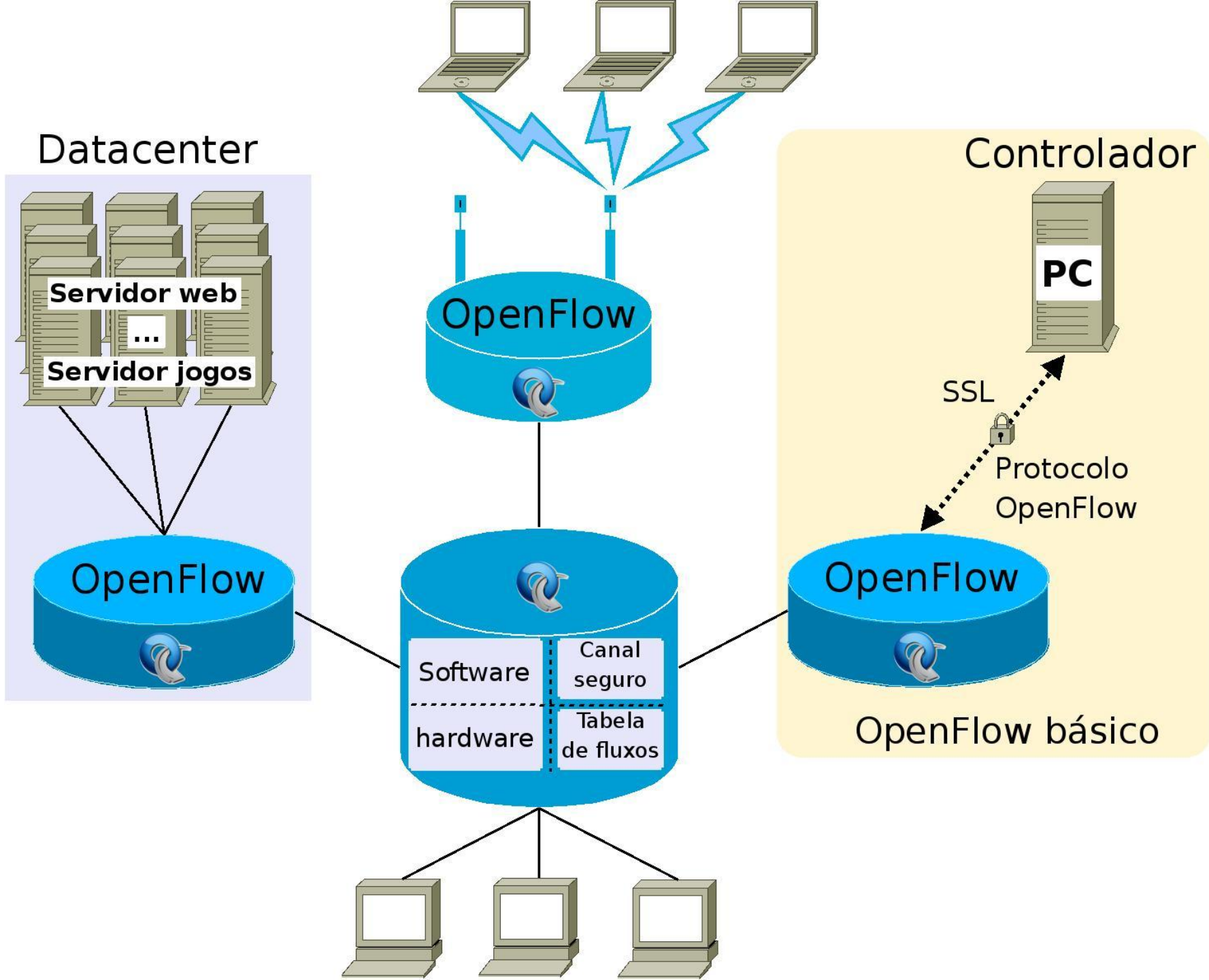
- Introdução
- **Redes Definidas por Software**
- Arquitetura Básica do RouteFlow
- Proxy RouteFlow em Java
- Resultados
- Conclusão

Redes Definidas por Software

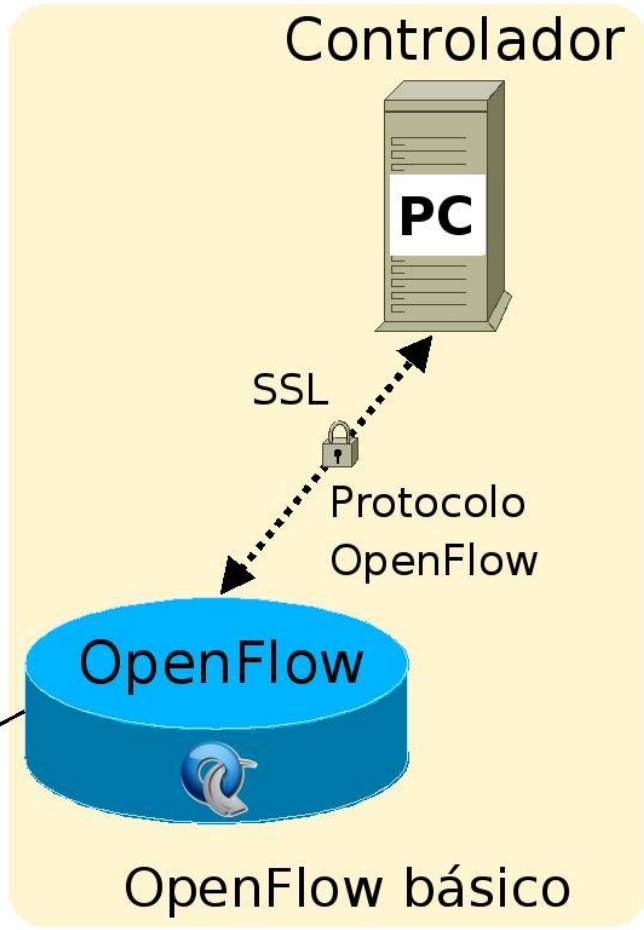
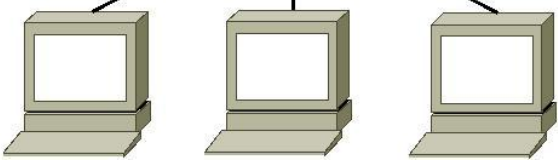
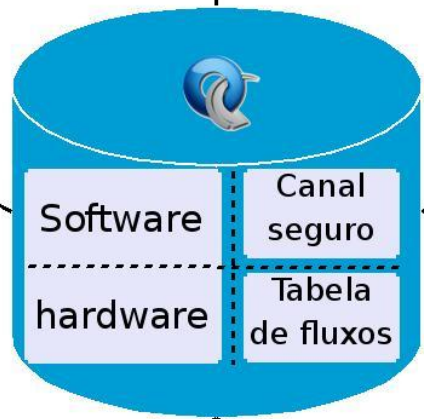
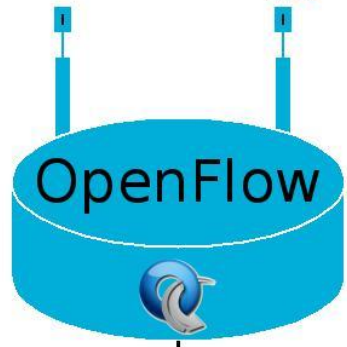
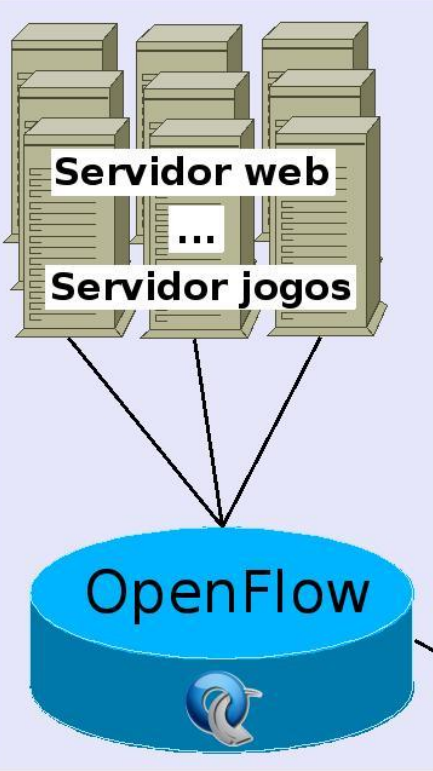
- Abordagem mais bem sucedida:
 - Separação entre o plano de controle e plano de encaminhamento
 - Uso do protocolo OpenFlow

Scope of OpenFlow Switch Specification





Datacenter



Agenda

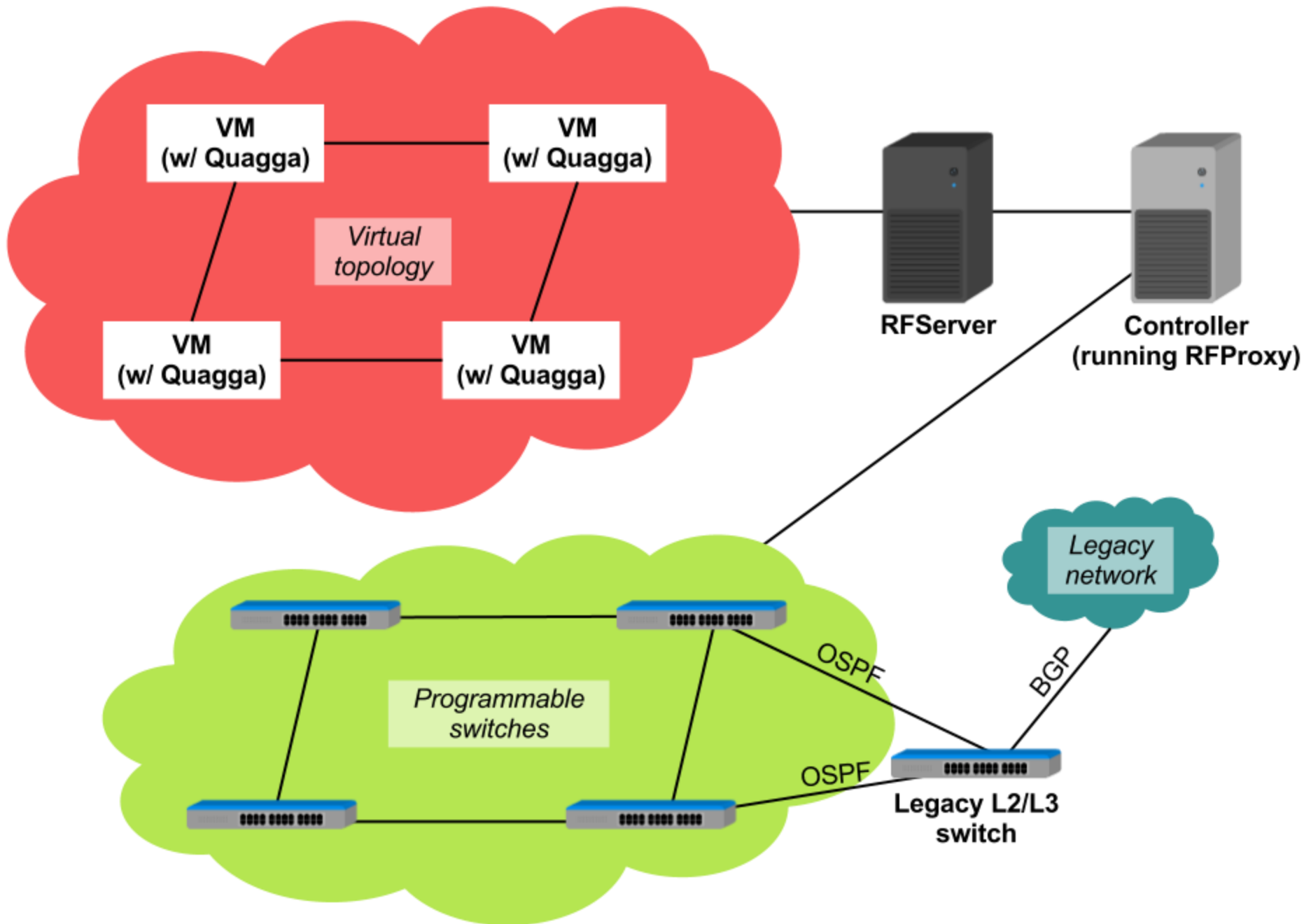
- Introdução
- Redes Definidas por Software
- **Arquitetura Básica do RouteFlow**
- Proxy RouteFlow em Java
- Resultados
- Conclusão

Arquitetura Básica do RouteFlow

- O que é RouteFlow?
 - Oferta de serviços de roteamento IP centralizada
 - Totalmente aberto

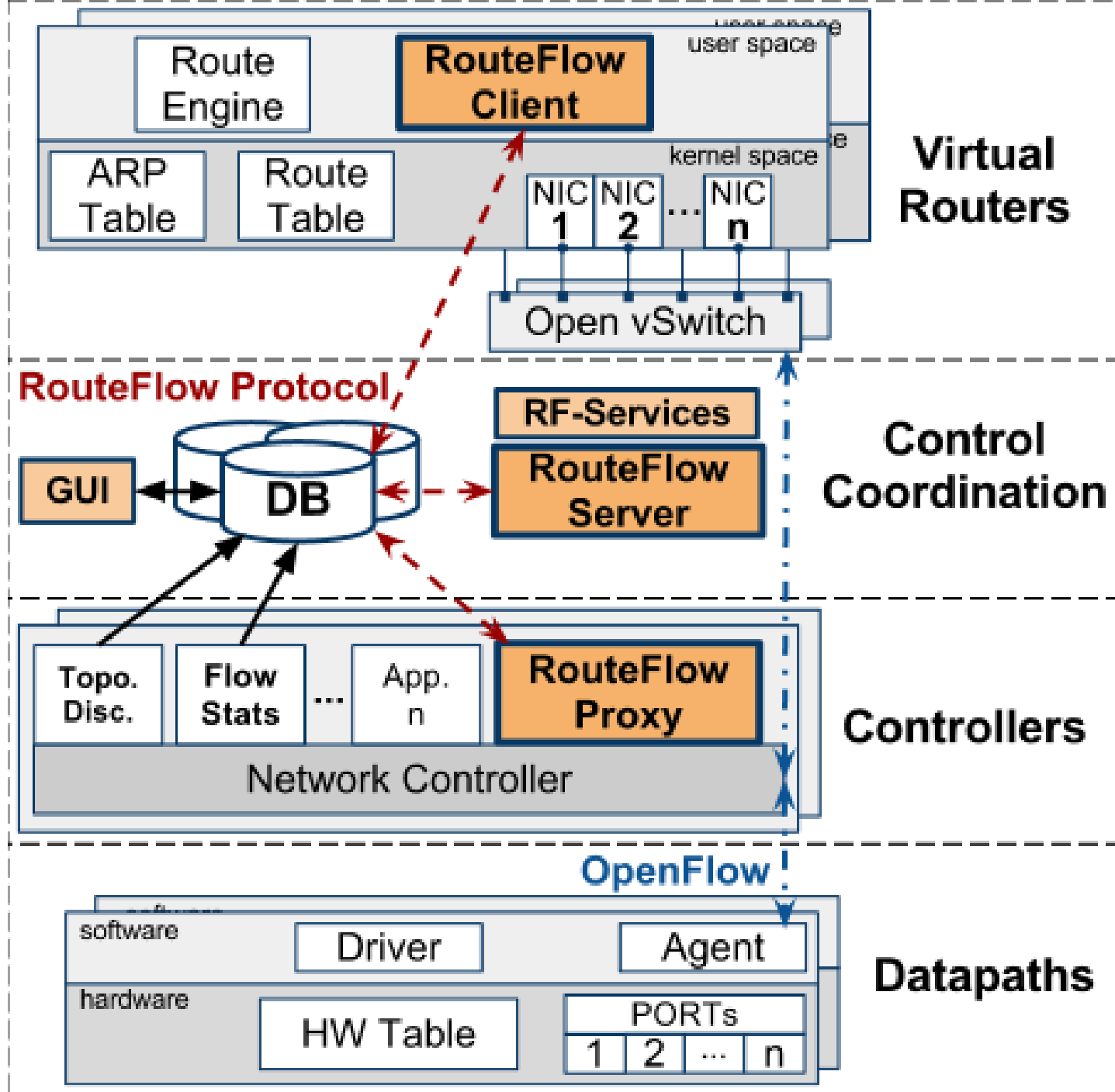
Arquitetura Básica do RouteFlow

- Estrutura Básica:



Arquitetura Básica do RouteFlow

- Estrutura Detalhada:



Arquitetura Básica do RouteFlow

- Componentes
 - Cliente RouteFlow
 - Servidor RouteFlow
 - Proxy RouteFlow

Arquitetura Básica do RouteFlow

- Cliente RouteFlow
 - Opera dentro das máquinas virtuais
 - Recolhe os dados da tabela de roteamento e envia ao Servidor RouteFlow

Arquitetura Básica do RouteFlow

- Servidor RouteFlow
 - Componente principal
 - Define como todo o ambiente irá se comportar de acordo com o mapeamento entre as máquinas virtuais e os switches OpenFlow

Arquitetura Básica do RouteFlow

- Proxy RouteFlow
 - Comunicação entre o ambiente físico e virtual
 - Traduz mensagens RouteFlow em OpenFlow e vice versa
 - Mapeamento dos eventos que vão ocorrendo no ambiente físico
 - Conta tudo para o Servidor RouteFlow

Agenda

- Introdução
- Redes Definidas por Software
- Arquitetura Básica do RouteFlow
- **Proxy RouteFlow em Java**
- Resultados
- Conclusão

Proxy RouteFlow em Java

- Totalmente escrito em Java
- Baseado no software de controle Floodlight
- Aplicação em sistemas de larga escala

Proxy RouteFlow em Java

- Etapas do Trabalho:
 1. Criação de um sistema de comunicação com o banco de dados Mongo.
 2. Criação do mecanismo de troca de mensagens baseado no banco de dados centralizado.
 3. Estudo aprofundado do controlador Floodlight.
 4. Criação do módulo de controle principal.

Proxy RouteFlow em Java

- Etapas do Trabalho:
 1. Criação de um sistema de comunicação com o banco de dados Mongo.
 2. Criação do mecanismo de troca de mensagens baseado no banco de dados centralizado.
 3. Estudo aprofundado do controlador Floodlight.
 4. Criação do módulo de controle principal.

Proxy RouteFlow em Java

- Não SQL
- Coleção ao invés de tabelas
 - Estrutura definida pelo padrão JSON

Proxy RouteFlow em Java

Conceito de envelopes de dados:

```
{  
  "name" : "MongoDB",  
  "type" : "database",  
  "count" : 1,  
  "info" : { x : 203, y : 102 }  
}
```


Proxy RouteFlow em Java

- Manipulação direta dos dados:

```
BasicDBObject doc = new BasicDBObject("name",  
"MongoDB")
```

```
.append("type", "database")
```

```
.append("count", 1)
```

```
.append("info", new BasicDBObject("x", 203)
```

```
.append("y", 102));
```

```
coll.insert(doc);
```

Proxy RouteFlow em Java

- Etapas do Trabalho:
 1. Criação de um sistema de comunicação com o banco de dados Mongo.
 2. Criação do mecanismo de troca de mensagens baseado no banco de dados centralizado.
 3. Estudo aprofundado do controlador Floodlight.
 4. Criação do módulo de controle principal.

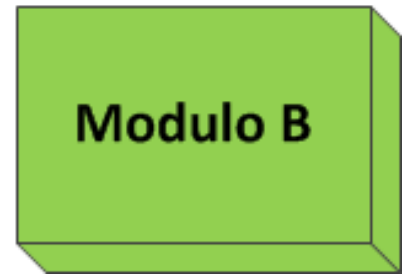
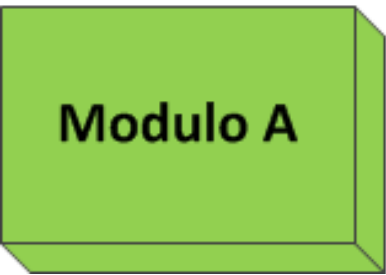
Proxy RouteFlow em Java

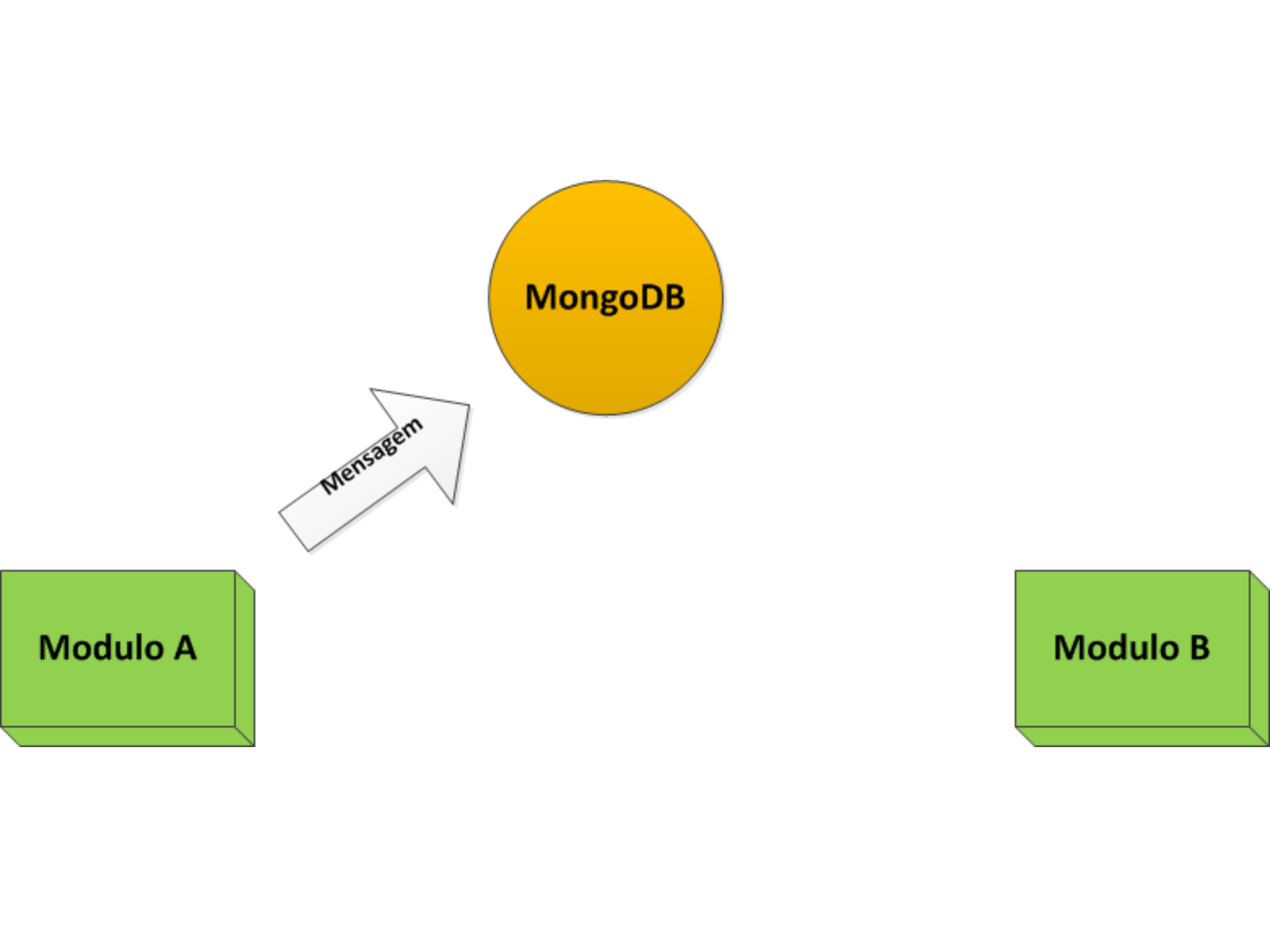
- Envelope básico para mensagem RouteFlow

```
{  
    "FROM_FIELD" : "origem",  
    "TO_FIELD" : "destino",  
    "TYPE_FIELD" : "tipo de mensagem",  
    "READ_FIELD" : "falso ou verdadeiro",  
    "CONTENT_FIELD" : envelope de mensagem  
}
```

Proxy RouteFlow em Java

- Exemplo de Operação:
 - Módulo A enviando mensagem ao Módulo B





Modulo A

MongoDB

Mensagem

Modulo B

Proxy RouteFlow em Java

- Campos Básicos:

```
{
```

```
    "FROM_FIELD" : "A",
```

```
    "TO_FIELD" : "B",
```

```
    "TYPE_FIELD" : "tipo de mensagem",
```

```
    "READ_FIELD" : "falso",
```

```
    "CONTENT_FIELD" : envelope de mensagem
```

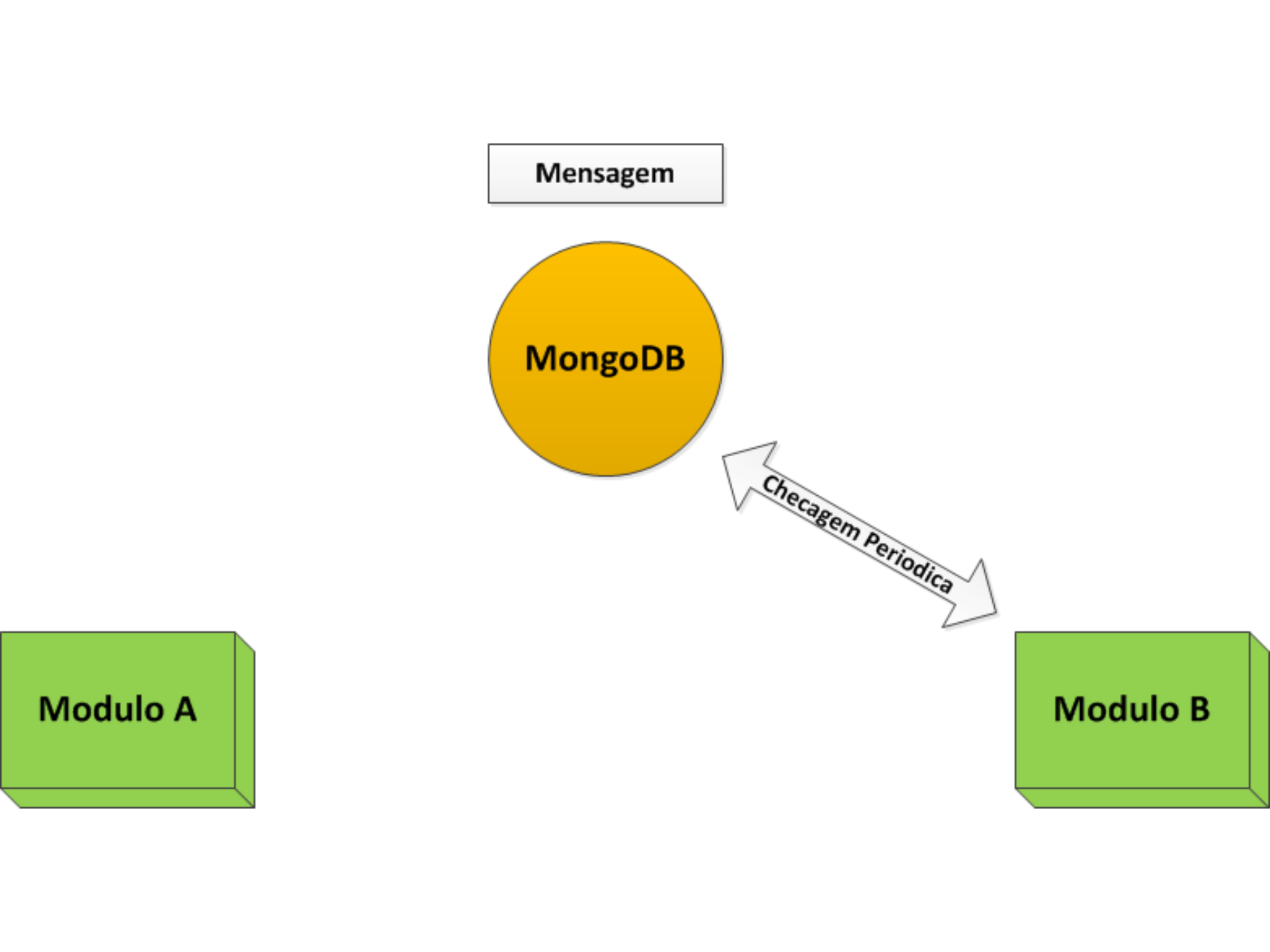
```
}
```

Proxy RouteFlow em Java

- Todo módulo checa a cada 50ms se chegou mensagens

```
query.put(fields.TO_FIELD, this.get_id());  
query.put(fields.READ_FIELD, false);
```

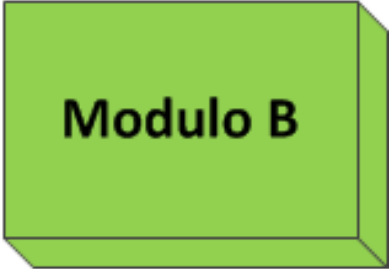
```
DBCursor cursor = collection.find(query);
```

Proxy RouteFlow em Java

- Extraí os campos da mensagem e atualiza o campo de leitura para verdadeiro

Mensagem



Mensagem

Proxy RouteFlow em Java

- As mensagens antigas nunca são apagadas, mantendo o passado do ambiente

Mensagem 1

Mensagem 2

Mensagem 3

Mensagem N

MongoDB

Modulo A

Modulo B

Proxy RouteFlow em Java

- Etapas do Trabalho:
 1. Criação de um sistema de comunicação com o banco de dados Mongo.
 2. Criação do mecanismo de troca de mensagens baseado no banco de dados centralizado.
 3. Estudo aprofundado do controlador Floodlight.
 4. Criação do módulo de controle principal.

Proxy RouteFlow em Java

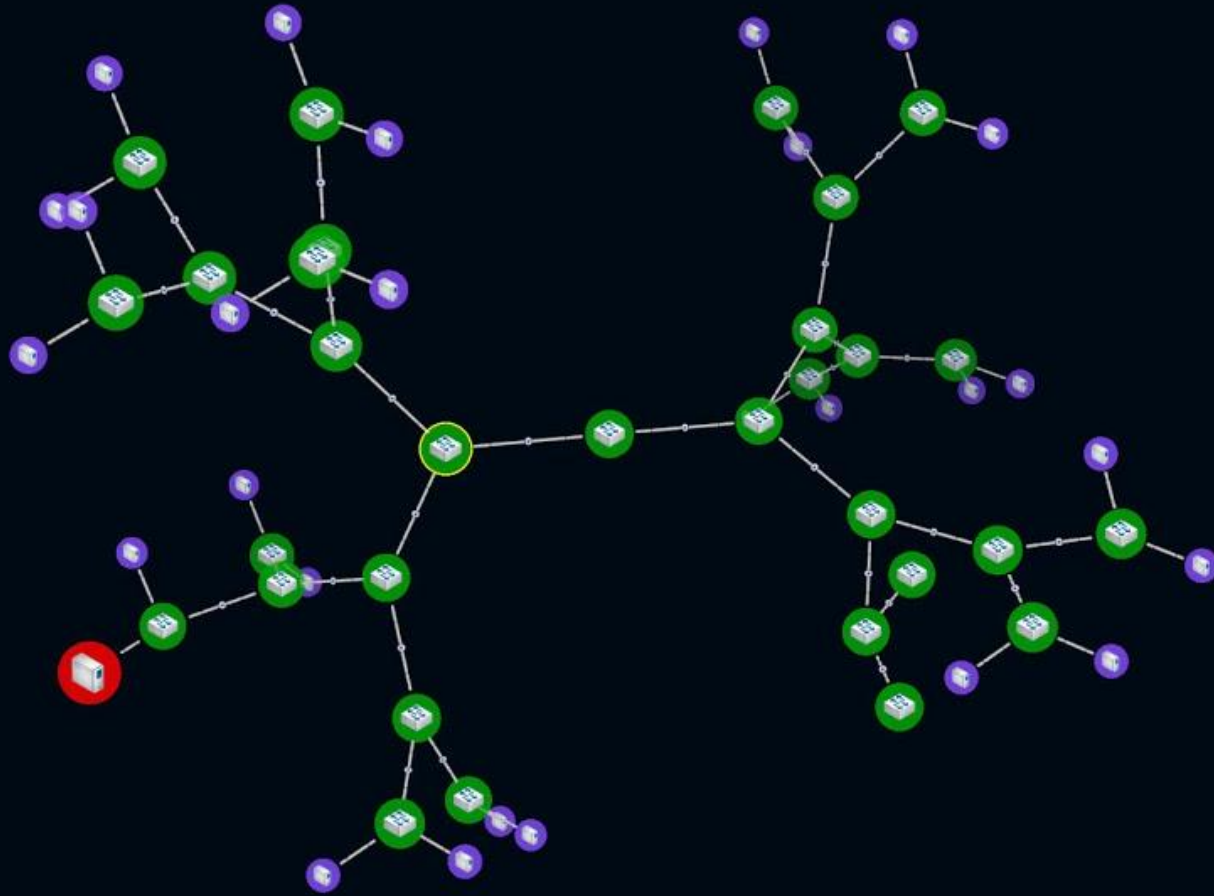
- Controlador Floodlight
 - Herdeiro do Beacon
 - 100% Java
 - Totalmente modularizado
 - Mantido oficialmente pela Big Switch Networks
 - Construído para ter alto desempenho
 - Suporte a tecnologia REST

Proxy RouteFlow em Java

- Criação de regras via REST API:
 - curl -d
'{"switch": "00:00:00:00:00:00:00:01", "name": "flow-mod-1",
"priority": "32768", "ingress-port": "1", "active": "true",
"actions": "output=2"}'
http://<controller_ip>:8080/wm/staticflowentrypusher/json
 - curl -X DELETE -d '{"name": "flow-mod-1"}'
http://<controller_ip>:8080/wm/staticflowentrypusher/json

Proxy RouteFlow em Java

- Ferramentas gráficas



Attributes ✖

Property	Value
datapathDescription	None
datasource	OpenFlow
foreignSourceId	00:00:00:00:00:00:22
hardwareDescription	Open vSwitch
length	1056
manufacturerDescription	Nicra Networks, Inc.
serialNumber	None
softwareDescription	1.1.2

Attributes Clusters Filter Explorer Filters

127.0.0.1 : 8080 | Iteration 67 (rev11988) | 123 nodes 88 links

Proxy RouteFlow em Java

- Etapas do Trabalho:
 1. Criação de um sistema de comunicação com o banco de dados Mongo.
 2. Criação do mecanismo de troca de mensagens baseado no banco de dados centralizado.
 3. Estudo aprofundado do controlador Floodlight.
 4. Criação do módulo de controle principal.

Proxy RouteFlow em Java

- Inserido como um módulo externo no software de controle Floodlight

Proxy RouteFlow em Java

Memory Storage

Provider

Device Manager

Flow Cache

Debug Interface

REST API

Counter Store

Proxy RouteFlow em Java

Memory Storage

Provider

Device Manager

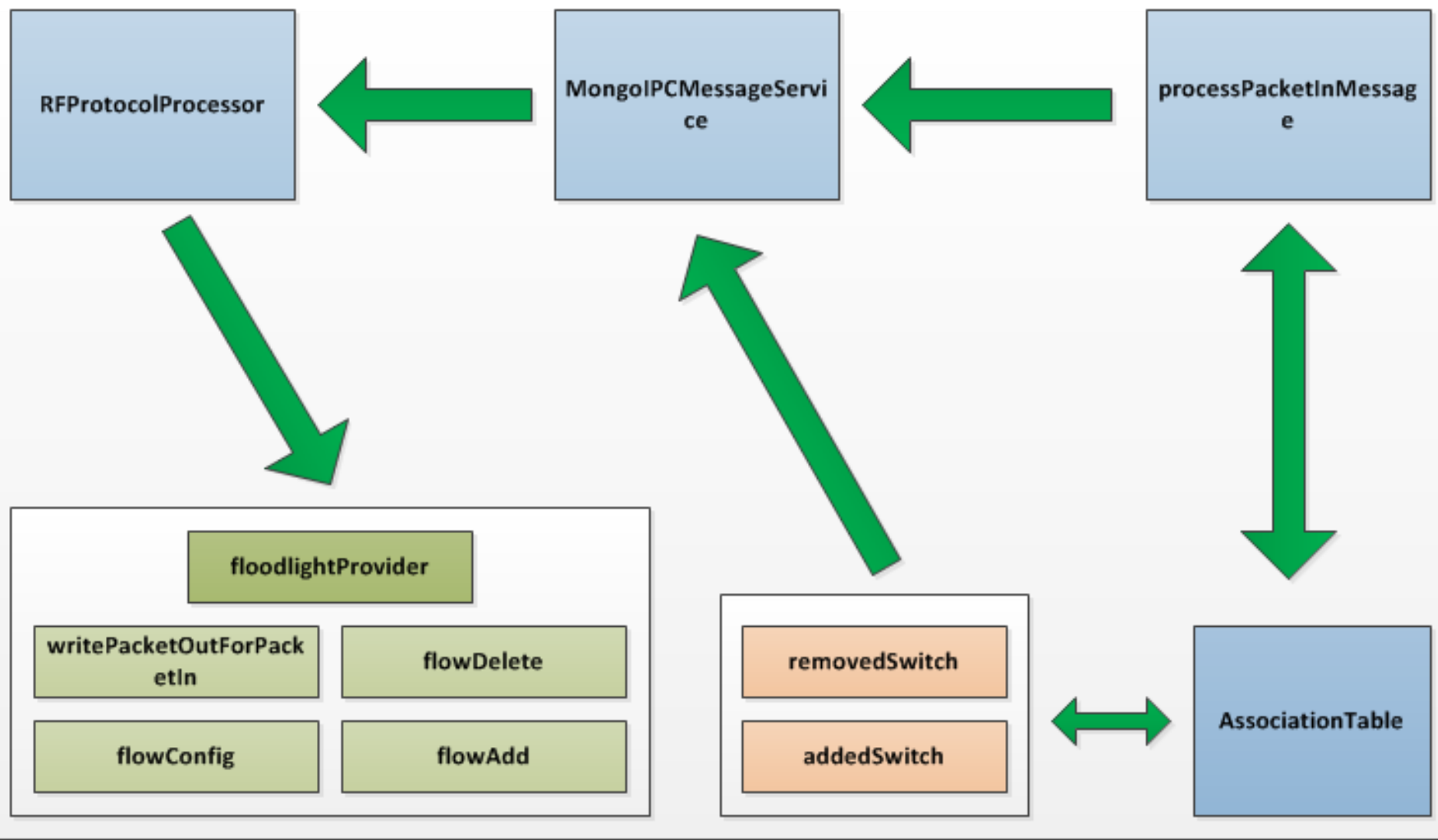
Flow Cache

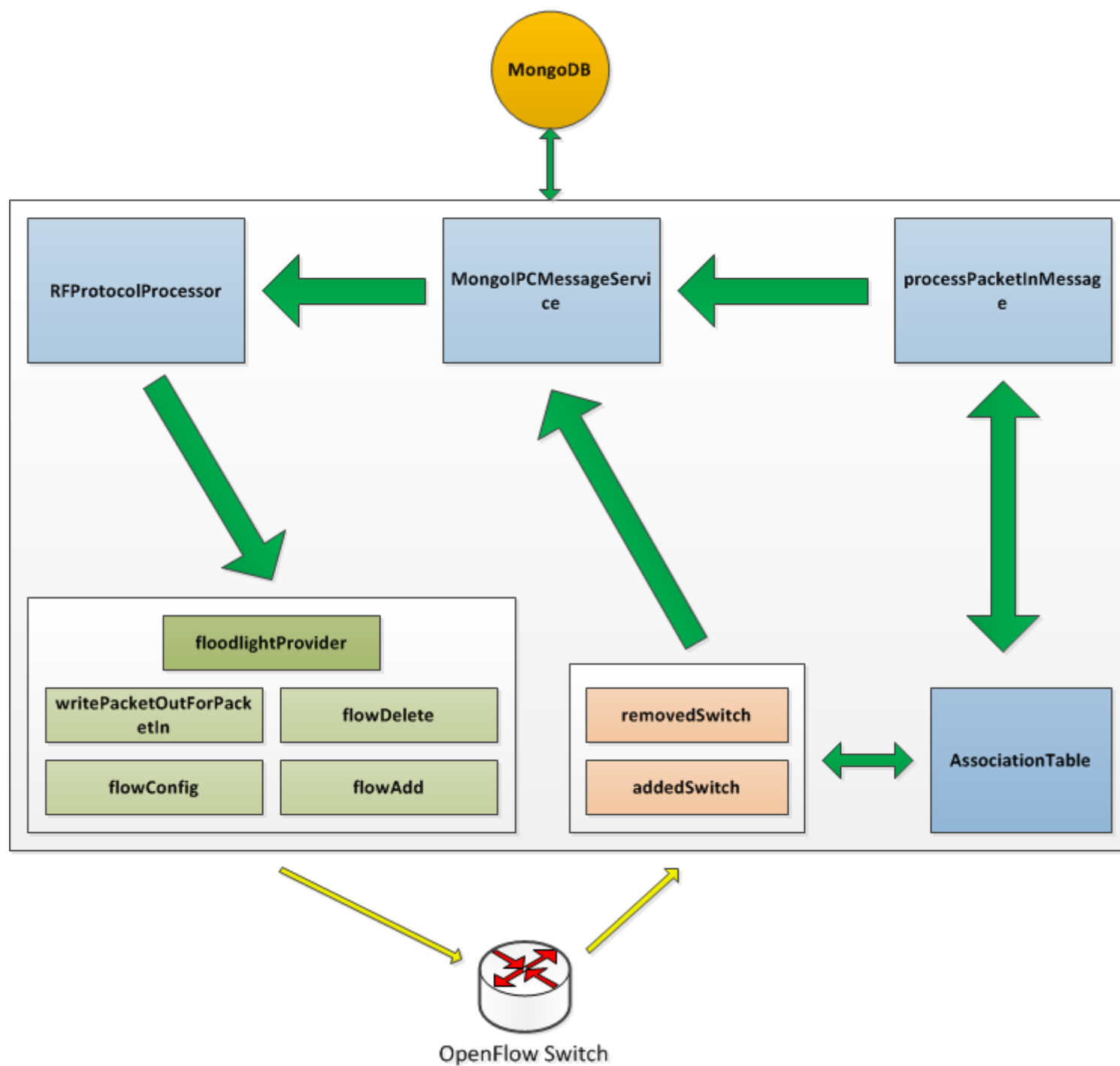
Debug Interface

RFProxy

REST API

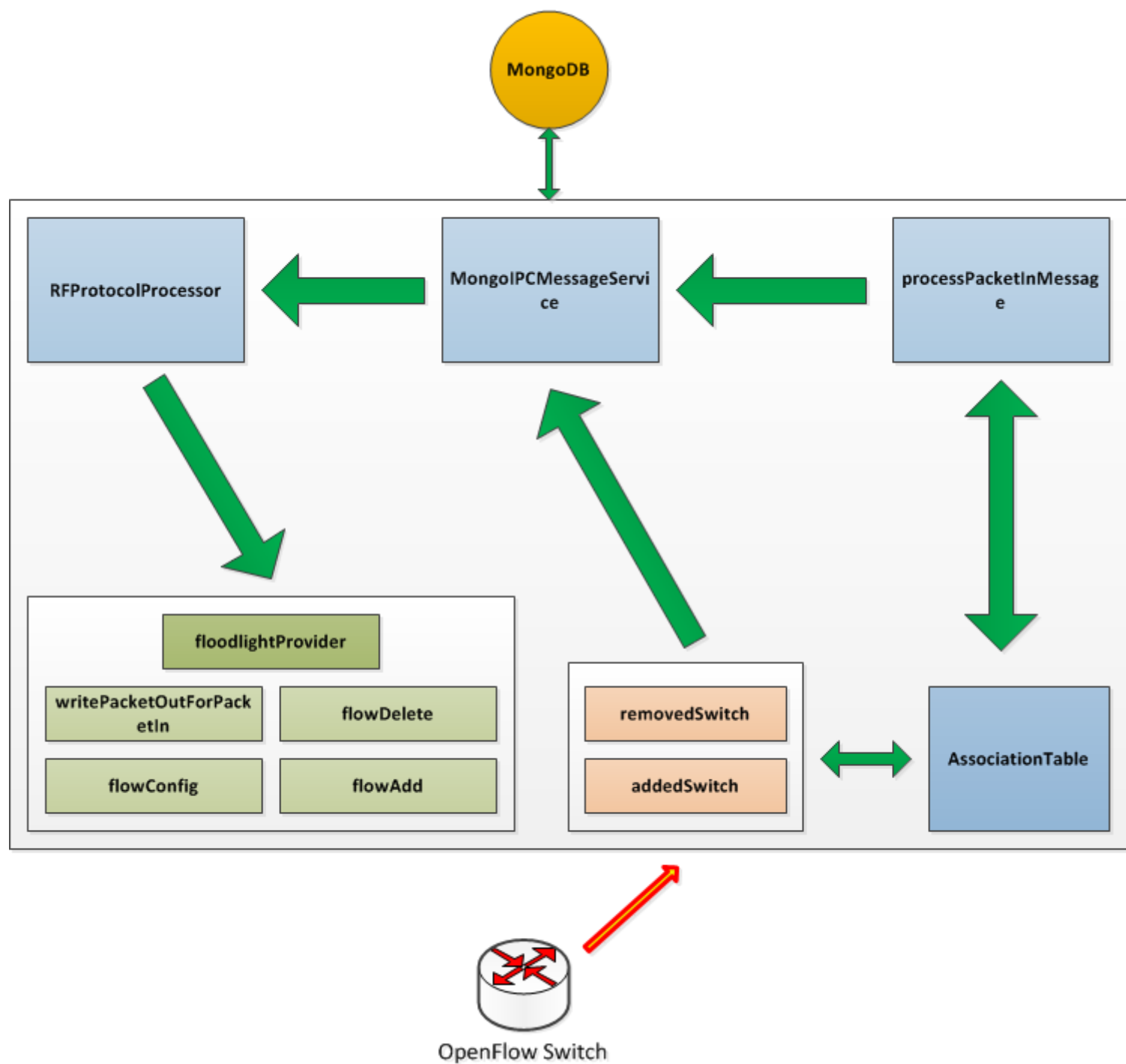
Counter Store

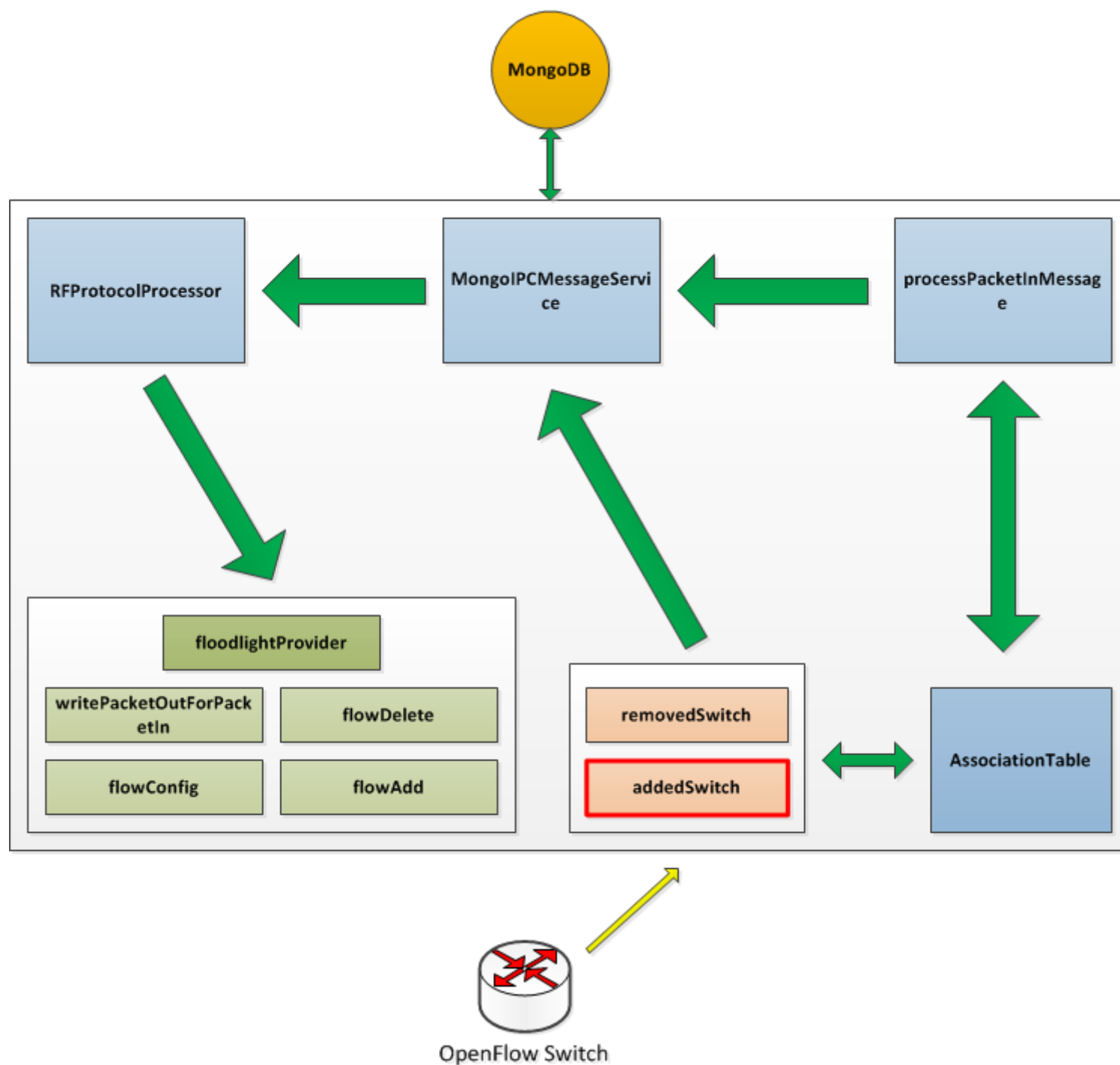


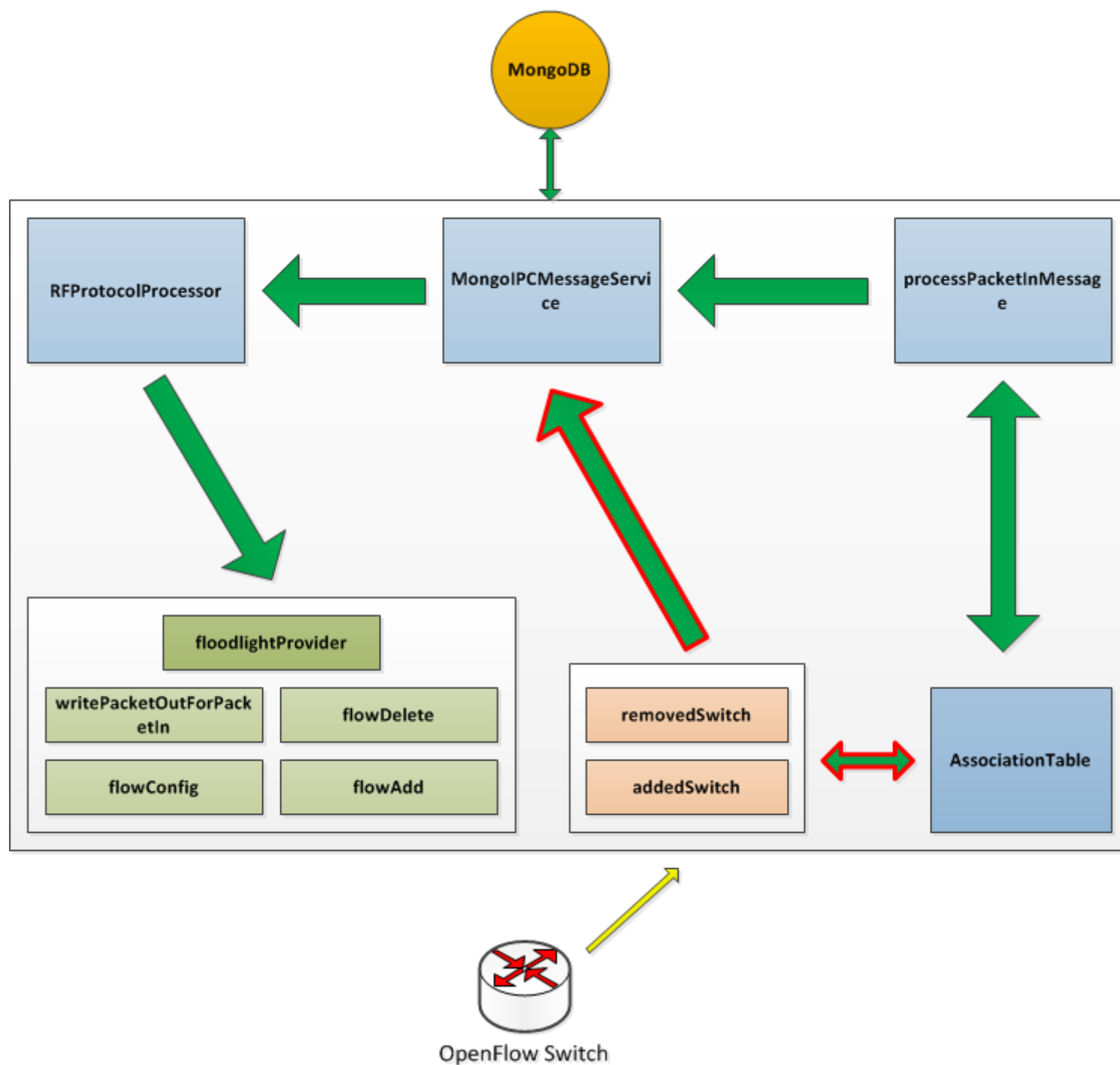


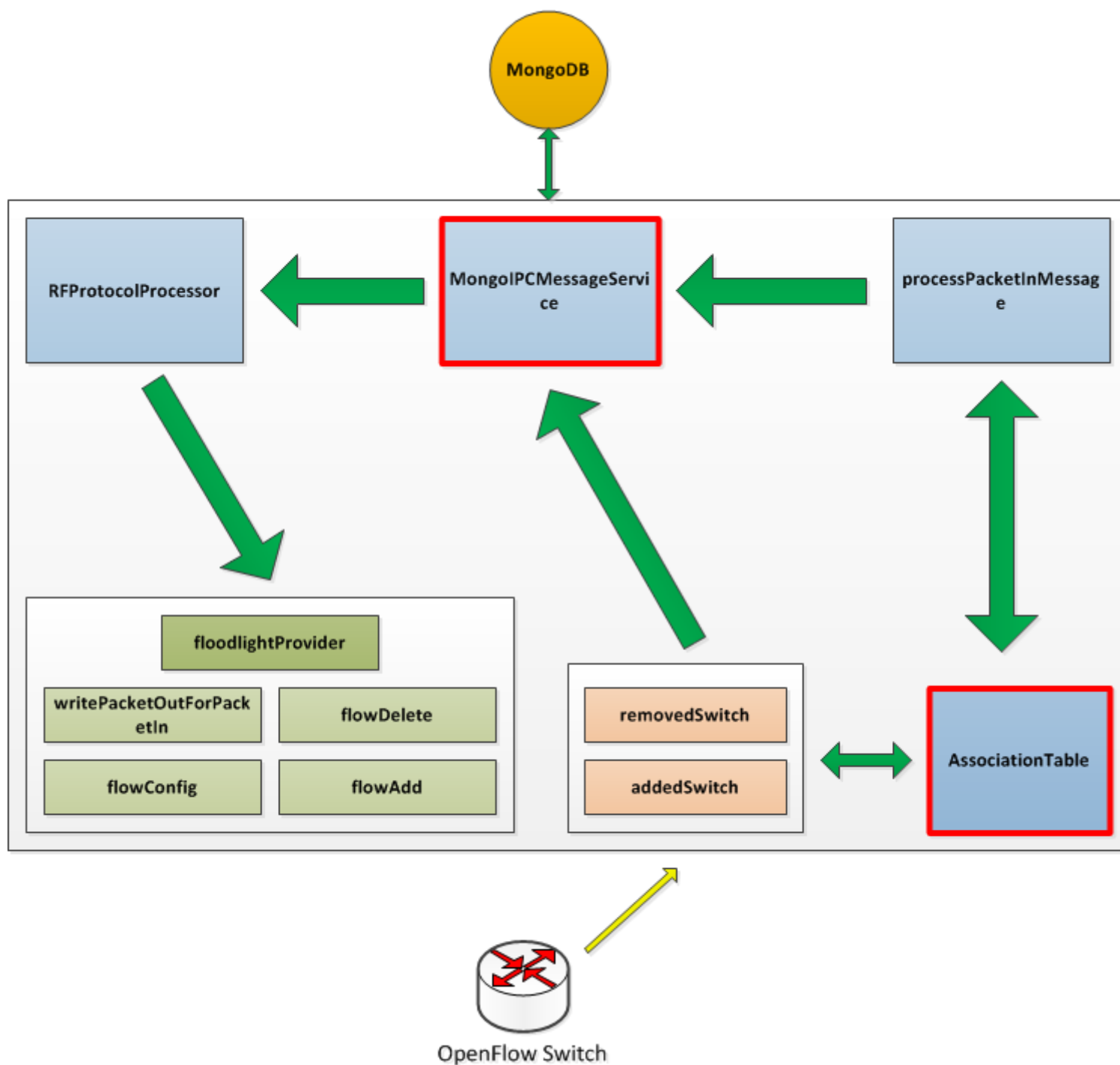
Proxy RouteFlow em Java

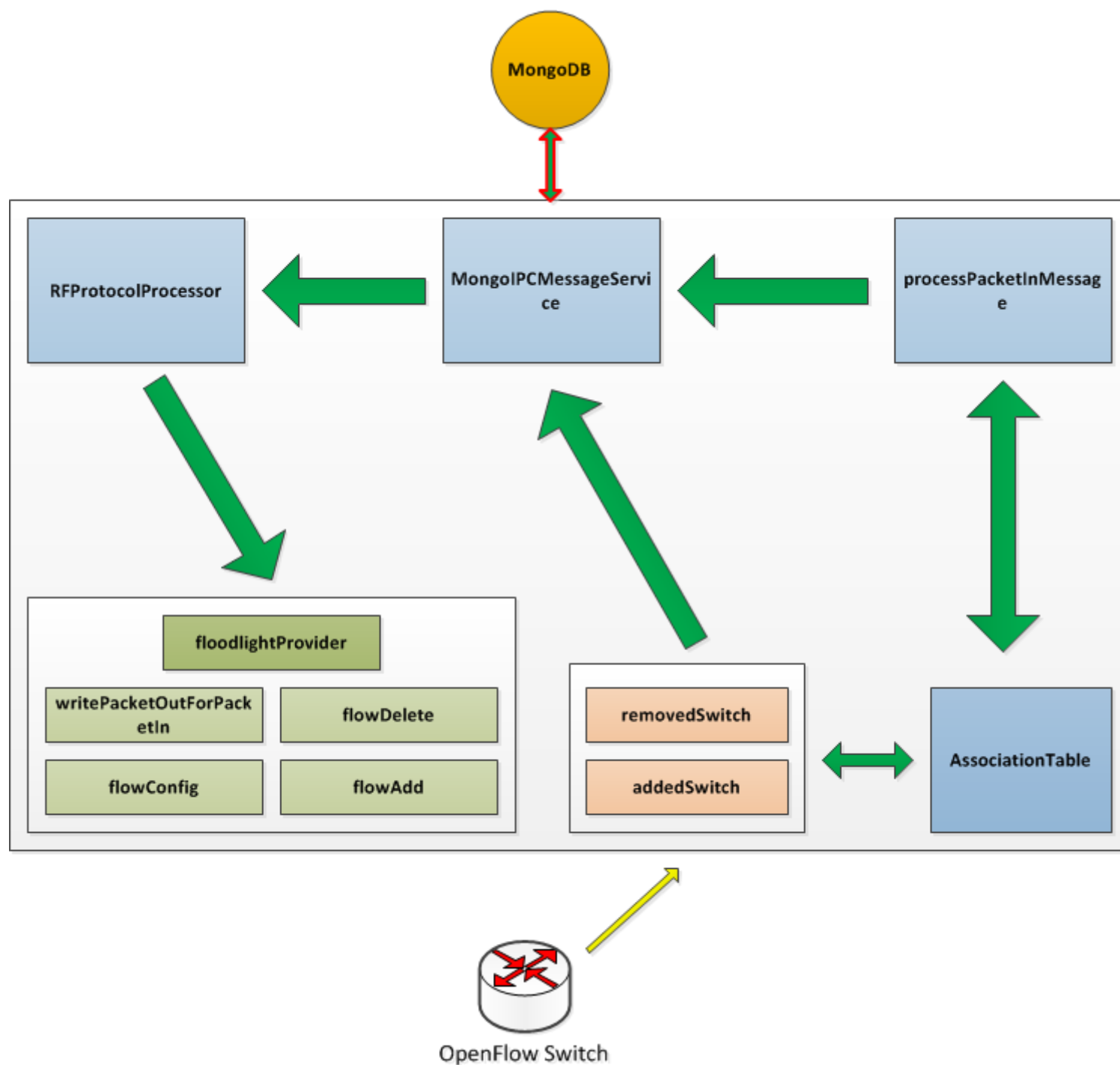
- Entrada de um novo switch OpenFlow no ambiente físico:
 - O mesmo deverá estar pré mapeado pelo ambiente virtual

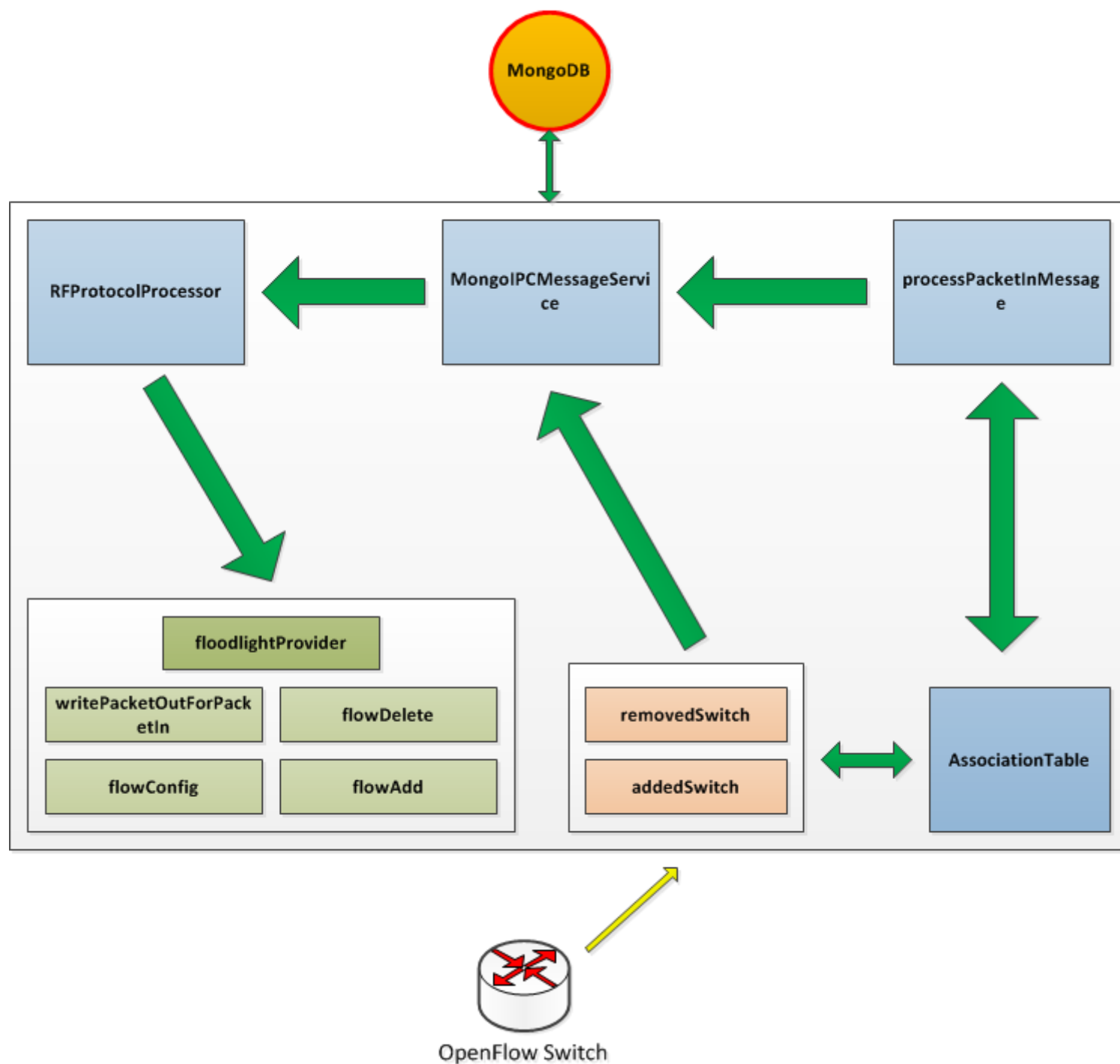


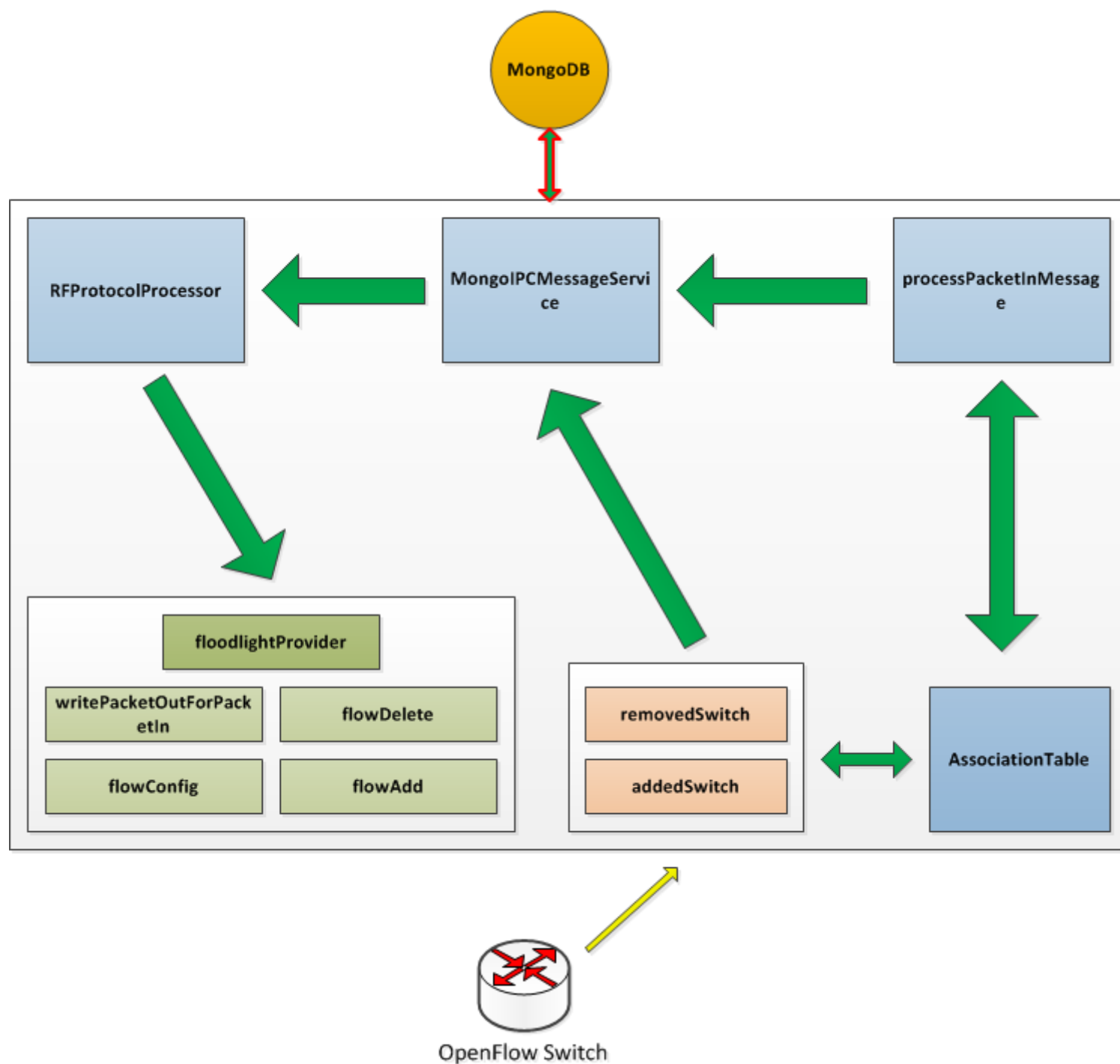


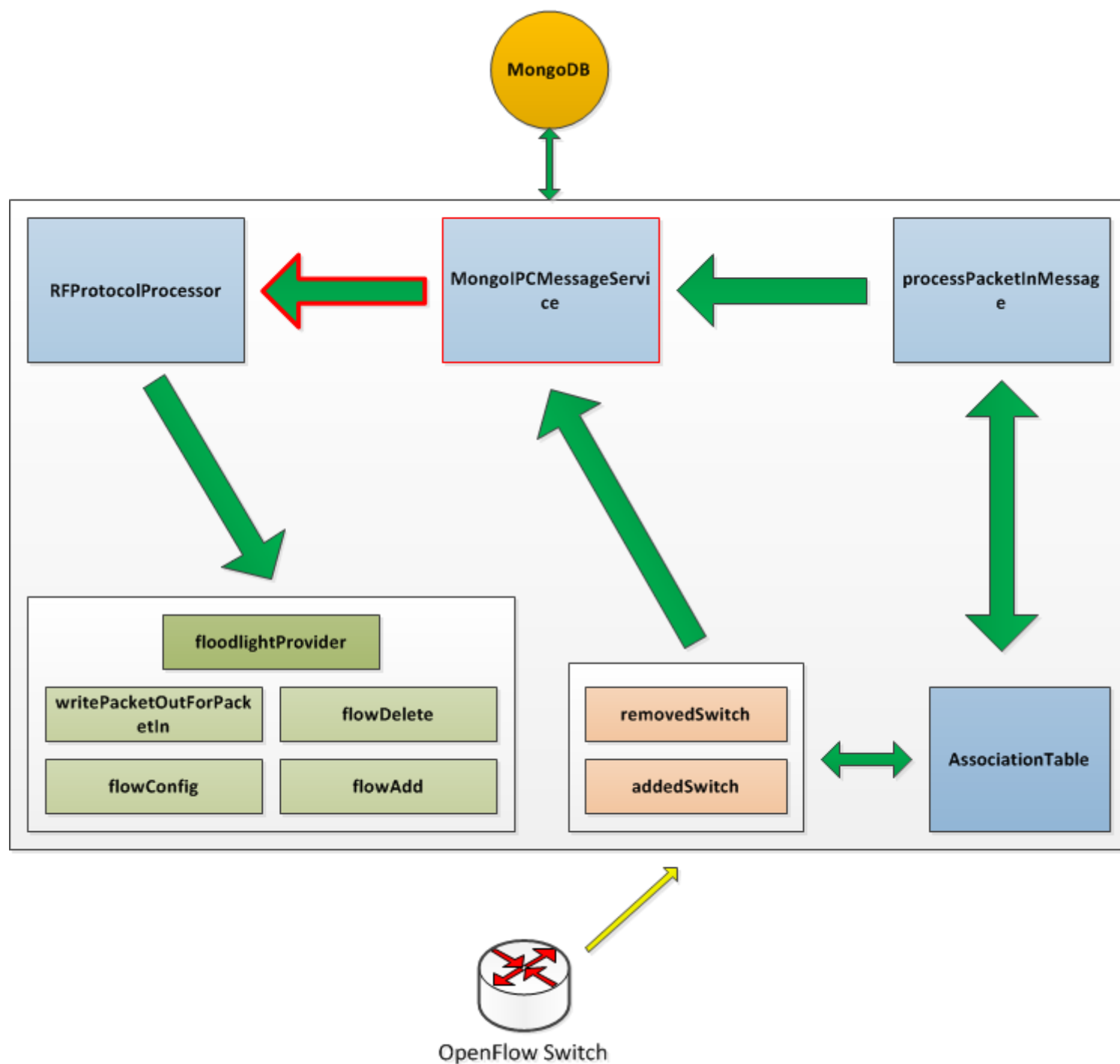


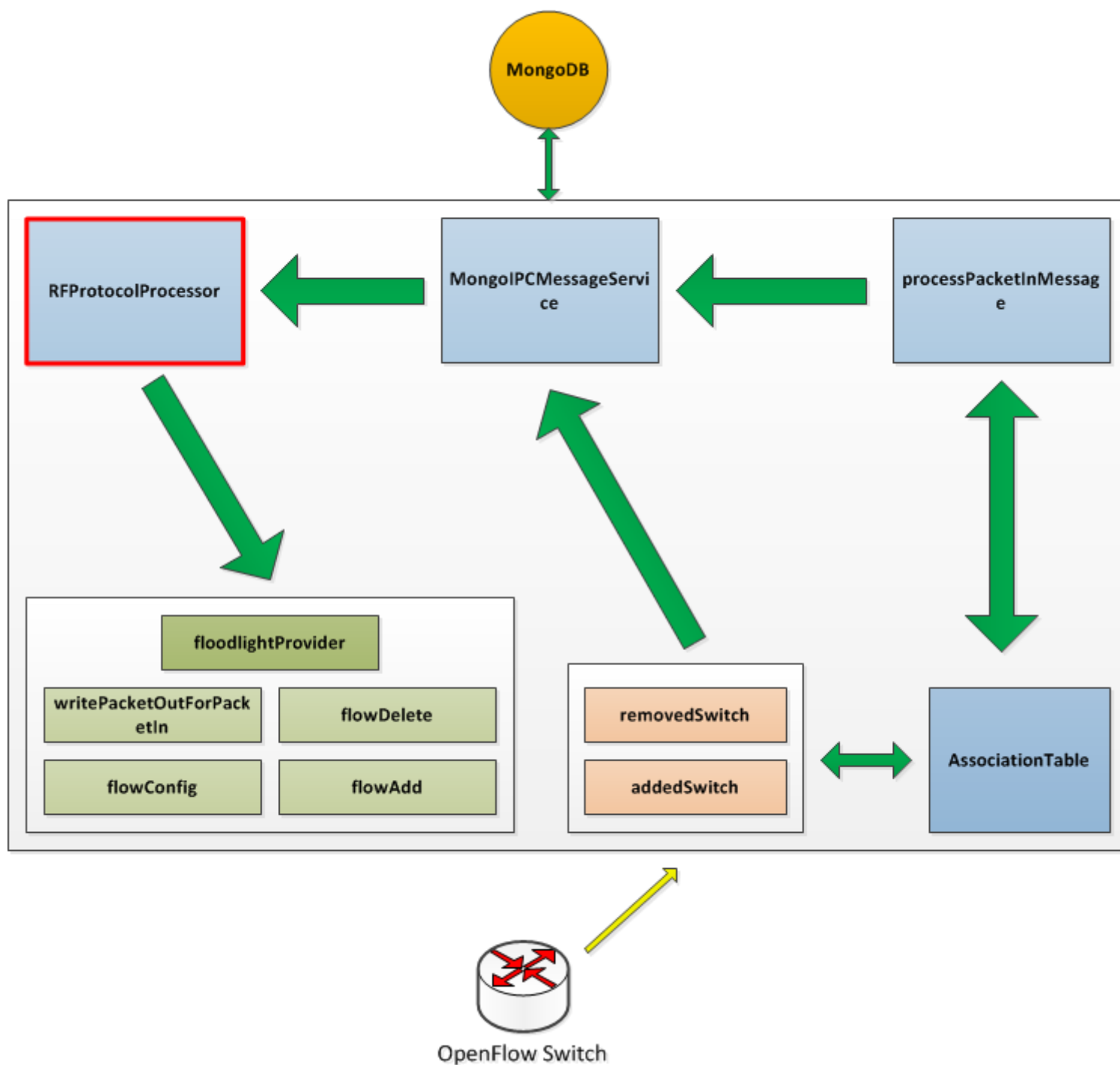


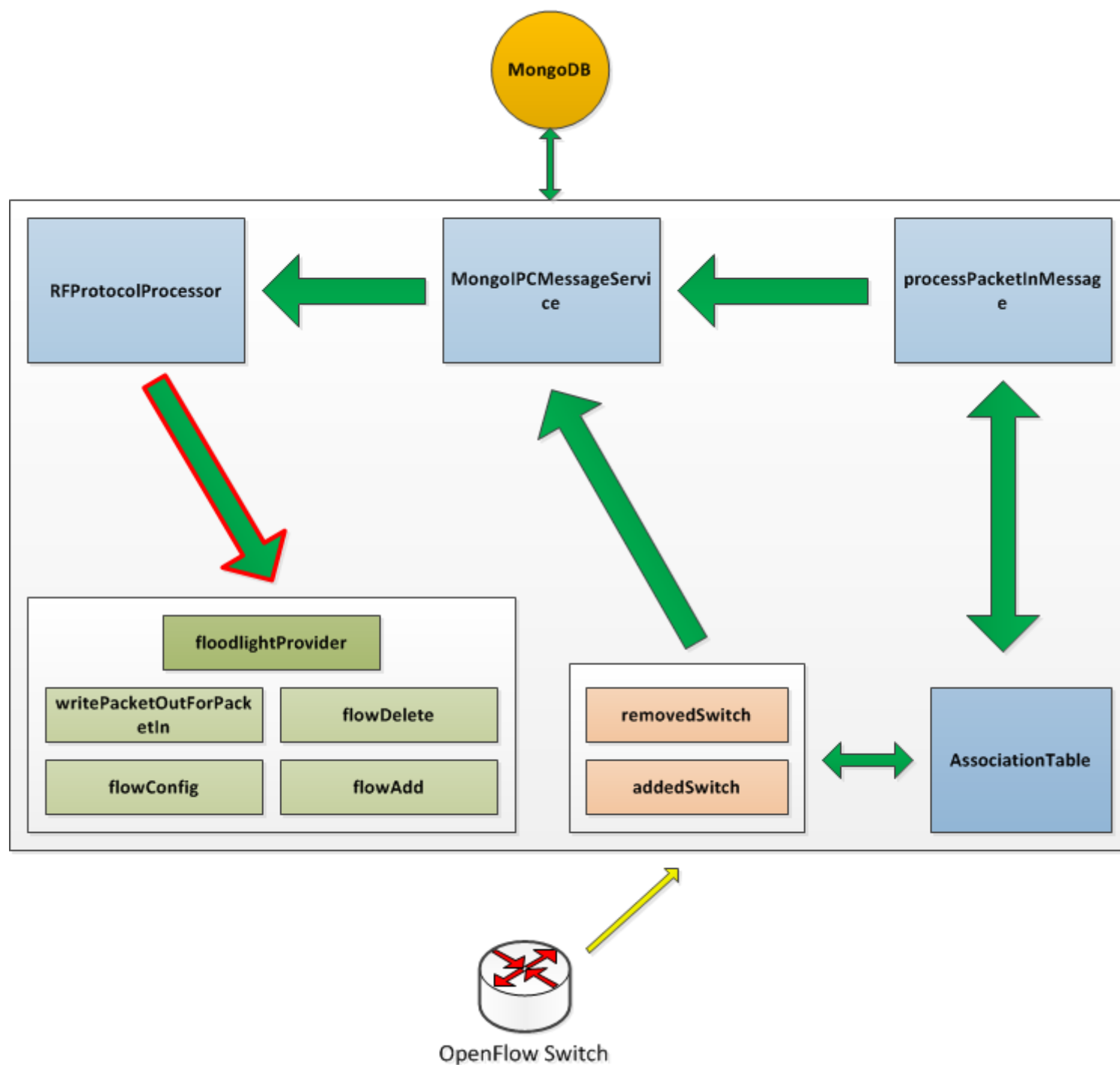


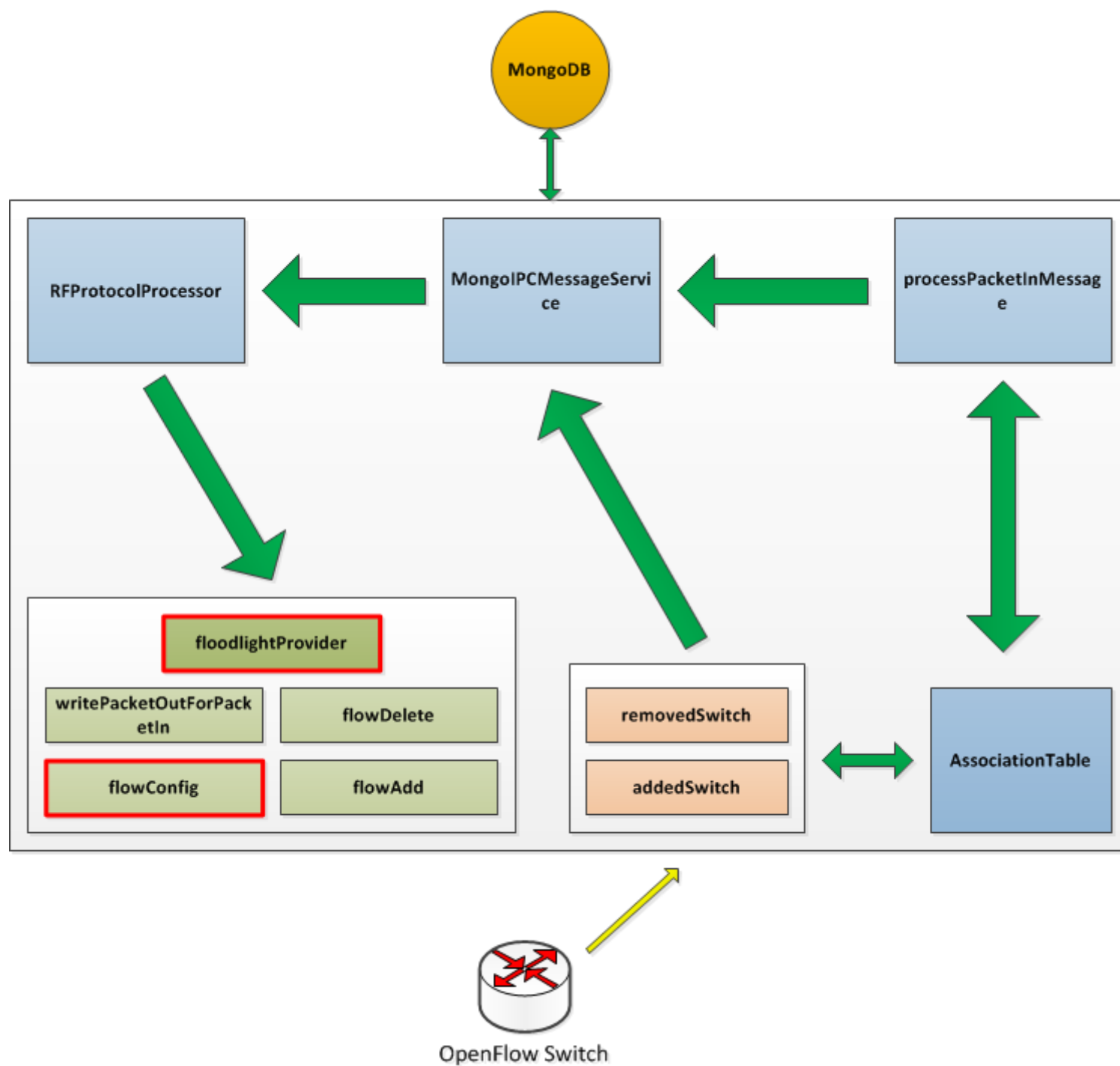


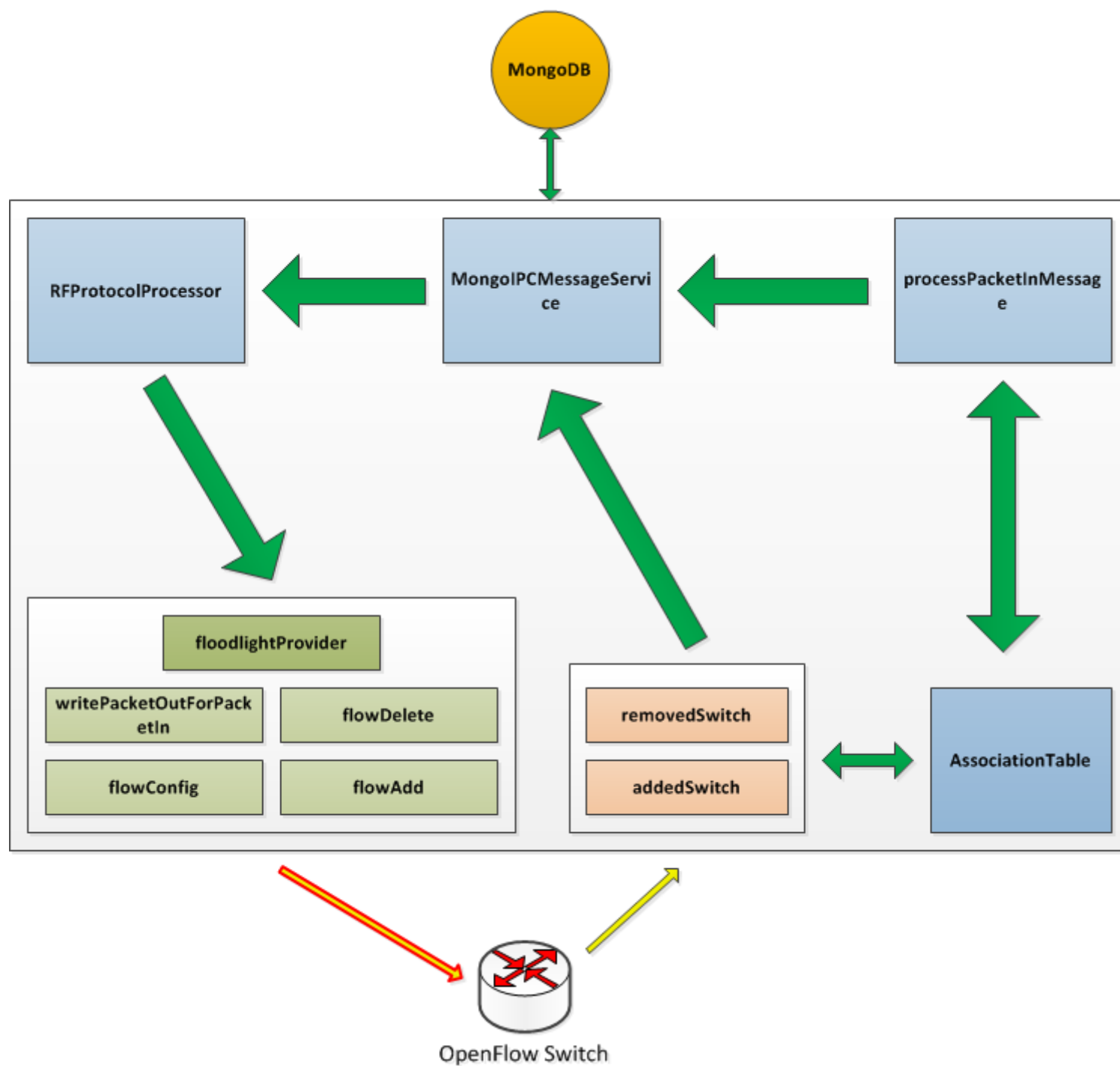












Agenda

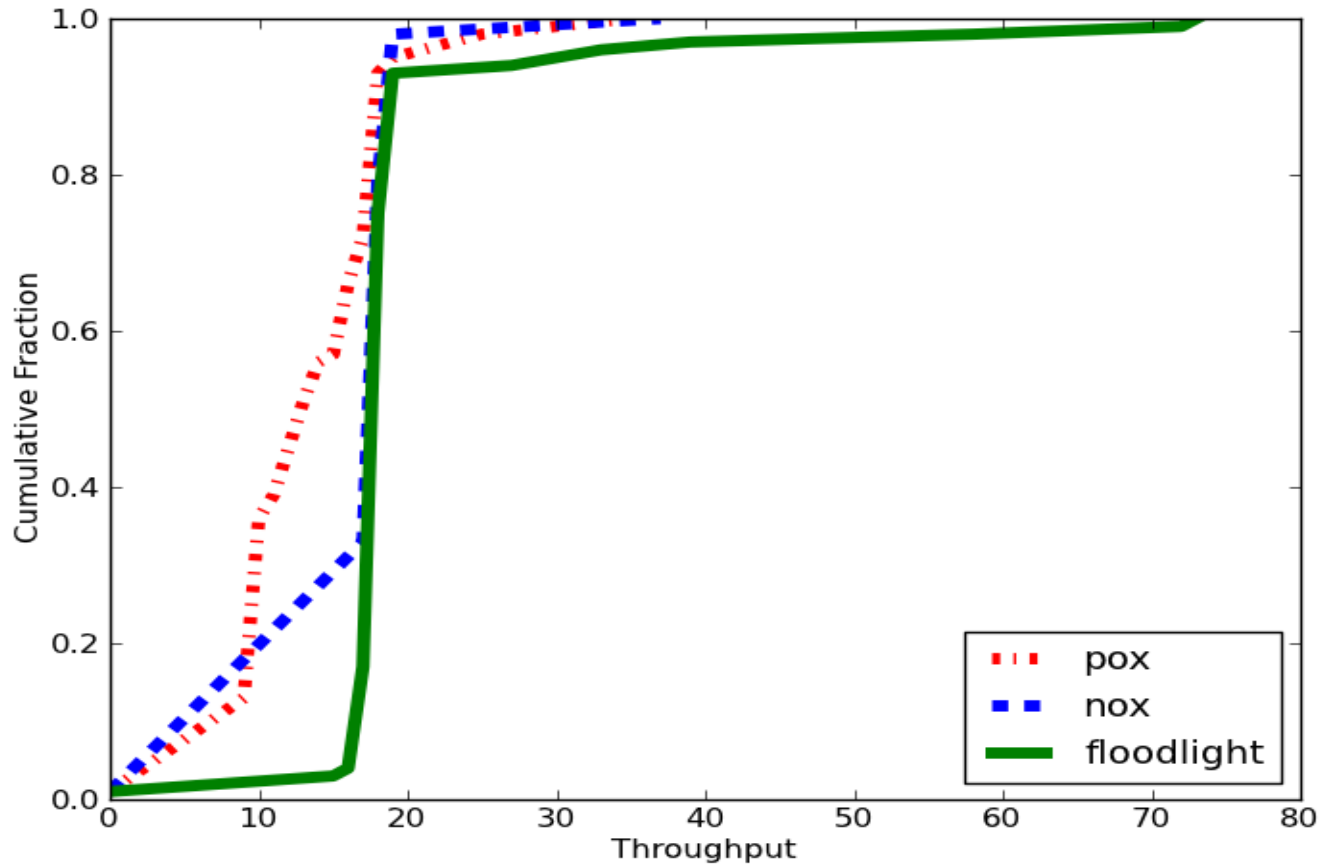
- Introdução
- Redes Definidas por Software
- Arquitetura Básica do RouteFlow
- Proxy RouteFlow em Java
- **Resultados**
- Conclusão

Resultados

- Obtidos com a ferramenta CBench

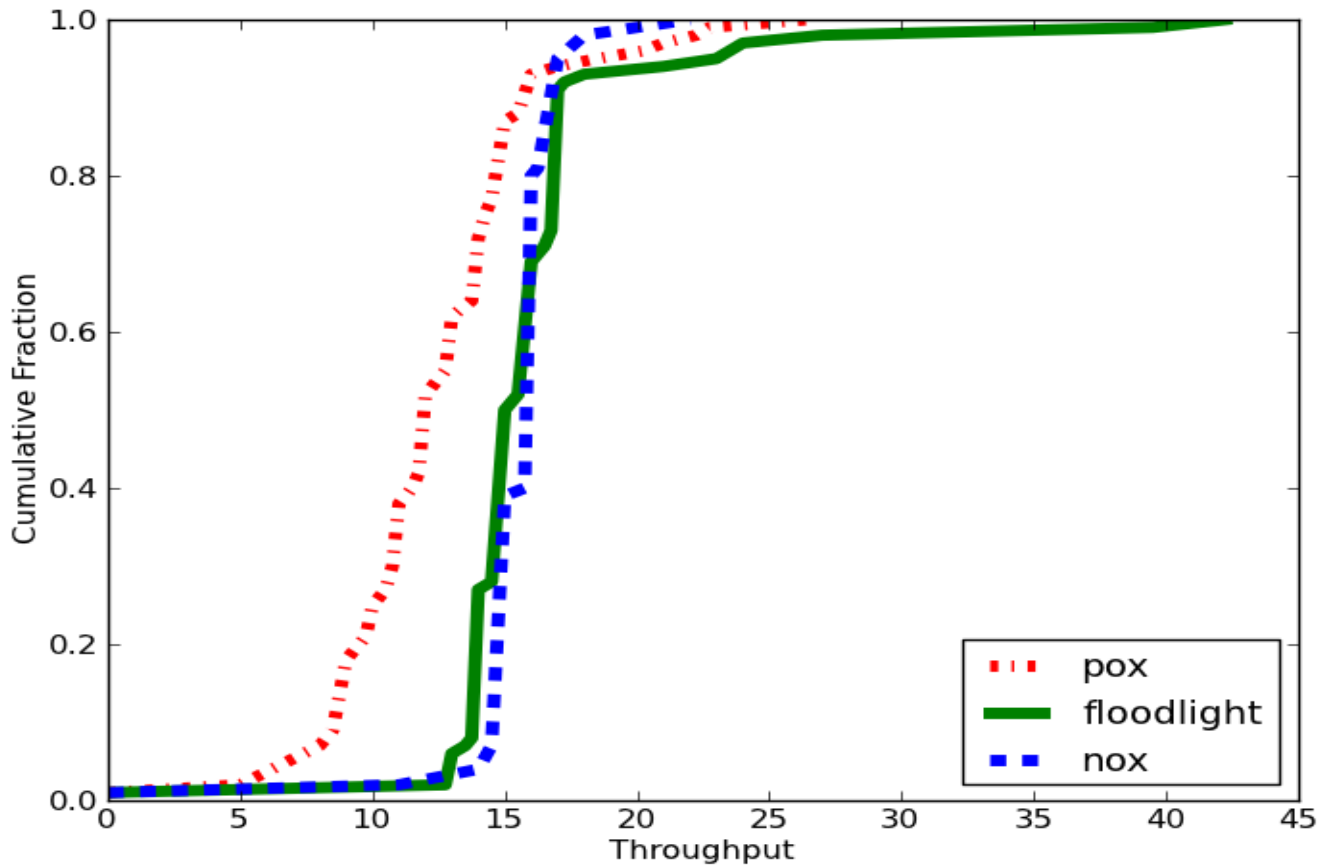
Resultados

- Latência para apenas um switch em ms



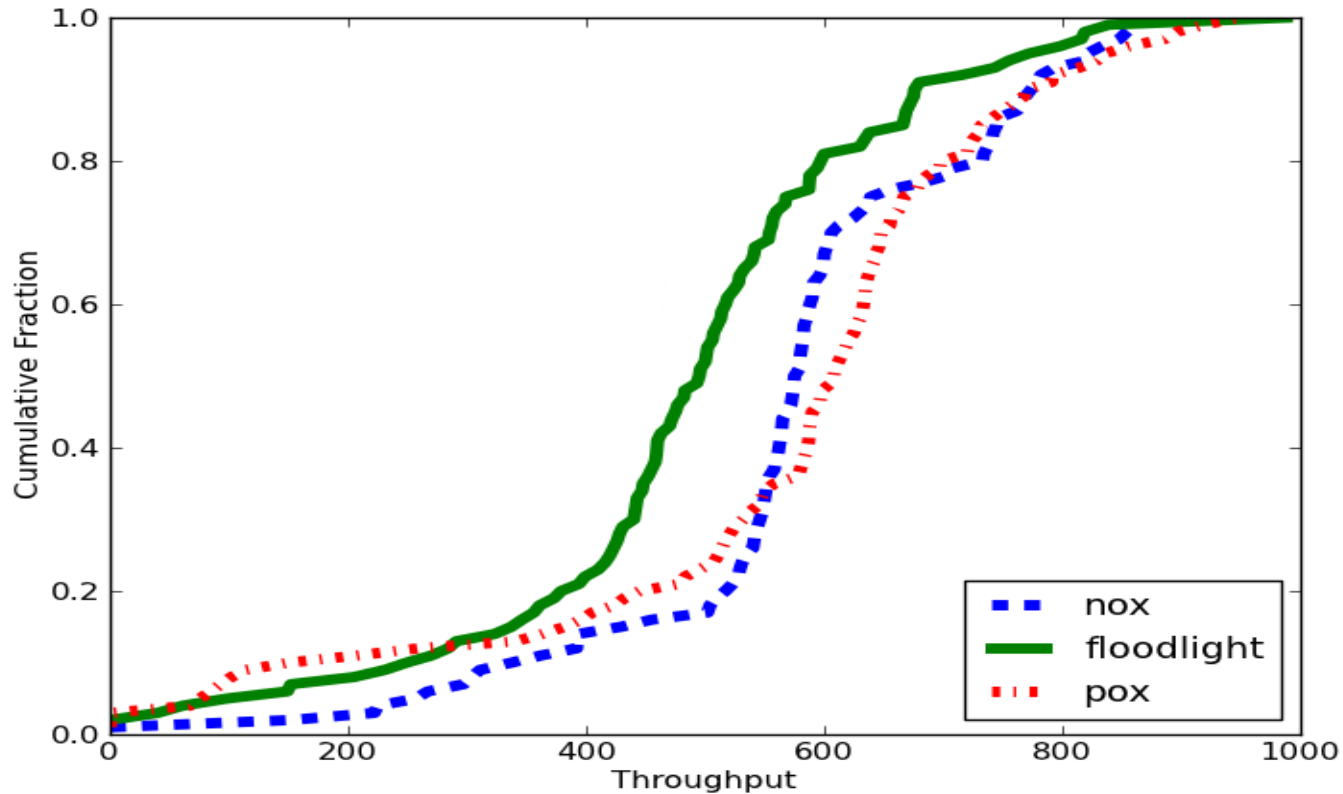
Resultados

- Latência para quatro switches em ms



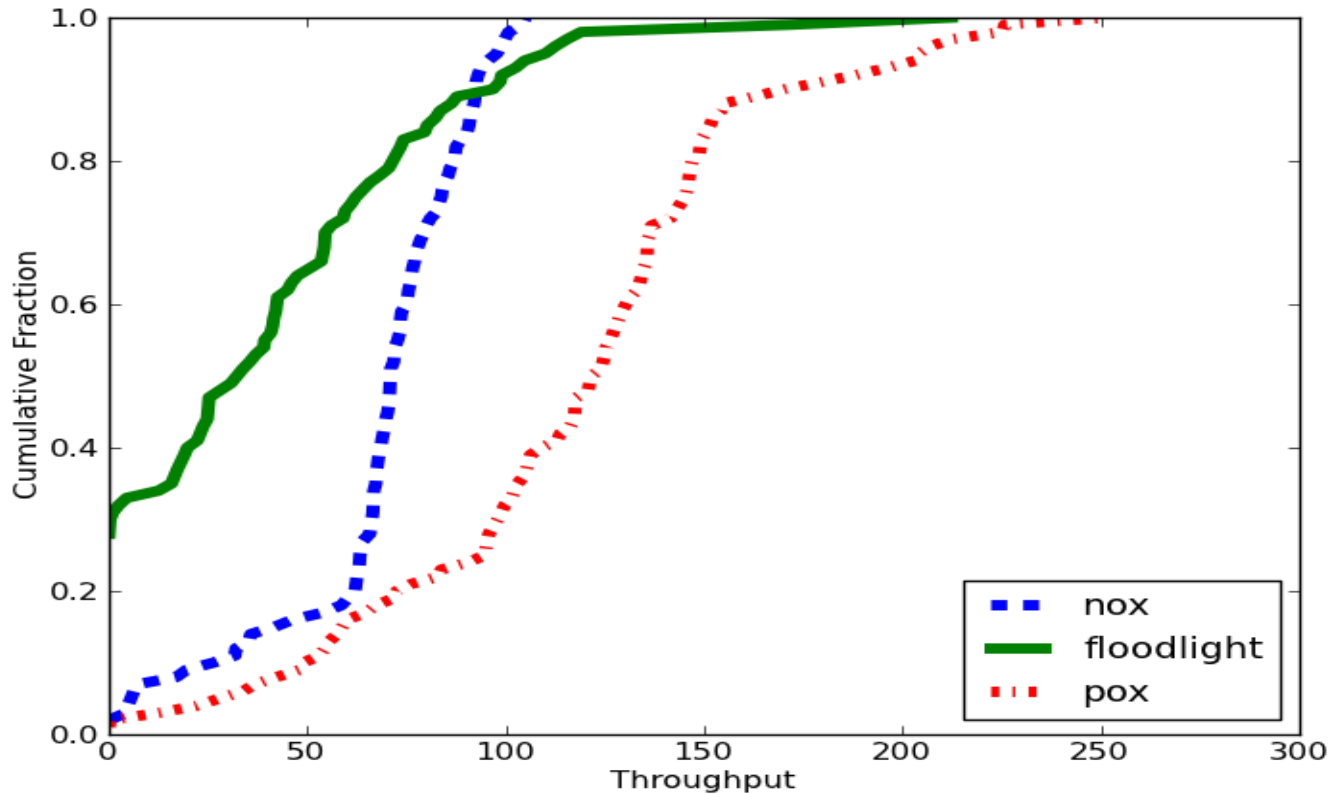
Resultados

- Desempenho com apenas um switch em fluxos por segundo



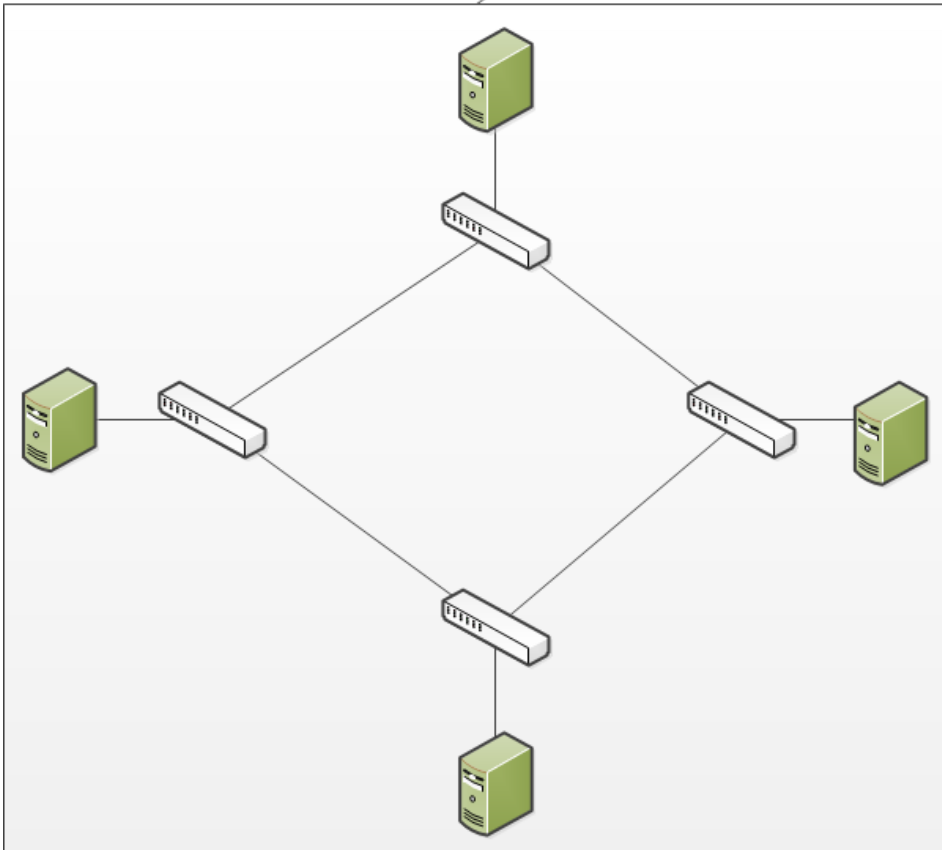
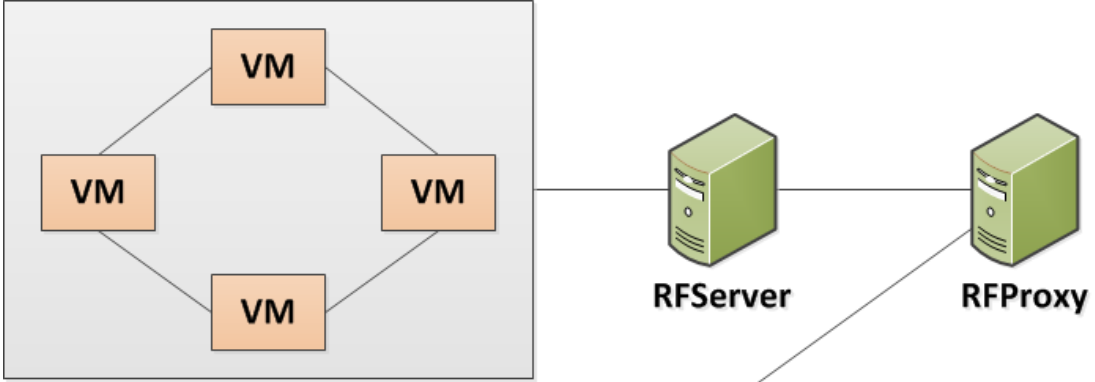
Resultados

- Desempenho para quatro switches em fluxos por segundo



Resultados

- Exemplo de execução do ambiente abaixo:



Agenda

- Introdução
- Redes Definidas por Software
- Arquitetura Básica do RouteFlow
- Proxy RouteFlow em Java
- Resultados
- Conclusão

Conclusão

- RouteFlow ganha suporte à mais um conjunto de tecnologias
- Atualização constante do proxy
- Aproximação com a equipe do RouteFlow

<https://github.com/fabianomathilde/RouteFlow>