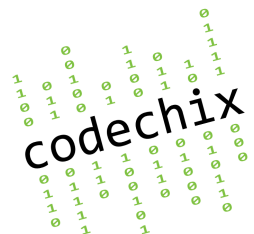


# OFConnect

An OpenSource OpenFlow Networking Library

Ramya Bolla  
Deepa Karnad Dhurka

18 Nov 2014

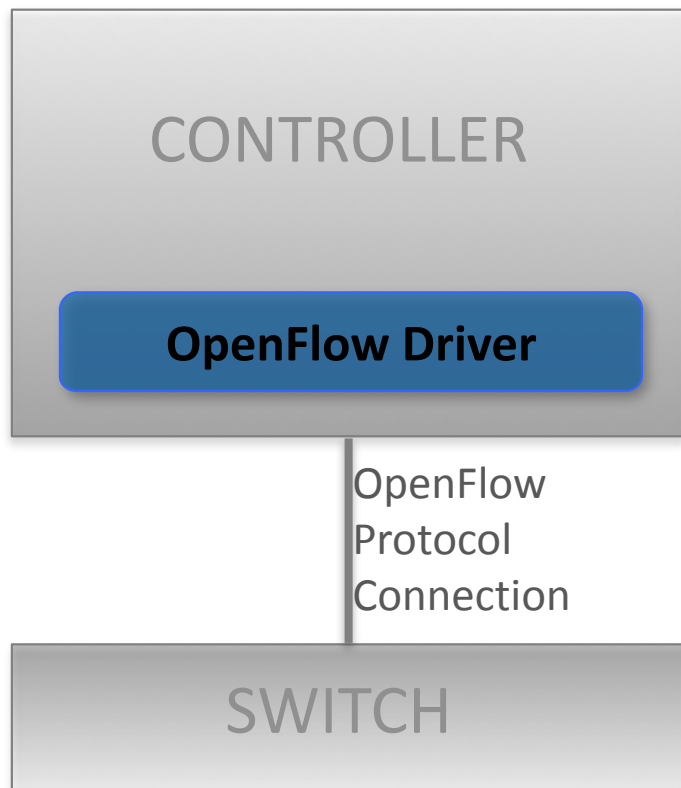


# Agenda

- About OFConnect
- Design & Architecture
- OFConnect API
- Demo Screenshots
- Get Started!

# What did we create?

## An OF network driver

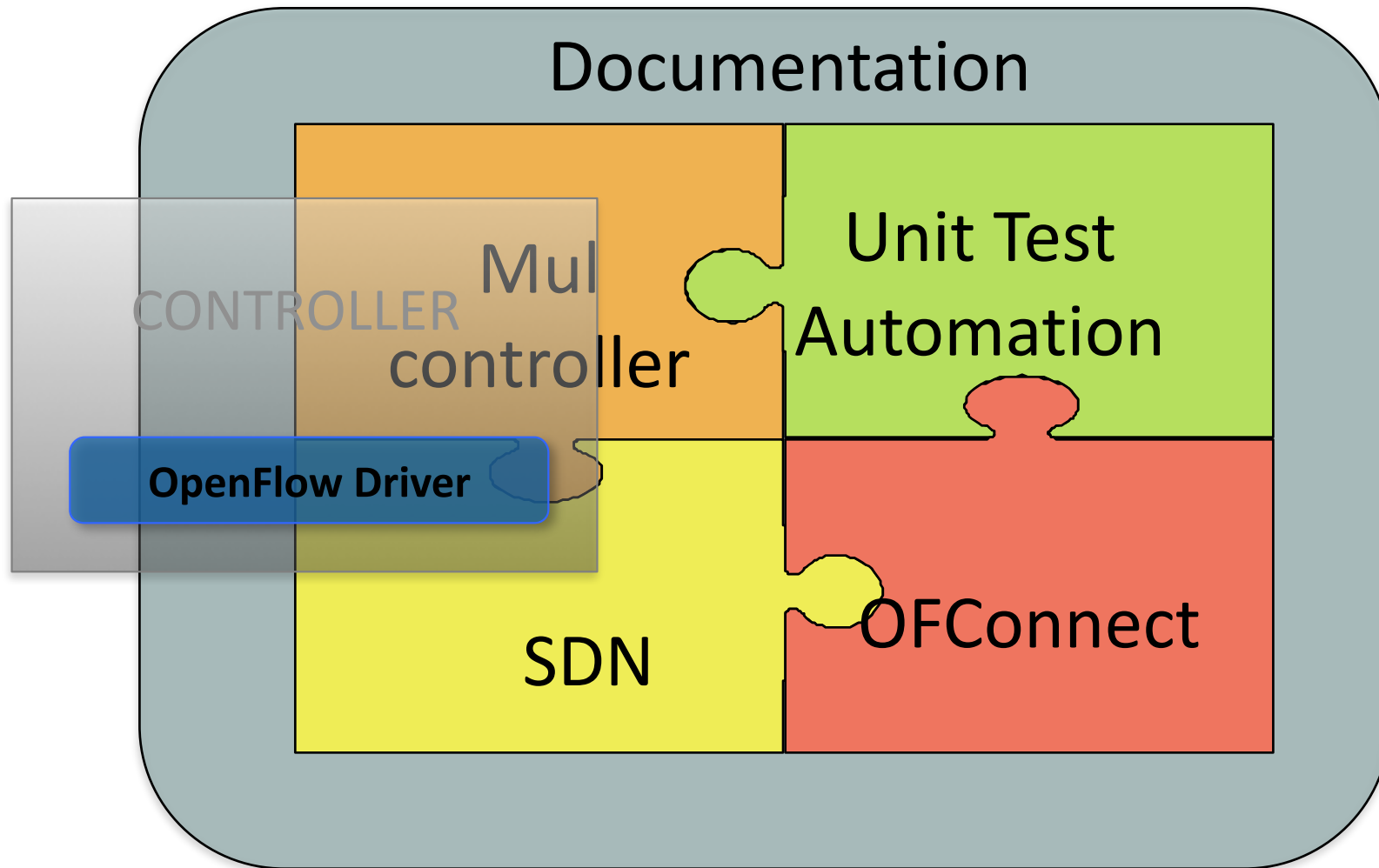


Open source library

with true OF implementation

to aid new OF controllers and switches

# What did we create?



# Requirements

## Functional

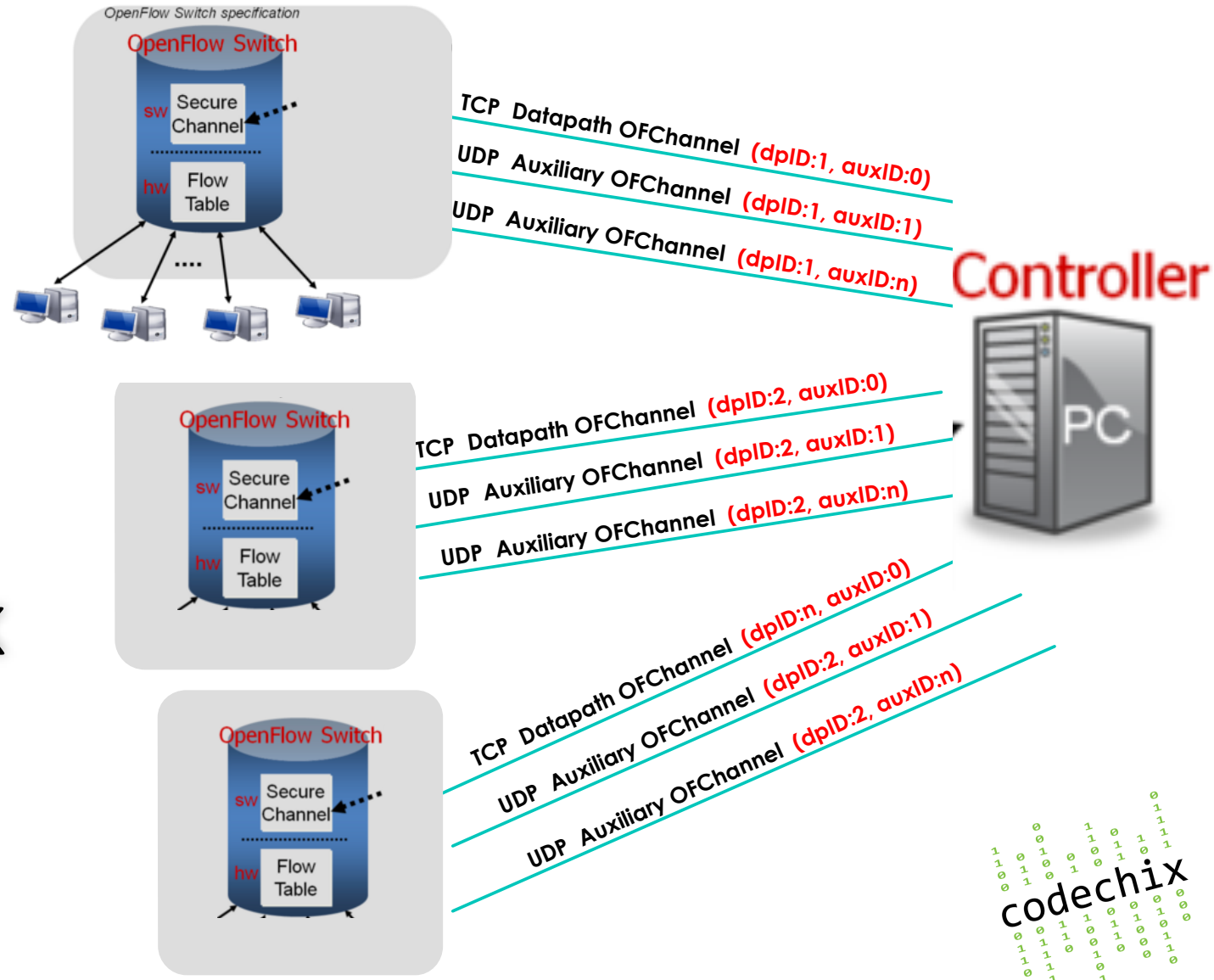
- Dynamically link with controller or switch
- Manage OF connections
- Support OF 1.3.1, 1.3 and 1.0
- Hide wire protocol

## Other

- Platform independence
- Multi-language bindings
- Full SDN Demo

# Design Goals

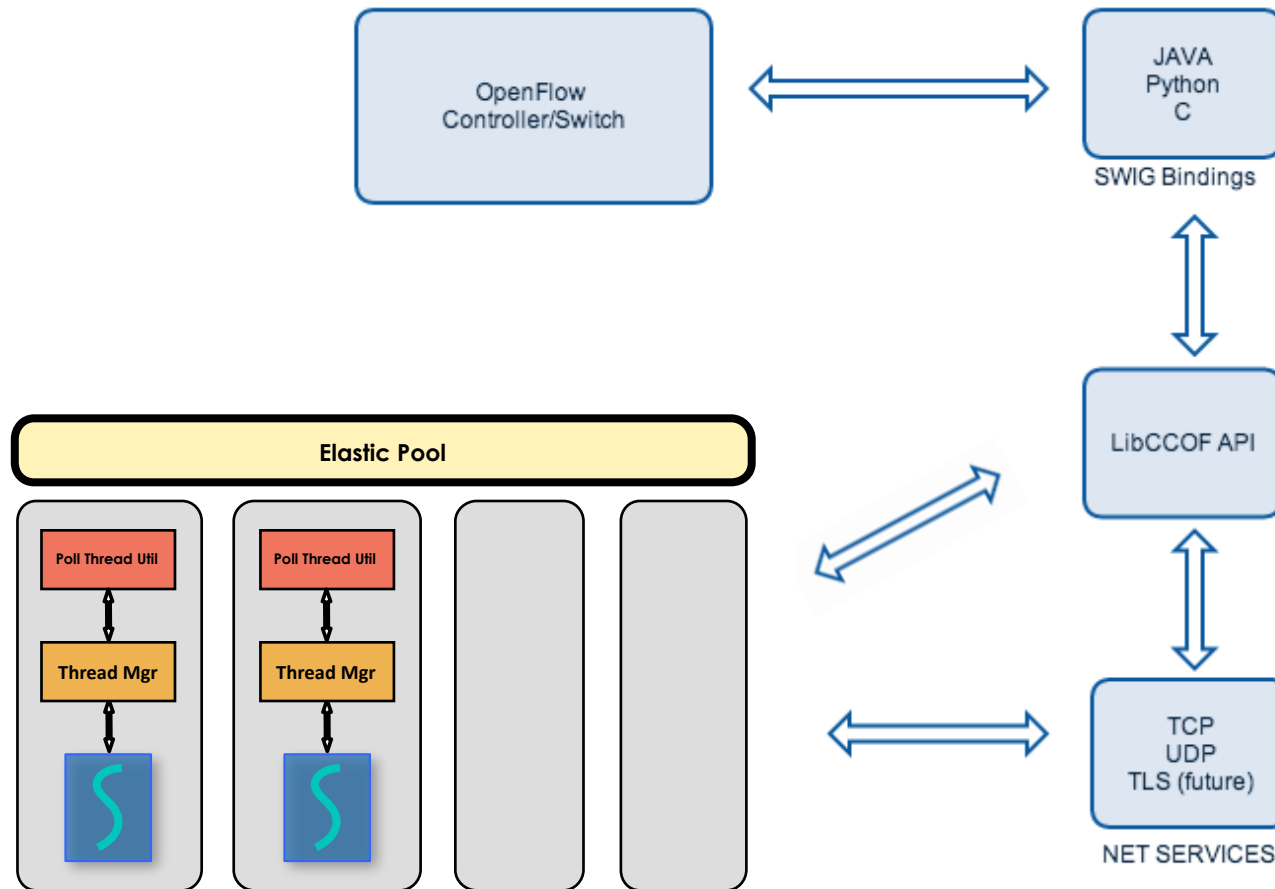
- Performance under scale
- Controller is capable of handling 256K connections



# Other Design Goals

- Generic framework to add new network protocols (TCP/UDP/TLS/IPv4/IPv6)
- A common API which works with both controller and switch

# Block Diagram





# OFConnect API

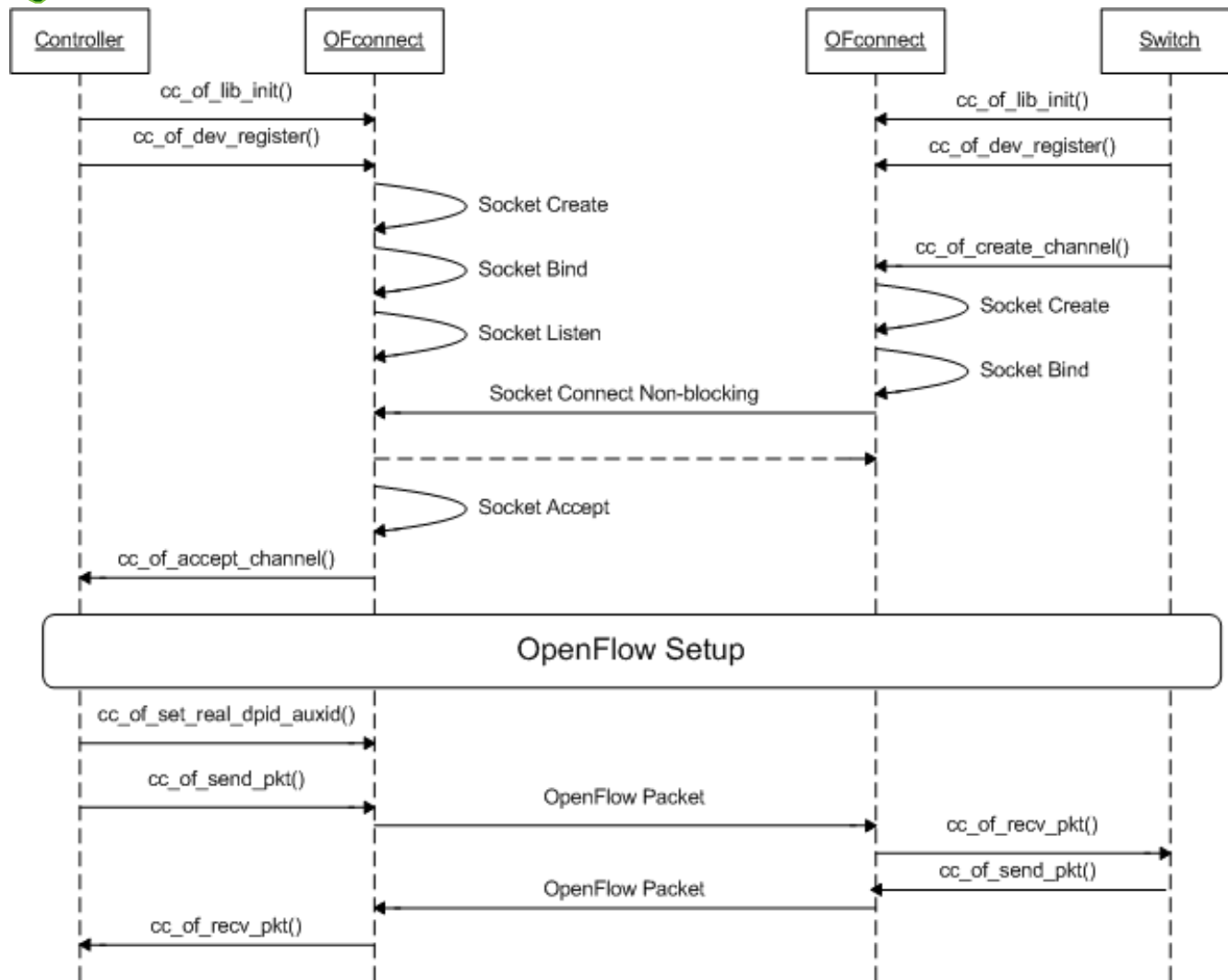
## cc\_of\_lib.h

```
cc_of_ret cc_of_lib_init();  
  
cc_of_ret cc_of_lib_free();  
  
cc_of_ret cc_of_dev_register();  
  
cc_of_ret cc_of_dev_free();  
  
cc_of_ret cc_of_create_channel();
```

## cc\_of\_lib.h

```
cc_of_ret cc_of_destroy_channel();  
  
cc_of_ret cc_of_send_pkt();  
  
cc_of_log_toggle();  
  
typedef int (*cc_of_recv_pkt)();  
  
typedef int (*cc_of_accept_channel)();  
  
typedef int (*cc_of_delete_channel)();
```

# OF Channel Setup



# Design Choices

- Static vs Dynamic thread model
- OFHeader Processing
- Poll vs Event based socket processing

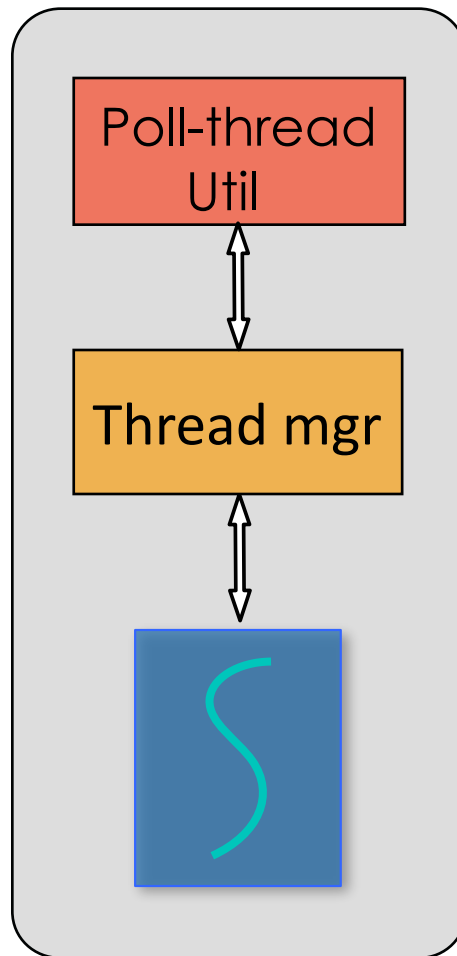
# Net Services

```
typedef struct net_svcs_ {
    int (*open_clientfd)(cc_ofdev_key_t key, cc_ofchannel_key_t ofchann_key);
    int (*open_serverfd)(cc_ofdev_key_t key);
    int (*accept_conn)(int listenfd, cc_ofdev_key_t key);
    int (*close_conn)(int sockfd);
    ssize_t (*read_data)(int sockfd, void *buf, size_t len, int flags,
                        struct sockaddr *src_addr, socklen_t *addrlen);
    ssize_t (*write_data)(int sockfd, const void *buf, size_t len, int flags,
                        const struct sockaddr *dest_addr, socklen_t addrlen);
} net_svcs_t;
```

```
/* Callback Registration for TCP */
```

```
net_svcs_t tcp_sockfns = {
    tcp_open_clientfd,
    tcp_open_listenfd,
    tcp_accept,
    tcp_close,
    tcp_read,
    tcp_write,
};
```

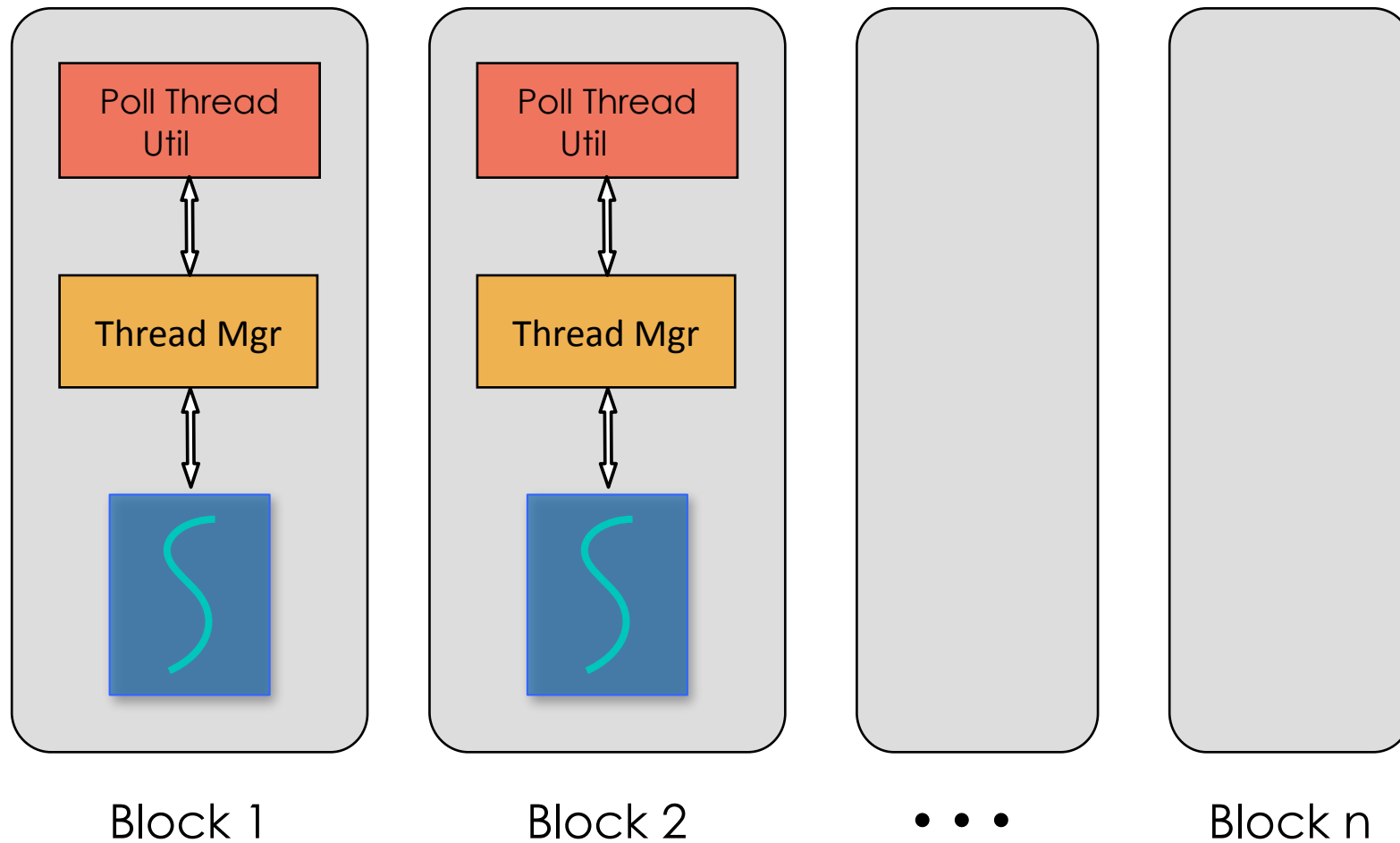
# Thread Management



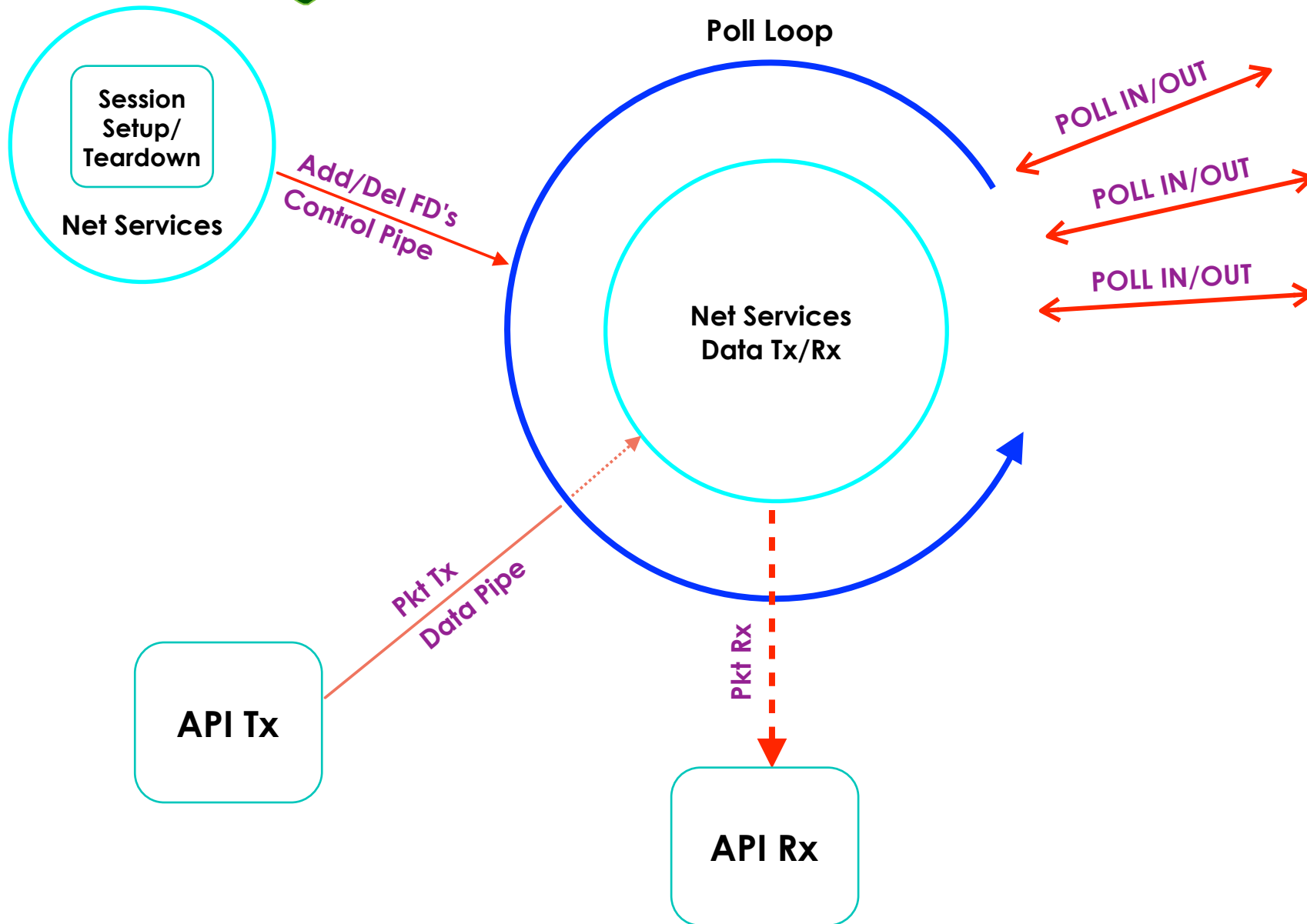
Thread Admin Block

# Elastic Pool

## Elastic Pool



# Inside A Thread



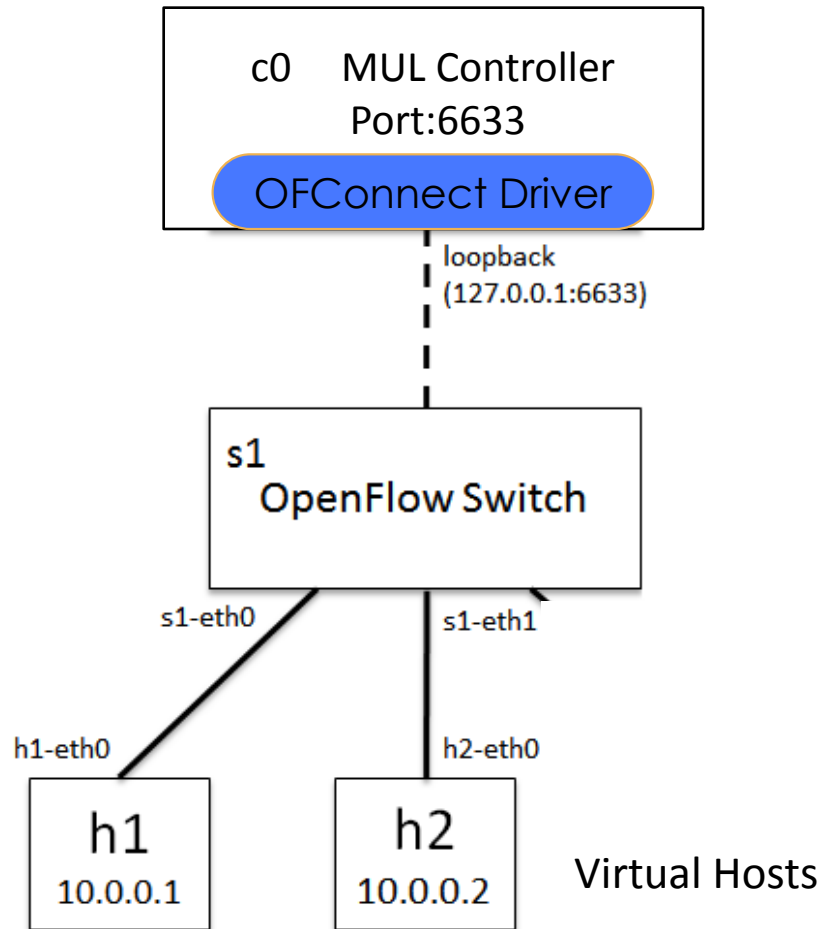
# Test Setup

open  
**MUL**

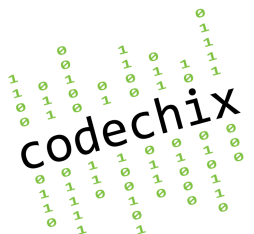


OPEN vSWITCH  
An Open Virtual Switch

MININET



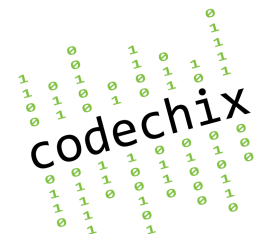
```
sudo mn --controller=remote, ip=127.0.0.1 --switch ovsk
```





# DEMO SCREENSHOTS

No.	Time	Source	Destination	Protocol	Length	Info
70	214.824121000	127.0.0.1	127.0.0.1	OpenFlow	74	Type: OFPT_HELLO
72	214.828774000	127.0.0.1	127.0.0.1	OpenFlow	74	Type: OFPT_HELLO
74	214.828815000	127.0.0.1	127.0.0.1	OpenFlow	74	Type: OFPT_FEATURES_REQUEST
76	214.831413000	127.0.0.1	127.0.0.1	OpenFlow	242	Type: OFPT_FEATURES_REPLY
77	214.849630000	::	ff02::1:ff00:2	OpenFlow	162	Type: OFPT_PACKET_IN
79	215.029464000	::	ff02::16	OpenFlow	174	Type: OFPT_PACKET_IN
80	215.066572000	::	ff02::1:ff00:1	OpenFlow	162	Type: OFPT_PACKET_IN
82	215.497545000	::	ff02::1:ffec:e7f8	OpenFlow	162	Type: OFPT_PACKET_IN
84	215.585386000	::	ff02::16	OpenFlow	174	Type: OFPT_PACKET_IN
86	215.589517000	::	ff02::16	OpenFlow	174	Type: OFPT_PACKET_IN
88	215.850296000	fe80::200:ff:fe00:2	ff02::16	OpenFlow	174	Type: OFPT_PACKET_IN
90	215.850333000	fe80::200:ff:fe00:2	ff02::2	OpenFlow	154	Type: OFPT_PACKET_IN
92	216.065463000	fe80::200:ff:fe00:1	ff02::2	OpenFlow	154	Type: OFPT_PACKET_IN
94	216.065492000	fe80::200:ff:fe00:1	ff02::16	OpenFlow	174	Type: OFPT_PACKET_IN
96	216.149372000	fe80::200:ff:fe00:2	ff02::16	OpenFlow	174	Type: OFPT_PACKET_IN
98	216.499151000	fe80::3429:dcff:feec:e	ff02::2	OpenFlow	154	Type: OFPT_PACKET_IN
100	216.499190000	fe80::3429:dcff:feec:e	ff02::16	OpenFlow	174	Type: OFPT_PACKET_IN
102	216.517476000	fe80::200:ff:fe00:1	ff02::16	OpenFlow	174	Type: OFPT_PACKET_IN
104	217.371180000	fe80::3429:dcff:feec:e	ff02::16	OpenFlow	174	Type: OFPT_PACKET_IN
108	219.623979000	127.0.0.1	127.0.0.1	OpenFlow	74	Type: OFPT_ECHO_REQUEST
110	219.624258000	127.0.0.1	127.0.0.1	OpenFlow	74	Type: OFPT_ECHO_REPLY
117	219.861742000	fe80::200:ff:fe00:2	ff02::2	OpenFlow	154	Type: OFPT_PACKET_IN
119	220.070019000	fe80::200:ff:fe00:1	ff02::2	OpenFlow	154	Type: OFPT_PACKET_IN
121	220.501672000	fe80::3429:dcff:feec:e	ff02::2	OpenFlow	154	Type: OFPT_PACKET_IN
123	220.611524000	00:00:00_00:00:01	Broadcast	OpenFlow	126	Type: OFPT_PACKET_IN
125	221.610056000	00:00:00_00:00:01	Broadcast	OpenFlow	126	Type: OFPT_PACKET_IN
127	222.610010000	00:00:00_00:00:01	Broadcast	OpenFlow	126	Type: OFPT_PACKET_IN
129	223.869913000	fe80::200:ff:fe00:2	ff02::2	OpenFlow	154	Type: OFPT_PACKET_IN
131	224.077427000	fe80::200:ff:fe00:1	ff02::2	OpenFlow	154	Type: OFPT_PACKET_IN
133	224.510030000	fe80::3429:dcff:feec:e	ff02::2	OpenFlow	154	Type: OFPT_PACKET_IN
135	224.622539000	127.0.0.1	127.0.0.1	OpenFlow	74	Type: OFPT_ECHO_REQUEST
137	224.622667000	127.0.0.1	127.0.0.1	OpenFlow	74	Type: OFPT_ECHO_REPLY
143	229.624369000	127.0.0.1	127.0.0.1	OpenFlow	74	Type: OFPT_ECHO_REQUEST
144	229.624539000	127.0.0.1	127.0.0.1	OpenFlow	74	Type: OFPT_ECHO_REPLY



# DEMO SCREENSHOTS

```
2014/11/17 18:14:43 MUL-CONTROLLER: (mul_cc_of_accept) Accept received dpid:32 auxid:32
2014/11/17 18:14:43 MUL-CONTROLLER: (mul_cc_of_accept) New switch context created
2014/11/17 18:14:43 MUL-CONTROLLER: (mul_cc_of_accept) switch:port str:127.0.0.1:36290
2014/11/17 18:14:43 MUL-CONTROLLER: (of_send_hello) Send Hello to library dummy_dpid:32
2014/11/17 18:14:43 MUL-CONTROLLER: (c_thread_tx) Sending packet to library
2014/11/17 18:14:43 MUL-CONTROLLER: (of_switch_rcv_msg) OF MSG RX TYPE (0) —> HELLO
2014/11/17 18:14:43 MUL-CONTROLLER: (c_thread_tx) Sending packet to library 32
2014/11/17 18:14:43 MUL-CONTROLLER: (of_switch_rcv_msg) OF MSG RX TYPE (6) —> FEATURES_REQUEST
2014/11/17 18:14:43 MUL-CONTROLLER: (of_rcv_features_reply) n_ports:3
2014/11/17 18:14:43 MUL-CONTROLLER: dpid:1 being looked up is_dummy:1
2014/11/17 18:14:43 MUL-CONTROLLER: (of_rcv_features_reply) dpid:1 version:1 n_tables:254 actions:0xfff
capabilities:0x0x7fa3000000c7 n_buffers:256
2014/11/17 18:14:43 MUL-CONTROLLER: (of_rcv_features_reply) real dpid sent to library
2014/11/17 18:14:43 MUL-CONTROLLER: (of_rcv_features_reply) switch is being registered
2014/11/17 18:14:48 MUL-CONTROLLER: (of_switch_rcv_msg) OF MSG RX TYPE (2) —> ECHO_REQUEST
2014/11/17 18:14:44 MUL-CONTROLLER: (of_switch_rcv_msg) OF MSG RX TYPE (10) —> PACKET_IN
2014/11/17 18:14:44 MUL-CONTROLLER: (of_rcv_packet_in) Packet IN DONE sw->dpid:1
2014/11/17 18:14:44 MUL-CONTROLLER: (of_dfl_fwd) sw->dpid:1
2014/11/17 18:14:44 MUL-CONTROLLER: Flow lookup fail
```

# Learnings

Platform Independence  
GLib

Language Bindings  
SWIG

Controller to Integrate  
MUL

GIT Workflow  
Upstream/Master/Dev

# Next Steps

Enhance  
TLS  
Greater OF Awareness

Scale  
1K switches per controller  
256K connections per controller

Perform  
Profile and Benchmark

# Summary

- OFConnect: OF Network Library
- Open Source GPL v2
- Join and Contribute

# Repos

- <https://github.com/CodeChix-OpenSource/OFconnect>
  - GPL v2
- <https://github.com/codechix/mul-OFconnect>
  - Private

# Thank you!

