The Hybrid-Open (“HOpen”) router architecture

Brian Field / Comcast

Background

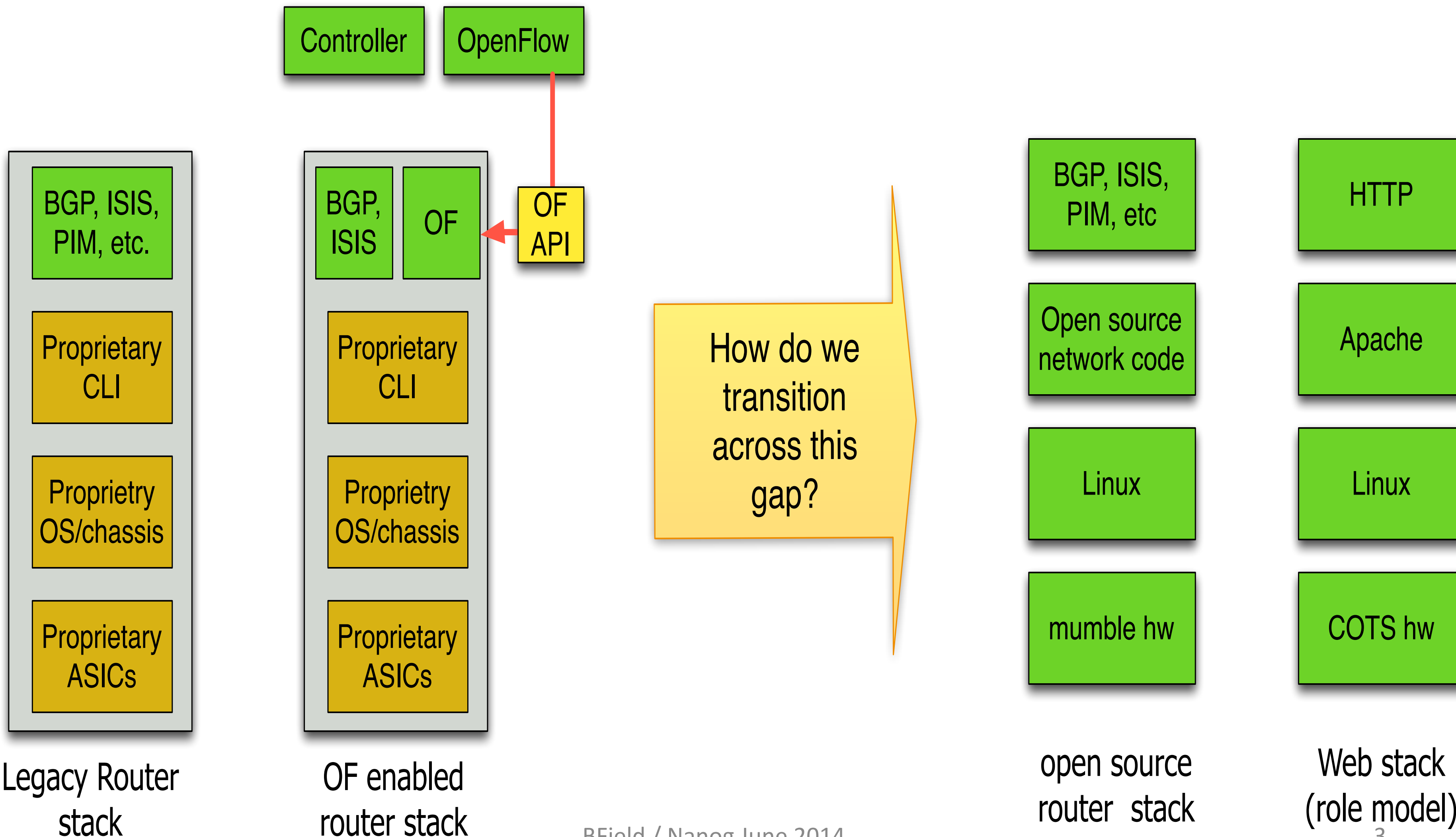
- Lots of "excitement" in the SDN space
  - Programmability (config)
  - Network virtualization (existing features)

- What if I need a new feature on a router-- today?

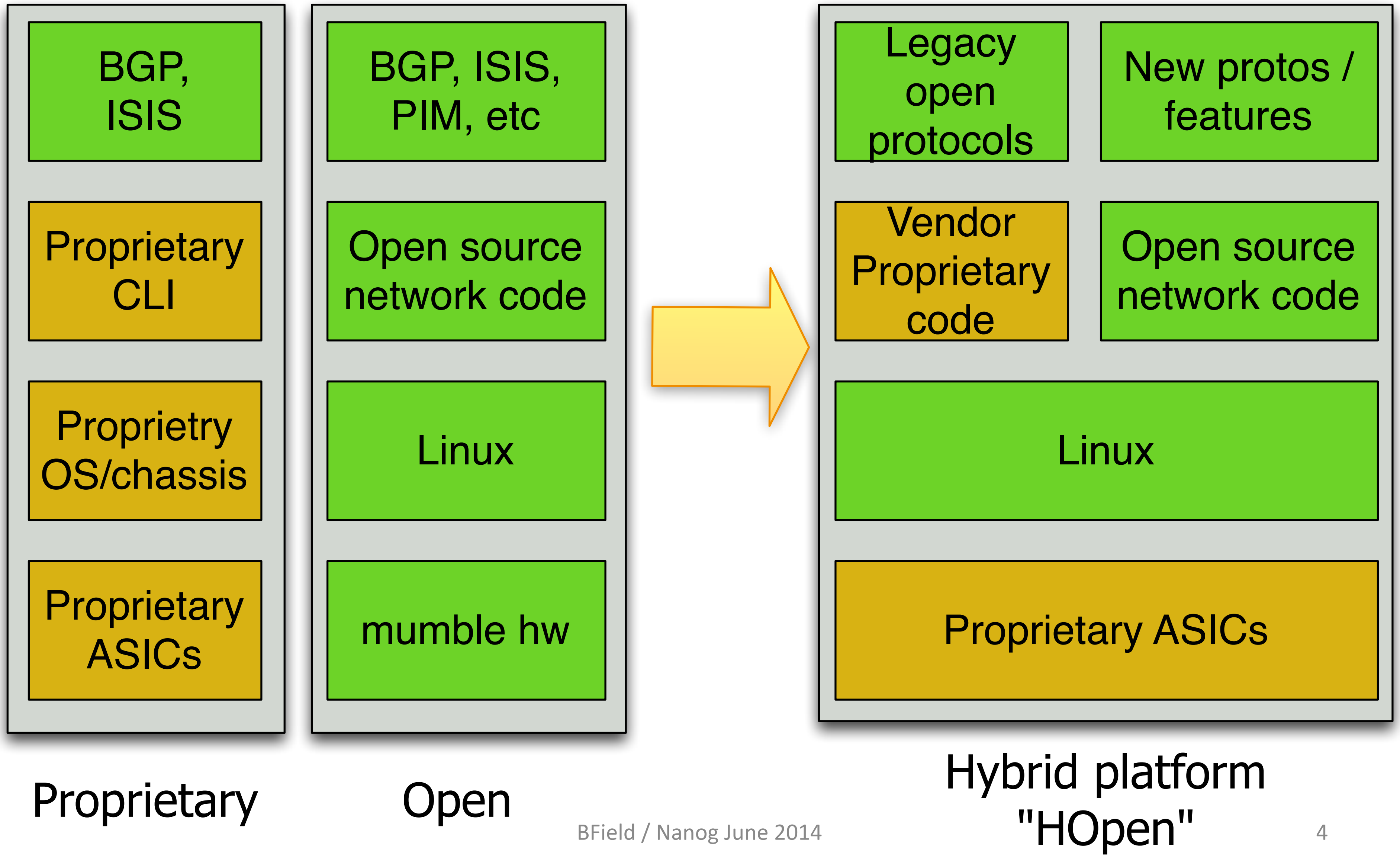
- Does the existing SDN paradigm help this?
    - No [not really]

- This talk presents a new "SDN" paradigm:
  - Something we can Do Now.

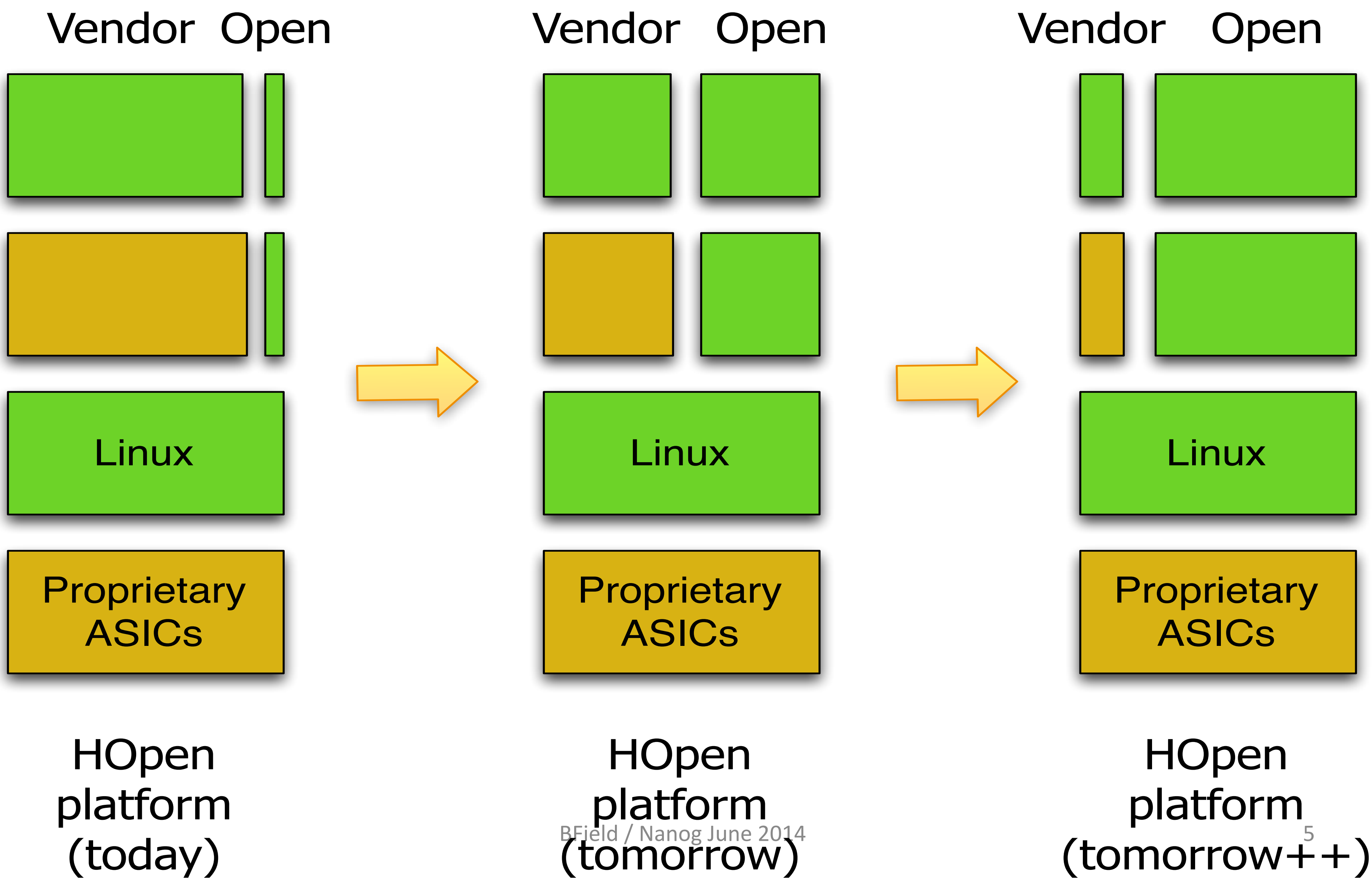
Router platform evolution



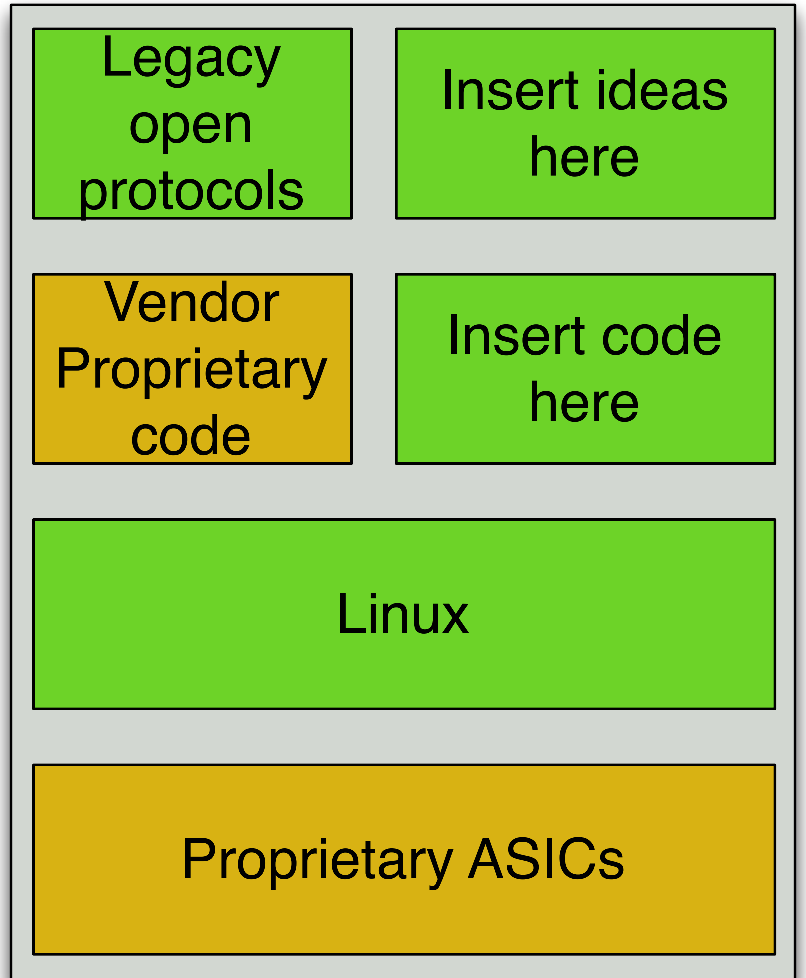
We glue them together… “HOpen”



In theory, over time….

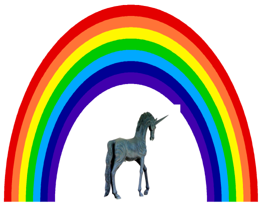


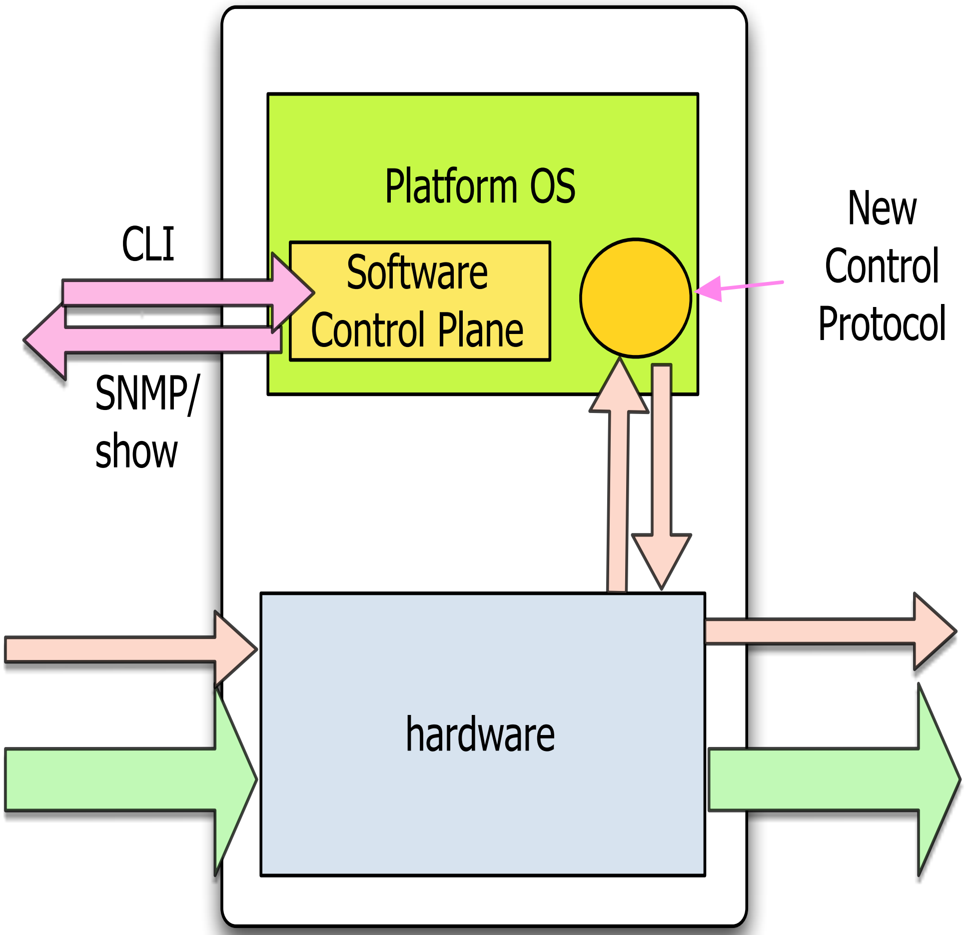
Unicorns and rainbows -- Wouldn't it be great if the HOpen platform existed today…?



Hybrid platform

Actually it does





- Linux based
- Drop in your own code / binaries next to vendor code
- What might be an interesting use case to develop on this platform?

Segment Routing

- Introduced to IETF in 2013
- Isn’t supported on deployed platforms

- SR 101:
  - Traffic engineering
  - Service chaining

- Comcast interested in IPv6 SR

Segment Router Packet

```
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 | Ver   |  Traffic Class|           Flow Label                   |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |        Payload Len             |    43 (SR)     | Hop Limit    |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                      Source Address                            |
...
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                   Destination Address                          |
...
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |    17 (TCP)    |  Hdr Ext Len   | Routing Type  | Next Segment |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 | Last Segment   | Flags |  HMAC Key ID  | Policy List Flags      |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
             Segment List[0] (128 bits ipv6 address)
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
...                     Segments (SIDs)
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
             Segment List[n] (128 bits ipv6 address)
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
...
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |           Policy List[0] (128 bits ipv6 address)               |
 |                        (optional)                              |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |           Policy List[1] (128 bits ipv6 address)               |
 |                        (optional)                              |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |           Policy List[2] (128 bits ipv6 address)               |
 |                        (optional)                              |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                    HMAC (256 bits)                             |
 |                      (optional)                                |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                    TCP header layer                            |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                       payload                                  |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

IPv6 Header

Segment Router Header

Transport & Payload

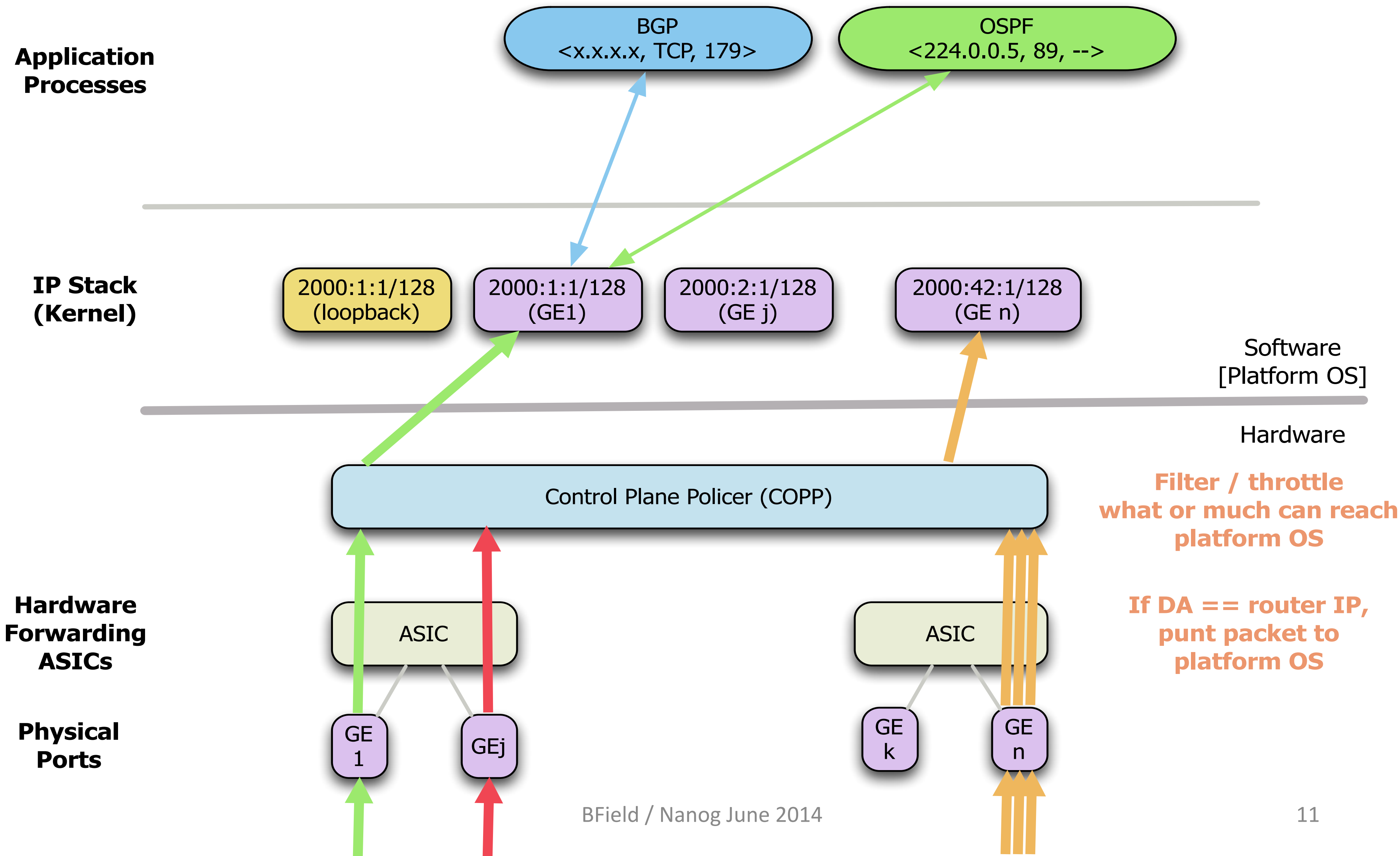
Segment Routing Operation

- IPv6 packet with SRH is delivered to Dest Addr .
  - Router or application
- Process data, then consult SRH and determine next "Segment" ID (SID)
- Update
  - SRH
  - IPv6 packet with next SID as next DA. Forward.

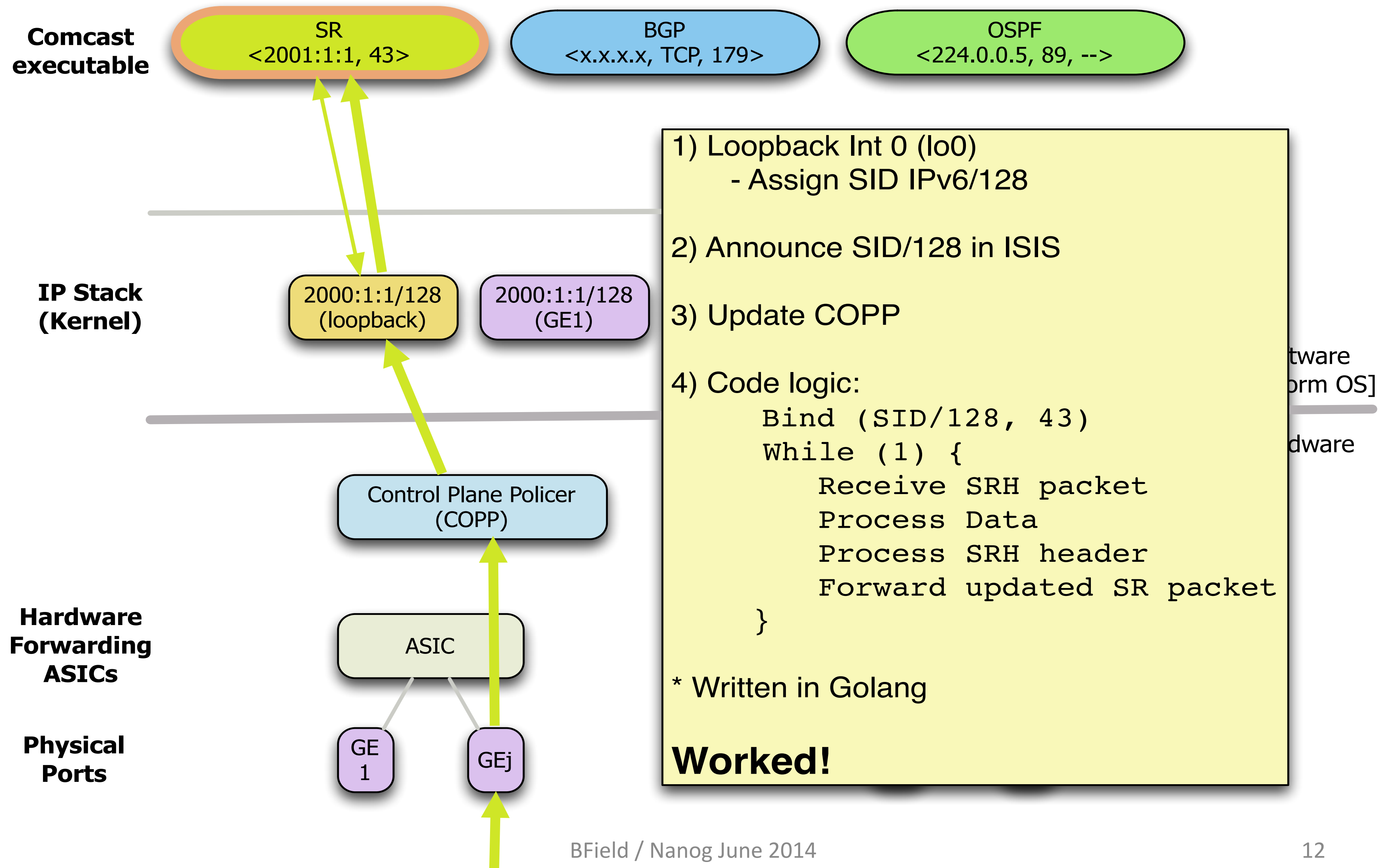
- Details shared in NANOG 58 – June 2013.
  - http://www.nanog.org/sites/default/files/wed.general.segment.filsfils.13.pdf

Router Control Plane Operation

[Punt traffic 101]



What we did



1) Loopback Int 0 (lo0)
   - Assign SID IPv6/128

2) Announce SID/128 in ISIS

3) Update COPP

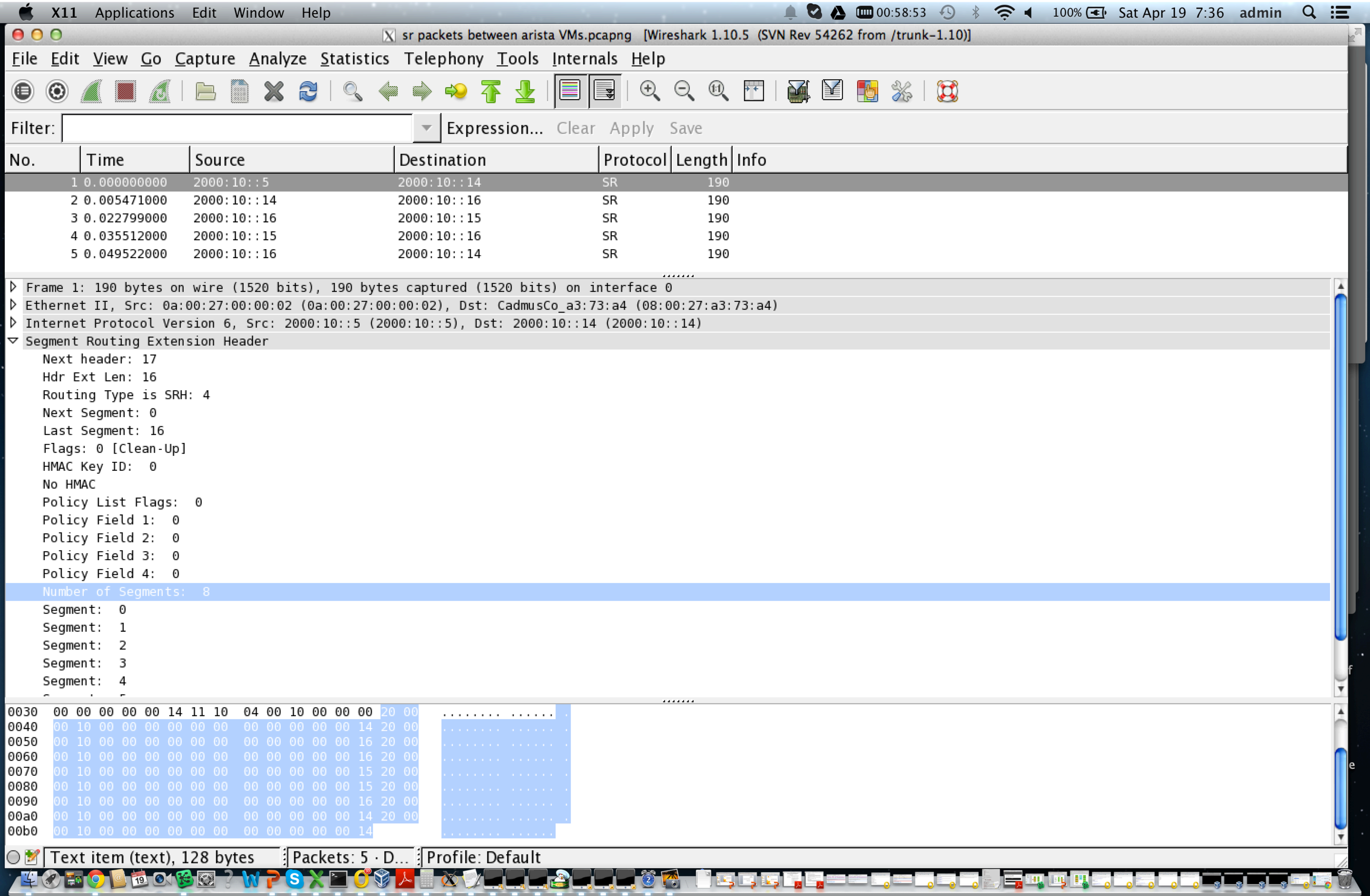
4) Code logic:

```
Bind (SID/128, 43)
While (1) {
    Receive SRH packet
    Process Data
    Process SRH header
    Forward updated SR packet
}
```

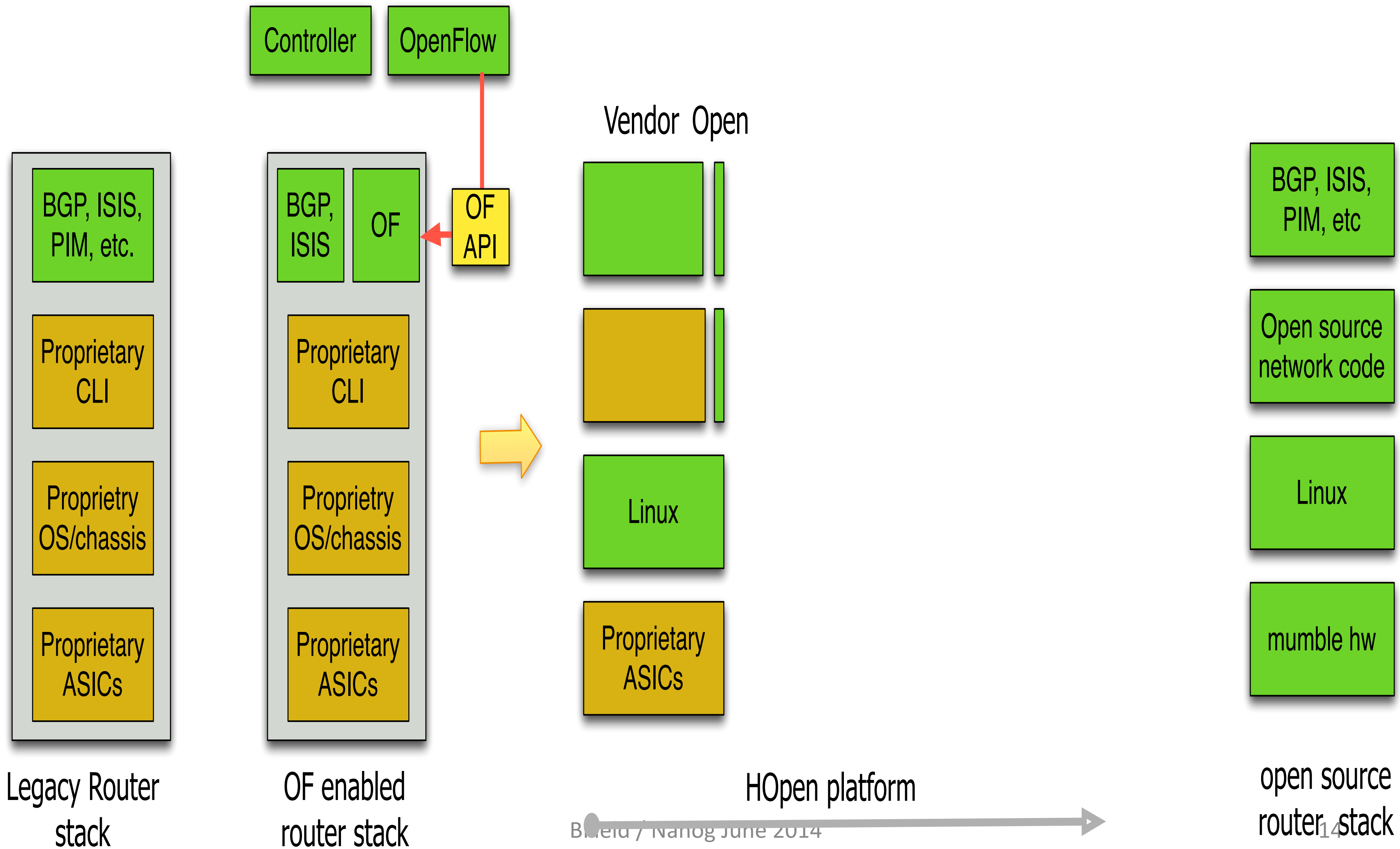
* Written in Golang

Worked!

SR packets on the wire



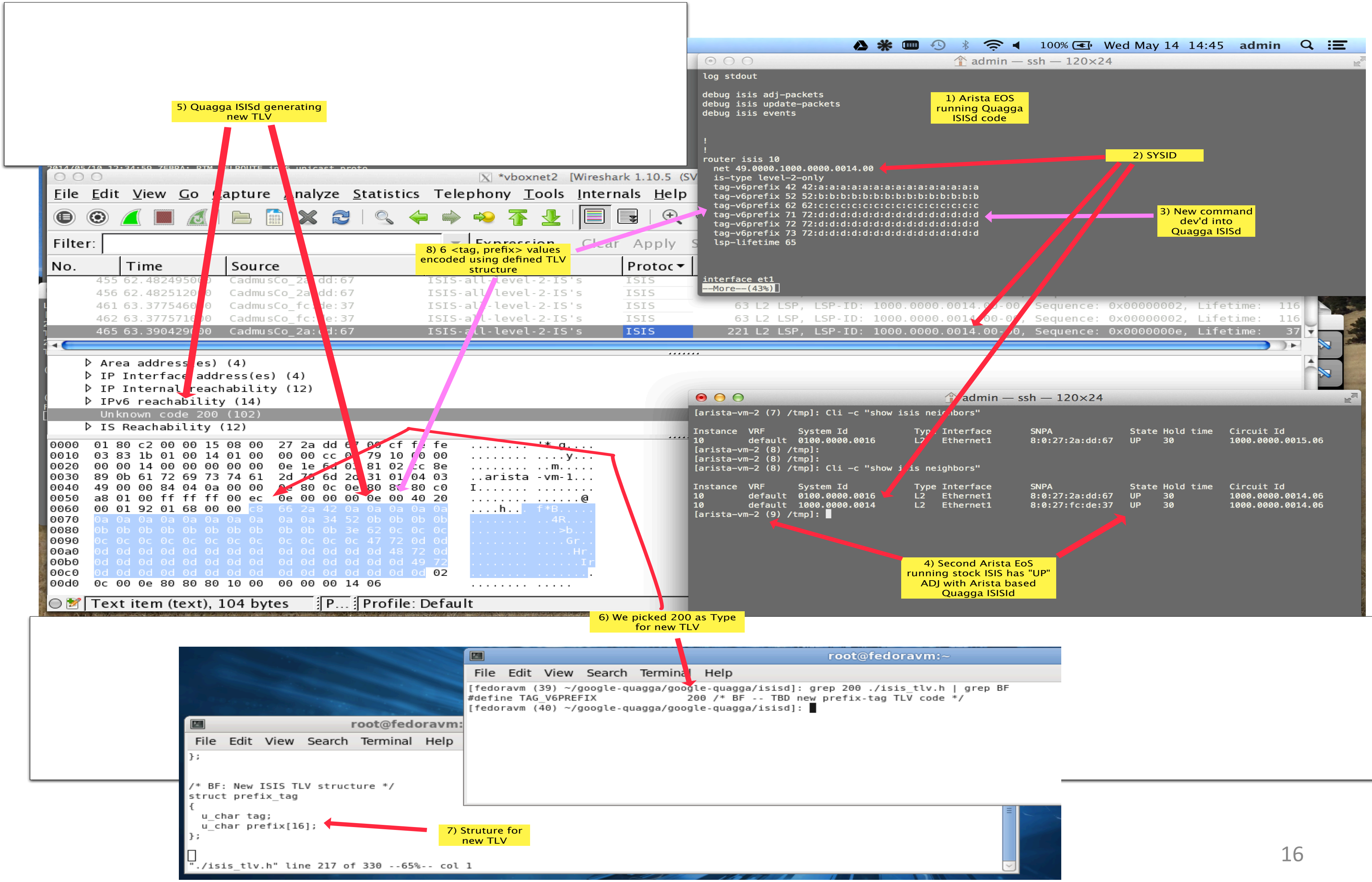
What did we just do? Took the first step



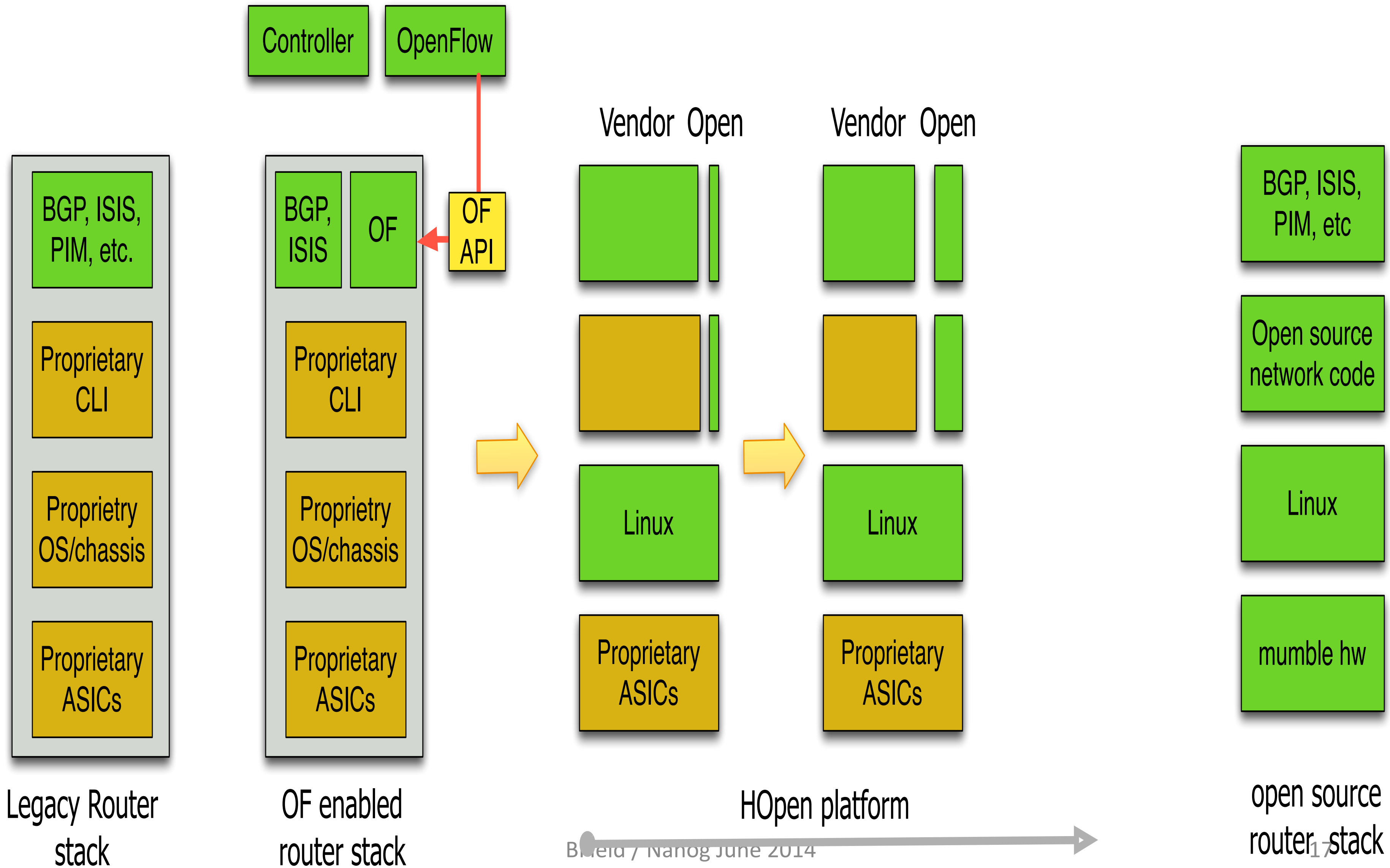
Extending … Wouldn’t it be great if…

- We had the ability for the SR process to tag it’s prefix announcements in ISIS that the /128 is “SR” capable?
  - Extend this to other services/features we might want to support per internal /128
- Crud– this vendor’s ISIS implementation doesn’t support ISIS admin tags…
- Wait, lets hack this new feature into open source ISIS and run that instead of the vendor’s ISIS!

Running Modified Open Source ISIS on HOpen Platform



Second step towards HOpen…



What does this mean?

- HOpen is an alternative SDN paradigm:
  - Leverage the control plane work the vendor has done
  - Supplement with Operator code or / Open Source for new features
- HOpen -- Best of both worlds:
  - Vendor support for “legacy” features
  - Operator can develop new features as needed, prove value, vendor rolls into their code base.
    - Maybe in an Open Source kinda way

Rough consensus and working code…

- This paradigm enables [unaffiliated] individuals to create, develop and deploy new control plane protocols on production routing platforms
- Does this change how things work in IETF?
- Might this be good?

- Does NANOG start to have:
  - Hackathon’s?
  - Inter-op work, etc.?

Quote time

"…With great power comes great responsibility…"

-- Voltaire (and later FDR, Ben Parker– Spiderman's uncle)

Thanks!

brian_field@cable.comcast.com