XORP and Virtual Routers

Mark Handley
Professor of Networked Systems,
University College London.

Building Blocks

- This talk is bottom up.
- We'll start with what we've got, explore what we can build with that, and speculate about where we might go.

- It's up to you to decide how to use these building blocks.
  - I've got my own ideas, but…

XORP

eXtensible Open Router Platform

Open source router software suite, *designed from the outset with extensibility in mind*.

- Main core unicast and multicast routing protocols.
- Event-driven multi-process architecture.
- BSD-style license
- 560,000 lines of C++

XORP Status: IGP Standards

RIP and RIPng:

- RFC 2453 (RIP version 2)
- RFC 2082 (RIP-2 MD5 Authentication)
- RFC 2080 (RIPng for IPv6)

OSPFv2:

- RFC 2328 (OSPF Version 2)
- RFC 3101 (The OSPF Not-So-Stubby Area (NSSA) Option)

XORP Status: BGP Standards

- **draft-ietf-idr-bgp4-26** (A Border Gateway Protocol 4 (BGP-4))
- **RFC 3392** (Capabilities Advertisement with BGP-4)
- **draft-ietf-idr-rfc2858bis-03** (Multiprotocol Extensions for BGP-4)
- **RFC 2545** (Multiprotocol Extensions for IPv6 Inter-Domain Routing)
- **RFC 3392** (Capabilities Advertisement with BGP-4)
- **RFC 1997** (BGP Communities Attribute)
- **RFC 2796** (BGP Route Reflection - An Alternative to Full Mesh IBGP)
- **RFC 3065** (Autonomous System Confederations for BGP)
- **RFC 2439** (BGP Route Flap Damping)

XORP Status: Multicast Standards

PIM-SM:

- draft-ietf-pim-sm-v2-new-11 (without SSM).
- draft-ietf-pim-sm-bsr-03

IGMP v1 and v2:

- RFC 2236

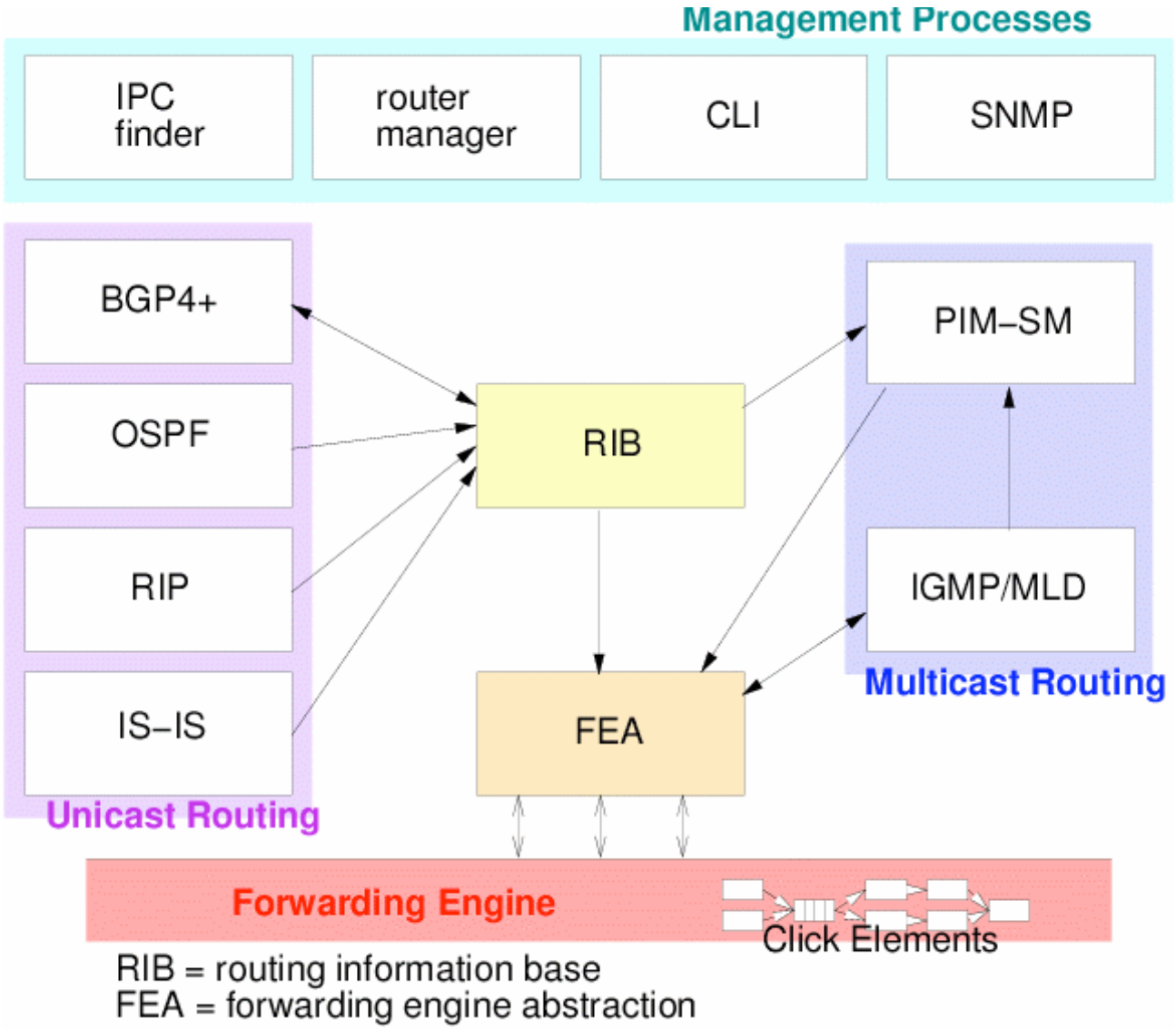
MLD v1:

- RFC 2710

XORP Processes

Multi-process architecture, providing isolation boundaries between separate functional elements.

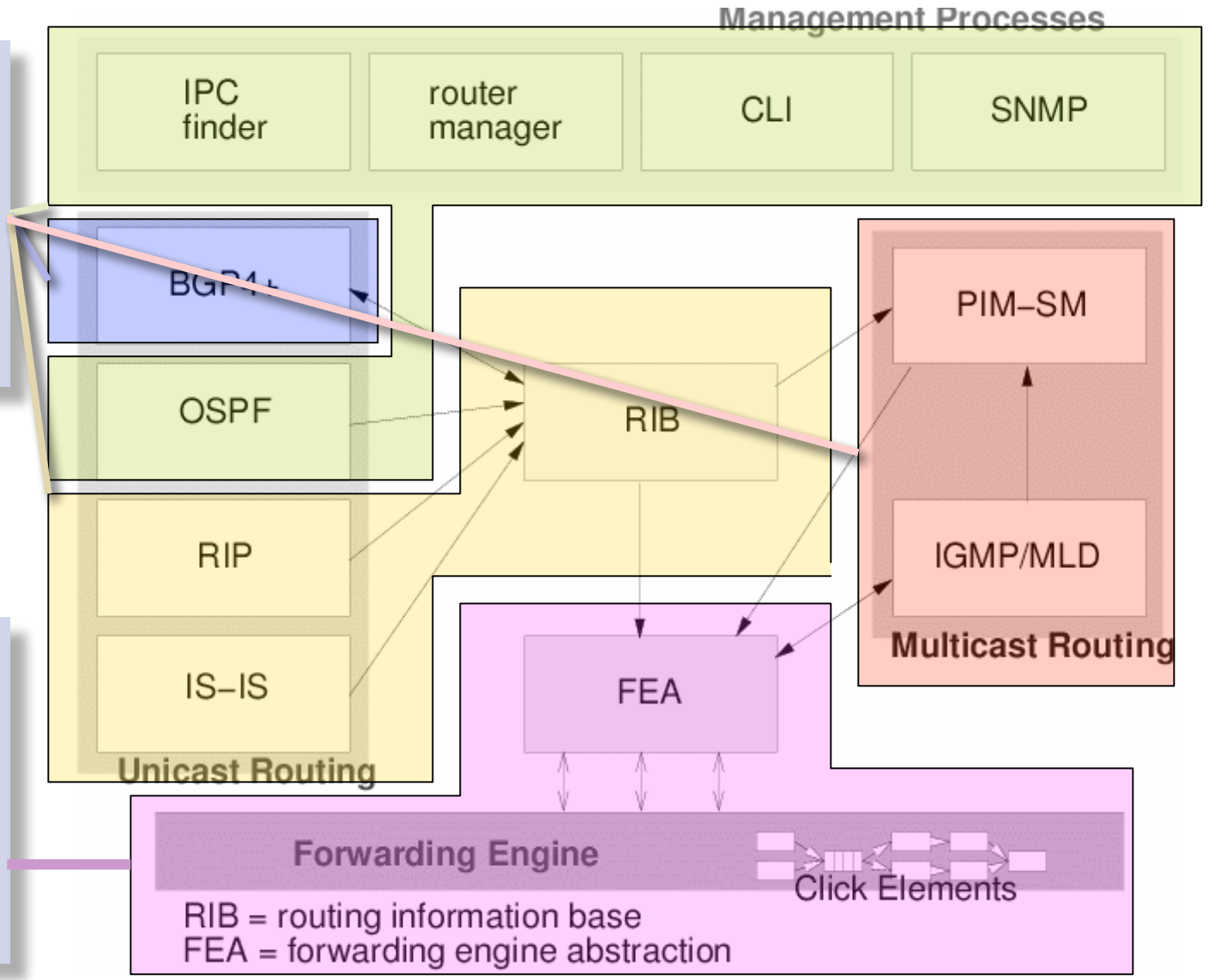
XRLs: Flexible IPC interface between modules



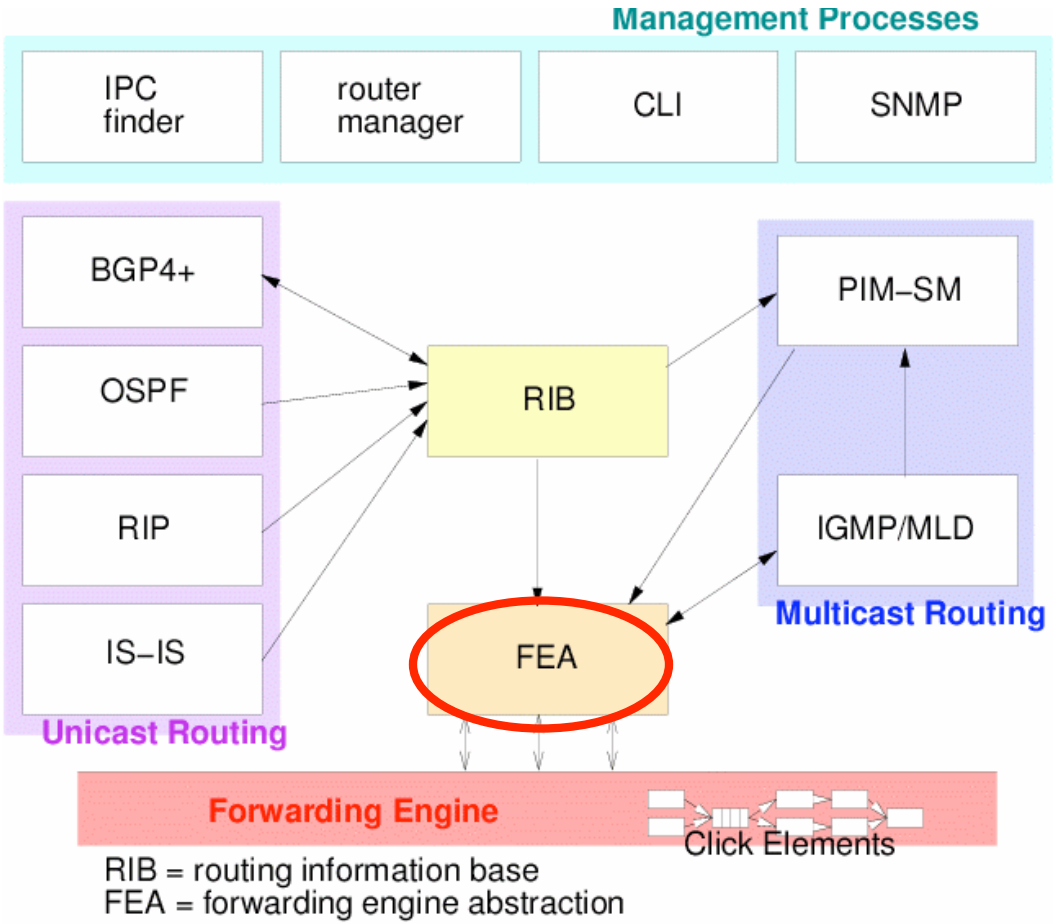
XORP and Distributed Routers (1)

Different routing and management processes can all be run on different machines

FEA talks to forwarding engine and provides an idealized API, remotely accessible



FEA: Forwarding Engine Abstraction

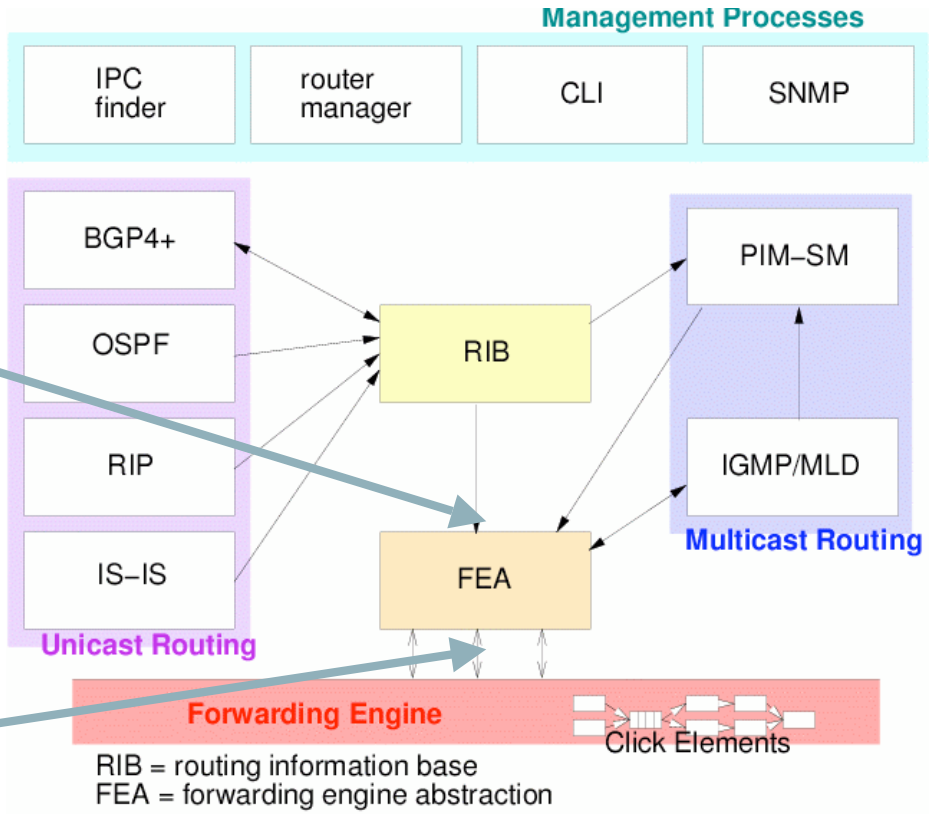


FEA: Forwarding Engine Abstraction

Main purpose of FEA is to provide a stable API to the forwarding engine.

Same XRL interface on All forwarding engines

Different OS calls.
Different kernel functionality.
Different hardware capabilities.
Multiple forwarding engines



FEA Functionality

Interface Management:

- Discover and configure network interfaces.
- Processes can register interest in interface state changes.

Routing:

- Sends unicast routes to the forwarding engine.
- Sets/removes multicast forwarding state.
- Relays notifications (IGMP, PIM Messages, etc)

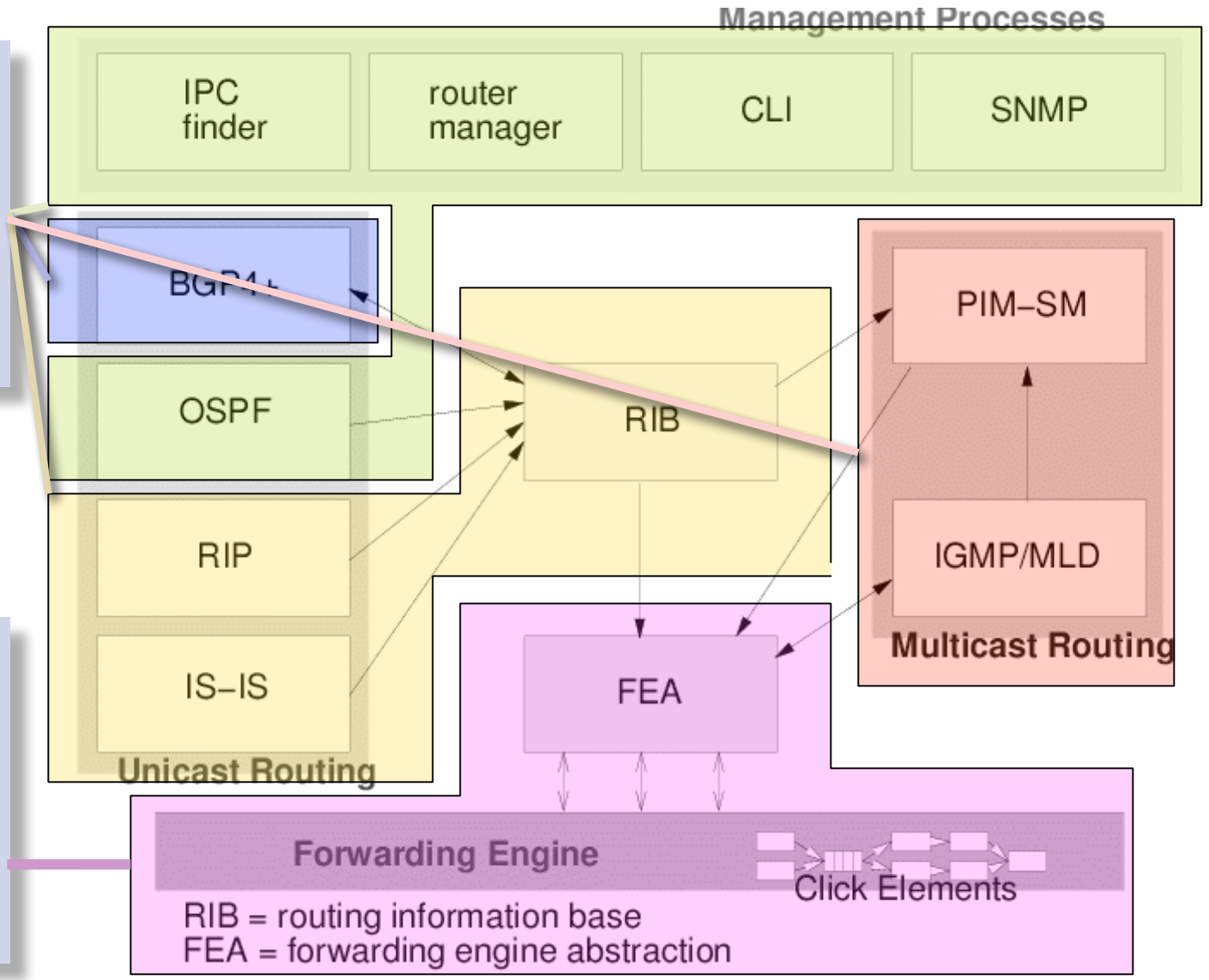
Relay Routing Traffic:

- Routing protocols send packets via XRLs to the FEA
- Don't run as root, even if they need to send on a raw socket.
- Sandboxing can limit what a bad process can send and receive.

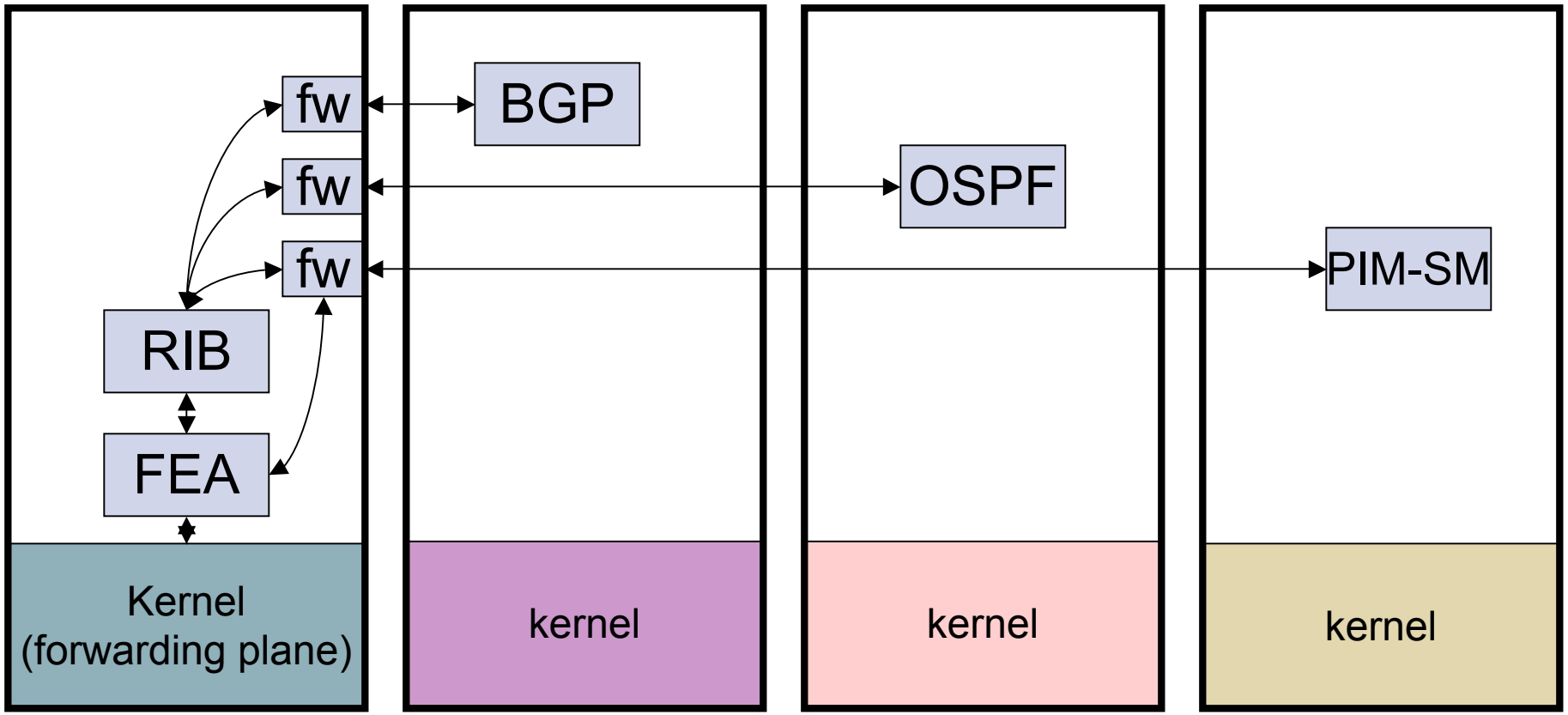
XORP and Distributed Routers (1)

Different routing and management processes can all be run on different machines

FEA talks to forwarding engine and provides an idealized API, remotely accessible



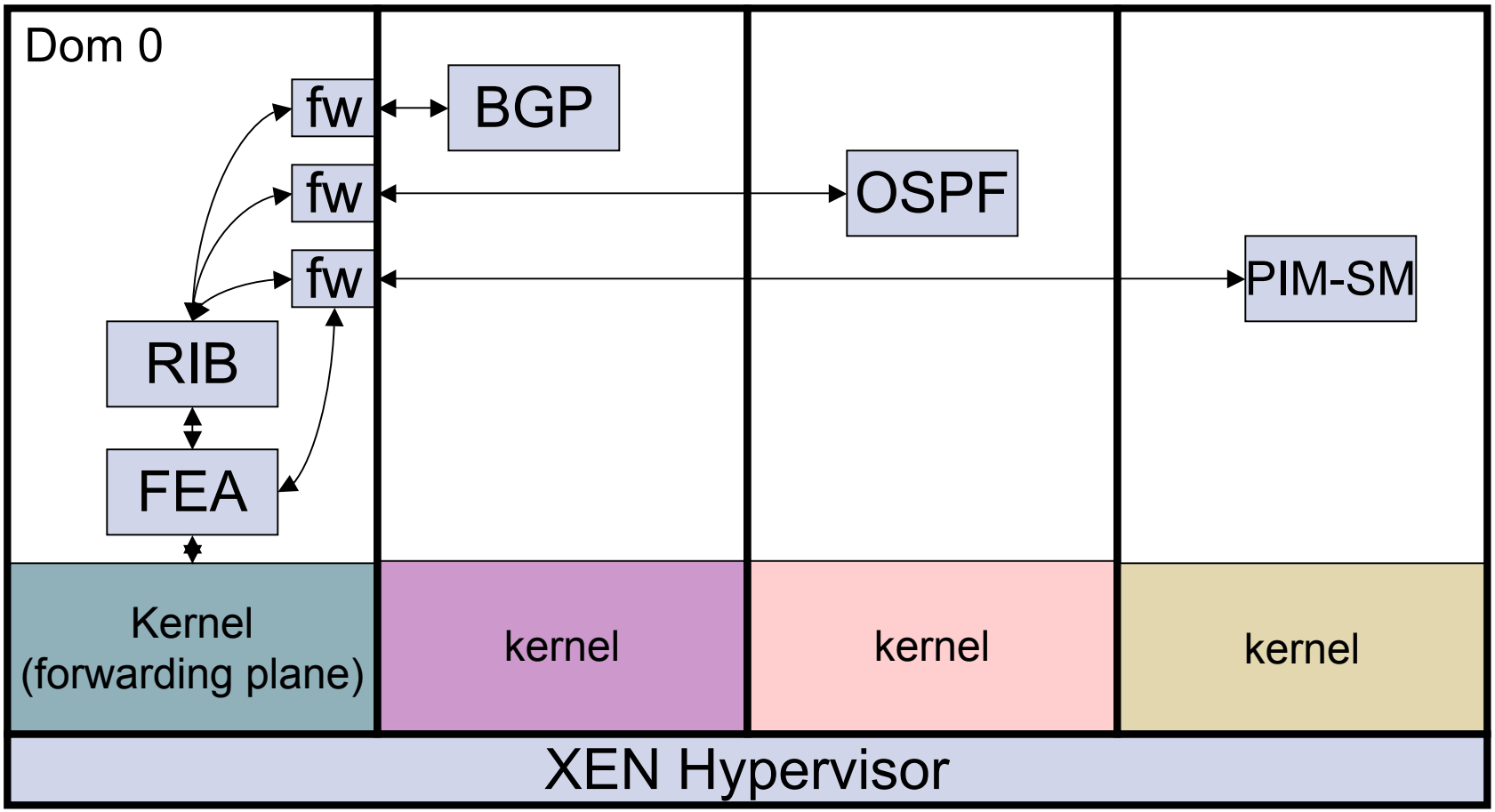
Distributed Router



Distributed Routers. So what?

- Many boxes:
  - Lots of CPU cycles: can do stuff you couldn't do on a router before.
  - Isolation - compromised PIM-SM doesn't affect your BGP.

Host Virtualization within a Router



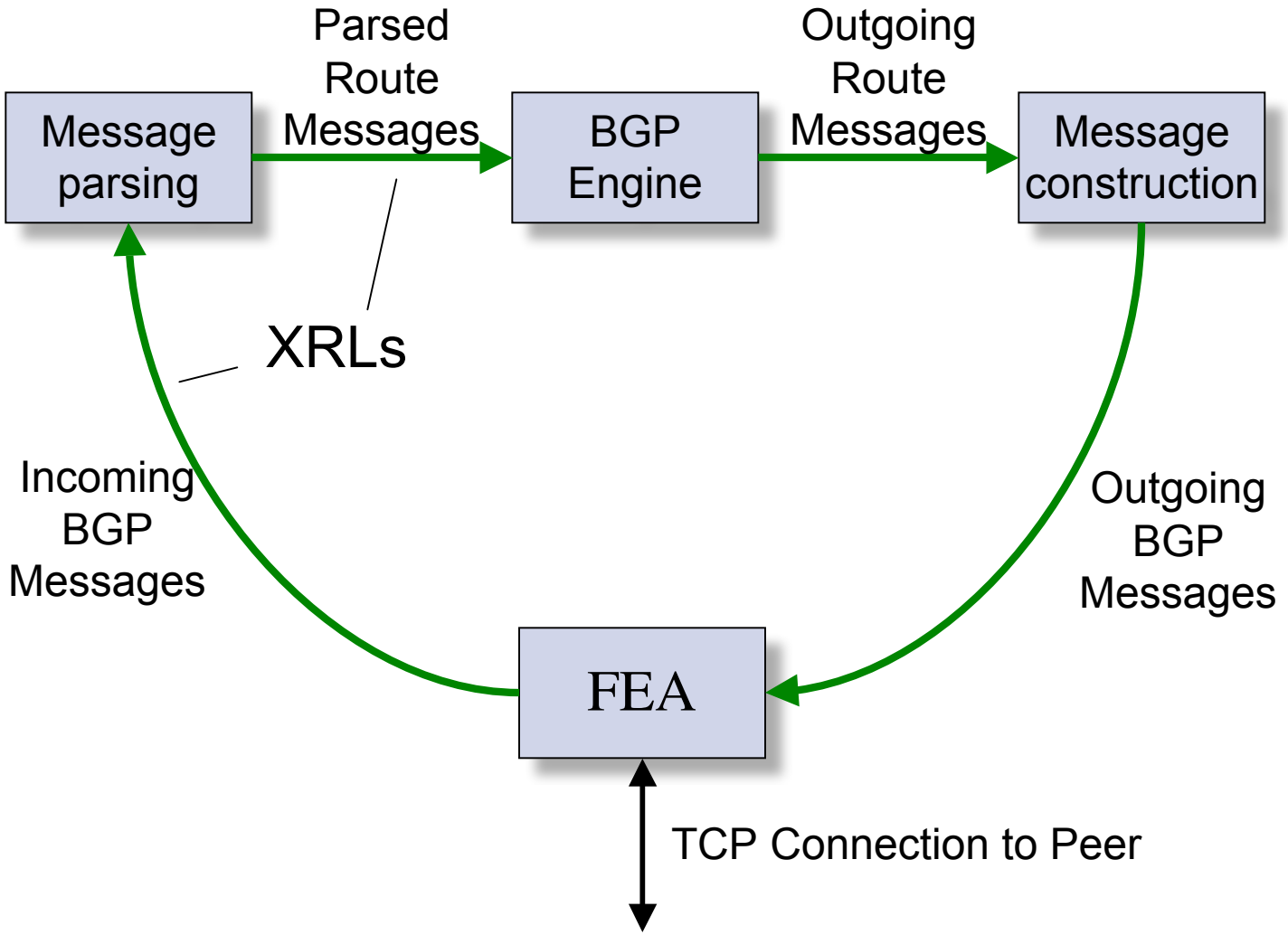
Virtualized Route Processor: Security

- Use virtualization to isolate processes.
  - Can only interact with the rest of the world through XRLs.
- Use XRL firewalling to restrict XRLs and their parameters needed for the task at hand.
  - Can use a template file to specify process permissions.
- Result:
  - An experimental process can do very little harm if it malfunctions or gets compromized.

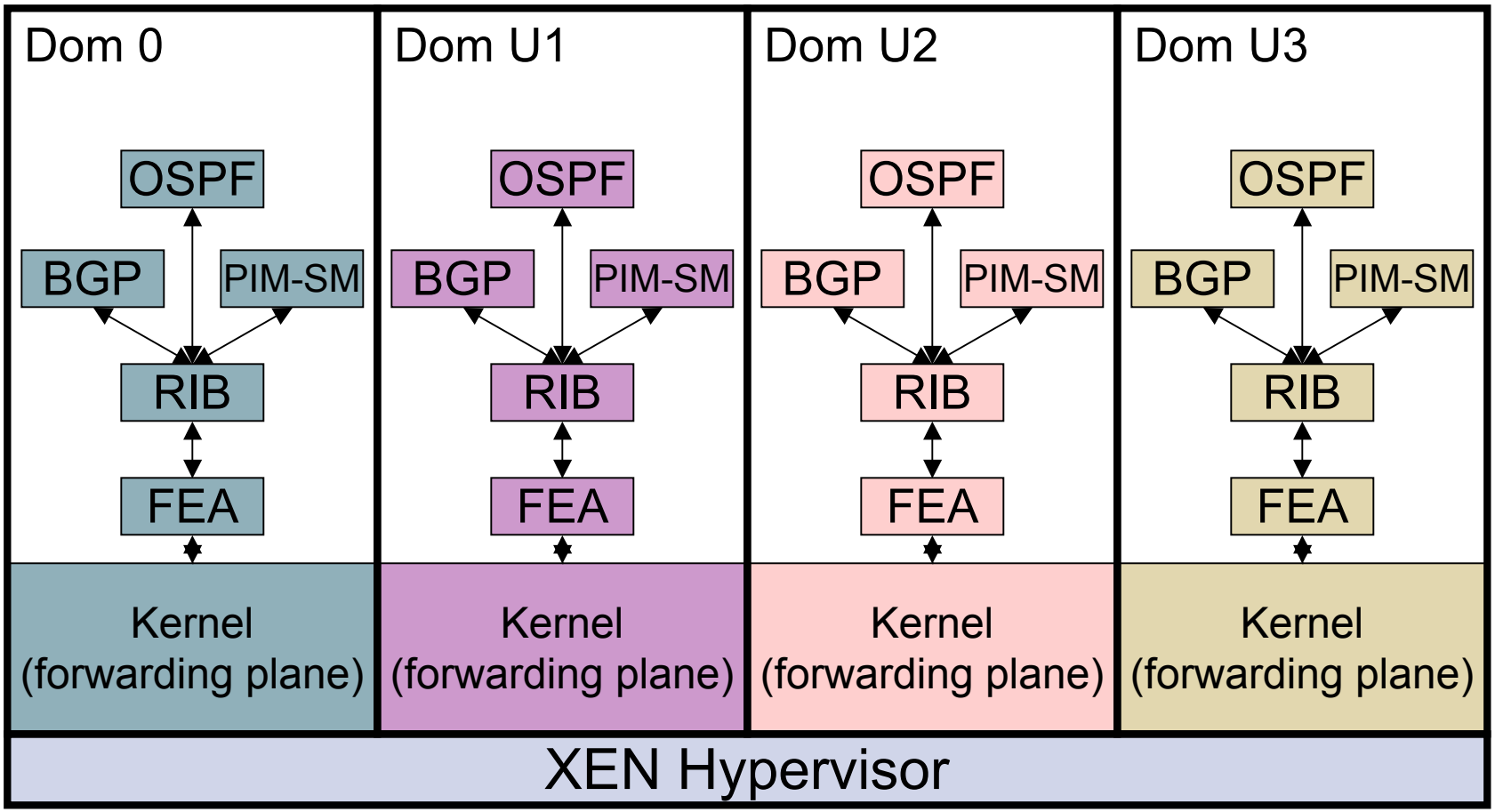
Router Worms

- If I can compromise one BGP, I can compromise them all and shut down your network.
  - Don’t need to break out of the sandbox to do damage.
- Can we prevent router worms?

Router Worm Prevention



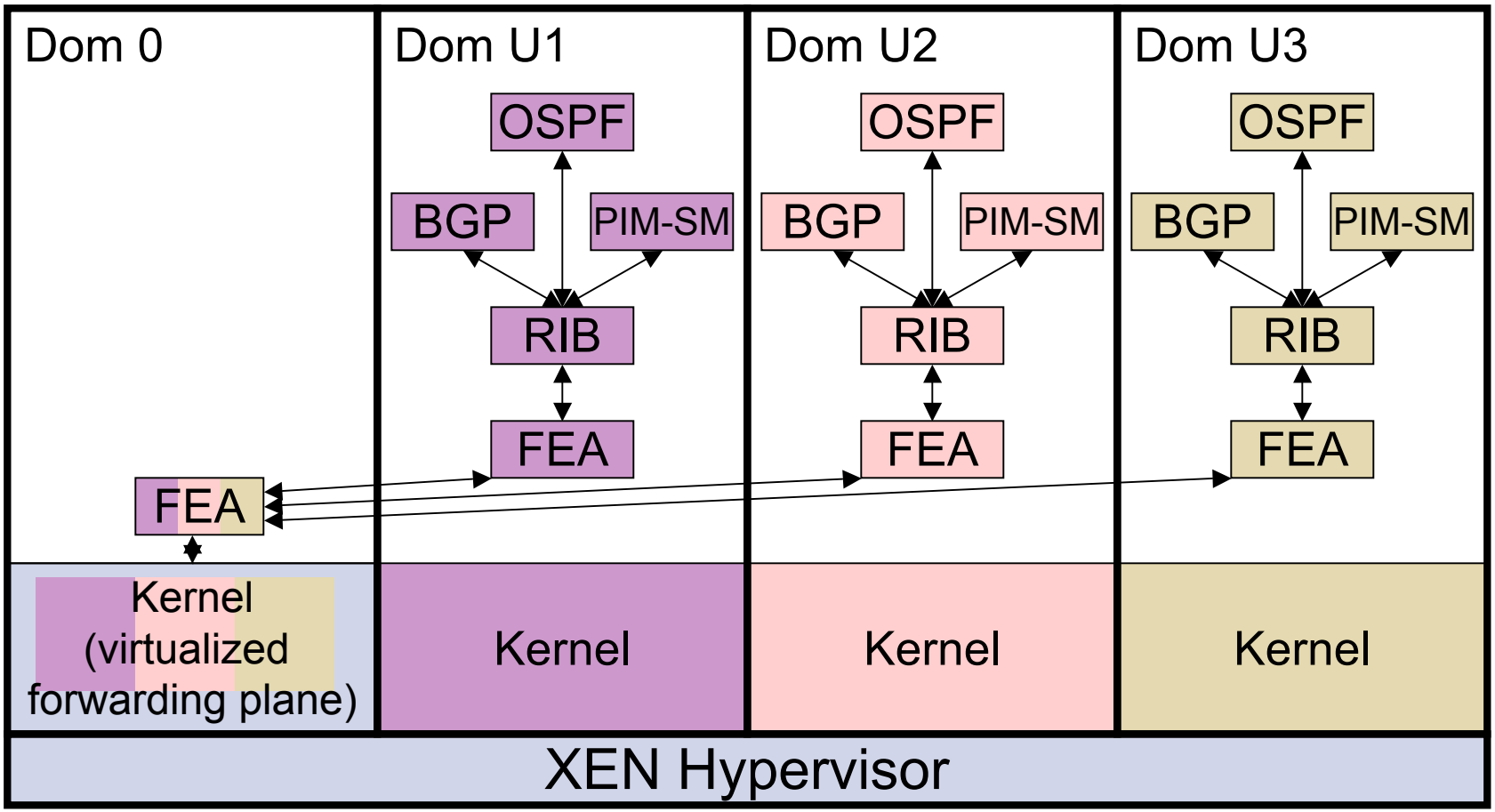
Virtual Routers



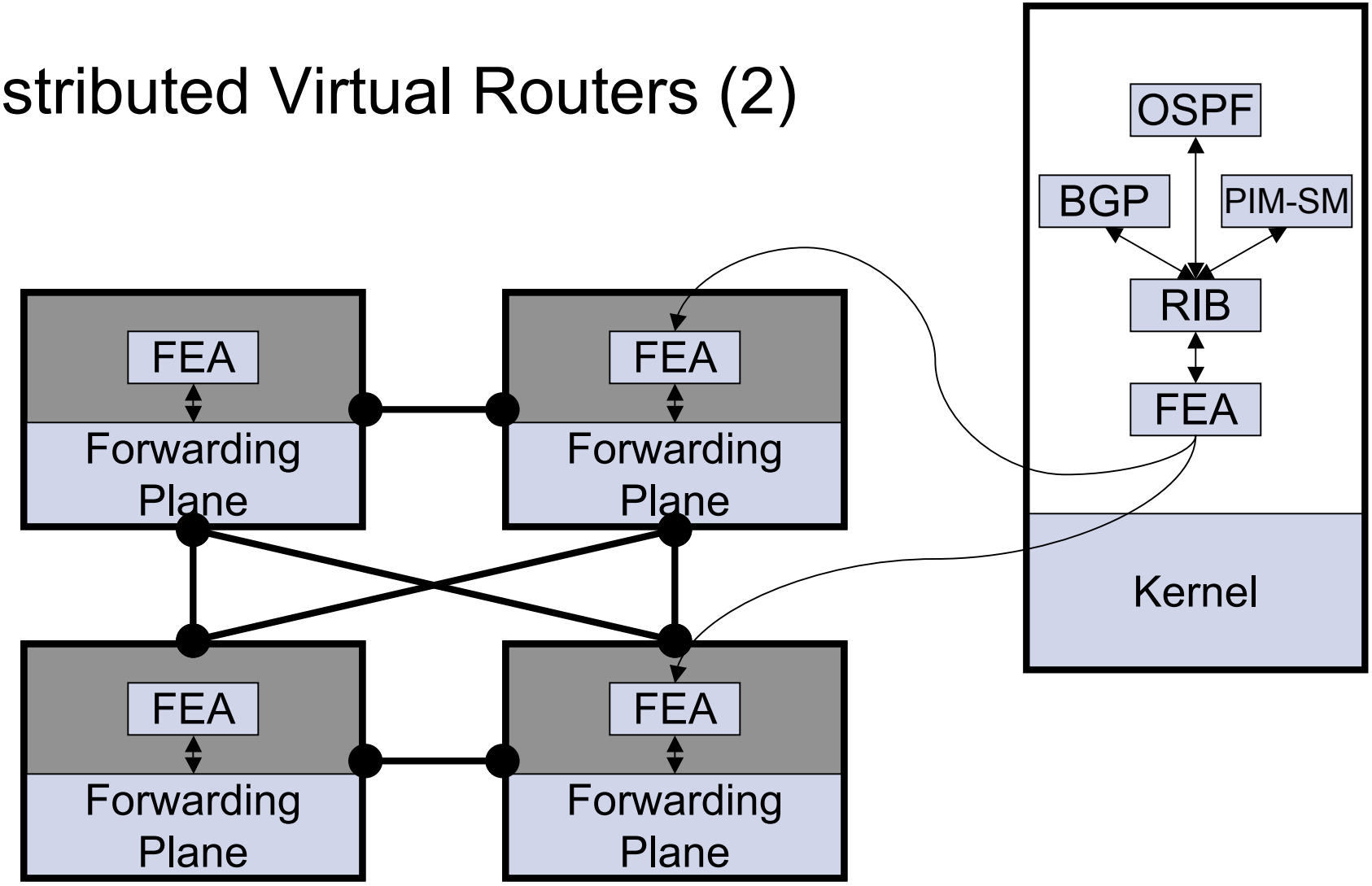
Virtual Routers: What's the Point?

- One box can play the role of multiple independent routers.
- Multiple organizations sharing a single physical router.
  - Small businesses within one building.
- Entire physical network can be shared.
  - Virtualize the routers.
  - Tunnel between virtual routers over shared IP infrastructure.
  - Great platform for experimentation (cf. VINI).
  - Alternative to MPLS infrastructure for VPNs.
    - VPN can be inter-domain, multi-homed.
    - Some virtual routers belong to ISP, lease slice to customer.
    - Others belong to edge-network; good isolation between internal and external traffic on same phyical links.
    - Integrate better into public IP infrastucture at multple locations.

Virtual Routers



Distributed Virtual Routers (2)



Distributed Virtual Router

- Many boxes form the forwarding plane of a single virtual router.
  - Outside world only sees one router in OSPF, BGP, etc.
  - Internally, packets may be forwarded between the nodes of the router.
- Nodes can be in same rack.
  - Build one big router from many small cheap components.
- Nodes can be in same POP.
  - POP internal topology isolated from external routing.
- Nodes can be widely distributed.
  - CF Centralized Routing.
  - Can tackle control algorithms that don't distribute well.
  - Need to be very careful about fate sharing.

“The Big Rack of PCs” Router

- Most backbone routers use a fast hardware forwarding path.
  - Hard to beat for pure dumb forwarding.
- Modern x86 CPUs good at forwarding at speeds of 1Gb/s.
  - Trend towards multiple cores good for software forwarding.
  - Software advantage is flexibility.
- Distributed software routers favour router applications.
  - Flexibility in distributing load across the cluster.
  - Most obvious applications are in security area:
    - IDS, IPS, Firewall, DoS defense.
  - Enabler for any CPU-intensive in-network tasks.

Conclusions

- XORP FEA aimed at abstracting out differences in forwarding places.
- Turns out having an abstract forwarding plane is ideal for building virtual routers.
  - Distribute routing protocols across multiple boxes.
  - Distribute routing protocols across multiple VMs.
  - Put multiple routers in multiple VMs on one box.
    - Virtualize entire network.
  - Make multiple boxes behave like one router.
    - Network is inside the router.
  - Secure router against worms.
- Suddenly good old IP networks seem a lot more flexible than they used to.

Summary

- Virtual routers are an enabling technology.
  - Actually several enabling technologies.
- Maybe provide deployment pathway for new architectures without expecting everyone to throw away the Internet first.
- What ideas do you have for using them?