Fibbing: Central Control over Distributed Routing

www.fibbing.net

Olivier Tilmans
UCLouvain

Routing Area Open Meeting
Nov. 15, 2016

Joint work with
S. Vissicchio (UCL), L. Vanbever (ETH Zürich) and J. Rexford (Princeton)
Fibbing
Fibbing

Control routers’ FIB, lying to routers
Consider this example network.
Consider this example network.
Link-state Interior Gateway Protocols (IGPs) exchange reachability information to infer the topology of the network.
The intra-domain traffic flows along the shortest path on the shared topology.

Control-Plane

Data-Plane
IGPs cause operators to follow a **descriptive** management process.
Fibbing: Central Control over Distributed Routing

1. Controlling distributed protocols
2. Fibbing today’s networks
3. Case study: On-demand load-balancing
4. Takeaways
Fibbing enables declarative management.

- Centralizes high level routing decisions. Operators describe their requirements to a controller.

- Keeps the route installation distributed. Each router independently computes its FIB.

- Leverages the IGP messages as API. We study which message to send.
Operators specify paths that must be enforced.

requirements
(A, R/one.pnum, R/two.pnum, C, blue)

Fibbing controller
The controller injects one IGP message adding a fake node and links.

requirements
(A, R1, R2, C, blue)

node fA (blue),
link (fA, A, 2),
map (fA, A) to (A, R1)
IGP flooding propagates the information.

requirements
(A, R1, R2, C, blue)

node fA (blue),
link (fA, A, 2),
map (fA, A) to (A, R1)
The Fibbing message *augments* the topology.

**requirements**

\(\text{A, R1, R2, C, blue}\)

node fA (blue),
link (fA, A, 2),
map (fA, A) to (A, R1)
Augmented topologies translate into new control-plane paths.

requirements
(A, R1, R2, C, blue)

node fA (blue),
link (fA, A, 2),
map (fA, A) to (A, R1)
Augmented topologies translate into new data-plane paths.

**Control-Plane**

- Requirements: (A, R1, R2, C, blue)
- Node fA (blue), link (fA, A, /two.pnum), map (fA, A) to (A, R1)

**Data-Plane**

- Connections and routes as shown in the diagram.
Chaining multiple fake nodes enables to program complex paths.

requirements
(A, R1, R4, C, blue)

node fA (blue), link (fA, A, 2), map (fA, A) to (A, R1)
Chaining multiple fake nodes enables to program complex paths.

requirements

(A, R1, R4, C, blue)

node fR1 (blue),
link (fR1, R1, 2),
map (fR1, r1) to (R1, R4)
Chaining multiple fake nodes enables to program complex paths.

requirements
(A, R1, R4, C,blue)
Augmented topologies can be reduced to optimize the number of fake nodes.

**Naive augmentation**

**Reduced augmentation**
Fibbing preserves the scalability of IGPs.

- **We can compute augmented topologies in $O(ms)$**
  Ensures quick reaction to changes

- **We can *reduce* augmented topologies in $O(s)$**
  Ensures limited control-plane overhead
Fibbing leverages the robustness of IGPs.

- Fast failure detection and recovery
- Survive controller failure
- Support fail-close and fail-open semantics
Fibbing can enforce any set of loop-free paths, on a per destination basis.
Fibbing: Central Control over Distributed Routing

www.fibbing.net

1. Controlling distributed protocols
2. Fibbing today’s networks
3. Case study: On-demand load-balancing
4. Takeaways
We have a working Fibbing controller prototype.

- The controller maintains an OSPF adjacency to one router
- Topology discovery using the adjacency
- Tested against IOS, NX-OS, JunOS
The "forwarding address" has one other application. It enables routers in the Autonomous System’s interior to function as "route servers".

RFC2328, §2.3—Use of external routing information
If the forwarding address is non-zero, [...] install the AS external path to N, with next hop equal to the list of next hops to the forwarding address.

RFC2328, §16.4—Calculating AS external routes
Fake nodes can be injected using LSA types 5/7.

RFC2328, §A.4.5—AS-external-LSAs
Using T5/7 LSAs comes at a price.

- Different expressivity model
- Can only affect prefixes from other T5/T7 LSAs
- Does not exist in IS-IS!
Using T5/7 LSAs has (almost) no overhead on routers and is fast.

- No measurable impact on SPF duration
- 10 000 LSAs eat 14.5 MB of DRAM
- 900 $\mu$s to push one fibbed route to the FIB
Fibbing: Central Control over Distributed Routing

www.fibbing.net

1. Controlling distributed protocols
2. Fibbing today’s networks
3. Case study: On-demand load-balancing
4. Takeaways
The initial IGP configuration has a bottleneck towards router C.

Control-Plane
The initial IGP configuration has a bottleneck towards router C.

Control-Plane

Data-Plane

Link load

Overloaded link

Server

Client
Fibbing can program on-demand ECMP to spread the load.

Control-Plane
Fibbing can program on-demand ECMP to spread the load.

Control-Plane
Fibbing can program on-demand ECMP to spread the load.

Control-Plane

Data-Plane
Fibbing can program on-demand ECMP to spread the load.

Control-Plane

Data-Plane
Fibbing controls the splitting ratios across equal-cost paths.

Control-Plane
Fibbing controls the splitting ratios across equal-cost paths.

**Control-Plane**

![Diagram showing control-plane network with nodes A, B, C, R1, R2, R3, and R4 connected by weighted edges.]
Fibbing controls the splitting ratios across equal-cost paths.
The Fibbing controller reacts and adds more paths to spread the load, in real time.

We initially have 1 flow from S1 to D1.

- At time $t = 14s$, we start 30 new flows from S1 to D1.
- At time $t = 35s$, we start 30 flows from S2 to D2.
Fibbing: Central Control over Distributed Routing

www.fibbing.net

1. Controlling distributed protocols
2. Fibbing today's networks
3. Case study: On-demand load-balancing
4. Takeaways
Modifying the shared topology is powerful.

- Enables optimal, real-time, TE in the control plane
- Does not impact data-plane
- Gives *some* control over BGP/MPLS-LDP
- Simplifies configurations through exception-based routing