Flexible Topologies
TC 5.0
Who am I?

Jeremy Mitchell

github.com/mitchell852

- Traffic Control contributor for 5+ years
- Traffic Control committer / PMC member
- Traffic Portal original author
- Traffic Ops API (Perl & Go) contributor
What is a topology?
[Cache Server] Topology

- Defines how CDN cache servers are arranged:
  - Geographically
  - Hierarchically

E = Edge Cache / Reverse Proxy
Global Cache Server Topology

E = Edge Cache / Reverse Proxy
M = Mid Cache / Forward Proxy
Delivery Service
Cache Server
Topology
Semi-Flexible Topologies (today)

- The number of caching layers or tiers for your DS cache server topology is semi-flexible:
  - Determined by DS type
    - 1 tier: EDGE Only (DNS_LIVE, HTTP_LIVE, HTTP_NO_CACHE)
    - 2 tiers: EDGE + MID

- The cache servers for your DS cache server topology is semi-flexible:
  - Determined by:
    - DS/Edge Server Assignments
    - Server capabilities
    - The parent/child relationships between cache groups
Semi-Flexible Topologies (today)

- The number of caching layers or tiers for your DS cache server topology is semi-flexible:
  - Determined by DS type
    - 1 tier: EDGE Only (DNS_LIVE, HTTP_LIVE, HTTP_NO_CACHE)
    - 2 tiers: EDGE + MID

- The cache servers for your DS cache server topology is semi-flexible:
  - Determined by:
    - DS/Edge Server Assignments
    - Server capabilities
    - The parent/child relationships between cache groups
Origin

Delivery Service
Topology
Example 1
Delivery Service
Topology
Example 2
Delivery Service Topology Example 1
Semi-Flexible Topologies (today)

- The number of caching layers or tiers for your DS cache server topology is semi-flexible:
  - Determined by DS type
    - 1 tier: EDGE Only (DNS_LIVE, HTTP_LIVE, HTTP_NO_CACHE)
    - 2 tiers: EDGE + MID

- The cache servers for your DS cache server topology is semi-flexible:
  - Determined by:
    - DS/Edge Server Assignments
    - Server capabilities
    - The parent/child relationships between cache groups
Flexible Topologies (TC 5.0)

- The number of caching layers or tiers for your DS cache server topology is MORE flexible:
  - 1+ tiers

- The cache servers for your DS cache server topology is MORE flexible:
  - Determined by:
    - The parent/child relationships between cache groups
    - Server capabilities
    - DS/Server Assignments
Demo
Flexible Topology Features in Traffic Ops, Traffic Monitor, and Traffic Router

As designed by Rawlin Peters
Presented by Zach Hoffman

Download slides from people.apache.org/~zrhoffman/flexible-topologies.pdf
About the Speaker

Zach Hoffman – Software Engineer 3 at Comcast

• With the Comcast VIPER CDN team since December 2019
• Was a full-stack developer supporting higher education before that
• Committer for Apache Traffic Control since 26 days ago
• I will play you in Tetris

GitHub: zrhoffman
Apache: zrhoffman@apache.org
POST /api/3.0/topologies

Post body:

```json
{
    "name": "demo1-top",
    "description": "A topology of the CDN-in-a-Box cachegroup parentage",
    "nodes": [
        {
            "cachegroup": "CDN_in_a_Box_Edge",
            "parents": [1]
        },
        {
            "cachegroup": "CDN_in_a_Box_Mid-01",
            "parents": [2]
        },
        {
            "cachegroup": "CDN_in_a_Box_Mid-02",
            "parents": []
        }
    ]
}
```
Topologies Validation

- A given cachegroup **cannot** be used twice in the same topology
- An Edge Cachegroup **is** able to parent another Edge Cachegroup
- An Edge Cachegroup **cannot** parent a Mid Cachegroup
- A mid cachegroup in a topology **must** have at least 1 child node
Topologies Validation—cycles

- A topology’s cachegroups/topology nodes must not form a *cycle*
Topologies Validation—more cycles

- If you combine all of the topologies, that is called the **static topology**
- The static topology must not contain any cycles
GET /api/3.0/cdns/{name}/snapshot

- Has a new `topologies` section
- Adds `server capabilities` in `contentServers` section
- Adds `requiredCapabilities` field to `deliveryServices` section
- No direct association between delivery services and servers, must be calculated using the `Snapshot`
Topologies and Server Capabilities

- Server capabilities: **RAM** and **HDD**
- Delivery service with ID **small-files** that requires capability **RAM**
An abbreviated Snapshot
Topologies changes in Traffic Monitor

• Almost no Topologies changes
• Creates Delivery Service~Server associations for Servers in the Cachegroups of a Topology-based Delivery Service
• Lets Traffic Monitor mark a Topology-based Delivery Service as healthy
Topologies changes in Traffic Router

• Only looks at the edge caches of the topology
• Adds each edge cache in the topology to the delivery service if
  the list of the edge cache's server capabilities contain all of the delivery service's required capabilities
• Traffic Router will only send requests to this delivery service if Traffic Monitor marks this delivery service as healthy
Questions?
Thank you!

Further reading:

Flexible Topologies Proposal Blueprint (by Rawlin Peters):
  github.com/apache/trafficcontrol/pull/4537

Flexible Topologies Milestone:
  github.com/apache/trafficcontrol/milestone/10

Link to slides:
  people.apache.org/~zrhoffman/flexible-topologies.pdf

Apache Traffic Control Documentation:
  traffic-control-cdn.readthedocs.io