



The Great Cloud Migration with Network Automation & Service Mesh

**We want 50% of our applications
on cloud in 2 years.**

**For the next 5 years, we will run
across two or more platforms.**

**** Draws lots of diagrams. ****



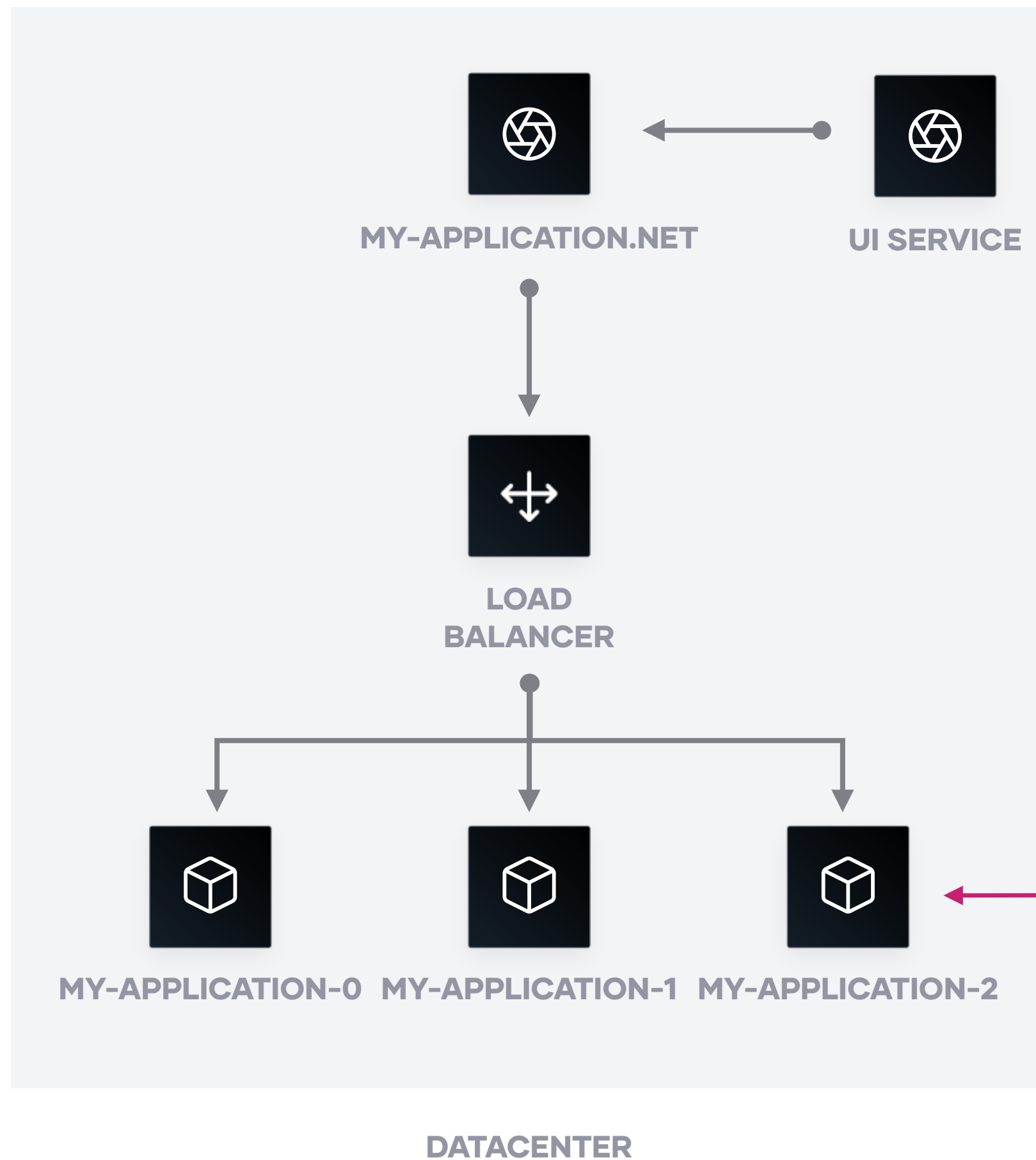
Rosemary Wang

(she/her)

Developer Advocate at HashiCorp
joatmon08.github.io

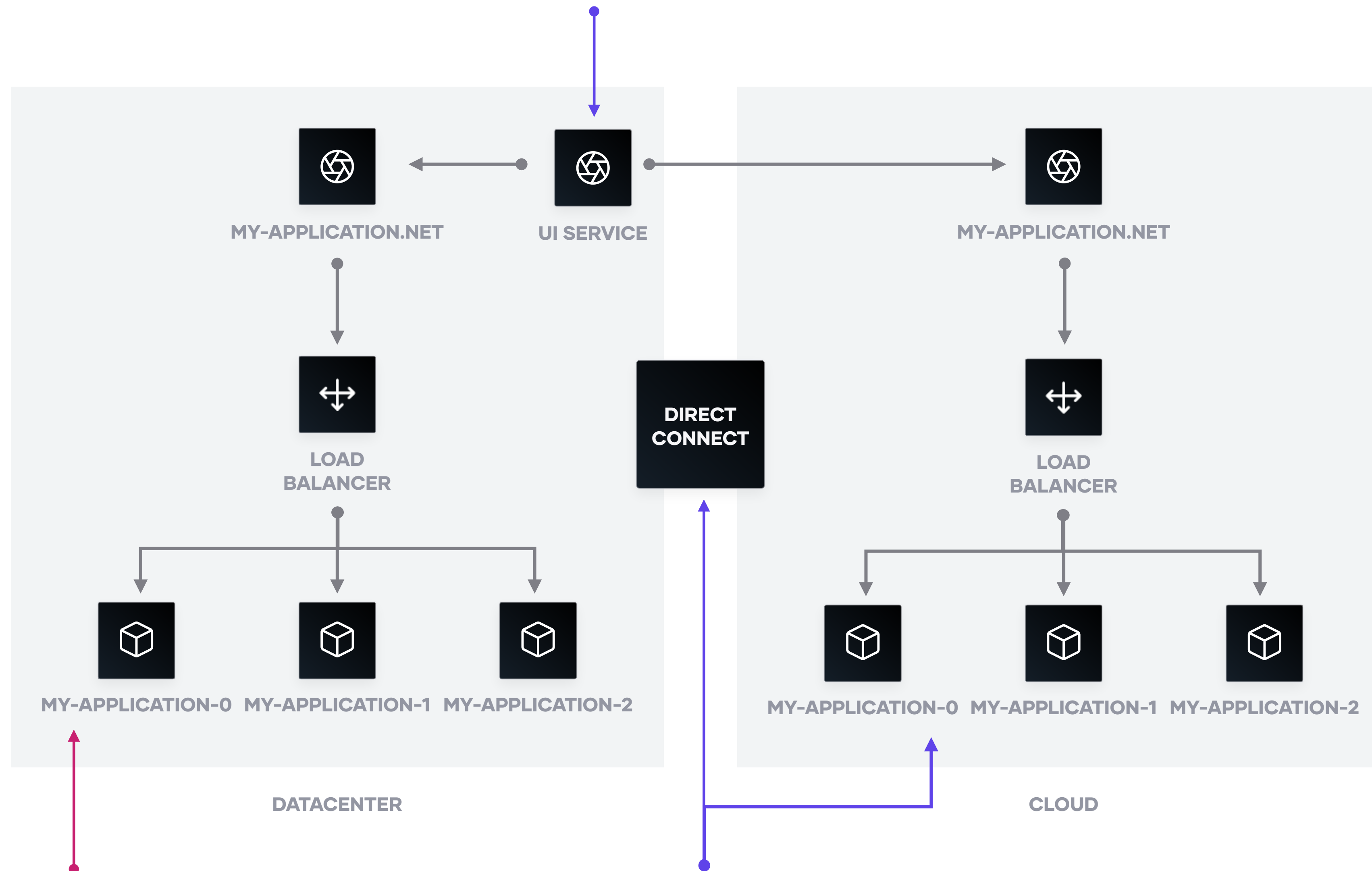


Premise



- Internal application
- Runs on a dedicated server
- Multiple instances for availability
- Used by UI in datacenter
- It must be re-platformed on cloud.

Step 3: Split traffic between datacenter and cloud

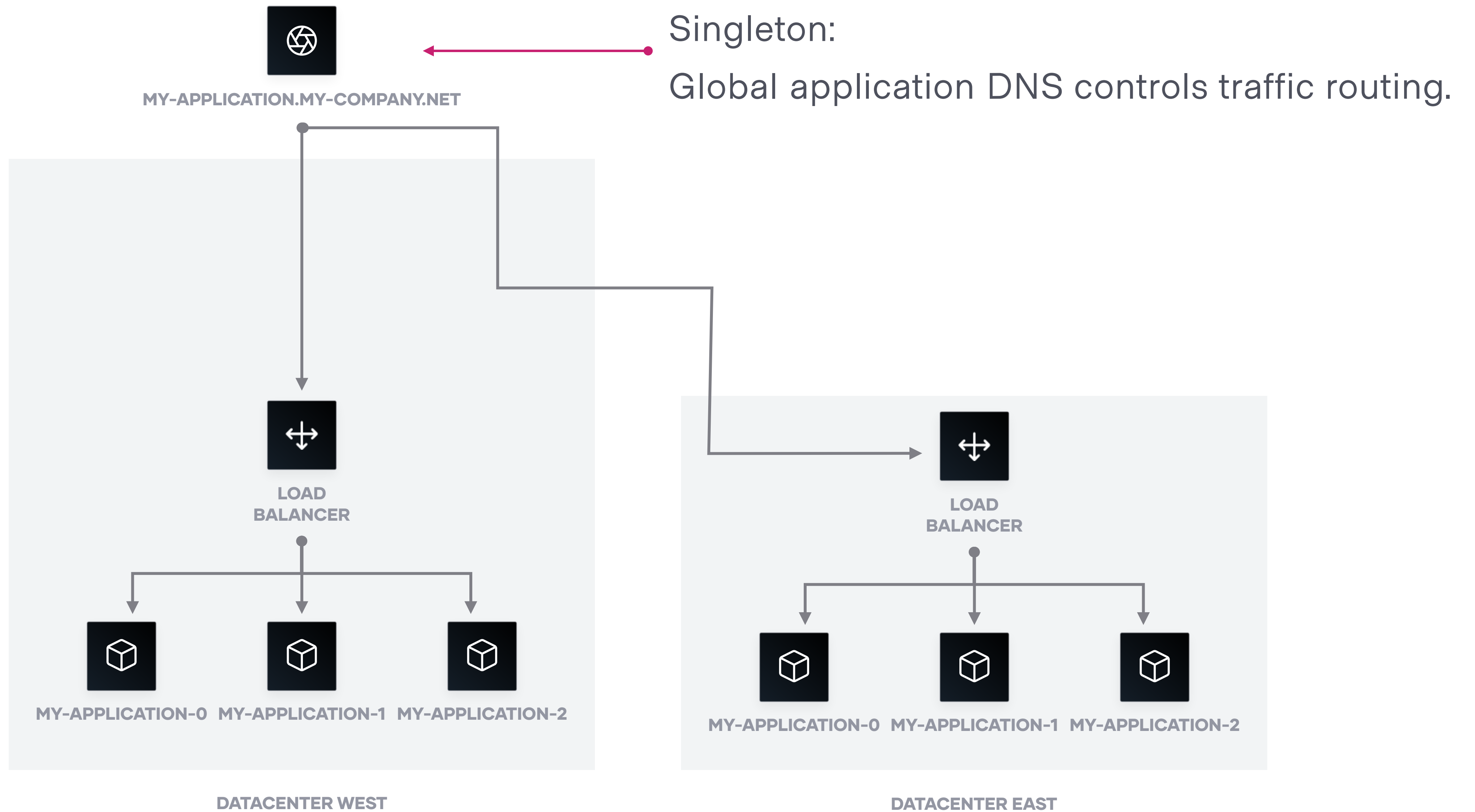


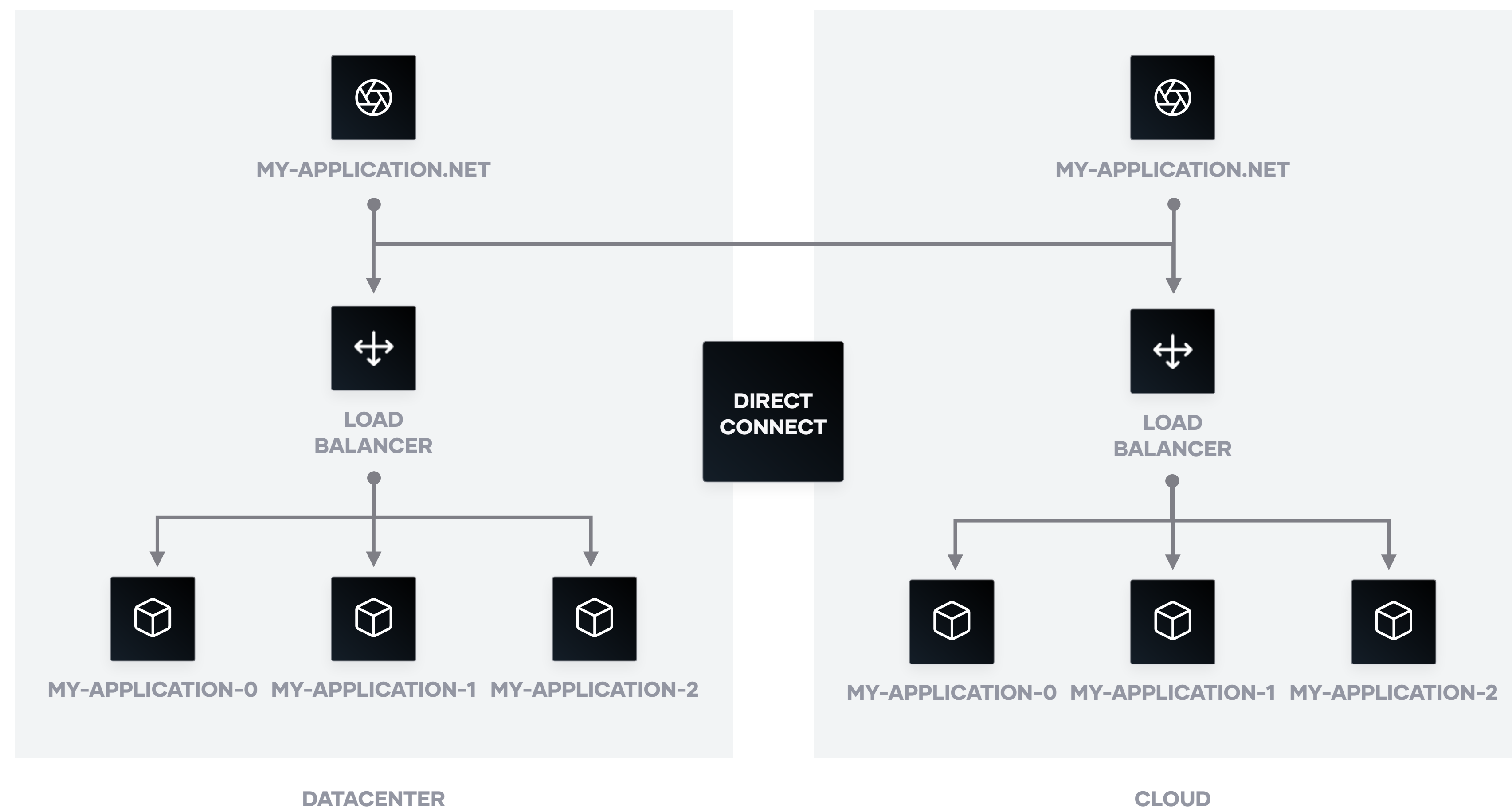
Step 1: Refactor application

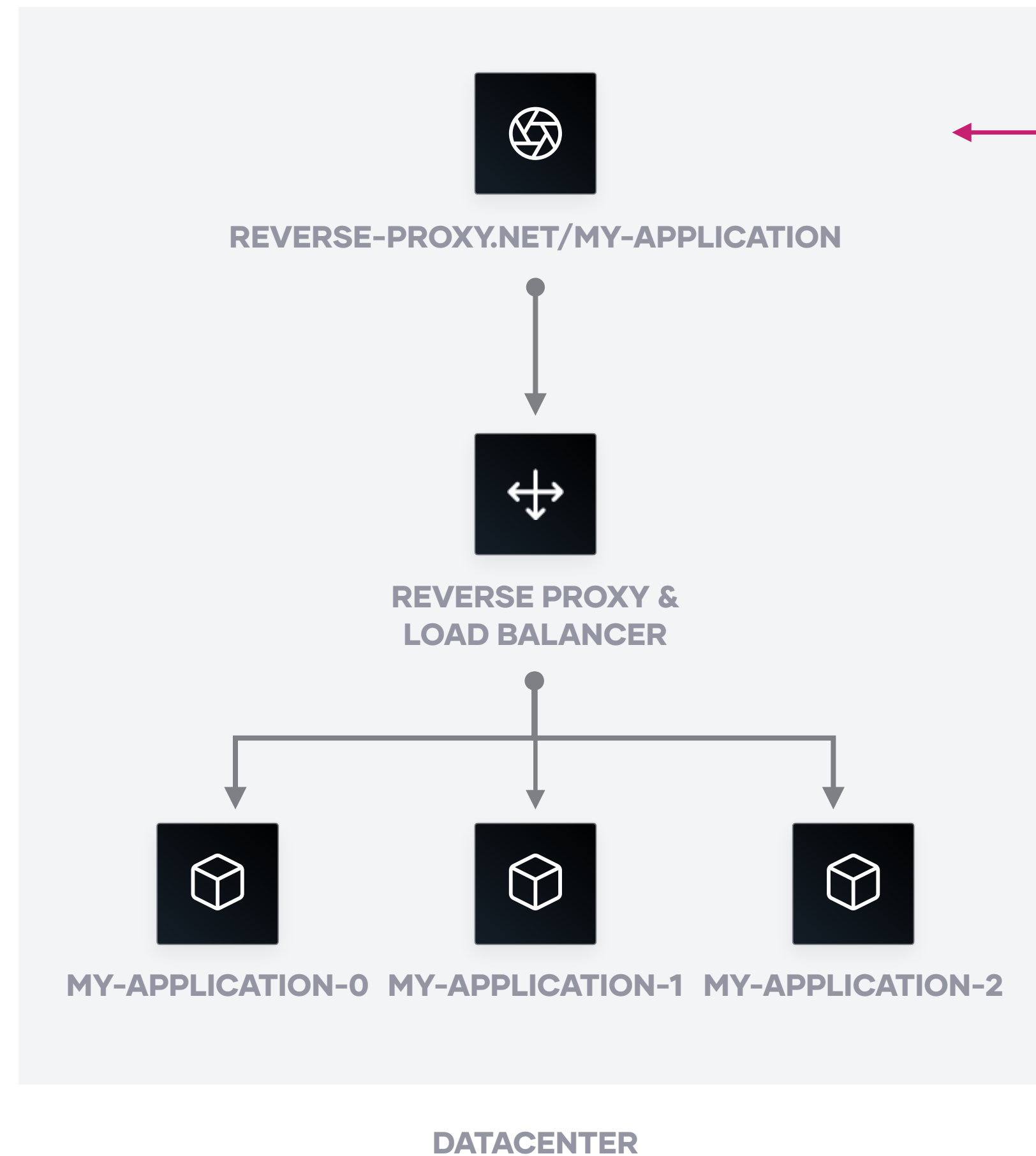
Step 2: Configure with infrastructure as code



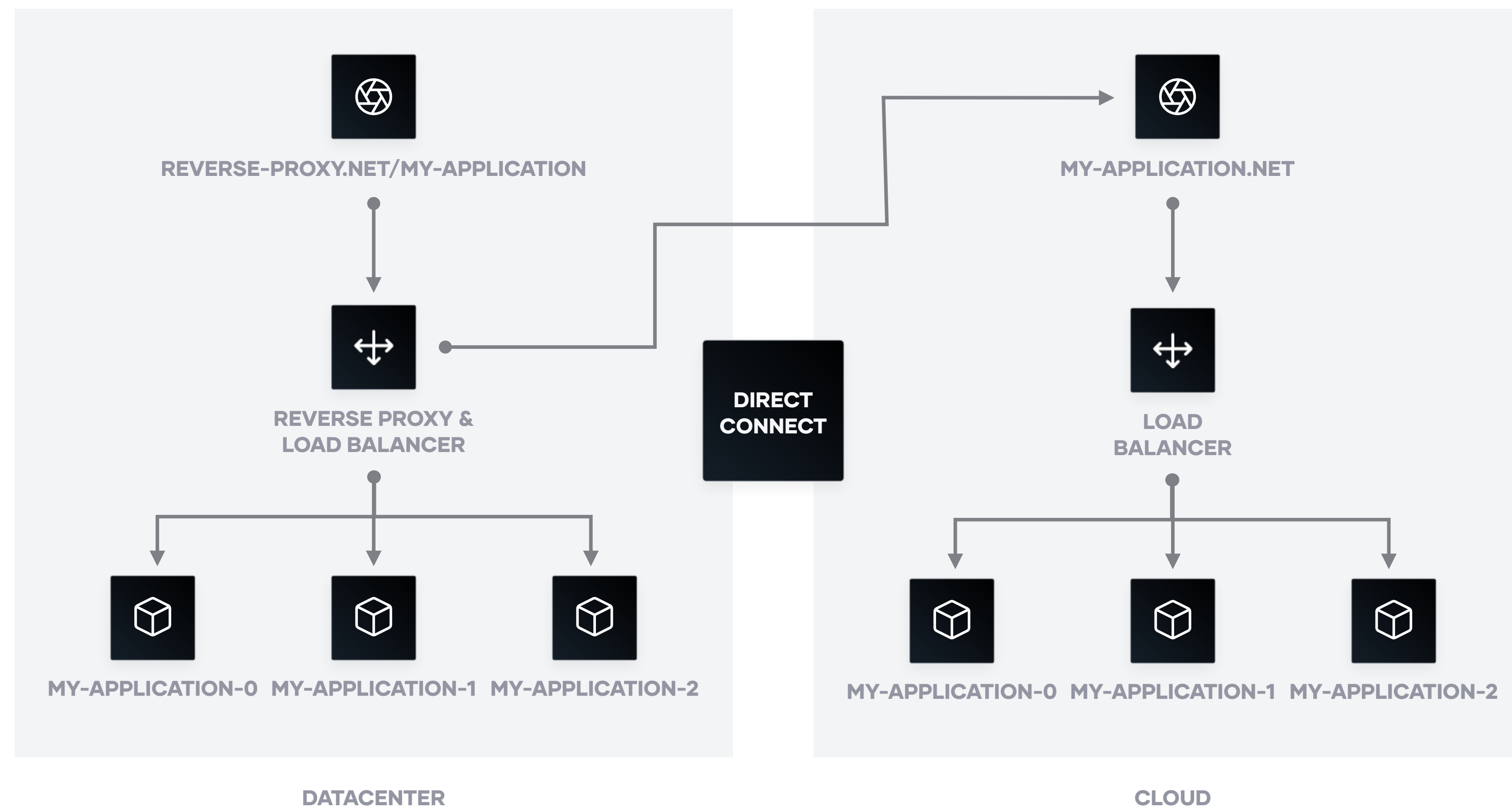
Patterns for Migration: Traffic Splitting

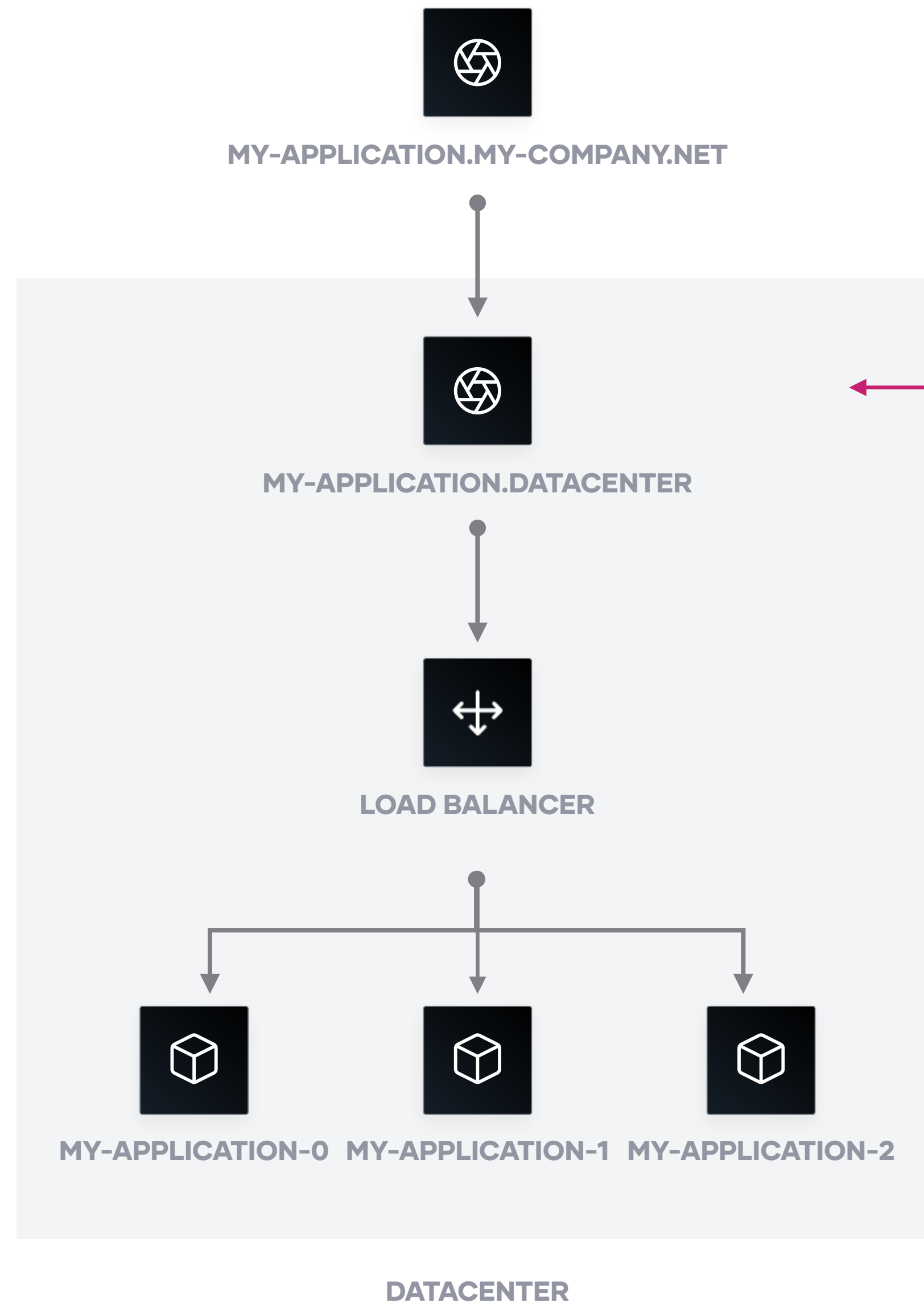




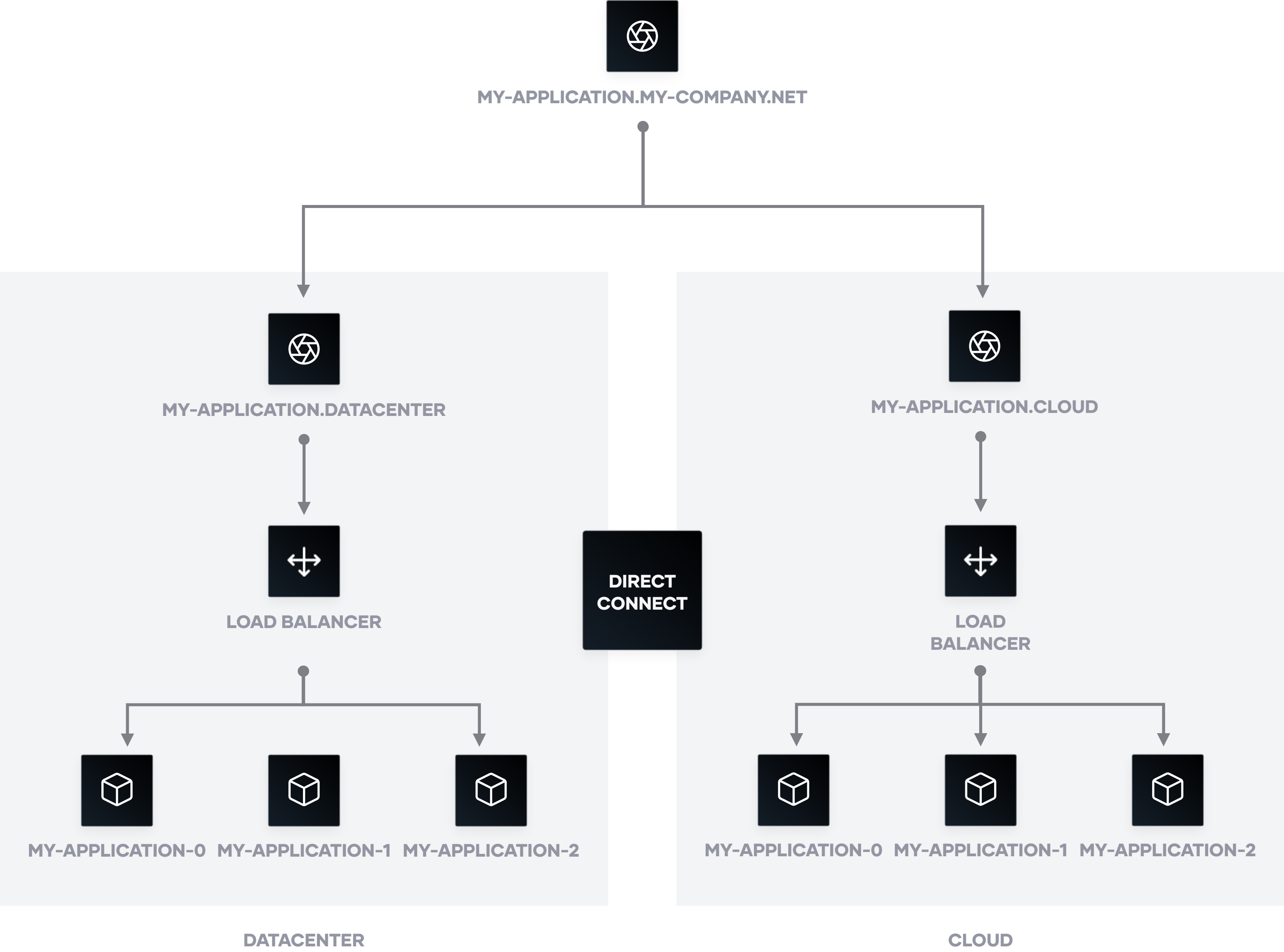


Reverse Proxy:
Based on path or header routing, transform request to the correct service.





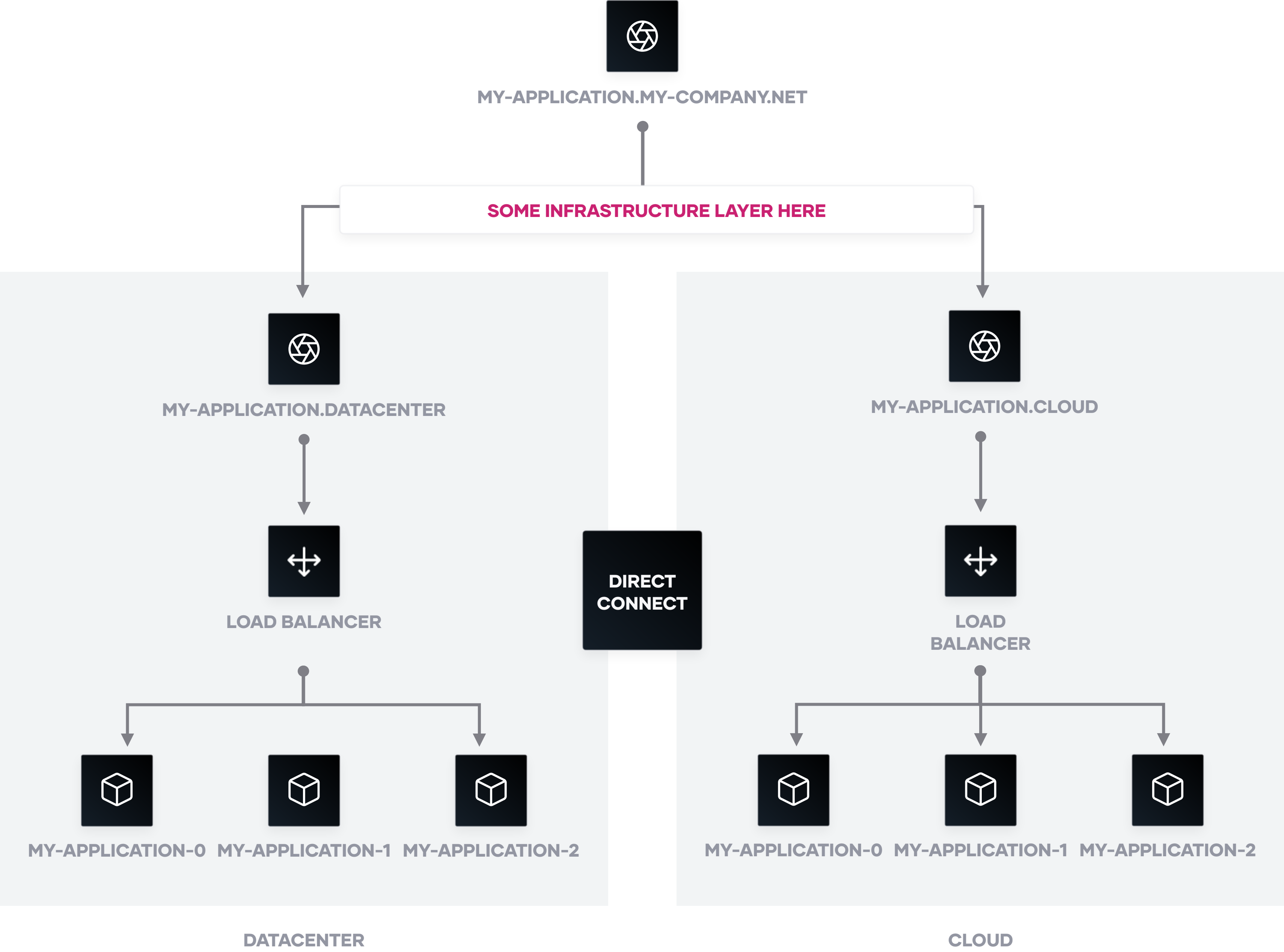
Composite:
Create a subdomain specific to datacenter or region.
Top-level domain references subdomains.





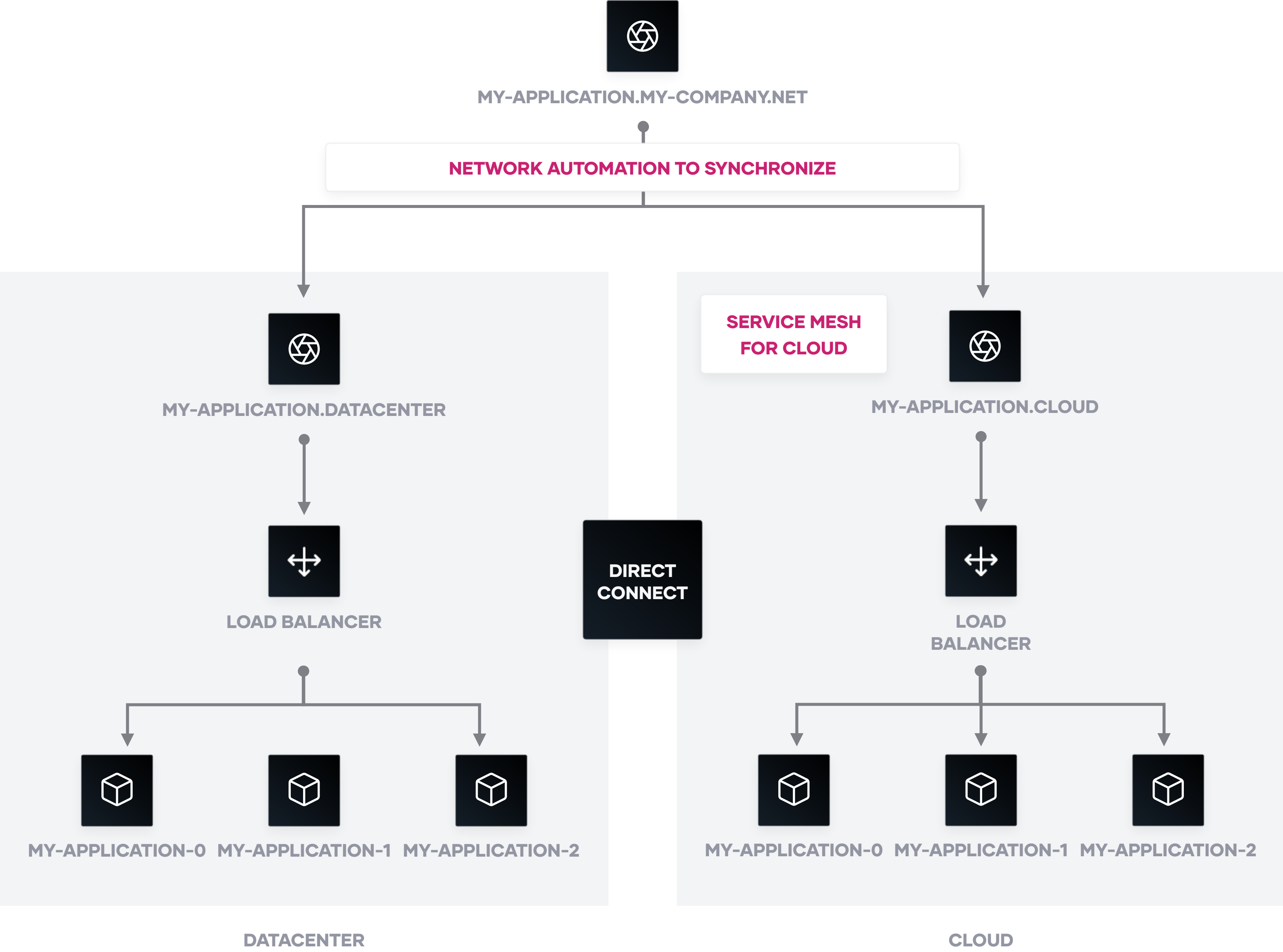
Supercharge!

(Scale & Evolve)

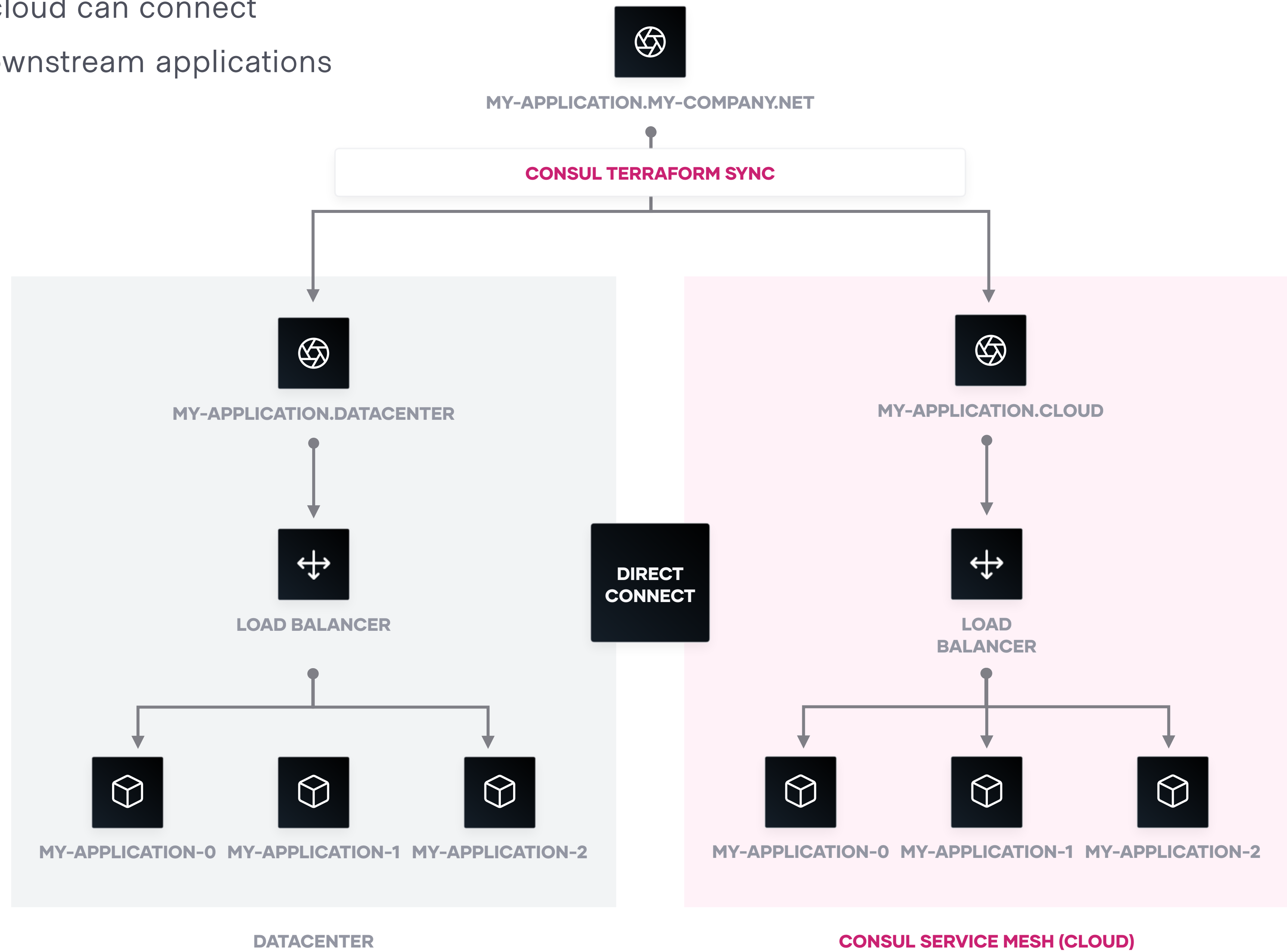


Service mesh

An infrastructure layer that facilitates communication between services.

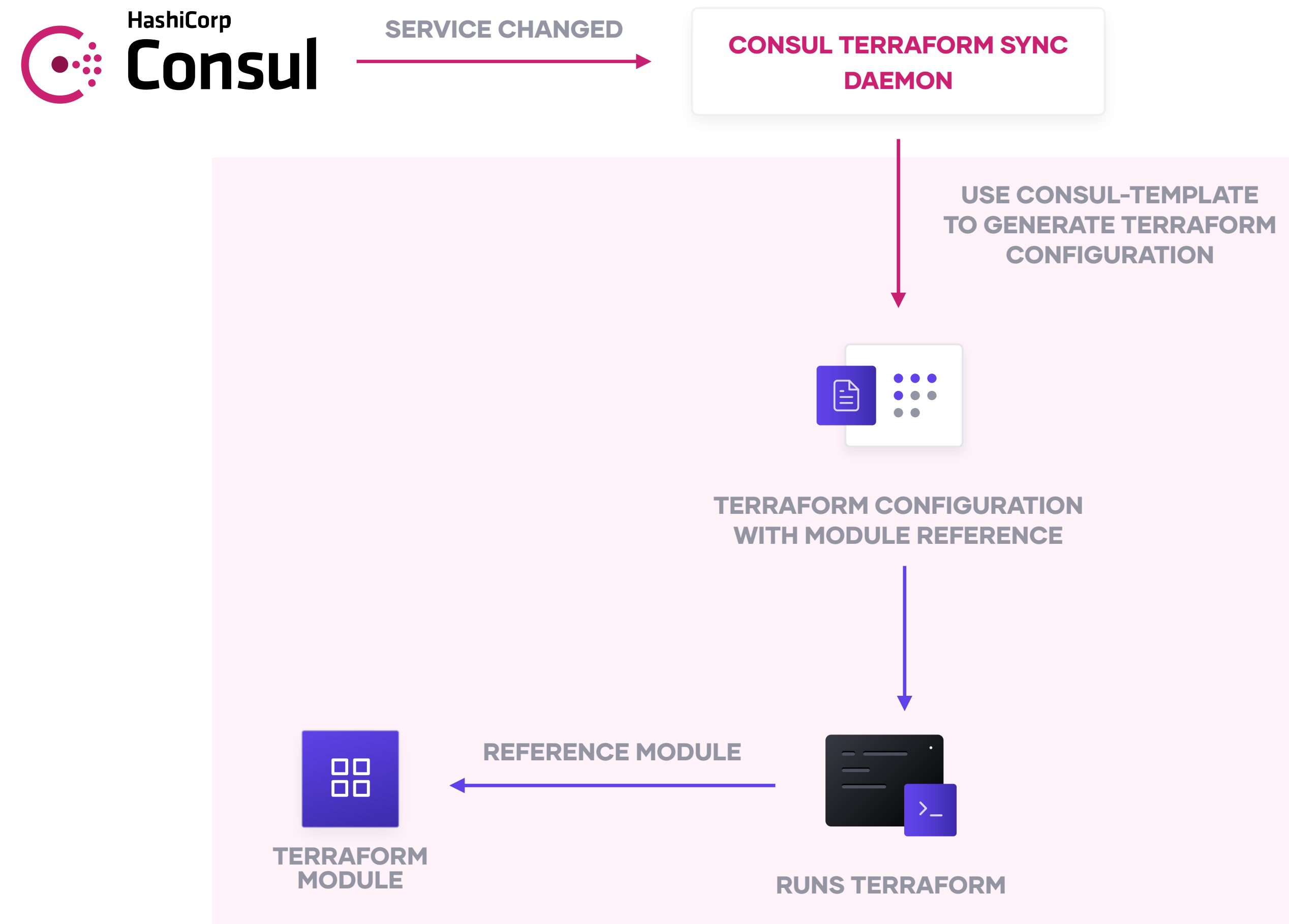


- 💡 If datacenter & cloud can connect
- ✅ No change to downstream applications



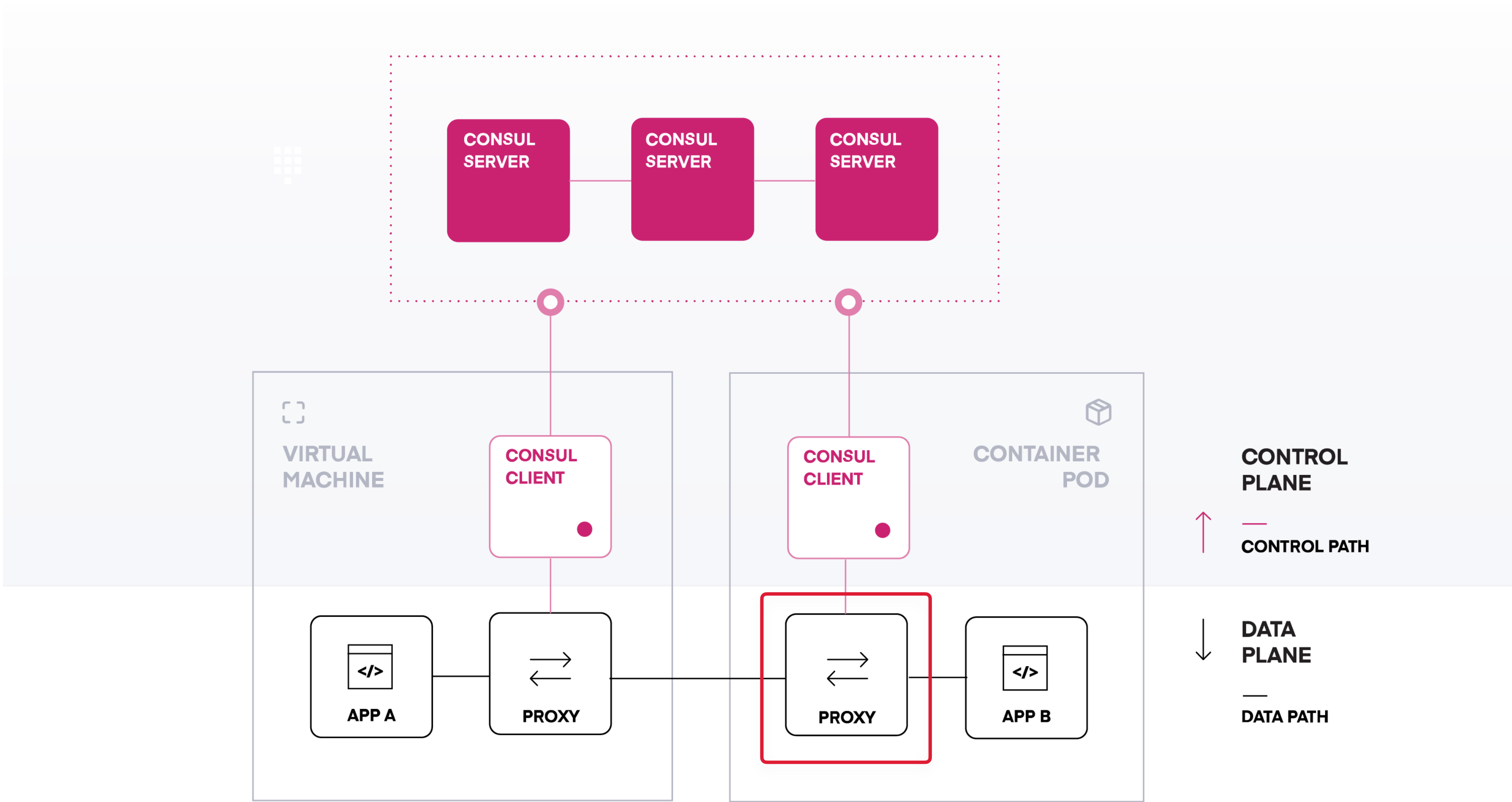
Consul Terraform Sync (CTS)

[github.com/hashicorp/
consul-terraform-sync](https://github.com/hashicorp/consul-terraform-sync)



Service Mesh

- Traffic Management
- Security
- Observability





Demo

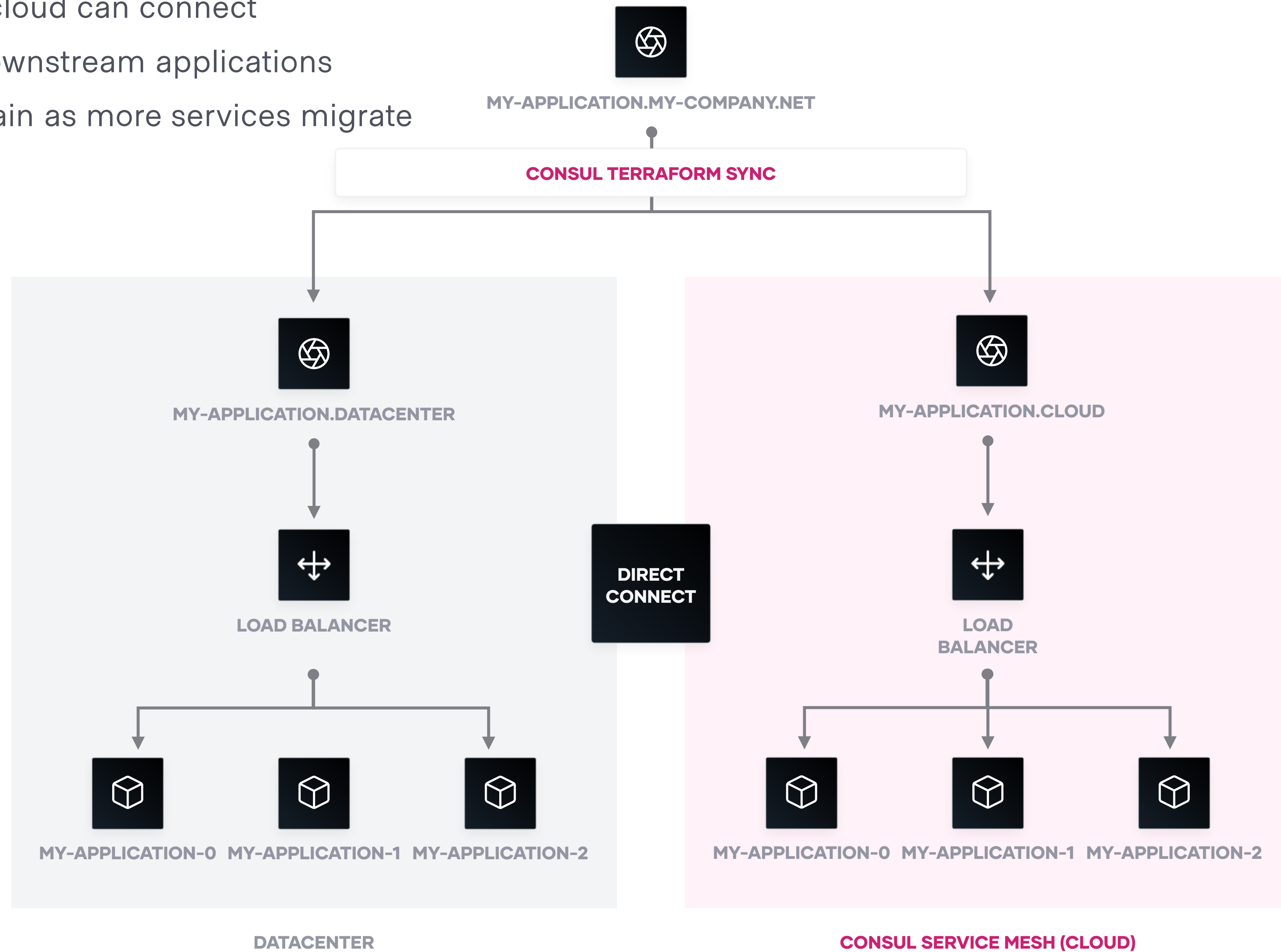
Network Automation + Service Mesh

Demo: github.com/joatmon08/cloud-migration

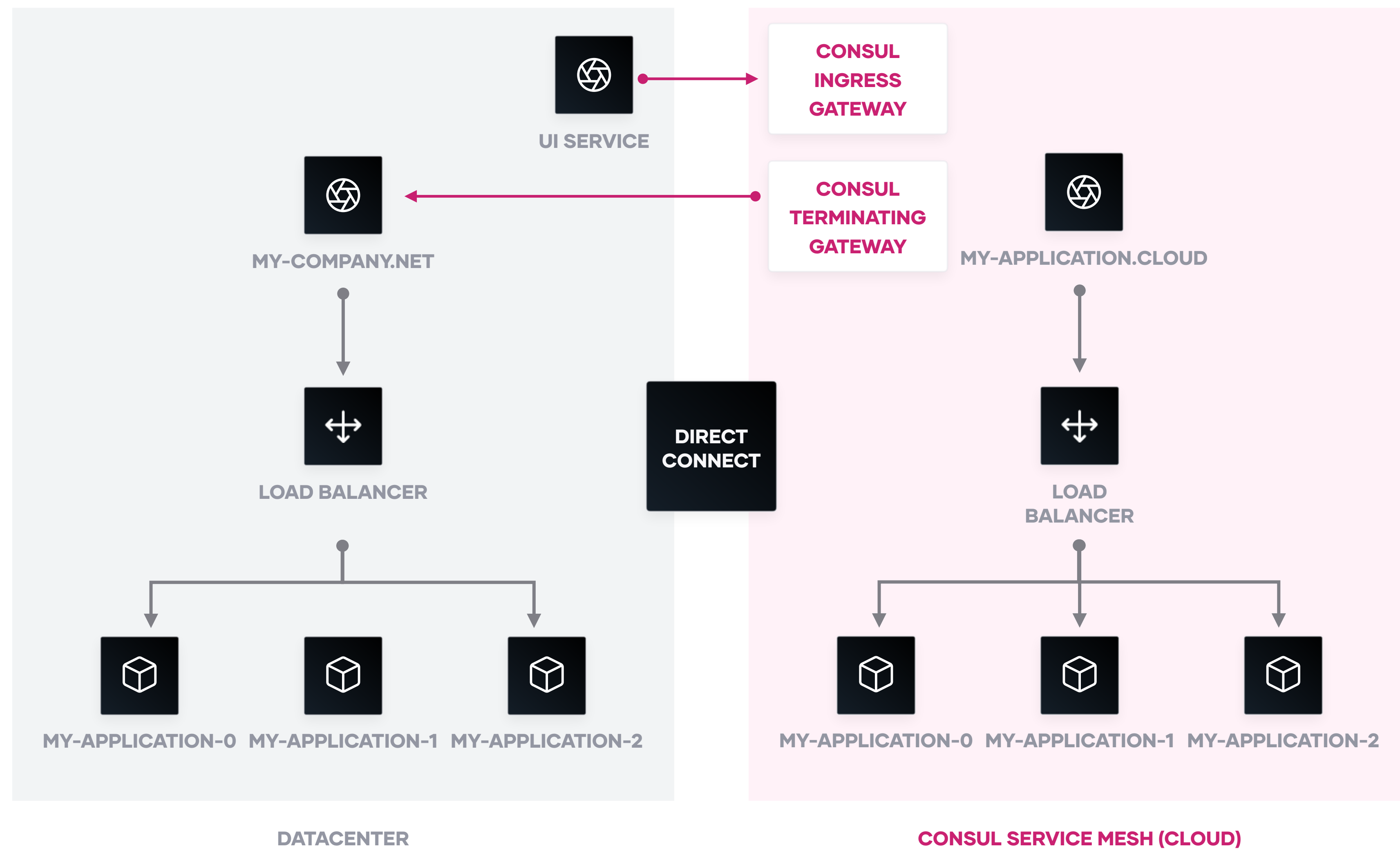
Terraform Module: github.com/joatmon08/terraform-aws-listener-rule

Note: “Datacenter” is AWS us-east-2. This can be done with datacenter load balancers as long as they can be configured with Terraform.

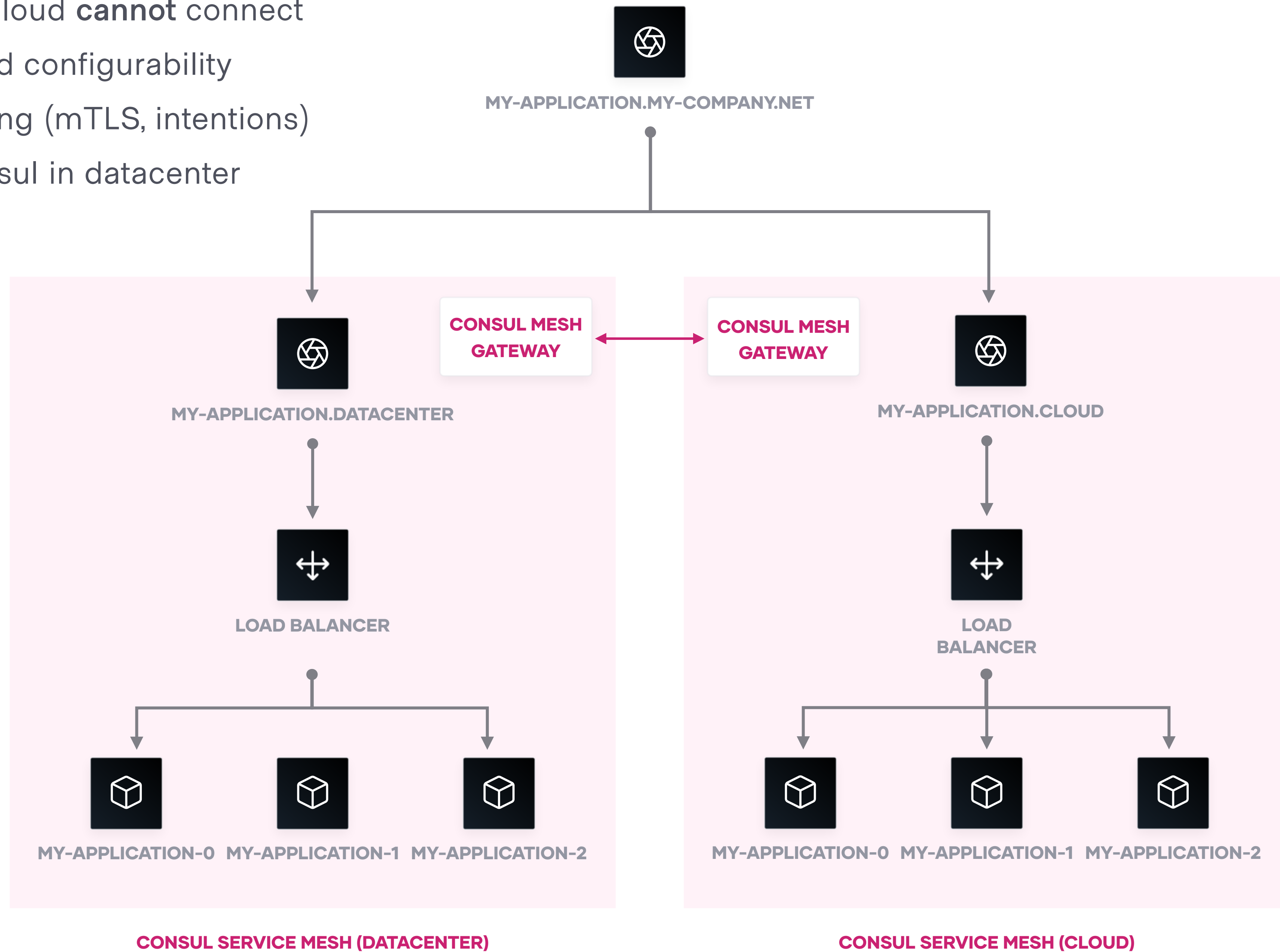
- 💡 If datacenter & cloud can connect
- ✅ No change to downstream applications
- ❗ Difficult to maintain as more services migrate



- 💡 If datacenter & cloud can connect
- ✅ Additional observability & configurability
- ❗ Refactor applications to use Consul for service discovery



- 💡 If datacenter & cloud **cannot** connect
- ✅ Observability and configurability
- ✅ Secure networking (mTLS, intentions)
- ❗ Must deploy Consul in datacenter



Want to learn more?



- [Datacenter Load Balancing] devcentral.f5.com/s/articles/Pushing-Updates-to-BIG-IP-w-Consul-Terraform-Sync
- [Consul Terraform Sync Integrations] hashicorp.com/blog/announcing-consul-terraform-sync-tech-preview
- [Consul Service Mesh] learn.hashicorp.com/collections/consul/developer-mesh
- [Slides & Code] joatmon08.github.io/03_speaking.html