☰   ⊡(/learn)   ⚙   📋

Variations

In this lesson, we'll look at some variations in the approach we've discussed so far.

We'll cover the following   ⌃

- Combination with frontend integration
- Combination with asynchronous communication
- Other load balancers
- Service meshes

Consul is very flexible and can be used in many different ways.

Combination with frontend integration

Like other approaches Consul can be combined with frontend integration (see chapter 3 (https://www.educative.io/collection/page/10370001/5441945024331776/6168726367895552)).

SSI (Server-side Includes) with Apache httpd is simple to combine since an Apache httpd is already present in the system.

Combination with asynchronous communication

Synchronous communication can be combined with asynchronous communication (see chapter 6 (https://www.educative.io/collection/page/10370001/5441945024331776/4516786718375936)). However, one type of communication should suffice normally.

Atom or other asynchronous approaches via HTTP (see chapter 8 (https://www.educative.io/collection/page/10370001/5441945024331776/5669712774037504)) are easy to integrate into an HTTP-based system like Consul.

Other load balancers

Instead of Apache httpd, a server like **nginx** or a load balancer like **HAProxy** can also be used for routing of requests from the outside.

Ribbon can be replaced by such a load balancer so that the internal load balancing is then configured with Consul Template.

In this case, only one type of load balancer is used for load balancing and routing. Unlike the Java library Ribbon, Apache httpd or nginx can be used with any programming language.

Each microservice then has its own httpd or nginx instance so that no bottleneck or single point of failure is created.

Service meshes

⚙️ 📋

> **Service meshes** provide a lot of useful features for resilience, monitoring, tracing, and logging. Istio is an example of a service mesh.

A service mesh injects proxies into the communication between the microservices. Istio supports Consul to achieve that.

With Istio, Consul can be extended to become a **complete platform for the operation** of microservices.

Q U I Z

---

1 Why is it easy to combine Consul with frontend?

○ **A)** Both are synchronous.

○ **B)** Both use a combination of communication.

○ **C)** Apache httpd is already present in the system.

Submit Answer

⚙️   📋

‹   Question 1 of 2
0 attempted   ›

Reset Quiz ↻

In the next lesson, we'll look at some experiments that can be conducted with the approach we've learned in this chapter.

← **Back**

Next →

DNS and Registrator

Experiments

✅ Mark as Completed

⚠️ Report an Issue