



Experiments

In this lesson, we'll look at some experiments that you can try out with SSI using ESI.

We'll cover the following ^

- Experiments
- Try it yourself!

Experiments

Here are some experiments you can do with all that you've learned in this chapter.

Supplement the system with an additional microservice.

Try it yourself!

Try all the experiments in the environment below.



```

version: '3'
services:
  common:
    image: educative1/mapi_scsesi_common
  order:
    image: educative1/mapi_scsesi_order
    ports:
      - "8090:8080"
  varnish:
    image: educative1/mapi_scsesi_varnish
    links:
      - common
      - order
    ports:
      - "8080:8080"

```

- A microservice that simply displays a static HTML page can serve as an example.
- The new microservice should integrate header, navigation bar, and footer of the common microservice via ESI tags.
- Package the microservice as a Docker image and reference it in `docker-compose.yml`. There you can also determine the name of the Docker container.
- The microservice has to be accessible via Varnish. To achieve that, you have to integrate a new backend with the name of the Docker container in `default.vcl` and adapt the routing in `vcl_recv()`.
- Now you should be able to access the new microservice at `http://localhost:8080/name` (`http://localhost:8080/name`) if the Docker containers run on the local computer and the new service is configured in Varnish's routing with `name`. The ESI tags should have been replaced by HTML code.
- The Go application returns only a dynamic HTML fragment – the navigation bar with the current time. Instead of the Go application, a web server can also deliver static pages. For example, replace the Go



fragments and the Bootstrap library. The time does not necessarily have to be displayed in the navigation bar, so a static HTML fragment is all that is needed.

- Change the caching so that the pages are immediately invalidated when new orders are received. You can, for example, use matching HTTP headers as explained in a chapter (<http://book.varnish-software.com/4.0/chapters/HTTP.html>) of the Varnish book. An alternative is to remove objects directly from the cache. The Varnish book also contains a chapter (http://book.varnish-software.com/4.0/chapters/Cache_Invalidation.html) on this option.
- Replace the Varnish cache in the example with Squid (<http://www.squid-cache.org/>), which also implements ESI.
- Replace ESI with SSI and replace the Varnish cache with an Apache httpd or nginx.
- What happens if the web servers fail? Simulate the failure with `docker-compose up --scale common=0` or `docker-compose up --scale order=0`. Which parts of the web page still work? Is it still possible to place orders, for example?

In the next lesson, we'll look at a quick chapter conclusion.

[← Back](#)[Variations](#)[Next →](#)[Introduction](#)[Mark as Completed](#)[Report an Issue](#)

