

Step 1

set up a git repository on the desktop
make a new folder called config
copy everything from files/config to the folder
git init
git add --all
git commit -m "Adding files"

Create a new module called config-server
com.frankmoley.talks.cloud

Selections:

Cloud Config/Config Server

Ops/Actuator

Annotate config with @EnableConfigServer

Modify application.properties

server.port=9000

spring.cloud.config.server.git.uri=file:\${user.home}/Desktop/config/

management.security.enabled=false

Start server

navigate to <http://localhost:9000/userservice/master>

Step 2

Create a new module called discovery-server
com.frankmoley.talks.cloud

Selections:

Cloud Discovery/Eureka Server

Ops/Actuator

Annotate config with @EnableEurekaServer

Modify application.properties

server.port=8761

eureka.client.register-with-eureka=false

eureka.client.fetch-registry=false

Start server

navigate to <http://localhost:8761/>

Step 3

Create a new module called person-service
com.frankmoley.talks.cloud

Selections:

Web/Web

SQL/JPA & H2

Cloud Config/Config Client

Cloud Discover/Eureka Discovery

Ops/Actuator

Rename application.properties to bootstrap.properties

spring.application.name=person-service

spring.cloud.config.uri=<http://localhost:9000>

Annotate config

@EnableEurekaClient

@EnableConfigurationProperties

Copy ./files/person/data.sql and ./files/person/schema.sql to /src/main/resources
Create entity: USER_ID, FIRST_NAME, LAST_NAME, EMAIL_ADDRESS
Create CrudRepository
Create PersonController (/people)
 inject PersonRepository into the constructor
 implement GetMapping
 Add Logger

Step 4

Create a new module called business-service
com.frankmoley.talks.cloud

Sections:

- Web/Web
- Cloud Config/Config Client
- Cloud Discover/Eureka Discovery
- Cloud Routing/Feign
- Cloud Circuit Breaker/Hystrix & Hystrix Dashboard
- Ops/Actuator

Rename application.properties to bootstrap.properties
spring.application.name=business-service
spring.cloud.config.uri=<http://localhost:9000>

Code

- Copy Person from other class, remove entity
- Extract controller interface to PersonServiceClient
- Move the Client over, add /people
- Create an implementation called PersonServiceFallbackImpl
 - Annotate with @Component
 - Implement method to return me
- Return to PersonServiceClient
 - Annotate with

@FeignClient(value="person-service", fallback = PersonServiceFallbackImpl.class)

- Implement root controller, call the service (inject through constructor)

Application Config

- @EnableDiscoveryClient
- @EnableFeignClients
- @EnableCircuitBreaker
- @EnableHystrix

Start it up -> navigate to localhost:9500

- Stop PersonService
- Navigate again and show the changes

Now add HystrixDashboard

Pom.xml

- org.springframework:spring-cloud-starter-hystrix-dashboard

ApplicationClass

- @EnableHystrixDashboard

Start person-service and then restart business service

- Navigate to localhost:9500/hystrix
- localhost:9500/hystrix.stream

start making calls to localhost:9500, show dashboard again

Kill person-service, make calls to localhost:9500, show dashboard again

start person-service and start making calls again, show dashboard

