

***Microservice
Coffee Service Example***

*MSA **Coffee** Project*

Member

Order

Status

MSA Coffee Project

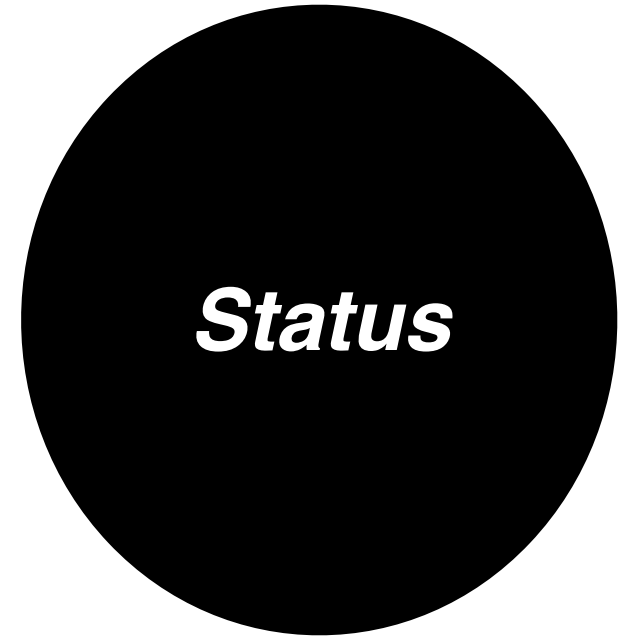
난 멤버관리



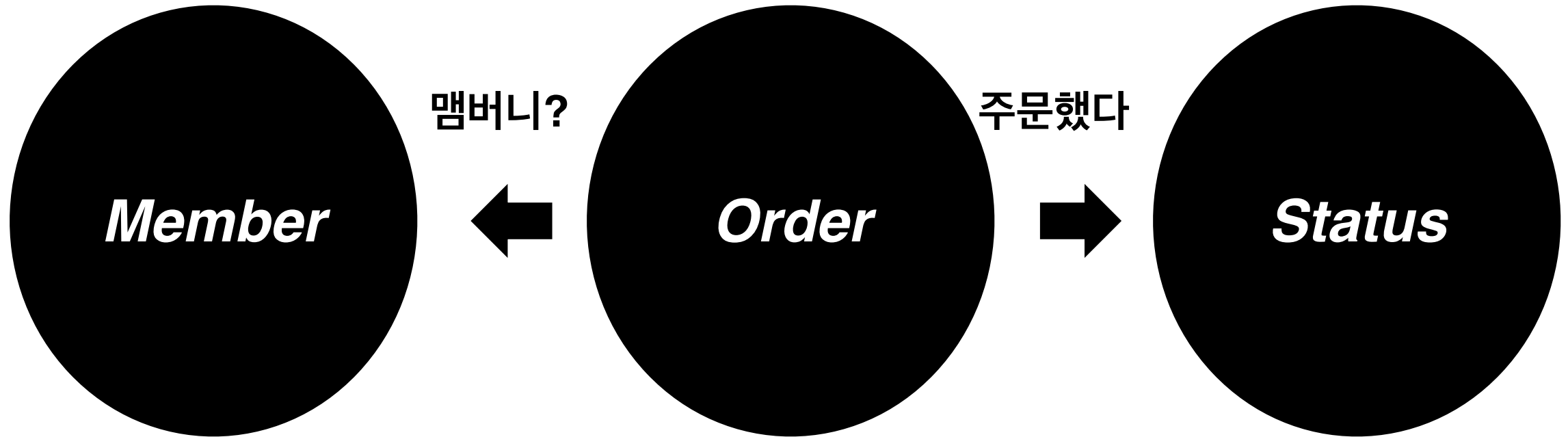
주문만 할건데..



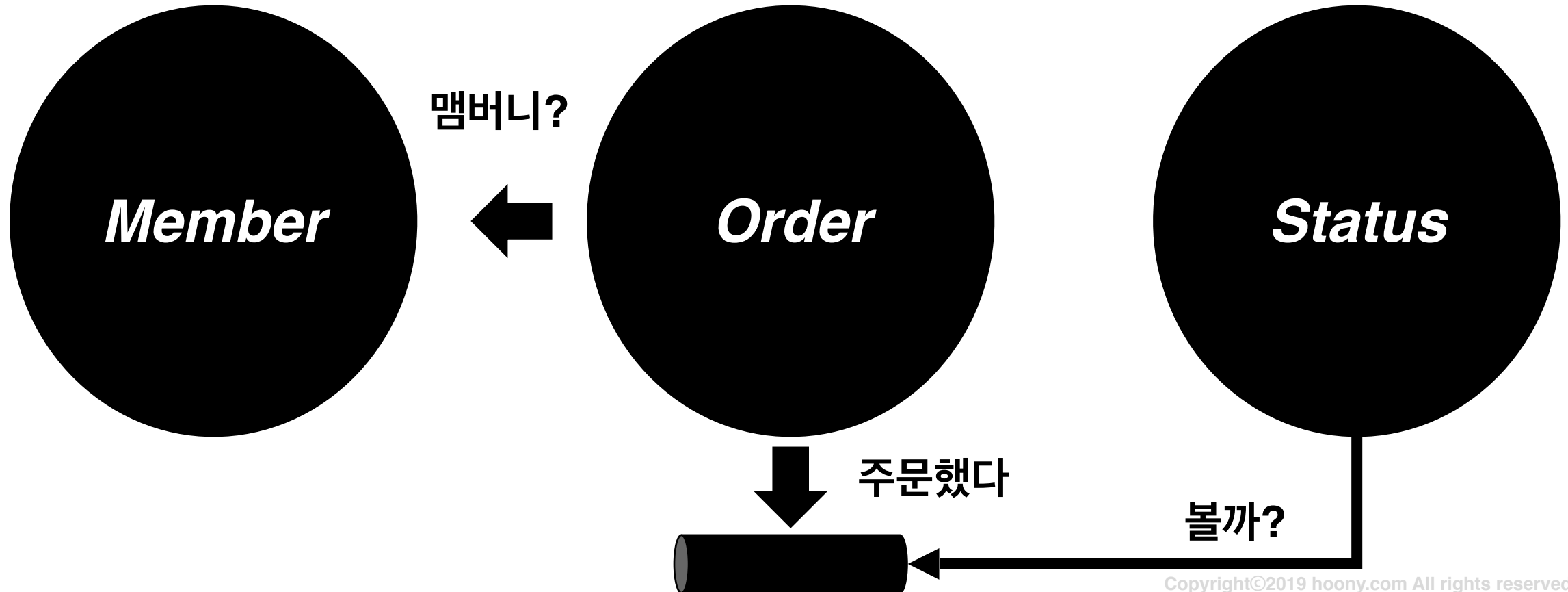
난 상태만 보여줄거야..



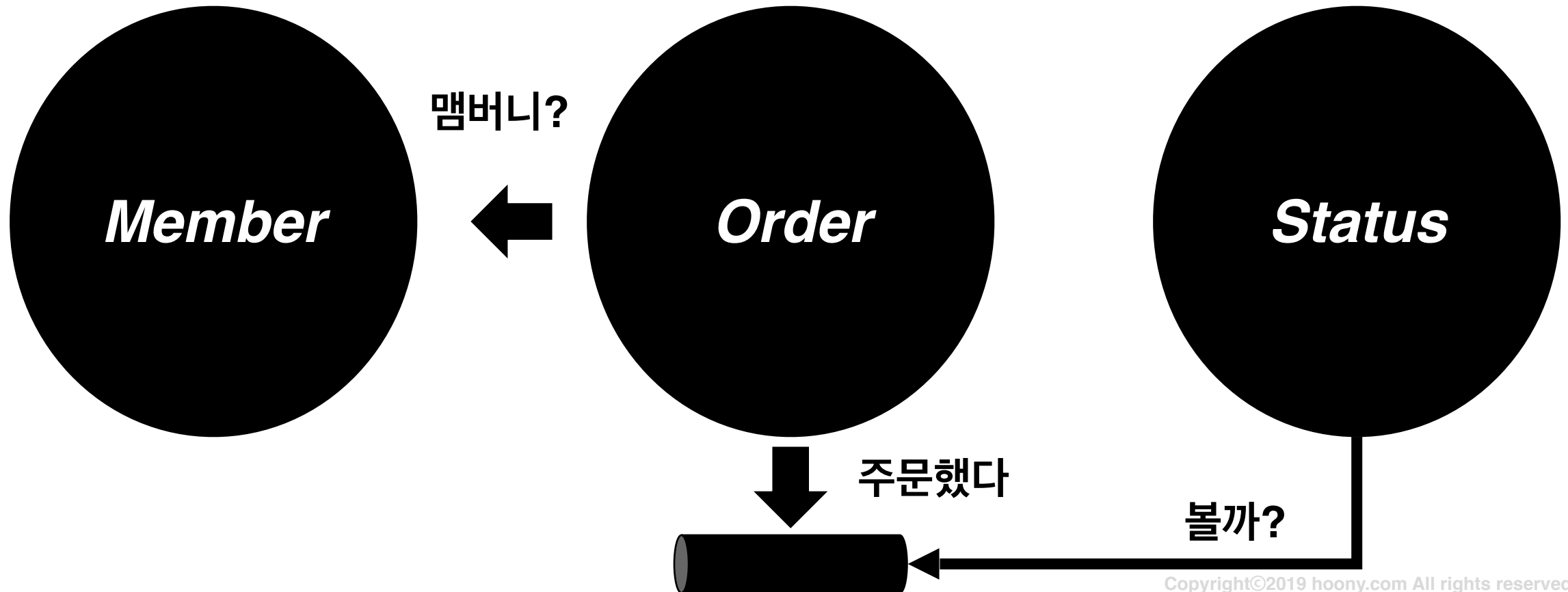
MSA Coffee Project



MSA Coffee Project



MSA Coffee Project



Microservices

```
graph TD; MS[Microservices] --- M1(( )); MS --- M2(( )); MS --- M3(( )); MS --- M4(( )); M1 --- Member((Member Sub modules)); M2 --- O1(( )); M3 --- Order((Order Sub modules)); M4 --- S1(( )); M4 --- S2(( )); S1 --- Status((Status Sub modules)); S2 --- S3(( ))
```

Member
Sub modules

Order
Sub modules

Status
Sub modules

microservices

멤버관리
마이크로서비스

msa-service-coffee-member

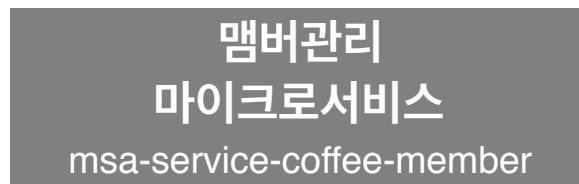
커피주문
마이크로서비스

msa-service-coffee-order

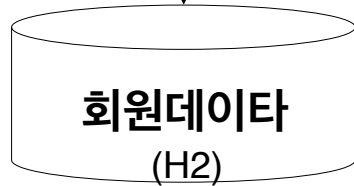
주문상태확인
마이크로서비스

msa-service-coffee-status

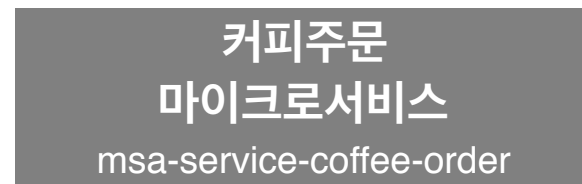
microservices



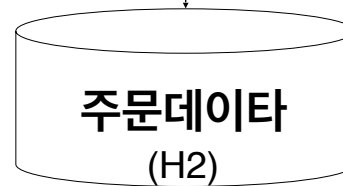
8081



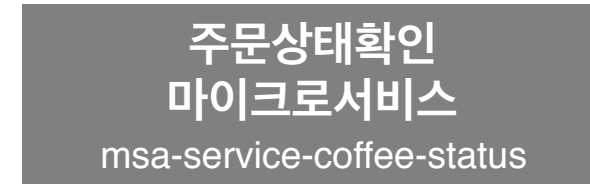
_id, userName



8082



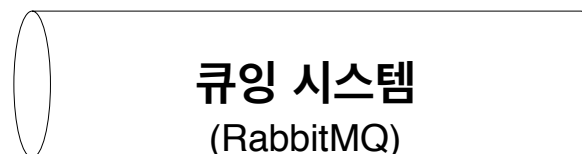
_id, orderName, userId



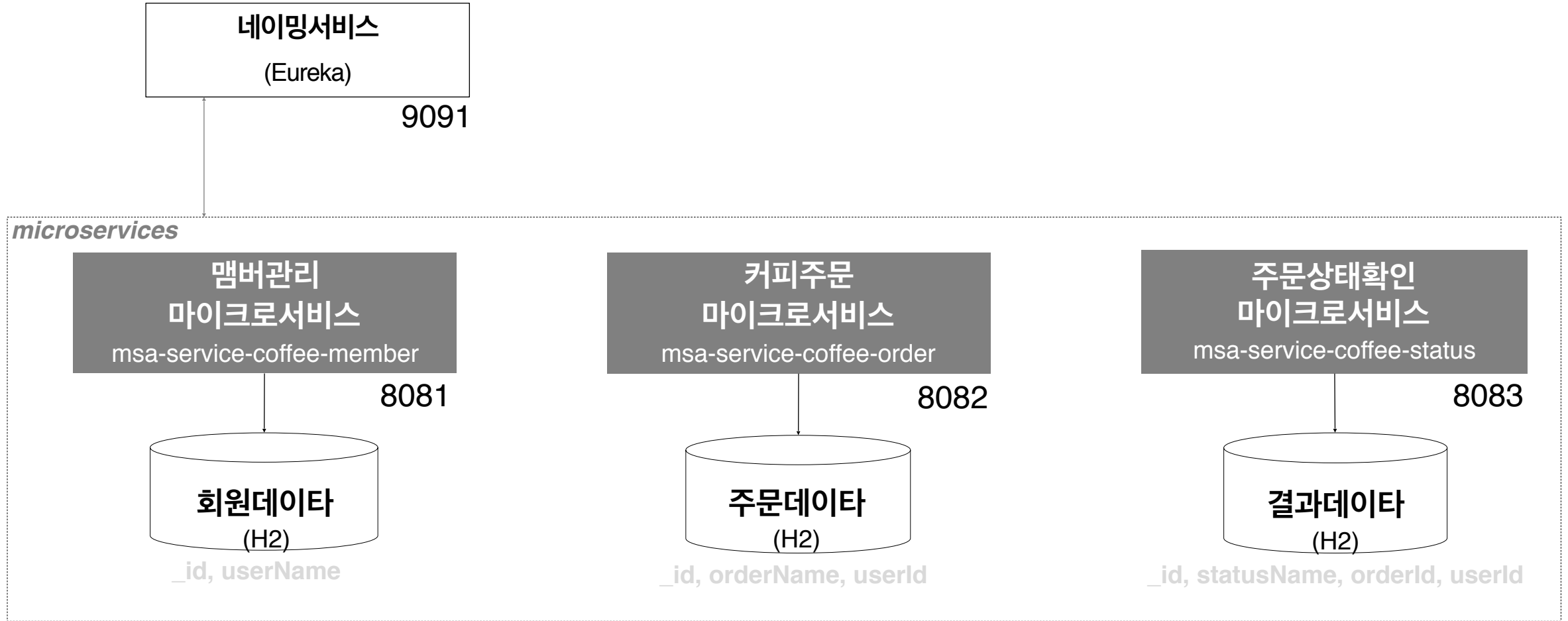
8083



_id, statusName, orderId, userId

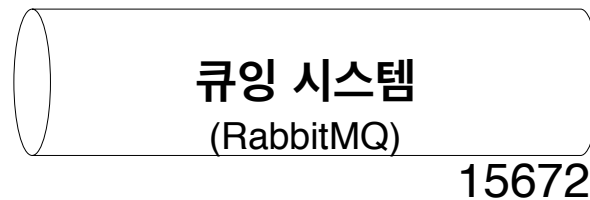


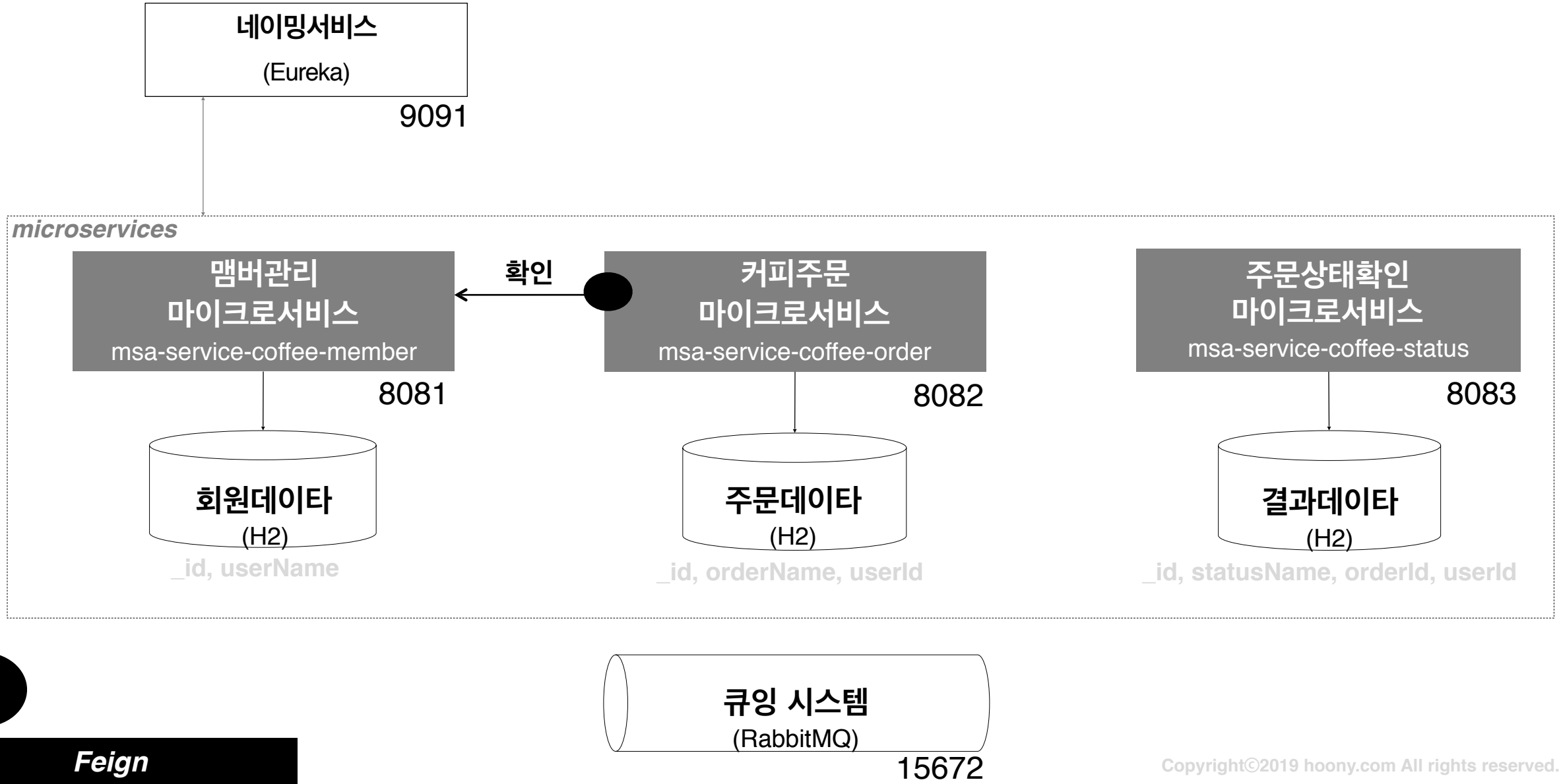
15672

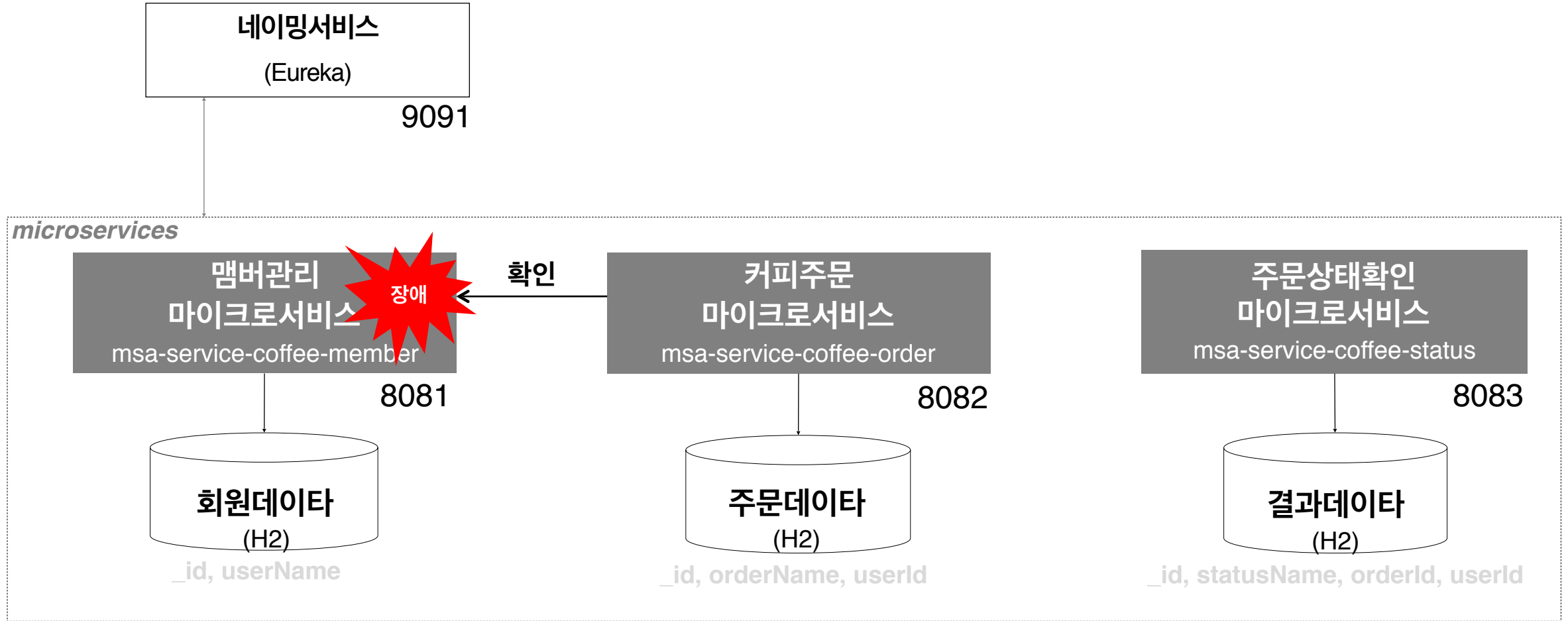


1

Eureka

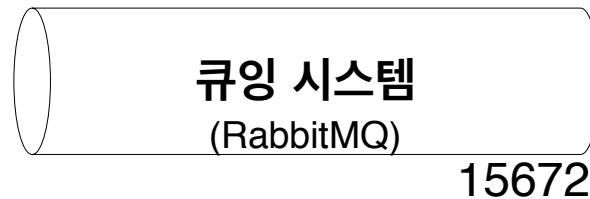


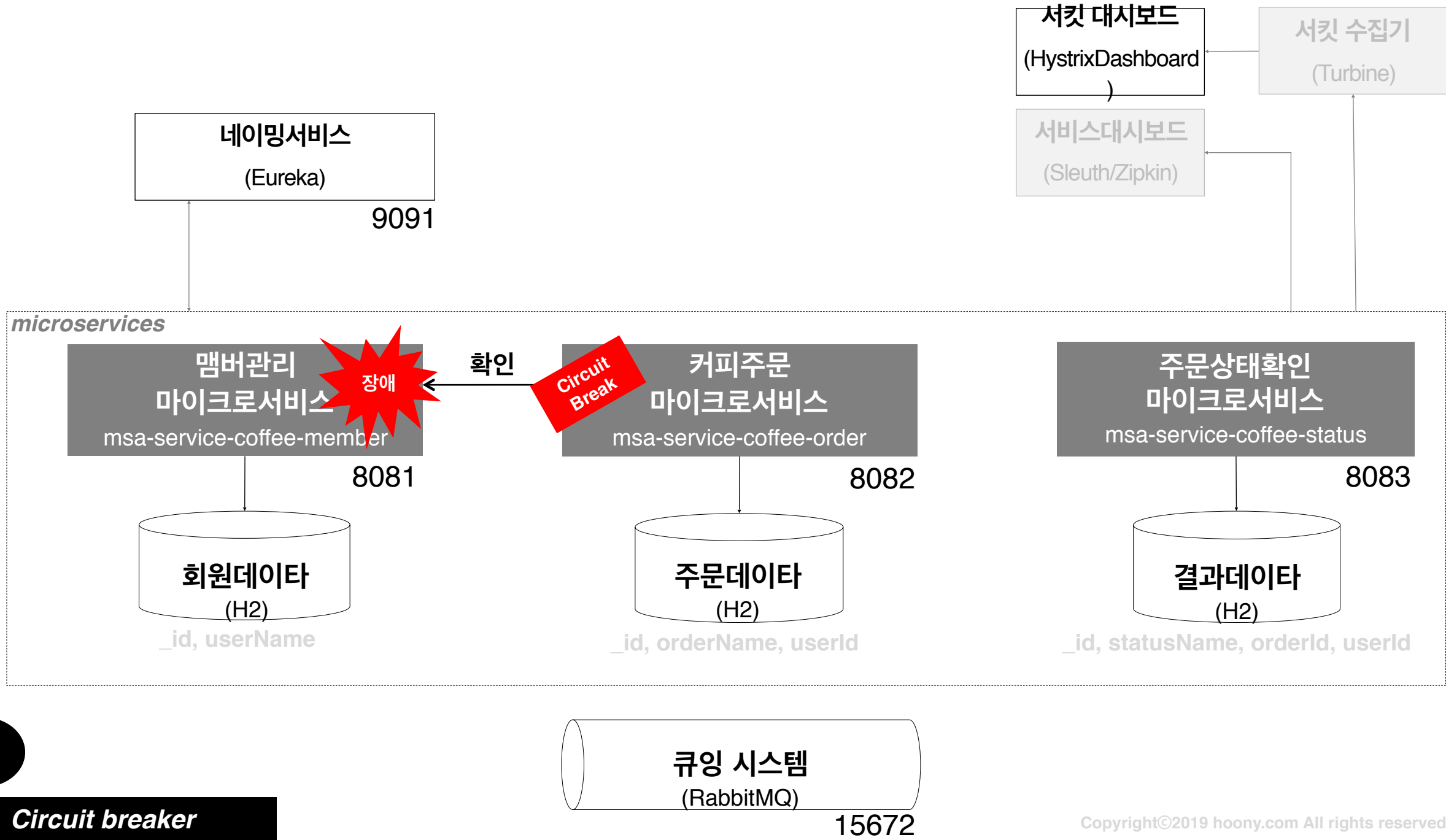


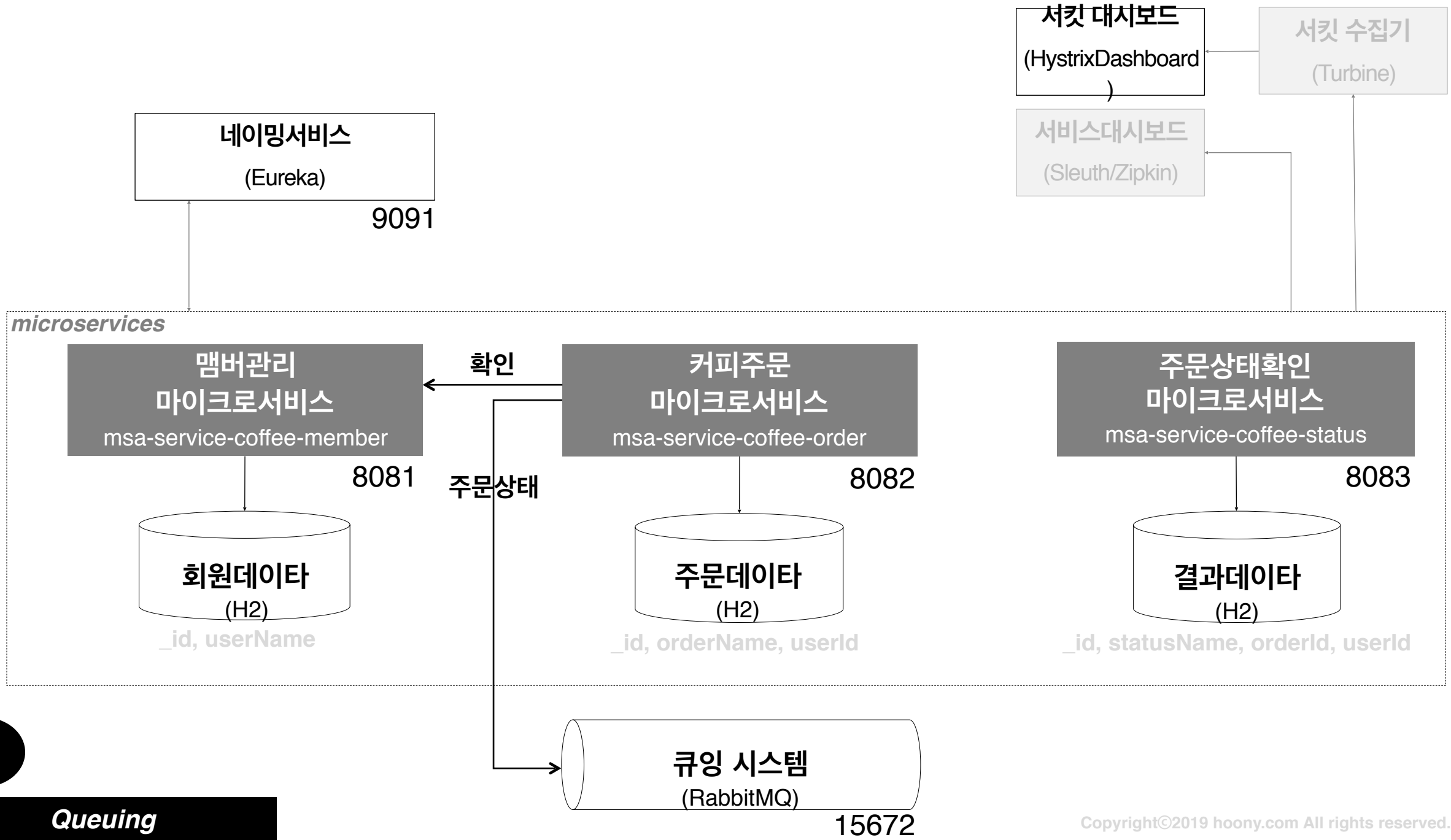


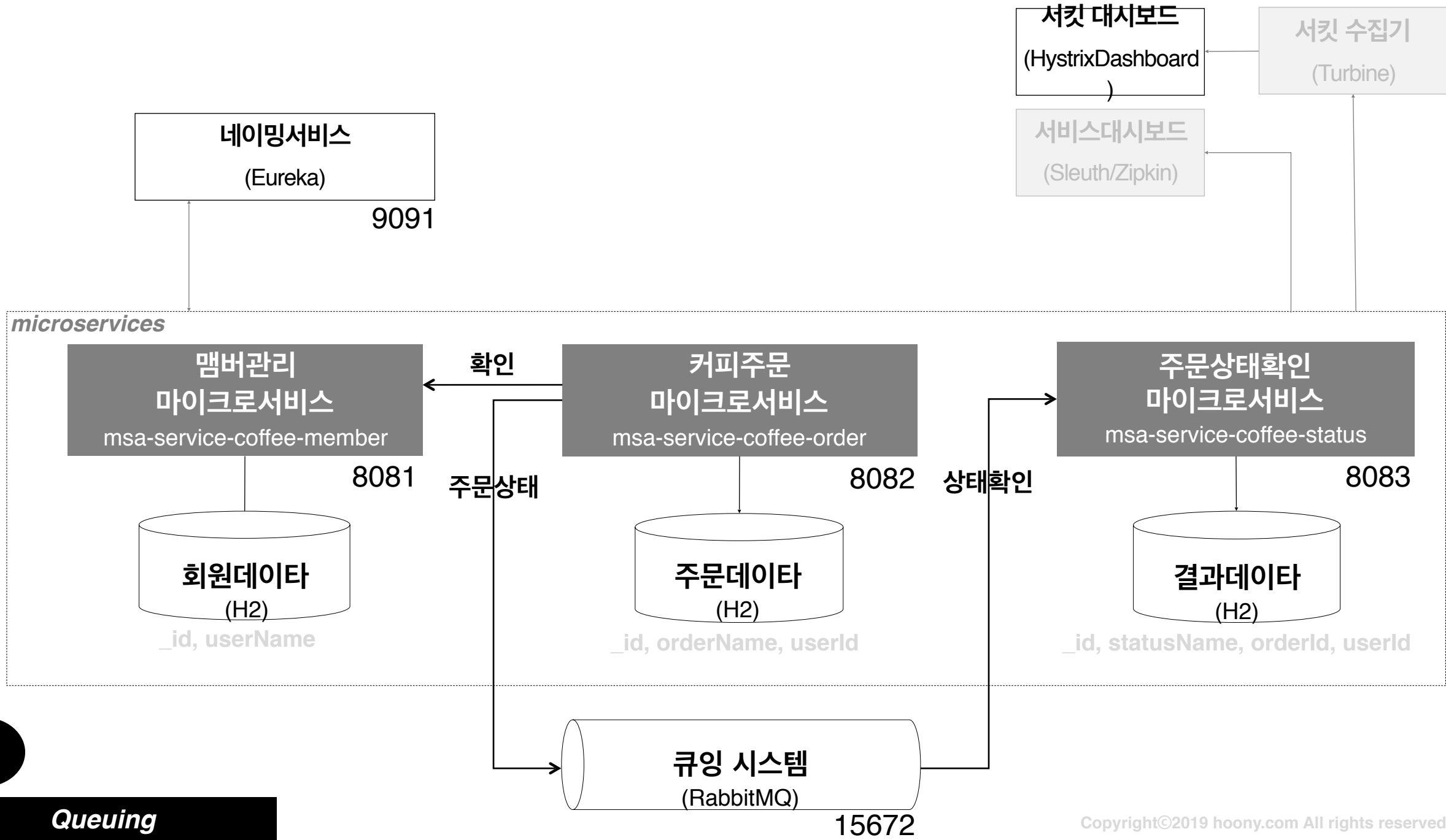
3

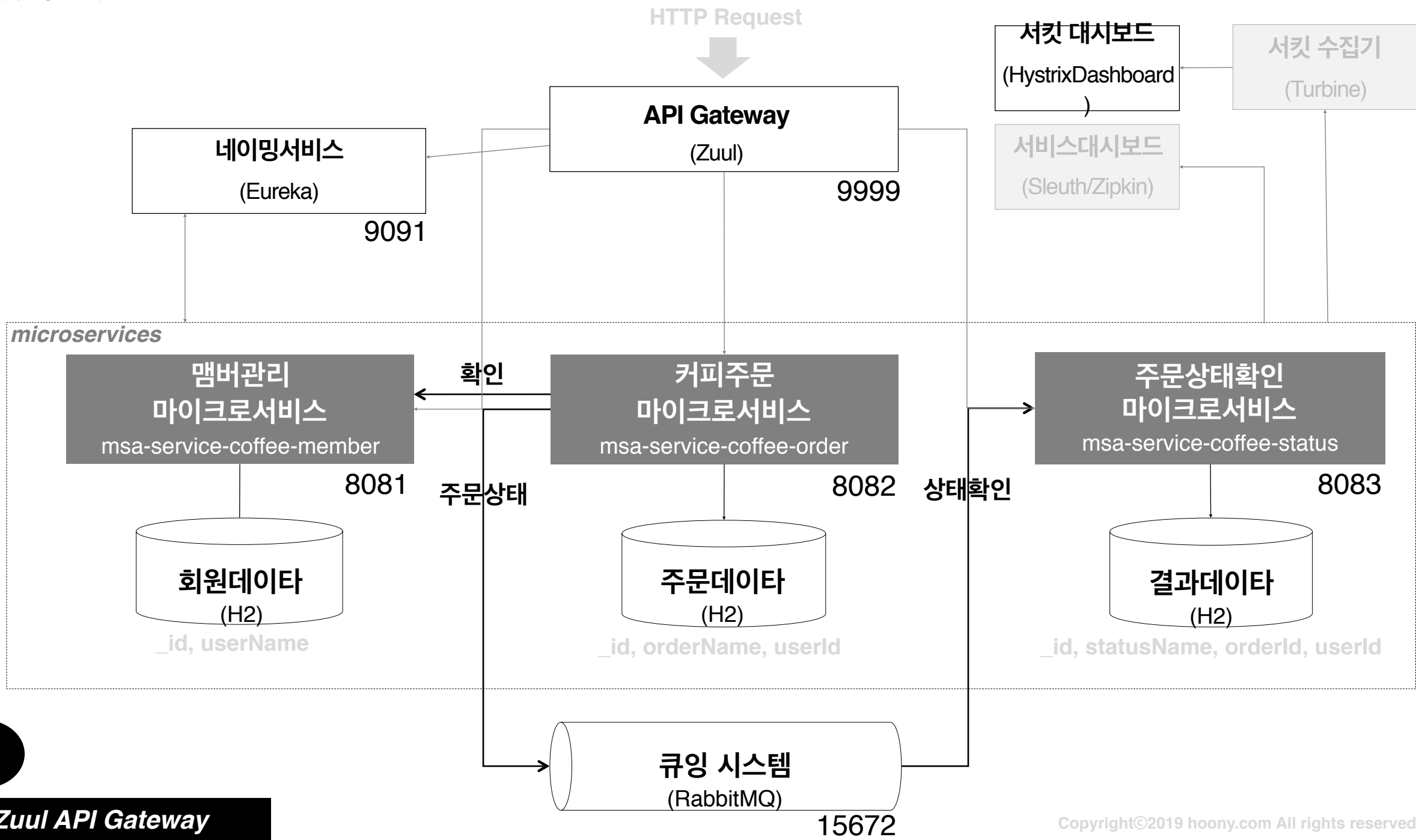
Circuit breaker

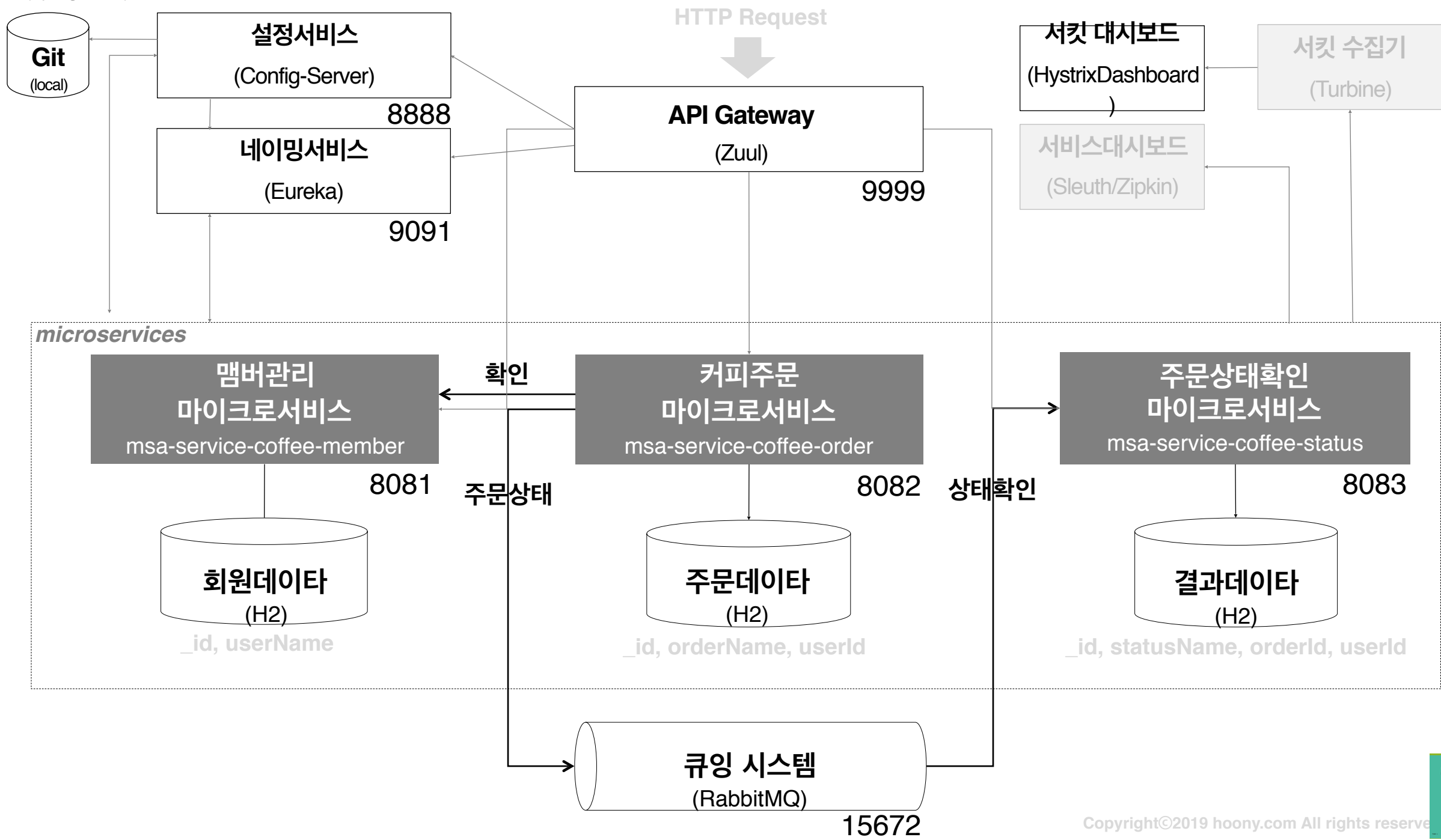




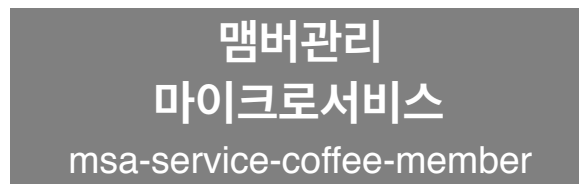




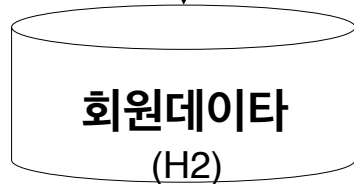




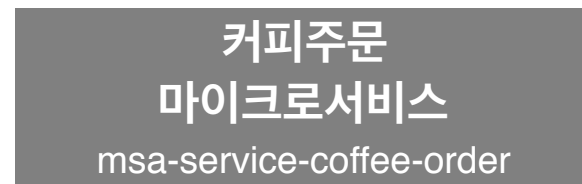
microservices



8081



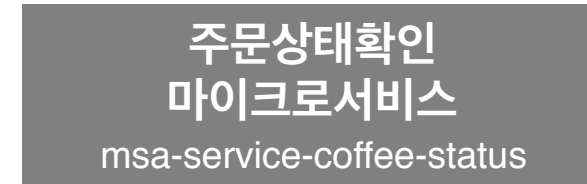
_id, userName



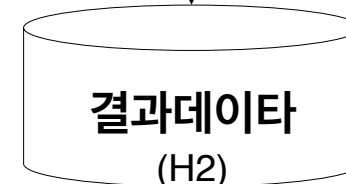
8082



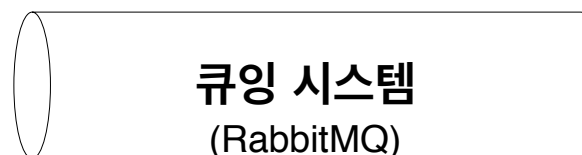
_id, orderName, userId



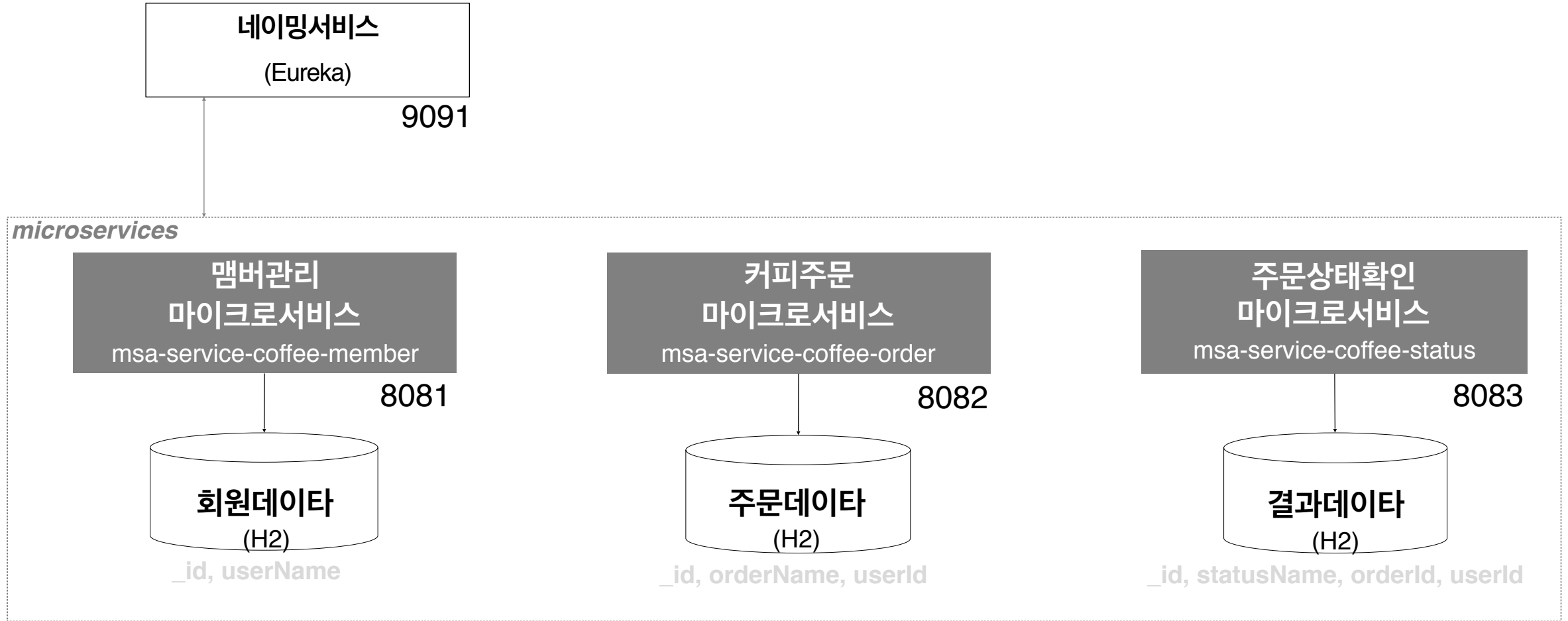
8083



_id, statusName, orderId, userId

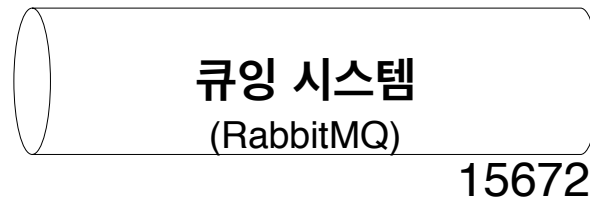


15672



1

Eureka



- IntelliJ 'Lombok' Error 발생 시 'domin패키지의 vo', 'sprin패키지의 jvo', 'system패키지의 vo'를 모두 'setter/getter'로 변경 후 IDE restart

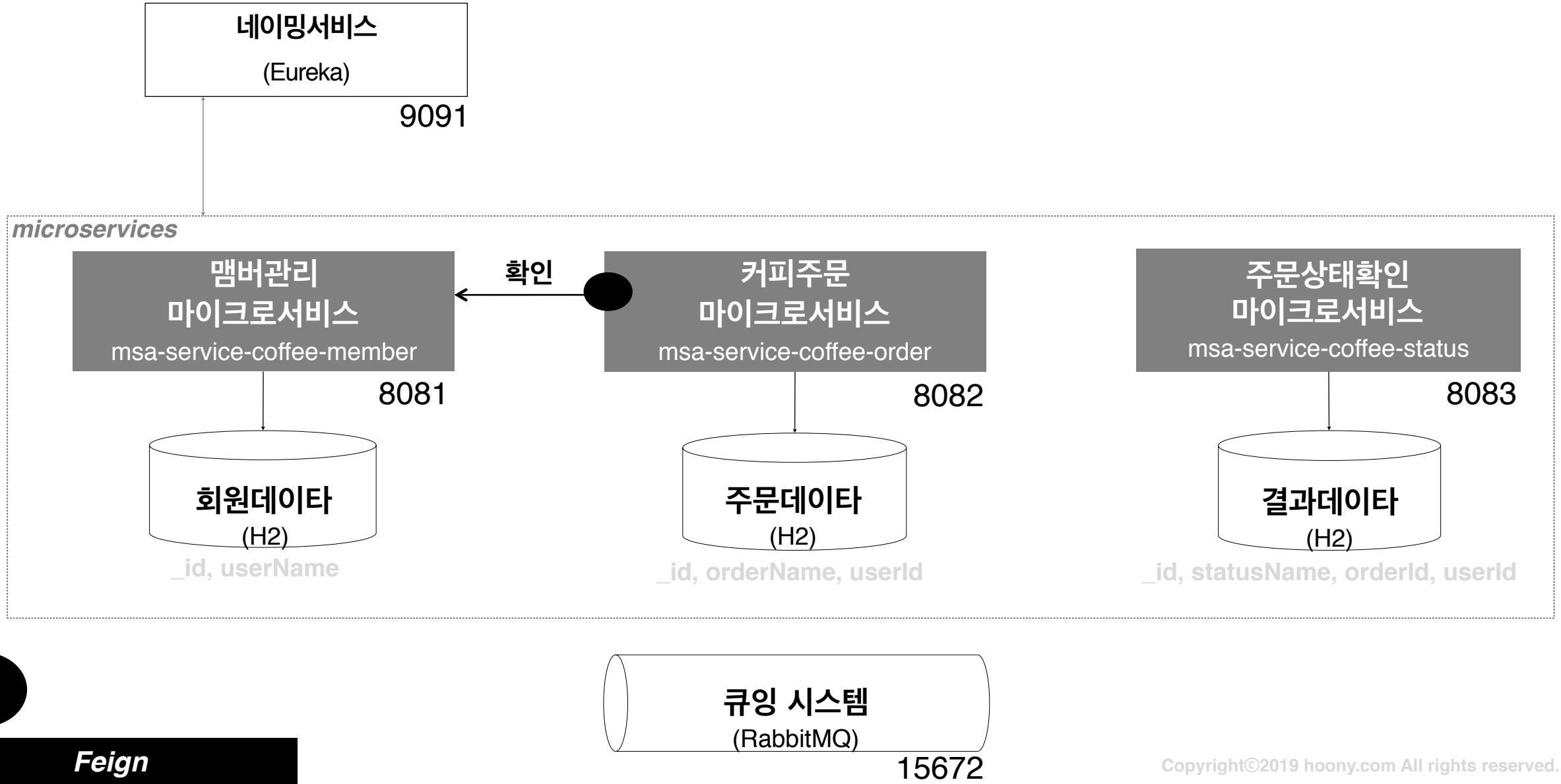
1. Eureka Server

- **build.gradle** 에 Eureka Server Library 추가
- **application.yml** 에 Eureka 설정확인, 주석처리 부분은 Eureka 서버 2개를 이중화 구성 할 경우에 설정
- **MicroServiceApplication.java** 에 @EnableEurekaServer 어노테이션 확인
- <http://localhost:9091> Eureka 대시보드 화면 확인

2. Member Service [멤버 등록, 조회]

- **build.gradle** 에 Eureka Client, Web, JPA, Hystrix, Config Client, Swagger 등 Library 확인
- **application.yml** 에 Eureka Client 설정확인, 이 외 DB 접속을 위한 H2, Logging 정보 확인
- **MicroServiceApplication.java** 에 @EnableEurekaClient 어노테이션 확인
- <http://localhost:9091> Eureka 대시보드 화면에서 '**MSA-SERVICE-COFFEE-MEMBER**' 서비스가 등록 되었는지 확인
- <http://localhost:8081/swagger-ui.html> 접속
- swagger-ui 화면에서 [POST] </api/v1/member> addUser API를 이용하여 'tester' 를 멤버로 등록
- swagger-ui 화면에서 [GET] </api/v1/member> getAllUser 에서 방금 등록한 'tester' 멤버를 확인

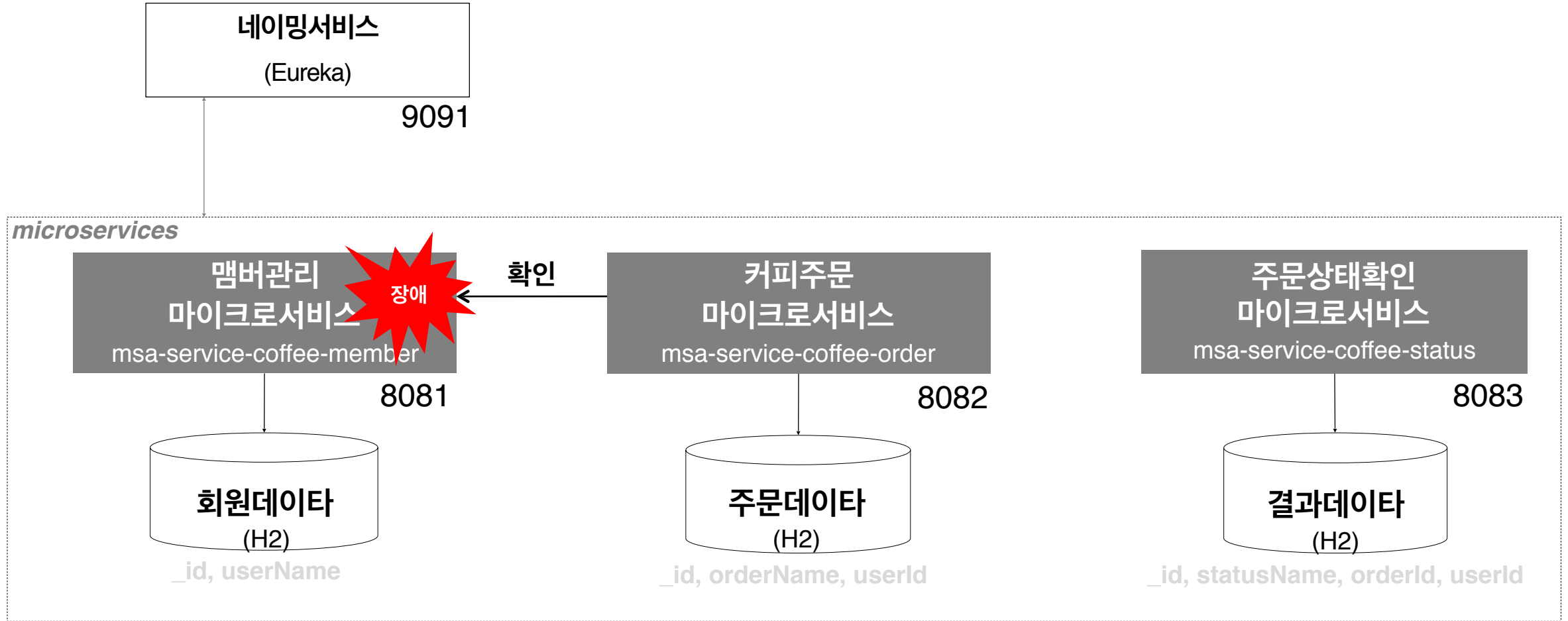
1 <http://localhost:8081/h2console> 로 접속, (JDBC URL: jdbc:h2:mem:testdb, user: sa, password:)



- IntelliJ 'Lombok' Error 발생 시 'domin패키지의 vo', 'sprin패키지의 jvo', 'system패키지의 vo'를 모두 'setter/getter'로 변경 후 IDE restart

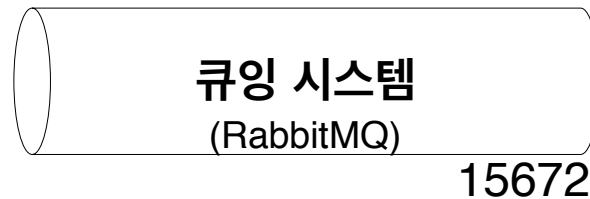
3. Order Service

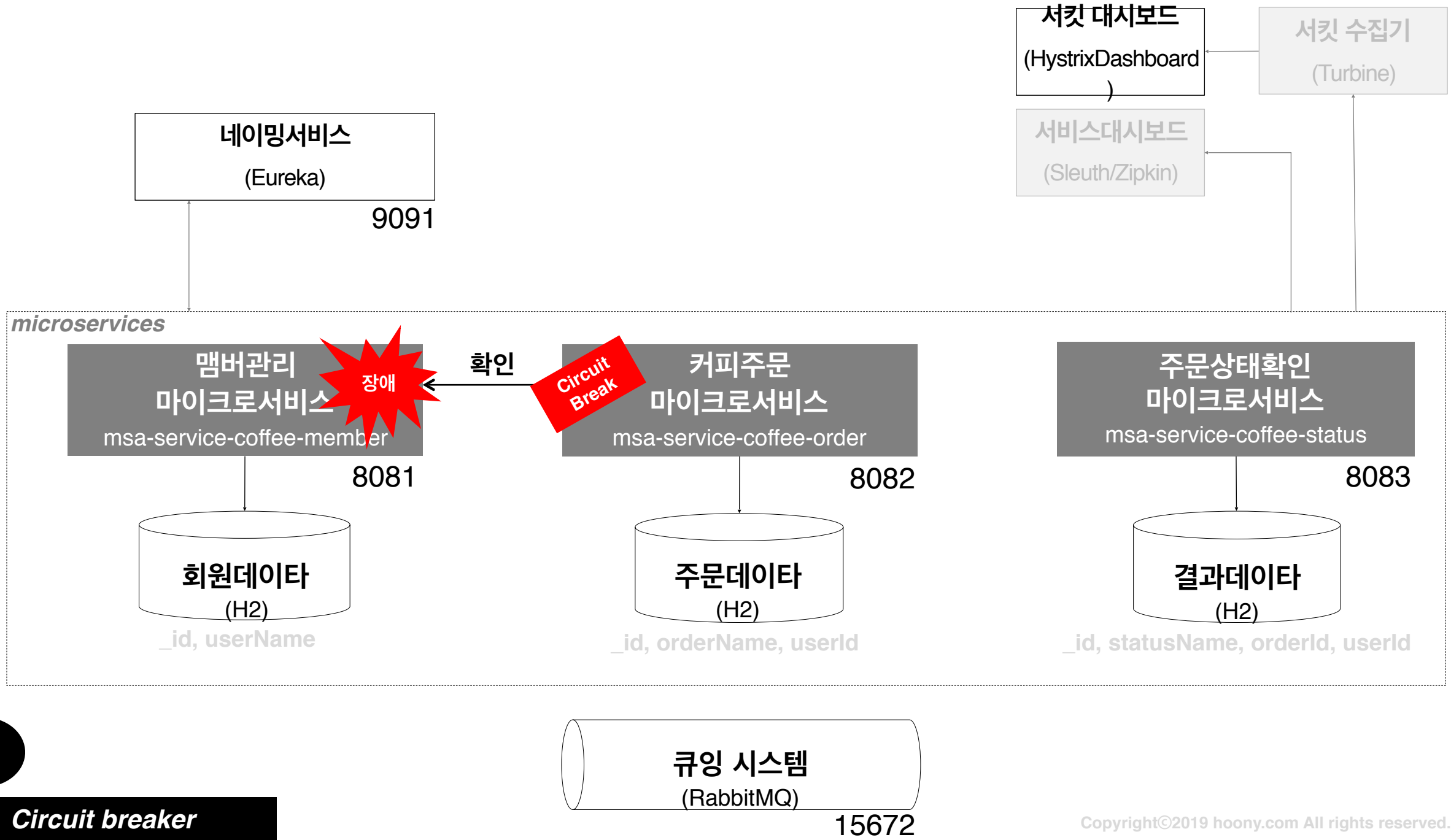
- **build.gradle** 에 Eureka Client, OpenFeign, AMQP, Web, JPA, Hystrix, Config Client, Swagger 등 Library 확인
- **application.yml** 에 OpenFeign, rabbitmq, eureka client, DB설정 등 확인
- **MicroServiceApplication.java** 에 @EnableFeignClients, EnableEurekaClient, HystrixDashboard 등 어노테이션 확인
- spring패키지의 Feign 및 system패키지의 Queue 구현체 확인
- Order Service 기동 후 <http://localhost:9091> Eureka 대시보드 화면에서 확인
- <http://localhost:8082/swagger-ui.html> 접속
- **swagger-ui** 화면에서 [POST] **/api/v1/order** addOrder API를 이용하여 오더를 등록
이 때, 'userId' 는 앞서 등록한 Member Service 의 'userId'를 입력한다.
- **swagger-ui** 화면에서 **/api/v1/order** getAllOrder API를 이용하여 확인, 조회 결과로 'userId'로 조회한 'username'을 확인 할 수 있다.



3

Circuit breaker



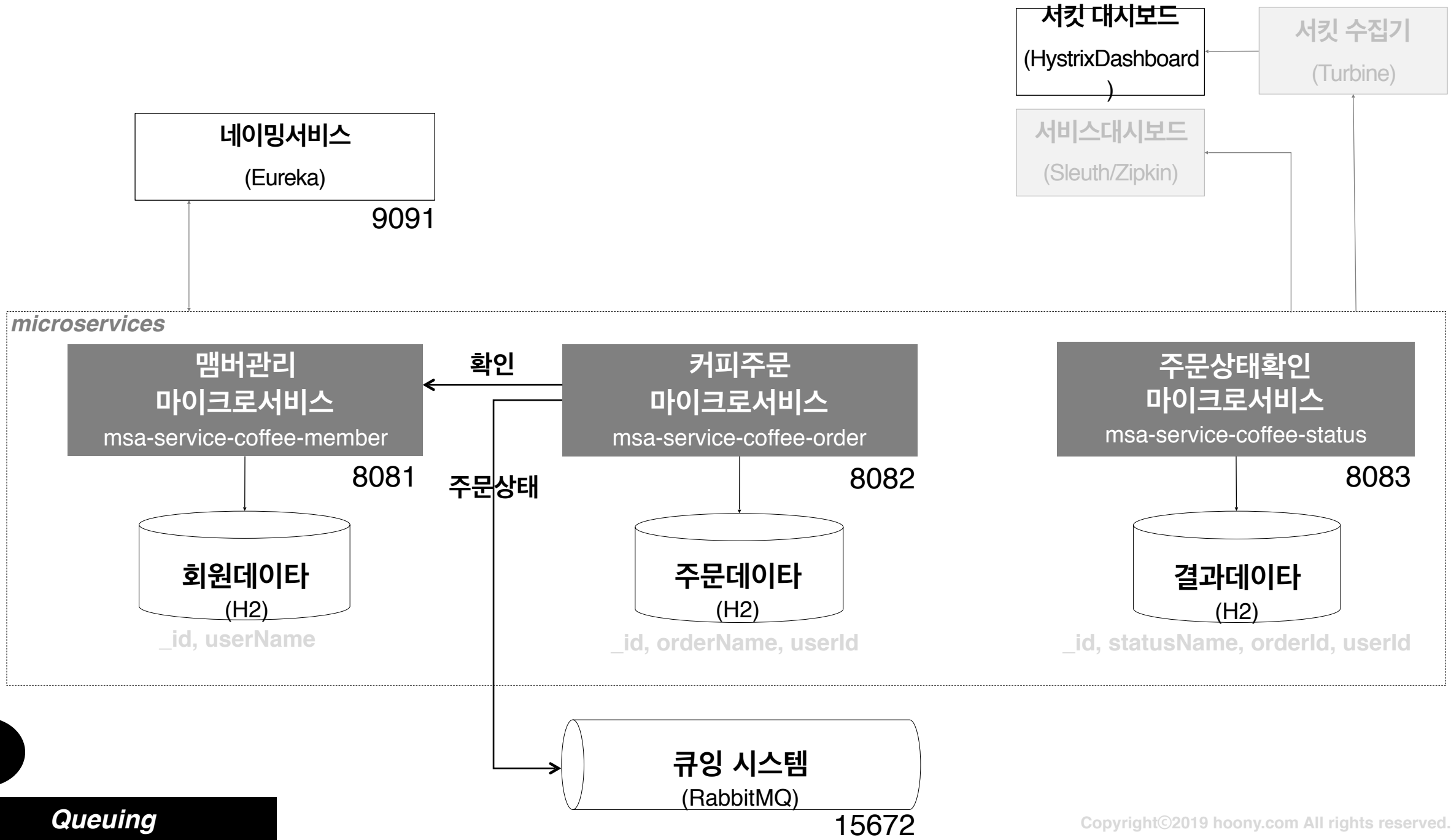


- IntelliJ 'Lombok' Error 발생 시 'domin패키지의 vo', 'sprin패키지의 jvo', 'system패키지의 vo'를 모두 'setter/getter'로 변경 후 IDE restart

4. Member Service Down

5. Order Service [Order 등록하여, Feign 동작확인]

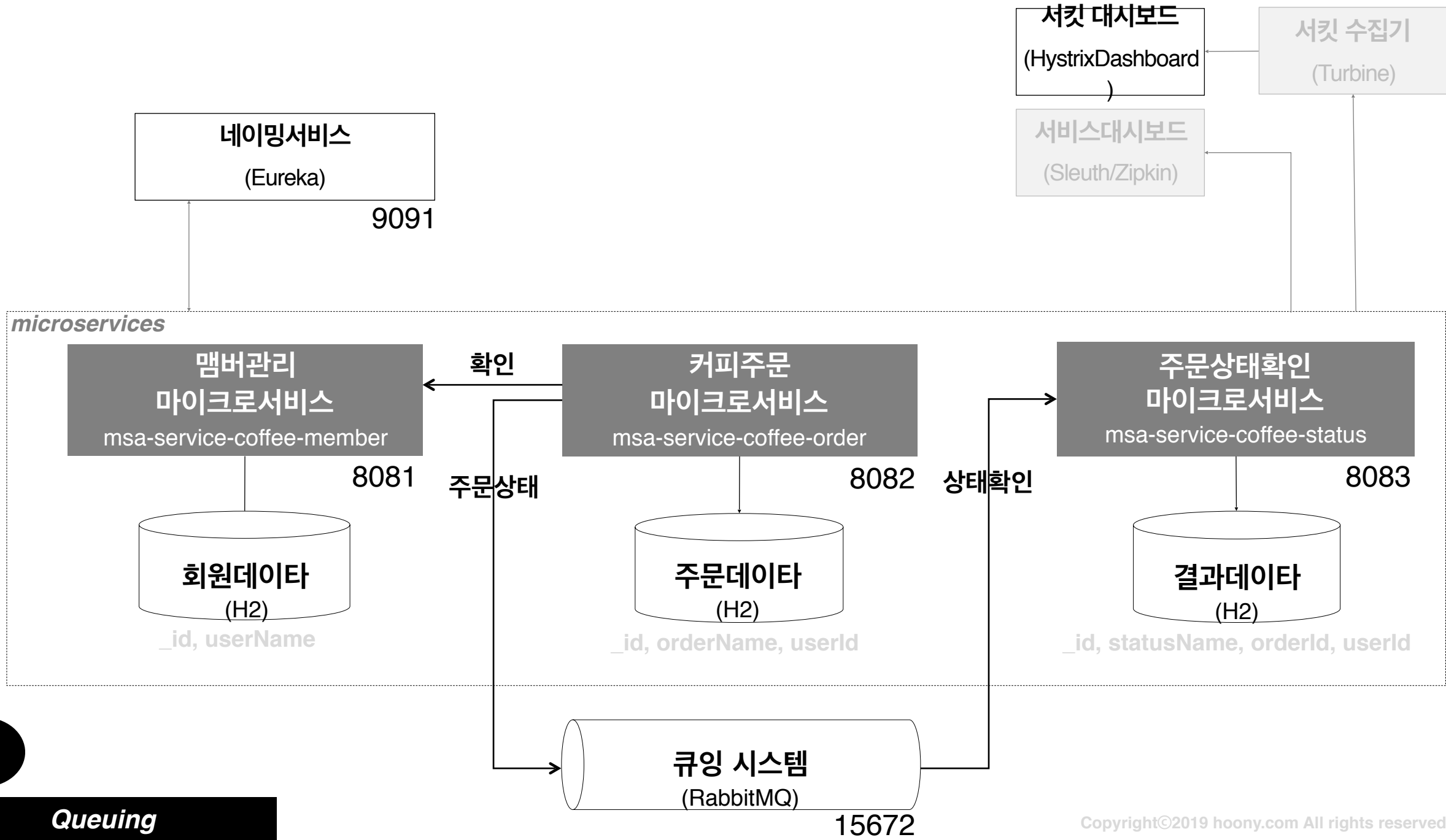
- <http://localhost:8082/hystrix> 접속
- 입력 박스에 <http://localhost:8082/actuator/hystrix.stream> 입력하고 'Monitor stream' 클릭하여 대시보드 화면으로 접속
 - * 만약, 대시보드 화면에 Stream 메시지가 표시되지 않으면, 브라우저 창을 하나 띄워서 <http://localhost:8082/actuator/hystrix.stream> 입력한다.
- <http://localhost:8082/swagger-ui.html> 접속해서, **[POST] /api/v1/order** addOrder API를 이용하여 오더를 등록한다. (3. Order Service 참고)



- IntelliJ 'Lombok' Error 발생 시 'domin패키지의 vo', 'sprin패키지의 jvo', 'system패키지의 vo'를 모두 'setter/getter'로 변경 후 IDE restart

6. Queue

- <http://localhost:15672> 접속 (guest / guest)
- “Queues” 탭에서 “ORDER_COFFEE_MESSAGE” 에 등록된 메시지 확인



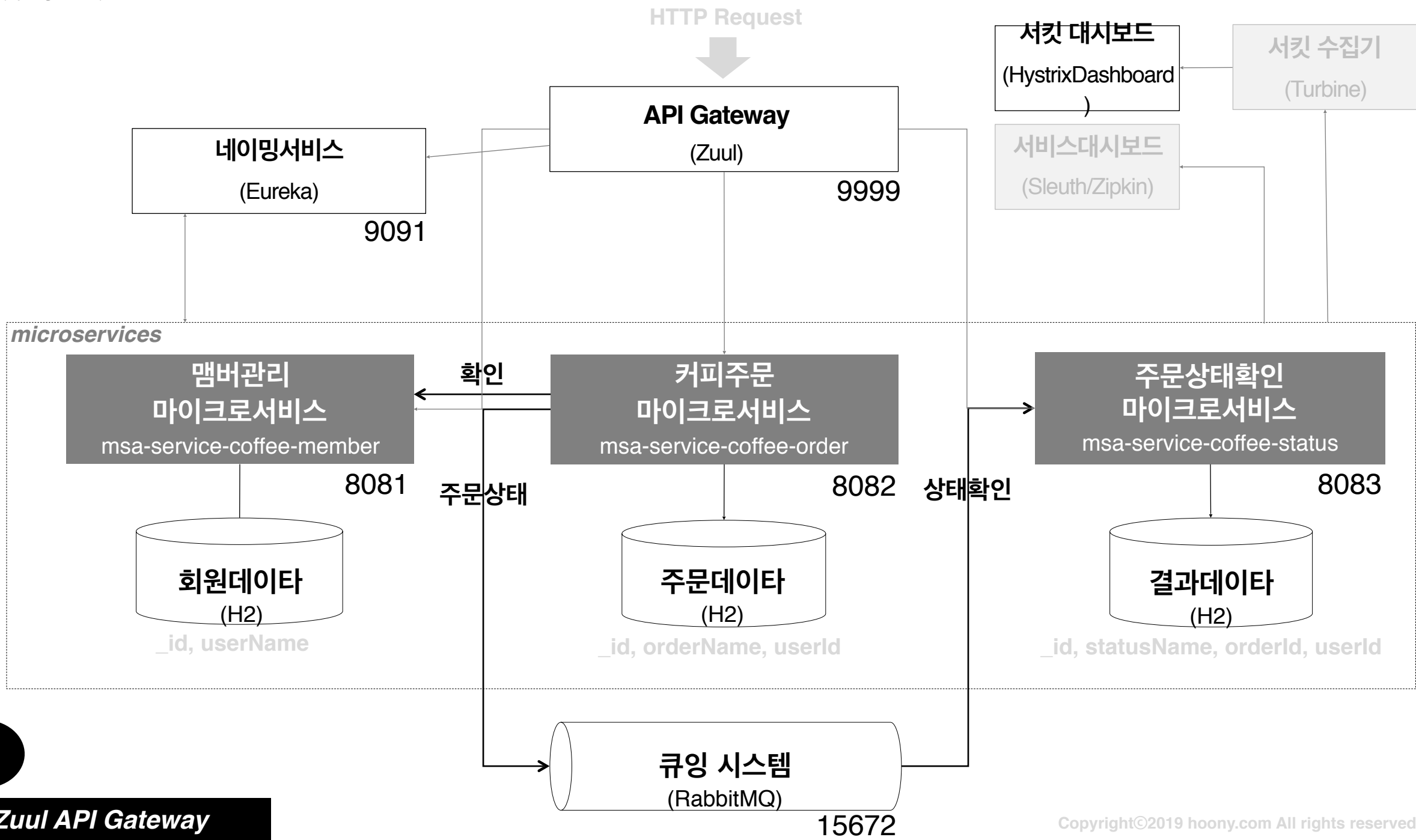
- IntelliJ 'Lombok' Error 발생 시 'domin패키지의 vo', 'sprin패키지의 jvo', 'system패키지의 vo'를 모두 'setter/getter'로 변경 후 IDE restart

7. Status Service

- **build.gradle** 에 Eureka Client, OpenFeign, AMQP, Web, JPA, Hystrix, Config Client, Swagger 등 Library 확인
- **application.yml** 에 rabbitmq, eureka client, DB설정 등 확인
- **MicroServiceApplication.java** 에 @EnableFeignClients, EnableEurekaClient 등 어노테이션 확인
- spring패키지의 Feign 및 system패키지의 Queue 구현체 확인
- Status Service 기동 후 <http://localhost:9091> Eureka 대시보드 화면에 접속하여 Status Service 등록 확인
- **swagger-ui** 화면에서 <http://localhost:8083/swagger-ui.html> 접속
- **swagger-ui** 화면에서 **/api/v1/status** getAllStatus API를 이용하여 확인, 결과를 조회

8. Queue

- <http://localhost:15672> 접속 (guest / guest)
- "Queues" 탭에서 "ORDER_COFFEE_MESSAGE" 에 메시지 확인



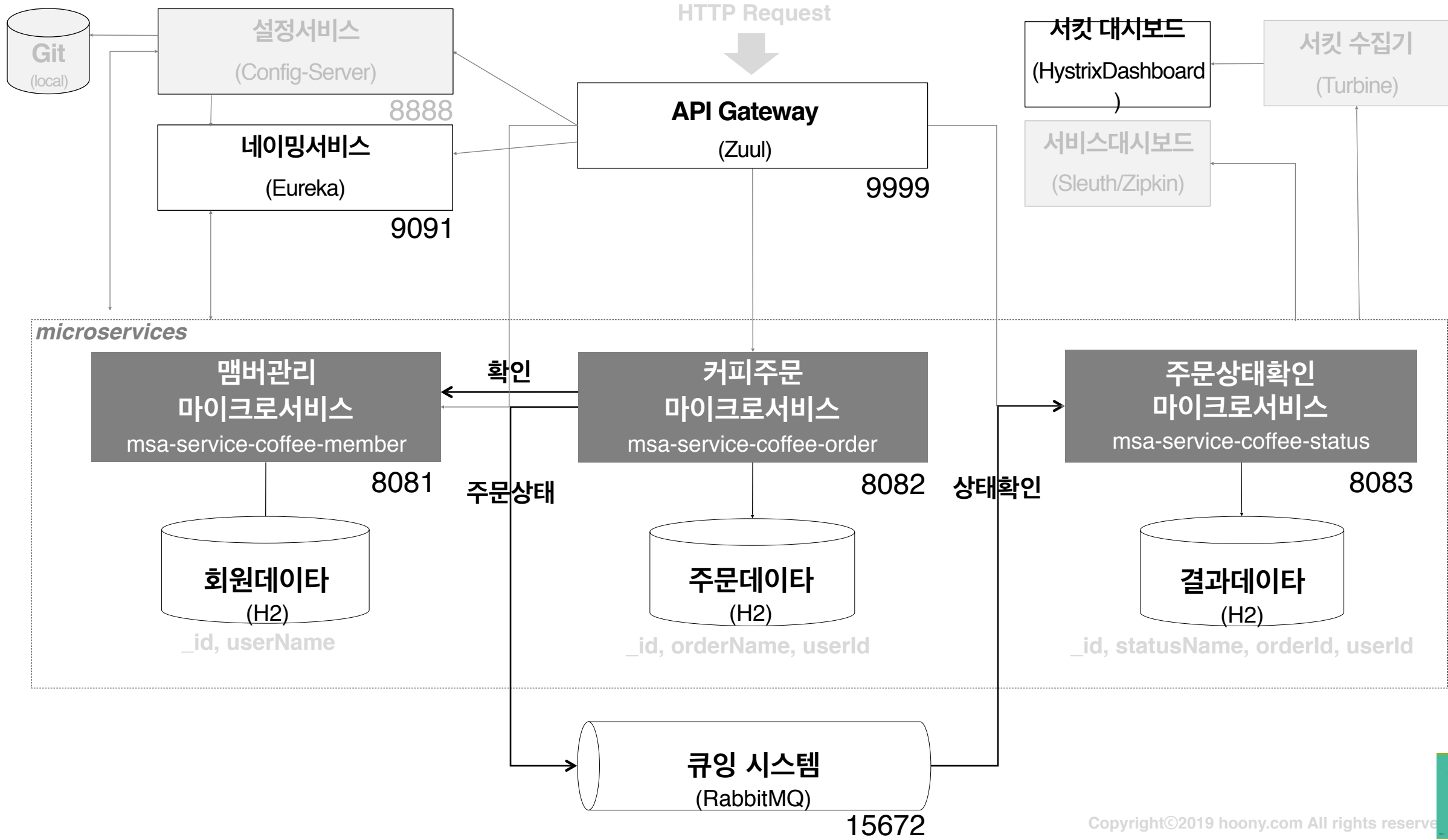
- IntelliJ 'Lombok' Error 발생 시 'domin패키지의 vo', 'sprin패키지의 jvo', 'system패키지의 vo'를 모두 'setter/getter'로 변경 후 IDE restart

9. Zuul Server

- **build.gradle** 에 Zuul, Eureka client 등 Library 확인
- **application.yml** 에 '라우팅(Routing)' 설정 확인
- **MicroServiceApplication.java** 에 @EnableZuulProxy 어노테이션 확인
- Zuul Server 기동 후 <http://localhost:9091/> Eureka 대시보드 화면에 접속하여 Zuul Service 등록 확인
- 인터넷 브라우저에서 <http://localhost:9999/api/v1/status> 를 입력하여 Status Service가 실행 확인

- [application.yml '라우팅(Routing)' 설정]

```
routes:
  member:
    path: /api/v1/member/**
    serviceId: msa-service-coffee-member
    strip-prefix: false
  order:
    path: /api/v1/order/**
    serviceId: msa-service-coffee-order
    strip-prefix: false
  status:
    path: /api/v1/status/**
    serviceId: msa-service-coffee-status
    strip-prefix: false
retryable: true
```



END