Spring BOOT Micro service - Level 2 Fault Tolerance and Resilience

Agenda

1 Understand challenges with availabilty

2 making micro service resilient and fault tolerant

Why do we resilient, fault tolerant, things we could face ,y we face

- fundamental concepts apart from framework

- One Sample Project with Spring Boot and Hystrix framework with these microservice hw to make it more fault tolerance and resilience

What is Fault Tolerance?

- given an application if there is a fault what is the impact of that fault how much tolerance that system have for specific fault

Ex: if one microservice instance goes down what will happen to microservice system where whole microservice application wil go down or part appliaction will go down or if some way of handling failuers in way there is no impact at all

- what Tolerance ur system has for a particualr fault is called fault tolerence

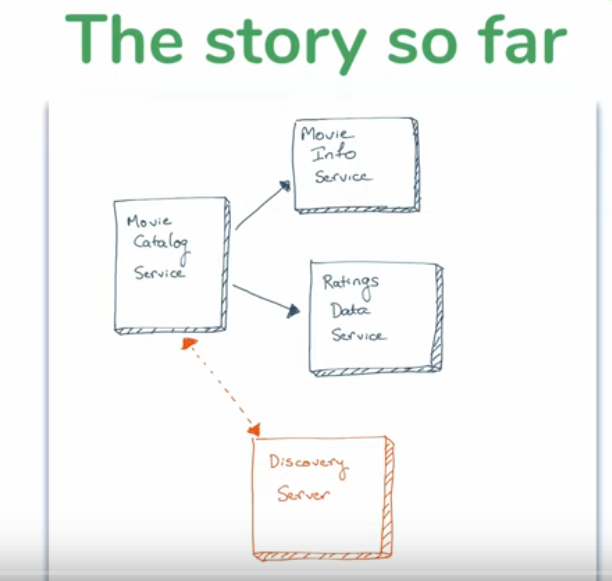
What is Resilience ?

- it is some what diff

- how many faults ur system can tolerate tells how resilient it is

- part of resilience also say how it will bounce back from fault when .any mechanism it correct itself all comes to resilience

We can have system which is very fault tolerant but not resilient its to hard



Level 1 : IMDB app

One Change in code

Using Movie DB in movie information

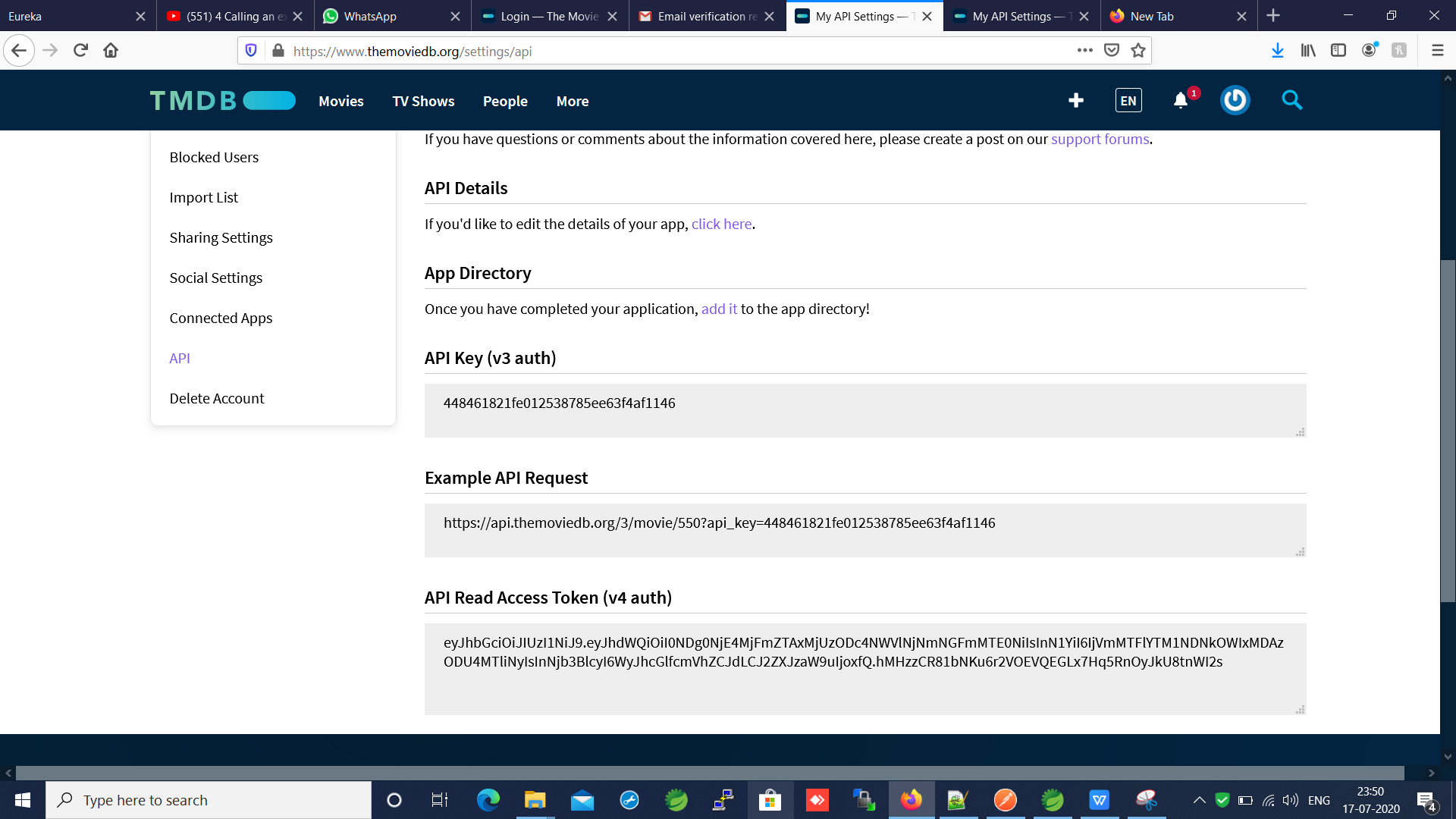
Calling an External APi To get Movie Info live one so more external call more failur we learn

<http://moviedb.org/>

<https://www.themoviedb.org/> go to more and API section here

Willgive how to discover movies

- create account 1st login -> profile and setting -> edit profile -> goto API- > check for Key if not folloe steps to create key



Example API:

<https://api.themoviedb.org/3/movie/550?api_key=448461821fe012538785ee63f4af1146>

<https://api.themoviedb.org/3/movie/150?api_key=448461821fe012538785ee63f4af1146>

Change movie id to get dynamic data

Now adding to code in movie info service

Create a class MovieSummary whose proprty matches the key of themoviedb api response keys to map it in resttemplate call

@GetMapping("/{movieId}")

public Movie getMovieInfoMovieDb(@PathVariable("movieId") String movieId) {

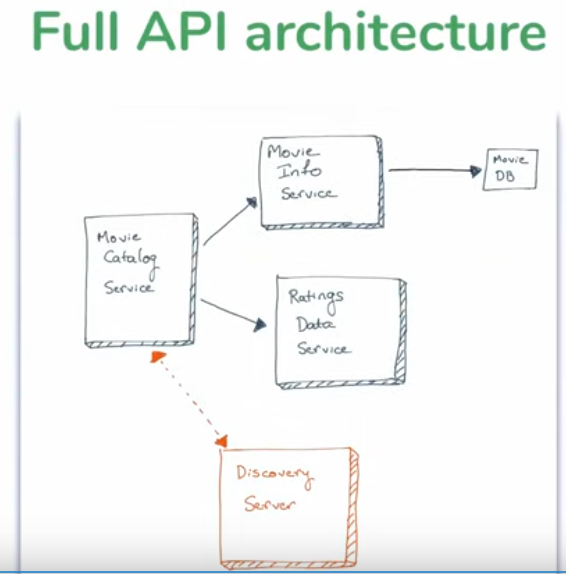
MovieSummary moviesummary = restTemplate.getForObject(

"https://api.themoviedb.org/3/movie/" + movieId + "?api\_key=" + apiKey, MovieSummary.class);

return new Movie(movieId, moviesummary.getTitle(), moviesummary.getOverview());

}

Run n test movie info service now run catalog service will get dynamic data from API



- How do we make these resilient?

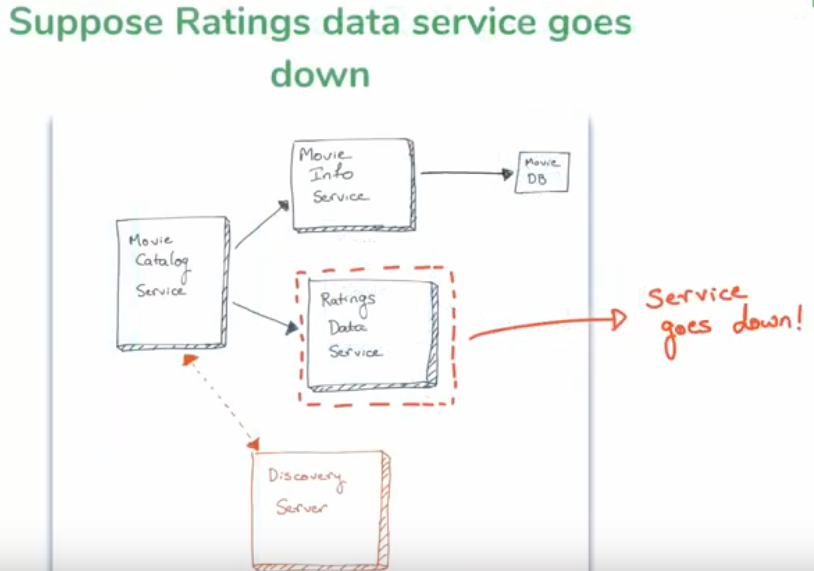
-first is this A/c is resilient and fault tolerence it is not , a/c is normal we dnt have try catch block not handling anything here

- 3 serveice with no fault tolerence no handling noting

- if one of the service of movie info, movie rating goes down whole movie cata log will go down

Lets make resilient before we should know what are the issues with microservice we can face, what are the things go wrong

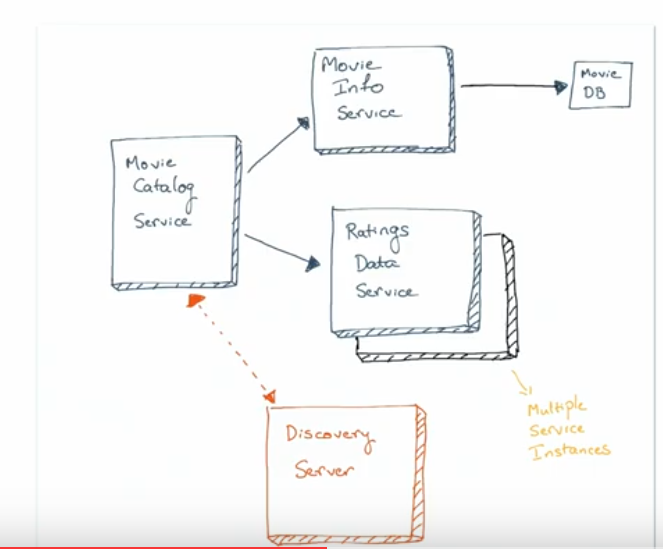
Issue 1: If instance go down



Solution : run multiple instances

- run multiple instance of movie rating service in different sysytem and port

- if one instance go down other will be used since discovery server will get registerd and client side we have load balance technology called Ribbon wt it does is round robin if we sees more then one instance then I called last time now I will cl another instance it will equally distribute it

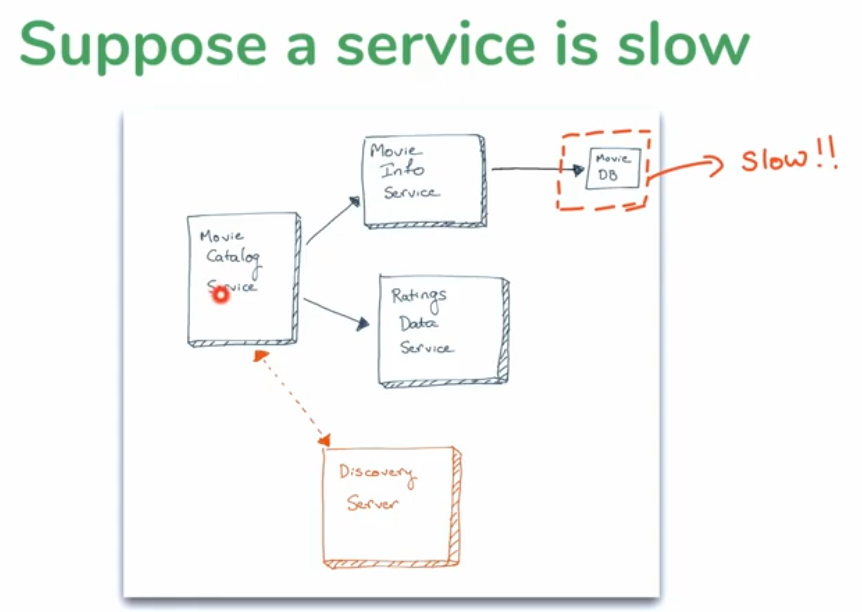


-if one instance to go down and tempoaly another instance will handle the load till another is up automated system of making it up r support guy need to make it up

Issue 2: What if Microservice instacne is slow

- micro service being slow is much more imp issue then down it is big problem

Ex: if movie Db is slow it makes movie info slow any api consuming movieinfoAPI/ service automaticaly it wil be slow now catalog will be slow since its consuming movie info





-bcz of movie info being slow it cause slow to movie rating api alsoy it is being slow even it is not related to movie info details?

- how it is possible how can unrelated service be slow if some other service being not related this is actuly slow

Timeout may be but its solution

- rating data servie not depend on movie info service

- but catalog service depened on both service but those 2 service are indepedent of each other

Correct Answer is: Thread

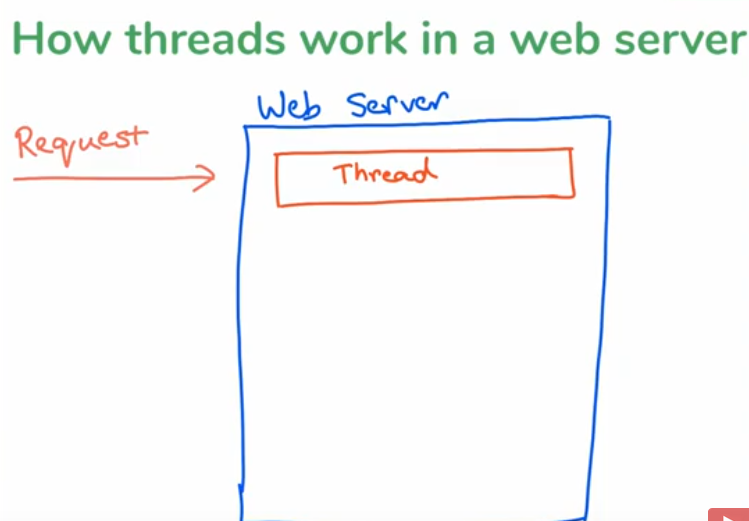
How thread work in web server

- webserver get request it has process request and give response

-so do that process it creates a thread to handle it to something to execute in webser

For eveery req thread will be created for that

- tomcat will create a thread fr that and excutes it once rquest is complete and repsond send back it deletes thread



What happens when request come thread created and thread is taking a while to process it and even before the thread is done we get another request another threads get created same like this c more req coming threads being created n not processed

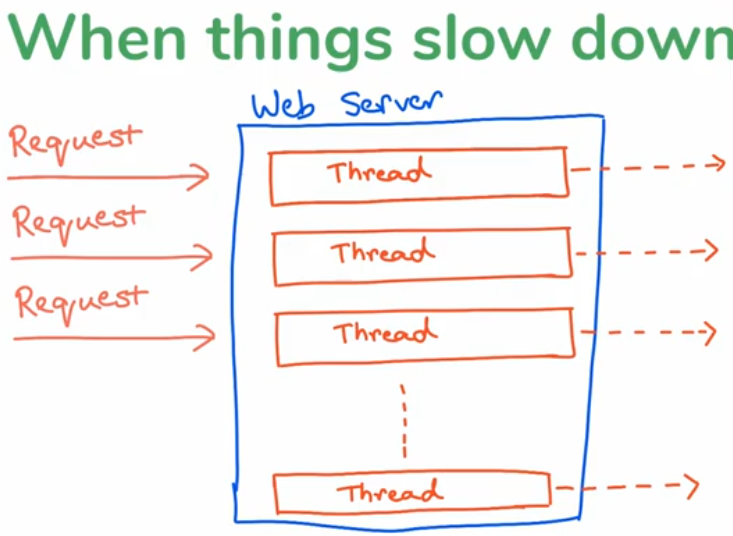
-more request coming as long as thread does its job return response back get freeed up everthing is cool

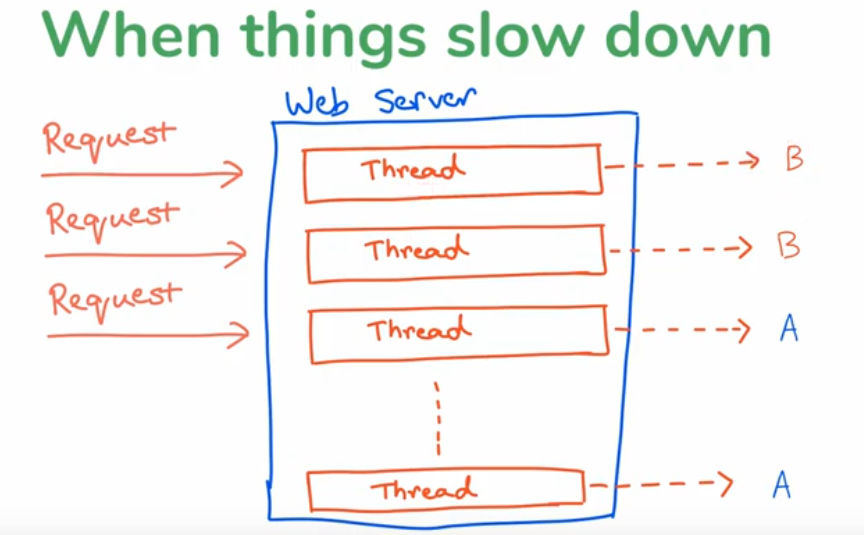
- req coming in a faster way then a threads get fulfilled and thread get removed if not will have bunch of threads waiting

- soon our reouserce get consumed like max num thread of tomcat or jboss get consumed max number of concurrent thread consumed like hardware run out

-this what happens in webserver making it slow in terms of thread since thread not being cleared out







- now think B as movie info service A as rating since B is slow thread not clearedtill it process like multiple B among that come n clears fast until the concurretn thread is maxed out by B service once it is max out even A is fast and no issues it will be slow bcz there is no free thread to allocate it.

When thread maxed out req come from A then it not have thread to create it so it will be slow waiting for other threads to cleared out so it will be slow automatially

Solve this Problem : Timeouts

sol1 Using timeouts

- given as much as time still your not written so return an error timeout then that concurrent thread will be cleared or go away

Sol2 : increase size of tomcat thread pool

Still not a soultion bcz size may increas even more with req

When ever web applciation is down usual we will do is refreshing it to check if its up or not but this will make again sending req to server

- at some point with all users doing refresh it will reach some extent point n goes down so increasing resource is not good thing not a long term solution

But with timeout technically we can fix this issue

-removing threads taking to much time

- if thread is taking to much time then it gone allow for faster threads to go through

- How do we set timeout we are using Spring RestTemplate make API call and return response back we are not given any time out or error handling changes

Setting timeouts on SpringRestTemplate

2 ways to set time out

First Timeout approach and simplest way

We want create a timeout to movie catalog app in where RestTemplate Bean Is created this calling two service wanna make that timeout

- any service calls another service can have time out

- movieinfo service can have timeout since its calling movie db

- any service calls an external service using resttemplate can have an timeout

- below is simple Bean Creating Simple RestTemplate Constructor but we have overiden RestTemplateConstructor with time as parameter

@Bean

@LoadBalanced

public RestTemplate getRestTemplate() {

return new RestTemplate();

}

@Bean

@LoadBalanced

public RestTemplate getRestTemplate() {

// this class what helps to create a timeout property and pass it to http client

HttpComponentsClientHttpRequestFactory clientHttpRequestFactory = new HttpComponentsClientHttpRequestFactory();

clientHttpRequestFactory.setConnectTimeout(3000); // 3 sec =3000

// return new RestTemplate();

// retruns RestTemplate with timeout so 3 sec will time out if not timeout will

// come fr req

return new RestTemplate(clientHttpRequestFactory);

}

Does doing this fix the issues ?

- No only part of issues is resolved only block thread problm will be resolved

- but prblm for short and long term also

- it doenst matter one service or many each req will be set fr 3sec timeout it wait for 3 sec if not no response time out

But also it is an issue y means still thread is existed on server not cleared it only made time out

-what if request are coming in are faster then then wt time out can do to remove something out of thread

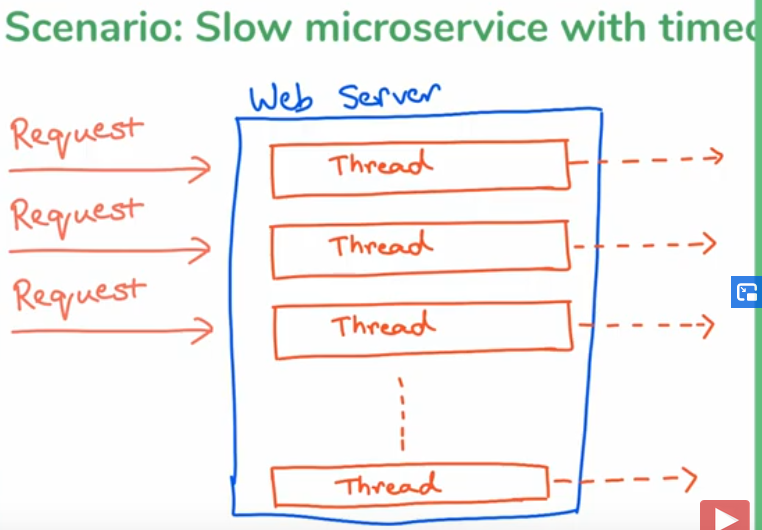
- Request is set to 3 sec I send req/sec

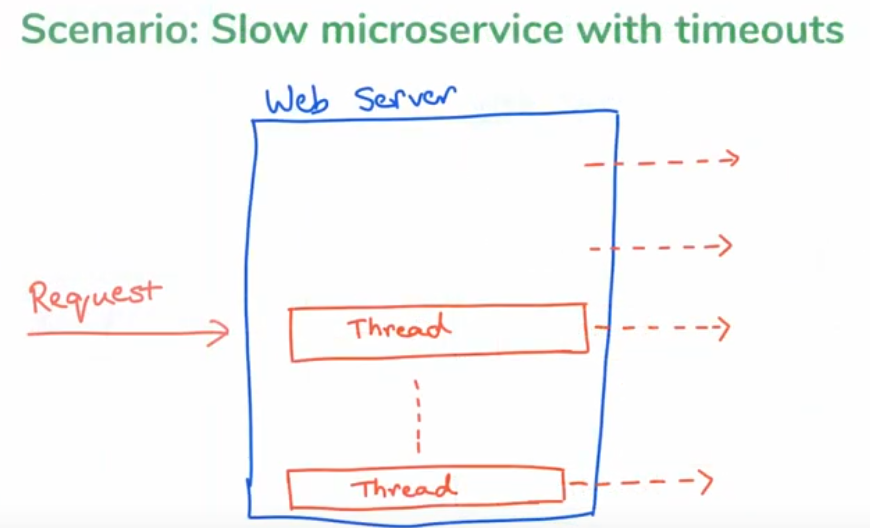
even though time out eventually clear the threads after time out period

The number of request coming in the frequnecy are to fast r faster then timecout can do to remove request from the thread

- remove thread from thread resource

- it is only temp if it lesser frequecny its ok





- Timeout solves issue but not fully - partly solved

So Another Solution

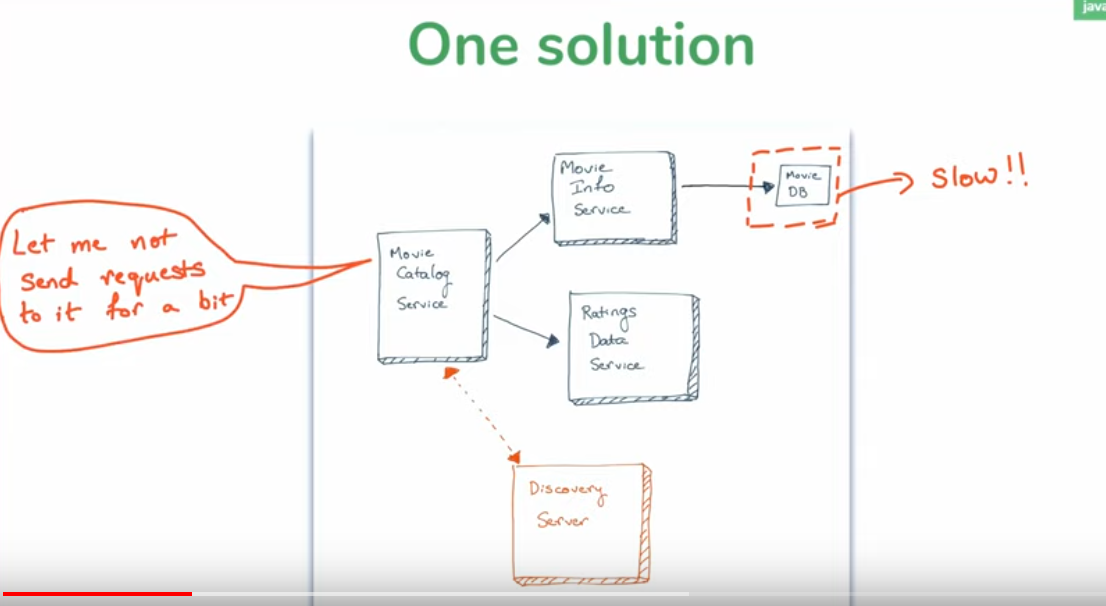
Making movie catalog sevice smarter

- movie catalog will say I am sending so many req to movie info he is responding not on time he is slow

- rather then continouing sending request eventually run out of resources for some other faster service

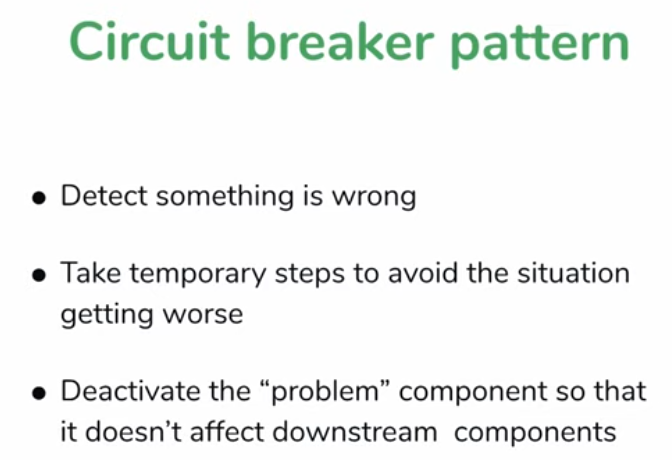
- will check which server is slow and not send request to it for a bit and try after period of time try if it recoverd if not again dnt send req for a bit

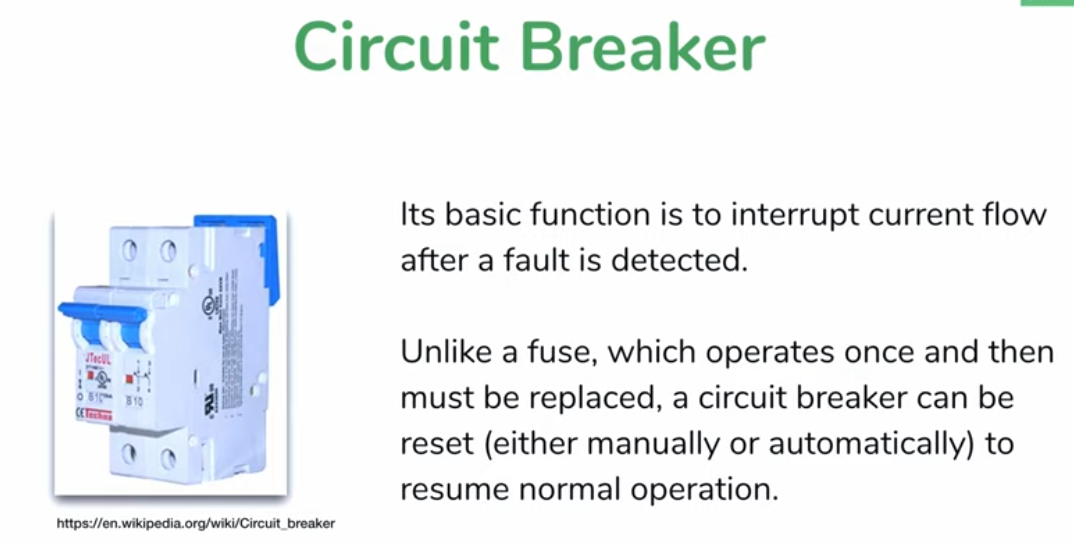
Solution : when u realize something wrong with selected microservice don’t even call that microservice give hold for while try after some time



This is Popular pattern for fault tolerence in micro service

Steps: Common steps for handling fault tolerence



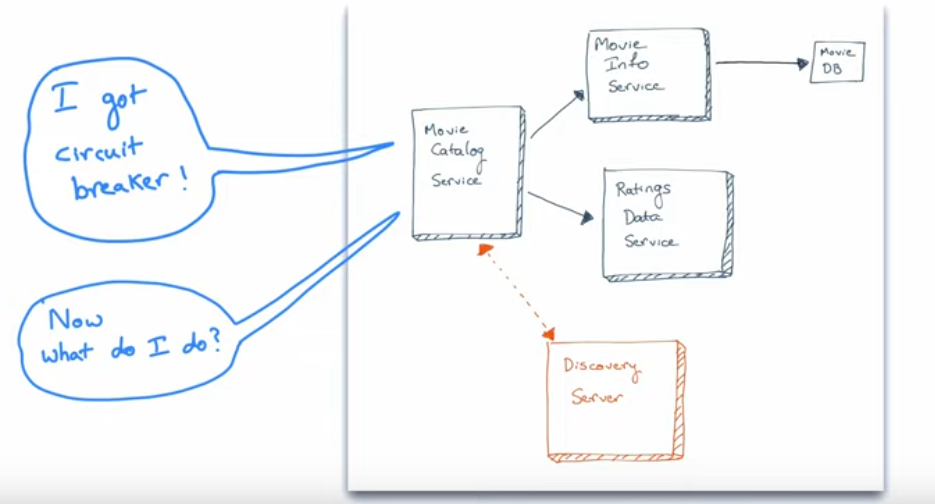


Applying to out Project

-Technically we can apply circuit breaker to any microservice calling an another microservice

- bcz when there is a call it can lead consumption of resourecs

- movie catalog service is a conumser so he should have circuit breaker



Need to provide one mecahnism like movie catalog knows movie info is slow if somebody ask for movie info it has to give something else with out calling movie info service so need to send back something so request thread can be closed

Circuit Breaker

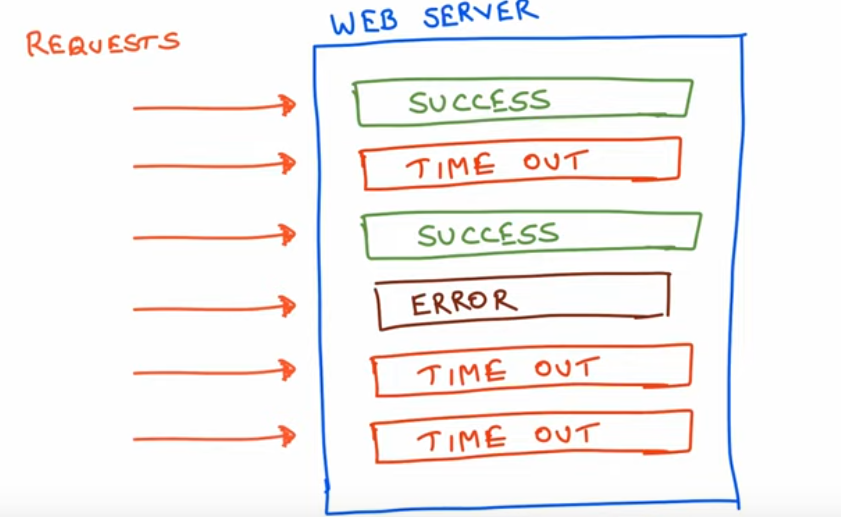
- Before we talk what to do when circuit break

- lets talk when a circuit should break . circuit break should be triggered

Consider one service as example continous request coming

One request success 2nd is timeout we cnt make it halt fr that only that will be harsh

- for one timeout we cant break the circuit that will result in other faults we should have certain pint to break it



- if any point of time any 3 last request failed/ timed out then we will break the circuit still it is not valid

- this logic need to be set before breaking circuit

- what are the parameters to break the circuit

Circuit breaking parameter

When Does Circuit Trip

1 considering n request for the decision ex: in last five req how many failed

- last n request need to be considered

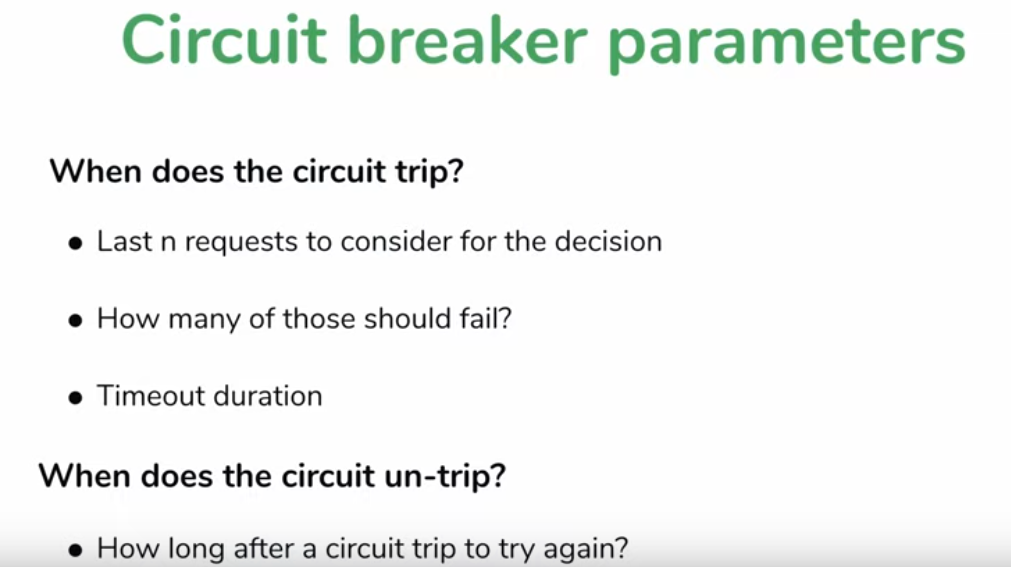
2 in last req how many should be failed in 5 if 3 failed

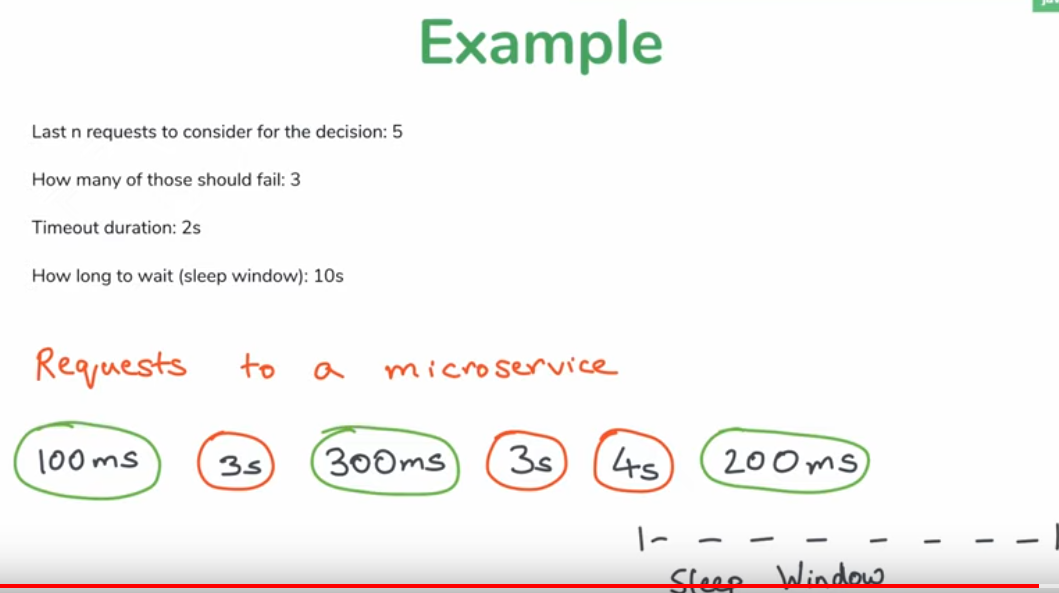
3 timeout duration - consider it time out request and make it failure if it crosees the line

When Does Circuit Un-trip

- when does circuit comes to normal

1 how long after a circuit trip to try again





These example parameter are bad choice for for production for testing its ok

- in example out of 5 req 3 are timed our so circuit break will happen considering that so it will go to sleep fr 10 sec in that 10sec sleep may be get success req faster r slower r timed but will not take risj will break circuit fr 10 sec

Circuit breaker depends on 2 factors

1 how many req coming

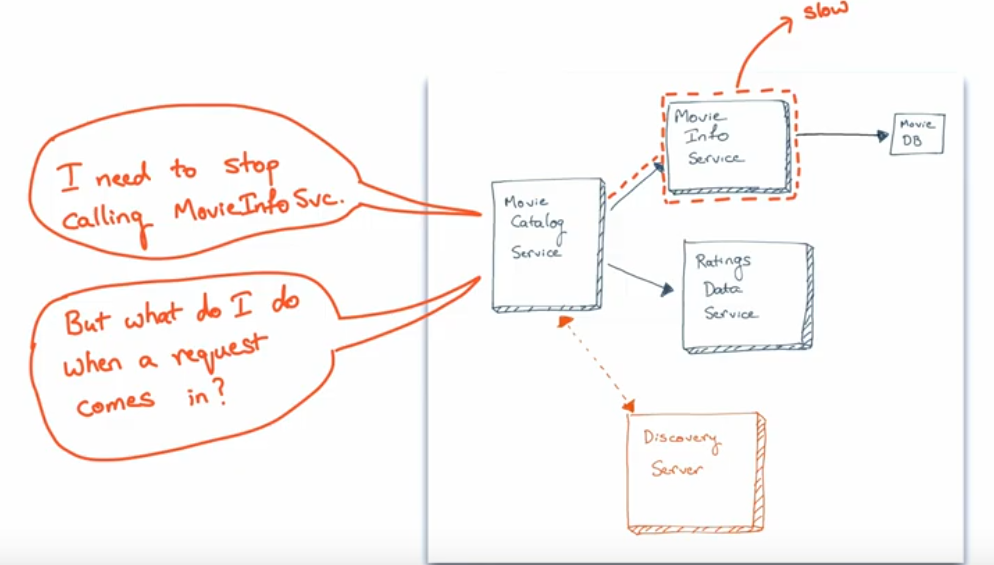
2 thread pool size

Above are results of what causing that issues to break

Next What To do when a when a circuit break



When service is failed now how to handle the request not go to failed server?



Solution: we need to have fallback mechanism alternate mechanism to to handle this when circuit is based

- when a circuit breaked don’t do usual thing do an alternate thing

1 st throw an error has response but not a good way to do

2 alternate is return an fallback “default ” response it is smarter like hardcoded response instead of calling serive whcih is done but not recommended but good option

3 cool option is save previous response(cache) and use that when possible best way to handle it

Ex : we are making an request with user service pass userid and get user info from user info micro service if that user service goes down and we have saved cache of all user information and when user req comes nxt time for whcih u already have in cache and service is down get it from cache end user dnt even know service is broked . getting sensible information may not be valid may nt be accurate but we get sensible information

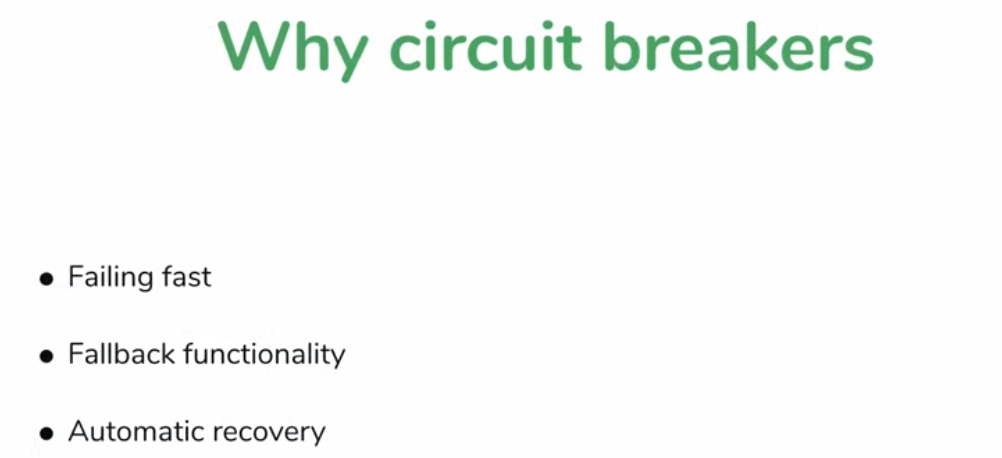
These are handling ways of fallback

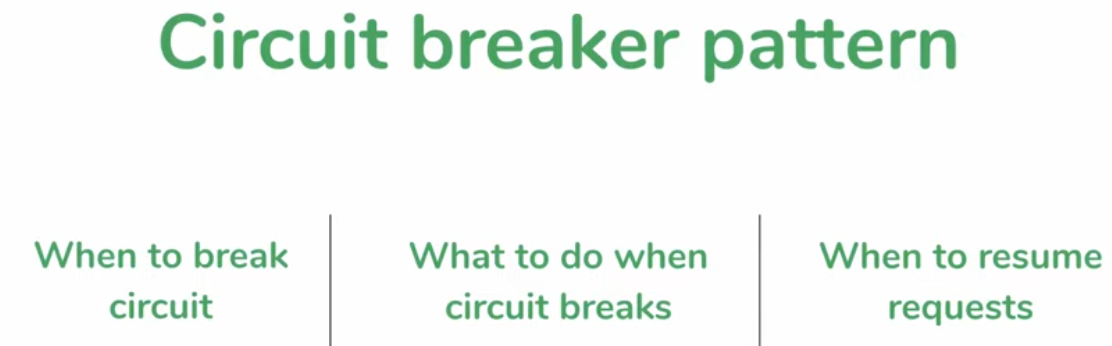


1 st not recommended

2 recommnded

3 is smart way





- Implementing circuit breaker is lot work thread programming concurrency program

Alternate we have a framework : Hystrix to do this

Que 1: how to identify thread pool size

Ans: its not available in PC its what config in servelt container r web server to know what has been configured

-in case of tomcat soem where in xml file

Que 2 : any theory to determne the circuit breaker parameter values

Ans: we will get but trail error can use performance testing to know it

Que3 : if default fallback response implemetned is the clent shold kow distinhwish between from nrml response

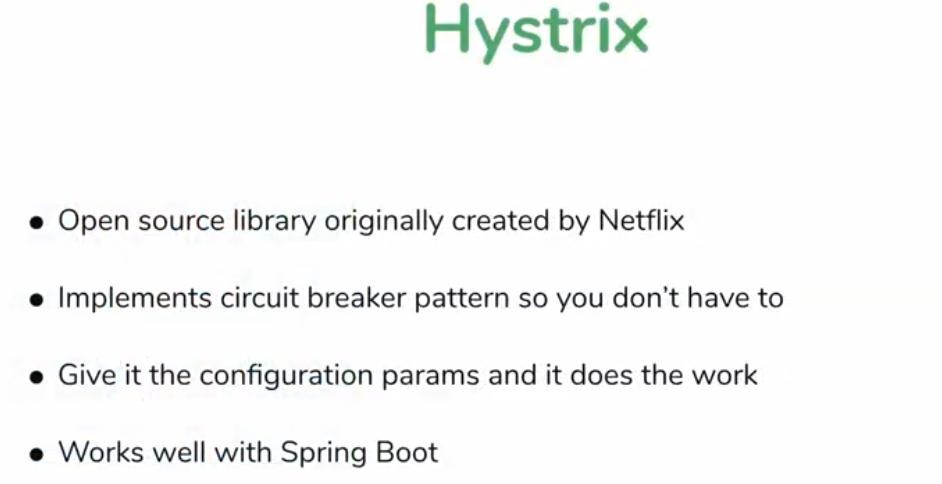
Ans: recommeded user should not know that

Even for throwing error think if its is down it will throw error the caller get error thrown he will cl catch block again he need to handle in his code

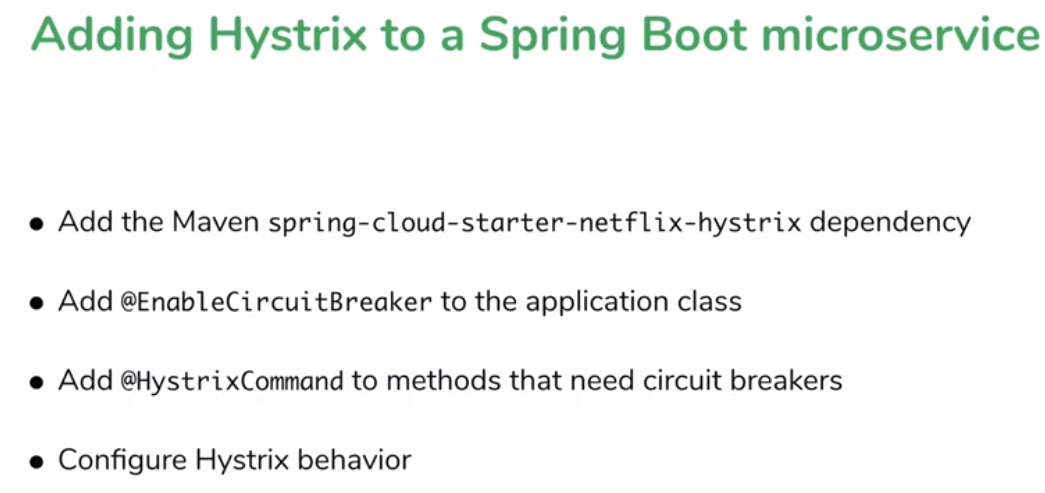
Que 4 : what if cached response coming from circuit breaker fallback is outdated

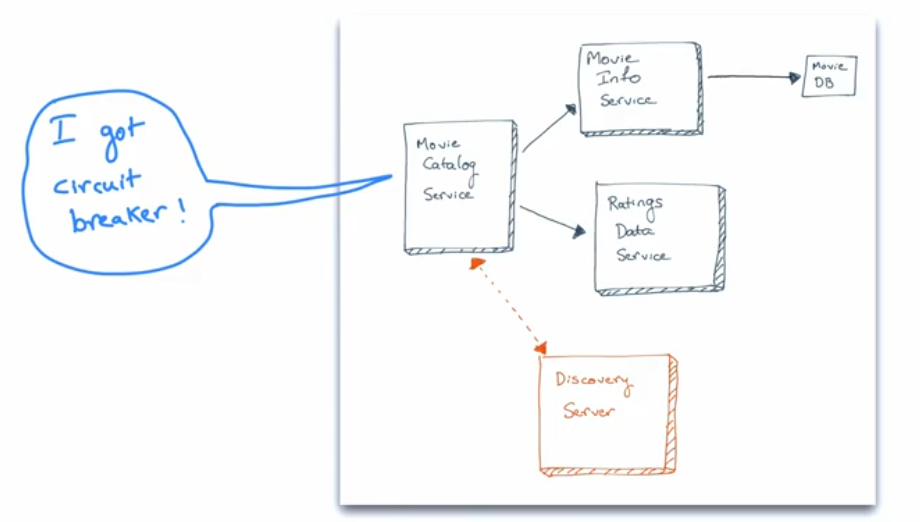
Ans: for bank application not recommended of cache

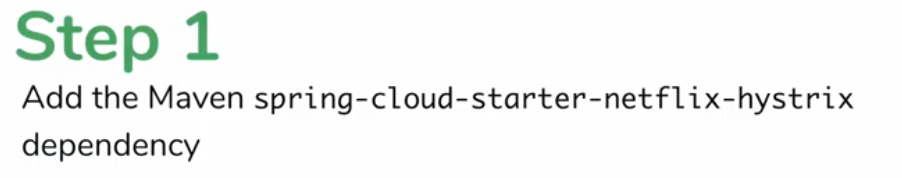
Hystrix

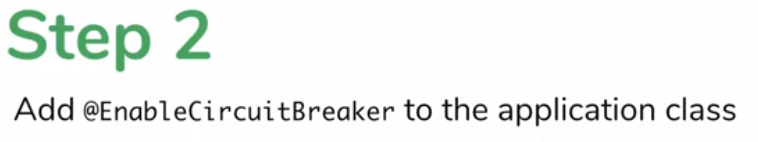


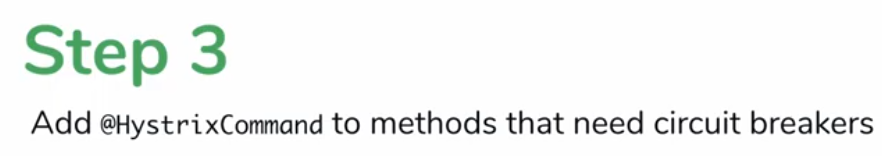
- only drawback is that currnetly Hystrix is not in develpoment mode its in mainatance delopement is stopped bcz it meet all his goals

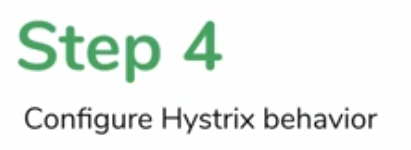












- set parameter

- look at fall back mechanism what happen when crcuit breaks

Config fall back method

Need to mention this method should not cause effect if it calling it should call this callback method to handle that if circuit break happens dnt hit the main method call the call back method

@HystrixCommand(fallbackMethod = "getFallbackCatalog")

<http://localhost:8081/catalog/rest/sachin>

Above Server working with fine data now fallback method not called since movie info service was up

- now make movie info service down and c if fallback methods calls or not

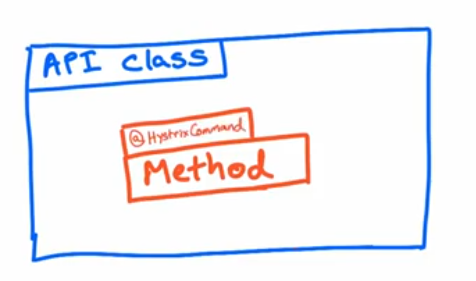
Once down we get fallback method response

How does hystrix work

- any of the one service fails in movie info or movie rating fails hystrix automatically calls fallback service

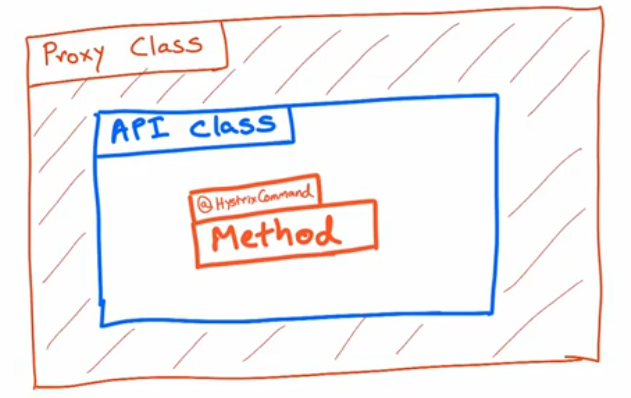
How does this work

Using proxy



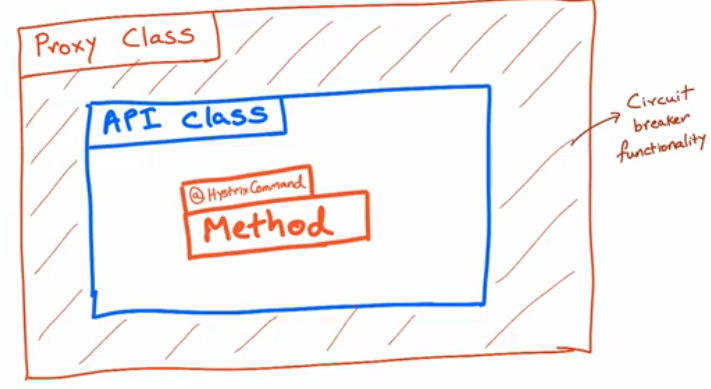
We have an API Class with method which need circuit break method will be annotated with @HystrixCommand and Fall back mechanism to mention fallback method

Hystrix wraps our class around proxy class



- when somebody asks give me instance of that call they dnt get instance of class they get instance of proxy class created by hystrix

- proxy class will have circuit breaking functionality



Circuit break is not quite we want

One method calling 2 API’s if any one service fails we will break circuit

Catalog service retrun zero if movie info is down but rating is fine

Rating is sending valid data but still catalog failed send valid data bcz of movie info service

We get Falllback method if any one api is down so need to fix it

Solution is separate method for calling Rating API and its call back method and separate method calling movie info and callback method

Use Extract method option create a method that Api’s

Now I made separate method seprate fallback method and all still we get error like below if one instance is down

No instances available for movie-info-service

java.lang.IllegalStateException: No instances available for movie-info-service

Guess y?

Bcz of proxy class

- proxy class is an wrapper around the instance of API class

When we have a spring bean/instance(@RestController spring bean) with hystrix commands what hystrix will do it will wrap it around a proxy class so that who ever holding on to the instance they are holding on to the instance of proxy not to the instance of api class

When they call what they think is an api class but actully its an proxy instance which as controll has the ability to intercpet the request call and do things

Prev we had one fall back method it was called by springframework bcz it had an instance extranal call coming bcz somebody hit the url and we mapped that url to method it think its calling method in class but its calling method in proxy and proxy as the abiltiy to say the service is down and going to call fallback

But in latest changes who is going to call movie info service with hystrix commands its nt an external call

Its not an extranl calls which holds on to proxy actullly its a method of that class

If external service calling hystix works fine

- if same class but method inside the class and call somethings which need circuit breaking then hystrix dnt have opprtunity to intercept two mthds calling hystrx dnt have controller over it

Solution is

By taking out that method into another bean class and having API method call not any mehtod of same class but method in different class/instance

Method is same class hystrix not gonna work

Refactoring the codes

Create 2 service

1 moveinfo api and fallback

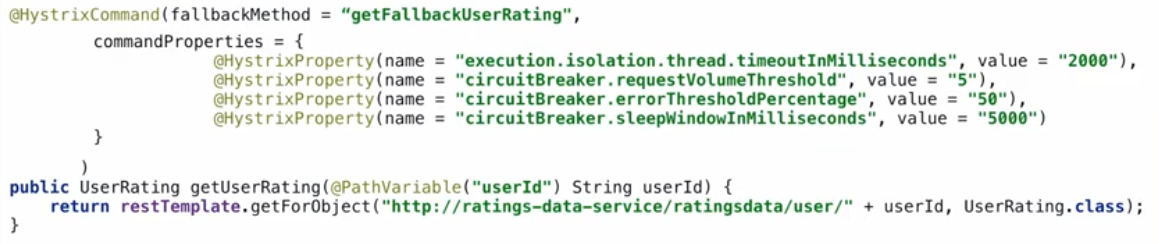
2 movie rating api and fallback

The main call annotated with hystrix commands and fallback should be in same class who ever is calling either the main API or fallback in seprate class

Refactor completed and fixed that issue

To test Time out create a wait in u service to respond after some time so we can tackle time out issues

Configuring Hystrix Parameter



Commnad properties is an array of hystrix properties

Hystrix Dashboard

- another feature provided by hystrix is to dispaly data in dashboard

- to check circuit breaker thier time out many more things can add in all app or in one app

- will add hystrix dashboard to movie catalog

Step 1 add dependencies

<dependency>

<groupId>org.springframework.cloud</groupId>

<artifactId>spring-cloud-starter-netflix-hystrix-dashboard</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-actuator</artifactId>

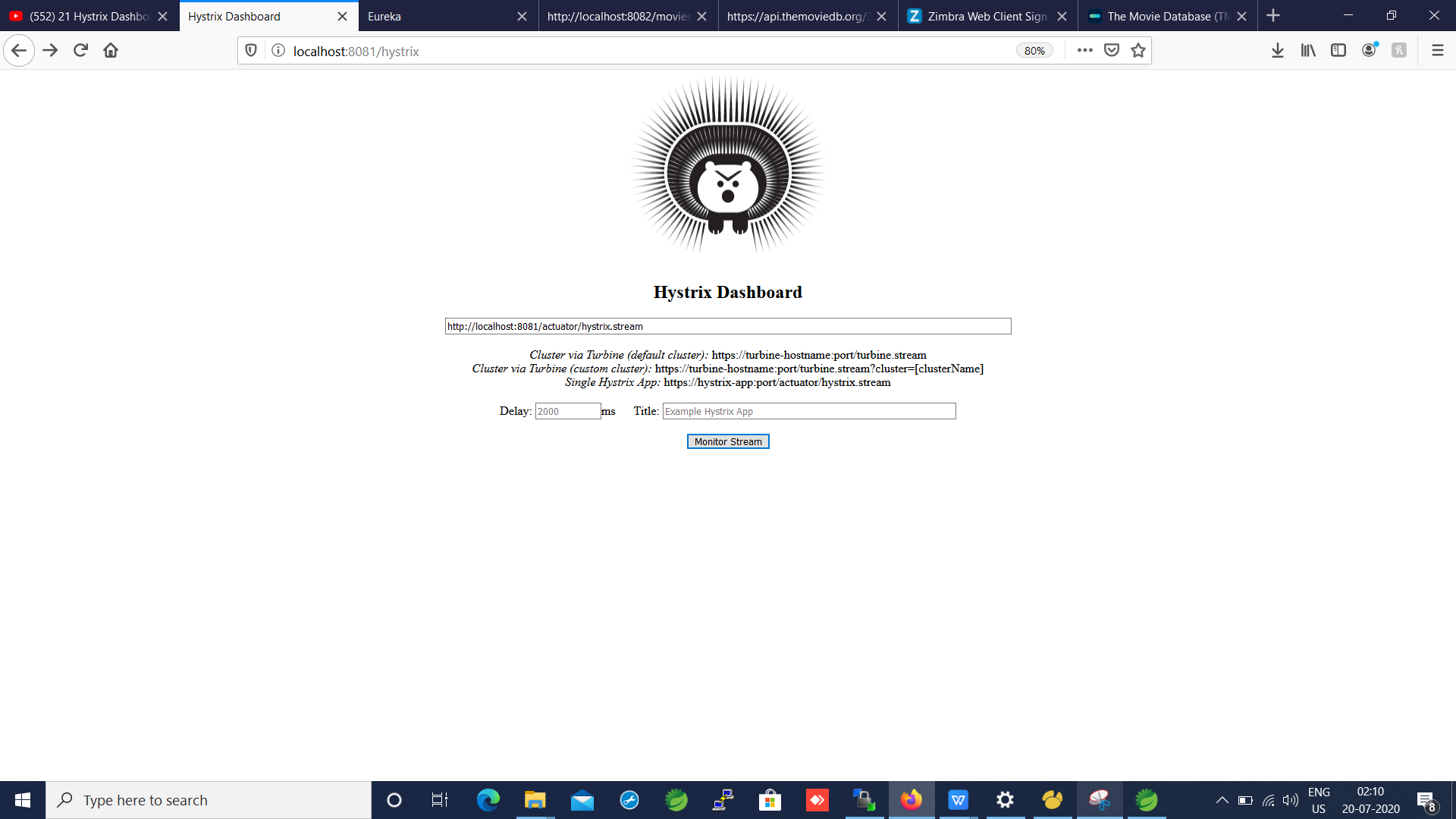
</dependency>

Step 2 add @EnableHystixDashboard in appliaction class

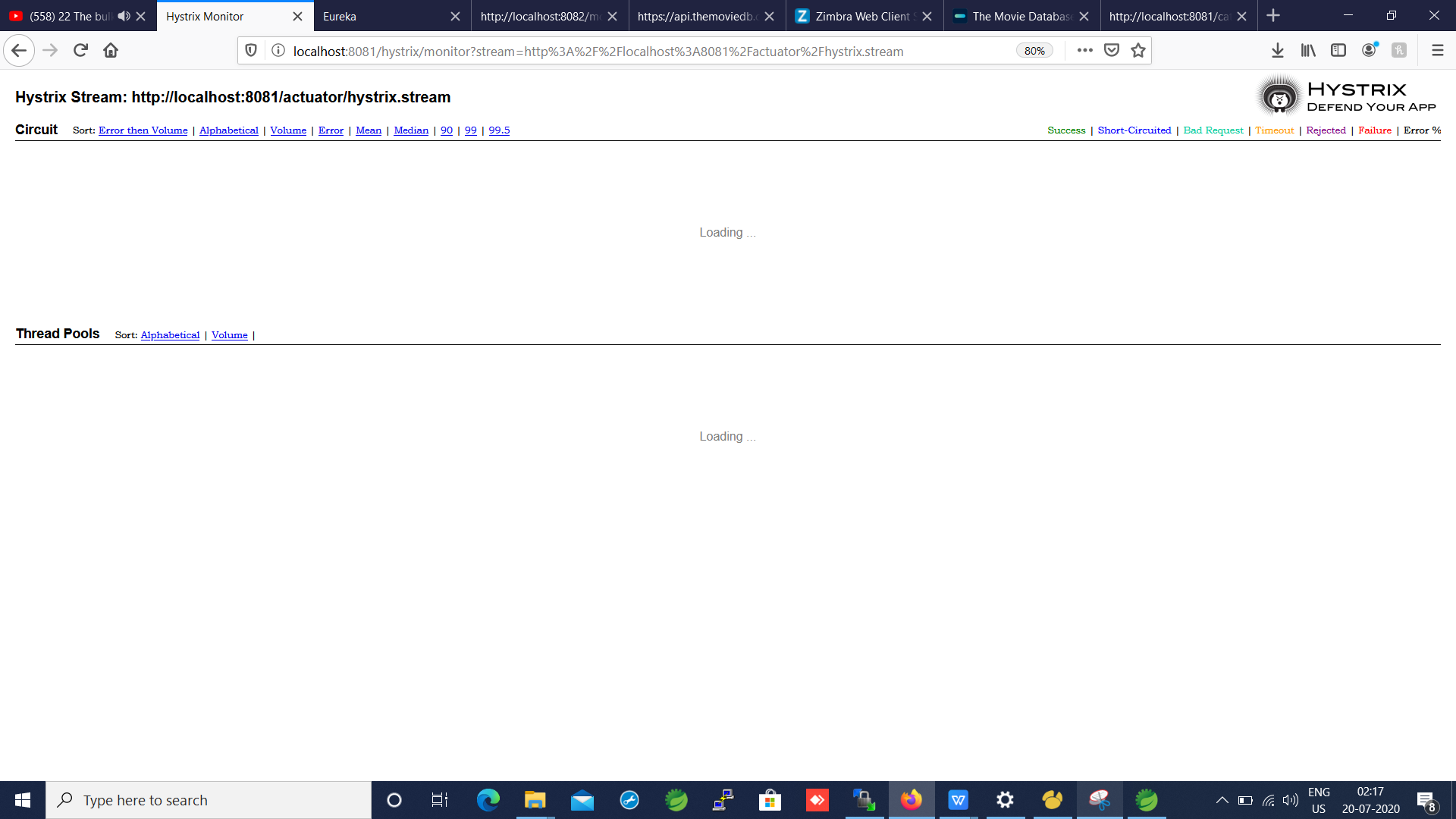
@EnableHystrixDashboard

Step 3 add hystrix endpoints in properties file with out that it wont work

management.endpoints.web.exposure.include=hystrix.stream



Enter url sinlge applicaion or clustor



Bulk Head Pattern

its a 3 rd way to handle outage like this

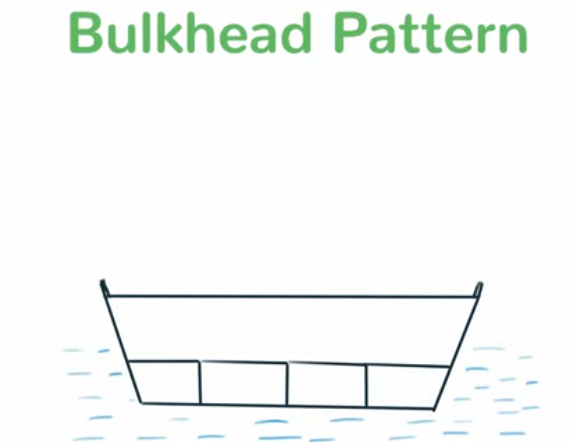
1 st naive way create more instance

2 circuit breaker smarter

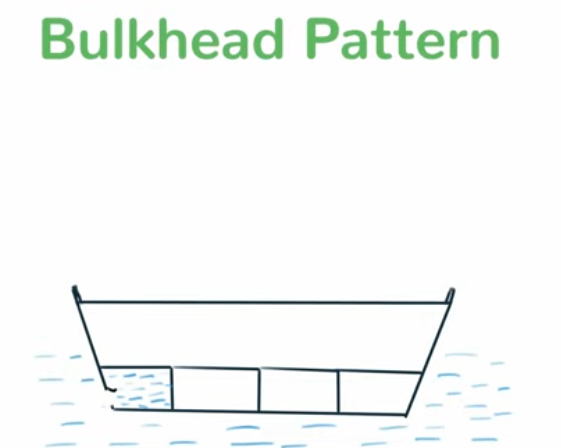
3 use pattern called bulk head pattern

Concept like ship building

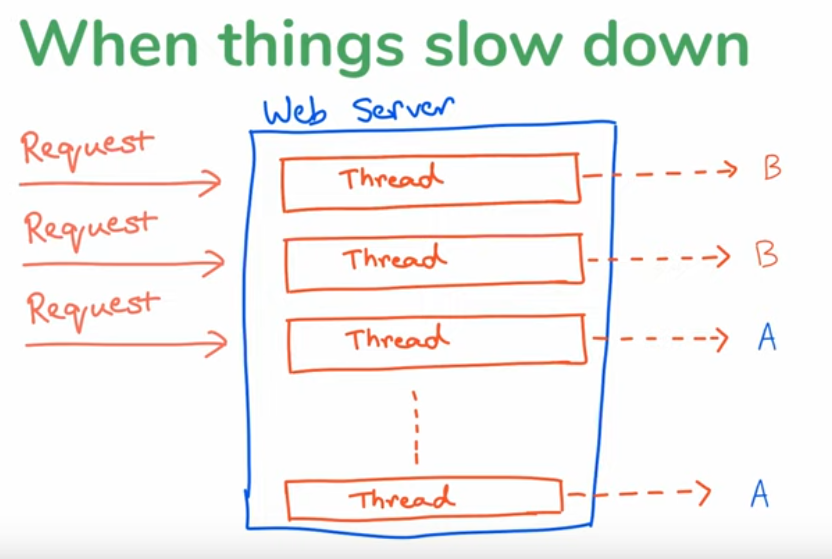
Ship sink if there is a hole in it preven failurs of ship



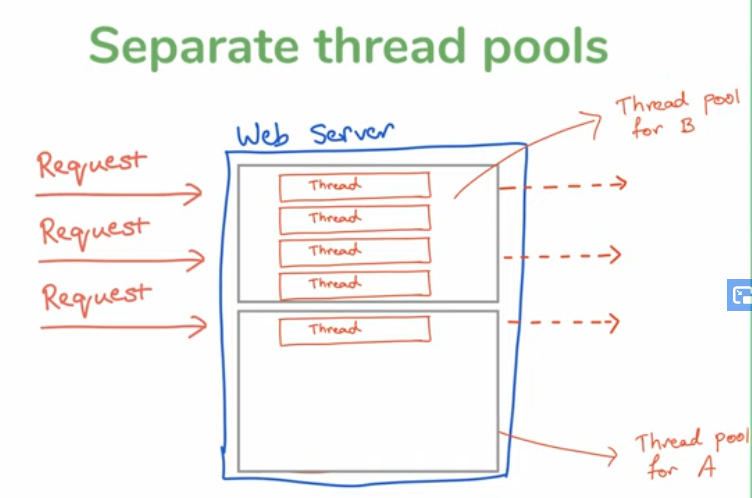
Create a bukets to strore water in hole palce



Use it microservice n threads



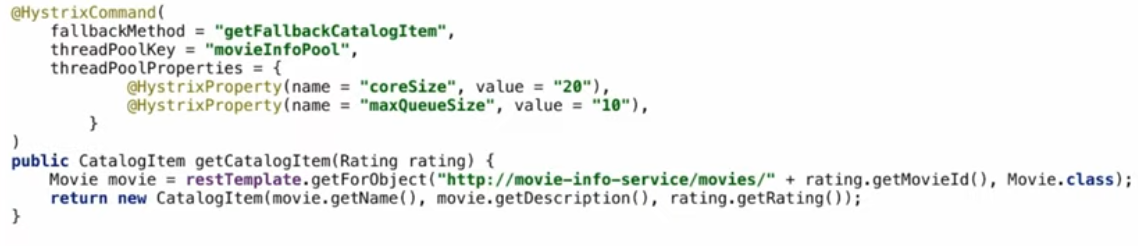
Similary what if we create a seprate compartment for water type B and water type A bcz since bcz of B beibg slow make thread unavailable to A make it slow so make it seprate containers



No circuit breaker just seprate thread pools

Max separtet thread pool fr A and max seprate thread pool for B

How to config



- threadPoolKey -> creating seprate space separte bulk head

It is imp so multiple req use same thread pool are indviudal req only share this

Set therad pool size above 20 threads

Make queue size to wait before tigger

Beyond this it will fallback method

