



# LoopBack

Write Scalable and Extensible  
Node.js Applications using  
LoopBack 4



# LoopBack Toronto Team



**Diana Lau**

[dhmlau@ca.ibm.com](mailto:dhmlau@ca.ibm.com)

 @dhmlau



**Janny Hou**

[juehuo@ca.ibm.com](mailto:juehuo@ca.ibm.com)

 @jannyHou



**Dominique Emond**

[dremond@ca.ibm.com](mailto:dremond@ca.ibm.com)

 @emonddr



**Agnes Lin**

[agneslin.lin@ibm.com](mailto:agneslin.lin@ibm.com)

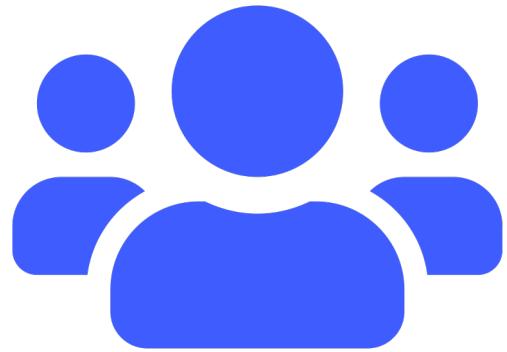
 @agnes512

# Let's Run a Few Polls

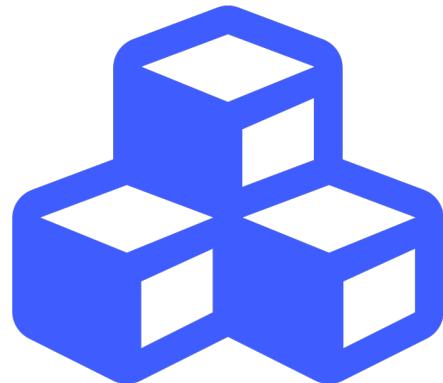


- Do you work on Node.js applications, frameworks, or modules?
- Do you think your Node.js project is large scale?
- What are the characteristics of a large scale Node.js project?

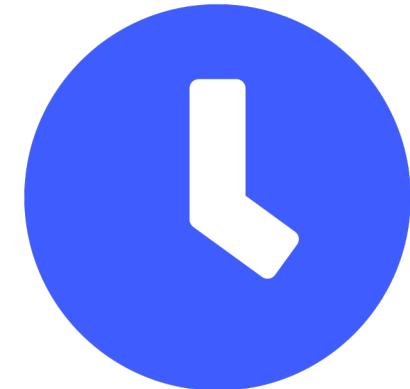
# A Large Scale Node.js Application



# of Teams,  
Developers, Users



# of NPM Packages,  
GitHub Repos

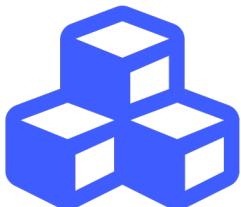


# of Years & Releases of  
Development and Maintenance

# LoopBack – a large scale Node.js framework



12,000+ GitHub Stars



50+ NPM Modules



150,000+ Monthly Downloads



8 Full-time Maintainers  
+ Many Community Contributors



Created 5+ Years Ago



2 Major Releases (v3 and v4)

**BEST IN  
API MIDDLEWARE**

**PRESENTED TO  
LoopBack**

**COMPANY  
IBM**

THE 2019  
**API AWARDS**



presented at

**{API:WORLD™}**

OCT 8-10, 2019 | SAN JOSE, CA

# Challenges We Addressed



Customize the behavior of certain features



Extend the capabilities of your framework/application



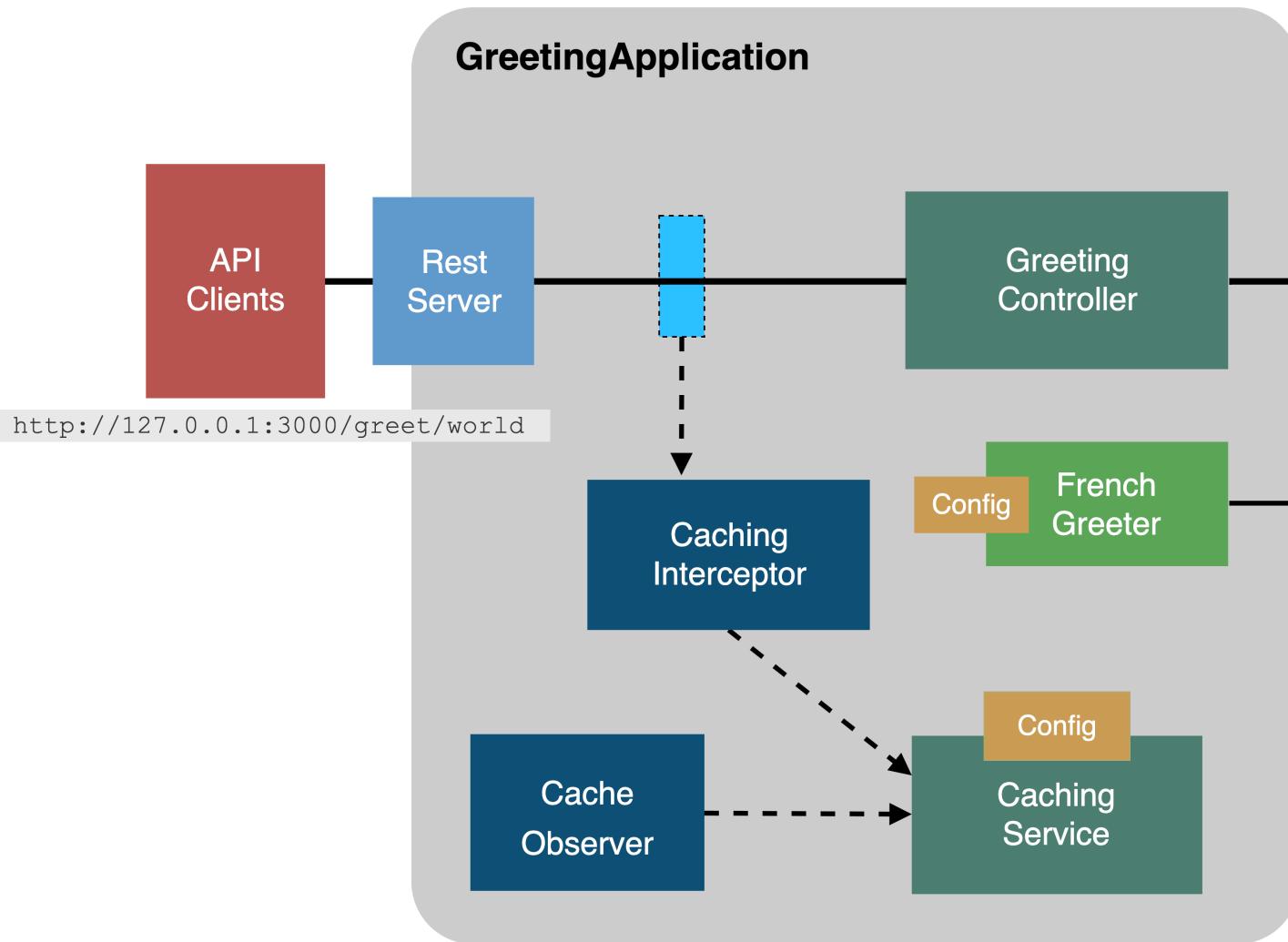
Decompose a sequence of processing flow into actions



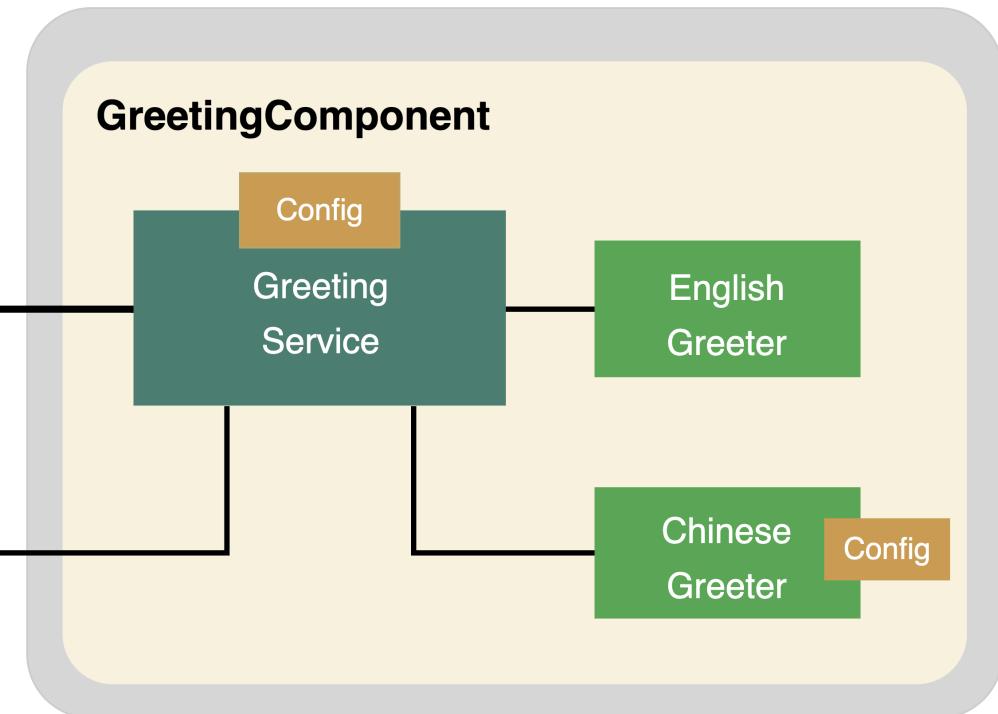
Compose a set of actions into a meaningful flow/sequence

# Application Scenario

**greeting-app:**



**greeter-extension:**



# Part 0

## Before We Get Started

- + Install Node.js version 8.9 or higher

- + Set up development environment

- + Install LoopBack 4 CLI

```
$ npm i -g @loopback/cli
```

- + Clone our git repository

```
$ git clone https://github.com/strongloop/cascon2019.git
$ cd cascon2019
```

>> The code you need for the workshop are all in the instruction:

<https://github.com/strongloop/cascon2019>

# Part 1

Enter the following command to start get started:

```
$ git checkout master  
$ cd greeting-app  
$ npm i
```

# Part 1

## Simple Application

---

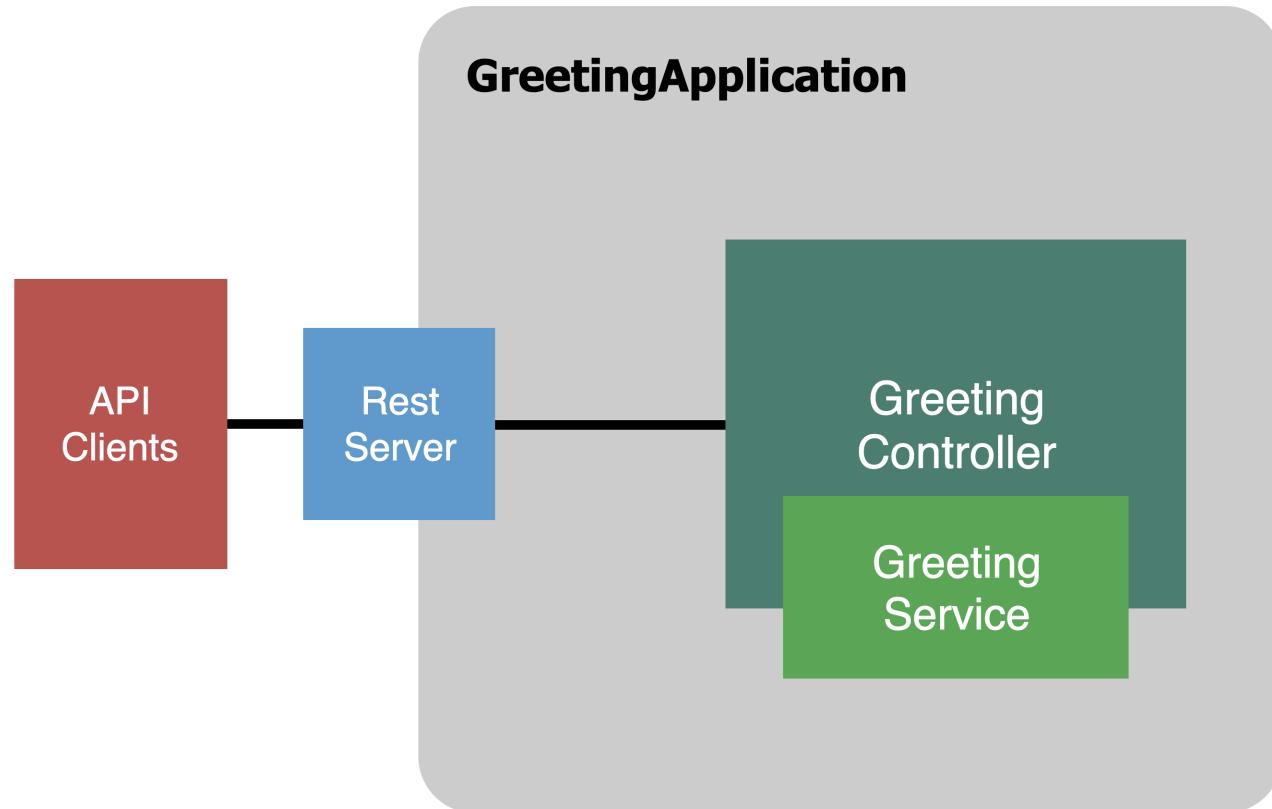


Simple



One Module

**greeting-app:**



# Part 2

If you haven't finished the previous step,  
enter the following commands to start on next part

```
$ git stash  
$ git checkout workshop-part1-completed  
$ npm i
```

# Part 2

## Decouple the Application

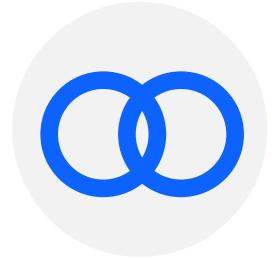
---



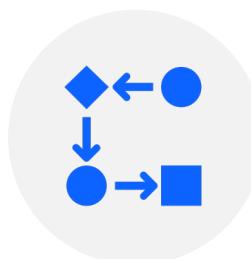
Component



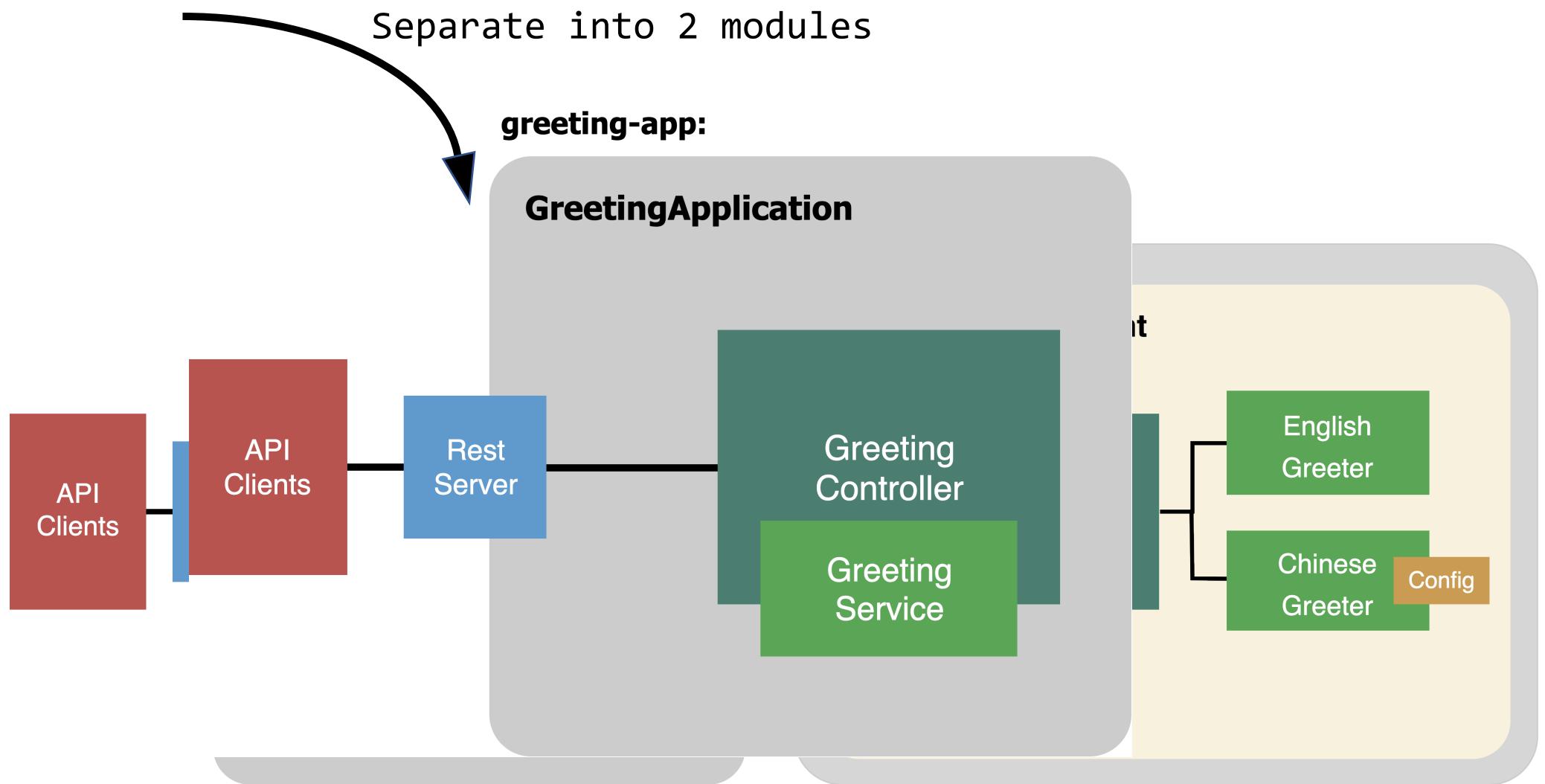
Dependency Injection

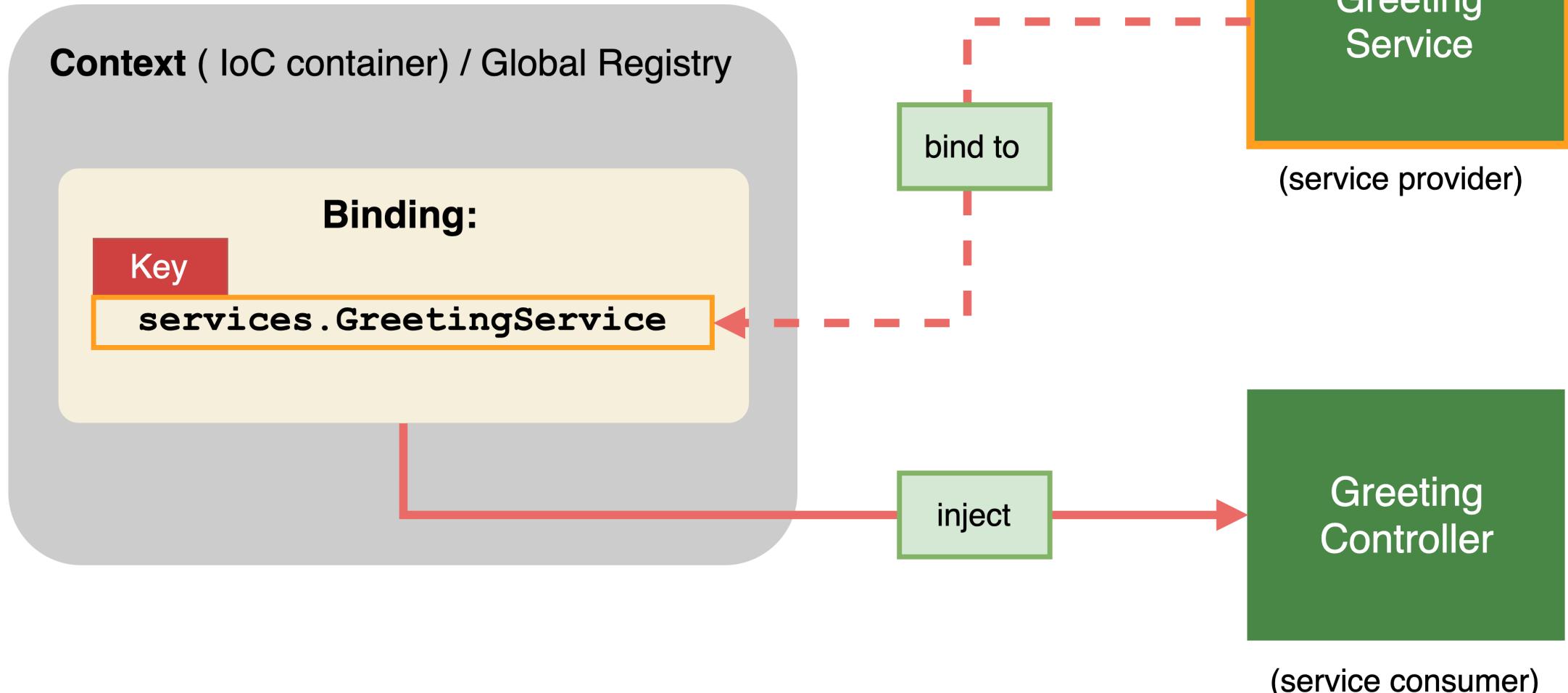


Binding

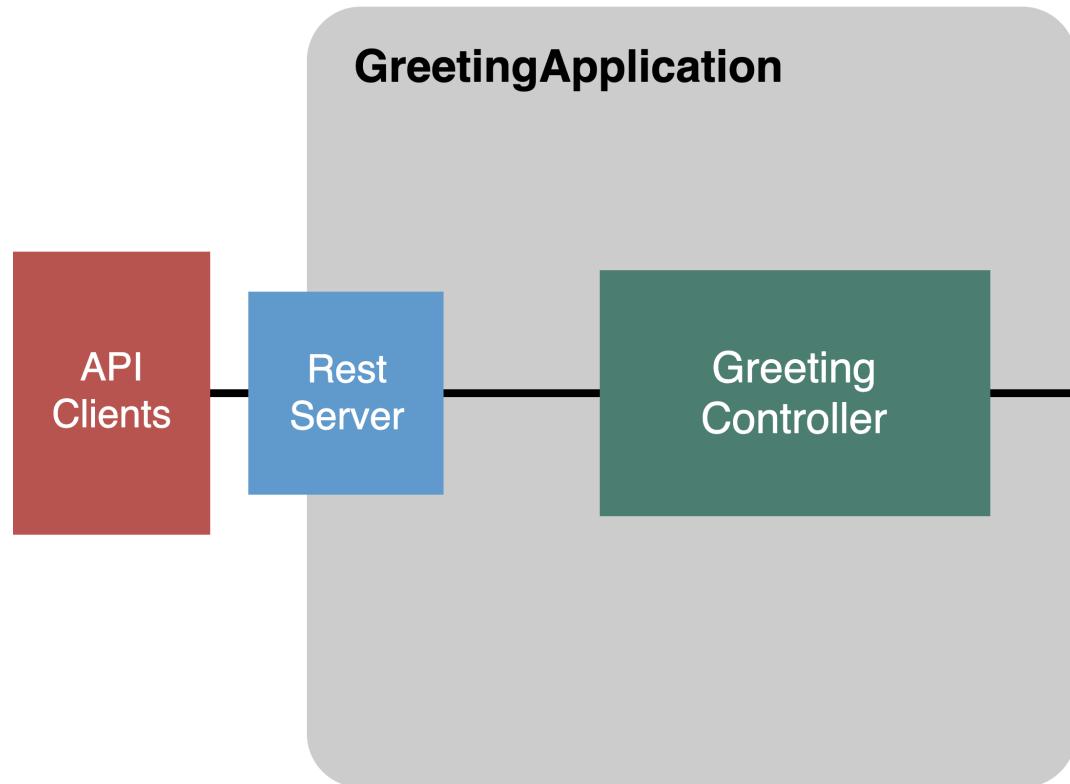


Inversion of Control

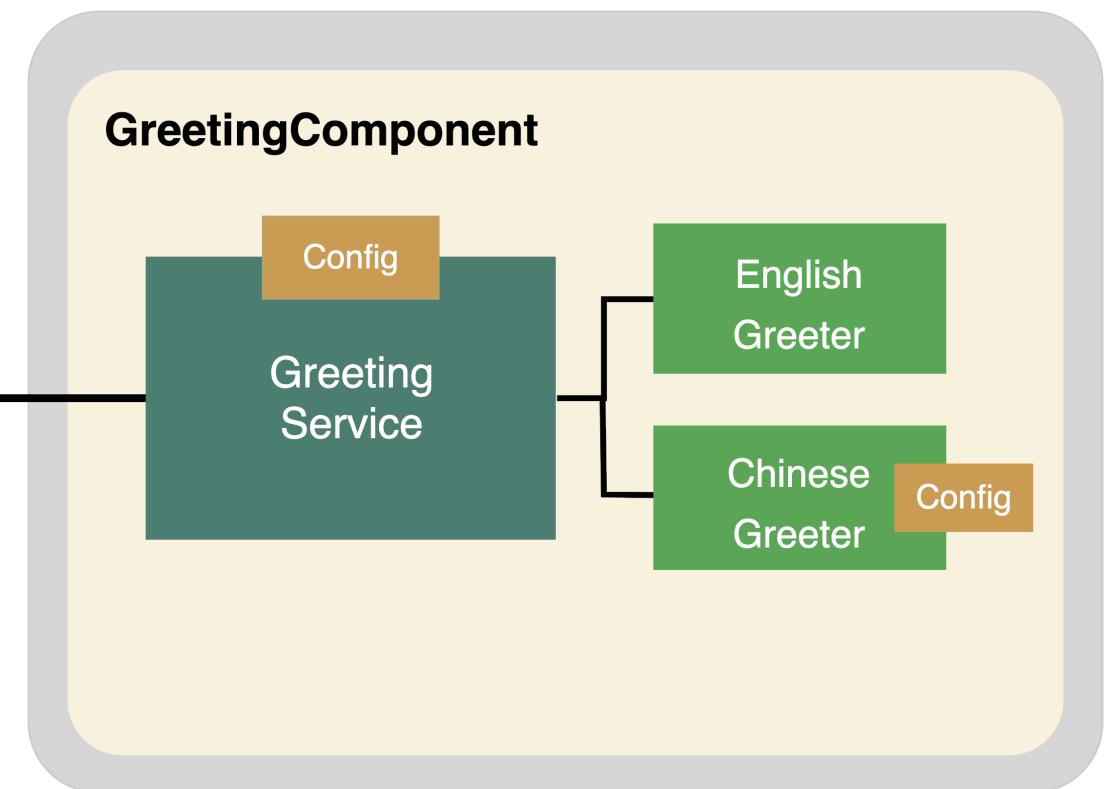




**greeting-app:**



**greeter-extension:**



# Part 3

If you haven't finished the previous step,  
enter the following commands to start on next part

```
$ git stash  
$ git checkout workshop-part2-completed  
$ npm i
```

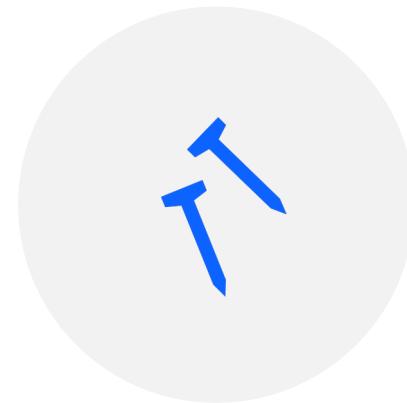
# Part 3.1

## Add a New Language Extension

---

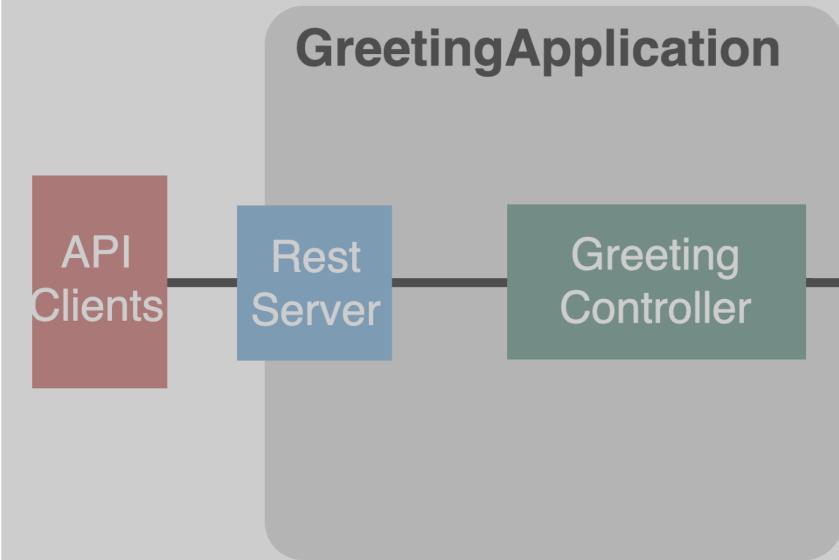


Extension Point

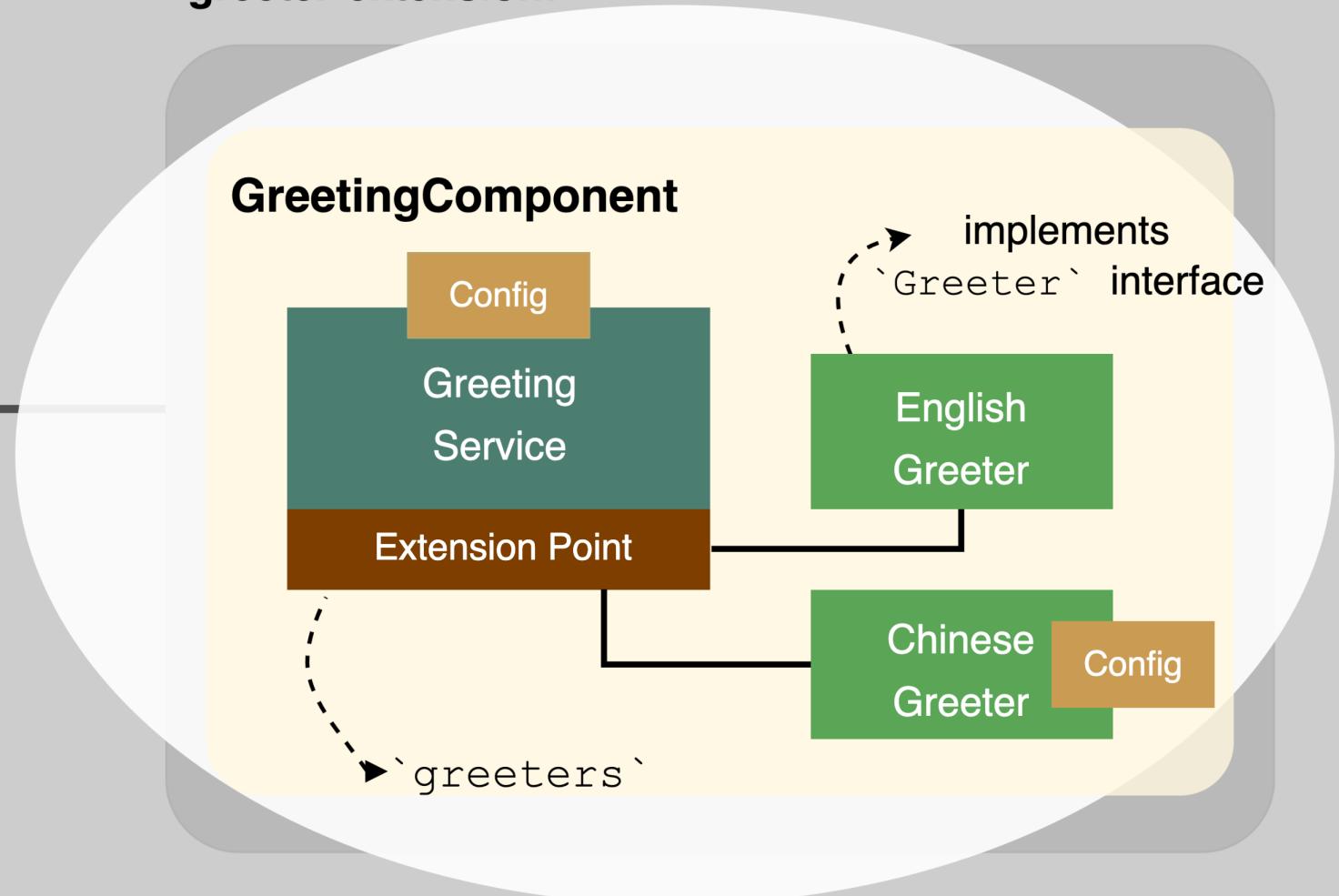


Extension

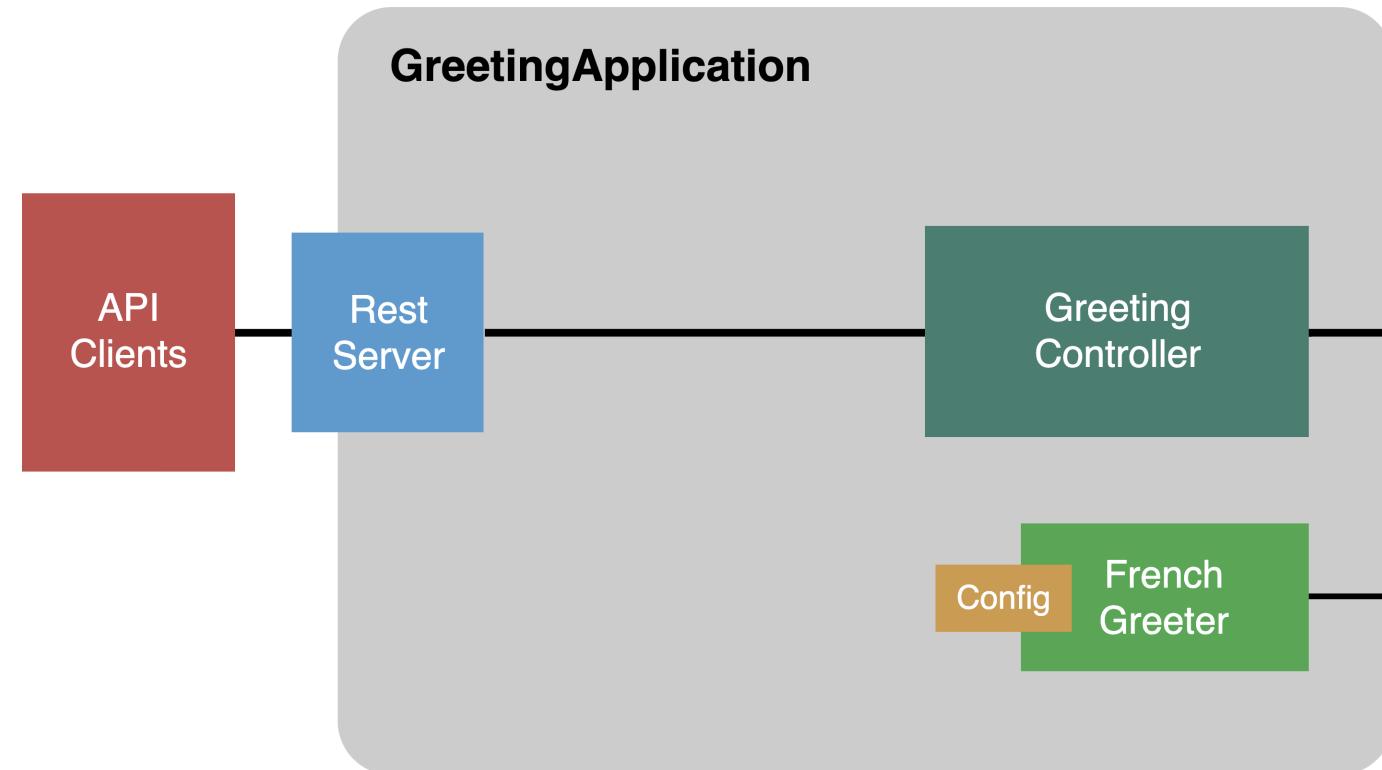
## greeting-app:



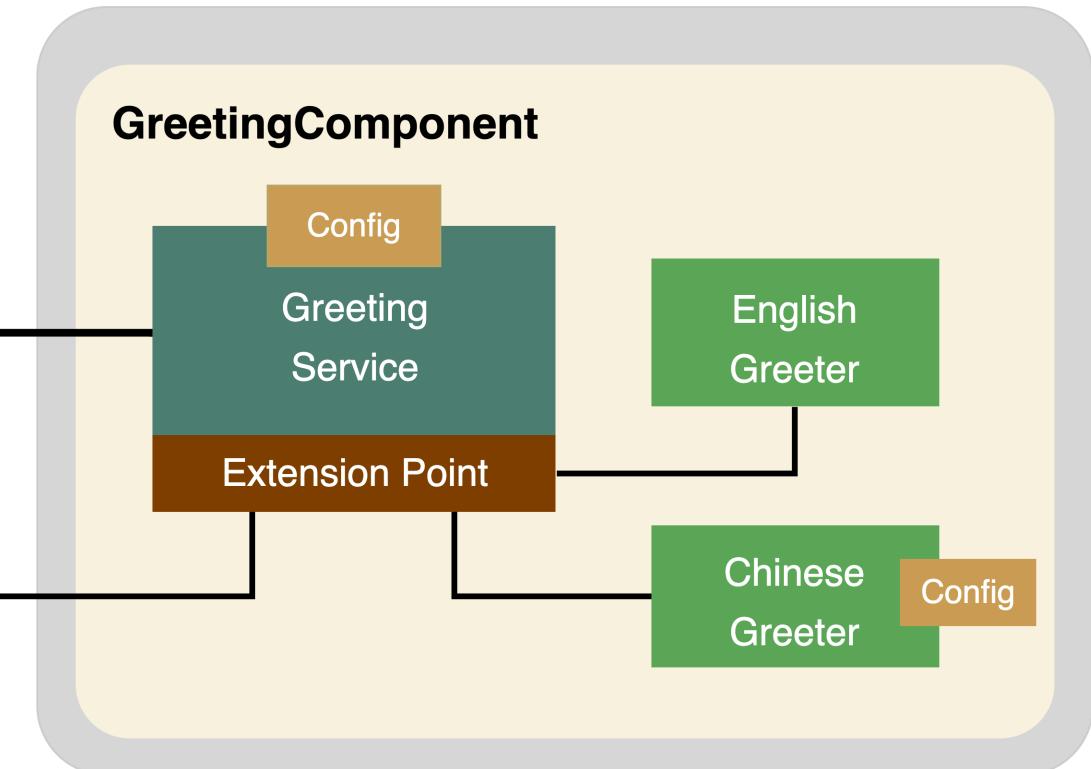
## greeter-extension:



**greeting-app:**



**greeter-extension:**



# Part 3.2

## Configuration

---



Configuration

**greeting-app:**

### GreetingApplication

API  
Clients

Rest  
Server

Greeting  
Controller

Config  
French  
Greeter

**greeter-extension:**

### GreetingComponent

Config

Greeting  
Service

Extension Point

English  
Greeter

Chinese  
Greeter

Config

# Part 4

If you haven't finished the previous step,  
enter the following commands to start on next part

```
$ git stash  
$ git checkout workshop-part3-completed  
$ npm i
```

# Part 4

## Enable Caching

---



Interceptor

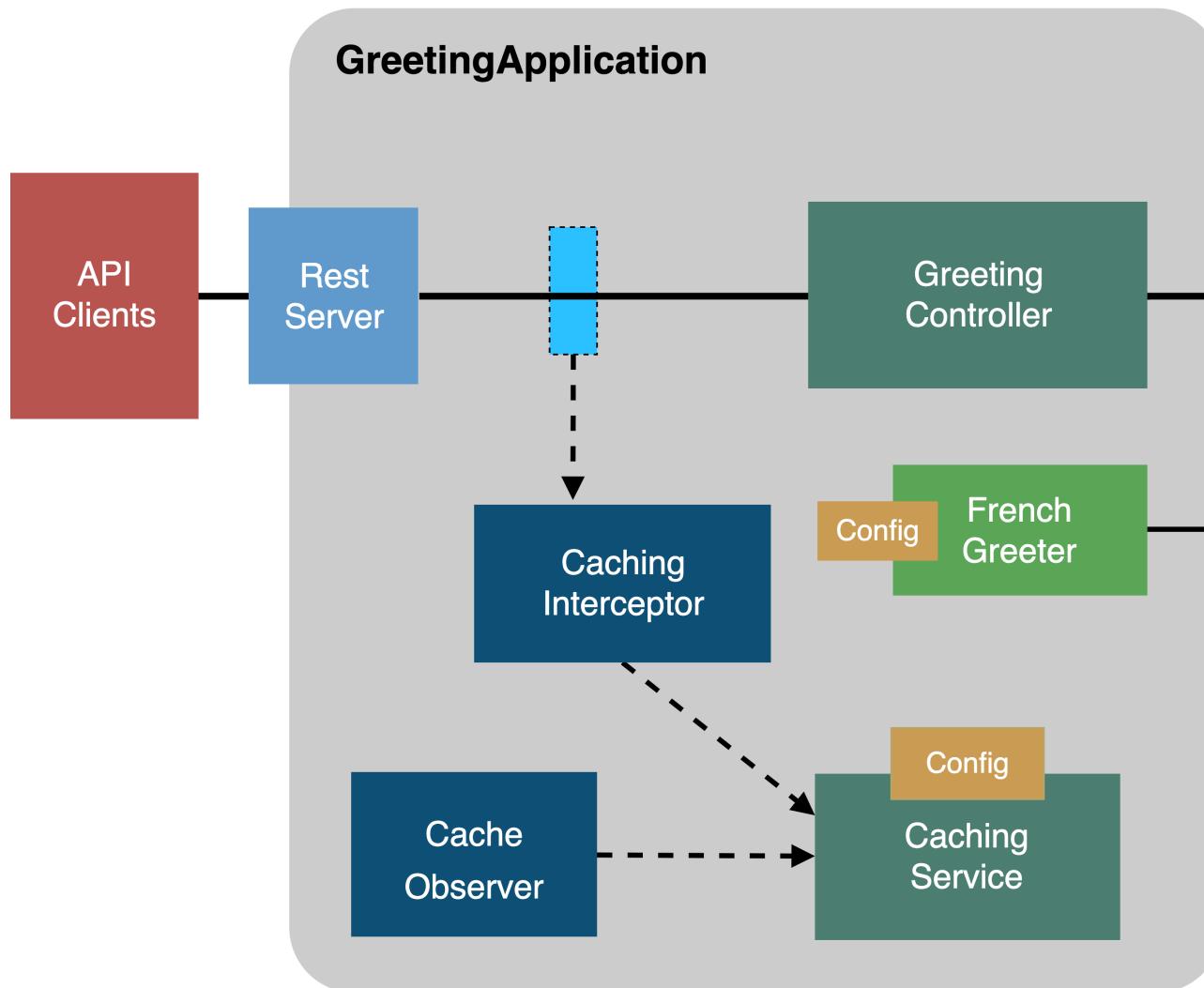


Observer

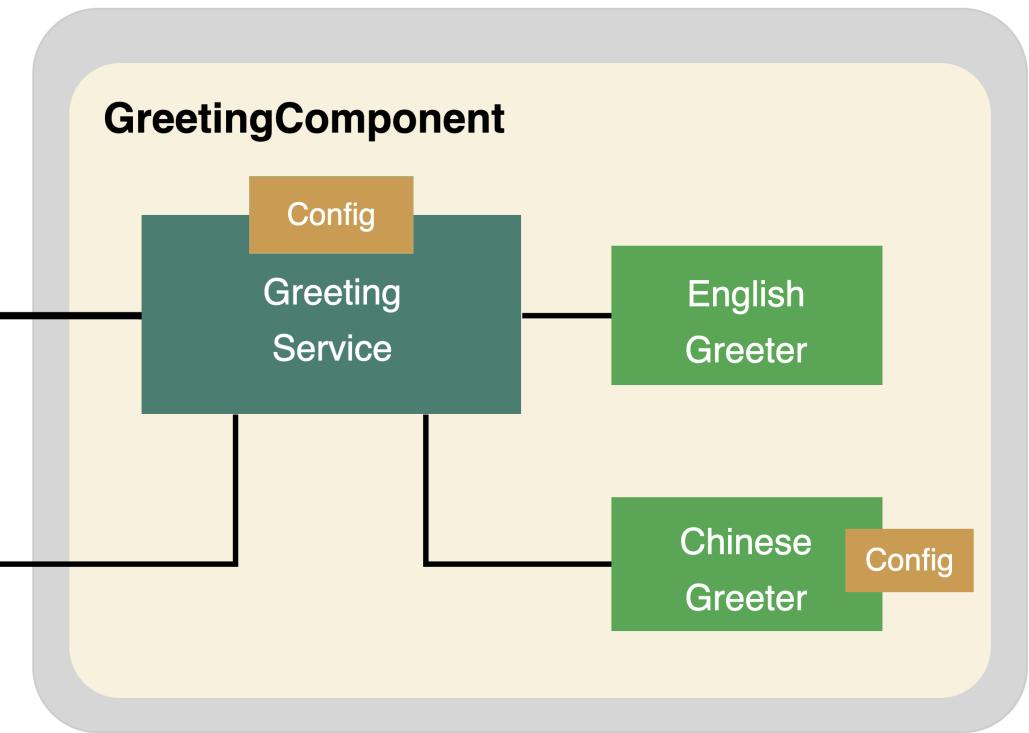
# Let's Build a Caching System That Does:

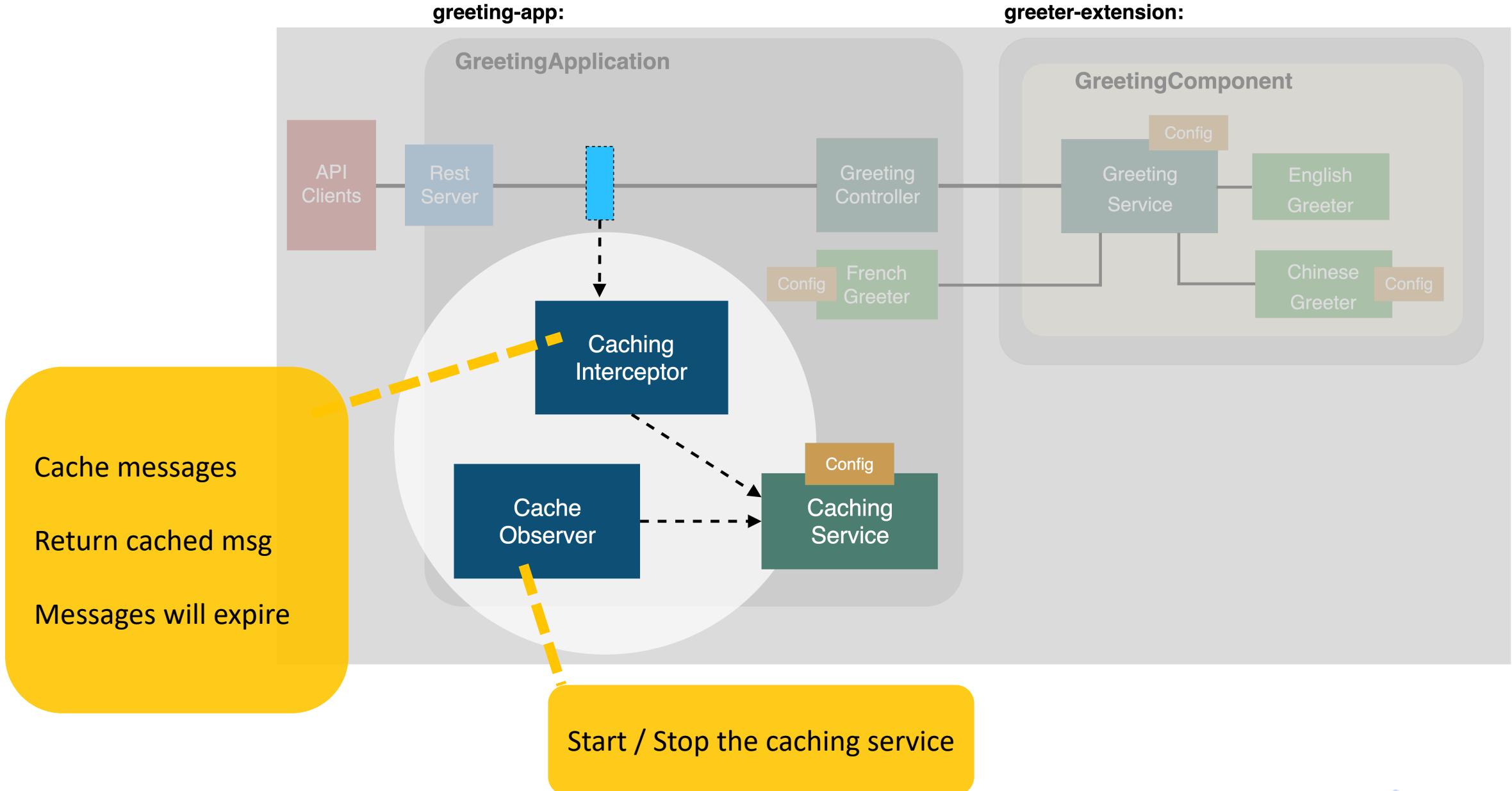
- + Cache the result using the request's URI and language as the key
- + The cached message has an expiration time
- + Expired messages will be swept out periodically

**greeting-app:**

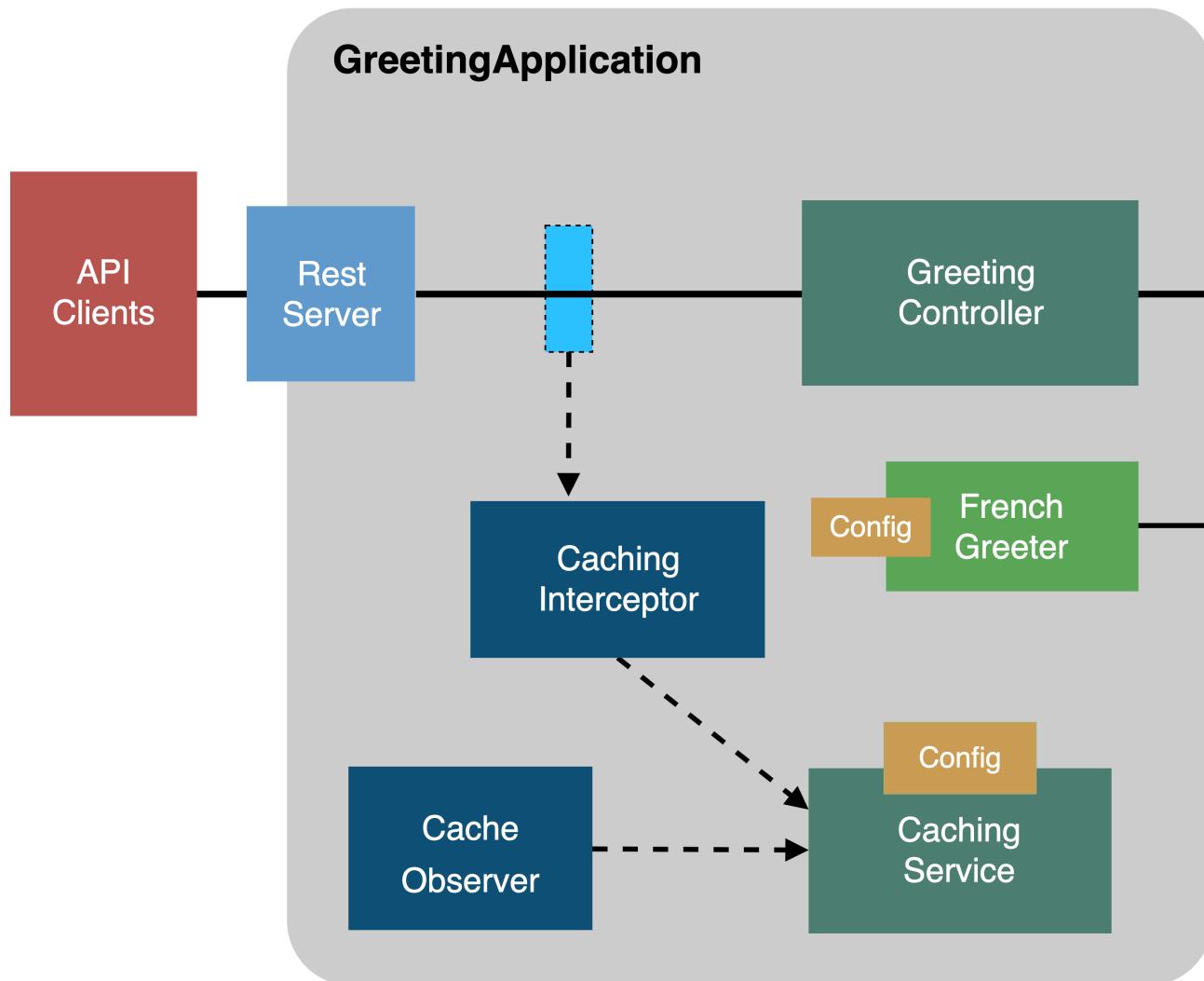


**greeter-extension:**

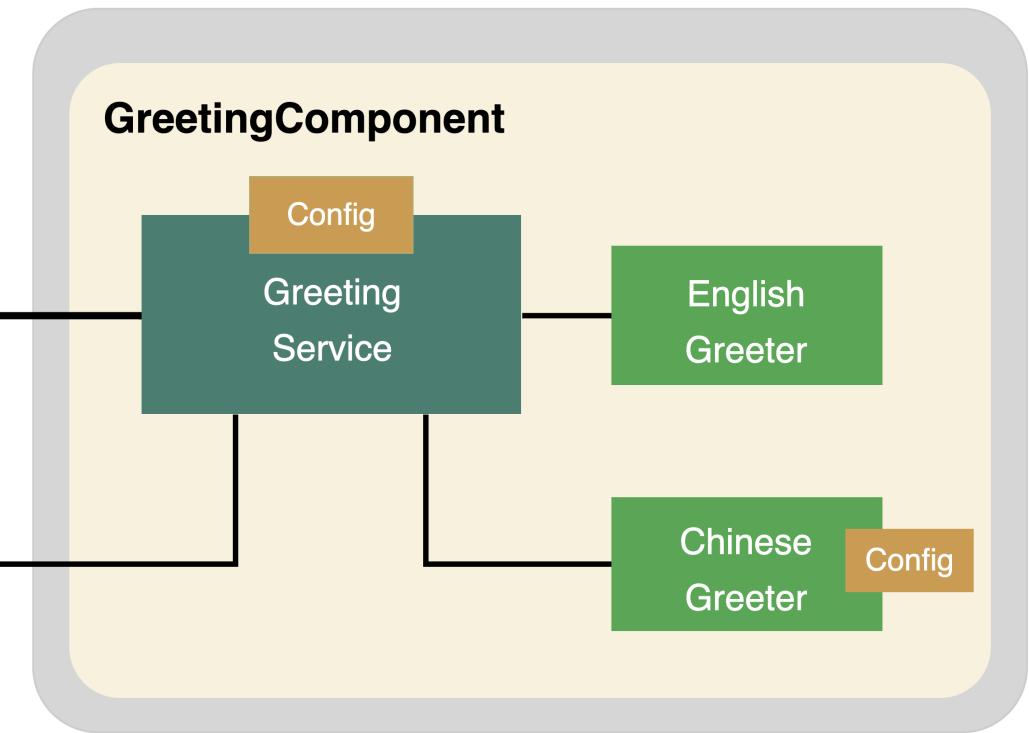




**greeting-app:**

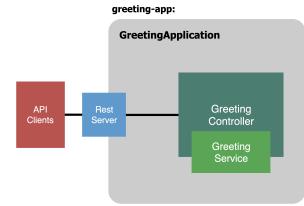


**greeter-extension:**



# Congratulations! You've completed the workshop

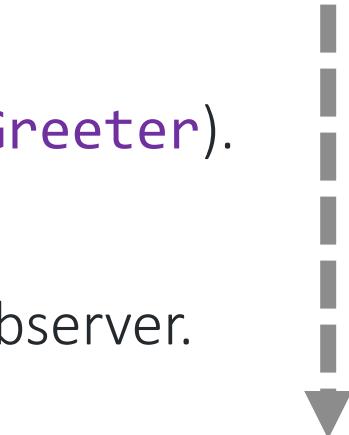
★ Built a LoopBack application and separated it into two modules.



★★ Enriched the functionality of the app by adding extensions (**FrenchGreeter**).

★★★ Created a caching service that is started and stopped by a lifecycle observer.

★★★★ Retrieved/stored request responses from/in the cache with an interceptor.



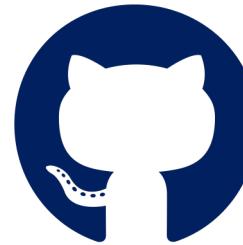
You can find the completed code of this workshop in the git branch:

```
$ git checkout workshop-part4-completed
```

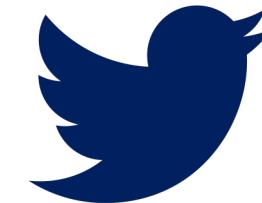
# Thank You!



[loopback.io](http://loopback.io)



[Strongloop/loopback-next](https://github.com/Strongloop/loopback-next)



[@StrongLoop](https://twitter.com/StrongLoop)

# Acknowledgement

- The LoopBack team at IBM and our contributors from the community making good things happen in open source
- Taranveer Virk and Raymond Feng - allowing us to reuse some of their slides
- Iconography by [FontAwesome](#) is licensed under [CC BY 4.0](#)