pulsar_function

C-2 创建:侯伯楠, 最后修改: 侯伯楠 刚刚

目录

Overview
Goals
Architecture
TopicManager
FunctionExecutor
DataFlow
Suggestions
Links项目地址

Overview

In this wiki, I will describe the detailed design of the task and show completion.

Goals

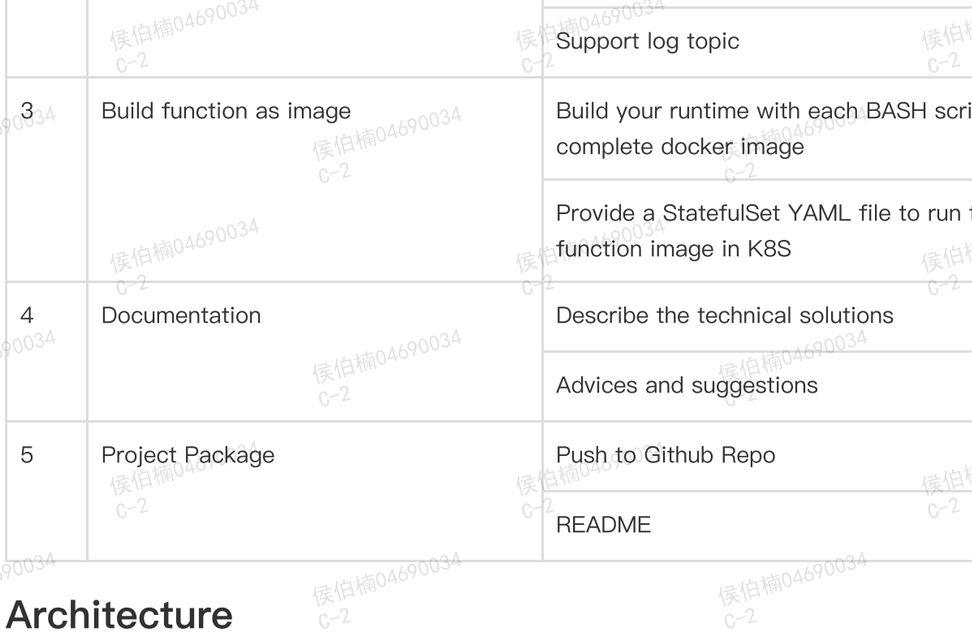
Here is the completion of tasks

#	goal	subgoal	state	progress	remark
1	Define serval BASH scripts as functions	Reverse Function: reverses the input message	DONE	100%	
2	Build a BASH runtime	Customize the input and output Pulsar topic	DONE	100%	
		Invoke the target BASH script	DONE		
		Parallel processing	DONE		
		Support log topic	DONE		
3	Build function as image	Build your runtime with each BASH script as a complete docker image	DONE	50%	
		Provide a StatefulSet YAML file to run the function image in K8S	UNDO		Maybe I need more time.
4	Documentation	Describe the technical solutions	DONE	100%	
		Advices and suggestions	DONE	100%	
5	Project Package	Push to Github Repo	DONE	100%	
		README	DONE	100%	

Architecture

The architecture disgarm if the project is shown below:

- The green parts are modules of pulsar function
- The red arrow is the direction of control surface
- The black arrow is the direction of the data plane



TopicManager

TopicManager is responsible for interacting with Pulsar, like a proxy between the user and Pulsar. The producerManager maintains theproducer instance and invokes the corresponding the methods. The consumerManager is in charge of subscription and listens messages which is produced to inputTopic, finally the messages are submitted to functionExecutor.

FunctionExecutor

FunctionExecutor is responsible for executing the bash script. When inputSubscription receives a message, it immediately creates a scriptTask and submits it to the thread pool for execution.

DataFlow

A complete data flow is show below:

1. Firstly, bash-runtime init with custom topics.
    a. User provides custom input&output topic to Pulsar function.
    b. ClientManager creates topics and listens subscriptions by input.
2. processing

a. User produce a message by Pulsar-client.

b. ClientManager receives the message.

c. ClientManager creates a scriptTask by this message and submits to thread pool.

d. scriptTask reput the result to output topic by ClientManager.

3. User receives result from output subscription.
4. Key log output to the log topic for troubleshooting and analysis.

Suggestions

The format of the interview task was unlike anything I had ever experienced before. I think it can test a lot of things, such as documentation ability, learning ability, basic knowledge and so on. It has many well-known advantages, so I'm just going to make suggestions here.

I think more time is needed for the interviewee. Although it isn't a difficult task, but it requires documentation, coding, and learning some unknown knowledge. It is not just known pulsar, I also need to learn Docker and K8s because I seldom use them. So maybe I need more time.

Links项目地址

https://github.com/danpi/pf_runtime