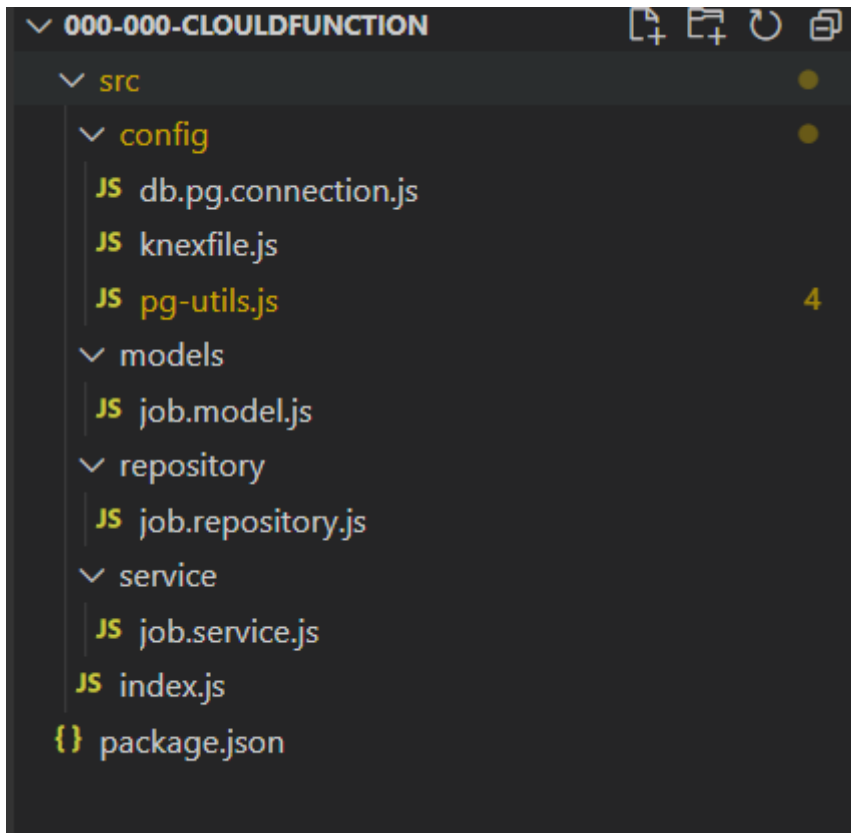# 开发

代码结构

文件

index.js

入口文件

```javascript
const archiveData = require('./job.service')

exports.clouldfunction = async (event) => {
    console.log("event", event);
    let pubsubMessage = event.data;
    let body = Buffer.from(pubsubMessage, 'base64').toString();
    if (!!body) {
        body = JSON.parse(body);
        console.log("body", JSON.stringify(body));

        await archiveData(body);
    } else {
        console.log('No event data.');
    }
}
```

PubSub

接收

> 触发CF。配置订阅的Topic，tipic被触发时将执行该CF。

```
let pubsubMessage = event.data;
let body = Buffer.from(pubsubMessage, 'base64').toString();
if (!!body) {
    body = JSON.parse(body);
}
```

发送

> 该topic被触发（app_27354_npd-archive-completion）订阅该topic的所有CF都将执行。

```
const publishMessage = async (eventObj, bool) => {
    const msgObj = {
        ContractInventoryKey: eventObj.ContractInventoryKey,
        ContractNbr: eventObj.ContractNbr,
        ModuleNm: 'CS',
        Result: bool ? 'Completed' : 'Failed'
    }
    console.log('publish Msg body', msgObj);
    try {
        const topicName = 'app_27354_npd-archive-completion';
        const pubsub = new PubSub();
        const dataBuffer = Buffer.from(JSON.stringify(msgObj));
        const messageId = await pubsub.topic(topicName).publish(dataBuffer);
        console.log(`${messageId} - Message published.`);
    } catch (err) {
        console.error("sendMsg error: ", err)
    }
}
```
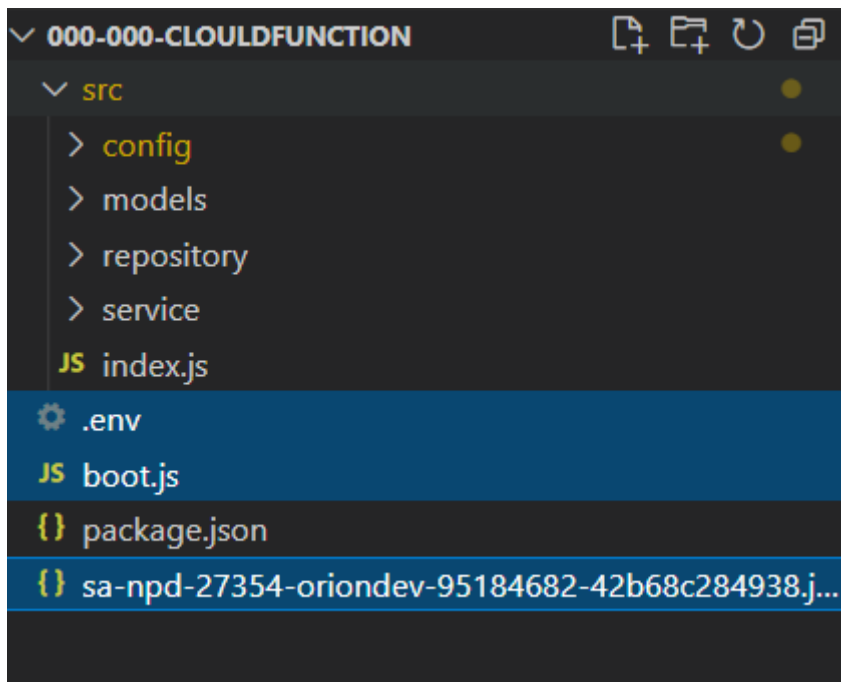
测试

本地

> 本地没有PubSub环境，只能改写code使用HTTP模拟CF被触发。本地无法测试PubSub的发送结果。**调式**程序方法同API项目相同

添加文件

增加三个文件并更改部分code，使CF能够接收HTTP 请求，然后用postman模拟CF被触发。

.env / sa 文件同API项目中相同

boot.js 文件需更改。使code变为可接收HTTP请求的Express App

```
12    server.use(express.json());
13    server.post('/', cf.clouldfunction); // Align with exporting module in index.js
14
```

改写入口Code

index.js event.body 即为 HTTP post请求中的body

```
exports.clouldfunction = async (event) => {
    console.log("event", event);
    let body = event.body;
    if (!!body) {
        await archiveData(body);
    } else {
        console.log('No event data.');
    }
}
```

安装本地测试所需的包

nodemon dotenv express

```
12        "dependencies": {
13          "@google-cloud/pubsub": "2.12.0",
14          "knex": "^0.95.11",
15          "lodash": "^4.17.21",
16          "pg": "^8.7.1",
17          "nodemon": "^2.0.15",
18          "dotenv": "^10.0.0",
19          "express": "^4.17.1"
20        }
21      }
22
```
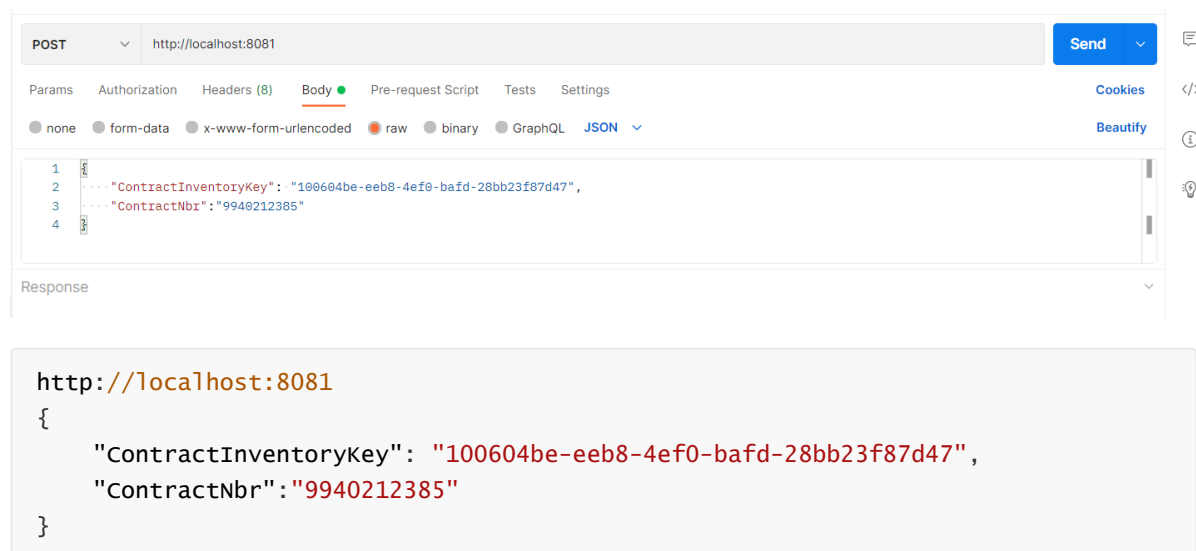
启动程序

npm run watch 通过boot.js为主文件启动程序

```
6  ∨    "scripts": {
7          "test": "echo \"Error: no test specified\" && exit 1",
8          "watch": "nodemon --inspect=9230 boot.js"
9        },
```

Postman 触发CF


```
http://localhost:8081
{
    "ContractInventoryKey": "100604be-eeb8-4ef0-bafd-28bb23f87d47",
    "ContractNbr":"9940212385"
}
```

DEV/STAGE

Pipeline触发CF

参照 Pipeline **Execute Archive Contract Command**

Edit Task

1. 根据想要执行的环境在**Active Account and Envi**的script中找到对应的环境名、SA文件名

Script * ⓘ

```
if  [[ "$(parameters.Environment)" == "STAGE" ]]
then
cp $(Agent.TempDirectory)/sa-npd-27354-orionstage-160dc65a-934f2437d60b-
20190717.json ./stage.json
ls -ail && pwd
gcloud config set verbosity debug

gcloud auth activate-service-account sa-npd-27354-orionstage-160dc6@npd-27354-
orionstage-160dc65a.iam.gserviceaccount.com --key-file=./stage.json --project npd-
27354-orionstage-160dc65a
```

2. 创建对应环境的Task

Pipeline
Build pipeline   ···

🗁 Get sources
   ⋈ NxtGenMMC_27354   ⑂ R22.2_Development

DEV Environment   +
   ▤ Run on agent

[📄 Download secure file - DEV   ✔ ⠿
    Download secure file   ]

📄 Download secure file - TEST   ○ ⠿
    Disabled: Download secure file

Download secure file ⓘ   🔗 Link settings ⟨⟩ View YAML 🗑 Remove

Task version 1.* ⌄

Display name *
[Download secure file - DEV   ]

Secure File * ⓘ
[sa-npd-27354-oriondev-95184682-42b68c284938-Dec132021.json ⌄] ↻ ⚙

3. 修改variables

Tasks **Variables** Triggers Options History   🖫 Save & queue ⌄   ↺ Discard   ☰ Summary   ▷ Queue   ···

Pipeline variables	Name ↑	Value
Variable groups	air_id	27354
Predefined variables ↗	Parameters.Environment	DEV

4. 修改测试所需数据和要触发的Tipic（一个Topic可能被多个CF订阅，运行Pipeline多个CF将被触发）

#! Install Google SDK
   Bash

#! Active Account and Envi
   Bash

[#! Execute Command   ✔ ⠿
    Bash   ]

```
gcloud pubsub topics publish app_27354_npd-archive --message "
{\"ContractInventoryKey\":\"100604be-eeb8-4ef0-bafd-
28bb23f87d47\",\"ContractNbr\":\"6999977701\"}"
```

5. 执行Pipeline

在GCP中查看Log

GCP 中找到对应CF，查看Log信息

← → C 🔒 https://console.cloud.google.com/functions/details/us-east1/npd-27354-archivecontractingstatus?env=gen1&project=npd-27354-oriondev-95184682&supported...

☰ Google Cloud Platform ⚡ npd-27354-oriondev ⌄   🔍 Search Products, resources, docs (/)   ⌄

⦿ Cloud Functions ← Function details   ✎ EDIT 🗑 DELETE 🗐 COPY

● npd-27354-archivecontractingstatus [1st gen]   Version
                                                  Version 10, deployed at Feb 17, 2022, 1:31:07 P... ⌄

METRICS DETAILS SOURCE VARIABLES TRIGGER PERMISSIONS **LOGS** TESTING

Logs Showing 50 log entries   Severity
                                 Default   ⌄   ☰ Filter Filter logs

▶ ☐ 2022-02-28T09:34:52.308Z npd-27354-archivecontractingstatus 5lt0kaduazby body {"ContractInventoryKey":"963d4a3f-6406-4a79-9c76-b243d7b5cd9f","ContractNbr":"9940012814"}
▶ ☐ 2022-02-28T09:34:52.366Z npd-27354-archivecontractingstatus 5lt0kaduazby body CS List: []
▶ ☐ 2022-02-28T09:34:52.367Z npd-27354-archivecontractingstatus 5lt0kaduazby publish Msg body {
▶ ☐ 2022-02-28T09:34:52.367Z npd-27354-archivecontractingstatus 5lt0kaduazby ContractInventoryKey: '963d4a3f-6406-4a79-9c76-b243d7b5cd9f',
▶ ☐ 2022-02-28T09:34:52.367Z npd-27354-archivecontractingstatus 5lt0kaduazby ContractNbr: '9940012814',
▶ ☐ 2022-02-28T09:34:52.367Z npd-27354-archivecontractingstatus 5lt0kaduazby ModuleNm: 'CS',
▶ ☐ 2022-02-28T09:34:52.367Z npd-27354-archivecontractingstatus 5lt0kaduazby Result: 'Completed'

发布

> 通过Azure Pipeline 将Code 发布至GCP中并创建新的CF（自动）

添加文件
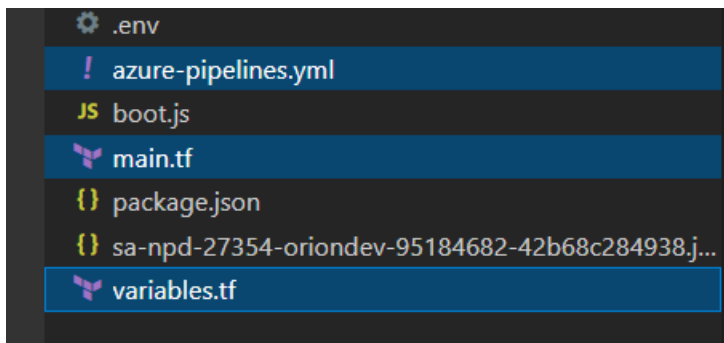
CI

azure-pipelines.yml

> 创建Pipeline时将配置为根据该yml文件设置pipeline

自动触发Pipeline的code branch

自动触发Pipeline的文件改动

```
trigger:
  batch: true
  branches:
    include:
    - development
    - release_V2
    - master_V2
    - release
    - R22.2_Development
  paths:
    include:
    - Infra/002-215-cf-archivecontractingstatus/*
```

根据项目名称更改

```
variables:
  TF_SRC_PATH: './Infra/002-215-cf-archivecontractingstatus'
  CFName: 'mmc-cf-archivecontractingstatus'
  NodeJSVer: '12.x'
  Archive_Src_Folder: 'dist_archive'
  WebPack_Output_Folder: 'Infra/002-215-cf-archivecontractingstatus/src'
```

创建Pipeline

> 在Azure 中创建新pipeline

1. Pipelines   New pipeline

New pipeline
Where is your code?

Azure Repos Git `YAML`
Free private Git repositories, pull requests, and code search

Bitbucket Cloud `YAML`
Hosted by Atlassian

GitHub `YAML`
Home to the world's largest community of developers

2.

GitHub Enterprise Server `YAML`
The self-hosted version of GitHub Enterprise

Other Git
Any generic Git repository

Subversion
Centralized version control by Apache

Team Foundation Version Control
Centralized version control repositories

Use the classic editor to create a pipeline without YAML.

Select a source
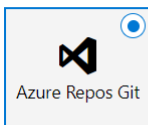
Azure Repos Git ⦿	TFVC	GitHub	GitHub Ent Serve

Other Git

Team project

3.

NxtGenMMC_27354 ▾

Repository

◆ NxtGenMMC_27354 ▾

Default branch for manual and scheduled builds

⑂ R22.2_Development ▾

Continue

Select a template

Or start with an 🏗 Empty job

Search

4.

Configuration as code

YAML
Looking for a better experience to configure your pipelines using YAML
files? Try the new YAML pipeline creation experience. Learn more

Featured

5.

6.
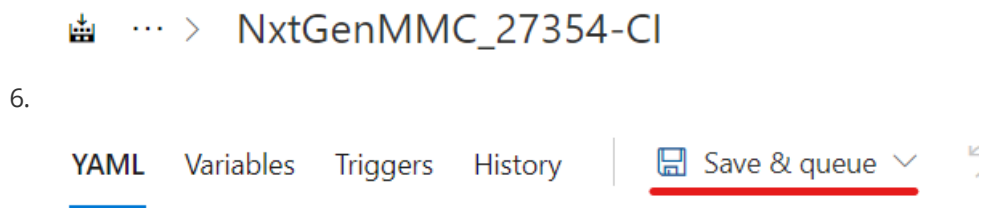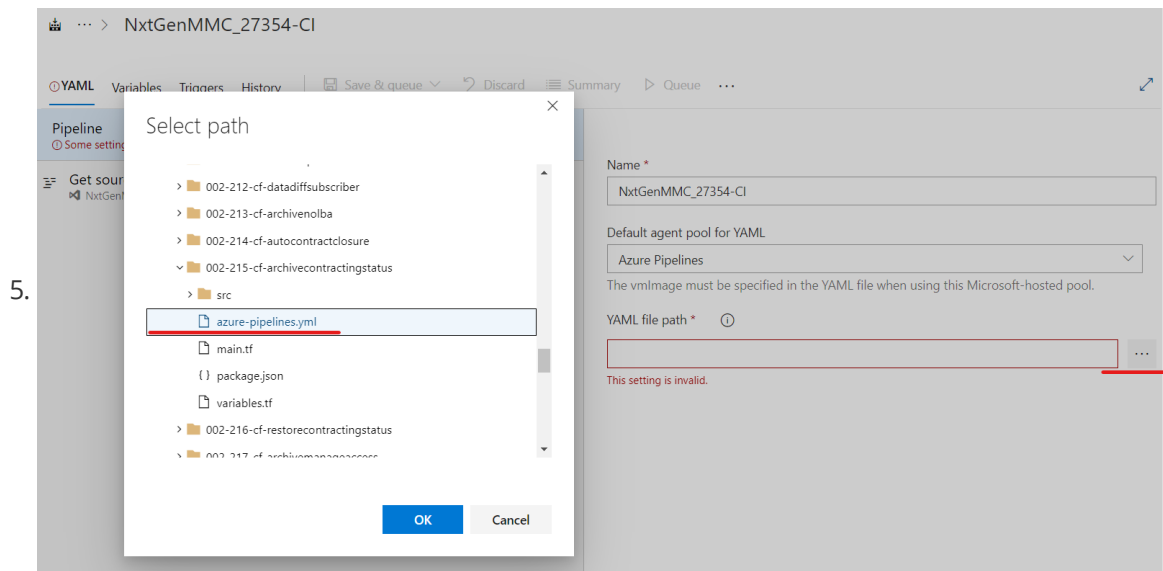
CD

main.tf

根据项目名称修改

```
module "google_cloud_function_archivecontractingstatus" {
    source                          = "acnciotfregistry.accenture.com/accenture-cio/function/google"
    version                         = "1.0.0"
    project_id                      = "${var.project_id}"
    google_function_name            = "archivecontractingstatus"
    google_function_entrypoint      = "archivecontractingstatus"
    region                          = ""
    available_memory_mb             = 2048
    timeout                         = 540
    runtime                         = "nodejs12"
    trigger_http                    = false
    trigger_event                   = "${var.trigger_event}"
    source_dir                      = "./src"
    path_to_data_to_upload          = "./mmc-archivecontractingstatus.zip"
    google_storage_bucket_name      = "${var.source_archive_bucket}"
    google_storage_bucket_object_name = "${var.source_archive_object}"
    app_execution_service_account   = "${var.app_execution_service_account}"
    environment_variables           = "${var.input_env_variables}"
}
```
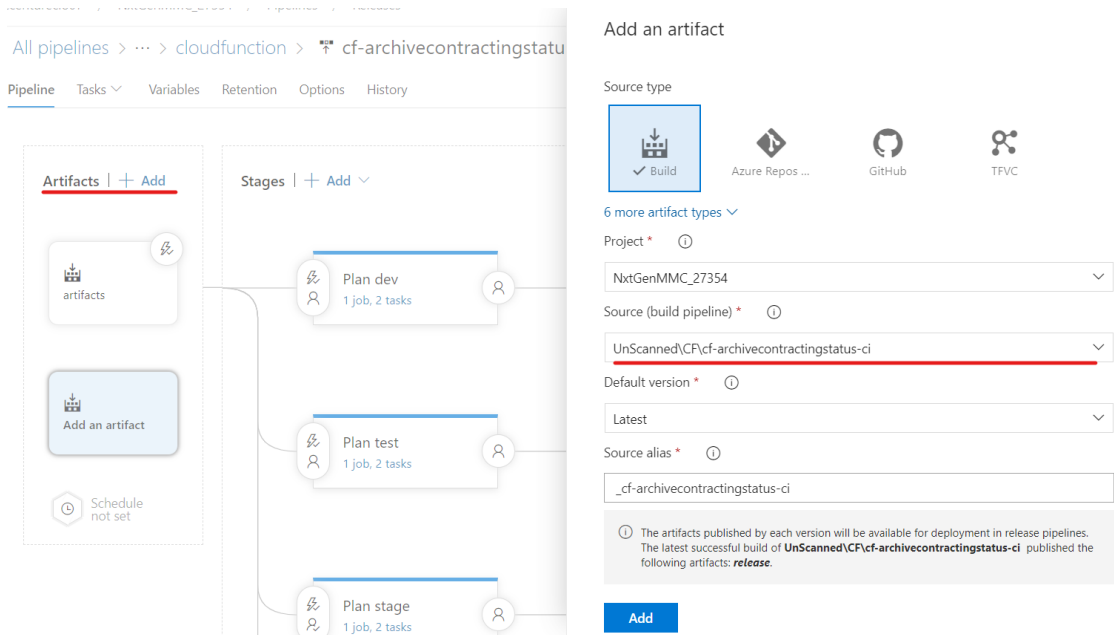
创建Release

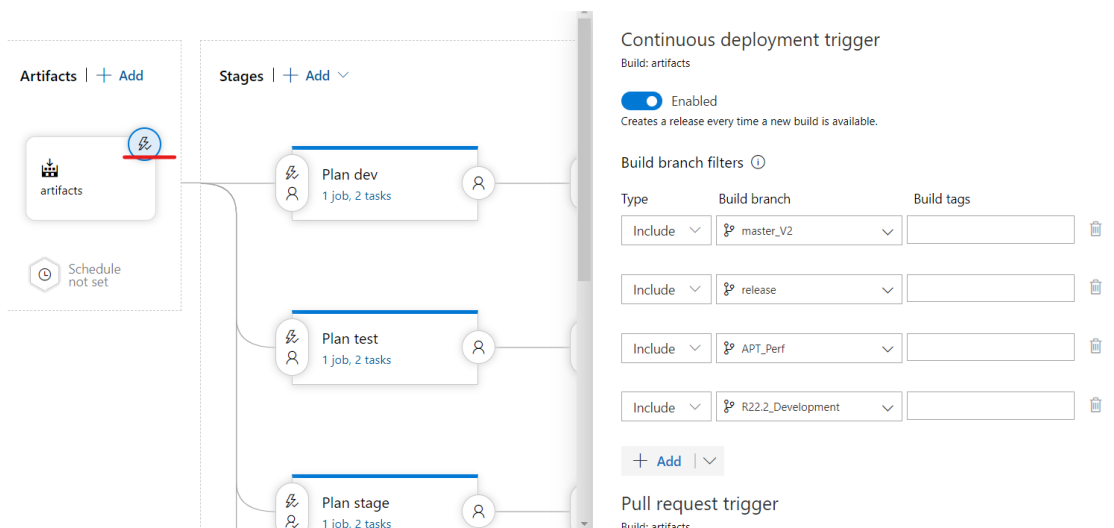> 在Azure 创建Release， 参考cf-archivecontractingstatus-cd

1. clone 现有 Release
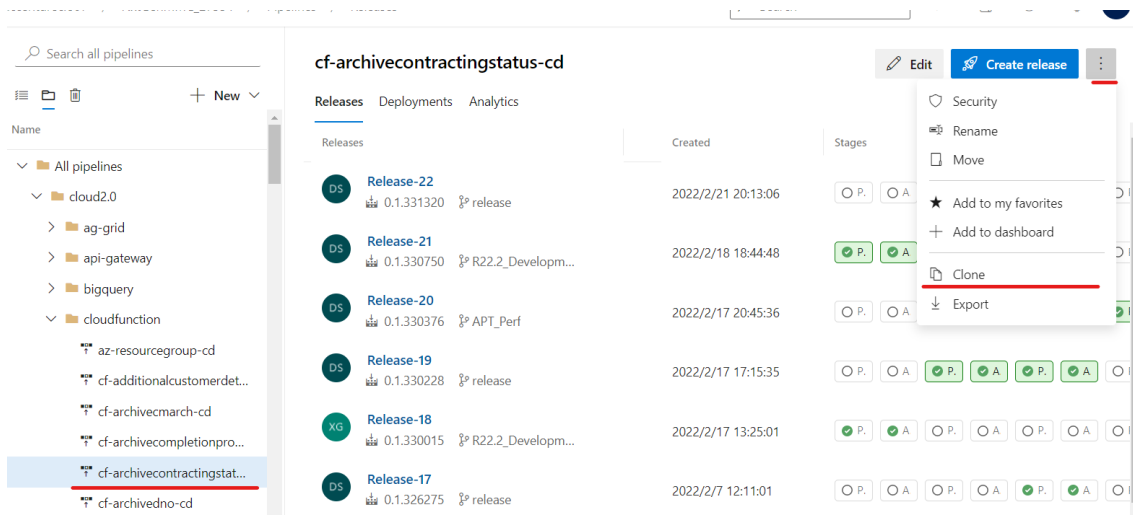
2. 新增Artifacts，选择之前创建好的CI的路径

3. 修改自动触发规则，**配置成功后如pipeline执行成功将自动执行release**

4. 修改Deployed approver （参照cf-archivecontractingstatus-cd 有些环境不需要设置approver）

5. 修改变量

$(CF_JOBARCHIVE_TRIGGER_EVENT) 对应 Variable groups 中各个环境中的值，此变量为该CF所订阅的Topic需要根据CF具体功能需求所更改

6. 检查Variable groups 中各个环境中是否都已经定义了 Pipeline variables 中所使用的变量 ($(CF_JOBARCHIVE_TRIGGER_EVENT))

Pipeline variables

Variable groups

Predefined variables ⧉

	Name	Valu
∨	**MMC_Cloud2.0_Common (17)**	Sco
∨	**MMC_Cloud2.0_Dev (164)**	⋯ Sco
∨	**MMC_Cloud2.0_Test (163)**	Sco
∨	**MMC_Cloud2.0_Stage (159)**	Sco
∨	**MMC_Cloud2.0_Perf (162)**	Sco
∨	**MMC_Cloud2.0_Prod (155)**	Sco
∨	**MMC_Cloud2.0_Prod - secondary (117)**	Sco
∨	**MMC_Cloud2.0_OPS (144)**	Sco