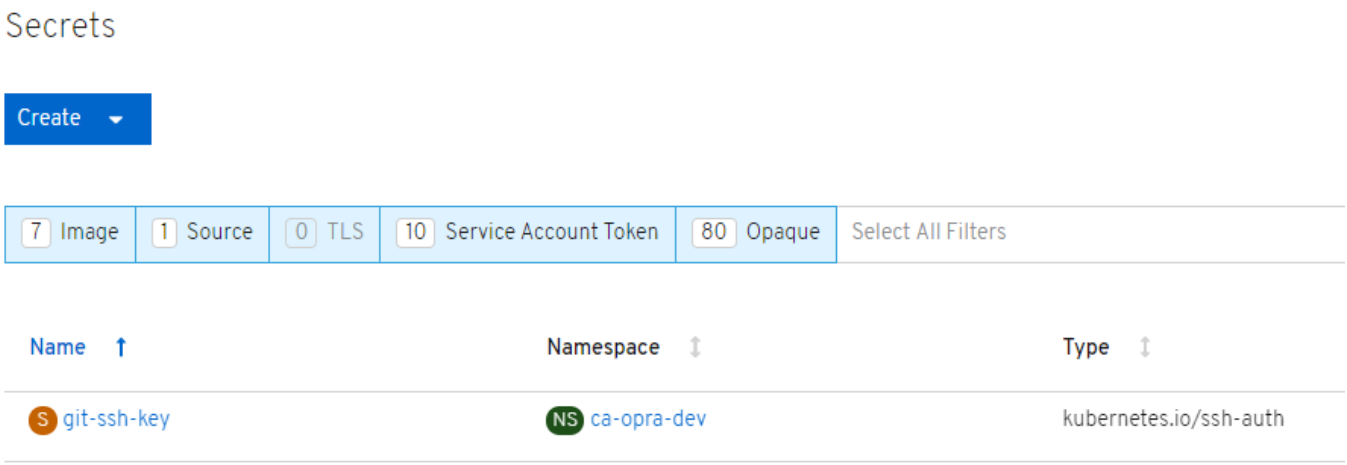
openshift 环境部署

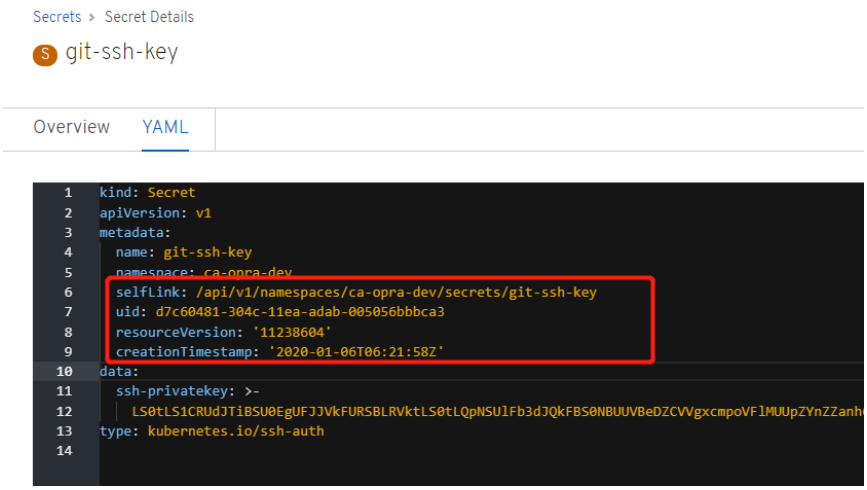
- 1 配置ssh
- 2 ImageStream创建
- 3 Build镜像安装
- 4 maven配置
- 5 模板安装
- 6 发布应用

配置ssh

从其他环境拷贝yml文件内容并修改namespace



删除红框中内容



或者直接拷贝下面的内容，注意修改命名空间

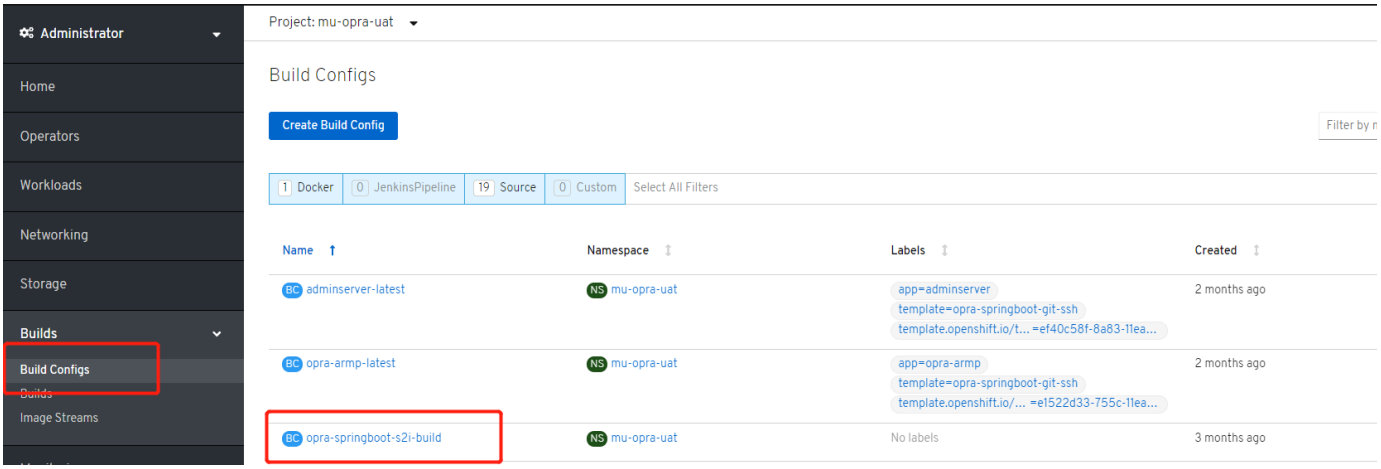
```
kind: Secret
apiVersion: v1
metadata:
  name: git-ssh-key
  namespace:
data:
  ssh-privatekey:
LS0tLS1CRUdJTiBSU0EgUFJJVkFURSBLRVktLS0tLQpNSUlFb3dJQkFBS0NBUUVBeDZCVVgxcmp
oVFlMUUpZYnZZanh6QlZiOWFKbzYwS01nMjZnUmhtRk1DbVlUaGwzCk9EdW1PdGJuSXo5V215cn
I2ZVlNT0VJQXR6UkNXbFhocHhIZWdVM3dHZFNxeW1qVTRTWWZlLzYrUytmaEJpRDEKSGRXSDRad
Eh1NWNhaXkvRlViU0FtNExnYlJvdUdkVGZFeHVLMnk1dUE2NUcza0RQQVpIT2h1QmlkbDBJK0hM
cwpBWk9wYlFydmJCQnFFM1p0bXRYTGVUckFjTVZuZ1haVS9JSlZ6bW80MTg1WFRUaitjNjg4Ykt
TbGdJak00Z2xsCmVDN1ZBNWxkQ1V1L3crN0x6RUdIeDlhbHNLQ0Z1ZFZQNUJ1VHN1WVkwRWFvV2
0zYlZNRi9ucmhtWEY5T1lETjIKMUNlNTNyZXRMcTdNWXZxTGZmUk9ER21CeWcweWVWNFpHM1BkS
ndJREFRQUJBb0lCQUUzNHc3dlRCbGM3dXIxcApaRUJ0d3ZRUnk2OXYxa2M5NFNCZEQyRVNDc0c3
SFhBejVpc2lzVlQrSjUzTzJZY1dCcnQvVFhxVDNTK3h5OUQ1CkJvQjVMRTJaSVpVVlNmaHhaZVR
qUFJ6emJXUzhsMW9MZVhKRHV5VlZ2M2o0RmpnVFdxRUZQYzZBR2lURE93cC8KNlBuUG9qYm1OYl
plVmZQdG9HMWVoQ3FPYmljSWt2cEh3azU3TDBKM2o5V0NRZkRQQklXTHBPdG1EeUZ2REovdgpLN
ncwUEFFRlhnVEtLZTRMNVhVTEhUTWF2eXJKTDZHZlJWcmM0Y21tNlI5dlRORnJoRjBMODBXNWIw
MWQ1VlpiCjVCQ1FYd1JHbUkvL28zc3JpcGh1dFlJZUNFdnlna3lwRWgrMWFuZGF4YUw5b3JFeVR
mV2NxOUlIOWlZOE51OW4KQTJka3BsRUNnWUVBN0NlWkJYb1E5WktXNDFueXo0d1JaWkVKZUlUbm
pZN3RNL1ZxWjUra2hGSm1qUy9iSjdHQgp1Q1VBQVl3SGJ4Q1FMTVpSc1ByZGtjWlROOW5CU0VLd
3lZL3hESXl6aG5KemtROVo2MDRJWU5NeGpHd2x6UHpOCk5oS292R3VxOS92OUFLQUVBb0RmeDVU
RWg2Ty9BM1pTRFRlejVISTNUUDQwc0RTZXovRzRDcTBDZ1lFQTJHYmwKUVduK1MzZjZOdkdESGZ
mdkVhSWwvSVJvRGhxNktJNXR4MTIxSFBBUjExdWZSYkFaYUVLM3hLcDlqQ1lVOXFXZQpqYVVvZm
pMWW9RTU5uVklUUDEvdkhFNk5PRlk1dXN5REsxWlRYakRWNGZPdGMvMVViQU9VTERnb1R5SmZBR
URDCi80SW1PTmkvb2ZtTmRkbUJ0L3JSeFFDZ1ZVSEFWZFF5N1VCMGRhTUNnWUVBenN0YUQxYTE5
U1Y1cGZzRlNrUEEKbEMwdW11L2ZDTEhvK3ovai9udm9oOHJVYkx4RWIvemJ1QklHY0ZSYnlpRTU
3MHQzQzhDU04rM3d2NDZhMTVrMgowRHFRL2NsVUI5Ni9YbjZQNGlMQjZXL25DcTFGYUx6VFFvK1
NYUFQza1JLRG1ZR0dJVjhURnVVcU9CRG0yclJJCjQ3UXNmbGFhSUh6V1RKcTNteVp6WUQwQ2dZQ
k1qWm5zV2lLcmIvWUVHY3ZZZmkwRWYrZ1FQSWJTOW9HRWQxNCsKZU5mcE5NUWErejZNMkk4a1l2
Vy9qTkI5RTd1MnhVQWh1NXRuaVhVNUFHdUFpOWhqUXZ6RGRzWmVkNkpVZ3FTawpZQlY5N0tpOUZ
nYjJVVmZjSGlmOHZLRUYzWkxkOUFzSGVITXRDUUJCdkpaZWFtT0pRaUdUSDNXWXMzbm00dnBoCk
U3d2ZJUUtCZ0dsT0h0M0JDNmEreVdJNG9nclNVUDBxLzVGRFpVNWViTm81VDBFbmg0dVk3RFhNe
jgvemtKeUcKeXRHQzAyemMyT2J6RmVMN2JQYytsSGpnYTd5YVJxRHJNeDBjU0t5NHNENldLN0M2
M3VGMlplTXpWTGt2M3ltZwpuOTZHUVpsYXhYbkd6Q25hNVY3T3MzNWxYdHk3K1dqSVBGV3Q2Tkh
oUDlhS0JHQ1V6aWxWCi0tLS0tRU5EIFJTQSBQUklWQVRFIEtFWS0tLS0t
type: kubernetes.io/ssh-auth
```

ImageStream创建

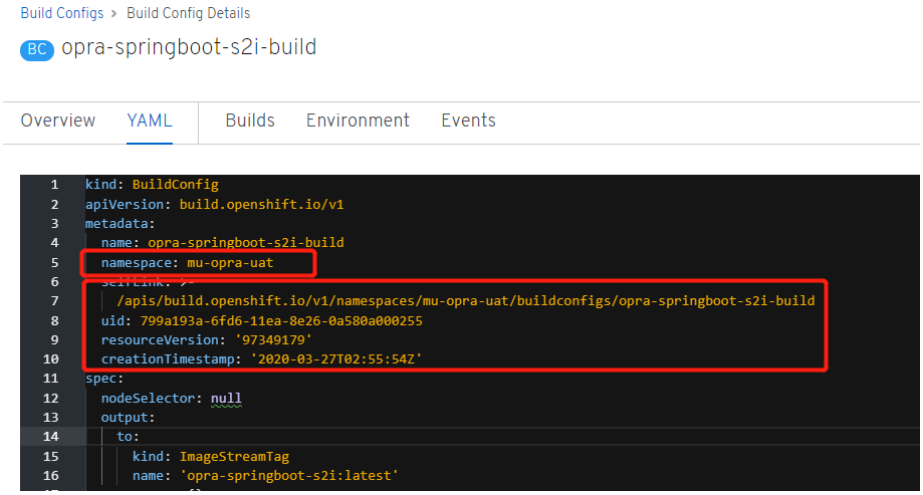
```
kind: ImageStream
apiVersion: image.openshift.io/v1
metadata:
  name: opra-springboot-s2i
  namespace:
spec:
  lookupPolicy:
    local: false
  tags:
    - name: latest
      annotations:
        description: OPRA Spring Boot S2I image
        iconClass: icon-jboss
        supports: 'springboot,java:8,maven:3'
        tags: 'builder,springboot,java,maven'
        version: '1.0'
      generation: 1
      importPolicy: {}
      referencePolicy:
        type: Source
```

Build镜像安装

从其他环境拷贝yaml配置文件



修改yaml中的namespace，并删除红框中无用信息



start build 之后生成s2i基础镜像

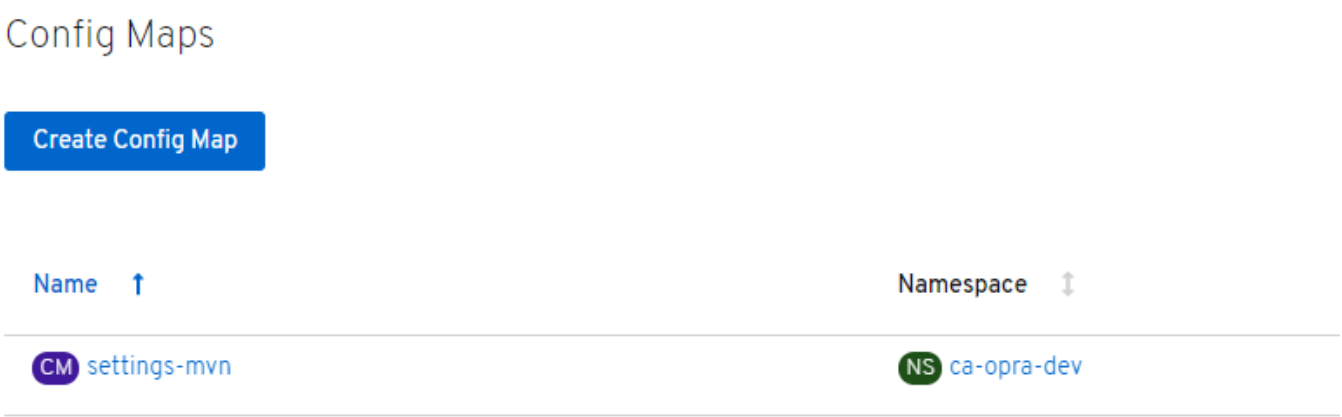
yaml参见下面代码：

```
kind: BuildConfig
apiVersion: build.openshift.io/v1
metadata:
  name: opra-springboot-s2i-build
  namespace:
spec:
  nodeSelector: null
  output:
    to:
      kind: ImageStreamTag
      name: 'opra-springboot-s2i:latest'
  resources: {}
  successfulBuildsHistoryLimit: 5
  failedBuildsHistoryLimit: 5
  strategy:
    type: Docker
    dockerStrategy:
      from:
        kind: DockerImage
        name: 'registry.test.ocp.acca/opra/base-centos7:latest'
  postCommit: {}
  source:
    type: Git
    git:
      uri: 'ssh://git@git.acca.com.cn:7999/opra-git/s2i-build.git'
      ref: master
    contextDir: springboot
    sourceSecret:
      name: git-ssh-key
  triggers: []
  runPolicy: Serial
status:
  lastVersion: 0
```

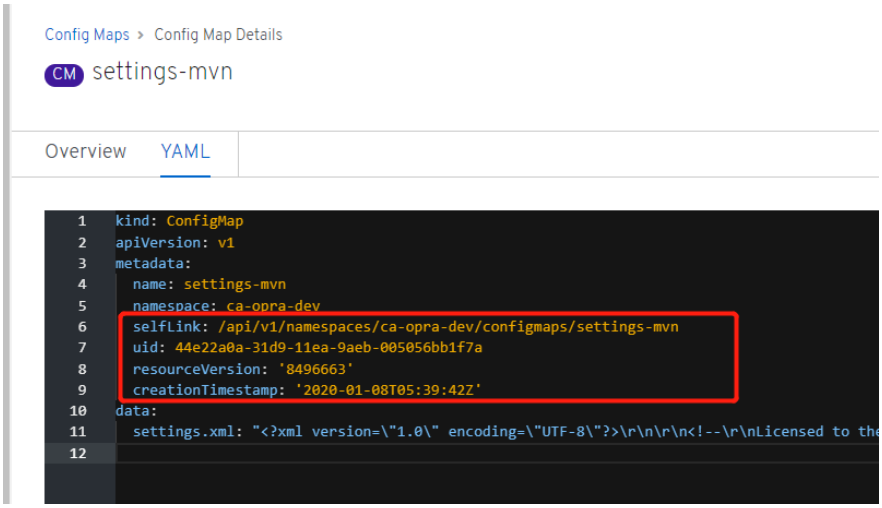
配置完成之后，点击start build，生成基础builder镜像。

maven配置

从其他环境拷贝yml文件并修改namespace



删除红框中信息



创建config map yaml:

```
kind: ConfigMap
apiVersion: v1
metadata:
  name: settings-mvn
  namespace:
data:
  settings.xml: "<?xml version=\"1.0\" encoding=\"UTF-8\"?>\r\n\r\n<!--
\r\nLicensed to the Apache Software Foundation (ASF) under one\r\nor more
contributor license agreements.  See the NOTICE file\r\ndistributed with
this work for additional information\r\nregarding copyright ownership.
The ASF licenses this file\r\nto you under the Apache License, Version 2.0
(the\r\n\"License\"); you may not use this file except in
compliance\r\nwith the License.  You may obtain a copy of the License
at\r\n\r\n    http://www.apache.org/licenses/LICENSE-2.0\r\n\r\nUnless
required by applicable law or agreed to in writing,\r\nsoftware
distributed under the License is distributed on an\r\n\"AS IS\" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY\r\nKIND, either express or
implied.  See the License for the\r\nspecific language governing
```

permissions and limitations\r\nunder the License.\r\n-->\r\n\r\n<!--\r\n | This is the configuration file for Maven. It can be specified at two levels:\r\n |\r\n | 1. User Level. This settings.xml file provides configuration for a single user, \r\n | and is normally provided in ${user.home}/.m2/settings.xml.\r\n |\r\n | NOTE: This location can be overridden with the CLI option:\r\n |\r\n | -s /path/to/user/settings.xml\r\n |\r\n | 2. Global Level. This settings.xml file provides configuration for all Maven\r\n | users on a machine (assuming they're all using the same Maven\r\n | installation). It's normally provided in \r\n | ${maven.home}/conf/settings.xml.\r\n |\r\n | NOTE: This location can be overridden with the CLI option:\r\n |\r\n | -gs /path/to/global/settings.xml\r\n |\r\n | The sections in this sample file are intended to give you a running start at\r\n | getting the most out of your Maven installation. Where appropriate, the default\r\n | values (values used when the setting is not specified) are provided.\r\n |\r\n |-->\r\n<settings xmlns=\" http://maven.apache.org/SETTINGS/1.0.0\" \r\n   xmlns:xsi=\" http://www.w3.org/2001/XMLSchema-instance\" \r\n   xsi: schemaLocation=\"http://maven.apache.org/SETTINGS/1.0.0 http://maven.apache.org/xsd/settings-1.0.0.xsd\">\r\n <!-- localRepository\r\n | The path to the local repository maven will use to store artifacts.\r\n |\r\n | Default: ~/.m2/repository\r\n <localRepository>/path/to/local/repo</localRepository>\r\n -->\r\n <!-- interactiveMode\r\n | This will determine whether maven prompts you when it needs input. If set to false,\r\n | maven will use a sensible default value, perhaps based on some other setting, for\r\n | the parameter in question.\r\n |\r\n | Default: true\r\n <interactiveMode>true</interactiveMode>\r\n -->\r\n\r\n <!-- offline\r\n | Determines whether maven should attempt to connect to the network when executing a build.\r\n | This will have an effect on artifact downloads, artifact deployment, and others.\r\n |\r\n | Default: false\r\n <offline>false</offline>\r\n -->\r\n\r\n <!-- pluginGroups\r\n | This is a list of additional group identifiers that will be searched when resolving plugins by their prefix, i.e.\r\n | when invoking a command line like \"mvn prefix:goal\". Maven will automatically add the group identifiers\r\n | \"org.apache.maven.plugins\" and \"org.codehaus.mojo\" if these are not already contained in the list.\r\n |-->\r\n <pluginGroups>\r\n   <!-- pluginGroup\r\n | Specifies a further group identifier to use for plugin lookup.\r\n   <pluginGroup>com.your.plugins</pluginGroup>\r\n   -->\r\n </pluginGroups>\r\n\r\n <!-- proxies\r\n | This is a list of proxies which can be used on this machine to connect to the network.\r\n | Unless otherwise specified (by system property or command-line switch), the first proxy\r\n | specification in this list marked as active will be used.\r\n |-->\r\n <proxies>\r\n   <!-- proxy\r\n | Specification for one proxy, to be used in connecting to the network.\r\n |\r\n   <proxy>\r\n   <id>optional</id>\r\n   <active>true</active>\r\n   <protocol>http</protocol>\r\n   <username>proxyuser</username>\r\n   <password>proxypass</password>\r\n   <host>proxy.host.net</host>\r\n   <port>80</port>\r\n   <nonProxyHosts>local.net|some.host.com</nonProxyHosts>\r\n   </proxy>\r\n   -->\r\n </proxies>\r\n\r\n\r\n <!-- servers\r\n | This is a list of authentication profiles, keyed by the server-id used within the system.

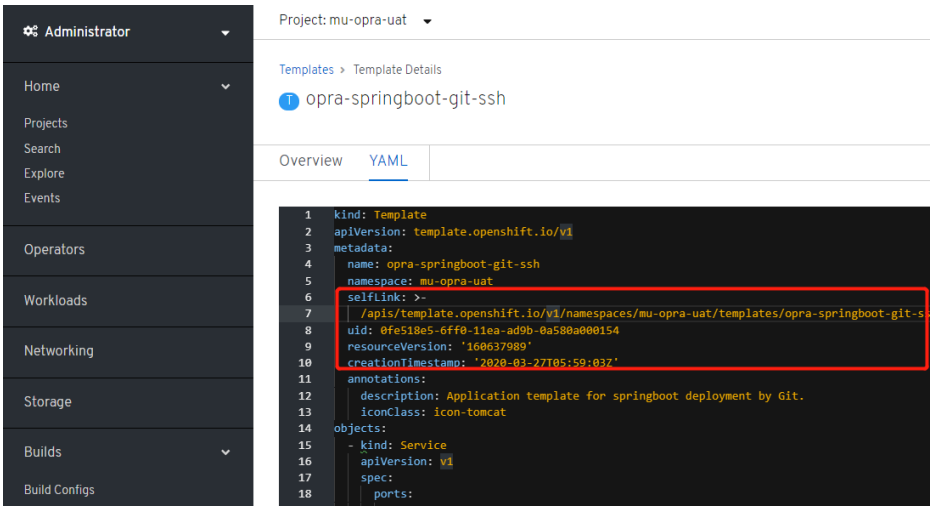
\r\n | Authentication profiles can be used whenever maven must make a connection to a remote server.\r\n |-->\r\n <servers>\r\n   <!-- server\r\n | Specifies the authentication information to use when connecting to a particular server, identified by\r\n | a unique name within the system (referred to by the 'id' attribute below).\r\n | \r\n | NOTE: You should either specify username/password OR privateKey/passphrase, since these pairings are \r\n | used together.\r\n |\r\n   <server>\r\n   <id>deploymentRepo</id>\r\n <username>repouser</username>\r\n   <password>repopwd</password>\r\n </server>\r\n   -->\r\n   \r\n   <!-- Another sample, using keys to authenticate.\r\n   <server>\r\n   <id>siteServer</id>\r\n <privateKey>/path/to/private/key</privateKey>\r\n <passphrase>optional; leave empty if not used.</passphrase>\r\n   </server>\r\n   -->\r\n\t<server>\r\n   <id>Snapshots</id>\r\n <username>deployment</username>\r\n   <password>deploy1234</password>\r\n   </server>\r\n\t<server>\r\n   <id>ACCA-Snapshots</id>\r\n   <username>admin</username>\r\n   <password>admin123</password>\r\n   </server>\r\n\t<server>\r\n   <id>ACCA-Releases</id>\r\n   <username>admin</username>\r\n   <password>admin123</password>\r\n   </server>\r\n\t<server>\r\n   <id>IATA-Snapshots</id>\r\n   <username>admin</username>\r\n   <password>admin123</password>\r\n   </server>\r\n </servers>\r\n\r\n <!-- mirrors\r\n | This is a list of mirrors to be used in downloading artifacts from remote repositories.\r\n | \r\n | It works like this: a POM may declare a repository to use in resolving certain artifacts.\r\n | However, this repository may have problems with heavy traffic at times, so people have mirrored\r\n | it to several places.\r\n |\r\n | That repository definition will have a unique id, so we can create a mirror reference for that\r\n | repository, to be used as an alternate download site. The mirror site will be the preferred \r\n | server for that repository.\r\n |-->\r\n <mirrors>\r\n   <!-- mirror\r\n | Specifies a repository mirror site to use instead of a given repository. The repository that\r\n | this mirror serves has an ID that matches the mirrorOf element of this mirror. IDs are used\r\n | for inheritance and direct lookup purposes, and must be unique across the set of mirrors.\r\n |\r\n   <mirror>\r\n   <id>mirrorId</id>\r\n <mirrorOf>repositoryId</mirrorOf>\r\n   <name>Human Readable Name for this Mirror.</name>\r\n   <url>http://my.repository.com/repo/path</url>\r\n   </mirror>\r\n\t\r\n   -->\r\n\t<mirror>\r\n <id>LocalMirrorId</id>\r\n   <mirrorOf>*</mirrorOf>\r\n <name>Nexus Public Mirror.</name>\r\n   <url>http://10.1.15.172:8080/nexus/content/groups/public</url>\r\n   </mirror>\r\n\r\n\r\n </mirrors>\r\n \r\n <!-- profiles\r\n | This is a list of profiles which can be activated in a variety of ways, and which can modify\r\n | the build process. Profiles provided in the settings.xml are intended to provide local machine-\r\n | specific paths and repository locations which allow the build to work in the local environment.\r\n |\r\n | For example, if you have an integration testing plugin - like cactus - that needs to know where\r\n | your Tomcat instance is installed, you can provide a variable here such that the variable is \r\n | dereferenced during the build process to configure the cactus plugin.\r\n |\r\n | As noted above, profiles can be activated in a variety of ways. One way - the activeProfiles\r\n | section of this document

```
(settings.xml) - will be discussed later. Another way essentially\r\n   |
relies on the detection of a system property, either matching a particular
value for the property,\r\n   | or merely testing its existence. Profiles
can also be activated by JDK version prefix, where a \r\n   | value of
'1.4' might activate a profile when the build is executed on a JDK version
of '1.4.2_07'.\r\n   | Finally, the list of active profiles can be
specified directly from the command line.\r\n   |\r\n   | NOTE: For
profiles defined in the settings.xml, you are restricted to specifying
only artifact\r\n   |       repositories, plugin repositories, and free-
form properties to be used as configuration\r\n   |       variables for
plugins in the POM.\r\n   |\r\n   |-->\r\n  <profiles>\r\n    <!--
profile\r\n     | Specifies a set of introductions to the build process,
to be activated using one or more of the\r\n     | mechanisms described
above. For inheritance purposes, and to activate profiles via
<activatedProfiles/>\r\n     | or the command line, profiles have to have
an ID that is unique.\r\n     |\r\n     | An encouraged best practice for
profile identification is to use a consistent naming convention\r\n     |
for profiles, such as 'env-dev', 'env-test', 'env-production', 'user-
jdcasey', 'user-brett', etc.\r\n     | This will make it more intuitive to
understand what the set of introduced profiles is attempting\r\n     | to
accomplish, particularly when you only have a list of profile id's for
debug.\r\n     |\r\n     | This profile example uses the JDK version to
trigger activation, and provides a JDK-specific repo.\r\n
<profile>\r\n      <id>jdk-1.4</id>\r\n\r\n      <activation>\r\n
<jdk>1.4</jdk>\r\n      </activation>\r\n\r\n
<repositories>\r\n        <repository>\r\n          <id>jdk14<
/id>\r\n          <name>Repository for JDK 1.4 builds</name>\r\n
<url>http://www.myhost.com/maven/jdk14</url>\r\n          <layout>default<
/layout>\r\n          <snapshotPolicy>always</snapshotPolicy>\r\n        <
/repository>\r\n      </repositories>\r\n    </profile>\r\n    --
>\r\n\r\n    <!--\r\n     | Here is another profile, activated by the
system property 'target-env' with a value of 'dev',\r\n     | which
provides a specific path to the Tomcat instance. To use this, your plugin
configuration\r\n     | might hypothetically look like:\r\n     |\r\n
| ...\r\n     | <plugin>\r\n     |   <groupId>org.myco.myplugins<
/groupId>\r\n     |   <artifactId>myplugin</artifactId>\r\n     |
\r\n     |   <configuration>\r\n     |     <tomcatLocation>${tomcatPath}<
/tomcatLocation>\r\n     |   </configuration>\r\n     | </plugin>\r\n
| ...\r\n     |\r\n     | NOTE: If you just wanted to inject this
configuration whenever someone set 'target-env' to\r\n     |
anything, you could just leave off the <value/> inside the activation-
property.\r\n     |\r\n    <profile>\r\n      <id>env-dev<
/id>\r\n\r\n      <activation>\r\n        <property>\r\n
<name>target-env</name>\r\n          <value>dev</value>\r\n        <
/property>\r\n      </activation>\r\n\r\n      <properties>\r\n
<tomcatPath>/path/to/tomcat/instance</tomcatPath>\r\n      <
/properties>\r\n    </profile>\r\n    -->\r\n <profile>\r\n
<id>snapshots</id>\r\n      <activation>\r\n        <activeByDefault>true<
/activeByDefault>\r\n      </activation>\r\n\r\n
<repositories>\r\n        <repository>\r\n          <id>LocalMirrorId<
/id>\r\n          <url>http://10.1.15.172:8080/nexus/content/groups/public<
/url>\r\n          <snapshots>\r\n            <enabled>true<
/enabled>\r\n            <updatePolicy>daily</updatePolicy>\r\n          <
```

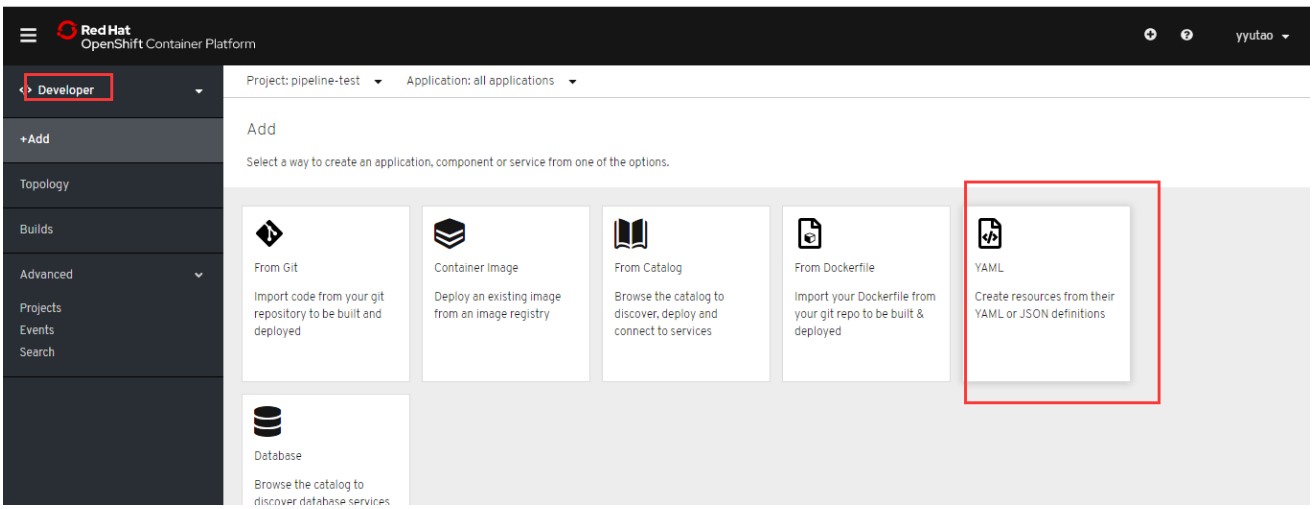
```
/snapshots>\r\n        </repository>\r\n      </repositories>\r\n    <
/profile>\r\n\r\n\r\n  </profiles>\r\n\r\n  <!-- activeProfiles\r\n   |
List of profiles that are active for all builds.\r\n   |\r\n
<activeProfiles>\r\n    <activeProfile>alwaysActiveProfile<
/activeProfile>\r\n    <activeProfile>anotherAlwaysActiveProfile<
/activeProfile>\r\n  </activeProfiles>\r\n  -->\r\n</settings>\r\n"
```

模板安装

从其他环境拷贝一份template，修改namespace,并删除红框中无用信息



选择develop模式



模板yaml，注意修改命名空间，两处； 注意拷贝之后可能有格式的错误，resources处，注意调整；注意修改spring.profiles.active= 后面的profile名字

```
kind: Template
apiVersion: template.openshift.io/v1
metadata:
  name: opra-springboot-git-ssh
  namespace:
  annotations:
    description: Application template for springboot deployment by Git.
    iconClass: icon-tomcat
objects:
  - kind: Service
    apiVersion: v1
    spec:
      ports:
        - name: 8080-tcp
          port: 8080
          targetPort: 8080
      selector:
        svc: '${APPLICATION_NAME}'
    metadata:
      name: '${APPLICATION_NAME}'
      labels:
        app: '${APPLICATION_NAME}'
      annotations:
        description: The web server's http port.
  - kind: Route
    apiVersion: v1
    id: '${APPLICATION_NAME}'
    metadata:
      name: '${APPLICATION_NAME}'
      labels:
        app: '${APPLICATION_NAME}'
      annotations:
        description: Route for application's http service.
    spec:
      host: '${APPLICATION_HOSTNAME}'
      path: '/${APPLICATION_PATH}'
      targetPort: 8080-tcp
      to:
        name: '${APPLICATION_NAME}'
  - kind: ImageStream
    apiVersion: v1
    metadata:
      name: '${APPLICATION_NAME}'
      labels:
        app: '${APPLICATION_NAME}'
  - kind: BuildConfig
    apiVersion: v1
    metadata:
      name: '${APPLICATION_NAME}-${APPLICATION_VERSION}'
      labels:
        app: '${APPLICATION_NAME}'
    spec:
```

```
    resouce:
      limits:
        cpu: '2'
        memory: 2Gi
      requests:
        cpu: '1'
        memory: 1Gi
    source:
      git:
        uri: '${GIT_URI}'
        ref: '${GIT_BRANCH}'
      contextDir: '${GIT_CONTEXT_DIR}'
      sourceSecret:
        name: '${GIT_SSH_KEY}'
      configMaps:
        - configMap:
            name: settings-mvn
          destinationDir: .m2
    strategy:
      type: Source
      sourceStrategy:
        from:
          kind: ImageStreamTag
          namespace:
          name: 'opra-springboot-s2i:latest'
        env:
          - name: APPLICATION_PATH
            value: '${APPLICATION_PATH}'
          - name: GIT_URI
            value: '${GIT_URI}'
          - name: APP_OPTIONS
            value: '${APP_OPTIONS}'
          - name: BUILDER_ARGS
            value: '${BUILDER_ARGS}'
          - name: JVM_OPTIONS
            value: '${JVM_OPTIONS}'
    output:
      to:
        kind: ImageStreamTag
        name: '${APPLICATION_NAME}:latest'
    triggers:
      - type: GitHub
        github:
          secret: '${GITHUB_TRIGGER_SECRET}'
      - type: Generic
        generic:
          secret: '${GENERIC_TRIGGER_SECRET}'
      - type: ImageChange
        imageChange: {}
  - kind: DeploymentConfig
    apiVersion: v1
    metadata:
      name: '${APPLICATION_NAME}-${APPLICATION_VERSION}'
```

```
      labels:
        app: '${APPLICATION_NAME}'
    spec:
      strategy:
        type: Rolling
        rollingParams:
          updatePeriodSeconds: 1
          intervalSeconds: 1
          timeoutSeconds: 600
          maxUnavailable: 25%
          maxSurge: 25%
      triggers:
        - type: ImageChange
          imageChangeParams:
            automatic: true
            containerNames:
              - '${APPLICATION_NAME}'
            from:
              kind: ImageStream
              name: '${APPLICATION_NAME}'
      replicas: 1
      selector:
        deploymentConfig: '${APPLICATION_NAME}-${APPLICATION_VERSION}'
      template:
        metadata:
          name: '${APPLICATION_NAME}-${APPLICATION_VERSION}'
          labels:
            svc: '${APPLICATION_NAME}'
            deploymentConfig: '${APPLICATION_NAME}-${APPLICATION_VERSION}'
            app: '${APPLICATION_NAME}'
        spec:
          nodeSelector:
            node-role.kubernetes.io/worker: ''
          containers:
            - resources:
                limits:
                  cpu: '2'
                  memory: 2Gi
                requests:
                  cpu: '0.5'
                  memory: 1Gi
              name: '${APPLICATION_NAME}'
              image: '${APPLICATION_NAME}'
              imagePullPolicy: Always
              readinessProbe:
                exec:
                  command:
                    - /bin/bash
                    - '-c'
                    - >-
                      curl
                      http://localhost:${APPLICATION_PORT}
/${APPLICATION_PATH}
```

```
      ports:
        - name: http
          containerPort: 8080
          protocol: TCP
      env:
        - name: APPLICATION_PATH
          value: '${APPLICATION_PATH}'
        - name: GIT_URI
          value: '${GIT_URI}'
    hostAliases:
      - hostnames:
          - paxc-p-hdfs-1
        ip: 10.1.21.11
      - hostnames:
          - paxc-p-hdfs-2
        ip: 10.1.21.12
      - hostnames:
          - paxc-p-hdfs-3
        ip: 10.1.21.13
      - hostnames:
          - paxc-p-hdfs-4
        ip: 10.1.21.14
      - hostnames:
          - paxc-p-hdfs-5
        ip: 10.1.21.15
      - hostnames:
          - paxc-p-hdfs-6
        ip: 10.1.21.16
      - hostnames:
          - paxc-p-hdfs-7
        ip: 10.1.21.17
      - hostnames:
          - paxc-p-hdfs-8
        ip: 10.1.21.18
      - hostnames:
          - paxc-p-hdfs-9
        ip: 10.1.21.19
      - hostnames:
          - paxc-p-hdfs-10
        ip: 10.1.21.20
      - hostnames:
          - paxc-p-hdfs-11
        ip: 10.1.21.21
      - hostnames:
          - paxc-p-hdfs-12
        ip: 10.1.21.22
      - hostnames:
          - paxc-p-hdfs-13
        ip: 10.1.21.23
      - hostnames:
          - paxc-p-hdfs-14
        ip: 10.1.21.24
      - hostnames:
```

```
              - paxc-p-hdfs-15
            ip: 10.1.21.25
          - hostnames:
              - paxc-p-hdfs-16
            ip: 10.1.21.26
          - hostnames:
              - paxc-p-hdfs-17
            ip: 10.1.21.27
          - hostnames:
              - paxc-p-hdfs-18
            ip: 10.1.21.28
          - hostnames:
              - paxc-p-zkp-1
            ip: 10.1.21.151
          - hostnames:
              - paxc-p-zkp-2
            ip: 10.1.21.152
          - hostnames:
              - paxc-p-zkp-3
            ip: 10.1.21.153
          - hostnames:
              - opra-t-sys-3
            ip: 10.1.9.221
parameters:
  - name: APPLICATION_NAME
    displayName: Application Name
    description: The name for the application.
    required: true
  - name: APPLICATION_VERSION
    displayName: Application Version
    description: The version for the application
    value: latest
    required: true
  - name: APPLICATION_PATH
    displayName: Application Path
    description: The path for the application. The file name for tomcat
webapps.
  - name: APPLICATION_PORT
    displayName: Application PORT
    description: The path for the application. The file name for tomcat
webapps.
    value: '8080'
  - name: APPLICATION_HOSTNAME
    displayName: Application Hostname
    description: >-
      Custom hostname for service routes.  Leave blank for default
hostname,
      e.g.: <application-name>.<project>.<default-domain-suffix>
  - name: GIT_URI
    displayName: Git source URI
    description: Git source URI for application
    required: true
  - name: GIT_BRANCH
```

```
    displayName: Git Branch
    description: Git Branch for application
    value: master
    required: true
  - name: GIT_CONTEXT_DIR
    displayName: Git Context Dir
    description: Override the default location inside the source code
repository
  - name: GIT_SSH_KEY
    displayName: GIT SSH KEY
    description: 'Set the Git source clone secret '
    value: git-ssh-key
  - name: APP_OPTIONS
    description: >-
      Application options. These options will be passed to the Spring Boot
      command line
    value: '--spring.profiles.active=dev'
  - name: BUILDER_ARGS
    description: Maven options. These options will be passed to the Maven
command line
  - name: JVM_OPTIONS
    description: JVM options. These options will be passed to the Spring
Boot command line
    value: '-Duser.timezone=GMT+08 -XX:MaxRAMFraction=2 -javaagent:/opt
/app-root/skywalking/agent/skywalking-agent.jar -Dskywalking.agent.
service_name='
  - name: MAVEN_OPTS
    description: >-
      Application options. These options will be passed to the Spring Boot
      command line
    value: '-Xmx2g'
labels:
  template: opra-springboot-git-ssh
```

发布应用

DevOps