

# 阿里云

实时计算（流计算）  
Flink半托管（基于ACK）

文档版本：20201103

## 法律声明

阿里云提醒您在使用或阅读本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。
2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。
3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的“现状”、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。
5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含“阿里云”、“Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。
6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

# 通用约定

格式	说明	样例
 危险	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	 危险 重置操作将丢失用户配置数据。
 警告	该类警示信息可能会导致系统重大变更甚至故障，或者导致人身伤害等结果。	 警告 重启操作将导致业务中断，恢复业务时间约十分钟。
 注意	用于警示信息、补充说明等，是用户必须了解的内容。	 注意 权重设置为0，该服务器不会再接受新请求。
 说明	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	 说明 您也可以通过按Ctrl+A选中全部文件。
>	多级菜单递进。	单击设置> 网络> 设置网络类型。
<b>粗体</b>	表示按键、菜单、页面名称等UI元素。	在结果确认页面，单击 <b>确定</b> 。
Courier字体	命令或代码。	执行 <code>cd /d C:/window</code> 命令，进入Windows系统文件夹。
斜体	表示参数、变量。	<code>bae log list --instanceid</code> <i>Instance_ID</i>
[ ] 或者 [a b]	表示可选项，至多选择一个。	<code>ipconfig [-all -t]</code>
{ } 或者 {a b}	表示必选项，至多选择一个。	<code>switch {active stand}</code>

# 目录

1.新功能发布记录	05
1.1. 2.1.5版本	05
1.2. 2.1.3版本	05
2.产品定价	07
3.产品概览	08
3.1. 概述	08
3.2. 管理控制台介绍	09
3.3. 开发控制台组件介绍	09
3.4. 支持的上下游存储	12
4.准备工作	13
4.1. 角色授权	13
4.2. RAM账号授权	15
5.开通流程	17
6.作业开发	18
6.1. 开发	18
6.2. 提交	18
7.运维管理	20
7.1. 作业运维	20
7.2. 开源自建监控组件（ACK）	22
7.3. 链路监控	24
7.4. 配置及查看作业日志	25
8.常见问题	34
9.相关协议	39

# 1.新功能发布记录

## 1.1. 2.1.5版本

本文为您介绍Flink半托管/基于ACK 2.1.5版本的重大功能变更。

### 版本重大功能变更

- 平台语言支持中文简体。

您也可以在控制台系统信息页面，切换为英文。

- 支持STS Token认证。

 **说明** Flink半托管/基于ACK 2.1.5版本支持STS Token认证。因此，您在购买Flink半托管/基于ACK服务时，不需要填写AccessKey信息，但已购集群仍然使用已存储的AccessKey信息进行安全认证。

- 优化API返回内容。

VVP-2.1.5版本的API中，添加如下固定结构，API返回内容将被填充到data字段，不再展开呈现。

```
{
  "success": true,
  "httpCode": 200,
  "message": null,
  "reason": null,
  "data": {}
}
```

## 1.2. 2.1.3版本

本文为您介绍云原生2.1.3版本的重大功能变更和主要修复缺陷。

### 版本重大功能变更

- 支持自定义Flink Pod Label。

详情请参见[Flink Pod Label](#)。

- 支持Flink 1.10镜像。
- 支持Flink 1.10用户包隔离功能。
- 支持Flink Configuration里使用占位符，动态填入配置信息。

详情请参见[Deployment Defaults](#)。

- Flink 1.10 默认使用阿里自研的Gemini State Backend。

在作业启动时，系统会自动添加以下配置，默认启动阿里自研的Gemini State Backend。

- `state.backend: com.alibaba.flink.statebackend.GeminiStateBackendFactory`
- `fs.oss.impl: org.apache.hadoop.fs.aliyun.oss.AliyunOSSFileSystem`
- `state.backend.gemini.snapshot.close.file: true`

 **说明** 如果需要使用您自己的State Backend，请在state.backend配置项中或者代码中指定State Backend。您可以使用Namespaced Defaults来设置Flink配置项默认值。

- API优化

- DeploymentTarget：新增spec.kind字段。

```
DeploymentTarget:
  metadata:
    ...
  spec:
    kind: kubernetes
  kubernetes:
    namespaces: vvp
```

- Job：新增job.spec.deploymentTarget.kind=kubernetes字段。

```
Job:
  metadata:
    ...
  spec:
    deploymentTarget:
      kind: kubernetes
    kubernetes:
      namespace: vvp
```

- Deployment创建页面支持三种视图

- Standard
- Advanced
- Yaml

- 扩展Deployment和Job的Status，返回更多信息

- Job
  - live status
  - restarts count
- Deployment
  - ClusterUnreachable
  - JobFailing
  - JobUnstable

## 主要缺陷修复

修复Flink1.8和Flink1.9镜像中oss log appender在debug log Level占用资源过大的问题。建议您使用新增的1.8.1-stream3-scala\_2.11和1.9.0-stream3-scala\_2.11镜像替换之前的镜像。

## 2. 产品定价

本文为您介绍Flink半托管/ACK产品的计费说明。

### Flink半托管 计费说明

您可以免费使用实时计算Flink版Flink半托管/ACK产品，但在使用Flink半托管/ACK过程中会创建相关资源，您需要为您使用的资源付费，计费说明如下：

- 容器服务ACK集群费用
  - 云服务器ECS计费说明请参见[按量付费](#)。
  - NAT网关计费说明请参见[计费概述](#)。
  - 负载均衡计费说明请参见[按量计费](#)。
- 实时计算Flink版系统费用

实时计算Flink版使用的是大小为100 GB的高效云盘，云盘费用详情请参见[创建云盘](#)。

## 3. 产品概览

### 3.1. 概述

本文为您介绍实时计算Flink半托管（原名Flink云原生）版本的优势以及选择Kubernetes作为容器调度平台的原因。

**Flink半托管**目前可以被部署在Kubernetes（容器服务ACK提供）。Flink半托管是Flink和容器服务ACK的深度集合，充分利用了ACK在云平台上整合阿里云虚拟化、存储、网络和安全的能力，为您提供极致的Flink使用体验。

- **开源**

产品提供多个的Flink版本，与开源Flink接口完全兼容，满足您的平滑迁移业务的需求。

- **免费**

平台各组件免费使用，您只需支付云服务器ECS（Elastic Compute Service）费用即可使用Flink半托管产品。

- **易用**

提供开箱即用的作业开发平台，以及Metric采集、展示、监控和报警的能力。满足分钟级别搭建平台的需求，节省时间。

- **增值**

未来将提供多种可插拔的增值组件，包括智能调优系统，机器学习组件等。

**Apache Flink**是Apache社区的先进开源项目，主攻流计算领域，具备高吞吐，低延时的特点，在流计算领域已经取代Storm/Spark Streaming，成为了目前流式处理领域的热门引擎。主要应用场景包括：实时数仓、实时风控和实时推荐等。

- **Kubernetes的优势**

**Kubernetes**是目前主流的开源的容器调度平台，它可以自动化应用容器的部署、扩展和操作，提供以容器为中心的基础架构，并快速高效地响应客户需求。Kubernetes具有架构简单、可扩展和自修复等特点。

Kubernetes包括以下优势：

- 强大的资源隔离。
- 快速部署、扩展和对接新应用。
- 节省资源、优化硬件资源的使用。
- Loosely Coupled、分布式、弹性、微服务化：应用程序分为更小的、独立的部件，支持动态部署和管理。

- **Flink选择Kubernetes的原因**

- 与传统大数据应用（例如Spark、Hadoop、MapReduce以及Hive等）不同的是，Flink作业进程处于常驻状态。作业发布后，即使上游源表没有数据，Flink作业也不停止，而是轮询拉取上游数据。
- 与其他批处理作业相比，流作业任务通常应用于实时风控和实时推荐的业务场景，其重要度与稳定性要求更高。

- **实时计算Flink半托管（基于ACK）的优势**

- **运维侧模型统一降低运维成本**

越来越多的企业将传统的在线任务服务切换至Kubernetes。Flink运行在Kubernetes上，可以统一在线和离线，降低用户运维成本。

- 弹性伸缩与成本优化

容器具有高弹性的特点。对于大数据业务而言，提高集群资源的利用率从而降低成本，是用户考虑的第一要素。Flink on Kubernetes模式，结合即将要推出的Flink单作业Autoscale功能，可以完美地应对单任务突发性的负载情况：

- 高流量时自动申请ECS，降低作业延时。
- 低流量时缩减作业资源、释放ECS，保证作业延时的同时降低用户成本。

## 3.2. 管理控制台介绍

本文为您介绍Flink半托管/ACK开发控制台、修改组件、释放集群和集群管理入口。

### 登录管理控制台

1. 使用阿里云账号，登录[阿里云官网](#)。
2. 单击产品分类 > 大数据 > 实时计算Flink版 > 管理控制台，进入Flink半托管/ACK管理控制台。
3. 在页面左上角，选择目标区域。



### 主要功能

- 释放入口  
在Flink半托管/ACK页签，单击目标工作空间操作列下的释放，单击确定即可删除所有服务。
- 修改组件入口  
在Flink半托管/ACK页签，单击目标工作空间操作列下的修改组件，您可以添加、升级和删除组件，此外您还可以查看组件日志、组件运行状态详情和修改组件CPU或内存。
- 开发控制台入口  
在Flink半托管/ACK页签，单击目标工作空间操作列下的控制台，即可进入Flink半托管/ACK开发控制台，进行业务开发和运维工作。
- 集群管理入口  
在Flink半托管/ACK页签，单击目标工作空间操作列下的集群管理，进入容器服务管理控制台进行相应集群管理操作。

## 3.3. 开发控制台组件介绍

Flink半托管版本提供Flink全球统一的官方商用开发控制平台VVP（Ververica Platform）。您可以在VVP平台一站式完成作业的开发、提交和运维。

### 作业管理

- JAR包管理  
您可以单击单击可添加文件或将JAR包拖到列表的方式上传JAR包。此外，您还可以下载或删除已上传的JAR包。
- 开发作业  
详情请参见[开发](#)。
- 提交作业  
详情请参见[提交](#)。

- 作业状态

页面右上角会显示两种作业状态：

- 期望状态

状态	说明
RUNNING	按作业配置定义，启动作业。
CANCELLED	停止任何正在运行的作业。
SUSPENDED	先对当前运行的作业创建Savepoint，再停止作业。

- 当前状态

状态	说明
RUNNING	按作业配置定义，启动作业。
CANCELLED	停止任何正在运行的作业。
SUSPENDED	先对当前运行的作业创建Savepoint，再停止作业。
FINISHED	作业运行完成。
TRANSITIONING	作业处于向期望状态的过度阶段。
FAILED	作业向期望状态转换失败，需要手动处理。

- 升级策略

您可以通过设置升级策略，修改运行中作业的资源、运行版本或作业运行参数等信息。

- **Stateless**：参数修改完成后，直接重启您的作业应用新配置。
- **Stateful**：参数修改完成后，作业将先创建一次Savepoint，再应用新的配置重启作业。
- **None**：不会自动重启作业。您需要手动操作，完成新配置的应用。

- 恢复策略

VVP支持您选择恢复作业的策略，包括以下三种：

- **Latest Savepoint**：使用最后一个通过VVP，创建成功的Savepoint。
- **Latest State**：最后一个可用State。
- **None**：不带状态恢复，不使用Savepoint。

 **说明** 升级策略和恢复策略需要配合使用。例如，如果升级策略选择STATEFUL，恢复策略选择NONE，则完成Savepoint后，作业会从作业的起始时间重新启动，而不是从Savepoint恢复。

- **Allow Non-Restore Strategy**

开启该功能后，在Savepoint不兼容的情况下，不会报错中断作业，而是重新启动作业。

## 系统管理

- 作业模板

您可以通过配置作业模板的方式，配置工作空间下所有作业日志默认输出到OSS或SLS。配置作业模板后，后续该工作空间下创建的所有作业的日志都会被存储到OSS或您配置的SLS项目下的Logstore。

### ● 部署目标

选择作业在K8s集群运行的项目空间。VVP服务启动时会创建一个默认的部署目标和K8s命名空间关联。作业在上线时可以选择部署目标运行在特定的K8s的命名空间下，您也可以创建新的关联信息。

 **说明** 请确认您填写的K8s的命名空间已存在，否则部署目标无效，使用该部署目标启动的作业会失败。

### ● API令牌

您可以创建令牌，对API用户进行授权。令牌角色分为owner、editor和viewer。

 **注意** 创建完令牌后，请立即复制令牌信息并妥善保存，否则刷新该页面后，您将无法查看令牌信息。

### ● 成员管理

您可以为阿里云RAM账号授权：

#### ○ 角色

不同角色功能差异如下。

功能	owner	editor	viewer
查看作业列表	Y	Y	Y
启停作业	Y	Y	N
更改作业配置	Y	Y	N
上传JAR	Y	Y	N
查看作业模板	Y	Y	Y
增删改作业模板	Y	Y	N
管理成员	Y	N	N

#### ○ 成员

固定格式为 `user:UID` 。其中：

- user: 固定值，不可修改。
- UID: 阿里云RAM账号。

 **注意** 英文冒号（:）和UID之间不能有空格，否则授权无效。

## 系统信息

您可以在系统信息页面，设置平台语言、用户界面时区。此外，您还可以查看版本信息和License信息。

 **说明** 目前时区设置仅决定属性的显示（例如作业启动时间等），不会参与Flink作业中的计算。

## 3.4. 支持的上下游存储

实时计算Flink半托管（基于ACK）版本支持丰富的上下游存储，包括Flink开源版本所支持的上下游存储和阿里云所支持的上下游存储。

### 支持开源的上下游存储

实时计算Flink半托管（基于ACK）版本支持Flink1.8、Flink1.9和Flink1.10三个版本，对应的版本支持所有开源的上下游存储（Connector）。各版本所支持的具体Connector信息，请参见Flink开源社区：

- [Flink1.8](#)
- [Flink1.9](#)
- [Flink1.10](#)

### 支持阿里云的上下游存储

实时计算Flink半托管（基于ACK）版本也支持阿里云的上下游存储（Connector）。您在Flink1.8、Flink1.9和Flink1.10三个版本中均可以使用以下阿里云产品：

- **源表（Source）**
  - 数据总线DataHub
  - 日志服务SLS
- **结果表（Sink）**
  - 云数据库HBase版
  - 数据总线DataHub
  - 日志服务SLS

 **说明** 对应存储的代码已上传至[GitHub](#)，您可以自行下载、编译和使用。如果您的Connector为日志服务SLS，则建议您使用OpenService的依赖，并采用Shade打包方式后重定位，避免产生包冲突问题。

## 4. 准备工作

### 4.1. 角色授权

实时计算Flink半托管模式访问您的K8s集群和OSS集群，需要您已完成Flink半托管角色授权。本文为您介绍如何为Flink半托管角色授权、如何查看权限策略，以及如何初始化权限策略。

 **注意** 角色授权完成后，请不要随意删除Flink和K8s任何角色，否则创建集群会报错。

#### Flink半托管角色授权

Flink半托管角色授权的流程分为**授权自动化角色**和**添加授权策略**两个步骤。在自动化角色授权的基础上，添加Flink半托管特有的授权策略后，即可开展Flink半托管的业务操作。

##### 1. 授权自动化角色

- i. 首次登录**实时计算统一控制台**，需要您进行角色授权。单击**前往授权**，进行角色授权。

 **说明** 角色授权提示，仅在您没有正确授予实时计算服务账号默认角色时出现。

- ii. 单击**AliyunStreamConsoleRole > 同意授权**，完成角色授权。

##### 2. 添加授权策略

完成RAM角色创建后，您可以将指定的授权策略添加至RAM角色中。实时计算支持以下两种添加授权策略的方式：

###### o 添加AliyunRAMFullAccess权限

- a. 登录RAM控制台。
  - 云账号登录**RAM控制台**。
  - RAM用户登录**RAM控制台**。
- b. 在**RAM角色管理**页面底部的**RAM角色名称**列中，单击**AliyunStreamConsoleRole**操作列下的**添加权限**。
- c. 选择**权限区域系统权限策略**右侧文本框中输入**AliyunRAMFullAccess**。
- d. 单击**权限策略名称**中的**AliyunRAMFullAccess**。
- e. 单击**确定**。

###### o 精确授权

如果您认为**AliyunRAMFullAccess**权限过大，也可以通过**精确授权**的方式为**AliyunStreamConsoleRole**授权。操作步骤如下：

- a. 登录RAM控制台。
  - 云账号登录**RAM控制台**。
  - RAM用户登录**RAM控制台**。
- b. 在左侧导航栏，单击**RAM角色管理**。
- c. 在**RAM角色名称**，找到目标RAM角色。
- d. 单击**精确授权**。
- e. 权限类型选择为**系统策略**。
- f. 在**输入策略名称**中，输入**AliyunStreamConsoleRolePolicy**。

- g. 单击确定。
- h. 单击关闭。

## 查看当前角色授权信息

您可以通过以下步骤查看当前已授权角色的信息。

1. 登录RAM控制台。
2. 在RAM角色管理页面底部的RAM角色名称列中单击AliyunStreamConsoleRole。
3. 在AliyunStreamConsoleRole页面中单击权限管理 > AliyunStreamConsoleRolePolicy。
4. 在策略内容页签内查看实时计算当前策略信息，如下所示。

```
{
  "Version": "1",
  "Statement": [
    {
      "Action": [
        "cs:GetClusterById",
        "cs:GetClusters",
        "cs:GetUserConfig",
        "cs:GetClusters"
      ],
      "Resource": "*",
      "Effect": "Allow"
    }
  ]
}
```

## 重新初始化

当您删除或修改AliyunStreamConsoleRolePolicy授权策略导致无法恢复时，您可以在RAM中删除AliyunStreamConsoleRole，并重新进入实时计算控制台，让实时计算重新授权自动化角色。

重新初始化步骤：

1. 登录RAM控制台。
2. 在RAM角色管理页面底部的RAM角色名称列中，单击AliyunStreamConsoleRole。
3. 在AliyunStreamConsoleRole页面的权限管理页签中，单击AliyunStreamConsoleRolePolicy操作列下的移除权限。

 说明 如果您已经删除AliyunStreamConsoleRolePolicy，则可跳过此步骤。

4. 单击AliyunStreamConsoleRole页面底部的RAM角色管理。
5. 在RAM角色管理页面底部，单击AliyunStreamConsoleRole的操作列下的删除。
6. 登录实时计算统一控制台。
7. 单击前往授权。
8. 单击同意授权。

9. 通过手机短信等验证方式，完成角色授权。

**说明** 对于授权，实时计算和RAM均有不同程度的缓存，有可能在删除角色或者授权角色后，实时计算仍然提示需要授权角色。此时系统已经完成授权任务，单击授权完毕即可。

## 4.2. RAM账号授权

本文为您介绍如何在Flink半托管/ACK开发控制台上，为阿里云RAM账号授权。

### 背景信息

如果您的RAM账号没有被主账号授权，当您进入到Flink半托管/ACK开发控制台后，会看到403信息。



### 操作步骤

1. 登录[实时计算控制台](#)。
2. 在Flink半托管/ACK页签，单击对应工作空间操作列下的控制台。
3. 在左侧导航栏上，单击系统管理 > 成员管理。
4. 单击添加成员。



5. 选择角色，填写成员信息。

- 角色

不同角色功能差异如下。

功能	owner	editor	viewer
查看作业列表	Y	Y	Y
启停作业	Y	Y	N
更改作业配置	Y	Y	N
上传JAR	Y	Y	N
编写SQL	Y	Y	N
注册UDF	Y	Y	N
注册元数据	Y	Y	N
查看作业模板	Y	Y	Y
增删改作业模板	Y	Y	N
管理成员	Y	N	N

- 成员

固定格式为 `user:UID`。其中：

- user：固定值，不可修改。

- UID: 阿里云RAM账号。

 **注意** 英文冒号（:）和UID之间不能有空格，否则授权无效。

6. 单击**确定**。

## 5. 开通流程

本文为您介绍Flink半托管/ACK的开通流程。

### 前提条件

- 完成Flink半托管/ACK角色授权。详情请参见[角色授权](#)。
- OSS路径用来存放Flink作业中生成的Savepoint或Checkpoint。如果您还没有开通OSS服务，请先完成开通，详情请参见[开通OSS服务](#)。
- 已创建专有网络VPC和虚拟交换机。详情请参见[创建专有网络](#)和[创建交换机](#)。

### 操作步骤

1. 登录[实时计算统一控制台](#)。
2. 单击Flink半托管/ACK下的购买产品。
3. 在基本配置区域，选择您的地域和可用区。
4. 在网络设置区域，填写您的专有网络、虚拟交换机和网络插件等信息。
5. 在集群配置区域，填写集群名称，并选择组件、实例规格和存储配置等信息。
6. 在存储配置区域，选择OSS存储信息并填写您的AccessKey Id和AccessKey Secret信息。

 **说明** 查看AccessKey Id和AccessKey Secret请参见[如何查看AccessKey ID和AccessKey Secret信息?](#)

7. 在服务说明区域，阅读服务说明，并选中我已了解和接受上述说明。
8. 单击确认订单并支付，支付成功后单击管理控制台。

## 6.作业开发

### 6.1. 开发

Flink半托管（基于ACK）产品需要您在线下完成作业开发后，再在Flink半托管（基于ACK）控制台上提交作业到集群上运行。

#### 代码开发

您可以参见以下文档开发Flink半托管（基于ACK）产品业务代码：

- Apache Flink是什么，以及它的体系架构、应用程序和特性功能等，请参见[Apache Flink介绍](#)。
- Apache Flink V1.10业务代码开发，请参见[Flink DataStream API开发指南](#)和[Flink Table API & SQL开发指南](#)。
- Apache Flink的编码、Java语言、Scala语言、组件和格式等指南，请参见[代码风格和质量指南](#)。
- Apache Flink编码过程中遇到的问题及解决方法，请参见[常见问题](#)。

#### 限制说明

由于Flink半托管（基于ACK）产品受部署环境、网络环境等因素的影响，所以开发Flink半托管（基于ACK）作业，需要注意以下限制：

- 目前仅支持开源Flink V1.10版本。
- 目前仅支持JAR形式的作业提交和运行。
- 目前仅支持运行单个JAR形式的作业。
- 不支持在Main函数中读取本地配置。
- Flink半托管（基于ACK）运行环境使用的是JDK1.8，请使用JDK1.8进行作业开发。

### 6.2. 提交

本文为您介绍如何提交Flink半托管（基于ACK）作业至集群运行。

#### 上传JAR包

1. 登录[实时计算统一控制台](#)。
2. 在Flink半托管/ACK页签，单击对应工作空间操作列下的控制台。
3. 在左侧导航栏上，单击作业管理 > Jar包管理。
4. 在页面右上角，单击单击可添加文件上传JAR包。

 说明 系统支持上传最大为50 MB的JAR包。

#### 创建作业

1. 登录[实时计算统一控制台](#)。
2. 在Flink半托管/ACK页签，单击对应工作空间操作列下的控制台。
3. 在左侧导航栏上，单击作业管理 > 创建作业。
4. 在标准页签，填写基本配置信息。

参数解释如下表所示。

配置项	说明
作业名称	作业名称
部署目标	Flink半托管（基于ACK）的部署目标，默认只有一个。
Jar URI	选择您上传的JAR包，或者输入可以被公开读的JAR包URI，此URI需要符合HTTP协议。
Entrypoint Class	程序的入口类 <div style="border: 1px solid #ccc; padding: 5px; background-color: #e6f2ff;"> <p><span style="font-size: 1em;">?</span> 说明 如果您的JAR未指定主类，请在此处输入您的Entrypoint Class类的标准路径。</p> </div>
Entrypoint main args	Main方法的参数
并行度	作业并发数

? 说明

- 您可以根据实际业务需要，在高级页签，配置Flink配置、资源和日志等信息。
- 您可以在YAML页签，直接修改配置信息。

5. 单击创建。

## 启动作业

1. 登录[实时计算统一控制台](#)。
2. 在Flink半托管/ACK页签，单击对应工作空间操作列下的控制台。
3. 在左侧导航栏上，单击作业管理 > 作业列表。
4. 单击您创建的目标作业名称。
5. 在作业详情页面，单击启动，即可将作业提交至集群运行。

# 7. 运维管理

## 7.1. 作业运维

本文为您介绍如何查看作业的详情和状态，以及如何修改作业状态和配置信息。

### 查看作业详情

1. 登录[实时计算统一控制台](#)。
2. 在Flink半托管/ACK页签，单击对应工作空间操作列下的控制台。
3. 在左侧导航栏上，单击作业管理 > 作业列表。
4. 单击目标作业名称。
5. 在作业详情页面，查看作业详情。

- 总览：作业参数总览。



- 拓扑图：作业运行的拓扑信息。



- 运行事件：作业运行中的事件。



运行事件页面上，默认展示消息发生的时间、消息内容和操作三部分信息，其中具体操作包括以下操作：

- 复制消息：复制消息内容。
- 实例详情：查看作业产生事件的详细信息，包括配置、资源（CPU和内存）、作业运行参数、JAR包参数和Savepoint信息等。

#### 说明

- 单击Flink UI，可以跳转到运行作业的开源Flink UI界面。
- 单击指标，可以跳转到运行作业的VVP（Veriverica Platform）的指标界面。指标界面默认展示的是从事件发生时间起30分钟内的指标信息。

- 仅显示此实例相关事件：展示当前事件对应的作业实例运行期间的事件。

- 实例：展示作业的历史信息，包括作业ID、开始时间、作业状态和重启次数等。



### 查看作业状态

1. 登录[实时计算统一控制台](#)。
2. 在Flink半托管/ACK页签，单击对应工作空间操作列下的控制台。
3. 在左侧导航栏上，单击作业管理 > 作业列表。
4. 单击目标作业名称。
5. 在作业总览页签，查看作业状态信息。页面右上角会显示两种作业状态：
  - 期望状态

状态	说明
RUNNING	按照作业配置定义，启动作业。
CANCELLED	停止任何正在运行的作业。
SUSPENDED	先对当前运行的作业创建Savepoint，再停止作业。

#### ○ 当前状态

状态	说明
RUNNING	按照作业配置定义，启动作业。
CANCELLED	停止任何正在运行的作业。
SUSPENDED	先对当前运行的作业创建Savepoint，再停止作业。
FINISHED	作业运行完成。
TRANSITIONING	作业处于向期望状态的过度阶段。
FAILED	作业向期望状态转换失败，需要手动处理。

**说明** 单击启动后，作业的期望状态会立刻变成RUNNING，当前状态会变成TRANSITIONING，并显示作业启动进度。前端开发平台VVP（Veriverica Platform）会保证Flink作业最终达到期望状态。如果Flink作业状态正在从某个状态变成期望状态的过程中，而作业的最终期望状态发生了改变，作业会先到达最初的期望状态，再转换为最终的期望状态。

## 修改作业状态

1. 登录[实时计算统一控制台](#)。
2. 在Flink半托管/ACK页签，单击对应工作空间操作列下的控制台。
3. 在左侧导航栏上，单击[作业管理](#) > [作业列表](#)。
4. 单击目标作业名称。
5. 单击[总览](#)页签，进入作业详情页面。
6. 根据需要单击以下按钮，修改作业状态。



其中，每个按钮含义如下表所示。

操作名称	说明
启动	启动作业。

操作名称	说明
暂停	<p>暂停作业，先对当前运行的作业创建Savepoint，再停止作业。暂停作业需要具备以下两个条件：</p> <ul style="list-style-type: none"> <li>Savepoint地址配置完成。如果Savepoint地址没有配置，暂停操作将直接停止作业，导致作业状态丢失。</li> <li>Savepoint操作成功。</li> </ul>
停止	停止作业。
Savepoint	<p>创建一个当前时刻的Savepoint。</p> <p> 说明 请先完成Savepoint地址配置，否则Savepoint会失败。</p>
克隆	<p>复制一个作业。</p> <p> 说明</p> <ul style="list-style-type: none"> <li>复制的作业处于CANCELLED状态。</li> <li>复制作业名称为Fork of 原作业名称，例如：Fork of test2。</li> </ul>
删除	删除作业。只能删除当前状态不为RUNNING的作业。

## 修改作业配置

作业配置包含作业上线阶段所有的配置内容。

1. 登录[实时计算统一控制台](#)。
2. 在Flink半托管/ACK页签，单击对应工作空间操作列下的控制台。
3. 在左侧导航栏上，单击作业管理 > 作业列表。
4. 单击目标作业名称。
5. 修改作业配置信息。您可以通过以下任何一种方式修改作业配置：
  - 在图形化界面修改作业配置  
在页面左上角，单击配置作业后，在标准页签修改配置信息。  

  - 在YAML文件中修改作业配置  
在页面左上角，单击配置作业后，在YAML页签修改配置信息。  

6. 单击Save Changes。

## 7.2. 开源自建监控组件（ACK）

实时计算Flink半托管（基于ACK）版本兼容开源自建监控组件Prometheus。

### 开通服务

1. 登录[实时计算Flink半托管（基于ACK）控制台](#)。
2. 在[Flink半托管（基于ACK）](#)页签对应集群操作列，单击组件后，然后单击添加/修改组件，添加报警组件。

 **注意** 自建报警仅适用于vwp-2.1.0及以上版本，如果当前版本为vwp-1.5.1，请升级到vwp-2.1.0。

3. 登录[阿里云ACK控制台](#)后，进入Prometheus pod内部控制台。

Prometheus pod内部控制台界面如下图所示。

4. 在Prometheus pod内部控制台上，输入pwd，确认当前路径为/Prometheus。
5. 在Prometheus pod内部控制台上，输入id，确认当前登录用户信息为uid=65534(nobody) gid=65534(nogroup) groups=65534(nogroup)。
6. 创建文件夹rules、文件alerting rules（yaml格式）和文件recording rules（yaml格式）。

```
# 进入默认目录。
cd /prometheus
# 创建rules文件夹。
mkdir rules
cd rules
# 创建alerting rules file和recording rules file。
# alerting rules file需要符合alerting_rule*.yaml的格式。
# recording rules file需要符合recording_rule*.yaml的格式。
touch alerting_rules-flink-job.yaml
touch recoding_rules-flink-job.yaml
```

7. 填写alerting rules和recording rules文件内容。alerting rules和recording rule语法规则相同。

```
groups:
  [- <rule_group>]
```

示例如下：

- o alerting rules

```
groups:
- name: example
  rules:
  - alert: HighRequestLatency
    expr: job:request_latency_seconds:mean5m{job="myjob"} > 0.5
    for: 10m
    labels:
      severity: page
    annotations:
      summary: High request latency
```

- o recording rule

```
groups:
- name: example
  rules:
  - record: job:http_inprogress_requests:sum
    expr: sum(http_inprogress_requests) by (job)
```

8. 执行以下命令加载prometheus的报警规则的配置信息。

```
# 发送SIGHUB信号。
kill -1 1
```

9. 查看更新后的报警规则信息。

### 配置钉钉报警

目前仅支持通过钉钉报警。

1. 在钉钉群中[添加机器人](#)。
2. 安全设置完成后，复制机器人的Webhook地址。格式为https://oapi.dingtalk.com/robot/send?access\_token=XXXXXX。
3. 在阿里云ACK控制台，找到vvp-xxx-dingtalk-alert-gateway的Deployment，选择更多 > 查看YAML。

4. 在弹出的YAML文本框中找到containers的args信息。

5. 将access\_token=替换为步骤二获得的token后，单击更新。如果有报警产生，可在钉钉内收到消息。

## 7.3. 链路监控

Flink半托管（基于ACK）模式提供全链路指标采集及展示能力。在初始化集群过程中，系统将会在您的K8s集群上部署监控组件，包括Prometheus、Push Gateway、Alert Manager、InfluxDB和Grafana。监控组件会占用一定的K8s资源。目前监控组件默认的资源配置能够支持100Core以内的Flink作业的正常运行。

## 指标写入

通过开发控制台提交的作业，将自动完成监控链路的相关配置，您无需进行额外配置，作业即可写入至部署的监控链路中。您可以通过App Manager上的Metrics模块查看监控信息。Flink半托管（基于ACK）模式的监控链路运行原理如下图所示。

□

## 指标查看

□

Grafana的前端界面已经整合至AppManager中，您可以在App Manager的Deployments > Metric详情界面查看监控信息。

# 7.4. 配置及查看作业日志

本文为您介绍如何配置作业日志输出到对象存储OSS和日志服务SLS，以及如何查看作业运行日志。

## 背景信息

在作业正常运行状态下，如果您需要查看当前作业运行日志，您可以单击Flink UI进行查看；如果出现以下情况，则需要您配置作业运行日志到外部存储（OSS或SLS）后，启动作业：

- 因为Job Manager无法正常启动导致Flink UI无法打开，需要查看Job Manager日志定位问题。
- 您需要查看、搜索和分析作业历史运行日志（系统默认保留最近5 MB大小的运行日志）。

## 查看作业运行日志

1. 登录[实时计算控制台](#)。
2. 在Flink半托管/ACK页签，单击对应工作空间操作列下的控制台。
3. 在左侧导航栏上，单击作业管理 > 作业列表。
4. 单击您创建的目标作业名称。

 **说明** 请选择作业期望状态和当前状态都为RUNNING的作业。

5. 单击Flink UI。
6. 在左侧导航栏，单击Job Manager。
7. 在Logs页签，查看作业运行日志。

## 配置单个作业（新建作业）日志输出到OSS

1. 登录[实时计算控制台](#)。
2. 在左侧导航栏上，单击作业管理 > 创建作业。
3. 在高级页签，填写作业基本信息、行为和配置等信息，详情请参见[提交](#)。
4. 日志模板选择为OSS。
5. 在页面右下角，单击创建。
6. 在左侧导航栏上，单击作业管理 > 作业列表。
7. 单击目标作业名称。
8. 在作业详情页面，单击启动。
9. 在OSS控制台，查看日志信息。
  - i. 登录[OSS管理控制台](#)。
  - ii. 在左侧导航栏，单击Bucket列表。

- iii. 单击您开通Flink全托管服务时，配置的Bucket名称。
- iv. 单击文件管理。
- v. 在对应目录，查看日志信息。

文件夹	说明
artifacts	上传的JAR包会被存储到该目录。
flink-jobs	Flink作业的Checkpoint会被存储到该目录。
flink-savepoints	在Flink半托管/ACK开发控制台上，单击Savepoint，会触发Savepoint操作，最终的Savepoint文件会被存储到该目录。
logs	<p>如果您作业的日志模板选择了OSS，则您的作业的日志会被存储到该目录。</p> <div style="border: 1px solid #ccc; background-color: #e6f2ff; padding: 5px; margin-top: 10px;"> <p><span style="color: #00aaff;">?</span> 说明 <code>logs</code>目录下作业存储路径为：<code>logs/\${workspaceID}/\${jobID}/\${instanceID}</code>。您可以通过该目录结构找到相应作业的JM（Job Manager）和TM（Task Manager）日志。</p> </div>

### 配置单个作业（已有作业）日志输出到OSS

1. 登录实时计算控制台。
2. 在左侧导航栏上，单击作业管理 > 作业列表。
3. 单击目标作业名称。
4. 在作业详情页面，单击配置作业。
5. 在高级页签，日志模板选择为OSS。
6. 单击Save Changes。
7. 在左侧导航栏上，单击作业管理 > 作业列表。
8. 单击目标作业名称。
9. 在作业详情页面，单击启动。
10. 在对象存储OSS控制台，查看日志信息。
  - i. 登录OSS管理控制台。
  - ii. 在导航栏，单击Bucket列表。
  - iii. 单击您开通Flink半托管/ACK服务时，配置的Bucket名称。
  - iv. 单击文件管理。

v. 在对应目录，查看日志信息。

文件夹	说明
artifacts	上传的JAR包会被存储到该目录。
flink-jobs	Flink作业的Checkpoint会被存储到该目录。
flink-savepoints	在Flink半托管/ACK开发控制台上，单击Savepoint，会触发Savepoint操作，最终的Savepoint文件会被存储到该目录。
logs	<p>如果您作业的日志模板选择了OSS，则您的作业的日志会被存储到该目录。</p> <div style="border: 1px solid #ccc; padding: 5px; background-color: #e6f2ff;"> <p><span style="color: #00aaff;">?</span> 说明 logs目录下作业存储路径为：logs/{[工作空间ID]}/{[作业ID]}/{[实例ID]}。您可以通过该目录结构找到相应作业的JM（Job Manager）和TM（Task Manager）日志。</p> </div>

### 配置工作空间下所有作业日志输出到OSS

您可以通过配置作业模板的方式，配置工作空间下所有作业日志默认输出到OSS。

1. 登录[实时计算控制台](#)。
2. 在左侧导航栏上，单击系统管理 > 作业模板。
3. 在高级页签，日志模板选择为OSS。
4. 单击保存更改。

? 说明 配置作业模板后，后续该工作空间下创建的所有作业的日志都会被存储到OSS。

### 配置单个作业（新建作业）日志输出到SLS

1. 登录[实时计算控制台](#)。
2. 在左侧导航栏上，单击作业管理 > 创建作业。
3. 在高级页签，填写作业基本信息、行为和配置等信息，详情请参见[作业提交](#)。
4. 日志模板选择为自定义。
5. 将以下文本粘贴到自定义模板的输入框中。

```
<?xml version="1.0" encoding="UTF-8"?>
<Configuration xmlns="http://logging.apache.org/log4j/2.0/config"
strict="true" packages="com.ververica.platform.logging.appender" status="WARN">
  <Appenders>
    <Appender name="StdOut" type="Console">
      <Layout pattern="%d{yyyy-MM-dd HH:mm:ss,SSS}{GMT+8} %-5p %-60c %x - %m%n" type="PatternLayout"/>
    </Appender>
    <Appender name="RollingFile" type="RollingFile" fileName="${sys:log.file}" filePattern="${sys:log.file}.%i">
  </Configuration>
```

```

<Layout pattern="%d{yyyy-MM-dd HH:mm:ss,SSS}{GMT+8} %-5p %-60c %x - %m%n" type="PatternLayout" />
<Policies>
  <SizeBasedTriggeringPolicy size="5 MB" />
</Policies>
<DefaultRolloverStrategy max="1" />
</Appender>
<Appender name="SLS" type="SLS">
  <Layout pattern="%d{yyyy-MM-dd HH:mm:ss,SSS}{GMT+8} %-5p %-60c %x - %m%n" type="PatternLayout" />

  <!-- The final effective log path is: ${baseUri}/logs/${namespace}/${deploymentId}/${jobId}/ -->
  <Property name="namespace">{{ namespace }}</Property> <!-- Do not modify this line -->
  <Property name="project">YOUR-SLS-PROJECT</Property>
  <Property name="logStore">YOUR-SLS-LOGSTORE</Property>
  <Property name="endpoint">YOUR-SLS-ENDPOINT</Property>
  <Property name="accessKeyId">YOUR-SLS-ACCESSKEYID</Property>
  <Property name="accessKeySecret">YOUR-SLS-ACCESSKEYSECRET</Property>
  <Property name="topic">{{ namespace }}:{{ deploymentId }}:{{ jobId }}</Property>
  <Property name="flushIntervalSeconds">10</Property>
  <Property name="flushIntervalEventCount">100</Property>
</Appender>
</Appenders>
<Loggers>
  <Logger level="INFO" name="org.apache.hadoop" />
  <Logger level="INFO" name="org.apache.kafka" />
  <Logger level="INFO" name="org.apache.zookeeper" />
  <Logger level="INFO" name="akka" />
  <Logger level="ERROR" name="org.jboss.netty.channel.DefaultChannelPipeline" />
  <Logger level="OFF" name="org.apache.flink.runtime.rest.handler.job.JobDetailsHandler" />
  {% for name, level in userConfiguredLoggers -%}
  <Logger level="{{ level }}" name="{{ name }}" />
  {% endfor -%}
  <Root level="{{ rootLoggerLogLevel }}">
    <AppenderRef ref="StdOut" />
    <AppenderRef ref="RollingFile" />
    <AppenderRef ref="SLS" />
  </Root>
</Loggers>
</Configuration>

```

6. 在自定义模板输入框中，修改以下5个参数。

文件夹	说明
YOUR-SLS-PROJECT	替换成您SLS的Project名称。
YOUR-SLS-LOGSTORE	替换成您SLS的Logstore名称。
YOUR-SLS-ENDPOINT	替换成您SLS所在地域的Endpoint，详情请参见 <a href="#">服务入口</a> 。
YOUR-SLS-ACCESSKEYID	替换成您开通Flink半托管/ACK服务账号的AccessKey ID。
YOUR-SLS-ACCESSKEYSECRET	替换成您开通Flink半托管/ACK服务账号的AccessKey Secret。

#### 说明

- 请确保您所使用的AccessKey ID和AccessKey Secret是您开通Flink半托管/ACK所使用账号下管理的密钥对。
- 请确保您开通Flink半托管/ACK服务所使用的账号，有读写该SLS项目的权限。

- 在页面右下角，单击**创建**。
- 在左侧导航栏上，单击**作业管理 > 作业列表**。
- 单击目标作业名称。
- 在作业详情页面，单击**启动**。

## 配置单个作业（已有作业）日志输出到SLS

- 登录[实时计算控制台](#)。
- 在左侧导航栏上，单击**作业管理 > 作业列表**。
- 单击目标作业名称。
- 在作业详情页面，单击**配置作业**。
- 在高级页签，日志模板选择为**自定义模板**。
- 将以下文本粘贴到自定义模板的输入框中。

```
<?xml version="1.0" encoding="UTF-8"?>
<Configuration xmlns="http://logging.apache.org/log4j/2.0/config"
strict="true" packages="com.ververica.platform.logging.appender" status="WARN">
  <Appenders>
    <Appender name="StdOut" type="Console">
      <Layout pattern="%d{yyyy-MM-dd HH:mm:ss,SSS}{GMT+8} %-5p %-60c %x - %m%n" type="PatternLayout"/>
    </Appender>
    <Appender name="RollingFile" type="RollingFile" fileName="${sys:log.file}" filePattern="${sys:log.file}.%i"
  >
    <Layout pattern="%d{yyyy-MM-dd HH:mm:ss,SSS}{GMT+8} %-5p %-60c %x - %m%n" type="PatternLayout"/>
  </Appender>
  <Policies>
    <SizeBasedTriggeringPolicy size="5 MB"/>
  </Policies>
</Configuration>
```

```

    <DefaultRolloverStrategy max="1"/>
  </Appender>
  <Appender name="SLS" type="SLS">
    <Layout pattern="%d{yyyy-MM-dd HH:mm:ss,SSS}{GMT+8} %-5p %-60c %x - %m%n" type="PatternLayout"/>

    <!-- The final effective log path is: ${baseUri}/logs/${namespace}/${deploymentId}/${jobId}/ -->
    <Property name="namespace">{{ namespace }}</Property> <!-- Do not modify this line -->
    <Property name="project">YOUR-SLS-PROJECT</Property>
    <Property name="logStore">YOUR-SLS-LOGSTORE</Property>
    <Property name="endpoint">YOUR-SLS-ENDPOINT</Property>
    <Property name="accessKeyId">YOUR-SLS-ACCESSKEYID</Property>
    <Property name="accessKeySecret">YOUR-SLS-ACCESSKEYSECRET</Property>
    <Property name="topic">{{ namespace }}:{{ deploymentId }}:{{ jobId }}</Property>
    <Property name="flushIntervalSeconds">10</Property>
    <Property name="flushIntervalEventCount">100</Property>
  </Appender>
</Appenders>
<Loggers>
  <Logger level="INFO" name="org.apache.hadoop"/>
  <Logger level="INFO" name="org.apache.kafka"/>
  <Logger level="INFO" name="org.apache.zookeeper"/>
  <Logger level="INFO" name="akka"/>
  <Logger level="ERROR" name="org.jboss.netty.channel.DefaultChannelPipeline"/>
  <Logger level="OFF" name="org.apache.flink.runtime.rest.handler.job.JobDetailsHandler"/>
  {%- for name, level in userConfiguredLoggers -%}
    <Logger level="{{ level }}" name="{{ name }}" />
  {%- endfor -%}
  <Root level="{{ rootLoggerLogLevel }}">
    <AppenderRef ref="StdOut"/>
    <AppenderRef ref="RollingFile"/>
    <AppenderRef ref="SLS"/>
  </Root>
</Loggers>
</Configuration>

```

7. 在自定义模板输入框中，修改以下5个参数。

文件夹	说明
YOUR-SLS-PROJECT	替换成您SLS的Project名称。
YOUR-SLS-LOGSTORE	替换成您SLS的Logstore名称。

文件夹	说明
YOUR-SLS-ENDPOINT	替换成您SLS所在地域的Endpoint，详情请参见 <a href="#">服务入口</a> 。
YOUR-SLS-ACCESSKEYID	替换成您开通Flink半托管/ACK服务账号的AccessKey ID。
YOUR-SLS-ACCESSKEYSECRET	替换成您开通Flink半托管/ACK服务账号的AccessKey Secret。

#### 说明

- 请确保您所使用的AccessKey ID和AccessKey Secret是您开通Flink半托管/ACK所使用账号下管理的密钥对。
- 请确保您开通Flink半托管/ACK服务所使用的账号，有读写该SLS项目的权限。

- 单击Save Changes。
- 在左侧导航栏上，单击作业管理 > 作业列表。
- 单击目标作业名称。
- 在作业详情页面，单击启动。

## 配置工作空间下所有作业日志输出到SLS

您可以通过配置作业模板的方式，配置工作空间下所有作业日志默认输出到SLS。

- 登录[实时计算控制台](#)。
- 在左侧导航栏，单击系统管理 > 作业模板
- 在高级页签，日志模板选择为自定义模板。
- 将以下文本粘贴到自定义模板的输入框中。

```
<?xml version="1.0" encoding="UTF-8"?>
<Configuration xmlns="http://logging.apache.org/log4j/2.0/config"
strict="true" packages="com.ververica.platform.logging.appender" status="WARN">
  <Appenders>
    <Appender name="StdOut" type="Console">
      <Layout pattern="%d{yyyy-MM-dd HH:mm:ss,SSS}{GMT+8} %-5p %-60c %x - %m%n" type="PatternLayout" />
    </Appender>
    <Appender name="RollingFile" type="RollingFile" fileName="${sys:log.file}" filePattern="${sys:log.file}.%i"
    >
      <Layout pattern="%d{yyyy-MM-dd HH:mm:ss,SSS}{GMT+8} %-5p %-60c %x - %m%n" type="PatternLayout" />
    </Appender>
    <Policies>
      <SizeBasedTriggeringPolicy size="5 MB"/>
    </Policies>
    <DefaultRolloverStrategy max="1"/>
  </Appenders>
  <Appender name="SLS" type="SLS">
    <Layout pattern="%d{yyyy-MM-dd HH:mm:ss,SSS}{GMT+8} %-5p %-60c %x - %m%n" type="PatternLayout" />
  </Appender>
</Configuration>
```

```

t"/>

<!-- The final effective log path is: ${baseUri}/logs/${namespace}/${deploymentId}/${jobId}/ -->
<Property name="namespace">{{ namespace }}</Property> <!-- Do not modify this line -->
<Property name="project">YOUR-SLS-PROJECT</Property>
<Property name="logStore">YOUR-SLS-LOGSTORE</Property>
<Property name="endpoint">YOUR-SLS-ENDPOINT</Property>
<Property name="accessKeyId">YOUR-SLS-ACCESSKEYID</Property>
<Property name="accessKeySecret">YOUR-SLS-ACCESSKEYSECRET</Property>
<Property name="topic">{{ namespace }}:{{ deploymentId }}:{{ jobId }}</Property>
<Property name="flushIntervalSeconds">10</Property>
<Property name="flushIntervalEventCount">100</Property>
</Appender>
</Appenders>
<Loggers>
<Logger level="INFO" name="org.apache.hadoop"/>
<Logger level="INFO" name="org.apache.kafka"/>
<Logger level="INFO" name="org.apache.zookeeper"/>
<Logger level="INFO" name="akka"/>
<Logger level="ERROR" name="org.jboss.netty.channel.DefaultChannelPipeline"/>
<Logger level="OFF" name="org.apache.flink.runtime.rest.handler.job.JobDetailsHandler"/>
{%- for name, level in userConfiguredLoggers -%}
  <Logger level="{{ level }}" name="{{ name }}" />
{%- endfor -%}
<Root level="{{ rootLoggerLogLevel }}">
  <AppenderRef ref="StdOut"/>
  <AppenderRef ref="RollingFile"/>
  <AppenderRef ref="SLS"/>
</Root>
</Loggers>
</Configuration>

```

5. 在自定义模板输入框中，修改以下5个参数。

文件夹	说明
YOUR-SLS-PROJECT	替换成您SLS的Project名称。
YOUR-SLS-LOGSTORE	替换成您SLS的Logstore名称。
YOUR-SLS-ENDPOINT	替换成您SLS所在地域的Endpoint，详情请参见 <a href="#">服务入口</a> 。
YOUR-SLS-ACCESSKEYID	替换成您开通Flink半托管/ACK服务账号的AccessKey ID。
YOUR-SLS-ACCESSKEYSECRET	替换成您开通Flink半托管/ACK服务账号的AccessKey Secret。

② 说明

- 请确保您所使用的AccessKey ID和AccessKey Secret是您开通Flink半托管/ACK所使用账号下管理的密钥对。
- 请确保您开通Flink半托管/ACK服务所使用的账号，有读写该SLS项目的权限。

6. 单击保存更改。

② 说明 配置作业模板后，后续该工作空间下创建的所有作业的日志都会被存储到您配置的SLS项目下的Logstore。

## 8. 常见问题

本文汇总了Flink半托管（基于ACK）常见报错与解决方案。

- [VVP上传JAR文件失败，报错undefined，应该如何处理？](#)
- [VVP上启动作业后，运行失败，应该如何处理？](#)
- [自建Ingress如何打通Flink半托管（基于ACK）？](#)
- [如何安装Ingress组件？](#)
- [如何创建命名空间？](#)
- [报错：com.alibaba.gemini.engine.exceptions.GeminiIllegalArgumentException](#)
- [报错：err client ip is not in whitelist](#)
- [报错：Service role not attach](#)

### VVP上传JAR文件失败，报错undefined，应该如何处理？

- 报错详情

VVP上上传JAR文件失败，报错undefined: undefined。



- 报错原因

- 上传的JAR文件大小超出了50MB的限制，导致报错。
- Kubernetes集群的ingress-controller把文件大小限制在了20MB。

- 解决方法

- 您可以在[OSS控制台页面](#)上传，或者直接使用[OSS命令行工具](#)上传JAR文件。
- 修改kube-system目录下的nginx-configuration文件的proxy-body-size参数，例如设置为50MB。修改后，重启ingress-controller，同时删除kube-system下的nginx-ingress-controller pod，即可解决问题。

### VVP上启动作业后，运行失败，应该如何处理？

- 问题描述

VVP上启动作业后，运行失败。

- 排查思路

- i. 在VVP控制台上，查看作业对应的实例ID。



- ii. 在pod对应的物理机上执行如下命令：

- a. 查看作业的pod状态

```
kubectl get pod -n namespace
```

说明 namespace为作业deploymentTarget里面配置的namespace。

## b. 查看对应pod的日志

- 查看pod的jobmanager日志。

```
kubectll logs job-jobs ID-jobmanager-xxx -n namespace
```

- 查看pod的Taskmanager日志。

```
kubectll logs job-jobs ID-taskmanager-xxx -n namespace
```

## 自建Ingress如何打通Flink半托管（基于ACK）？

- 报错详情

在创建Kubernetes集群时，绑定了自建的Ingress后，在Flink半托管页签，单击开发控制台，出现如下报错。



- 报错原因

vvp-xxx namespace下面的Ingress默认使用alicontainer提供的域名。

- 解决方法

如果您需要使用自建的ingress-controller，请您[提交工单](#)，要求指定集群的alicontainer域名为自建的ingress-controller。

**说明**

- 需要您提前申请负载均衡SLB，SLB必须绑定公网IP。
- 提交工单时，产品名称请选择容器服务。
- 工单描述建议写为：创建了一套nginx-ingress-controller，集群默认的xxx.alicontainer.com域名解析的IP失效，需要重新订正。

## 如何安装Ingress组件？

- 报错详情

在Flink半托管（基于ACK）页签，单击开发控制台，出现如下报错。



- 报错原因

在创建K8s集群时，没有选择Ingress。

- 解决方法

- 在容器服务控制台集群列表页面，单击目标集群操作列中的管理。



- 在组件管理页面，找到Ingress组件后，单击安装。



- 在容器服务控制台左侧菜单栏，单击无状态后，选择集群，且命名空间选择kube-system。



如果出现Ingress组件，则表示组件安装成功。



## 如何创建命名空间？

您可以通过以下步骤创建命名空间：

1. 在容器服务控制台左侧菜单栏，单击命名空间，选择VVP集群后，查看VVP自动创建的K8s命名空间。



2. 在本地新建一个TXT文件，复制如下代码后，替换{{ original-namespace }}和{{ target-namespace }}值。

### 说明

- 将{{ original-namespace }}替换为步骤1获取到的K8s命名空间。
- 将{{ target-namespace }}替换为步骤4中的目标命名空间。

```
apiVersion: rbac.authorization.k8s.io/v1beta1
kind: Role
metadata:
  name: vvp-{{ target-namespace }}-role
rules:
- apiGroups: ["apps", "extensions"]
  resources: ["deployments", "daemonsets"]
  verbs: ["create", "delete", "get", "list", "patch", "update", "watch"]
- apiGroups: [""]
  resources: ["configmaps", "pods", "services", "secrets", "serviceaccounts"]
  verbs: ["create", "delete", "get", "list", "patch", "update", "watch"]
- apiGroups: ["batch"]
  resources: ["jobs"]
  verbs: ["create", "delete", "get", "list", "patch", "update", "watch"]
- apiGroups: ["rbac.authorization.k8s.io"]
  resources: ["roles", "rolebindings"]
  verbs: ["create", "delete", "get", "list", "patch", "update", "watch"]
---
apiVersion: rbac.authorization.k8s.io/v1beta1
kind: RoleBinding
metadata:
  name: vvp-{{ target-namespace }}-rolebinding
subjects:
- kind: ServiceAccount
  name: {{ original-namespace }}-core
  namespace: {{ original-namespace }}
roleRef:
  kind: Role
  name: vvp-{{ target-namespace }}-role
apiGroup: rbac.authorization.k8s.io
```

3. 在容器服务控制台左侧菜单栏，选择应用 > 无状态后，单击页面右上角的使用模板创建。

4. 在使用模板创建详情页面，选择正确的集群名称和命名空间后，使用步骤2编辑好的文本替换模板中已有的内容。

5. 在页面右下角，单击创建。

**报错：****com.alibaba.gemini.engine.exceptions.GeminiIllegalArgumentException**

## • 报错详情

Caused by: com.alibaba.gemini.engine.exceptions.GeminiIllegalArgumentException: Gemini needs more native (off-heap) memory for better performance. The memory lower bound is 268435456 bytes, while current configured memory is only 192937984 bytes. The recommended native memory size is 2GB. Please contact dev if you have any question.

at com.alibaba.gemini.engine.dbms.SupervisorImpl.<init>(SupervisorImpl.java:174)

at com.alibaba.gemini.engine.GeminiDB.<init>(GeminiDB.java:145)

at com.alibaba.flink.statebackend.gemini.GeminiKeyedStateBackendBuilder.build(GeminiKeyedStateBackendBuilder.java:166)

at com.alibaba.flink.statebackend.gemini.GeminiStateBackend.createKeyedStateBackend(GeminiStateBackend.java:404)

at org.apache.flink.streaming.api.operators.StreamTaskStateInitializerImpl.lambda\$keyedStatedBackend\$1(StreamTaskStateInitializerImpl.java:288)

at org.apache.flink.streaming.api.operators.BackendRestorerProcedure.attemptCreateAndRestore(BackendRestorerProcedure.java:142)

at org.apache.flink.streaming.api.operators.BackendRestorerProcedure.createAndRestore(BackendRestorerProcedure.java:121)

## • 解决方法

添加配置参数：state.backend.gemini.use.offheap= false。

**报错：err client ip is not in whitelist**

## • 报错详情

## • 报错原因

外部存储的白名单中未配置Flink半托管/基于ACK集群所在交换机的IP地址。

## • 解决方法

外部存储的配置白名单中，不仅需要配置Flink半托管/基于ACK集群的IP地址，而且还需要配置Flink半托管/基于ACK集群所在交换机的IP地址。

**报错：Service role not attach**

## • 报错详情

创建集群时，报错 **Service role not attach**。

- 报错原因

您的账号不具有容器服务ACK集群角色及权限。

- 解决方法

请正确授权并检查权限，详情请参见[容器服务默认角色](#)。

 说明 角色授权后，请勿随意删除任何权限。

## 9.相关协议