

ALIBABA CLOUD

阿里云

资源编排
最佳实践

文档版本：20200903

 阿里云

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。
粗体	表示按键、菜单、页面名称等UI元素。	在结果确认页面，单击确定。
<code>Courier</code> 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。
<i>斜体</i>	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.通过ROS模板一键部署WordPress和phpMyAdmin	05
2.使用Ansible ali_ros_stack部署LNMP环境实例	11
3.使用云效流水线自动创建资源栈	22
4.使用Count功能批量创建资源	27
5.使用Count功能进行大规模部署	39

1.通过ROS模板一键部署WordPress和phpMyAdmin

本文为您介绍如何使用ROS模板一键部署WordPress站点和phpMyAdmin应用。

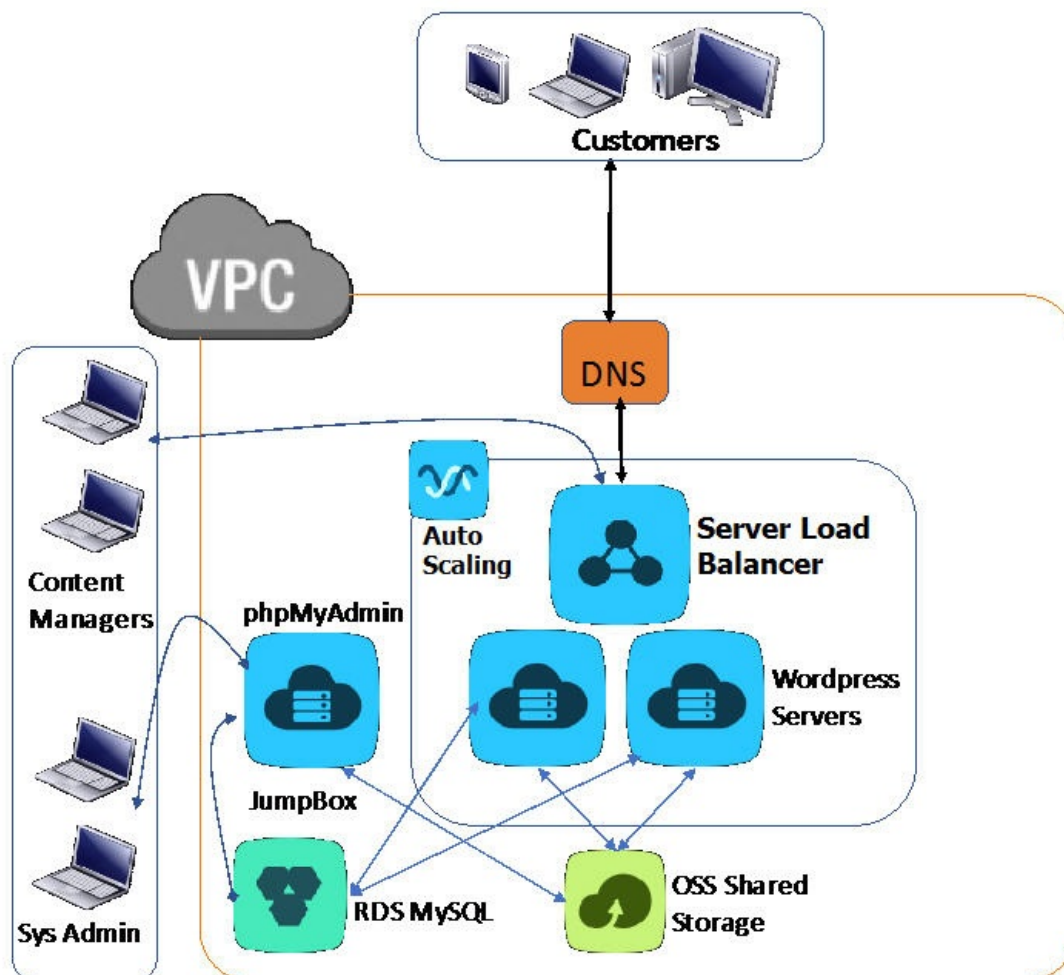
背景信息

如果您只有管理网站内容的人员，而无专业技术人员进行网站建立和管理，则只能部署基础网站。如果您的网站有更高的需求，需要定制化服务、高可用性和高弹性，则需寻求其他解决方案。

本文介绍的ROS资源栈模板（WordPressCluster-phpMyAdmin.ros）可以帮助您实现对高可用性和高弹性的需求。通过此模板，您可以快速部署整个VPC、负载均衡、弹性伸缩、ECS、云数据库RDS版等实例组成的资源栈。同时部署WordPress和phpMyAdmin，并配置弹性伸缩。系统会根据需要，自动添加、配置新的实例，无须您手动操作。

架构原理概览

下图为通过ROS资源栈模板（WordPressCluster-phpMyAdmin.ros）创建资源栈的架构图。



有三类用户将会访问这些基础设施：

- 端用户

端用户通过URL访问托管在WordPress上的网站。

WordPress部署在Apache Web服务器中。服务器的文档根目录为 `/wwwroot`。根目录所在的OSS Bucket是Web服务器通过OSSFS（阿里云官方提供的基于FUSE的文件系统）共用的一个存储空间。

RAM用户具有OSS Bucket的访问权限，可以将OSS Bucket挂载到ECS实例上。

RDS MySQL数据库存放WordPress的内容。通过内网连接字符串从Web服务器访问数据库。

- 系统管理员

系统管理员通过SSH登录JumpBox（堡垒机），进入VPC环境。

JumpBox具有弹性公网IP，可通过Internet访问。

通过JumpBox访问可管理VPC中的产品实例。

phpMyAdmin安装在JumpBox上，通过Internet访问。

如此，系统管理员便可管理云数据库RDS版。

- 内容负责人

内容负责人可通过Internet访问WordPress管理控制台。

所有服务的访问权限可通过安全组，根据环境配置来控制。

资源站模板概览

单击[WordPressCluster-phpMyAdmin.ros](#)下载该资源栈模板。

 说明 您可以根据资源编排控制台中支持的ECS可用区和镜像，在模板中修改ZoneId和ImageId。

根据WordPressCluster-phpMyAdmin.ros，系统将创建和配置VPC、负载均衡、VSwitch、NAT网关、ECS实例、弹性公网IP、弹性伸缩和云数据库RDS版实例等。

在创建资源栈时，以下参数可满足任何地域的用户需要。

ECS

Zone Id: cn-hangzhou-i
确保可用区域支持RDS、SLB和ECS资源规格的创作;
[查看可用区信息](#)

Instance Type: ecs.c5.large
请在可用区域下确认此规格的库存; 详见: [实例规格族](#)

Image Id: centos_7_04_64_20G_alibase_201701015.vhd
镜像Id, 详见: [查找镜像](#)

Disk Category: cloud_efficiency
磁盘类型, 默认为cloud_efficiency。

Data Disk Size: 100
堡垒机数据盘大小, 磁盘单位: GB, 范围: 20-32768。

Data Disk Mount Point: /data
数据盘挂载点

RDS

Instance Type: rds.mysql.s1.small
根据数据库引擎类型和可用区域支持选择实例规格;
详见: [实例规格表](#)

Engine: MySQL-5.6
数据库引擎类型, 默认MySQL。

Network type: Intranet
数据库实例网络类型, Internet用于公共访问, Intranet用于私有访问。

Instance Storage: 10
每5GB进行递增, 单位:GB

Database (DB) name: dbname
由小写字母、数字和特殊字符(-_)组成。

User Name: dbname
最多16个字符, 包括小写字母、数字、下划线、字母开头、字母或数字结尾。

根据模板，系统将在JumpBox上安装httpd、mysql-client、PHP、OSSFS、phpMyAdmin和WordPress，并通过资源ALIYUN::ECS::Instance的UserData段配置这些应用。

以下是UserData段节选：

```
"ossbucketendpoint=",  
{  
  "Ref": "OSSBucketEndPoint"
```

```
},
"\n",
"DatabaseUser=",
{
"Ref": "MasterUserName"
},
"\n",
"DatabasePwd=",
{
"Ref": "MasterDBPassword"
},
"\n",
"DatabaseName=",
{
"Ref": "DBName"
},
"\n",
"DatabaseHost=",
{
"Fn::GetAtt": ["Database", "InnerConnectionString"]
},
"\n",
"yum install -y curl httpd mysql-server php php-common php-mysql\n",
"yum install -y php-gd php-imap php-ldap php-odbc php-pear php-xml php-xmlrpc\n",
"yum install -y phpmyadmin\n",
"sed -i \"s%localhost%$DatabaseHost%\" /etc/phpMyAdmin/config.inc.php\n",
"sed -i \"s%Deny,Allow%Allow,Deny%\" /etc/httpd/conf.d/phpMyAdmin.conf\n",
"sed -i \"s%Deny from All%Allow from All%\" /etc/httpd/conf.d/phpMyAdmin.conf\n",
"sed -i \"/<RequireAny>/a Require all Granted\" /etc/httpd/conf.d/phpMyAdmin.conf\n",
"chkconfig httpd on\n",
"service httpd stop\n",
"wget
https://github.com/aliyun/ossfs/releases/download/v1.80.3/ossfs_1.80.3_centos6.5_x86_64.rpm\n",
"yum install -y ossfs_1.80.3_centos6.5_x86_64.rpm\n",
"echo $ossbucket:$ossbucketaccesskey:$ossbucketsecret >> /etc/passwd-ossfs\n",
"chmod 600 /etc/passwd-ossfs\n",
"mkdir $ossbucketmountpoint\n",
"chmod -R 755 $ossbucketmountpoint\n",
"echo #This script will automount the ossbucket\n",
"echo umount $ossbucketmountpoint >> /usr/local/bin/ossfs-automount.sh\n",
"echo #Mounting OSS Bucket\n",
"echo ossfs $ossbucket $ossbucketmountpoint -ourl=http://$ossbucketendpoint -o allow_other -o mp_um
```



```
ask=0022 -oid=48 -ogid=48 >> /usr/local/bin/ossfs-automount.sh\n",
"chmod 755 /usr/local/bin/ossfs-automount.sh\n",
"echo /usr/local/bin/ossfs-automount.sh >> /etc/rc.d/rc.local\n",
"chmod +x /etc/rc.d/rc.local\n",
"/usr/local/bin/./ossfs-automount.sh\n",
"wget http://WordPress.org/latest.tar.gz\n",
"tar -xzf latest.tar.gz\n",
"sed -i \"s%database_name_here%$DatabaseName%\" WordPress/wp-config-sample.php\n",
"sed -i \"s%username_here%$DatabaseUser%\" WordPress/wp-config-sample.php\n",
"sed -i \"s%password_here%${DatabasePwd:-$DatabasePwdDef}%\" WordPress/wp-config-sample.php\n",
"sed -i \"s%localhost%$DatabaseHost%\" WordPress/wp-config-sample.php\n",
"mv WordPress/wp-config-sample.php WordPress/wp-config.php\n",
"cp -a WordPress/* $ossbucketmountpoint\n",
"chmod -R 755 /wwwroot/*\n",
"rm -rf WordPress*\n",
"service httpd start\n",
"done\n"
```

通过UserData段，将WordPress部署在OSS Bucket上。OSS Bucket可挂载到弹性伸缩创建的Web服务器上，确保所有Web服务器都具有来自根目录的最新内容。

通过弹性伸缩配置的UserData段，实现httpd、PHP和ossutil的安装和配置、挂载DocumentRoot以及启动所有服务。

以下是弹性伸缩配置的UserData段节选：

```
"DatabaseHost=",
{
  "Fn::GetAtt": ["Database", "InnerConnectionString"]
},
"\n",
"yum install -y curl httpd mysql-server php php-common php-mysql\n",
"yum install -y php-gd php-imap php-ldap php-odbc php-pear php-xml php-xmlrpc\n",
"chkconfig httpd on\n",
"service httpd stop\n",
"DocumentRoot='/var/www/html'\n",
"sed -i \"s%$DocumentRoot%$ossbucketmountpoint%\" /etc/httpd/conf/httpd.conf\n",
"Directory='/var/www'\n",
"sed -i \"s%$Directory%$ossbucketmountpoint%\" /etc/httpd/conf/httpd.conf\n",
"wget https://github.com/aliyun/ossfs/releases/download/v1.80.3/ossfs_1.80.3_centos6.5_x86_64.rpm\n",
"yum install -y ossfs_1.80.3_centos6.5_x86_64.rpm\n",
"echo $ossbucket:$ossbucketaccesskey:$ossbucketsecret >> /etc/passwd-ossfs\n",
"chmod 600 /etc/passwd-ossfs\n",
"mkdir $ossbucketmountpoint\n",
"chmod -R 755 $ossbucketmountpoint\n",
"echo #This script will automount the ossbucket\n",
"echo umount $ossbucketmountpoint >> /usr/local/bin/ossfs-automount.sh\n",
"echo #Mounting OSS Bucket\n",
"echo ossfs $ossbucket $ossbucketmountpoint -ourl=http://$ossbucketendpoint -o allow_other -o mp_umask=0022 -oid=48 -ogid=48 >> /usr/local/bin/ossfs-automount.sh\n",
"chmod 755 /usr/local/bin/ossfs-automount.sh\n",
"echo /usr/local/bin/ossfs-automount.sh >> /etc/rc.d/rc.local\n",
"chmod +x /etc/rc.d/rc.local\n",
"/usr/local/bin/./ossfs-automount.sh\n",
"chmod -R 755 /wwwroot/*\n",
"service httpd start\n",
"done\n"
]
```

更多信息

- [创建资源栈](#)
- [通过示例模板创建资源栈](#)
- [使用可视化编辑器编写模板](#)
- [使用RAM控制资源访问](#)

2.使用Ansible ali_ros_stack部署LNMP环境实例

本文为您介绍如何通过Ansible ali_ros_stack模块调用ROS的API接口部署LNMP（Linux+Nginx+MySQL+PHP）环境实例。

前提条件


确保您已经在Linux系统使用pip3安装并配置了Ansible模块，详情请参见[安装和配置Ansible](#)。

操作步骤

1. 创建一个名称为 `create_inmp.yml` 的文件，然后通过VI编辑器打开。

```
vi create_inmp.yml
```

2. 在编辑模式下，将以下完整的playbook示例代码粘贴到 `create_inmp.yml` 文件中。

 说明 ali_ros_stack模块参数详情，请参见[参数说明](#)。

```
- hosts: localhost
remote_user: root
tasks:
- name: Create LNMP Instance
  ali_ros_stack:
  state: present
  stack_name: create_inmp_instance
  template: create_inmp_instance.json
  timeout_in_minutes: 60
  template_parameters:
  ZonId: cn-beijing-g
  ImageId: centos_7_03_64_20G_alibase_2017****.vhd
  InstancePassword: XXXXXXXX
  SystemDiskCategory: cloud_ssd
  InstanceType: ecs.c5.large
  DBName: MyDatabase
  DBUser: DefaultUser
  DBRootPassword: XXXXXX
  DBPassword: XXXXXX
  NginxDownloadUrl: http://nginx.org/packages/centos/7/noarch/RPMS/nginx-release-centos-7-0.el7.n
  gx.noarch.rpm
```

3. 保存后，退出编辑模式。
4. 创建一个名称为 `create_inmp_instance.json` 的文件，然后通过VI编辑器打开。

```
vi create_inmp_instance.json
```

5. 在编辑模式下，将以下完整的playbook示例代码粘贴到`create_inmp_instance.json`文件中。

```
{
  "Description": "Deploy LNMP(Linux+Nginx+MySQL+PHP) stack on 1 ECS instance. *** WARNING *** Only support CentOS-7.",
  "Parameters": {
    "NginxDownloadUrl": {
      "Type": "String",
      "Description": {
        "en": "The download path of nginx-*.rpm",
        "zh-cn": "nginx-*.rpm的下载路径"
      },
      "Label": "Nginx Download Url",
      "Default": "http://nginx.org/packages/centos/7/noarch/RPMS/nginx-release-centos-7-0.el7.noarch.rpm"
    },
    "DBPassword": {
      "NoEcho": true,
      "Type": "String",
      "Description": {
        "en": "The MySQL password, consisting of letters, numbers, and underline(_), 6 to 32 characters in length",
        "zh-cn": "MySQL密码，长度为6~32个字符，可包含英文字符、数字和下划线（_）。"
      },
      "Label": "DB Password",
      "ConstraintDescription": "Consisting of letters, numbers, and underline(_), 6 to 32 characters in length",
      "MinLength": 6,
      "MaxLength": 32
    },
    "ZoneId": {
      "Type": "String",
      "AssociationProperty": "ALIYUN::ECS::Instance:ZoneId",
      "Description": {
        "en": "ECS Available Zone ID,</font><a href='https://www.alibabacloud.com/help/doc-detail/123712.html' target='_blank'><b> View region and zone info</b><font color='blue'></a>",
        "zh-cn": "ECS可用区ID,</font><a href='https://help.aliyun.com/document_detail/123712.html' target='_blank'><b> 查看可用区信息</b><font color='blue'></a>"
      },
      "Label": "Available Zone ID"
    }
  }
}
```

```
"ImageId": {
  "Type": "String",
  "Description": {
    "en": "Image ID, represents the image resource to startup one ECS instance, <font><a href='https://www.alibabacloud.com/help/doc-detail/112977.html' target='_blank'><b>View image resources</b></font color='blue'></a>",
    "zh-cn": "镜像ID, ECS实例的镜像资源, <font><a href='https://help.aliyun.com/document_detail/112977.html' target='_blank'><b>查看镜像资源</b></font color='blue'></a>"
  },
  "Label": "Image ID",
  "Default": "cent****"
},
"DBName": {
  "Type": "String",
  "Description": {
    "en": "MySQL database name, [1, 64] English or Chinese characters, must start with a letter or Chinese in size, can contain numbers, '_' or '-',",
    "zh-cn": "MySQL数据库名称, 长度为1~64个英文或中文字符, 必须以英文字符或中文字符开头, 可以包含数字、下划线(_)、英文句点(.)和短划线(-)。"
  },
  "Label": "DB Name",
  "ConstraintDescription": "Must begin with a letter and contain only alphanumeric characters.",
  "MinLength": 1,
  "MaxLength": 64,
  "Default": "MyDatabase"
},
"DBUser": {
  "Type": "String",
  "Description": {
    "en": "Username for MySQL database access.It consists of lowercase letters, numbers and underscores (_), and begins with a letter. Not longer than 16 characters.",
    "zh-cn": "MySQL数据库访问的用户名。长度不超过16个字符, 必须以英文字符开头, 可包含英文字符、数字和下划线(_)。"
  },
  "Label": "DB Username",
  "ConstraintDescription": "Must begin with a letter and contain only alphanumeric characters.",
  "MinLength": 1,
  "MaxLength": 16,
  "Default": "DefaultUser"
},
"DBRootPassword": {
  "NoEcho": true,
```

```
"Type": "String",
"Description": {
  "en": "Root password for MySQL, consisting of letters, numbers, and underline(_), 6 to 32 characters in length",
  "zh-cn": "MySQL的root密码。长度为6~32个字符，可包含英文字符、数字和下划线（_）。"
},
"Label": "DB Root Password",
"ConstraintDescription": "Consisting of letters, numbers, and underline(_), 6 to 32 characters in length",
"MinLength": 6,
"MaxLength": 32
},
"InstanceType": {
  "Type": "String",
  "Description": {
    "en": "The ECS instance type, go to the product console to ensure the current instance is available, <font><a href='https://www.alibabacloud.com/help/doc-detail/25378.html' target='_blank'><b>View instance types</b></font color='blue'></a>",
    "zh-cn": "ECS实例类型，进入产品控制台确保当前实例可用，<font><a href='https://help.aliyun.com/document_detail/25378.html' target='_blank'><b>查看实例类型</b></font color='blue'></a>"
  },
  "Label": "Instance Type",
  "Default": "ecs.c5.large"
},
"SystemDiskCategory": {
  "Type": "String",
  "Description": {
    "en": "System disk category: efficient cloud disk(cloud_efficiency) or SSD cloud disk(cloud_ssd)",
    "zh-cn": "系统磁盘类别：高效云盘（cloud_efficiency）或固态硬盘（cloud_ssd）"
  },
  "AllowedValues": [
    "cloud_efficiency",
    "cloud_ssd"
  ],
  "Label": "System Disk Category",
  "Default": "cloud_ssd"
},
"InstancePassword": {
  "NoEcho": true,
  "Type": "String",
  "Description": {
    "en": "The 8-30 long login password of instance, consists of the uppercase, lowercase letter and numb
```

```
er. <br> special characters include()~!@#%&*_-+={[]:;<>,./"/,
"zh-cn": "长度为8~30个字符，必须包含大写字母、小写字母、数字或特殊符号其中三种；<br>特殊字符包括：()~!
@#%&*_-+={[]:;<>,./"/"
},
"AllowedPattern": "[0-9A-Za-z\\_\\-;<>,=%`~!@#\\(\\)\\$\\^\\*\\+\\|\\/\\{\\}\\[\\]\\\\.\\.\\.\\?\\/\\+/$",
"Label": "Instance Password",
"ConstraintDescription": "Length 8-30, must contain upper case letters, lower case letters, Numbers, s
pecial symbols three; special characters include: ()~!@#%&*_-+={[]:;<>,./"/",
"MinLength": "8",
"MaxLength": "30"
}
},
"ROSTemplateFormatVersion": "2015-09-01",
"Metadata": {
"ALIYUN::ROS::Interface": {
"ParameterGroups": [
{
"Parameters": [
"ZoneId",
"ImageId",
"InstanceType",
"SystemDiskCategory",
"InstancePassword"
],
"Label": {
"default": "ECS"
}
},
{
"Parameters": [
"DBName",
"DBUser",
"DBPassword",
"DBRootPassword"
],
"Label": {
"default": "DATABASE"
}
},
{
"Parameters": [
"NginxDownloadUrl"
1
```

```
},
"Label": {
  "default": "Nginx"
}
],
"TemplateTags": [
  "Deploy LNMP(Linux+Nginx+MySQL+PHP) stack on 1 ECS instance."
]
},
"Outputs": {
  "NginxWebsiteURL": {
    "Description": "URL for newly created Nginx home page.",
    "Value": {
      "Fn::Join": [
        "",
        [
          "http://",
          {
            "Fn::GetAtt": [
              "WebServer",
              "PublicIp"
            ]
          }
        ],
        ":80/test.php"
      ]
    }
  },
  "Resources": {
    "VSwitch": {
      "Type": "ALIYUN::ECS::VSwitch",
      "Properties": {
        "VpcId": {
          "Fn::GetAtt": [
            "Vpc",
            "VpcId"
          ]
        }
      }
    },
    "ZoneId": {
      "Ref": "ZoneId"
    }
  }
}
```



```
},
  "CidrBlock": "192.168.1.0/24"
}
},
  "WebServerConditionHandle": {
    "Type": "ALIYUN::ROS::WaitConditionHandle"
  },
  "WebServer": {
    "Type": "ALIYUN::ECS::Instance",
    "Properties": {
      "InternetMaxBandwidthOut": 80,
      "IoOptimized": "optimized",
      "VpcId": {
        "Fn::GetAtt": [
          "Vpc",
          "VpcId"
        ]
      },
      "UserData": {
        "Fn::Replace": [
          {
            "ros-notify": {
              "Fn::GetAtt": [
                "WebServerConditionHandle",
                "CurlCli"
              ]
            }
          }
        ],
        "Fn::Join": [
          "",
          [
            "#!/bin/bash \n",
            "NginxUrl=",
            {
              "Ref": "NginxDownloadUrl"
            },
            "\n",
            "dbname=",
            {
              "Ref": "DBName"
            }
          ]
        ]
      }
    }
  }
}
```

```

"\n",
"dbuser=",
{
"Ref": "DBUser"
},
"\n",
"dbpassword=",
{
"Ref": "DBPassword"
},
"\n",
"dbrootpassword=",
{
"Ref": "DBRootPassword"
},
"\n",
"export HOME=/root \n",
"export HOSTNAME=`hostname` \n",
"systemctl stop firewalld.service \n",
"systemctl disable firewalld.service \n",
"sed -i 's/^SELINUX=# SELINUX=/' /etc/selinux/config \n",
"sed -i '/# SELINUX=/a SELINUX=disabled' /etc/selinux/config \n",
"setenforce 0 \n",
"yum install yum-priorities -y \n",
"yum -y install aria2 \n",
"aria2c $NginxUrl \n",
"rpm -ivh nginx-*.rpm \n",
"yum -y install nginx \n",
"systemctl start nginx.service \n",
"systemctl enable nginx.service \n",
"yum -y install php-fpm \n",
"systemctl start php-fpm.service \n",
"systemctl enable php-fpm.service \n",
"sed -i '/FastCGI/,htaccess/s/ #/ /' /etc/nginx/conf.d/default.conf \n",
"sed -i '/FastCGI/s/^ / #' /etc/nginx/conf.d/default.conf \n",
"sed -i '/htaccess/s/^ / #' /etc/nginx/conf.d/default.conf \n",
"sed -i '/SCRIPT_FILENAME/s/\\scripts/\\usr\\share\\nginx\\html/' /etc/nginx/conf.d/default.c
onf \n",
"yum -y install mariadb mariadb-server \n",
"systemctl start mariadb.service \n",
"systemctl enable mariadb.service \n",
"yum -y install php php-mysql php-gd libjpeg* php-ldap php-odbc php-pear php-xml php-xmlrpc php-m
bstring php-bcmath php-mbstring php-mcrypt \n"

```

```

    - name: Install LNMP stack
      ansible.builtin.command:
        - mbstring php-openssl php-mbstring php-mcrypt \n ,
        - "MDSRING=`find / -name mbstring.so` \n",
        - "echo extension=$MDSRING >> /etc/php.ini \n",
        - "systemctl restart mariadb.service \n",
        - "mysqladmin -u root password \"$dbrootpassword\" \n",
        - "${mysql $dbname -u root --password=\"$dbrootpassword\" >/dev/null 2>&1 </dev/null}; (( $? != 0 )) \n",
        - "echo CREATE DATABASE $dbname \\\; > /tmp/setup.mysql \n",
        - "echo GRANT ALL ON $dbname.* TO \"$dbuser\"@\"localhost\" IDENTIFIED BY \"$dbpassword\" \\\; >> /tmp/setup.mysql \n",
        - "mysql -u root --password=\"$dbrootpassword\" < /tmp/setup.mysql \n",
        - "${mysql $dbname -u root --password=\"$dbrootpassword\" >/dev/null 2>&1 </dev/null}; (( $? != 0 )) \n",
        - "cd /root \n",
        - "systemctl restart php-fpm.service \n",
        - "systemctl restart nginx.service \n",
        - "echo \\<?php > /usr/share/nginx/html/test.php \n",
        - "echo \\$conn=mysql_connect\\(\"127.0.0.1\", \"$dbuser\", \"$dbpassword\")\\; >> /usr/share/nginx/html/test.php \n",
        - "echo if \\(\\$conn\\){ >> /usr/share/nginx/html/test.php \n",
        - "echo echo \\\"LNMP platform connect to mysql is successful\\!\\\"\\; >> /usr/share/nginx/html/test.php \n",
        - "echo }else{ >> /usr/share/nginx/html/test.php \n",
        - "echo echo \\\"LNMP platform connect to mysql is failed\\!\\\"\\; >> /usr/share/nginx/html/test.php \n",
        - "echo } >> /usr/share/nginx/html/test.php \n",
        - "echo phpinfo\\(\\)\\; >> /usr/share/nginx/html/test.php \n",
        - "echo \\?\\> >> /usr/share/nginx/html/test.php \n",
        - "ros-notify -d \"{data: 'Install LNMP stack.'}\" \n
      ]
    ]
  }
]
},
"SecurityGroupId": {
  "Ref": "SecurityGroup"
},
"VSwitchId": {
  "Ref": "VSwitch"
},
"ImageId": {
  "Ref": "ImageId"
},
},

```

```
"InstanceType": {
  "Ref": "InstanceType"
},
"SystemDiskCategory": {
  "Ref": "SystemDiskCategory"
},
"Password": {
  "Ref": "InstancePassword"
}
},
"WebServerWaitCondition": {
  "Type": "ALIYUN::ROS::WaitCondition",
  "DependsOn": "WebServer",
  "Properties": {
    "Timeout": 1800,
    "Count": 1,
    "Handle": {
      "Ref": "WebServerConditionHandle"
    }
  }
},
"Vpc": {
  "Type": "ALIYUN::ECS::VPC",
  "Properties": {
    "CidrBlock": "192.168.0.0/16"
  }
},
"SecurityGroup": {
  "Type": "ALIYUN::ECS::SecurityGroup",
  "Properties": {
    "VpcId": {
      "Ref": "Vpc"
    }
  },
  "SecurityGroupIngress": [
    {
      "PortRange": "-1/-1",
      "Priority": 1,
      "SourceCidrIp": "0.0.0.0/0",
      "IpProtocol": "all",
      "NicType": "intranet"
    }
  ]
}
```

```
],
  "SecurityGroupEgress": [
  {
    "PortRange": "-1/-1",
    "Priority": 1,
    "IpProtocol": "all",
    "DestCidrIp": "0.0.0.0/0",
    "NicType": "intranet"
  }
]
}
}
}
```

6. 保存后，退出编辑模式。
7. 运行Ansible playbook部署LNMP（Linux+Nginx+MySQL+PHP）环境实例。

```
ansible-playbook create_lnpm.yml
```

3.使用云效流水线自动创建资源栈

您可以通过云效流水线，使用ROS模块自动创建资源栈。本文以更新一个包含专有网络的ECS为例，为您介绍使用云效流水线自动创建、更新资源栈的方法。

前提条件

- 您需要创建服务角色，并对角色进行授权。
 - 请确保您已创建服务角色（例如：TestTeamRole），详情请参见[创建可信实体为阿里云服务的RAM角色](#)。
 - 请确保您已为角色进行授权，详情请参见[为RAM角色授权](#)。权限详情如下表所示。

权限策略名称	权限说明
AliyunTeambitionRolePolicy	管理云效
AliyunROSFullAccess	管理ROS
AliyunECSECSFullAccess	管理ECS
AliyunVPCFullAccess	管理VPC
AliyunRDSFullAccess	管理RDS

- 请确保您已准备代码仓库作为流水线的触发源，例如Git仓库。

背景信息

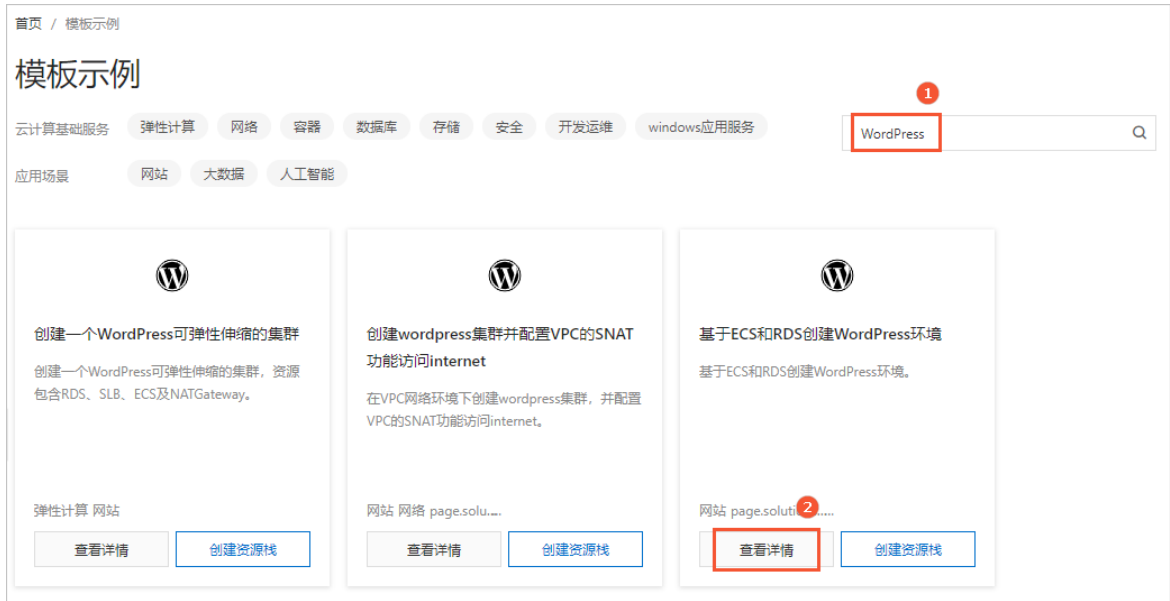
云效是企业级一站式DevOps解决方案，提供从“需求 ->开发->测试->发布->运维->运营”端到端的协同服务和研发工具。云效集成了ROS，您可以通过云效调用代码仓库中的ROS模板自动创建、更新资源栈。云效流水线详情，请参见[云效流水线](#)。

例如：您创建、更新资源栈时，每个操作都需要手动与ROS进行交互。而使用云效创建流水线，您可以建立一个发布流程，实现上述多个操作的自动化。

本示例通过将ECS和RDS创建WordPress环境模板代码上传至代码仓库，使用云效创建流水线调用该代码，实现为WordPress站点自动创建、更新资源栈。

步骤1：将模板代码上传到代码仓库

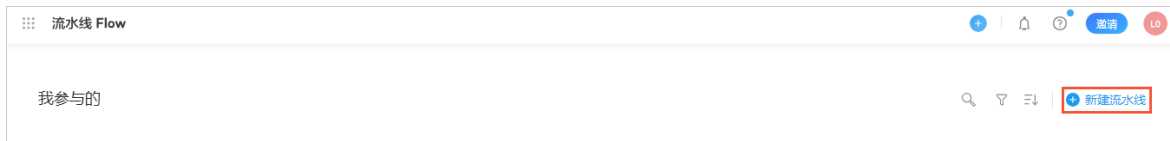
1. 登录[资源编排控制台](#)。
2. 在左侧导航栏，单击模板 > 模板示例。
3. 在文本框输入WordPress搜索示例模板基于ECS和RDS创建WordPress环境，单击查看详情。



4. 将模板代码上传到代码仓库。

步骤2：通过创建流水线自动创建资源栈

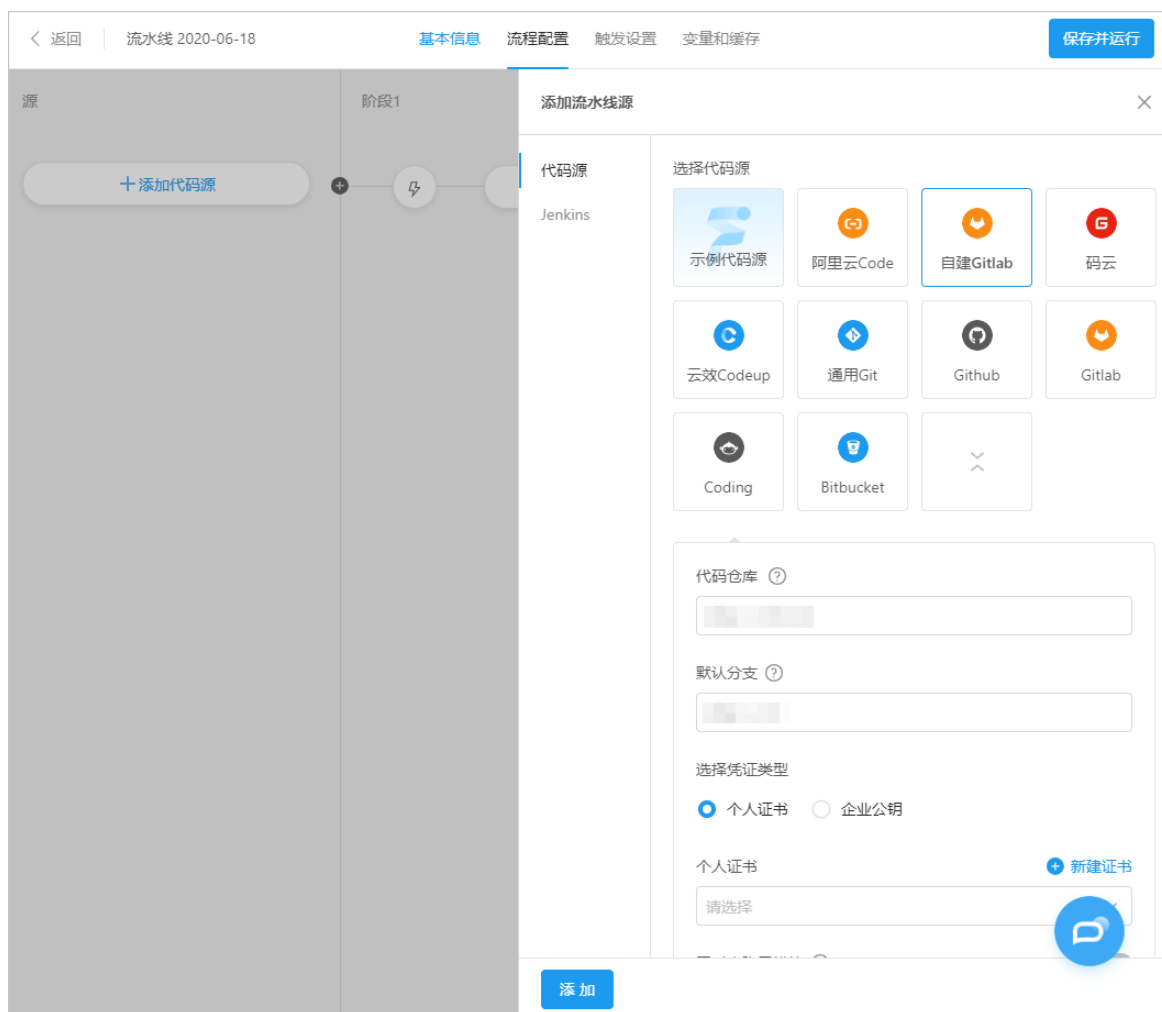
1. 登录云效。
2. 单击右上角新建流水线。



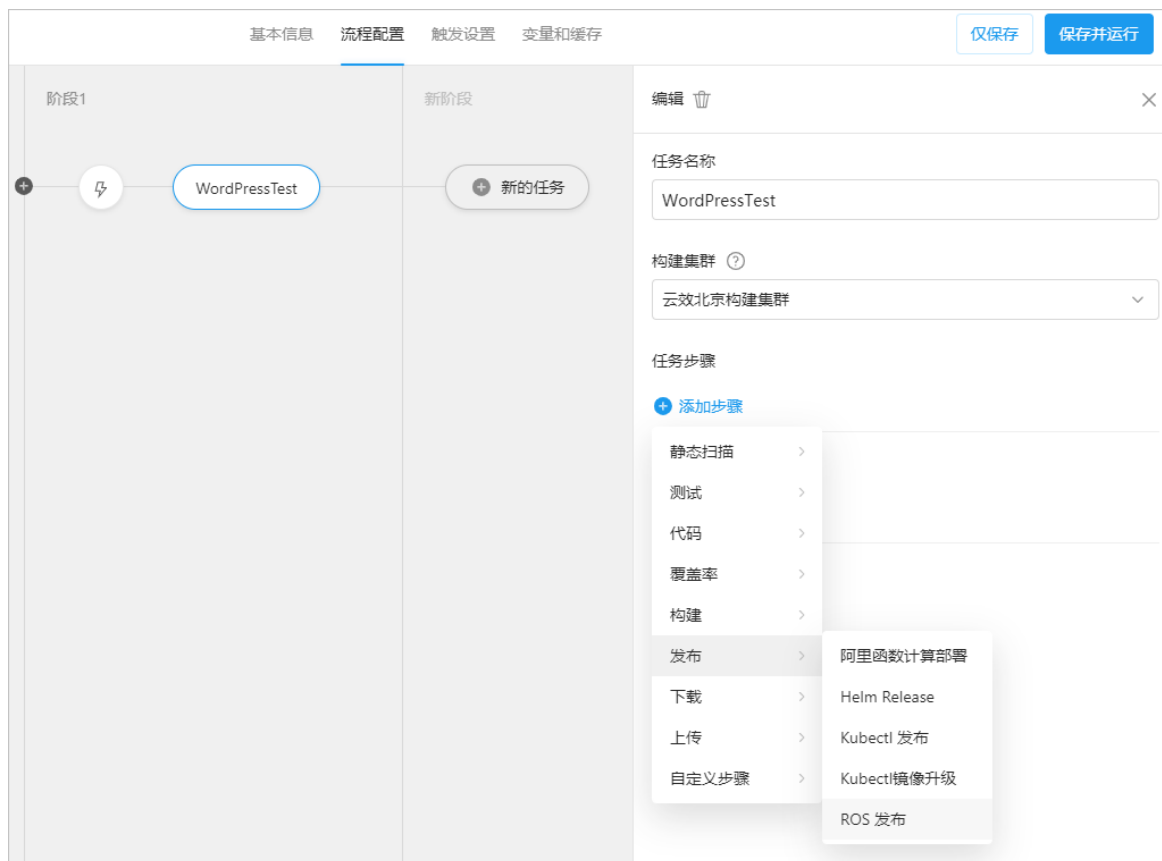
3. 在左侧导航栏单击其他，选择空模板，单击创建。



4. 在弹出的添加流水线源对话框中选择代码源，单击添加。



5. 单击空任务，在弹出的编辑对话框中输入任务名称，单击添加步骤 > 发布 > ROS发布。



6. 设置选择授权、阿里云Region、ROS扮演角色、Stack名称、Stack文件路径、超时时间，单击右上角仅保存。

说明 首次创建流水线时，单击保存并运行，在弹出的确认运行流水线对话框中单击确定，流水线将自动创建资源栈。

7. 单击新的任务，在弹出的选择任务组对话框中，单击左侧导航栏的工具。
8. 在工具区域，单击人工卡点。
9. 在弹出的编辑对话框中，设置任务名称、验证者类型和验证人。
10. 单击保存并运行，在弹出的确认运行流水线对话框中单击确定，流水线将自动创建资源栈。您可以登录资源编排控制台，单击资源栈，在资源栈列表页面查看已创建的资源栈信息。

在进入下一步任务之前，流水线有审批环节。

- 如果审批通过，您可以执行流水线的下一步任务。
- 如果审批不通过，您可以修改模板或配置文件，重新执行流水线，再次提交审批。

步骤3：更新资源栈

如果您更新了模板，并将新模板代码上传至代码仓库。使用云效流水线调用代码时，资源栈将自动更新。

4.使用Count功能批量创建资源

资源编排服务ROS（Resource Orchestration Service）支持Count功能，用于批量创建资源。

背景信息

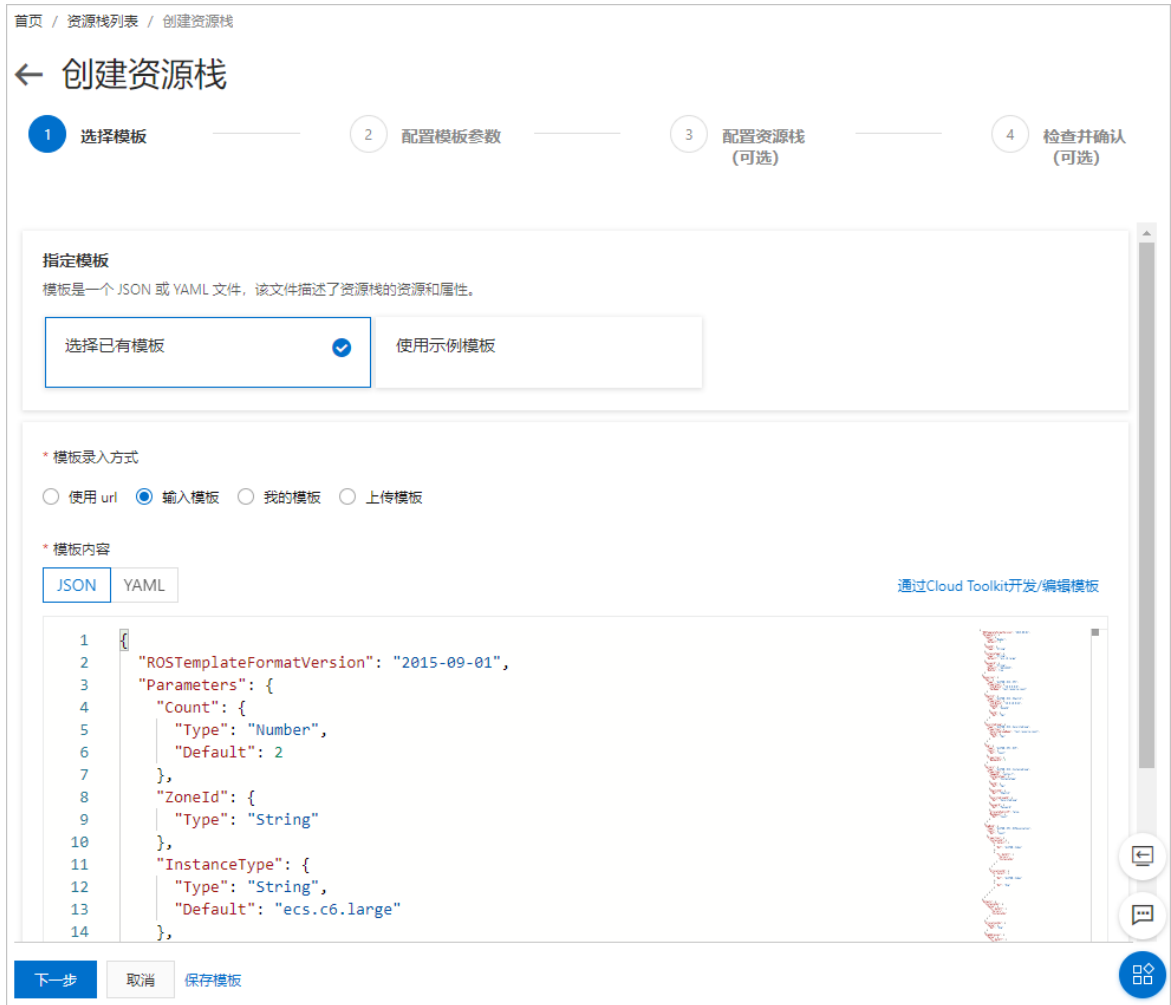
ALIYUN::VPC::EIP类型用于申请弹性公网IP（EIP）。如果需要申请多个弹性公网IP，则需要在模板中写多个ALIYUN::VPC::EIP资源，这样模板会变得冗长。此时您可以使用Count功能批量创建资源。Count功能详情，请参见[Count](#)。

本文将使用Count功能，批量创建ECS实例和EIP，并为ECS实例绑定EIP。本示例将会创建以下资源：

- 1个VPC（专有网络）
- 1个VSwitch（交换机）
- 1个SecurityGroup（安全组）
- 2个ECS（按量付费的ECS）
- 2个EIP（弹性公网IP）

操作步骤

1. 登录[资源编排控制台](#)。
2. 在左侧导航栏单击资源栈。
3. 在页面左上角的地域下拉列表，选择资源栈的所在地域。
4. 在资源栈列表页面，单击创建资源栈。
5. 在选择模板页面的指定模板区域，单击选择已有模板。
6. 在模板录入方式区域单击输入模板，输入模板内容，单击下一步。



模板代码如下：

```

{
  "ROSTemplateFormatVersion": "2015-09-01",
  "Parameters": {
    "Count": {
      "Type": "Number",
      "Default": 2
    },
    "ZoneId": {
      "Type": "String"
    },
    "InstanceType": {
      "Type": "String",
      "Default": "ecs.c6.large"
    },
    "Password": {
      "Type": "String",
      "Default": "Abc1****",

```

```
"NoEcho": true
}
},
"Resources": {
  "Vpc": {
    "Type": "ALIYUN::ECS::VPC",
    "Properties": {
      "CidrBlock": "10.0.0.0/8",
      "VpcName": "test-resource-count"
    }
  },
  "VSwitch": {
    "Type": "ALIYUN::ECS::VSwitch",
    "Properties": {
      "CidrBlock": "10.0.10.0/24",
      "ZoneId": {
        "Ref": "ZoneId"
      },
      "VpcId": {
        "Ref": "Vpc"
      }
    }
  },
  "SecurityGroup": {
    "Type": "ALIYUN::ECS::SecurityGroup",
    "Properties": {
      "SecurityGroupName": "test-resource-count",
      "VpcId": {
        "Ref": "Vpc"
      }
    }
  },
  "Eip": {
    "Type": "ALIYUN::VPC::EIP",
    "Count": {
      "Ref": "Count"
    },
    "Properties": {
      "Bandwidth": 5
    }
  },
  "Servers": {
    "Type": "ALIYUN::ECS::InstanceGroup"
```

```
    Type : ALIYUN::ECS::InstanceGroup ,
    "Properties": {
      "ImageId": "centos_7",
      "InstanceType": {
        "Ref": "InstanceType"
      },
      "VpcId": {
        "Ref": "Vpc"
      },
      "VSwitchId": {
        "Ref": "VSwitch"
      },
      "SecurityGroupId": {
        "Ref": "SecurityGroup"
      },
      "Password": {
        "Ref": "Password"
      },
      "AllocatePublicIP": false,
      "MaxAmount": {
        "Ref": "Count"
      }
    },
    "EipBind": {
      "Type": "ALIYUN::VPC::EIPAssociation",
      "Count": {
        "Ref": "Count"
      },
      "Properties": {
        "InstanceId": {
          "Fn::Select": [
            {
              "Ref": "ALIYUN::Index"
            },
            {
              "Fn::GetAtt": [
                "Servers",
                "InstanceIds"
              ]
            }
          ]
        }
      }
    }
  },
```

```
"AllocationId": {
  "Fn::Select": [
    {
      "Ref": "ALIYUN::Index"
    },
    {
      "Ref": "Eip"
    }
  ]
},
"Outputs": {
  "InstanceIds": {
    "Value": {
      "Fn::GetAtt": [
        "Servers",
        "InstanceIds"
      ]
    }
  },
  "AllocationIds": {
    "Value": {
      "Ref": "Eip"
    }
  },
  "EipAddresses": {
    "Value": {
      "Fn::GetAtt": [
        "Eip",
        "EipAddress"
      ]
    }
  }
}
```

模板说明如下：

- 创建2个EIP。

Count取值为2。使用Count功能，创建了2个EIP资源。通过ROS的预处理，会生成名为Eip[0]和Eip[1]的两个资源。

```
{
  "Eip": {
    "Type": "ALIYUN::VPC::EIP",
    "Count": {
      "Ref": "Count"
    },
  },
  "Properties": {
    "Bandwidth": 5
  }
}
```

- 创建2个ECS实例。

Count取值为2。将ALIYUN::ECS::InstanceGroup的MaxAmount指定为Count，取值为2，生成两个ECS资源。


```
{
  "Servers": {
    "Type": "ALIYUN::ECS::InstanceGroup",
    "Properties": {
      "ImageId": "centos_7",
      "InstanceType": {
        "Ref": "InstanceType"
      },
      "VpcId": {
        "Ref": "Vpc"
      },
      "VSwitchId": {
        "Ref": "VSwitch"
      },
      "SecurityGroupId": {
        "Ref": "SecurityGroup"
      },
      "Password": {
        "Ref": "Password"
      },
      "AllocatePublicIP": false,
      "MaxAmount": {
        "Ref": "Count"
      }
    }
  }
}
```

- 创建2个EipBind资源，通过ALIYUN::Index伪参数逐一绑定ECS实例和EIP。

Count取值为2。使用Count功能，创建了2个EipBind资源。通过ROS的预处理，会生成名为EipBind[0]和EipBind[1]的两个资源。ALIYUN::Index在Count中使用，在预处理时会被替换为相应的数值。本示例中将逐一绑定ECS实例与EIP，即第一个实例绑定Eip[0]，第二个实例绑定Eip[1]。

```
{
  "EipBind": {
    "Type": "ALIYUN::VPC::EIPAssociation",
    "Count": {
      "Ref": "Count"
    },
    "Properties": {
      "InstanceId": {
        "Fn::Select": [
          {
            "Ref": "ALIYUN::Index"
          },
          {
            "Fn::GetAtt": [
              "Servers",
              "InstanceIds"
            ]
          }
        ]
      },
      "AllocationId": {
        "Fn::Select": [
          {
            "Ref": "ALIYUN::Index"
          },
          {
            "Ref": "Eip"
          }
        ]
      }
    }
  }
}
```

7. 在配置模板参数页面，根据控制台提示，配置资源栈名称和参数录入，单击下一步。
8. 在配置资源栈页面，根据控制台提示，配置资源栈策略、失败时回滚、超时设置、删除保护、RAM角色和标签，单击下一步。资源的创建或更新未在超时设置的时间内完成，系统自动判断该操作失败，再根据失败时回滚设置，判断是否回滚到创建或更新资源之前的状态。
9. 在检查并确认页面，单击创建。

执行结果

资源栈创建后，您可以在资源栈详情页面单击资源页签，查看经过ROS预处理后的资源信息。

您也可以在资源栈详情页面单击模板页签，查看经过ROS预处理后的模板信息。

```
{
  "ROSTemplateFormatVersion": "2015-09-01",
  "Parameters": {
    "Count": {
      "Default": 2,
      "Type": "Number"
    },
    "Password": {
      "Default": "Abc12345",
      "NoEcho": true,
      "Type": "String"
    },
    "InstanceType": {
      "Default": "ecs.c6.large",
      "Type": "String"
    },
    "ZoneId": {
      "Type": "String"
    }
  },
  "Resources": {
    "Vpc": {
      "Type": "ALIYUN::ECS::VPC",
      "Properties": {
        "VpcName": "test-resource-count",
        "CidrBlock": "10.0.0.0/8"
      }
    },
    "VSwitch": {
      "Type": "ALIYUN::ECS::VSwitch",
      "Properties": {
        "VpcId": {
          "Ref": "Vpc"
        },
        "CidrBlock": "10.0.10.0/24",
        "ZoneId": {
          "Ref": "ZoneId"
        }
      }
    },
    "SecurityGroup": {
```

```
"Type": "ALIYUN::ECS::SecurityGroup",
"Properties": {
  "VpcId": {
    "Ref": "Vpc"
  },
  "SecurityGroupName": "test-resource-count"
},
"Eip[0]": {
  "Type": "ALIYUN::VPC::EIP",
  "Properties": {
    "Bandwidth": 5
  }
},
"Eip[1]": {
  "Type": "ALIYUN::VPC::EIP",
  "Properties": {
    "Bandwidth": 5
  }
},
"Servers": {
  "Type": "ALIYUN::ECS::InstanceGroup",
  "Properties": {
    "VpcId": {
      "Ref": "Vpc"
    },
    "MinAmount": {
      "Ref": "Count"
    },
    "SecurityGroupId": {
      "Ref": "SecurityGroup"
    },
    "ImageId": "centos_7",
    "AllocatePublicIP": false,
    "VSwitchId": {
      "Ref": "VSwitch"
    },
    "Password": {
      "Ref": "Password"
    },
    "InstanceType": {
      "Ref": "InstanceType"
    }
  }
}
```

```
},
"MaxAmount": {
  "Ref": "Count"
}
},
"EipBind[0]": {
  "Type": "ALIYUN::VPC::EIPAssociation",
  "Properties": {
    "InstanceId": {
      "Fn::Select": [
        0,
        {
          "Fn::GetAtt": [
            "Servers",
            "InstanceIds"
          ]
        }
      ]
    }
  },
  "AllocationId": {
    "Ref": "Eip[0]"
  }
},
"EipBind[1]": {
  "Type": "ALIYUN::VPC::EIPAssociation",
  "Properties": {
    "InstanceId": {
      "Fn::Select": [
        1,
        {
          "Fn::GetAtt": [
            "Servers",
            "InstanceIds"
          ]
        }
      ]
    }
  },
  "AllocationId": {
    "Ref": "Eip[1]"
  }
},
```

```
}
}
},
"Outputs": {
  "AllocationIds": {
    "Value": [
      {
        "Ref": "Eip[0]"
      },
      {
        "Ref": "Eip[1]"
      }
    ]
  },
  "InstanceIds": {
    "Value": {
      "Fn::GetAtt": [
        "Servers",
        "InstanceIds"
      ]
    }
  },
  "EipAddresses": {
    "Value": [
      {
        "Fn::GetAtt": [
          "Eip[0]",
          "EipAddress"
        ]
      },
      {
        "Fn::GetAtt": [
          "Eip[1]",
          "EipAddress"
        ]
      }
    ]
  }
}
```

5.使用Count功能进行大规模部署

资源编排服务ROS (Resource Orchestration Service) 支持Count功能，用于批量创建资源，满足复杂场景或者动态化场景的需求。本文为您介绍如何使用Count功能进行大规模部署。

扩容场景

本场景将使用Count功能，批量创建ECS（按量付费的ECS）。以创建3000个ECS为例，共计需要36183个资源，具体如下：

资源	数量	说明
ALIYUN::ECS::NetworkInterface	9000个	1个ECS要绑定3个弹性网卡（ENI）。
ALIYUN::ECS::NetworkInterfaceAttachment	9000个	该资源用于绑定弹性网卡（ENI）到专有网络（VPC）类型实例上。
ALIYUN::VPC::EIP	9000个	1个弹性网卡（ENI）要绑定1个弹性公网IP（EIP）。
ALIYUN::VPC::EIPAssociation	9000个	该资源用于绑定弹性公网IP（EIP）。
ALIYUN::ECS::InstanceGroup	3个	1个ALIYUN::ECS::InstanceGroup最多包含1000个ECS资源。
ALIYUN::VPC::CommonBandwidthPackage	90个	100个弹性公网IP（EIP）需要1个共享带宽包。
ALIYUN::VPC::CommonBandwidthPackage	90个	该资源用于添加弹性公网IP（EIP）到共享带宽中。

因为每个资源栈最多容纳300个资源，无法满足需求，故需结合嵌套资源栈功能和Count功能设计解决方案。

方案一：基础方案

首先，假定ECS数量为M，将资源划分到子资源栈中，并使用父资源栈进行整合。


- 子资源栈A：用于创建ECS，绑定ENI和EIP。
假定ECS数量为N， $0 <= N <= 20$ ，那么子资源栈最多包含 $1+240=241$ 个资源。
 - 20个ECS：1个ALIYUN::ECS::InstanceGroup资源。
 - ENI+EIP： $20*12=240$ 。
- 子资源栈B：用于创建共享带宽包，关联共享带宽包和EIP。
子资源栈中共享带宽包数量为3，那么子资源栈最多包含 $(1+1)*3=6$ 个资源。
 - 因为1个ECS绑定了3个EIP，对应于三条网络线路，所以放3个共享带宽包。
 - 因为100个EIP要加入1个共享带宽包中，所以1个子资源栈B最多可以容纳300个EIP。1个子资源栈A产生最多60个EIP，所以一个子资源栈B对应于5个子资源栈A。
- 父资源栈：主要用于整合子资源栈。
资源总数为 $150+30=180$ ，能够放入一个资源栈中。
 - 子资源栈A：数量为 $(M+20-1)//20$ ，最大值为150。因为每5个子资源栈A的输出会作为1个子资源B的输入。子资源栈A的个数需要对齐到5，这样会包含一些N取值为0的子资源栈A。所以最终数量为 $((M+20-1)//20+4)//5*5$ ，最大值仍为150。

- 子资源栈B: 数量为子资源栈A的1/5, 即 $((M+20-1)//20+4)//5$, 最大值为30。

其次, 进行模板设计。

- 子资源栈A

- Eni[0]、Eni[1]、Eni[2]绑定到Servers[0], Eni[3]、Eni[4]、Eni[5]绑定到Servers[1], 依此类推, Eni[3*i]、Eni[3*i+1]、Eni[3*i+2]绑定到Servers[i], $0 \leq i \leq N-1$ 。
- Eip-ChinaTelecom[i]绑定到Eni[3*i]、Eip-ChinaUnicom[i]绑定到Eni[3*i+1]、Eip-ChinaMobile[i]绑定到Eni[3*i+2], $0 \leq i \leq N-1$ 。这样1个ECS就绑定了3个不同ISP的EIP。

 说明 此处ISP使用ChinaTelecom、ChinaUnicom和ChinaMobile仅作为示例。

```
{
  "ROSTemplateFormatVersion": "2015-09-01",
  "Parameters": {
    "NumberOfEcs": {
      "Type": "Number",
      "MinValue": 0,
      "MaxValue": 20
    }
  },
  "Conditions": {
    "NonEmpty": {
      "Fn::Not": {
        "Fn::Equals": [
          0,
          {
            "Ref": "NumberOfEcs"
          }
        ]
      }
    }
  },
  "Resources": {
    "Servers": {
      "Type": "ALIYUN::ECS::InstanceGroup",
      "Condition": "NonEmpty",
      "Properties": {
        "MinAmount": {
          "Ref": "NumberOfEcs"
        },
        "MaxAmount": {
          "Ref": "NumberOfEcs"
        }
      }
    },
    ,
```



```
}
},
"Eni": {
  "Type": "ALIYUN::ECS::NetworkInterface",
  "Count": {
    "Fn::Calculate": [
      "{0}*3",
      0,
      [
        {
          "Ref": "NumberOfEcs"
        }
      ]
    ]
  },
  "Properties": null
},
"EniBinder": {
  "Type": "ALIYUN::ECS::NetworkInterfaceAttachment",
  "Count": {
    "Fn::Calculate": [
      "{0}*3",
      0,
      [
        {
          "Ref": "NumberOfEcs"
        }
      ]
    ]
  },
  "Properties": {
    "InstanceId": {
      "Fn::Select": [
        {
          "Fn::Calculate": [
            "{0}/3",
            0,
            [
              "Ref": "ALIYUN::Index"
            ]
          ]
        }
      ]
    }
  }
},
{
```

```
,
"Fn::GetAtt": [
  "Servers",
  "InstanceIds"
]
}
]
},
"NetworkInterfaceId": {
  "Fn::Select": [
    {
      "Ref": "ALIYUN::Index"
    },
    {
      "Ref": "Eni"
    }
  ]
},
"Eip-ChinaTelecom": {
  "Type": "ALIYUN::VPC::EIP",
  "Count": {
    "Ref": "NumberOfEcs"
  },
  "Properties": {
    "Isp": "ChinaTelecom"
  }
},
"Eip-ChinaTelecom-Binder": {
  "Type": "ALIYUN::VPC::EIPAssociation",
  "Count": {
    "Ref": "NumberOfEcs"
  },
  "Properties": {
    "InstanceId": {
      "Fn::Select": [
        {
          "Fn::Calculate": [
            "{0}*3",
            0,
            [
              {
```

```
"Ref": "ALIYUN::Index"
}
]
},
{
  "Ref": "Eni"
}
],
"AllocationId": {
  "Fn::Select": [
    {
      "Ref": "ALIYUN::Index"
    },
    {
      "Ref": "Eip-ChinaTelecom"
    }
  ]
},
"Eip-ChinaUnicom": {
  "Type": "ALIYUN::VPC::EIP",
  "Count": {
    "Ref": "NumberOfEcs"
  },
  "Properties": {
    "Isp": "ChinaUnicom"
  }
},
"Eip-ChinaUnicom-Binder": {
  "Type": "ALIYUN::VPC::EIPAssociation",
  "Count": {
    "Ref": "NumberOfEcs"
  },
  "Properties": {
    "InstanceId": {
      "Fn::Select": [
        {
          "Fn::Calculate": [
            "{0}*3+1",
```

```
0,
[
{
  "Ref": "ALIYUN::Index"
}
]
},
{
  "Ref": "Eni"
}
],
"AllocationId": {
  "Fn::Select": [
    {
      "Ref": "ALIYUN::Index"
    },
    {
      "Ref": "Eip-ChinaUnicom"
    }
  ]
},
"Eip-ChinaMobile": {
  "Type": "ALIYUN::VPC::EIP",
  "Count": {
    "Ref": "NumberOfEcs"
  },
  "Properties": {
    "Isp": "ChinaMobile"
  }
},
"Eip-ChinaMobile-Binder": {
  "Type": "ALIYUN::VPC::EIPAssociation",
  "Count": {
    "Ref": "NumberOfEcs"
  },
  "Properties": {
    "InstanceId": {
      "Fn::Select": [

```

```
1
"Fn::Calculate": [
  "{0}*3+2",
  0,
  [
    {
      "Ref": "ALIYUN::Index"
    }
  ]
],
{
  "Ref": "Eni"
}
],
"AllocationId": {
  "Fn::Select": [
    {
      "Ref": "ALIYUN::Index"
    },
    {
      "Ref": "Eip-ChinaMobile"
    }
  ]
}
},
"Outputs": {
  "Eips-ChinaTelecom": {
    "Value": {
      "Ref": "Eip-ChinaTelecom"
    }
  },
  "Eips-ChinaUnicom": {
    "Value": {
      "Ref": "Eip-ChinaUnicom"
    }
  },
  "Eips-ChinaMobile": {
    "Value": {
      "Ref": "Eip-ChinaMobile"
    }
  }
}
```

```
}  
}  
}  
}
```

- 子资源栈B

这是一个普通模板，ChinaTelecom的EIP加入ChinaTelecom的共享带宽包，ChinaUnicom的EIP加入ChinaUnicom的共享带宽包，ChinaMobile的EIP加入ChinaMobile的共享带宽包。

 说明 此处使用ChinaTelecom、ChinaUnicom和ChinaMobile仅作为示例。

```
{  
  "ROSTemplateFormatVersion": "2015-09-01",  
  "Parameters": {  
    "Eips-ChinaTelecom": {  
      "Type": "Json"  
    },  
    "Eips-ChinaUnicom": {  
      "Type": "Json"  
    },  
    "Eips-ChinaMobile": {  
      "Type": "Json"  
    }  
  },  
  "Resources": {  
    "CommonBandwidthPackage-ChinaTelecom": {  
      "Type": "ALIYUN::VPC::CommonBandwidthPackage",  
      "Properties": null  
    },  
    "CommonBandwidthPackage-ChinaTelecom-IpBinder": {  
      "Type": "ALIYUN::VPC::CommonBandwidthPackageIp",  
      "Properties": {  
        "Eips": {  
          "Ref": "Eips-ChinaTelecom"  
        }  
      },  
      "BandwidthPackageId": {  
        "Ref": "CommonBandwidthPackage-ChinaTelecom"  
      }  
    },  
    "CommonBandwidthPackage-ChinaUnicom": {  
      "Type": "ALIYUN::VPC::CommonBandwidthPackage",  
      "Properties": null  
    },  
    "CommonBandwidthPackage-ChinaUnicom-IpBinder": {  
      "Type": "ALIYUN::VPC::CommonBandwidthPackageIp",  
      "Properties": {  
        "Eips": {  
          "Ref": "Eips-ChinaUnicom"  
        }  
      },  
      "BandwidthPackageId": {  
        "Ref": "CommonBandwidthPackage-ChinaUnicom"  
      }  
    },  
    "CommonBandwidthPackage-ChinaMobile": {  
      "Type": "ALIYUN::VPC::CommonBandwidthPackage",  
      "Properties": null  
    },  
    "CommonBandwidthPackage-ChinaMobile-IpBinder": {  
      "Type": "ALIYUN::VPC::CommonBandwidthPackageIp",  
      "Properties": {  
        "Eips": {  
          "Ref": "Eips-ChinaMobile"  
        }  
      },  
      "BandwidthPackageId": {  
        "Ref": "CommonBandwidthPackage-ChinaMobile"  
      }  
    }  
  }  
}
```

```

"Properties": null
},
"CommonBandwidthPackage-ChinaUnicom-IpBinder": {
  "Type": "ALIYUN::VPC::CommonBandwidthPackageIp",
  "Properties": {
    "Eips": {
      "Ref": "Eips-ChinaUnicom"
    },
    "BandwidthPackageId": {
      "Ref": "CommonBandwidthPackage-ChinaUnicom"
    }
  }
},
"CommonBandwidthPackage-ChinaMobile": {
  "Type": "ALIYUN::VPC::CommonBandwidthPackage",
  "Properties": null
},
"CommonBandwidthPackage-ChinaMobile-IpBinder": {
  "Type": "ALIYUN::VPC::CommonBandwidthPackageIp",
  "Properties": {
    "Eips": {
      "Ref": "Eips-ChinaMobile"
    },
    "BandwidthPackageId": {
      "Ref": "CommonBandwidthPackage-ChinaMobile"
    }
  }
}
}
}
}
}
}
}
}
}
}
}

```

- 父资源栈

嵌套资源栈要求模板通过URL提供，将两个子资源栈的模板加入OSS Bucket。假定子资源栈A的模板地址为oss://templates/resourses-a，子资源栈B的模板地址为oss://templates/resourses-b。

- 资源A为子资源栈A。

NumberOfEcs的取值表达式为 $(1-({1}+1)/((\{0\}+19)//20+1)+999)//1000*(((\{0\}-20*\{1\})//20+\{0\})//(\{0\}+1)*(20-\{\{0\}-\{0\}\}/20*20))+(\{0\}-\{0\}\}/20*20)$ ，计算方法如下：

- 限制条件：Fn::Calculate只能使用加、减、乘、浮点除 (/)、整数除 (//) 五种运算。

- 取值规则：从第0组开始，每次从M中取20个；如果不足20个，则取剩下的所有；如果已取完，则为0。通过举例了解子资源栈A中NumberOfEcs参数应该的取值：
 - 当M=1时，A.Count为5，NumberOfEcs依次为1、0、0、0、0。
 - 当M=20时，A.Count为5，NumberOfEcs依次为20、0、0、0、0。
 - 当M=99时，A.Count为5，NumberOfEcs依次为20、20、20、20、19。
 - 当M=100时，A.Count为5，NumberOfEcs依次为20、20、20、20、20。
 - 当M=101时，A.Count为10，NumberOfEcs依次为20、20、20、20、20、1、0、0、0、0。
- 数学技巧：
 - 技巧1：如何将序列 0, 1, 2, ... 转换成 0, 1, 1, ...?

使用 $F(x, t) = (x + t) // (t + 1)$ ，其中 $t \geq \text{Max}(x)$ 。当 $x=0$ 时， $F(0)=0$ ；当 $x \geq 1$ ， $F(x) \geq 1$ ，而 $F(x) = (x + t) // (t + 1) \leq (x + \text{Max}(x)) // (\text{Max}(x) + 1) \leq (\text{Max}(x) + \text{Max}(x)) // (\text{Max}(x) + 1) < 2$ ，所以 $F(x)=1$ 。
 - 技巧2：如何将0、1的序列转换成P、Q的序列？

令 $G(x, P, Q) = x*(Q-P)+P$ ， x 取值为0或1。 $f(0)=P$ ， $f(1)=Q$ 。
 - 技巧3：如果计算M和N的余数？

$M\%N = M - M//N*N$ 。
- 表达式：

定义 $N=20$ ， $U=(M+N-1)//N$ ， $V=(U+4)//5*5$ ，则U表示对齐到5前子资源栈A的数量，V表示对齐到5后子资源栈A的数量，也就是实际数量。

当M=101时，观察 $f(i)=(M-N*i)//N$ (i为编号，对应于ALIYUN::Index伪参数， $0 \leq i < V$ ， $U=6$ ， $V=10$)，其值依次为 $f(0)=5$ 、 $f(1)=4$ 、 $f(2)=3$ 、 $f(3)=2$ 、 $f(4)=1$ 、 $f(5)=0$ 、 $f(6)=-1$ 、 $f(7)=-2$ 、 $f(8)=-3$ 、 $f(9)=-4$ 。

 - $0 \leq i < U$

序列为5、4、3、2、1、0，利用技巧1，可以将其转换为1、1、1、1、1、0。

令 $t = M \geq \text{Max}(f(i))$ ，则 $g(i) = F(f(i), M) = ((M-N*i)//N + M) // (M+1)$ ，这是序列函数。利用技巧2，可以将序列转换为20、20、20、20、20、1，也就是 $0 \leq i < U$ 时NumberOfEcs的取值。

令 $P=M\%N$ ， $Q=N$ ， $h(i) = G(g(i), M\%N, N) = (((M-N*i)//N + M) // (M+1)) * (N - (M-M//N*N)) + (M-M//N*N)$ 。
 - $i \geq U$

$0 \leq i < U$ 时，值为1； $i \geq U$ 时，值为0。观察 $k(i)=(i+1)//(U+1)$ ，当 $0 \leq i < U$ 时，值为0； $i \geq U$ 时，值 ≥ 1 。

使用技巧1，即可转换成0和1的序列。我们估算下k(i)的最大值： $\text{Max}(k(i))=V//(U+1)=(U+4)//5*5//(U+1) \leq 4$ ，t取4即可，这里我们取999。

新序列 $p(i) = 1 - F(k(i), 999) = 1 - ((i+1)//((M+N-1)//N+1) + 999) // 1000$ 。当 $0 \leq i < U$ 时，值为1； $i \geq U$ 时，值0。

最终的序列 $q(i) = p(i) * h(i) = (1 - ((i+1)//((M+N-1)//N+1) + 999) // 1000) * (((M-N*i)//N + M) // (M+1)) * (N - (M-M//N*N)) + (M-M//N*N)$ 。令 $M=\{0\}$ ， $i=\{1\}$ ， $N=20$ 代入q(i)得到表达式： $(1 - ((\{1\}+1)//((\{0\}+19)//20+1)+999)//1000)*(((\{0\}-20*\{1\})//20+\{0\})//(\{0\}+1)*(20-(\{0\}-\{0\})//20*20))+(\{0\}-\{0\})//20*20)$ 。
- 资源B为子资源栈B。

子资源栈B[i]会将子资源栈A[5*i]、A[5*i+1]、A[5*i+2]、A[5*i+3]、A[5*i+4]的输出拼接在一起作为输入。

```
{
```



```
"ROSTemplateFormatVersion": "2015-09-01",
"Parameters": {
  "NumberOfEcs": {
    "Type": "Number",
    "MinValue": 0
  }
},
"Resources": {
  "A": {
    "Type": "ALIYUN::ROS::Stack",
    "Count": {
      "Fn::Calculate": [
        "({{0}}+19)//20+4)//5*5",
        0,
        [
          {
            "Ref": "NumberOfEcs"
          }
        ]
      ],
      "Properties": {
        "TemplateURL": "oss://templates/resources-a",
        "Parameters": {
          "NumberOfEcs": {
            "Fn::Calculate": [
              "(1-(((1}+1)//((0}+ 19)//20+1)+999)//1000)*(((0}-20*{1})//20+{0})//({0}+ 1)*(20-({0}-{0})//20*20)+({0}-{0})//20*20
            )",
              0,
              [
                {
                  "Ref": "NumberOfEcs"
                }
              ],
              [
                {
                  "Ref": "ALIYUN::Index"
                }
              ]
            ]
          }
        }
      ]
    }
  }
}
```

```
"B": {
  "Type": "ALIYUN::ROS::Stack",
  "Count": {
    "Fn::Calculate": [
      "(((0)+19)//20+4)//5",
      0,
      [
        {
          "Ref": "NumberOfEcs"
        }
      ]
    ],
    "Properties": {
      "TemplateURL": "oss://templates/resources-b",
      "Parameters": {
        "Eips-ChinaTelecom": {
          "Fn::ListMerge": [
            {
              "Fn::Select": [
                {
                  "Fn::Calculate": [
                    "5*{0}",
                    0,
                    [
                      {
                        "Ref": "ALIYUN::Index"
                      }
                    ]
                  ],
                  "Fn::GetAtt": [
                    "A",
                    "Eips-ChinaTelecom"
                  ]
                }
              ]
            }
          ]
        }
      ]
    }
  ],
  "Fn::Select": [
    {
      "Fn::Calculate": [
```

```
Fn::Calculate : [
  "5*{0}+1",
  0,
  [
    {
      "Ref": "ALIYUN::Index"
    }
  ]
],
{
  "Fn::GetAtt": [
    "A",
    "Eips-ChinaTelecom"
  ]
},
{
  "Fn::Select": [
    {
      "Fn::Calculate": [
        "5*{0}+2",
        0,
        [
          {
            "Ref": "ALIYUN::Index"
          }
        ]
      ],
      "Fn::GetAtt": [
        "A",
        "Eips-ChinaTelecom"
      ]
    }
  ]
},
{
  "Fn::Select": [
    {
      "Fn::Calculate": [
```

```
"5*{0}+3",
0,
[
{
"Ref": "ALIYUN::Index"
}
]
],
{
"Fn::GetAtt": [
"A",
"Eips-ChinaTelecom"
]
}
],
{
"Fn::Select": [
{
"Fn::Calculate": [
"5*{0}+4",
0,
[
{
"Ref": "ALIYUN::Index"
}
]
]
},
{
"Fn::GetAtt": [
"A",
"Eips-ChinaTelecom"
]
}
]
}
]
}
}
```

```
}  
}  
}
```

5个资源栈A对应一个资源栈B，5是固定的，无法实现动态。因为Fn::Select已经支持slice选取功能，通过Start:Stop:Step的方式从列表中选取多个元素，所以可以实现动态化。

结合宙斯场景，使用Fn::Min简化表达式。新模板如下：

```
{  
  "ROSTemplateFormatVersion": "2015-09-01",  
  "Parameters": {  
    "NumberOfEcs": {  
      "Type": "Number",  
      "MinValue": 0  
    }  
  },  
  "Resources": {  
    "A": {  
      "Type": "ALIYUN::ROS::Stack",  
      "Count": {  
        "Fn::Calculate": [  
          "({0}+19)//20",  
          0,  
          [  
            {  
              "Ref": "NumberOfEcs"  
            }  
          ]  
        ]  
      },  
      "Properties": {  
        "TemplateURL": "oss://templates/resources-a",  
        "Parameters": {  
          "NumberOfEcs": {  
            "Fn::Min": [  
              20,  
              {  
                "Fn::Calculate": [  
                  "{0}-{1}*20",  
                  0,  
                  [  
                    {  
                      "Ref": "NumberOfEcs"  
                    }  
                  ]  
                }  
              ]  
            }  
          }  
        }  
      }  
    }  
  }  
}
```

```
},
{
  "Ref": "ALIYUN::Index"
}
]
]
}
]
}
}
},
"B": {
  "Type": "ALIYUN::ROS::Stack",
  "Count": {
    "Fn::Calculate": [
      "({0}+19)//20+4)//5",
      0,
      [
        {
          "Ref": "NumberOfEcs"
        }
      ]
    ],
    "Properties": {
      "TemplateURL": "oss://templates/resources-b",
      "Parameters": {
        "Eips-ChinaTelecom": {
          "Fn::ListMerge": [
            {
              "Fn::Select": [
                {
                  "Fn::Replace": [
                    {
                      "Start": {
                        "Fn::Calculate": [
                          "5*{0}",
                          0,
                          [
                            {
                              "Ref": "ALIYUN::Index"
```

```
}
]
]
},
"Stop": {
  "Fn::Calculate": [
    "5*({0}+1)",
    0,
    [
      {
        "Ref": "ALIYUN::Index"
      }
    ]
  ]
},
"Start:Stop"
]
},
{
  "Fn::GetAtt": [
    "A",
    "Eips-ChinaTelecom"
  ]
}
]
}
}
}
}
}
}
}
}
}
```

方案二：优化方案

将资源进行重新划分：

- 子资源栈A：将ALIYUN::VPC::CommonBandwidthPackageIp加入子资源栈A中。
- 子资源栈B：将所有ALIYUN::VPC::CommonBandwidthPackage加入一个子资源栈B中。子资源栈B本身的数量不再需要使用Count，但里面的资源需要使用Count。子资源栈B输出创建的共享带宽包ID列表，并传递给子资源栈A。

将所有参数动态化：

- 将子资源A的ECS数量最大值20，变成参数MaxNumberOfEcsPerStack。
- 将100个EIP存放的共享带宽包数量1，变成参数MaxNumberOfEipPerBandwidthPackage。
- 在父资源中创建子资源A时，传递EipIndexOffset=ALIYUN::Index * MaxNumberOfEcsPerStack。
- 在子资源栈A中通过计算构建动态数量的ALIYUN::VPC::CommonBandwidthPackageIp，将EIP分组后绑定到不同的共享带宽包。

优化后模板如下：

- 子资源栈A

令

$A = \text{EipIndexOffset}$, $B = \text{NumberOfEcs}$, $C = \text{MaxNumberOfEipPerBandwidthPackage}$, $i = \text{ALIYUN::Index}$:
 $A // C$ 为Stack内第一个EIP要绑定的共享带宽包的编号。 $(A+B-1) // C$ 为Stack内最后一个EIP要绑定的共享带宽包的编号。所以 $\text{CommonBandwidthPackage-ChinaTelecom-IpBinder.Count} = ((A+B-1) // C - A // C) + 1$ 。

所以 $\text{CommonBandwidthPackage-ChinaTelecom-IpBinder.BandwidthPackageId} = \text{CommonBandwidthPackage-ChinaTelecom-List}[A // C + i]$ 。组内的Eip-ChinaTelecom该如何分散到CommonBandwidthPackage-ChinaTelecom-IpBinder，我们从全局的角度看：

全局编号在 $[(A // C + i) * C, (A // C + i + 1) * C)$ 之间的CommonBandwidthPackage-ChinaTelecom-IpBinder $[(A // C + i)]$ 中。将全局编号转换成局部编号，只需要减去偏移，即 $[(A // C + i) * C - A, (A // C + i + 1) * C - A)$ 。但这个范围必须限制到 $[0, B]$ 中，所以结果为 $[\text{Max}((A // C + i) * C - A, 0), \text{Min}((A // C + i + 1) * C - A, B)]$ 。

```
{
  "ROSTemplateFormatVersion": "2015-09-01",
  "Parameters": {
    "NumberOfEcs": {
      "Type": "Number",
      "MinValue": 1
    },
    "MaxNumberOfEipPerBandwidthPackage": {
      "Type": "Number",
      "MinValue": 1
    },
    "EipIndexOffset": {
      "Type": "Number"
    },
    "CommonBandwidthPackage-ChinaTelecom-List": {
      "Type": "Json"
    },
    "CommonBandwidthPackage-ChinaUnicom-List": {
      "Type": "Json"
    },
    "CommonBandwidthPackage-ChinaMobile-List": {
      "Type": "Json"
    }
  }
}
```



```
"Conditions": {
  "NonEmpty": {
    "Fn::Not": {
      "Fn::Equals": [
        0,
        {
          "Ref": "NumberOfEcs"
        }
      ]
    }
  },
  "Resources": {
    "CommonBandwidthPackage-ChinaTelecom-IpBinder": {
      "Type": "ALIYUN::VPC::CommonBandwidthPackageIp",
      "Count": {
        "Fn::Calculate": [
          "(((0)+{1}-1)//{2}-{0}://{2})+1",
          0,
          [
            {
              "Ref": "EipIndexOffset"
            },
            {
              "Ref": "NumberOfEcs"
            },
            {
              "Ref": "MaxNumberOfEipPerBandwidthPackage"
            }
          ]
        ]
      },
      "Properties": {
        "Eips": {
          "Fn::Select": [
            {
              "Fn::Replace": [
                {
                  "Start": {
                    "Fn::Max": [
                      0,
                      {
                        "Fn::Calculate": [
```

```
Fn::Calculate : [
  "({0}/{2}+{1})*{2}-{0}",
  0,
  [
  {
    "Ref": "EipIndexOffset"
  },
  {
    "Ref": "ALIYUN::Index"
  },
  {
    "Ref": "MaxNumberOfEipPerBandwidthPackage"
  }
  ]
]
},
"Stop": {
  "Fn::Min": [
  {
    "Ref": "NumberOfEcs"
  },
  {
    "Fn::Calculate": [
      "({0}/{2}+{1}+1)*{2}-{0}",
      0,
      [
      {
        "Ref": "EipIndexOffset"
      },
      {
        "Ref": "ALIYUN::Index"
      },
      {
        "Ref": "MaxNumberOfEipPerBandwidthPackage"
      }
      ]
    ]
  }
  ]
}
```

```
"Start:Stop"
]
},
{
  "Ref": "Eip-ChinaTelecom"
}
]
},
"BandwidthPackageId": {
  "Fn::Select": [
    {
      "Fn::Calculate": [
        "${0}/${2}+{1}",
        0,
        [
          {
            "Ref": "EipIndexOffset"
          },
          {
            "Ref": "ALIYUN::Index"
          },
          {
            "Ref": "MaxNumberOfEipPerBandwidthPackage"
          }
        ]
      ],
      "Ref": "CommonBandwidthPackage-ChinaTelecom-List"
    }
  ]
}
}
}
}
}
```

- 子资源栈B

```
{
  "ROSTemplateFormatVersion": "2015-09-01",
  "Parameters": {
    "Count": {
```

```
"Type": "Number"
}
},
"Resources": {
  "CommonBandwidthPackage-ChinaTelecom": {
    "Type": "ALIYUN::VPC::CommonBandwidthPackage",
    "Count": {
      "Ref": "Count"
    },
    "Properties": null
  },
  "CommonBandwidthPackage-ChinaUnicom": {
    "Type": "ALIYUN::VPC::CommonBandwidthPackage",
    "Count": {
      "Ref": "Count"
    },
    "Properties": null
  },
  "CommonBandwidthPackage-ChinaMobile": {
    "Type": "ALIYUN::VPC::CommonBandwidthPackage",
    "Count": {
      "Ref": "Count"
    },
    "Properties": null
  }
},
"Outputs": {
  "CommonBandwidthPackage-ChinaTelecom-List": {
    "Value": {
      "Ref": "CommonBandwidthPackage-ChinaTelecom"
    }
  },
  "CommonBandwidthPackage-ChinaUnicom-List": {
    "Value": {
      "Ref": "CommonBandwidthPackage-ChinaUnicom"
    }
  },
  "CommonBandwidthPackage-ChinaMobile-List": {
    "Value": {
      "Ref": "CommonBandwidthPackage-ChinaMobile"
    }
  }
}
```

```
}  
}
```

- 父资源栈

```
{  
  "ROSTemplateFormatVersion": "2015-09-01",  
  "Parameters": {  
    "NumberOfEcs": {  
      "Type": "Number",  
      "MinValue": 0  
    },  
    "MaxNumberOfEcsPerStack": {  
      "Type": "Number",  
      "MinValue": 1,  
      "Default": 20  
    },  
    "MaxNumberOfEipPerBandwidthPackage": {  
      "Type": "Number",  
      "MinValue": 1,  
      "Default": 100  
    }  
  },  
  "Resources": {  
    "B": {  
      "Type": "ALIYUN::ROS::Stack",  
      "Properties": {  
        "TemplateURL": "oss://templates/resources-b",  
        "Parameters": {  
          "Count": {  
            "Fn::Calculate": [  
              "({0}+{1}-1)//{1}",  
              0,  
              [  
                {  
                  "Ref": "NumberOfEcs"  
                },  
                {  
                  "Ref": "MaxNumberOfEipPerBandwidthPackage"  
                }  
              ]  
            ]  
          }  
        }  
      }  
    }  
  }  
}
```

```
}
}
},
"A": {
  "Type": "ALIYUN::ROS::Stack",
  "Count": {
    "Fn::Calculate": [
      "({0}+{1}-1)//{1}",
      0,
      [
        {
          "Ref": "NumberOfEcs"
        },
        {
          "Ref": "MaxNumberOfEcsPerStack"
        }
      ]
    ],
    "Properties": {
      "TemplateURL": "oss://templates/resources-a",
      "Parameters": {
        "NumberOfEcs": {
          "Fn::Min": [
            {
              "Ref": "MaxNumberOfEcsPerStack"
            },
            {
              "Fn::Calculate": [
                "{0}-{1}*{2}",
                0,
                [
                  {
                    "Ref": "NumberOfEcs"
                  },
                  {
                    "Ref": "ALIYUN::Index"
                  },
                  {
                    "Ref": "MaxNumberOfEcsPerStack"
                  }
                ]
              ]
            }
          ]
        }
      }
    }
  }
}
```

```
    ]
  }
]
},
"MaxNumberOfEipPerBandwidthPackage": {
  "Ref": "MaxNumberOfEipPerBandwidthPackage"
},
"EipIndexOffset": {
  "Fn::Calculate": [
    "{0}*{1}",
    0,
    [
      {
        "Ref": "MaxNumberOfEcsPerStack"
      },
      {
        "Ref": "ALIYUN::Index"
      }
    ]
  ]
},
"CommonBandwidthPackage-ChinaTelecom-List": {
  "Fn::GetAtt": [
    "B",
    "CommonBandwidthPackage-ChinaTelecom-List"
  ]
},
"CommonBandwidthPackage-ChinaUnicom-List": {
  "Fn::GetAtt": [
    "B",
    "CommonBandwidthPackage-ChinaUnicom-List"
  ]
},
"CommonBandwidthPackage-ChinaMobile-List": {
  "Fn::GetAtt": [
    "B",
    "CommonBandwidthPackage-ChinaMobile-List"
  ]
}
}
```

```
}  
}
```