

# 对象存储服务(OBS)

## 快速入门

文档版本 01

发布日期 2025-12-09



版权所有 © 华为云计算技术有限公司 2025。保留一切权利。

未经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

## 商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

## 注意

您购买的产品、服务或特性等应受华为云计算技术有限公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为云计算技术有限公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

# 华为云计算技术有限公司

地址：贵州省贵安新区黔中大道交兴功路华为云数据中心 邮编：550029

网址：<https://www.huaweicloud.com/>

# 目 录

---

1 通过控制台快速使用 OBS.....	1
2 通过 OBS Browser+图形化工具快速使用 OBS.....	8
3 通过 obsutil 命令行工具快速使用 OBS.....	14
4 通过 OBS SDK 快速使用 OBS.....	16
5 入门实践.....	27

# 1

# 通过控制台快速使用 OBS

控制台是网页形式的操作平台，提供更直观的界面操作，简单易上手。

本章节以上传对象至“test-example-bucket”桶为例，帮助您通过控制台快速使用OBS的基础功能，包括创建桶、上传对象、下载对象、分享对象。

## 准备工作

使用OBS之前，您需要拥有一个华为账号并实名认证，然后为账号充值。

**步骤1** 注册华为账号并实名认证。

如果您已有华为账号，请忽略此步骤。如果您还没有华为账号，请执行以下操作：

1. [注册华为账号并开通华为云](#)。
2. 参考[个人账号如何完成实名认证](#)或[企业账号如何完成实名认证](#)，完成个人或企业账号实名认证。

**步骤2** 为账号充值。

您需要确保账号有足够的余额，才能正常使用OBS等相关资源。请参考[账户充值](#)。

----结束

## 操作视频

### 步骤一：创建桶

桶是OBS中用于存储对象的容器。在上传对象前，您需要先创建桶。

本步骤仅针对示例中的关键参数进行设置和介绍，其他参数保持默认，更多创建桶的详细信息请参见[创建桶](#)。

**步骤1** 登录控制台，进入[创建桶](#)页面。

**步骤2** 设置“基础配置”。

## 基础配置

## 复制桶配置

[选择源桶](#)

该项可选。选择后可复制源桶的以下配置信息：区域 / 数据冗余策略 / 存储类别 / 桶策略 / 服务端加密 / 归档数据直读 / 企业项目 / 标签。

## 区域

[华北-北京四](#)

不同区域的资源之间内网互不相通,请选择靠近您业务的区域,可以降低网络时延,提高访问速度。桶创建成功后不支持变更区域,请谨慎选择。[如何选择区域](#)

参数	示例	说明
区域	华北-北京四	桶所属区域。请选择靠近您业务的区域,以降低网络时延,提高访问速度。 桶创建成功后,不支持变更区域,请谨慎选择。

## 步骤3 设置“桶配置”。其他参数保持默认,也可在桶创建后修改。

## 桶配置

## 桶名称

[test-example-bucket](#) [查看命名规则](#)

桶访问域名: &lt;名称&gt;.obs.cn-north-4.myhuaweicloud.com

## 数据冗余存储策略

[多AZ存储](#) [单AZ存储](#)

数据在同区域的多个AZ中存储,可用性更高。

数据冗余策略启用后**不支持修改**。数据冗余存储多AZ,在单个可用区内做数据冗余存储的基础上,同时在多个可用区间再做数据冗余存储,可靠性更高,采用相对**较高计费标准**。[价格详情](#)

## 存储类型

创建桶时选择的存储类别会作为上传对象的默认存储类别。[了解存储类别差异](#)

## 桶策略

[私有](#) [公共读](#) [公共读写](#) [复制桶策略](#)

桶的拥有者拥有完全控制权限,其他用户在未经授权的情况下均无访问权限。

## 企业项目

[default](#) [查看企业项目](#)

参数	示例	说明
桶名称	<b>test-example-bucket</b>	<p>桶的名称。桶创建成功后，不支持修改名称。</p> <p>桶名称命名规则如下：</p> <ul style="list-style-type: none"><li>全局唯一，不能与已有的任何桶（包含其他账号创建的桶）名称重复。删除桶后，需等待30分钟才能创建同名桶或并行文件系统。</li><li>长度范围为3到63个字符，仅支持小写字母、数字、中划线（-）、英文句号（.）。</li><li>禁止两个英文句号（.）相邻，禁止英文句号（.）和中划线（-）相邻，禁止以英文句号（.）和中划线（-）开头或结尾。</li><li>禁止使用IP地址。</li></ul> <p><b>说明</b> 当用户使用虚拟主机方式通过HTTPS协议访问OBS时，如果桶名称中包含英文句号（.），会导致证书校验失败。因此该场景下，建议桶名称不要使用英文句号（.）。</p>
数据冗余存储策略	<b>多AZ存储</b>	<ul style="list-style-type: none"><li><b>多AZ存储：</b>数据存储至同区域内的多个可用区（AZ），可靠性更高，同时存储成本相对更高。</li><li><b>单AZ存储：</b>数据仅存储在单个可用区（AZ），成本更低。</li></ul> <p>桶创建成功后，不支持更改数据冗余存储策略。 多AZ存储桶不支持为桶设置归档存储类别和深度归档存储类别。</p>
存储类型	<b>标准存储</b>	<p>桶的存储类别。不同的存储类别可以满足客户业务对存储性能、成本的不同诉求。</p> <ul style="list-style-type: none"><li><b>标准存储：</b>适用于有大量热点文件或小文件，且需要频繁访问（平均一个月多次）并快速获取数据的业务场景。</li><li><b>低频访问存储：</b>适用于不频繁访问（平均一年少于12次），但需要快速获取数据的业务场景。</li><li><b>归档存储：</b>适用于很少访问（平均一年一次），且对数据获取速率要求不高的业务场景。</li><li><b>深度归档存储：</b>适用于很少访问，访问频率低于归档存储，且对数据获取速率要求不高的业务场景。</li></ul> <p>更多详情请参见<a href="#">存储类别</a>。</p>

参数	示例	说明
桶策略	私有	<p>桶的读写权限控制。</p> <ul style="list-style-type: none"><li>私有：除桶ACL授权外的其他用户无桶的访问权限。</li><li>公共读：任何用户都可以对桶内对象进行读操作。</li><li>公共读写：任何用户都可以对桶内对象进行读/写/删除操作。</li><li>复制桶策略：复制源桶的桶策略。当且仅当您选择了源桶时，该选项处于可选状态。</li></ul>
企业项目	default	<p>将桶加入到企业项目中统一管理。如无特殊的企业项目划分和管理需求，此处可直接选择默认企业项目“default”。</p> <p>如果您想要了解更多关于如何通过企业项目管理OBS桶，具体请参见<a href="#">创建桶</a>中的“企业项目”参数说明。</p>

**步骤4** 单击页面右下角的“立即创建”，并确认提示信息。

创建完成后，将会出现创建成功弹窗，确认后在桶列表页即可看到已创建的桶。

----结束

## 步骤二：上传对象

桶创建成功后，您可以通过OBS控制台上传一个不超过5GB大小的对象；或单次批量上传数量不超过100个、总大小不超过5GB的对象。

本步骤仅针对示例进行参数设置和介绍，更多上传对象的详细信息请参见[上传对象](#)。

**步骤1** 在桶列表页面，单击已创建好的桶名称，进入“对象”页面。

**步骤2** 单击搜索框上方的“上传对象”。

**步骤3** 将本地文件拖拽至“上传对象”区域框内来添加对象。

也可以通过单击区域框内的“添加文件”，选择本地文件进行添加。

**步骤4** 其他参数保持默认，单击“上传”。

右侧自动弹出任务中心页面，可在任务中心查看对象上传状态。上传成功的对象将在对象列表中展示。



----结束

### 步骤三：下载对象

当对象上传至桶后，您可以将对象下载至浏览器默认保存路径或本地指定路径。

**步骤1** 在对象列表页，找到已上传的对象。

**步骤2** 下载对象。

- **下载单个对象**

单击待下载对象右侧操作列的“下载”，将对象下载至浏览器默认保存路径。

- **批量下载多个对象：**

选中多个对象，然后单击对象列表页搜索框上方的“更多 > 下载”，将多个对象下载至浏览器默认的保存路径下。

----结束

### 步骤四：分享对象

对象上传至桶后，您可以将对象URL分享给其他用户，供其下载或查看对象。

**步骤1** 在对象列表页，单击已上传对象右侧操作列的“分享”。

**步骤2** 在“分享文件”弹窗，设置URL有效期。其他参数保持默认。

- 有效期范围为1分钟~18小时，有效期内任何用户都可以通过URL直接访问该对象。本示例设置为10分钟。
- 该弹窗打开时，链接信息已生效并开始计时，有效期默认为5分钟。修改URL有效期后会重新开始计时，同时链接信息会相应变化，您需要重新复制链接信息。



### 步骤3 分享对象。

- 单击“复制链接”，并将该链接分享给其他用户，用户可以在浏览器中通过此链接直接访问该对象。

#### 说明

通过浏览器访问对象，会强制下载对象，如果您希望被分享人可以在线预览对象，请参见[如何在浏览器中在线预览OBS中的对象？](#)

- 单击“复制路径”，并将该路径分享给拥有对象所在桶权限的用户，用户可以在对应桶的对象列表页中部的对象搜索框中粘贴该路径并按“Enter”键，来搜索查看对象信息或访问对象。

----结束

## 相关信息

当您完成创建桶、上传对象、下载对象等基本操作后，您还可以结合业务需求使用以下OBS的高阶功能。

- 生命周期：**通过为桶配置生命周期规则，可以实现定时转换对象存储类别或定时删除对象。

- **权限控制**: OBS提供IAM权限、桶策略、桶ACL、对象ACL多种权限控制方式，实现对桶和对象更精细化的权限控制。
- **域名管理**: 通过配置自定义域名实现通过自定义域名访问桶和对象；配置CDN加速域名实现加速访问桶和对象。
- **静态网站托管**: 您可以将静态网站文件上传至OBS的桶中，并对这些文件赋予匿名用户可读权限，然后将该桶配置成静态网站托管模式，实现使用桶域名访问该网站。
- **多版本控制**: 为桶开启多版本控制后，可以在桶中保留多个版本的对象，方便检索和还原各个版本，在意外操作或应用程序故障时帮助快速恢复数据。
- **跨区域复制**: 是OBS提供的跨区域数据容灾能力，支持将源桶数据复制到与其不同区域的目标桶。
- **服务端加密**: 通过服务端加密功能，对上传至OBS桶中的数据进行加密保护。
- **对象锁定 (WORM)**: 为桶开启WORM后，可实现桶中的对象在您指定的时间段内不能被删除或篡改。
- **跨域资源共享 (CORS)**: 通过配置CORS规则，可以实现跨域名访问OBS。

# 2 通过 OBS Browser+图形化工具快速使用 OBS

OBS Browser+是一款图形化工具，适用于Windows10, Windows Server2016以及Mac平台，可以非常方便地让用户在本地对OBS的数据进行管理。

本章节以上传一个文件至“test-example-bucket”桶为例，帮助您快速通过OBS Browser+使用OBS的基础功能，包括创建桶、上传对象、下载对象、分享对象。

## 准备工作

使用OBS Browser+图形化工具之前，您需要拥有一个华为账号并实名认证、为账号充值，然后下载并安装OBS Browser+图形化工具，并获取访问密钥。

### 步骤1 注册华为账号并实名认证。

如果您已有华为账号，请忽略此步骤。如果您还没有华为账号，请执行以下操作：

1. [注册华为账号并开通华为云](#)。
2. 参考[个人账号如何完成实名认证](#)或[企业账号如何完成实名认证](#)，完成个人或企业账号实名认证。

### 步骤2 为账号充值。

您需要确保账号有足够的余额，才能正常使用OBS等相关资源。请参考[账户充值](#)。

### 步骤3 [下载](#)并安装OBS Browser+图形化工具。

### 步骤4 获取访问密钥，详情请参见[获取访问密钥（AK和SK）](#)。

----结束

## 步骤一：使用访问密钥 AK/SK 登录 OBS Browser+



参数	示例	说明
账号名	exampleAccount	该账号名仅用于区分您在本地登录OBS Browser+的不同账号，可自定义，方便您记忆和管理。账号名长度范围为3-63个字符，不能包含：\.*?'<> !@#\$%^~字符。
服务提供商	华为对象存储服务 (默认)	选择华为对象存储服务：提供了华为云所支持的所有区域的桶的访问。
Access Key ID & Secret Access Key	-	填写访问密钥AK/SK。您可以参考 <a href="#">获取访问密钥</a> 章节在“ <b>我的凭证</b> ”页面创建永久AK/SK。

### 说明

更多登录方式请参见[登录OBS Browser+](#)。

## 步骤二：创建桶

桶是对象存储中存储对象的容器。您需要先创建一个桶，然后才能在桶中存储数据。

本步骤仅针对示例中的关键参数进行设置和介绍，其他参数保持默认，更多创建桶的详细信息请参见[创建桶](#)。

**步骤1** 在桶列表页左上方单击“创建桶”。

步骤2 在弹窗中填写桶信息，单击“确定”。



参数	示例	说明
区域	华北-北京四	桶所属区域。请选择靠近您业务的区域，以降低网络时延，提高访问速度。 桶创建成功后，不支持变更区域，请谨慎选择。
存储类别	标准存储	桶的存储类别。不同的存储类别可以满足客户业务对存储性能、成本的不同诉求。 <ul style="list-style-type: none"><li>标准存储：适用于有大量热点文件或小文件，且需要频繁访问（平均一个月多次）并快速获取数据的业务场景。</li><li>低频访问存储：适用于不频繁访问（平均一年少于12次），但需要快速获取数据的业务场景。</li><li>归档存储：适用于很少访问（平均一年一次），且对数据获取速率要求不高的业务场景。</li><li>深度归档存储：适用于很少访问，访问频率低于归档存储，且对数据获取速率要求不高的业务场景。</li></ul> <a href="#">更多详情请参见存储类别</a> 。

参数	示例	说明
桶ACL	私有	<p>桶的读写权限控制。</p> <ul style="list-style-type: none"><li>私有：除桶ACL授权外的其他用户无桶的访问权限。</li><li>公共读：任何用户都可以对桶内对象进行读操作。</li><li>公共读写：任何用户都可以对桶内对象进行读/写/删除操作。</li></ul>
桶名称	test-example-bucket	<p>桶的名称。桶创建成功后，不支持修改名称。</p> <p>OBS中桶按照DNS规范进行命名，DNS规范为全球通用规则，其具体命名规则如下：</p> <ul style="list-style-type: none"><li>全局唯一，不能与已有的任何桶（包括其他用户创建的桶）名称重复。删除桶后，需要等待30分钟才能创建同名桶或并行文件系统。</li><li>长度范围为3到63个字符，仅支持小写字母、数字、中划线（-）或英文句号（.）。</li><li>禁止两个英文句号（.）相邻，禁止英文句号（.）和中划线（-）相邻，禁止以英文句号（.）和中划线（-）开头或结尾。</li><li>禁止使用IP地址。</li></ul>

**步骤3** 创建完成后，您可以在桶列表页查看到已创建的桶。

----结束

### 步骤三：上传对象

桶创建成功后，您可以通过OBS Browser+上传一个不超过48.8TB大小的单个对象；或单次批量上传数量不超过500个、总大小不超过48.8TB大小的对象。

**步骤1** 在桶列表页面单击创建好的桶。

**步骤2** 在对象列表页面单击左上角的“上传”。

**步骤3** 选择“添加文件”，选择本地文件进行添加。



#### 步骤4 其他参数保持默认, 单击“确定”。

对象上传后, 您可以在对象列表页查看到已上传的对象。

----结束

### 步骤四：下载对象

当对象上传至桶后, 您可以将对象下载至本地指定路径。

#### 步骤1 在对象列表页, 找到已上传的对象, 单击右侧的 图标。



#### 步骤2 在对话框中选择存放文件或文件夹的本地路径后, 单击“确定”。

对象下载后, 您可以在本地指定路径下查看到下载的对象。

----结束

### 步骤五：分享对象

对象上传至桶后, 您可以将对象URL分享给其他用户, 供其下载或查看对象。

#### 步骤1 单击已上传对象右侧的 图标。

#### 步骤2 在“分享文件”弹窗中, 设置URL有效期, 本示例中有效期设置为1天。

- 有效期内任何用户都可以通过URL直接访问该对象。
- 该弹窗打开时, 分享链接已生效并开始计时, 有效期默认为1天。修改有效期后会重新开始计时, 同时分享链接会相应变化, 您需要重新复制分享链接。

### 步骤3 分享对象。

- 单击“复制链接”，并将该链接分享给其他用户，用户可以在浏览器中通过此链接直接访问该对象。

#### 说明

通过浏览器访问对象，会强制下载对象，如果您希望被分享人可以在线预览对象，请参见[如何在浏览器中在线预览OBS中的对象？](#)

- 单击“复制路径”，并将该路径分享给拥有对象所在桶权限的用户，用户可以在对应桶的对象列表页中部的对象搜索框中粘贴该路径并按“Enter”键，来搜索查看对象信息或访问对象。

----结束

## 相关信息

当您完成创建桶、上传对象、下载对象等基本操作后，您还可以结合业务需求使用以下OBS Browser+的高阶功能。

- 生命周期**：通过为桶配置生命周期规则，可以实现定时转换对象存储类别或定时删除对象。
- 桶ACL权限**：OBS Browser+提供桶ACL访问权限方式，桶的所有者可以通过修改桶ACL，实现对桶更精细化的权限控制。
- 桶策略**：OBS Browser+提供桶策略访问权限方式，桶的所有者可以通过编写桶策略，实现对桶更精细化的权限控制。

# 3 通过 obsutil 命令行工具快速使用 OBS

obsutil 是一款命令行工具，对于熟悉命令行的用户，obsutil 能在执行批量处理、自动化任务的场景中为您带来更优体验。

本章节以 Linux 操作系统为例，以上传一个文件至“bucket-test”桶为例，帮助您快速通过 obsutil 使用 OBS 的基础功能，包括创建桶、上传对象、下载对象、分享对象。

## 准备工作

使用 obsutil 命令行工具之前，您需要拥有一个华为账号并实名认证、为账号充值，然后下载并安装 obsutil 命令行工具，并获取访问密钥。

**步骤1** 注册华为账号并实名认证。

如果您已有华为账号，请忽略此步骤。如果您还没有华为账号，请执行以下操作：

1. [注册华为账号并开通华为云](#)。
2. 参考[个人账号如何完成实名认证](#)或[企业账号如何完成实名认证](#)，完成个人或企业账号实名认证。

**步骤2** 为账号充值。

您需要确保账号有足够的余额，才能正常使用 OBS 等相关资源。请参考[账户充值](#)。

**步骤3** 下载并安装 obsutil 命令行工具，详情请参见[安装 obsutil 命令行工具](#)。

**步骤4** 获取访问密钥，详情请参见[获取访问密钥（AK 和 SK）](#)。

----结束

## 步骤一：创建桶

运行以下命令，在华北-北京四区域创建一个名为 bucket-test（桶名可自定义且不能重复）的新桶：

```
./obsutil mb obs://bucket-test -location=cn-north-4
```

回显结果如下所示：

```
Create bucket [bucket-test] successfully, request id [00000190CA0531DE4015FE3458C7B4C0]
```

## 步骤二：上传对象

运行以下命令，将本地 test.txt 文件上传至 bucket-test 桶中：

**./obsutil cp /temp/test.txt obs://bucket-test/test.txt**

回显结果如下所示：

```
Parallel: 5          Jobs: 5
Threshold: 52428800    PartSize: 5242880
VerifyLength: false    VerifyMd5: false
CheckpointDir: /root/.obsutil_checkpoint

test.txt:[j1] [=====] 100.00% 48.47 KB/s 0s
Upload successfully, 4.44KB, /temp/test.txt --> obs://bucket-test1/test.txt
```

## 步骤三：下载对象

运行以下命令，将bucket-test桶中的test.txt对象下载至本地：

**./obsutil cp obs://bucket-test/test.txt /temp/test1.txt**

回显结果如下所示：

```
Parallel: 5          Jobs: 5
Threshold: 52428800    PartSize: 5242880
VerifyLength: false    VerifyMd5: false
CheckpointDir: /root/.obsutil_checkpoint

test.txt:[=====] 100.00% 775.52 KB/s 0s
Download successfully, 4.44KB, obs://bucket-test1/test.txt --> /temp/test1.txt
```

## 步骤四：分享对象

**步骤1** 运行以下命令，生成对象的下载链接：

**obsutil sign obs://bucket-test/test.txt**

回显结果如下所示：

```
Download url of [obs://bucket-test/test.txt] is:
http://bucket-test.obs.cn-north-4.myhuaweicloud.com/bucket-test/test.txt?
AccessKeyId=xxxx&Expires=1552548758&Signature=xxxx
```

**步骤2** 复制URL并将链接分享给其他用户，用户可以在浏览器中通过此链接直接访问该对象。

### 说明

通过浏览器访问对象，会强制下载对象，如果您希望被分享人可以在线预览对象，请参见[如何在浏览器中在线预览OBS中的对象？](#)

----结束

## 相关信息

当您完成创建桶、上传对象、下载对象等基本操作后，您还可以结合业务需求使用以下obsutil的高阶功能。

- **桶策略**：obsutil提供桶策略权限控制方式，实现对桶和对象更精细化的权限控制。

# 4 通过 OBS SDK 快速使用 OBS

OBS SDK对OBS服务提供的REST API进行封装，以简化用户的开发工作。您直接调用OBS SDK提供的接口函数即可使用OBS管理数据。

本章节以Java、Python、Go三种SDK为例，帮助您快速上手OBS的基础功能，包括创建桶、上传对象、下载对象、列举对象。

## 准备工作

使用Java SDK之前，您需要拥有一个华为账号并实名认证、为账号充值，接着获取访问密钥、准备开发环境，然后下载并安装Java SDK。

### 步骤1 注册华为账号并实名认证。

如果您已有华为账号，请忽略此步骤。如果您还没有华为账号，请执行以下操作：

1. [注册华为账号并开通华为云](#)。
2. 参考[个人账号如何完成实名认证](#)或[企业账号如何完成实名认证](#)，完成个人或企业账号实名认证。

### 步骤2 为账号充值。

您需要确保账号有足够的余额，才能正常使用OBS等相关资源。请参考[账户充值](#)。

### 步骤3 获取访问密钥，详情请参见[获取访问密钥（AK和SK）](#)。

### 步骤4 准备开发环境，详情请参见[准备开发环境](#)。

### 步骤5 下载与安装Java SDK，详情请参见[下载与安装Java SDK](#)。

----结束

## 步骤一：创建桶

本示例用于创建名为examplebucket的桶，并设置所在区域在华北-北京四（cn-north-4）中国-香港（ap-southeast-1），桶的权限访问控制策略是私有桶，将存储类别设置为标准存储，数据冗余策略选择多AZ存储。

### Java

```
import com.obs.services.ObsClient;
import com.obs.services.exception.ObsException;
import com.obs.services.model.AccessControlList;
```

```
import com.obs.services.model.AvailableZoneEnum;
import com.obs.services.model.CreateBucketRequest;
import com.obs.services.model.ObsBucket;
import com.obs.services.model.StorageClassEnum;

public class CreateBucket001 {
    public static void main(String[] args) {
        // 您可以通过环境变量获取访问密钥AK/SK，也可以使用其他外部引入方式传入。如果使用硬编码可能会存在泄露风险。
        // 您可以登录访问管理控制台获取访问密钥AK/SK
        String ak = System.getenv("ACCESS_KEY_ID");
        String sk = System.getenv("SECRET_ACCESS_KEY_ID");

        // endpoint填写桶所在的endpoint, 此处以华北-北京四为例，其他地区请按实际情况填写。
        String endPoint = "https://obs.cn-north-4.myhuaweicloud.com";
        // 创建ObsClient实例
        // 使用永久AK/SK初始化客户端
        ObsClient obsClient = new ObsClient(ak, sk, endPoint);

        try {
            CreateBucketRequest request = new CreateBucketRequest();
            //示例桶名
            String exampleBucket = "examplebucket";
            //示例桶区域位置
            String exampleLocation = "cn-north-4";
            request.setBucketName(exampleBucket);
            // 设置桶访问权限为私有读写，默认也是私有读写
            request.setAcl(AccessControlList.REST_CANNED_PRIVATE);
            // 设置桶的存储类别为标准存储
            request.setBucketStorageClass(StorageClassEnum.STANDARD);
            // 设置桶区域位置(以区域为华北-北京四为例)，location 需要与 endpoint的位置信息一致
            request.setLocation(exampleLocation);
            // 指定创建多AZ桶，如果不设置，默认创建单AZ桶
            request.setAvailableZone(AvailableZoneEnum.MULTI_AZ);
            // 创建桶
            ObsBucket bucket = obsClient.createBucket(request);
            // 创建桶成功
            System.out.println("CreateBucket successfully");
            System.out.println("RequestId:" + bucket.getRequestId());

            } catch (ObsException e) {
                System.out.println("CreateBucket failed");
                // 请求失败,打印http状态码
                System.out.println("HTTP Code: " + e.getHttpResponseCode());
                // 请求失败,打印服务端错误码
                System.out.println("Error Code:" + e.getErrorCode());
                // 请求失败,打印详细错误信息
                System.out.println("Error Message: " + e.getErrorMessage());
                // 请求失败,打印请求id
                System.out.println("Request ID:" + e.getRequestId());
                System.out.println("Host ID:" + e.getHostId());
            } catch (Exception e) {
                System.out.println("CreateBucket failed");
                // 其他异常信息打印
                e.printStackTrace();
            }
        }
    }
}
```

## Python

```
from obs import CreateBucketHeader, HeadPermission
from obs import ObsClient
import os
import traceback

# 推荐通过环境变量获取AKSK，这里也可以使用其他外部引入方式传入，如果使用硬编码可能会存在泄露风险。
```

```
# 您可以登录访问管理控制台获取访问密钥AK/SK
ak = os.getenv("AccessKeyId")
sk = os.getenv("SecretAccessKey")
# server填写Bucket对应的Endpoint, 这里以华北-北京四为例, 其他地区请按实际情况填写。
server = "https://obs.cn-north-4.myhuaweicloud.com"
# 创建obsClient实例
obsClient = ObsClient(access_key_id=ak, secret_access_key=sk, server=server)
try:
    # 创建桶的附加头域, 桶的访问控制策略是私有桶, 存储类型是标准存储, 多AZ方式存储
    header = CreateBucketHeader(aclControl=HeadPermission.PRIVATE, storageClass="STANDARD",
availableZone="3az")
    # 指定存储桶所在区域, 此处以 "cn-north-4" 为例, 必须跟传入的Endpoint中Region保持一致。
    location = "cn-north-4"
    bucketName = "examplebucket"
    # 创建桶
    resp = obsClient.createBucket(bucketName, header, location)
    # 返回码为2xx时, 接口调用成功, 否则接口调用失败
    if resp.status < 300:
        print('Create Bucket Succeeded')
        print('requestId:', resp.requestId)
    else:
        print('Create Bucket Failed')
        print('requestId:', resp.requestId)
        print('errorCode:', resp.errorCode)
        print('errorMessage:', resp.errorMessage)
except:
    print('Create Bucket Failed')
    print(traceback.format_exc())
```

## Go

```
package main

import (
    "fmt"
    "os"
    obs "github.com/huaweicloud/huaweicloud-sdk-go-obs/obs"
)

func main() {
    //推荐通过环境变量获取AKSK, 这里也可以使用其他外部引入方式传入, 如果使用硬编码可能会存在泄露风险。
    //您可以登录访问管理控制台获取访问密钥AK/SK
    ak := os.Getenv("AccessKeyId")
    sk := os.Getenv("SecretAccessKey")
    // endpoint填写Bucket对应的Endpoint, 这里以华北-北京四为例, 其他地区请按实际情况填写。
    endPoint := "https://obs.cn-north-4.myhuaweicloud.com"
    // 创建obsClient实例
    obsClient, err := obs.New(ak, sk, endPoint)
    if err != nil {
        fmt.Printf("Create obsClient error, errMsg: %s", err.Error())
    }
    input := &obs.CreateBucketInput{}
    // 指定存储桶名称
    input.Bucket = "examplebucket"
    // 指定存储桶所在区域, 此处以 "cn-north-4" 为例, 必须跟传入的Endpoint中Region保持一致。
    input.Location = "cn-north-4"
    // 指定存储桶的权限控制策略, 此处以obs.AclPrivate为例。
    input.ACL = obs.AclPrivate
    // 指定存储桶的存储类型, 此处以obs.StorageClassWarm为例。如果未指定该参数, 则创建的桶为标准存储类型。
    input.StorageClass = obs.StorageClassWarm
    // 指定存储桶的AZ类型, 此处以 "3AZ" 为例。不携带时默认为单AZ, 如果对应region不支持多AZ存储, 则该桶的存储类型仍为单AZ。
    input.AvailableZone = "3az"
    // 创建桶
    output, err := obsClient.CreateBucket(input)
    if err == nil {
        fmt.Printf("Create bucket:%s successful!\n", input.Bucket)
```

```
        fmt.Printf("RequestId:%s\n", output.RequestId)
        return
    }
    fmt.Printf("Create bucket:%s fail!\n", input.Bucket)
    if obsError, ok := err.(obs.ObsError); ok {
        fmt.Println("An ObsError was found, which means your request sent to OBS was rejected with an error response.")
        fmt.Println(obsError.Error())
    } else {
        fmt.Println("An Exception was found, which means the client encountered an internal problem when attempting to communicate with OBS, for example, the client was unable to access the network.")
        fmt.Println(err)
    }
}
```

## 步骤二：上传对象

以下示例展示了上传本地文件localfile到examplebucket桶中，并将对象名设置为objectname。

### Java

```
import com.obs.services.ObsClient;
import com.obs.services.exception.ObsException;
import com.obs.services.model.PutObjectRequest;
import java.io.File;
public class PutObject004 {
    public static void main(String[] args) {
        // 您可以通过环境变量获取访问密钥AK/SK，也可以使用其他外部引入方式传入。如果使用硬编码可能会存在泄露风险
        // 您可以登录访问管理控制台获取访问密钥AK/SK
        String ak = System.getenv("ACCESS_KEY_ID");
        String sk = System.getenv("SECRET_ACCESS_KEY_ID");
        // endpoint填写桶所在的endpoint，此处以华北-北京四为例，其他地区请按实际情况填写
        String endPoint = "https://obs.cn-north-4.myhuaweicloud.com";

        // 创建ObsClient实例
        // 使用永久AK/SK初始化客户端
        ObsClient obsClient = new ObsClient(ak, sk,endPoint);

        try {
            // 文件上传
            PutObjectRequest request = new PutObjectRequest();
            // 上传文件到examplebucket桶中
            request.setBucketName("examplebucket");
            // 指定上传到examplebucket桶中的文件名称
            request.setObjectKey("objectname");
            // 指定本地待上传文件的路径
            request.setFile(new File("localfile"));
            obsClient.putObject(request);
            System.out.println("putObject successfully");
        } catch (ObsException e) {
            System.out.println("putObject failed");
            // 请求失败,打印http状态码
            System.out.println("HTTP Code:" + e.getHttpResponseCode());
            // 请求失败,打印服务端错误码
            System.out.println("Error Code:" + e.getErrorCode());
            // 请求失败,打印详细错误信息
            System.out.println("Error Message:" + e.getErrorMessage());
            // 请求失败,打印请求id
            System.out.println("Request ID:" + e.getErrorRequestId());
            System.out.println("Host ID:" + e.getErrorHostId());
            e.printStackTrace();
        } catch (Exception e) {
            System.out.println("putObject failed");
            // 其他异常信息打印
            e.printStackTrace();
        }
    }
}
```

```
    }
```

## Python

```
from obs import ObsClient
from obs import PutObjectHeader
import os
import traceback

# 推荐通过环境变量获取AKSK，这里也可以使用其他外部引入方式传入，如果使用硬编码可能会存在泄露风险。
# 您可以登录访问管理控制台获取访问密钥AK/SK
# 运行本代码示例之前，请确保已设置环境变量AccessKeyId和SecretAccessKey。
ak = os.getenv("AccessKeyId")
sk = os.getenv("SecretAccessKey")
# server填写Bucket对应的Endpoint，这里以华北-北京四为例，其他地区请按实际情况填写。
server = "https://obs.cn-north-4.myhuaweicloud.com"
# 创建obsClient实例
obsClient = ObsClient(access_key_id=ak, secret_access_key=sk, server=server)
try:
    # 上传对象的附加头域
    headers = PutObjectHeader()
    # 上传到名为examplebucket的桶中
    bucketName = "examplebucket"
    # 对象名，即上传后的文件名
    objectKey = "objectname"
    # 待上传文件/文件夹的本地完整路径，如aa/bb.txt，或aa/
    file_path = 'localfile'
    # 文件上传
    resp = obsClient.putFile(bucketName, objectKey, file_path, headers)
    # 返回码为2xx时，接口调用成功，否则接口调用失败
    if resp.status < 300:
        print('Put File Succeeded')
        print('requestId:', resp.requestId)
        print('etag:', resp.body.etag)
        print('versionId:', resp.body.versionId)
        print('storageClass:', resp.body.storageClass)
    else:
        print('Put File Failed')
        print('requestId:', resp.requestId)
        print('errorCode:', resp.errorCode)
        print('errorMessage:', resp.errorMessage)
except:
    print('Put File Failed')
    print(traceback.format_exc())
```

## Go

```
package main
import (
    "fmt"
    "os"
    obs "github.com/huaweicloud/huaweicloud-sdk-go-obs/obs"
)
func main() {
    //推荐通过环境变量获取AKSK，这里也可以使用其他外部引入方式传入，如果使用硬编码可能会存在泄露风险。
    //您可以登录访问管理控制台获取访问密钥AK/SK
    ak := os.Getenv("AccessKeyId")
    sk := os.Getenv("SecretAccessKey")
    // endpoint填写Bucket对应的Endpoint，这里以华北-北京四为例，其他地区请按实际情况填写。
    endPoint := "https://obs.cn-north-4.myhuaweicloud.com"
    // 创建obsClient实例
    obsClient, err := obs.New(ak, sk, endPoint)
    if err != nil {
        fmt.Printf("Create obsClient error, errMsg: %s", err.Error())
    }
    input := &obs.PutFileInput{
        // 指定存储桶名称
    }
```

```
input.Bucket = "examplebucket"
// 指定上传对象，此处以 objectname 为例。
input.Key = "objectname"
// 指定本地文件，此处以localfile为例
input.SourceFile = "localfile"
// 文件上传
output, err := obsClient.PutFile(input)
if err == nil {
    fmt.Printf("Put file(%s) under the bucket(%s) successful!\n", input.Key, input.Bucket)
    fmt.Printf("StorageClass:%s, ETag:%s\n",
        output.StorageClass, output.ETag)
    return
}
fmt.Printf("Put file(%s) under the bucket(%s) fail!\n", input.Key, input.Bucket)
if obsError, ok := err.(obs.ObsError); ok {
    fmt.Println("An ObsError was found, which means your request sent to OBS was rejected with an error
response.")
    fmt.Println(obsError.Error())
} else {
    fmt.Println("An Exception was found, which means the client encountered an internal problem when
attempting to communicate with OBS, for example, the client was unable to access the network.")
    fmt.Println(err)
}
```

### 步骤三：下载对象

本示例用于下载examplebucket桶中的objectname对象。

#### Java

```
import com.obs.services.ObsClient;
import com.obs.services.exception.ObsException;
import com.obs.services.model.ObsObject;
import java.io.ByteArrayOutputStream;
import java.io.InputStream;
public class GetObject001 {
    public static void main(String[] args) {
        // 您可以通过环境变量获取访问密钥AK/SK，也可以使用其他外部引入方式传入。如果使用硬编码可能会存
        // 在泄露风险。
        // 您可以登录访问管理控制台获取访问密钥AK/SK
        String ak = System.getenv("ACCESS_KEY_ID");
        String sk = System.getenv("SECRET_ACCESS_KEY_ID");
        // endpoint填写桶所在的endpoint，此处以华北-北京四为例，其他地区请按实际情况填写。查看桶所在的
        // endpoint请参见：https://support.huaweicloud.com/usermanual-obs/obs_03_0312.html。
        String endPoint = "https://obs.cn-north-4.myhuaweicloud.com";

        // 创建ObsClient实例
        // 使用永久AK/SK初始化客户端
        ObsClient obsClient = new ObsClient(ak, sk,endPoint);

        try {
            // 流式下载
            ObsObject obsObject = obsClient.getObject("examplebucket", "objectname");
            // 读取对象内容
            System.out.println("Object content:");
            InputStream input = obsObject.getObjectContent();
            byte[] b = new byte[1024];
            ByteArrayOutputStream bos = new ByteArrayOutputStream();
            int len;
            while ((len = input.read(b)) != -1) {
                bos.write(b, 0, len);
            }
            System.out.println("getObjectContent successfully");
            System.out.println(new String(bos.toByteArray()));
            bos.close();
            input.close();
        } catch (ObsException e) {
```

```
System.out.println("getObjectContent failed");
// 请求失败,打印http状态码
System.out.println("HTTP Code:" + e.getResponseCode());
// 请求失败,打印服务端错误码
System.out.println("Error Code:" + e.getErrorCode());
// 请求失败,打印详细错误信息
System.out.println("Error Message:" + e.getErrorMessage());
// 请求失败,打印请求id
System.out.println("Request ID:" + e.getErrorRequestId());
System.out.println("Host ID:" + e.getErrorHostId());
e.printStackTrace();
} catch (Exception e) {
    System.out.println("getObjectContent failed");
    // 其他异常信息打印
    e.printStackTrace();
}
}
```

## Python

```
from obs import GetObjectRequest
from obs import ObsClient
import os
import traceback

# 推荐通过环境变量获取AKSK, 这里也可以使用其他外部引入方式传入。如果使用硬编码可能会存在泄露风险
# 您可以登录访问管理控制台获取访问密钥AK/SK
ak = os.getenv("AccessKeyId")
sk = os.getenv("SecretAccessKey")
# server填写Bucket对应的Endpoint, 这里以华北-北京四为例, 其他地区请按实际情况填写。
server = "https://obs.cn-north-4.myhuaweicloud.com"
# 创建obsClient实例
obsClient = ObsClient(access_key_id=ak, secret_access_key=sk, server=server)
try:
    # 下载对象的附加请求参数
    getObjectRequest = GetObjectRequest()
    # 获取对象时重写响应中的Content-Type头。
    getObjectRequest.content_type = 'text/plain'
    bucketName="examplebucket"
    objectKey="objectname"
    #流式下载
    resp = obsClient.getObject(bucketName=bucketName,objectKey=objectKey,
    getObjectRequest=getObjectRequest, loadStreamInMemory=False)
    # 返回码为2xx时, 接口调用成功, 否则接口调用失败
    if resp.status < 300:
        print('Get Object Succeeded')
        print('requestId:', resp.requestId)
        # 读取对象内容
        while True:
            chunk = resp.body.response.read(65536)
            if not chunk:
                break
            print(chunk)
        resp.body.response.close()
    else:
        print('Get Object Failed')
        print('requestId:', resp.requestId)
        print('errorCode:', resp.errorCode)
        print('errorMessage:', resp.errorMessage)
except:
    print('Get Object Failed')
    print(traceback.format_exc())
```

## Go

```
package main
import (
    "fmt"
```

```

"os"
  obs "github.com/huaweicloud/huaweicloud-sdk-go-obs/obs"
)
func main() {
  //推荐通过环境变量获取AKSK，这里也可以使用其他外部引入方式传入，如果使用硬编码可能会存在泄露风险。
  //您可以登录访问管理控制台获取访问密钥AK/SK
  ak := os.Getenv("AccessKeyID")
  sk := os.Getenv("SecretAccessKey")
  // endpoint填写Bucket对应的Endpoint, 这里以华北-北京四为例，其他地区请按实际情况填写。
  endPoint := "https://obs.cn-north-4.myhuaweicloud.com"
  // 创建obsClient实例
  obsClient, err := obs.New(ak, sk, endPoint)
  if err != nil {
    fmt.Printf("Create obsClient error, errMsg: %s", err.Error())
  }
  input := &obs.GetObjectInput{}
  // 指定存储桶名称
  input.Bucket = "examplebucket"
  // 指定下载对象，此处以 objectname 为例。
  input.Key = "objectname"
  // 流式下载对象
  output, err := obsClient.GetObject(input)
  if err == nil {
    // output.Body 在使用完毕后必须关闭，否则会造成连接泄漏。
    defer output.Body.Close()
    fmt.Printf("Get object(%s) under the bucket(%s) successful!\n", input.Key, input.Bucket)
    fmt.Printf("StorageClass:%s, ETag:%s, ContentType:%s, ContentLength:%d, LastModified:%s\n",
      output.StorageClass, output.ETag, output.ContentType, output.ContentLength, output.LastModified)
    // 读取对象内容
    p := make([]byte, 1024)
    var readErr error
    var readCount int
    for {
      readCount, readErr = output.Body.Read(p)
      if readCount > 0 {
        fmt.Printf("%s", p[:readCount])
      }
      if readErr != nil {
        break
      }
    }
    return
  }
  fmt.Printf("List objects under the bucket(%s) fail!\n", input.Bucket)
  if obsError, ok := err.(obs.ObsError); ok {
    fmt.Println("An ObsError was found, which means your request sent to OBS was rejected with an error response.")
    fmt.Println(obsError.Error())
  } else {
    fmt.Println("An Exception was found, which means the client encountered an internal problem when attempting to communicate with OBS, for example, the client was unable to access the network.")
    fmt.Println(err)
  }
}

```

## 步骤四：简单列举对象

以下代码展示如何简单列举examplebucket桶下的对象。

### Java

```

import com.obs.services.ObsClient;
import com.obs.services.exception.ObsException;
import com.obs.services.model.ObjectListing;
import com.obs.services.model.ObsObject;
public class ListObjects001 {
  public static void main(String[] args) {

```

```
// 您可以通过环境变量获取访问密钥AK/SK，也可以使用其他外部引入方式传入。如果使用硬编码可能会存在泄露风险。  
// 您可以登录访问管理控制台获取访问密钥AK/SK  
String ak = System.getenv("ACCESS_KEY_ID");  
String sk = System.getenv("SECRET_ACCESS_KEY_ID");  
// endpoint填写桶所在的endpoint, 此处以华北-北京四为例，其他地区请按实际情况填写。  
String endPoint = "https://obs.cn-north-4.myhuaweicloud.com";  
  
// 创建ObsClient实例  
// 使用永久AK/SK初始化客户端  
ObsClient obsClient = new ObsClient(ak, sk,endPoint);  
  
try {  
    // 简单列举  
    ObjectListing result = obsClient.listObjects("examplebucket");  
    for (ObsObject obsObject : result.getObjects()) {  
        System.out.println("listObjects successfully");  
        System.out.println("ObjectKey:" + obsObject.getObjectKey());  
        System.out.println("Owner:" + obsObject.getOwner());  
    }  
} catch (ObsException e) {  
    System.out.println("listObjects failed");  
    // 请求失败,打印http状态码  
    System.out.println("HTTP Code:" + e.getHttpResponseCode());  
    // 请求失败,打印服务端错误码  
    System.out.println("Error Code:" + e.getErrorCode());  
    // 请求失败,打印详细错误信息  
    System.out.println("Error Message:" + e.getErrorMessage());  
    // 请求失败,打印请求id  
    System.out.println("Request ID:" + e.getRequestId());  
    System.out.println("Host ID:" + e.getHostId());  
    e.printStackTrace();  
} catch (Exception e) {  
    System.out.println("listObjects failed");  
    // 其他异常信息打印  
    e.printStackTrace();  
}  
}
```

## Python

```
from obs import ObsClient  
import os  
import traceback  
  
# 推荐通过环境变量获取AKSK，这里也可以使用其他外部引入方式传入，如果使用硬编码可能会存在泄露风险。  
# 您可以登录访问管理控制台获取访问密钥AK/SK  
ak = os.getenv("AccessKeyId")  
sk = os.getenv("SecretAccessKey")  
# server填写Bucket对应的Endpoint, 这里以华北-北京四为例，其他地区请按实际情况填写。  
server = "https://obs.cn-north-4.myhuaweicloud.com"  
# 创建obsClient实例  
obsClient = ObsClient(access_key_id=ak, secret_access_key=sk, server=server)  
try:  
    bucketName = "examplebucket"  
    # 指定列举对象的前缀  
    prefix = 'test/'  
    # 指定单次列举对象个数为100  
    max_keys = 100  
    # 列举桶内对象  
    resp = obsClient.listObjects(bucketName, prefix, max_keys=max_keys, encoding_type='url')  
  
    # 返回码为2xx时，接口调用成功，否则接口调用失败  
    if resp.status < 300:  
        print('List Objects Succeeded')  
        print('requestId:', resp.requestId)  
        print('name:', resp.body.name)  
        print('prefix:', resp.body.prefix)
```

```
print('max_keys:', resp.body.max_keys)
print('is_truncated:', resp.body.is_truncated)
index = 1
for content in resp.body.contents:
    print('object [' + str(index) + ']')
    print('key:', content.key)
    print('lastModified:', content.lastModified)
    print('etag:', content.etag)
    print('size:', content.size)
    print('storageClass:', content.storageClass)
    print('owner_id:', content.owner.owner_id)
    print('owner_name:', content.owner.owner_name)
    index += 1
else:
    print('List Objects Failed')
    print('requestId:', resp.requestId)
    print('errorCode:', resp.errorCode)
    print('errorMessage:', resp.errorMessage)
except:
    print('List Objects Failed')
    print(traceback.format_exc())
```

## Go

```
package main
import (
    "fmt"
    "os"
    obs "github.com/huaweicloud/huaweicloud-sdk-go-obs/obs"
)
func main() {
    //推荐通过环境变量获取AKSK，这里也可以使用其他外部引入方式传入，如果使用硬编码可能会存在泄露风险。
    //您可以登录访问管理控制台获取访问密钥AK/SK
    ak := os.Getenv("AccessKeyId")
    sk := os.Getenv("SecretAccessKey")

    // endpoint填写Bucket对应的Endpoint, 这里以华北-北京四为例，其他地区请按实际情况填写。
    endPoint := "https://obs.cn-north-4.myhuaweicloud.com"
    // 创建obsClient实例
    obsClient, err := obs.New(ak, sk, endPoint)
    if err != nil {
        fmt.Printf("Create obsClient error, errMsg: %s", err.Error())
    }
    input := &obs.ListObjectsInput{}
    // 指定存储桶名称
    input.Bucket = "examplebucket"
    // 列举桶内对象
    output, err := obsClient.ListObjects(input)
    if err == nil {
        fmt.Printf("List objects under the bucket(%s) successful!\n", input.Bucket)
        fmt.Printf("RequestId:%s\n", output.RequestId)
        for index, val := range output.Contents {
            fmt.Printf("Content[%d]-OwnerId:%s, ETag:%s, Key:%s, LastModified:%s, Size:%d\n",
                index, val.Owner.ID, val.ETag, val.Key, val.LastModified, val.Size)
        }
        return
    }
    fmt.Printf("List objects under the bucket(%s) fail!\n", input.Bucket)
    if obsError, ok := err.(obs.ObsError); ok {
        fmt.Println("An ObsError was found, which means your request sent to OBS was rejected with an error response.")
        fmt.Println(obsError.Error())
    } else {
        fmt.Println("An Exception was found, which means the client encountered an internal problem when attempting to communicate with OBS, for example, the client was unable to access the network.")
        fmt.Println(err)
    }
}
```

## 相关信息

当您完成创建桶、上传对象、下载对象等基本操作后，您还可以结合业务需求使用以下Java SDK的高阶功能。

- **生命周期**：通过为桶配置生命周期规则，可以实现定时转换对象存储类别或定时删除对象。
- **桶ACL权限**：Java SDK提供桶ACL访问权限方式，桶的所有者可以通过编写桶ACL，实现对桶更精细化的权限控制。
- **桶策略**：Java SDK提供桶策略访问权限方式，桶的所有者可以通过编写桶策略，实现对桶更精细化的权限控制。
- **静态网站托管**：您可以将静态网站文件上传至OBS的桶中，并对这些文件赋予匿名用户可读权限，然后将该桶配置成静态网站托管模式，实现使用桶域名访问该网站。
- **多版本控制**：为桶开启多版本控制后，可以在桶中保留多个版本的对象，方便检索和还原各个版本，在意外操作或应用程序故障时帮助快速恢复数据。
- **服务端加密**：通过服务端加密功能，对上传至OBS桶中的数据进行加密保护。
- **跨域资源共享 (CORS)**：通过配置CORS规则，可以实现跨域名访问OBS。

# 5 入门实践

当您完成了创建桶、上传对象和下载对象等基本操作后，可以根据自身的业务需求使用OBS提供的一系列常用实践。

表 5-1 常用最佳实践

实践		描述
数据安全	<a href="#">安全最佳实践</a>	本最佳实践提供了OBS使用过程中的安全最佳实践，旨在为提高整体安全能力提供可操作的规范性指导。根据该指导文档您可以持续评估OBS资源的安全状态，更好的组合使用OBS提供的多种安全能力，提高对OBS资源的整体安全防御能力，保护存储在OBS桶内的数据不泄露、不被篡改，以及数据传输过程中不泄露、不被篡改。
	<a href="#">企业数据权限控制最佳实践</a>	本最佳实践提供了企业开通OBS后可以设置的四种常见权限控制。 <ul style="list-style-type: none"><li>为不同职能部门的员工设置不同的访问权限，以此达到不同部门人员访问公司数据的权限隔离。</li><li>设置权限允许其他部门/项目用户下载共享数据，禁止写删。</li><li>给各业务部门分配各自所需的IAM用户，通过桶策略给每个业务部门下的IAM用户授予独立的资源权限。</li><li>通过OBS Browser+添加外部桶的方式实现业务部门之间桶资源隔离。</li></ul>
数据迁移	<a href="#">搬迁本地数据至OBS</a>	华为云提供多种搬迁方案，帮助用户将自建存储服务器上的数据短时间、低成本、安全、高效地搬迁至OBS。用户可根据数据量、耗时、费用等需求选择适合的方案进行数据搬迁。

实践	描述
	<a href="#">迁移第三方云厂商数据至OBS</a> 针对迁移第三方云厂商的对象数据至OBS的场景，华为云提供OMS服务。通过迁移服务，用户只需在控制台配置简单的连接参数以及迁移任务，即可把数据从第三方云厂商轻松、平滑地迁移至OBS。
	<a href="#">大数据场景下使用OBS实现存算分离</a> 本最佳实践主要提供华为云大数据存算分离方案中不同大数据平台和大数据组件与对象存储服务OBS的对接指导，以及HDFS数据迁移至对象存储服务OBS的方案。
数据备份	<a href="#">使用备份软件实现本地数据备份至OBS</a> Commvault、爱数云备份服务（AnyBackup Cloud）等第三方备份软件，都支持对接OBS进行数据备份。通过这些备份软件，用户可以根据自身需求制定合适的备份策略，达到安全、高效的备份目的。
数据访问	<a href="#">在ECS上通过内网访问OBS</a> 在ECS上可以通过内网访问OBS，即可在内网进行数据读取、备份归档等业务，而不影响外网带宽。
	<a href="#">通过CDN加速访问OBS</a> OBS提供低成本的存储，华为云CDN可以提供网站加速、文件下载加速和点播加速。将数据存放在OBS中然后通过配置CDN加速，这样构造的业务系统可以在降低成本的同时，提高终端用户使用感受。当终端用户发起访问请求时，会首先通过CDN查找对此域名响应速度最快的CDN节点，并查询此节点是否有缓存终端用户请求的内容。
	<a href="#">通过Nginx反向代理访问OBS</a> 出于安全考虑，某些企业出于安全要求对可访问的外部地址需要设置黑白名单，而这个时候对于OBS的访问则需要一个固定的IP地址。同样出于安全考虑，华为云OBS桶访问域名通过DNS解析的IP地址是会发生变化的，所以用户无法获取某个桶长期有效的固定IP地址。 此时，可以通过在ECS上搭建Nginx反向代理服务器，来实现通过固定IP地址访问OBS。
	<a href="#">使用自定义域名托管静态网站</a> 当企业有大量静态网站需要提供给用户访问，却不想搭建网站服务器时，可以将静态网站托管在OBS桶中，用户可以直接通过OBS桶绑定的自定义域名访问托管的静态网站。

实践	描述
数据传输	<a href="#">Web端通过PostObject接口直传OBS</a>
	<a href="#">移动应用直传</a>
	<a href="#">小程序直传OBS</a>
	<a href="#">将WordPress远程附件存储到OBS</a>
数据处理	<a href="#">CDN下载加速</a>
	<a href="#">内容审核-图片审核</a>
	<a href="#">语音识别解决方案</a>
	<a href="#">无服务器图片生成缩略图</a>