

目录

目录	1
存储类型介绍	2
存储类型对比	2
创建存储空间	2
命名规则	2
操作方式	2
权限控制	3
查看存储空间列表	3
操作方式	3
删除存储空间	3
操作方式	3
重定向回源（原镜像）	3
使用场景	3
操作方式	4
重定向回源流程	4
细节说明	4
文件预推	4
镜像回源	4
使用场景	4
使用限制	5
操作方式	5
镜像方式	5
细节说明	5
多条规则匹配情况举例：	6
回源	6
1. 镜像回源	6
2. 重定向回源，即原镜像	6

存储类型介绍

根据数据的访问频度和成本的不同需求，KS3提供标准、低频、归档三种存储类型：

- **标准存储类型**：提供通用的对象存储服务，适合频繁访问、有热点存在的各类音视频、图片、网站静态资源的数据，较低的延迟和较高的吞吐量性能的数据。
- **低频访问存储类型**：适用于较低访问频率的业务场景，适合长期保存、较少访问的数据。低频存储在降低存储价格的基础上同时保持访问延时在毫秒级，保证了用户在取回数据的场景下，无需等待，高速读取。但对存储时长有一定要求，存储时间短于30天的文件提前删除会产生提前删除费用。无论是在内网还是外网，数据获取都会产生数据取回费用。
- **归档存储类型**：适用于数据存储后访问频率极低的业务场景，适合保存需要长期存储、极少访问的数据。对比标准存储和低频存储，归档存储能提供相同的数据可靠性和服务可用性，同时又大幅降低了存储的成本，但数据进入到可读取状态需要1~10分钟的解冻时间，适合需要长期保存的医疗影像、档案、业务日志及影像素材等。归档存储类型有最短存储时间，存储时间短于90天的Object提前删除会产生提前删除费用，另外，读取归档存储中的数据时，会产生数据取回费用。有关金山云KS3各类型存储价格的相关信息，请参见[产品报价](#)。

存储类型对比

对比指标	标准存储类型	低频访问存储	归档存储类型
数据可靠性	99.999999999%	99.999999999%	99.999999999%
访问可用性	99.95%	99.9%	99%（数据解冻后）
短存储时间	无	30天	90天
访问延时	实时访问 ms延迟	实时访问 ms延迟	解冻时间1~10分钟，解冻后可正常访问
图片处理	支持	支持	解冻后支持
适应场景	音视频分发源站、图片分享、大型网站、移动应用	企业应用与数据库备份，监控摄像头数据，影视素材	医疗影像、档案、业务日志，影像素材

- 有关金山云KS3低频存储的接口访问细节，请参见[低频存储](#)。
- 有关金山云KS3归档存储的接口访问细节，请参见[归档存储](#)和[解冻接口](#)。

创建存储空间

在上传任何文件到 KS3 之前，您需要首先创建存储空间以用来存储文件。存储空间具有各种配置属性，包括其地理区域、访问权限以及其他元数据。

说明：创建存储空间API详情请参见[PUT Bucket](#)。

注意：存储空间的命名必须符合命名规范。所选定的存储空间名称在KS3的所有现有存储空间名称中必须具有唯一性。创建后不支持更改存储空间名称。

命名规则

- 仅包含小写英文字母（a-z），数字，点（.），中线，即： abcdefghijklmnopqrstuvwxyz0123456789.-
- 必须由字母或数字开头
- 长度在3和63个字符之间
- 不能是IP的形式，类似192.168.0.1
- 不能以kss开头

操作方式

操作方式	说明
控制台	Web 应用程序，直观易用
JAVA SDK	
PHP SDK	丰富、完整的各类语言 SDK demo
Python SDK	
Android SDK	
IOS SDK	
JavaScript SDK	
Node.js SDK	
C# SDK	
C/C++ SDK	

[GO SDK](#)

权限控制

您可以在创建存储空间的时候设置相应的存储空间权限（ACL），也可以在创建之后修改 ACL。如果不设置 ACL，默认值为私有。更多信息，请参见[ACL](#)。

查看存储空间列表

存储空间创建之后，您可以通过KS3 API的Get Service接口获取存储空间列表信息。

说明：查看存储空间API详情请参见[GET Service](#)。

操作方式

操作方式

[控制台](#)

[GUI图形化界面工具 KS3 Explorer](#)

[JAVA SDK](#)

[PHP SDK](#)

[Python SDK](#)

[Android SDK](#)

[IOS SDK](#)

[JavaScript SDK](#)

[Node.js SDK](#)

[C# SDK](#)

[C/C++ SDK](#)

[GO SDK](#)

说明

Web 应用程序，直观易用

图形化工具，易操作

丰富、完整的各类语言 SDK demo

删除存储空间

如果您不再需要存储空间，请将其删除以免进一步产生费用。

说明：删除存储空间之前请确保其中存储的文件已经全部清空，否则无法删除存储空间。

删除存储空间API详情请参见[DELETE Bucket](#)。

操作方式

操作方式

[控制台](#)

[JAVA SDK](#)

[PHP SDK](#)

[Python SDK](#)

[Android SDK](#)

[IOS SDK](#)

[JavaScript SDK](#)

[Node.js SDK](#)

[C# SDK](#)

[C/C++ SDK](#)

[GO SDK](#)

说明

Web 应用程序，直观易用

丰富、完整的各类语言 SDK demo

重定向回源（原镜像）

使用场景

当Bucket配置了重定向回源后，用户向KS3访问一个文件（Object），而该文件并没有保存在KS3中时，KS3会返回给用户一个

Location指向源站的302重定向请求；同时，KS3也会异步向回源地址请求该文件，并保存至KS3中。

重定向回源可以应用到以下场景：

- 其他数据源向KS3的无缝迁移：用户异步的从自己的数据源向KS3迁移数据，在此过程中未迁移到KS3的数据通过URL rewrite的方式返回给用户一个302重定向请求，用户的客户端根据302中的Location从自己的数据源读回数据。
- 配置页面跳转功能：例如用户希望隐藏自己的某些前缀开头的Object，给访问者返回一个特殊的页面。
- 配置发生404错误时的跳转页面：发生以上错误的时候用户可以看到一个预先设定的页面，不至于系统发生错误的时候向用户完全暴露KS3的错误。

注意：

- 利用重定向回源进行数据迁移，单个文件最大支持50GB。
- 当用户访问KS3上没有的文件时，会触发KS3异步向源站下载该文件，KS3会尽快将文件下载下来并保存在KS3，但KS3不保证时间和成功率。
- 支持同时配置多个镜像源，镜像源之间以英文“;”隔开，回源时按照源站配置顺序拉取，拉取成功后，不再从剩余源站尝试拉取。
- 若镜像源还有一个301到另一个源的过程，那么此源上的文件不会被异步拉取到KS3上（支持302）
- 回源地址支持输入域名与IP，多个回源地址之间以英文“;”隔开，地址最大长度为512个字符。

操作方式

支持控制台和API两种操作方式：	操作方式	参考文档
控制台	控制台空间管理-镜像	
API	<ul style="list-style-type: none">• PUT BucketMirror: 为目标桶设置与更新一组镜像回源规则。• GET BucketMirror: 获得源存储空间的镜像回源与重定向回源配置。• DELETE BucketMirror: 删除某个Bucket的BucketMirror设置。	

重定向回源流程

重定向功能的作用是根据设置的回源条件，以及相应的跳转的配置，向用户返回一个302跳转。具体流程如下图所示。

细节说明

1. 触发条件 当客户端Get Object请求的文件不存在时，且用户配置了重定向回源规则，会触发重定向回源，向用户返回指向源站的302跳转。

2. 异步迁移 设置镜像源之后，源站资源会被自动平滑迁移到KS3。访问格式例如：<https://{空间名称}.ks3-cn-bei.jing.ksyuncs.com/{资源路径}>。（注：金山云存储的空间不能作为镜像源）。源站数据默认同步到设置的空间下。

配置镜像源后，由于镜像源和镜像空间的内容基本一致，源站有可能会被搜索引擎封锁，可以配置搜索引擎配置文件 robots.txt 避免该情况发生。

3. 访问控制 您可以将上传到云存储中的文件设定为公开或者私密。

文件预推

您可根据模板中的资源列表，将文件从镜像源站拉取到KS3中，并根据所设定的规则进行存储。

注意：

- 当配置好重定向回源时，才会出现文件预推的配置，归档存储类型的bucket不支持文件预推设置。
- 当设置文件名和文件预推功能同时存在时，设置文件名对文件预推功能无效。

说明：若镜像源还有一个301到另一个源的过程，那么此源上的文件不会被异步拉取到KS3上（支持302）

镜像回源

使用场景

KS3提供同步镜像回源功能，用于数据无缝迁移到KS3的场景，在迁移数据的同时，同时保证业务的正常进行。当用户向KS3请求的文件不存在时，KS3会向回源地址请求这个文件，返回给用户。对于完整文件的请求，KS3会同时请求所有数据返回给用户，并将该文件存入KS3；对于Range请求，KS3会同时请求部分数据返回给用户，然后异步拉取数据存储至KS3。

使用限制

1. 您最多可以配置20条回源规则，系统将按照规则顺序依次执行。
2. KS3镜像回源单个文件最大支持50GB。
3. 该功能目前支持北京、青岛、上海地区。
4. 使用该功能需相应的权限，可联系商务或[提交工单](#)申请权限。

特别说明：

1. 拉取到KS3的数据默认以标准存储类型存储。
2. 拉取到KS3的文件不支持用KS3提供的设置文件名功能来修改文件名。
3. 源站拉取完成后，KS3不感知源站的文件变化。
4. 若用户设置了原图保护，在同步回源时失效。
5. 若用户设置了文件服务端加密/客户端传输密钥加密，在同步回源时失效。

操作方式

支持控制台和API两种操作方式：	操作方式	参考文档
控制台	控制台使用手册-镜像	
API	<ul style="list-style-type: none">• PUT BucketMirror: 为目标桶设置与更新一组镜像回源规则。• GET BucketMirror: 获得源存储空间的镜像回源与重定向回源配置。• DELETE BucketMirror: 删除某个Bucket的BucketMirror设置。	

镜像方式

同步镜像回源流程

当您的客户向Bucket请求的某个Object不存在时，KS3会向回源地址请求这个Object，并且，为了避免等待整个文件需要较长时间，KS3会向源站流式拉取部分数据，源站返回数据后，KS3返回给用户，并且存在对应Bucket中。具体流程如下图所示。

1. 完整文件请求流程



2. 部分（range）文件请求流程：



细节说明

1. 同步回源规则触发条件

当客户端Get Object请求的文件不存在时，且用户配置了该文件的同步镜像回源规则，并且该文件满足规则时，触发同步镜像回源。

2. 请求失败返回规则

- 如果镜像源也不存在此文件，即镜像源返回给KS3的HTTP状态码为404，那么KS3也将返回404给用户。
- 如果镜像源返回给KS3的HTTP status为其他非200的状态码（包括因为网络原因等获取不到文件的错误情况），KS3将返回424给用户，错误码为MirrorFailed。
- 如果源站返回3xx（3XX包含302与301），根据用户的回源规则配置，决定是否follow源站3xx重定向请求获取到资源。

3. 同步回源文件更新规则

如果文件通过镜像回源，存储到了KS3中，之后源站里该文件发生了变化，KS3不会更新已存在于KS3上的该文件。

4. 同步回源文件元信息

Content-Type
Content-Disposition
Cache-Control
Expires

5. HTTP请求规则

传给KS3的Header信息不会传递给源站，QueryString信息是否会传递给源站取决于控制台回源规则中的配置。Header传递规则控制台输入显示参见[控制台使用手册-镜像回源](#)。

如果源站是chunked编码返回，那么KS3首次回源时返回给用户的也是chunked编码。

多条规则匹配情况举例：

当客户存在多条规则且回源条件相同与部分重叠时：

规则一：回源条件全部文件404时回test.com。

规则二：回源条件全部文件404时回141.0.0.12。

规则三：回源条件文件名前缀为333的文件404时回back.com。

KS3的处理说明：

1. 请求优先匹配到规则一，同时test.com可以正常拉取数据，则不再匹配规则二。
2. 请求优先匹配到规则一，但是KS3回源服务前往 test.com时，该源站不可用，则KS3再去前往规则二配置的141.0.0.12站。
3. 若规则二配置的141.0.0.12也恰巧不可用时，如果文件名符合前缀为333，则KS3继续前往规则三配置back.com；如果文件名前缀不是333，则KS3给用户返回404。

回源

KS3的回源规则允许请求者向存储空间（Bucket）请求不存在的文件时，从回源规则设置的源站获取目标文件。回源规则分为镜像回源和重定向回源两种，可以满足您对于数据热迁移、特定请求的重定向等需求。

1. 镜像回源

当客户端通过GetObject请求Bucket中的某个文件不存在时，KS3会向回源地址请求这个文件。获取到该文件后，KS3会将文件存入目标Bucket，并返回给用户。具体流程如下图所示。



- 使用场景 镜像回源主要用于数据无缝迁移到KS3的场景。例如某服务已经在自己建立的源站或者在其他云产品上运行。现因业务发展，需要将数据迁移到KS3上，但是又不能停止服务，此时可以在迁移数据的同时，使用镜像回源功能保证业务的正常进行。
- 细节分析
 - 同步回源规则触发条件 当客户端Get Object请求的文件不存在时，且用户配置了该文件的同步镜像回源规则，并且该文件满足规则时，触发同步镜像回源。
 - 请求失败返回规则 如果镜像源也不存在此文件，即镜像源返回给KS3的HTTP状态码为404，那么KS3也将返回404给用户。如果镜像源返回给KS3的HTTP status为其他非200的状态码（包括因为网络原因等获取不到文件的错误情况），KS3将返回424给用户，错误码为MirrorFailed。如果源站返回3xx（3XX包含302与301），根据用户的回源规则配置，决定是否follow源站3xx重定向请求获取到资源。
 - 同步回源文件更新规则 如果文件通过镜像回源，存储到了KS3中，之后源站里该文件发生了变化，KS3不会更新已存在于KS3上的该文件。
 - 同步回源文件元信息：

```
Content-Type
Content-Disposition
Cache-Control
Expires
```

2. 重定向回源，即原镜像

重定向功能的作用是根据设置的回源条件，以及相应的跳转的配置，向用户返回一个3xx跳转。具体流程如下图所示。



- 使用场景：
 - 其他数据源KS3的无缝迁移 用户异步的从自己的数据源向KS3迁移数据，在此过程中未迁移到KS3的数据通过URL

- rewrite的方式返回给用户一个302重定向请求，用户的客户端根据302中的Location从自己的数据源读回数据。
- 配置页面跳转功能 例如用户希望隐藏自己的某些前缀开头的Object，给访问者返回一个特殊的页面。
 - 配置发生404错误时的跳转页面 发生以上错误的时候用户可以看到一个预先设定的页面，不至于系统发生错误的时候向用户完全暴露KS3的错误。