Google Cloud

Contact Us   Start free

Cloud Storage > Documentation > Reference

Buckets

Send feedback

The Buckets resource represents a bucket in Cloud Storage. There is a single global namespace shared by all buckets. For more information, see bucket name requirements.

Buckets contain objects which can be accessed by their own methods. In addition to the `acl` property, buckets contain bucketAccessControls, for use in fine-grained manipulation of an existing bucket's access controls.

A bucket is always owned by the associated project's `projectOwner` convenience value.

To try out the methods for this resource, such as creating a new bucket, see Methods.

Resource representations

```
{
  "kind": "storage#bucket",
  "selfLink": string,
  "id": string,
  "name": string,
  "projectNumber": "unsigned long",
  "metageneration": "long",
  "location": string,
  "storageClass": string,
  "etag": string,
  "defaultEventBasedHold": boolean,
  "timeCreated": "datetime",
  "updated": "datetime",
  "hierarchicalNamespace": {
    "enabled": boolean
  },
  "encryption": {
    "defaultKmsKeyName": string
  },
  "acl": [
    bucketAccessControls Resource
  ],
  "defaultObjectAcl": [
    defaultObjectAccessControls Resource
```

Property name	Value	Description	Notes
`acl[]`	list	Access controls on the bucket, containing one or more bucketAccessControls Resources. If `iamConfiguration.uniformBucketLevelAccess.enabled` is set to `true`, this field is omitted in responses, and requests that specify this field fail with a `400 Bad Request` response.	writable
`autoclass`	object	The bucket's Autoclass configuration, which, when enabled, controls the storage class of objects based on how and when the objects are accessed.	writable
`autoclass.enabled`	boolean	Whether or not Autoclass is enabled. By default, this boolean is not set, and Autoclass is disabled.	writable

`autoclass.terminalStorageClass`	`string`	The coldest storage class that an object in an Autoclass-enabled bucket transitions to if it is not accessed. Valid values are `NEARLINE` and `ARCHIVE`. The default value is `NEARLINE`.	writable
`autoclass.terminalStorageClassUpdateTime`	`datetime`[1]	The time at which the terminal storage class was last updated for this bucket, in RFC 3339 format.	
`autoclass.toggleTime`	`datetime`[1]	The time at which Autoclass was last enabled or disabled for this bucket, in RFC 3339 format.	
`billing`	`object`	The bucket's billing configuration.	writable
`billing.requesterPays`	`boolean`	When set to true, Requester Pays is enabled for this bucket.	writable
`cors[]`	`list`	The bucket's Cross-Origin Resource Sharing (CORS) configuration.	writable
`cors[].maxAgeSeconds`	`integer`	The value, in seconds, to return in the Access-Control-Max-Age header used in preflight responses.	writable
`cors[].method[]`	`list`	The list of HTTP methods on which to include CORS response headers, such as `"GET"`, `"OPTIONS"`, and `"POST"`. Note: `"*"` is permitted in the list of methods, and means "any method".	writable
`cors[].origin[]`	`list`	The list of Origins eligible to receive CORS response headers. Note: `"*"` is permitted in the list of origins, and means "any Origin".	writable
`cors[].responseHeader[]`	`list`	The list of HTTP headers other than the simple response headers to give permission for the user-agent to share across domains.	writable
`customPlacementConfig`	`object`	Applicable only if a bucket is located in a configurable dual-region. The bucket's custom location configuration.	writable for insert requests
`customPlacementConfig.dataLocations[]`	`list`	The list of individual regions that comprise a configurable dual-region bucket. See Cloud Storage bucket locations for a list of acceptable regions.	writable for insert requests
`defaultEventBasedHold`	`boolean`	Whether or not to automatically apply an eventBasedHold to new objects added to the bucket.	writable
`defaultObjectAcl[]`	`list`	Default access controls to apply to new objects when no ACL is provided. This list contains one or more defaultObjectAccessControls Resources. If `iamConfiguration.uniformBucketLevelAccess.enabled` is set to `true`, this field is omitted in responses, and requests that specify this field fail.	writable
`encryption`	`object`	Encryption configuration for a bucket.	writable
`encryption.defaultKmsKeyName`	`string`	A Cloud KMS key that will be used to encrypt objects written to this bucket if no encryption method is specified as part of the object write request.	writable
`etag`	`string`	HTTP 1.1 Entity tag for the bucket.	
`hierarchicalNamespace.enabled` (preview)	`boolean`	Whether or not hierarchical namespace is enabled for this bucket.	writeable for insert requests
`iamConfiguration`	`object`	The IAM configuration for a bucket.	writable
`iamConfiguration.publicAccessPrevention`	`string`	The bucket's public access prevention status, which is either `"inherited"` or `"enforced"`. If `"inherited"`, the bucket uses public access prevention only if the bucket is subject to the	writable

		public access prevention organization policy constraint. Defaults to `"inherited"`.	
`iamConfiguration.uniformBucketLevelAccess`	object	The bucket's uniform bucket-level access configuration. Note: `iamConfiguration` also includes the `bucketPolicyOnly` field, which uses a legacy name but has the same functionality as the `uniformBucketLevelAccess` field. We recommend only using `uniformBucketLevelAccess`, as specifying both fields may result in unreliable behavior.	writable
`iamConfiguration.uniformBucketLevelAccess.enabled`	boolean	Whether or not the bucket uses uniform bucket-level access. If set, access checks only use bucket-level IAM policies or above.	writable
`iamConfiguration.uniformBucketLevelAccess.lockedTime`	datetime[1]	The deadline time for changing `iamConfiguration.uniformBucketLevelAccess.enabled` from `true` to `false`, in RFC 3339 ⧉ format. `iamConfiguration.uniformBucketLevelAccess.enabled` may be changed from `true` to `false` until the locked time, after which the field is immutable.	
`id`	string	The ID of the bucket. For buckets, the `id` and `name` properties are the same.	
`kind`	string	The kind of item this is. For buckets, this is always `"storage#bucket"`.	
`labels`	object	User-provided bucket labels, in key/value pairs.	writable
`labels.(key)`	string	An individual label entry.	writable
`lifecycle`	object	The bucket's lifecycle configuration. See lifecycle management for more information.	writable
`lifecycle.rule[]`	list	A lifecycle management rule, which is made of an action to take and the condition(s) under which the action will be taken.	writable
`lifecycle.rule[].action`	object	The lifecycle action to take.	writable
`lifecycle.rule[].action.storageClass`	string	The new storage class when `action.type` is `"SetStorageClass"`. See lifecycle actions for a table of supported storage class transitions.	writable
`lifecycle.rule[].action.type`	string	Type of the action. Currently, `"Delete"`, `"SetStorageClass"`, and `"AbortIncompleteMultipartUpload"` are supported.	writable
`lifecycle.rule[].condition`	object	The condition(s) under which the action will be taken.	writable
`lifecycle.rule[].condition.age`	integer	Age of an object (in days). This condition is satisfied when an object reaches the specified age.	writable
`lifecycle.rule[].condition.createdBefore`	date[1]	A date in the RFC 3339 ⧉ format YYYY-MM-DD. This condition is satisfied when an object is created before midnight of the specified date in UTC.	writable
`lifecycle.rule[].condition.customTimeBefore`	date[1]	A date in the RFC 3339 ⧉ format YYYY-MM-DD. This condition is satisfied when the `customTime` metadata for the object is set to an earlier date than the date used in this lifecycle condition.	writable
`lifecycle.rule[].condition.daysSinceCustomTime`	integer	Days since the date set in the `customTime` metadata for the object. This condition is satisfied when the current date and time is at least the specified number of days after the `customTime`.	writable
`lifecycle.rule[].condition.daysSinceNoncurrentTime`	integer	Relevant only for versioned objects. This condition is satisfied when an object has been noncurrent for more than the specified number of days.	writable

`lifecycle.rule[].condition.isLive`	`boolean`	Relevant only for versioned objects. If the value is `true`, this condition matches the live version of objects; if the value is `false`, it matches noncurrent versions of objects.	writable
`lifecycle.rule[].condition.matchesPrefix[]`	`list of strings`	The prefix of an object. This condition is satisfied when the beginning of an object's name is an exact case-sensitive match with the prefix. The prefix must meet object naming requirements. You can specify one or multiple strings as a list.	writable
`lifecycle.rule[].condition.matchesStorageClass[]`	`list`	Objects having any of the storage classes specified by this condition will be matched. Values include `"STANDARD"`, `"NEARLINE"`, `"COLDLINE"`, `"ARCHIVE"`, `"MULTI_REGIONAL"`, `"REGIONAL"`, and `"DURABLE_REDUCED_AVAILABILITY"`.	writable
`lifecycle.rule[].condition.matchesSuffix[]`	`list of strings`	The suffix of an object. This condition is satisfied when the end of an object's name is an exact case-sensitive match with the suffix. The suffix must meet object naming requirements. You can specify one or multiple strings as a list.	writable
`lifecycle.rule[].condition.noncurrentTimeBefore`	`date`[1]	Relevant only for versioned objects. A date in the RFC 3339 ⎘ format YYYY-MM-DD. This condition is satisfied for objects that became noncurrent on a date prior to the one specified in this condition.	writable
`lifecycle.rule[].condition.numNewerVersions`	`integer`	Relevant only for versioned objects. If the value is `N`, this condition is satisfied when there are at least `N` versions (including the live version) newer than this version of the object.	writable
`location`	`string`	The location of the bucket. Object data for objects in the bucket resides in physical storage within this location. Defaults to `"US"`. See Cloud Storage bucket locations for the authoritative list.	writable for insert requests
`locationType`	`string`	The type of location that the bucket resides in. Possible values include `region`, `dual-region`, and `multi-region`.	
`logging`	`object`	The bucket's logging configuration, which defines the destination bucket and optional name prefix for the current bucket's usage logs and storage logs.	writable
`logging.logBucket`	`string`	The destination bucket where the current bucket's logs should be placed.	writable
`logging.logObjectPrefix`	`string`	A prefix for log object names. The default prefix is the bucket name.	writable
`metageneration`	`long`[1]	The metadata generation of this bucket.	
`name`	`string`	The name of the bucket.	writable for insert requests
`objectRetention`	`object`	The bucket's object retention configuration.	
`objectRetention.mode`	`string`	When set to `Enabled`, retention configurations can be set on objects in the bucket.	
`owner`	`object`	The owner of the bucket. This is always the project team's owner group.	
`owner.entity`	`string`	The entity, in the form `"project-owner-`*`projectId`*`"`.	
`owner.entityId`	`string`	The ID for the entity.	
`projectNumber`	`unsigned long`[1]	The project number of the project the bucket belongs to.	
`selfLink`	`string`	The URI of this bucket.	
`retentionPolicy`	`object`	The bucket's retention policy, which defines the minimum age an object in the bucket must reach before it can be deleted or	writable

replaced.

retentionPolicy.effectiveTime	datetime[1]	The time from which the retentionPolicy was effective, in RFC 3339 ⧉ format.	
retentionPolicy.isLocked	boolean	Whether or not the retentionPolicy is locked. If true, the retentionPolicy cannot be removed and the retention period cannot be reduced.	
retentionPolicy.retentionPeriod	unsigned integer[1]	The period of time, in seconds, that objects in the bucket must be retained and cannot be deleted, replaced, or made noncurrent. The value must be greater than 0 seconds and less than 3,155,760,000 seconds.	writable
rpo	string	The recovery point objective for cross-region replication of the bucket. Applicable only for dual- and multi-region buckets. "DEFAULT" uses default replication. "ASYNC_TURBO" enables turbo replication, valid for dual-region buckets only. If rpo is not specified when the bucket is created, it defaults to "DEFAULT". For more information, see redundancy across regions.	writable
softDeletePolicy	object	The bucket's soft delete policy, which defines the period of time during which objects in the bucket are retained in a soft-deleted state after being deleted. Objects in a soft-deleted state cannot be permanently deleted, and are restorable until their hardDeleteTime.	writeabl
softDeletePolicy.effectiveTime	datetime	The datetime at which the soft delete policy becomes effective, in RFC 3339 format. softDeletePolicy.effectiveTime is updated whenever softDeletePolicy.retentionDurationSeconds is increased.	
softDeletePolicy.retentionDurationSeconds	long	The period of time during which a soft-deleted object is retained and cannot be permanently deleted, in seconds. The value must be greater than or equal to 604800 (7 days) and less than 7776000 (90 days). The value can also be set to 0, which disables the soft delete policy.	writeabl
storageClass	string	The bucket's default storage class, used whenever no storageClass is specified for a newly-created object. If storageClass is not specified when the bucket is created, it defaults to "STANDARD". For available storage classes, see Storage classes.	writable
timeCreated	datetime[1]	The creation time of the bucket in RFC 3339 ⧉ format.	
updated	datetime[1]	The time at which the bucket's metadata or IAM policy was last updated, in RFC 3339 ⧉ format.	
versioning	object	The bucket's versioning configuration. For more information, see Object Versioning.	writable
versioning.enabled	boolean	While set to true, versioning is fully enabled for this bucket.	writable
website	object	The bucket's website configuration, controlling how the service behaves when accessing bucket contents as a web site. See the Static Website Examples for more information.	writable
website.mainPageSuffix	string	If the requested object path is missing, the service will ensure the path has a trailing '/', append this suffix, and attempt to retrieve the resulting object. This allows the creation of index.html objects to represent directory pages.	writable
website.notFoundPage	string	If the requested object path is missing, and any mainPageSuffix object is missing, if applicable, the service will return the named object from this bucket as the content for a 404 Not Found ⧉ result.	writabl

[1] This property is a string formatted as the specified value type.

Methods

Available methods for Buckets resources are as follows:

delete

  Deletes a bucket.

get

  Returns metadata for the specified bucket.

getIamPolicy

  Returns an IAM policy for the specified bucket.

insert

  Creates a new bucket.

list

  Retrieves a list of buckets for a given project.

listChannels

  Retrieves a list of active Object Change Notification channels for a bucket.

lockRetentionPolicy

  Irreversibly sets the retention policy on a bucket.

patch

  Updates a bucket. Changes to the bucket will be readable immediately after writing, but configuration changes may take time to propagate. This method supports patch semantics.

setIamPolicy

  Updates an IAM policy for the specified bucket.

testIamPermissions

  Tests a set of permissions on the given bucket to see which, if any, are held by the caller.

update

  Updates a bucket. Changes to the bucket will be readable immediately after writing, but configuration changes may take time to propagate. This method sets the complete metadata of a bucket. If you want to change some of a bucket's metadata while leaving other parts unaffected, use the PATCH method instead.

For information about status and error codes returned by these APIs, see the reference page.

Try it for yourself

If you're new to Google Cloud, create an account to evaluate how Cloud Storage performs in real-world scenarios. New customers also get $300 in free credits to run, test, and deploy workloads.

Try Cloud Storage free

Send feedback

Why Google

Choosing Google Cloud

Trust and security

Modern Infrastructure Cloud

Multicloud

Global infrastructure

Customers and case studies

Analyst reports

Whitepapers

Blog

Products and pricing

Google Cloud pricing

Google Workspace pricing

See all products

Solutions

Infrastructure modernization

Databases

Application modernization

Smart analytics

Artificial Intelligence

Security

Productivity & work transformation

Industry solutions

DevOps solutions

Small business solutions

See all solutions

Resources

Google Cloud Affiliate Program

Google Cloud documentation

Google Cloud quickstarts

Google Cloud Marketplace

Learn about cloud computing

Support

Code samples

Cloud Architecture Center

Training

Certifications

Google for Developers

Google Cloud for Startups

System status

Release Notes

Engage

Contact sales

Find a Partner

Become a Partner

Events

Podcasts

Developer Center

Press Corner

Google Cloud on YouTube

Google Cloud Tech on YouTube

Follow on X

Join User Research

We're hiring. Join Google Cloud!

Google Cloud Community

About Google | Privacy | Site terms | Google Cloud terms

🍃 Our third decade of climate action: join us

Sign up for the Google Cloud newsletter   **Subscribe**