

Package: cozip (via r-universe)

May 22, 2026

Type Package

Title Cloud-Optimized ZIP - Open a ZIP Like a Table

Version 2026.5.22

Date 2026-05-15

Description R bindings to libcozip, the reference library for the Cloud-Optimized ZIP (cozip) format.

License MIT + file LICENSE

URL <https://github.com/asterisk-labs/cozip>

BugReports <https://github.com/asterisk-labs/cozip/issues>

Encoding UTF-8

Language en-US

Depends R (>= 4.1)

Imports arrow (>= 14.0.0), bit64, duckdb, DBI, tibble,

Suggests testthat (>= 3.0.0), withr, sf, geoarrow

SystemRequirements GNU make

Config/testthat/edition 3

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Config/pak/sysreqs cmake make libssl-dev xz-utils

Repository <https://asterisk-labs.r-universe.dev>

Date/Publication 2026-05-22 00:03:48 UTC

RemoteUrl <https://github.com/asterisk-labs/cozip>

RemoteRef main

RemoteSha 522fe90dbdb1dc80ccdc429f26f07990768fc455

RemoteSubdir r

Contents

create	2
read	2
stage_create	3
stage_metadata	4
Index	5

create	<i>Write a cozip archive in one call</i>
--------	--

Description

Wraps `stage_metadata()` + `arrow::write_parquet()` + `stage_create()`. For GeoParquet or custom write options, use the staging functions directly.

Usage

```
create(out_path, table, temp_dir = NULL)
```

```
write(out_path, table, temp_dir = NULL)
```

Arguments

<code>out_path</code>	Destination archive path.
<code>table</code>	<code>arrow::Table</code> with name and path.
<code>temp_dir</code>	Directory for the temp metadata parquet (defaults to <code>tempdir()</code>).

Value

Absolute path of the created archive.

read	<i>Read the manifest of a FLAT-profile cozip archive</i>
------	--

Description

Thin wrapper around the DuckDB cozip extension. `name`, `offset`, `size` (and `cozip:gdal_vsi` when `gdal_vsi = TRUE`) are always included; use `columns` to bring extras, `NULL` brings all.

Usage

```
read(source, columns = NULL, gdal_vsi = TRUE)
```

Arguments

source	Local path or http(s)/s3/gcs/azure/hf URL to the .zip.
columns	Character vector of extra columns. NULL returns every column.
gdal_vsi	Include the cozip:gdal_vsi column.

Value

A tibble.

stage_create	<i>Pack a cozip archive from sources and a user-written parquet</i>
--------------	---

Description

The metadata parquet is embedded verbatim, so any GeoParquet or schema metadata you wrote is preserved.

Usage

```
stage_create(out_path, paths, metadata_parquet, validate = TRUE)
```

Arguments

out_path	Destination archive path.
paths	data.frame with name and path in the same row order as metadata_parquet.
metadata_parquet	Path to a parquet with name, offset, size columns and no path column.
validate	If TRUE, re-runs the plan and checks that the parquet matches.

Value

Absolute path of the created archive.

stage_metadata	<i>Plan a cozip FLAT archive</i>
----------------	----------------------------------

Description

Computes offsets and sizes for the user entries and returns the canonical metadata table plus a row-aligned paths data.frame. No I/O.

Usage

```
stage_metadata(table)
```

Arguments

table arrow::Table with name and path. Extras are preserved in the metadata.

Details

If you reorder metadata before writing it, reorder paths the same way or [stage_create\(\)](#) will reject the mismatch.

Value

List with metadata (arrow::Table) and paths (data.frame).

Index

`arrow::write_parquet()`, 2

`create`, 2

`read`, 2

`stage_create`, 3

`stage_create()`, 2, 4

`stage_metadata`, 4

`stage_metadata()`, 2

`tempdir()`, 2

`write(create)`, 2