

Package: pavdata (via r-universe)

June 19, 2026

Title Transportation Infrastructure Data Toolbox

Version 0.1.0

Author Vilmar Faustino do Nascimento [aut, cre], Carlos David Rodrigues Melo [aut], Nelson de Oliveira Quesado Filho [aut], Jorge Barbosa Soares [con]

Maintainer Vilmar Faustino do Nascimento <vilmarfaustinok@gmail.com>

Description An open-source toolbox for storing, validating, managing, and exploring transportation infrastructure data. Provides a relational data model for binders, aggregates, mixtures, and test results, with a human-readable file format (.pavdata) aligned with FAIR principles.

License MIT + file LICENSE

Encoding UTF-8

Depends R (>= 4.0)

Imports jsonlite, uuid, graphics

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Config/roxygen2/version 8.0.0

NeedsCompilation no

Repository <https://cran.r-universe.dev>

Date/Publication 2026-06-19 18:11:51 UTC

RemoteUrl <https://github.com/cran/pavdata>

RemoteRef HEAD

RemoteSha 5efc55153efa445a6fe3c57a25e2ae3c1d2292bf

Contents

is.pavdata	2
pav_check	3
pav_check_integrity	3

pav_library	4
pav_library_load	4
pav_list	5
pav_load	6
pav_new	6
pav_read	7
pav_view	8
pav_write	8
pavdata_sample_path	9
pavdata_type	10

Index	11
--------------	-----------

is.pavdata	<i>Test whether an object is a PavData object</i>
------------	---

Description

Test whether an object is a PavData object

Usage

```
is.pavdata(x)
```

Arguments

x Any R object.

Value

A single logical value: TRUE if x is a PavData object, FALSE otherwise.

Examples

```
is.pavdata(pav_new("sample", name = "Am_001"))
```

pav_check *Validate a single PavData object*

Description

Checks a PavData object for required fields (level 1) and for values outside physically plausible ranges (level 2). Missing optional fields do not cause failure.

Usage

```
pav_check(x, verbose = TRUE)
```

Arguments

x A PavData object created by [pav_new](#).
 verbose Logical; if TRUE (default), prints the validation outcome and any issues.

Value

Invisibly, a list with `valid` (logical) and `issues` (a character vector of messages).

Examples

```
pav_check(pav_new("binder", name = "CAP 50/70", binder_type = "CAP 50/70"))
```

pav_check_integrity *Check relational integrity of a collection*

Description

Validates every object in a collection and verifies that reference fields (foreign keys) point to existing objects, reporting failed objects and broken references.

Usage

```
pav_check_integrity(objects, verbose = TRUE)
```

Arguments

objects A named list of PavData objects, as returned by [pav_read](#).
 verbose Logical; if TRUE (default), prints a summary of the check.

Value

Invisibly, a list with `valid` (logical), `total`, `failed` and `broken_refs`.

Examples

```
b <- pav_new("binder", name = "CAP 50/70", binder_type = "CAP 50/70")
pav_check_integrity(list(b))
```

pav_library	<i>Create a PavData library</i>
-------------	---------------------------------

Description

Creates an empty in-memory library to store, index and query PavData objects.

Usage

```
pav_library()
```

Value

A pav_library object (an environment) with an empty object store and type index.

Examples

```
lib <- pav_library()
```

pav_library_load	<i>Load a .pavdata file into a library</i>
------------------	--

Description

Reads a .pavdata file and adds all of its objects to an existing library, updating the type index.

Usage

```
pav_library_load(lib, path)
```

Arguments

lib	A pav_library object created by pav_library .
path	Character string with the path to a .pavdata file.

Value

The library lib, invisibly.

Examples

```
b <- pav_new("binder", name = "CAP 50/70", binder_type = "CAP 50/70")
tmp <- tempfile(fileext = ".pavdata")
pav_write(b, tmp)
lib <- pav_library()
pav_library_load(lib, tmp)
```

pav_list

List objects in a library

Description

Lists objects stored in a library, optionally filtered by type, by a substring of the name, or by binder type, up to a maximum number of results.

Usage

```
pav_list(
  lib,
  type = NULL,
  name_contains = NULL,
  binder_type = NULL,
  limit = 20
)
```

Arguments

lib	A pav_library object created by pav_library .
type	Character string with an object type to filter by, or NULL (default) for all types.
name_contains	Character string matched against object names (case-insensitive), or NULL (default).
binder_type	Character string with a binder type to filter by, or NULL (default).
limit	Integer; maximum number of objects to return (default 20).

Value

Invisibly, a list with the matching objects.

Examples

```
b <- pav_new("binder", name = "CAP 50/70", binder_type = "CAP 50/70")
tmp <- tempfile(fileext = ".pavdata")
pav_write(b, tmp)
lib <- pav_library()
pav_library_load(lib, tmp)
pav_list(lib, type = "binder")
```

pav_load *Read and validate a .pavdata file*

Description

Reads a .pavdata file with `pav_read` and then validates each object, warning if any fail validation.

Usage

```
pav_load(path)
```

Arguments

path Character string with the path to a .pavdata file.

Value

A named list of PavData objects, keyed by object id.

Examples

```
b <- pav_new("binder", name = "CAP 50/70", binder_type = "CAP 50/70")
tmp <- tempfile(fileext = ".pavdata")
pav_write(b, tmp)
objects <- pav_load(tmp)
```

pav_new *Create a PavData object*

Description

Constructs a PavData object of a given type, attaching the common metadata (id, name, type, version, creation timestamp, source, references, notes and history) shared by all object types.

Usage

```
pav_new(object_type, ...)
```

Arguments

object_type Character string with the object type. One of "sample", "binder", "aggregate", "mixture", "binder_test", "aggregate_test", "mixture_test" or "reference".

... Named arguments with the type-specific fields (for example, binder_type for a binder, or binder_id and aggregate_id for a mixture).

Value

A PavData object: a list with S3 classes pavdata_<type>, pavdata_object and list.

Examples

```
b <- pav_new("binder", name = "CAP 50/70", binder_type = "CAP 50/70")
m <- pav_new("mixture", name = "Mixture 1", binder_id = b$id,
            aggregate_id = "agg_001")
```

pav_read

Read a .pavdata file

Description

Reads a .pavdata file and reconstructs the PavData objects, restoring their S3 classes and indexing them by id.

Usage

```
pav_read(path)
```

Arguments

path Character string with the path to a .pavdata file.

Value

A named list of PavData objects, keyed by object id.

Examples

```
b <- pav_new("binder", name = "CAP 50/70", binder_type = "CAP 50/70")
tmp <- tempfile(fileext = ".pavdata")
pav_write(b, tmp)
objects <- pav_read(tmp)
```

pav_view *View the details of an object*

Description

Prints the metadata and the filled fields of a single object stored in a library, including its non-empty nested values.

Usage

```
pav_view(lib, id)
```

Arguments

lib A pav_library object created by [pav_library](#).
id Character string with the id of the object to view.

Value

Invisibly, the object, or NULL if it is not found.

Examples

```
b <- pav_new("binder", name = "CAP 50/70", binder_type = "CAP 50/70")
tmp <- tempfile(fileext = ".pavdata")
pav_write(b, tmp)
lib <- pav_library()
pav_library_load(lib, tmp)
pav_view(lib, b$id)
```

pav_write *Write PavData objects to a .pavdata file*

Description

Serializes one or more PavData objects to a .pavdata file (JSON content), grouping them by type. The .pavdata extension is appended when missing.

Usage

```
pav_write(x, path, pretty = TRUE)
```

Arguments

x	A single PavData object or a list of PavData objects.
path	Character string with the output file path.
pretty	Logical; if TRUE (default), the JSON is indented for readability.

Value

The output path, invisibly.

Examples

```
b <- pav_new("binder", name = "CAP 50/70", binder_type = "CAP 50/70")
tmp <- tempfile(fileext = ".pavdata")
pav_write(b, tmp)
```

pavdata_sample_path *Path to the bundled example dataset*

Description

Returns the file path to the example .pavdata dataset shipped with the package (296 Brazilian asphalt mixture records).

Usage

```
pavdata_sample_path()
```

Value

A character string with the full path to the dataset file, or an empty string if the package is not installed.

Examples

```
path <- pavdata_sample_path()
```

pavdata_type	<i>Get the type of a PavData object</i>
--------------	---

Description

Get the type of a PavData object

Usage

```
pavdata_type(x)
```

Arguments

x A PavData object created by [pav_new](#).

Value

A character string with the object type (for example, "binder"), taken from its S3 class.

Examples

```
pavdata_type(pav_new("binder", name = "CAP 50/70", binder_type = "CAP 50/70"))
```

Index

is.pavdata, 2

pav_check, 3

pav_check_integrity, 3

pav_library, 4, 4, 5, 8

pav_library_load, 4

pav_list, 5

pav_load, 6

pav_new, 3, 6, 10

pav_read, 3, 6, 7

pav_view, 8

pav_write, 8

pavdata_sample_path, 9

pavdata_type, 10