

Package: fastsas (via r-universe)

October 11, 2026

Title Fast SAS7BDAT Reader Backed by Rust

Version 0.1.0

Description High-performance reader for SAS7BDAT files. The parsing core is implemented in Rust (the 'sas7bdat' crate) and exposed to R via 'extendr'. Returns a 'tibble' with column types compatible with 'haven': Date, POSIXct, hms, character and double.

License MIT + file LICENSE

URL <https://github.com/tkragholm/sas7bdat-parser-rs>

BugReports <https://github.com/tkragholm/sas7bdat-parser-rs/issues>

Encoding UTF-8

Copyright See inst/AUTHORS for the copyright holders of the Rust crates bundled in src/rust/vendor.tar.xz.

Language en-US

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.1

SystemRequirements Cargo (Rust's package manager), rustc >= 1.89

Suggests tibble, hms, haven, vctrs, testthat (>= 3.0.0)

Config/testthat/edition 3

Config/pak/sysreqs libclang-dev

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

Date/Publication 2026-09-11 16:38:05 UTC

RemoteUrl <https://github.com/tkragholm/sas7bdat-parser-rs>

RemoteRef HEAD

RemoteSha 985de44e25887e8627fdbf7c60571288d74afcb7

RemoteSubdir crates/r-plugin

Contents

read_sas7bdat 2

read_sas7bdat	<i>Read a SAS7BDAT file</i>
---------------	-----------------------------

Description

Reads a .sas7bdat file into a data frame using the Rust parsing core. Column types follow **haven** conventions: SAS numerics become double, character columns become UTF-8 character, SAS dates become Date, datetimes become POSIXct (UTC), and times become hms.

Usage

```
read_sas7bdat(  
  path,  
  catalog = NULL,  
  categorical = FALSE,  
  io_backend = c("auto", "mmap", "buffered"),  
  threads = NULL  
)  
  
read_sas(  
  path,  
  catalog = NULL,  
  categorical = FALSE,  
  io_backend = c("auto", "mmap", "buffered"),  
  threads = NULL  
)
```

Arguments

path	Path to a .sas7bdat file.
catalog	Optional path to a .sas7bcat value-label catalog. If NULL (the default), a same-stem .sas7bcat sibling next to path is used when present.
categorical	If TRUE, plain (non-value-labelled) character columns are returned as factor instead of character. This is faster to build and uses less memory on low-cardinality columns, and speeds up downstream grouping/joining. Default FALSE (haven-style character).
io_backend	How to read the file: "auto" (default) memory-maps local files and reads network shares sequentially, "mmap" always memory-maps, "buffered" always reads sequentially. See the "Network drives" section.
threads	Decode threads. NULL (the default) lets the reader decide from the file: it uses every logical core on a file with enough rows and pages to repay the threads, and one otherwise. That is not a slower path for small files, it is a faster one – spawning twelve threads for a few thousand rows costs more than it saves. Set a number to override, to leave the machine room for other work, or to bound memory: in-flight batches scale with it. Read concurrency is capped separately by the reader and is not affected.

Details

SAS variable labels are attached as the column label attribute, with any surrounding whitespace preserved exactly as **haven** reports it. When a value-label catalog is available, labelled columns are returned as `haven_labelled` vectors (a labels named vector plus the `haven/vctrs` class).

SAS special missing values (`.A-.Z`, `._`) become `tagged_na` values on every numeric and temporal column.

Value

A tibble (or a base data.frame if the **tibble** package is not installed).

Name clash with haven

haven also exports a `read_sas()`, so whichever package is attached second masks the other and `library(fastsas); library(haven)` silently changes which one `read_sas()` means. `read_sas7bdat()` is the same function under a name nothing else claims; prefer it in scripts, or qualify the call as `fastsas::read_sas()`.

Network drives

"auto" decides by asking the operating system whether the path is remote, which it can only do on Windows — a UNC path, or a mapped drive that reports as remote. Elsewhere every path looks local and is memory-mapped.

That distinction matters because memory-mapping a file on a share turns each access into a network round-trip with no readahead. If a read from a share is unexpectedly slow, or you are on Linux or macOS with a mounted share, pass `io_backend = "buffered"`.

Text encoding

Files that declare a single-byte Western encoding are decoded as Windows-1252 rather than strict ISO-8859-1. The two agree everywhere except bytes `0x80-0x9F`, which ISO-8859-1 leaves as control characters and which real SAS-on-Windows files use for curly quotes, dashes and the euro sign. **haven** decodes those as controls, so the two packages differ on exactly those bytes.

Examples

```
## Not run:
df <- read_sas7bdat("path/to/file.sas7bdat")
df <- read_sas7bdat("data.sas7bdat", catalog = "formats.sas7bcat")
df <- read_sas7bdat("survey.sas7bdat", categorical = TRUE)

# Reading from a network share on a platform where "auto" cannot detect one:
df <- read_sas7bdat("//server/share/data.sas7bdat", io_backend = "buffered")

# Leave half the machine free:
df <- read_sas7bdat("big.sas7bdat", threads = 8)

## End(Not run)
```

Index

`read_sas (read_sas7bdat),` [2](#)

`read_sas7bdat,` [2](#)

`tagged_na,` [3](#)