

Package: roxigraph (via r-universe)

October 10, 2026

Title 'RDF' and 'SPARQL' for R using 'Oxigraph'

Version 0.1.2

Description Provides 'RDF' storage and 'SPARQL' 1.1 query capabilities by wrapping the 'Oxigraph' graph database library <https://github.com/oxigraph/oxigraph>. Supports in-memory and persistent ('RocksDB') storage, multiple 'RDF' serialization formats ('Turtle', 'N-Triples', 'RDF/XML', 'N-Quads', 'TriG'), and full 'SPARQL' 1.1 Query and Update support. Built using the 'extendr' framework for 'Rust'-R bindings.

License MIT + file LICENSE

URL <https://github.com/cboettig/roxigraph>

BugReports <https://github.com/cboettig/roxigraph/issues>

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Config/rextendr/version 0.4.2

SystemRequirements Cargo (Rust's package manager), rustc >= 1.75.0, libclang-dev (for RocksDB bindings), libsnappy-dev (for RocksDB compression)

Depends R (>= 4.2)

Suggests testthat (>= 3.0.0), knitr, rmarkdown, spelling

Config/testthat/edition 3

VignetteBuilder knitr

Language en-US

Config/pak/sysreqs libclang-dev

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

Date/Publication 2026-05-04 04:26:58 UTC

RemoteUrl <https://github.com/cboettig/roxigraph>

RemoteRef HEAD

RemoteSha c8d9a3033a3ea118f32f31d2566a648e628e0fb0

Contents

rdf_add	2
rdf_load	3
rdf_load_file	3
rdf_remove	4
rdf_serialize	5
rdf_size	5
rdf_store	6
sparql_query	6
sparql_update	7
Index	8

rdf_add	<i>Add a Triple</i>
---------	---------------------

Description

 Adds a single triple to the store.

Usage

 rdf_add(store, subject, predicate, object, graph = NULL)

Arguments

store	An RDF store handle
subject	Subject IRI (e.g., "<http://example.org/s>") or blank node ("_:b1")
predicate	Predicate IRI (e.g., "<http://example.org/p>")
object	Object: IRI, blank node, or literal (e.g., '"value"')
graph	Optional named graph IRI

Value

 Invisibly returns NULL

Examples

```
store <- rdf_store()
rdf_add(store, "<http://example.org/s>", "<http://example.org/p>", '"hello"')
rdf_size(store)
```

rdf_load	<i>Load RDF Data</i>
----------	----------------------

Description

Loads RDF data into the store from a string.

Usage

```
rdf_load(store, data, format = "turtle", base_iri = NULL)
```

Arguments

store	An RDF store handle
data	RDF data as a character string
format	RDF format: "turtle", "ntriples", "rdxml", "nquads", or "trig"
base_iri	Optional base IRI for resolving relative URIs

Value

Invisibly returns NULL

Examples

```
store <- rdf_store()  
rdf_load(store, '<http://example.org/s> <http://example.org/p> "value" .', format = "ntriples")
```

rdf_load_file	<i>Load RDF from File</i>
---------------	---------------------------

Description

Loads RDF data into the store from a file.

Usage

```
rdf_load_file(store, file, format = NULL, base_iri = NULL)
```

Arguments

store	An RDF store handle
file	Path to the RDF file
format	RDF format. If NULL, guessed from file extension.
base_iri	Optional base IRI for resolving relative URIs

Value

Invisibly returns NULL

Examples

```
store <- rdf_store()
# Create a temporary RDF file
tmp <- tempfile(fileext = ".nt")
writeLines('<http://example.org/s> <http://example.org/p> "value" .', tmp)
rdf_load_file(store, tmp)
rdf_size(store)
```

`rdf_remove`

Remove a Triple

Description

Removes a single triple from the store.

Usage

```
rdf_remove(store, subject, predicate, object, graph = NULL)
```

Arguments

<code>store</code>	An RDF store handle
<code>subject</code>	Subject IRI or blank node
<code>predicate</code>	Predicate IRI
<code>object</code>	Object: IRI, blank node, or literal
<code>graph</code>	Optional named graph IRI

Value

Invisibly returns NULL

Examples

```
store <- rdf_store()
rdf_add(store, "<http://example.org/s>", "<http://example.org/p>", '"hello"')
rdf_remove(store, "<http://example.org/s>", "<http://example.org/p>", '"hello"')
rdf_size(store)
```

rdf_serialize	<i>Serialize RDF Data</i>
---------------	---------------------------

Description

Serializes the store contents to a string.

Usage

```
rdf_serialize(store, format = "turtle")
```

Arguments

store	An RDF store handle
format	RDF format: "turtle", "ntriples", "rdxml", "nquads", or "trig"

Value

The serialized RDF data as a character string

Examples

```
store <- rdf_store()
rdf_load(store, '<http://example.org/s> <http://example.org/p> "value" .', format = "ntriples")
rdf_serialize(store, format = "turtle")
```

rdf_size	<i>Get Store Size</i>
----------	-----------------------

Description

Returns the number of quads (triples) in the store.

Usage

```
rdf_size(store)
```

Arguments

store	An RDF store handle
-------	---------------------

Value

The number of quads as an integer

Examples

```
store <- rdf_store()
rdf_size(store)
```

rdf_store	<i>Create an RDF Store</i>
-----------	----------------------------

Description

Creates a new RDF store, either in-memory or backed by persistent storage.

Usage

```
rdf_store(path = NULL)
```

Arguments

path	Optional path for persistent storage. If NULL (default), creates an in-memory store.
------	--

Value

An RDF store handle (integer)

Examples

```
# In-memory store
store <- rdf_store()

# Persistent store (not supported on Windows)

if (.Platform$OS.type != "windows") {
  store <- rdf_store(file.path(tempdir(), "my_store"))
}
```

sparql_query	<i>Execute a SPARQL Query</i>
--------------	-------------------------------

Description

Executes a SPARQL query against the RDF store.

Usage

```
sparql_query(store, query)
```

Arguments

store	An RDF store handle
query	A SPARQL query string

Value

For SELECT queries, a data.frame with results. For ASK queries, a logical. For CONSTRUCT/DESCRIBE queries, a data.frame with subject, predicate, object columns.

Examples

```
store <- rdf_store()
rdf_load(store, '<http://example.org/s> <http://example.org/p> "hello" .', format = "ntriples")
sparql_query(store, "SELECT * WHERE { ?s ?p ?o }")
```

sparql_update	<i>Execute SPARQL Update</i>
---------------	------------------------------

Description

Executes a SPARQL UPDATE query to modify the store.

Usage

```
sparql_update(store, update)
```

Arguments

store	An RDF store handle
update	A SPARQL UPDATE query string

Value

Invisibly returns NULL

Examples

```
store <- rdf_store()
sparql_update(store, "INSERT DATA { <http://example.org/s> <http://example.org/p> 'value' }")
```

Index

`rdf_add`, [2](#)
`rdf_load`, [3](#)
`rdf_load_file`, [3](#)
`rdf_remove`, [4](#)
`rdf_serialize`, [5](#)
`rdf_size`, [5](#)
`rdf_store`, [6](#)

`sparql_query`, [6](#)
`sparql_update`, [7](#)