

Package: OmopOnSpark (via r-universe)

May 24, 2026

Title Using a Common Data Model on Spark

Version 0.1.0

Description Use health data in the Observational Medical Outcomes Partnership Common Data Model format in Spark.

License Apache License (>= 2)

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Depends R (>= 4.1.0)

Imports cli, datasets, DBI, dbplyr, dplyr, glue, omopgenerics (>= 1.3.1), purrr, rlang, stringr

Suggests testthat (>= 3.0.0), omock, knitr, rmarkdown, CDMConnector, OmopSketch, odbc, R6, crayon, sparklyr, DatabaseConnector

Config/testthat/edition 3

Config/testthat/parallel false

VignetteBuilder knitr

URL <https://OHDSI.github.io/OmopOnSpark/>

Config/pak/sysreqs libicu-dev

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

Date/Publication 2025-10-13 13:44:56 UTC

RemoteUrl <https://github.com/ohdsi/omoponspark>

RemoteRef HEAD

RemoteSha b36fdbf9e03df056d2d2a973965cc3512e31e6fb

Contents

cdmDisconnect.spark_cdm	2
cdmFromSpark	2
createOmopTablesOnSpark	3

dropSourceTable.spark_cdm	4
insertTable.spark_cdm	5
mockSparkCdm	5

Index 7

cdmDisconnect.spark_cdm
Disconnect the connection of the cdm object

Description

Disconnect the connection of the cdm object

Usage

```
## S3 method for class 'spark_cdm'
cdmDisconnect(cdm, dropWriteSchema = FALSE, ...)
```

Arguments

cdm	cdm reference
dropWriteSchema	Whether to drop tables in the writeSchema
...	Not used

cdmFromSpark *Create a cdm_reference object from a sparklyr connection.*

Description

Create a cdm_reference object from a sparklyr connection.

Usage

```
cdmFromSpark(
  con,
  cdmSchema,
  writeSchema,
  cohortTables = NULL,
  cdmVersion = NULL,
  cdmName = NULL,
  achillesSchema = NULL,
  .softValidation = FALSE,
  writePrefix = NULL,
  cdmPrefix = NULL
)
```

Arguments

con	A spark connection created with: <code>sparklyr::spark_connect()</code> .
cdmSchema	Schema where omop standard tables are located. Schema is defined with a named character list/vector; allowed names are: 'catalog', 'schema' and 'prefix'.
writeSchema	Schema where with writing permissions. Schema is defined with a named character list/vector; allowed names are: 'catalog', 'schema' and 'prefix'.
cohortTables	Names of cohort tables to be read from <code>writeSchema</code> .
cdmVersion	The version of the cdm (either "5.3" or "5.4"). If NULL <code>cdm_source\$cdm_version</code> will be used instead.
cdmName	The name of the cdm object, if NULL <code>cdm_source\$cdm_source_name</code> will be used instead.
achillesSchema	Schema where achilled tables are located. Schema is defined with a named character list/vector; allowed names are: 'catalog', 'schema' and 'prefix'.
.softValidation	Whether to use soft validation, this is not recommended as analysis pipelines assume the cdm fullfill the validation criteria.
writePrefix	A prefix that will be added to all tables created in the <code>write_schema</code> . This can be used to create namespace in your database <code>write_schema</code> for your tables.
cdmPrefix	A prefix used with the OMOP CDM tables.

Value

A cdm reference object

Examples

```
## Not run:
con <- sparklyr::spark_connect(...)
cdmFromSpark(
  con = con,
  cdmSchema = c(catalog = "...", schema = "...", prefix = "..."),
  writeSchema = list() # use `list()`/`c()`/`NULL` to use temporary tables
)

## End(Not run)
```

```
createOmopTablesOnSpark
```

Create OMOP CDM tables

Description

Create OMOP CDM tables

Usage

```
createOmopTablesOnSpark(
  con,
  schemaName,
  cdmVersion = "5.4",
  overwrite = FALSE,
  bigInt = FALSE,
  cdmPrefix = NULL
)
```

Arguments

con	Connection to a Spark database.
schemaName	Schema in which to create tables.
cdmVersion	Which version of the OMOP CDM to create. Can be "5.3" or "5.4".
overwrite	Whether to overwrite existing tables.
bigInt	Whether to use big integers for person identifier (person_id or subject_id)
cdmPrefix	Whether to cdmPrefix tables created (not generally recommended).

Value

OMOP CDM tables created in database

```
dropSourceTable.spark_cdm
      Drop spark tables
```

Description

Drop Spark tables in the write schema of the connection behind the cdm reference.

Usage

```
## S3 method for class 'spark_cdm'
dropSourceTable(cdm, name)
```

Arguments

cdm	A cdm reference
name	The names of the tables to drop. Tidysselect statements can be used.

Value

Drops the Spark tables.

insertTable.spark_cdm *Insert a table to a cdm object*

Description

Insert a local dataframe into the cdm.

Usage

```
## S3 method for class 'spark_cdm'
insertTable(cdm, name, table, overwrite = TRUE, temporary = FALSE, ...)
```

Arguments

cdm	A cdm reference.
name	The name of the table to insert.
table	The table to insert.
overwrite	Whether to overwrite an existing table.
temporary	If TRUE, a spark dataframe will be written (that will persist to the end of the current session). If FALSE, a spark table will be written (which will persist beyond the end of the current session).
...	For compatability

Value

The cdm reference with the table added.

mockSparkCdm *creates a cdm reference to local spark OMOP CDM tables*

Description

creates a cdm reference to local spark OMOP CDM tables

Usage

```
mockSparkCdm(path)
```

Arguments

path	A directory for files
------	-----------------------

Value

A cdm reference with synthetic data in a local spark connection

Examples

```
## Not run:  
mockSparkCdm()  
  
## End(Not run)
```

Index

`cdmDisconnect.spark_cdm`, [2](#)
`cdmFromSpark`, [2](#)
`createOtopTablesOnSpark`, [3](#)
`dropSourceTable.spark_cdm`, [4](#)
`insertTable.spark_cdm`, [5](#)
`mockSparkCdm`, [5](#)