

# Package ‘CatalogueExport’

March 7, 2021

**Type** Package

**Title** Exports Descriptive Statistics Summary for the EHDEN Database Catalogue

**Version** 1.0.1

**Date** 2021-03-07

**Maintainer** Peter Rijnbeek <p.rijnbeek@erasmusmc.nl>

**LazyData** true

**Description** Exports descriptive statistics summary for the EHDEN Database Catalogue.

**Depends** DatabaseConnector (>= 2.0.0)

**Imports** SqlRender (>= 1.6.0),

dplyr,  
rjson,  
readr,  
ParallelLogger

**Suggests** testthat,

R.utils,  
shiny,  
DT,  
shinydashboard,  
magrittr,  
tidyr,  
knitr,  
rmarkdown

**VignetteBuilder** knitr

**License** Apache License

**Roxygen** list(wrap = FALSE)

**RoxygenNote** 7.1.1

**Encoding** UTF-8

## R topics documented:

catalogueExport . . . . .	2
createIndices . . . . .	4
dropAllScratchTables . . . . .	5
exportResultsToCSV . . . . .	5
getAnalysisDetails . . . . .	6
printAnalysesSql . . . . .	7

**Index****8**


---

catalogueExport	<i>The main CatalogueExport analyses (for v5.x)</i>
-----------------	---

---

**Description**

CatalogueExport exports a set of descriptive statistics summary from the CDM, to be uploaded in the Database Catalogue.

**Usage**

```
catalogueExport(
  connectionDetails,
  cdmDatabaseSchema,
  resultsDatabaseSchema = cdmDatabaseSchema,
  scratchDatabaseSchema = resultsDatabaseSchema,
  vocabDatabaseSchema = cdmDatabaseSchema,
  oracleTempSchema = resultsDatabaseSchema,
  sourceName = "",
  analysisIds = "",
  createTable = TRUE,
  smallCellCount = 5,
  cdmVersion = "5",
  createIndices = TRUE,
  numThreads = 1,
  tempPrefix = "tmpach",
  dropScratchTables = TRUE,
  sqlOnly = FALSE,
  outputFolder = "output",
  verboseMode = TRUE
)
```

**Arguments**

connectionDetails

An R object of type connectionDetails created using the function createConnectionDetails in the DatabaseConnector package.

cdmDatabaseSchema

Fully qualified name of database schema that contains OMOP CDM schema. On SQL Server, this should specify both the database and the schema, so for example, on SQL Server, 'cdm\_instance.dbo'.

resultsDatabaseSchema

Fully qualified name of database schema that we can write final results to. Default is cdmDatabaseSchema. On SQL Server, this should specify both the database and the schema, so for example, on SQL Server, 'cdm\_results.dbo'.

scratchDatabaseSchema

Fully qualified name of the database schema that will store all of the intermediate scratch tables, so for example, on SQL Server, 'cdm\_scratch.dbo'. Must be accessible to/from the cdmDatabaseSchema and the resultsDatabaseSchema. Default is resultsDatabaseSchema. Making this "" will run CatalogueExport in single-threaded mode and use temporary tables instead of permanent tables.

vocabDatabaseSchema	String name of database schema that contains OMOP Vocabulary. Default is cdmDatabaseSchema. On SQL Server, this should specify both the database and the schema, so for example 'results.dbo'.
oracleTempSchema	For Oracle only: the name of the database schema where you want all temporary tables to be managed. Requires create/insert permissions to this database.
sourceName	String name of the data source name. If blank, CDM_SOURCE table will be queried to try to obtain this.
analysisIds	(OPTIONAL) A vector containing the set of CatalogueExport analysisIds for which results will be generated. If not specified, all analyses will be executed. Use <a href="#">getAnalysisDetails</a> to get a list of all CatalogueExport analyses and their Ids.
createTable	If true, new results tables will be created in the results schema. If not, the tables are assumed to already exist, and analysis results will be inserted (slower on MPP).
smallCellCount	To avoid patient identifiability, cells with small counts ( $\leq$ smallCellCount) are deleted. Set to NULL if you don't want any deletions.
cdmVersion	Define the OMOP CDM version used: currently supports v5 and above. Use major release number or minor number only (e.g. 5, 5.3)
createIndices	Boolean to determine if indices should be created on the resulting CatalogueExport tables. Default= TRUE
numThreads	(OPTIONAL, multi-threaded mode) The number of threads to use to run CatalogueExport in parallel. Default is 1 thread.
tempPrefix	(OPTIONAL, multi-threaded mode) The prefix to use for the scratch CatalogueExport analyses tables. Default is "tmpach"
dropScratchTables	(OPTIONAL, multi-threaded mode) TRUE = drop the scratch tables (may take time depending on dbms), FALSE = leave them in place for later removal.
sqlOnly	Boolean to determine if CatalogueExport should be fully executed. TRUE = just generate SQL files, don't actually run, FALSE = run CatalogueExport
outputFolder	Path to store logs and SQL files
verboseMode	Boolean to determine if the console will show all execution steps. Default = TRUE

## Details

CatalogueExport exports a set of descriptive statistics summary from the CDM, to be uploaded in the Database Catalogue.

## Value

An object of type catalogueResults containing details for connecting to the database containing the results

## Examples

```
## Not run:
connectionDetails <- createConnectionDetails(dbms="sql server", server="some_server")
results <- achilles(connectionDetails = connectionDetails,
```

```

    cdmDatabaseSchema = "cdm",
    resultsDatabaseSchema="results",
    scratchDatabaseSchema="scratch",
    sourceName="Some Source",
    cdmVersion = "5.3",
    numThreads = 10,
    outputFolder = "output")

## End(Not run)

```

---

createIndices

*Create indices*


---

## Description

Create indices

## Usage

```

createIndices(
  connectionDetails,
  resultsDatabaseSchema,
  outputFolder,
  sqlOnly = FALSE,
  verboseMode = TRUE,
  catalogueTables = c("catalogue_results", "catalogue_results_dist")
)

```

## Arguments

connectionDetails

An R object of type connectionDetails created using the function createConnectionDetails in the DatabaseConnector package.

resultsDatabaseSchema

Fully qualified name of database schema that we can write final results to. Default is cdmDatabaseSchema. On SQL Server, this should specify both the database and the schema, so for example, on SQL Server, 'cdm\_results.dbo'.

outputFolder

Path to store logs and SQL files

sqlOnly

TRUE = just generate SQL files, don't actually run, FALSE = run Achilles

verboseMode

Boolean to determine if the console will show all execution steps. Default = TRUE

catalogueTables

Which CatalogueExport tables should be indexed? Default is both catalogue\_results and catalogue\_results\_dist.

## Details

Post-processing, create indices to help performance. Cannot be used with Redshift.

---

dropAllScratchTables	<i>Drop all possible scratch tables</i>
----------------------	---

---

**Description**

Drop all possible scratch tables

**Usage**

```
dropAllScratchTables(  
  connectionDetails,  
  scratchDatabaseSchema,  
  tempPrefix = "tmpach",  
  numThreads = 1,  
  tableTypes = "catalogueExport",  
  outputFolder,  
  verboseMode = TRUE  
)
```

**Arguments**

connectionDetails	An R object of type connectionDetails created using the function createConnectionDetails in the DatabaseConnector package.
scratchDatabaseSchema	string name of database schema that CatalogueExport scratch tables were written to.
tempPrefix	The prefix to use for the "temporary" (but actually permanent) CatalogueExport analyses tables. Default is "tmpach"
numThreads	The number of threads to use to run this function. Default is 1 thread.
tableTypes	The types of scratch tables to drop: catalogueExport
outputFolder	Path to store logs and SQL files
verboseMode	Boolean to determine if the console will show all execution steps. Default = TRUE

**Details**

Drop all possible CatalogueExport scratch tables

---

exportResultsToCSV	<i>The main CatalogueExport analyses (for v5.x)</i>
--------------------	---

---

**Description**

exportResults exports the results to a csv file for upload to the Catalogue

**Usage**

```
exportResultsToCSV(
  connectionDetails,
  resultsDatabaseSchema,
  analysisIds = c(),
  smallCellCount = 5,
  exportFolder = "../output"
)
```

**Arguments**

- |                       |  |
|-----------------------|--|
| connectionDetails     | An R object of type connectionDetails created using the function createConnectionDetails in the DatabaseConnector package.   |
| resultsDatabaseSchema | Fully qualified name of database schema that we can write final results to. Default is cdmDatabaseSchema. On SQL Server, this should specify both the database and the schema, so for example, on SQL Server, 'cdm_results.dbo'. |
| analysisIds           | A vector containing the set of CatalogueExport analysisIds for which results will be generated.  |
| smallCellCount        | To avoid patient identifiability, cells with small counts ( $\leq$ smallCellCount) are deleted. Set to NULL if you don't want any deletions.   |
| exportFolder          | Folder to export the results to.   |

---

getAnalysisDetails	<i>Get all analysis details</i>
--------------------	---------------------------------

---

**Description**

Get all analysis details

**Usage**

```
getAnalysisDetails()
```

**Details**

Get a list of all analyses with their analysis IDs and strata.

**Value**

A data.frame with the analysis details.

---

printAnalysesSql	<i>Print the sql of an analysis</i>
------------------	-------------------------------------

---

**Description**

Print the sql of an analysis

**Usage**

```
printAnalysesSql(analysisId, connectionDetails)
```

**Arguments**

analysisId      An analysisId for which the sql will be printed.

connectionDetails

An R object of type connectionDetails created using the function createConnectionDetails in the DatabaseConnector package.

**Details**

Print the parameterized SQL that is run for an analysisId.

**Value**

None

# Index

catalogueExport, [2](#)  
createIndices, [4](#)  
dropAllScratchTables, [5](#)  
exportResultsToCSV, [5](#)  
getAnalysisDetails, [3](#), [6](#)  
printAnalysesSql, [7](#)