

Package: dtsmartr (via r-universe)

June 17, 2026

Title Interactive Virtualized Data Explorer Grid Widget

Version 0.1.0

Description Provides an interactive, virtualized data explorer widget for 'R'. Built on 'React' (via 'reactR') and 'htmlwidgets', it offers column-type detection, multi-value checkbox filtering, sorting, column visibility toggling, virtual scrolling for large datasets, and a full-viewport modal. Includes 'dtsmartr_launch()' with an interactive, zero-code file upload wizard using 'datamods'. Widgets can be embedded in 'R Markdown' / 'Quarto' documents, 'Shiny' applications, or exported as standalone HTML files via 'save_dtsmartr()'.

License MIT + file LICENSE

URL <https://github.com/wagh-nikhil/dtsmartr>

BugReports <https://github.com/wagh-nikhil/dtsmartr/issues>

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Imports datamods, htmltools, htmlwidgets, jsonlite, reactR, shiny, tools, utils

Suggests bslib, pharmaverseadam, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

Config/pak/sysreqs cmake make libuv1-dev libssl-dev libx11-dev zlib1g-dev

Repository <https://wagh-nikhil.r-universe.dev>

Date/Publication 2026-06-09 11:21:12 UTC

RemoteUrl <https://github.com/wagh-nikhil/dtsmartr>

RemoteRef HEAD

RemoteSha 7868e8b46974f9a96814ab7e19ec7a5cc11c8671

Contents

dtsmartr	2
dtsmartr-shiny	3
dtsmartr_launch	4
dtsmartr_options	5
save_dtsmartr	6

Index	9
--------------	----------

dtsmartr	<i>dtsmartr - Interactive virtualized data explorer grid widget</i>
----------	---

Description

Renders a high-performance, virtualized data grid powered by React to explore datasets, filter dynamically, and generate reproducible queries.

Usage

```
dtsmartr(
  data,
  width = NULL,
  height = NULL,
  elementId = NULL,
  datasetName = NULL,
  options = dtsmartr_options(),
  skip_routing = FALSE
)
```

Arguments

data	A <code>data.frame</code> to explore.
width	Widget width. Defaults to "100%".
height	Widget height. Defaults to "100vh" (full screen/viewport).
elementId	CSS ID for the widget container.
datasetName	Custom string name representing the dataset in generated reproducible code. If omitted, automatically extracts the R variable name.
options	Named list of UI options generated by <code>dtsmartr_options()</code> .
skip_routing	Logical. Internal flag used by <code>save_dtsmartr()</code> to bypass the automatic re-routing to <code>dtsmartr_launch()</code> for large datasets. End users should not set this parameter. Defaults to FALSE.

Value

An object of class `htmlwidget` (and sub-class `dtsmartr`) representing the interactive virtualized grid. In interactive R sessions, this will display the explorer in the RStudio/Positron Viewer pane or system browser.

Examples

```
if (interactive()) {  
  dtsmartr(mtcars, options = dtsmartr_options(hidden_columns = "cyl"))  
}
```

dtsmartr-shiny

Shiny bindings for dtsmartr

Description

Output and render functions for using dtsmartr within Shiny applications and interactive Rmd documents.

Usage

```
dtsmartrOutput(outputId, width = "100%", height = "400px")  
  
renderDtsmartr(expr, env = parent.frame(), quoted = FALSE)
```

Arguments

outputId	output variable to read from
width, height	Must be a valid CSS unit (like '100%', '400px', 'auto') or a number, which will be coerced to a string and have 'px' appended.
expr	An expression that generates a dtsmartr
env	The environment in which to evaluate expr.
quoted	Is expr a quoted expression (with quote())? This is useful if you want to save an expression in a variable.

Value

dtsmartrOutput returns a Shiny output element of class shiny.tag.list that can be placed in a Shiny user interface. renderDtsmartr returns a Shiny render function of class shiny.render.function that can be assigned to an output slot.

dtsmartr_launch	<i>Launch the dtsmartr Data Explorer in your default Web Browser</i>
-----------------	--

Description

Launches a temporary, lightweight Shiny server in the background and opens the interactive grid in your default external web browser. This bypasses local browser origin security policies (CORS) that block file:// resources.

Usage

```
dtsmartr_launch(data = NULL, port = NULL, options = dtsmartr_options())
```

Arguments

data	A data.frame to explore, or NULL (default) to launch the Zero-Code Data Ingestion Wizard.
port	Optional port number. If NULL (default), a free port is chosen automatically.
options	Named list of UI options generated by <code>dtsmartr_options()</code> .

Details

Zero-Code Ingestion Wizard (Wizard Mode):

When `data = NULL` (default), the Shiny server boots into Wizard Mode. An interactive interface (powered by the `datamods` package) allows users to upload local files (CSV, Excel, SAS, RDS). It also features direct "View" and "Update" panels to verify and customize columns or variable classes before loading.

Once a file is successfully uploaded and confirmed, the wizard parses the dataset and feeds it directly into the React-powered virtualized grid explorer.

Value

No return value, called for the side effect of starting a local background Shiny application and opening it in the default web browser.

Examples

```
if (interactive()) {  
  # 1. Launch wizard mode to upload local files  
  dtsmartr_launch()  
  
  # 2. Launch directly with a dataset  
  dtsmartr_launch(mtcars)  
}
```

dtsmartr_options *Generate Configuration Options for dtsmartr*

Description

Helper function to specify UI customization, initial grid visibility states, and rendering options for the dtsmartr interactive grid widget.

Usage

```
dtsmartr_options(
  advanced_filter = TRUE,
  show_labels = TRUE,
  column_picker = TRUE,
  allow_export = TRUE,
  theme = "auto",
  na_string = "NA",
  hidden_columns = NULL
)
```

Arguments

advanced_filter	Logical. If TRUE (default), renders the multi-condition Advanced Query Builder panel.
show_labels	Logical. If TRUE (default), displays R column attributes (like 'label') in table headers.
column_picker	Logical. If TRUE (default), displays the top-right column show/hide visibility dropdown.
allow_export	Logical. If TRUE (default), displays clipboard copy buttons and the R & SQL "Query Code" generator modal.
theme	Character. Specifies the UI color theme: "auto" (default, inherits from browser settings), "light", or "dark".
na_string	Character. Custom string placeholder displayed in cells with missing values (NA or null). Defaults to "NA".
hidden_columns	Character vector. Vector of column names to hide by default on initial widget rendering.

Value

A named list of validated configuration settings.

Examples

```
dtsmartr_options(advanced_filter = FALSE, hidden_columns = c("STUDYID", "USUBJID"))
```

save_dtsmartr	<i>Save a dtsmartr widget to an HTML file</i>
---------------	---

Description

Exports any `data.frame` as a fully interactive, standalone HTML file powered by the `dtsmartr` widget. The resulting file can be opened in any modern web browser without an active R session or internet connection, making it ideal for offline use and sharing with collaborators.

Usage

```
save_dtsmartr(
  data,
  file,
  selfcontained = TRUE,
  title = "dtsmartr",
  open = FALSE,
  background = "white",
  libdir = NULL,
  width = NULL,
  height = NULL,
  elementId = NULL,
  options = dtsmartr_options(),
  verbose = TRUE
)
```

Arguments

<code>data</code>	A <code>data.frame</code> (or object coercible to one) to explore.
<code>file</code>	Character string. Path to the output HTML file. The <code>.html</code> extension is appended automatically if omitted.
<code>selfcontained</code>	Logical. When <code>TRUE</code> (default), all JavaScript, CSS, and data are embedded directly inside the HTML file, producing a single portable file. When <code>FALSE</code> , a companion <code><file>_files/</code> directory is created next to the HTML file containing the JS/CSS assets - both must be kept together when sharing. Use <code>FALSE</code> for large datasets where a single file would be impractically large.
<code>title</code>	Character string. Browser tab / window title for the saved HTML page. Defaults to <code>"dtsmartr"</code> .
<code>open</code>	Logical. When <code>TRUE</code> and the session is interactive, the saved HTML file is opened automatically in the default web browser immediately after saving. Defaults to <code>FALSE</code> .
<code>background</code>	Character string. CSS colour for the page background. Defaults to <code>"white"</code> .
<code>libdir</code>	Character string or <code>NULL</code> . When <code>selfcontained = FALSE</code> , the path to write the dependency libraries. Passed directly to <code>htmlwidgets::saveWidget()</code> . Defaults to <code>NULL</code> (creates <code><file>_files/</code> next to the output file).

width	Numeric or NULL. Widget width in pixels. NULL (default) uses the full page width.
height	Numeric or NULL. Widget height in pixels. NULL (default) fills the viewport (maximised mode).
elementId	Character string or NULL. CSS id for the widget's root <div>. Defaults to NULL (auto-generated).
options	Named list of UI options generated by <code>dtsmartr_options()</code> .
verbose	Logical. When TRUE, prints a confirmation message with the resolved absolute path of the saved file. Defaults to TRUE.

Details

Selfcontained vs. non-selfcontained:

selfcontained	Output	Best for
TRUE (default)	Single .html file	Email / sharing
FALSE	.html + _files/ folder	Local use / large datasets

File size:

Self-contained files embed the React runtime and widget code (~120 KB gzipped) along with the data. For very large datasets (> 100 k rows) or when generating many files, prefer `selfcontained = FALSE`.

Browser compatibility:

The generated HTML works in all modern browsers (Chrome, Firefox, Edge, Safari 14+). No internet connection is required.

Value

The absolute path to the saved HTML file, invisibly.

See Also

`dtsmartr()` for creating the widget object interactively, `htmlwidgets::saveWidget()` for the underlying save mechanism.

Examples

```
# == Basic usage =====
# Save mtcars as a self-contained HTML (single portable file)
tmp_file <- tempfile(fileext = ".html")
save_dtsmartr(mtcars, tmp_file)

# Open in the browser right after saving
tmp_file_open <- tempfile(fileext = ".html")
save_dtsmartr(mtcars, tmp_file_open, open = TRUE)
```

```
# == Custom options =====  
  
tmp_file_opts <- tempfile(fileext = ".html")  
save_dtsmartr(  
  mtcars,  
  tmp_file_opts,  
  options = dtsmartr_options(hidden_columns = "cyl", advanced_filter = FALSE)  
)
```

Index

`dtsmartr`, 2
`dtsmartr()`, 7
`dtsmartr-shiny`, 3
`dtsmartr_launch`, 4
`dtsmartr_launch()`, 2
`dtsmartr_options`, 5
`dtsmartr_options()`, 2, 4, 7
`dtsmartrOutput (dtsmartr-shiny)`, 3

`htmlwidgets::saveWidget()`, 6, 7

`renderDtsmartr (dtsmartr-shiny)`, 3

`save_dtsmartr`, 6
`save_dtsmartr()`, 2