Package: rjd3bridge (via r-universe)

October 22, 2026

Type Package

Title Bridge Between 'RJDemetra' (V2) and 'rjd3' (V3)

Version 0.1.0

Description Provides bidirectional tools to convert seasonal adjustment models and specifications between 'RJDemetra' ('JDemetra+' v2) and the 'rjd3' ecosystem ('JDemetra+' v3, packages 'rjd3x13' and 'rjd3tramoseats'). Facilitates workflow migration and backward compatibility analysis.

Depends R (>= 4.1.0)

Imports rJava, RJDemetra (>= 0.2.7), rjd3toolkit(>= 3.3.0), rjd3x13 (>= 3.3.0), rjd3tramoseats (>= 3.3.0), rjd3jars (>= 0.0.5)

URL https://github.com/AQLT/rjd3bridge, https://aqlt.github.io/rjd3bridge/

BugReports https://github.com/AQLT/rjd3bridge/issues

SystemRequirements Java (>= 17)

License EUPL

Encoding UTF-8

Config/roxygen2/version 8.1.1

NeedsCompilation no

Author Alain Quartier-la-Tente [aut, cre, cph] (ORCID: <https://orcid.org/0000-0001-7890-3857>)

Maintainer Alain Quartier-la-Tente <alain.quartier@yahoo.fr>

Config/pak/sysreqs default-jdk

Repository https://cran.r-universe.dev

Date/Publication 2026-10-22 10:00:02 UTC

RemoteUrl https://github.com/cran/rjd3bridge

RemoteRef HEAD

RemoteSha 6304e8260d5e196b70c49ceb18d89f7ea1cb93a9

Contents

as_rjd2 2
as_rjd3 4

Index **6**

`as_rjd2` *Convert 'rjd3' Objects to 'RJDemetra' (v2) Format*

Description

Converts seasonal adjustment models, outputs, or specifications from the modern 'rjd3' ecosystem ('JDemetra+' v3) back to legacy 'RJDemetra' ('JDemetra+' v2) objects for backward compatibility and legacy reporting workflows.

Usage

```
as_rjd2(x, ...)

## S3 method for class 'JD3_TRAMOSEATS_SPEC'
as_rjd2(x, context = NULL, ...)

## S3 method for class 'JD3_TRAMO_SPEC'
as_rjd2(x, context = NULL, ...)

## S3 method for class 'JD3_TRAMOSEATS_OUTPUT'
as_rjd2(
  x,
  context = NULL,
  specification = c("estimation_spec", "result_spec"),
  ...
)

## S3 method for class 'JD3_TRAMO_OUTPUT'
as_rjd2(
  x,
  context = NULL,
  specification = c("estimation_spec", "result_spec"),
  y = NULL,
  ...
)

## S3 method for class 'JD3_X13_SPEC'
as_rjd2(x, context = NULL, ...)

## S3 method for class 'JD3_REGARIMA_SPEC'
as_rjd2(x, context = NULL, ...)
```

```
## S3 method for class 'JD3_X11_SPEC'
as_rjd2(x, context = NULL, ...)

## S3 method for class 'JD3_X13_OUTPUT'
as_rjd2(
  x,
  context = NULL,
  specification = c("estimation_spec", "result_spec"),
  ...
)

## S3 method for class 'JD3_REGARIMA_OUTPUT'
as_rjd2(
  x,
  context = NULL,
  specification = c("estimation_spec", "result_spec"),
  y = NULL,
  ...
)
```

Arguments

x	An object from the 'rjd3' ecosystem (e.g., from packages `rjd3x13` or `rjd3tramoseats`).
...	Additional arguments passed to specific methods.
context	Optional `rjd3toolkit::modelling_context()` containing user-defined variables.
specification	The specification of the seasonal adjustment model to use for the conversion. Can be `"estimation_spec"` (the default, specification used to estimate the model), or `"result_spec"` (the specification used to produce the final results).
y	Input time series data. Needed for `as_rjd2.JD3_REGARIMA_OUTPUT()` and `as_rjd2.JD3_TRAMO_OUTPUT` If not provided, the default year will by default be equal to 1.

Details

The parameters `ub1` and `ub2` are not translated since it seems to have a different convention between versions.

Value

An equivalent object compatible with the 'RJDemetra' package (v2 structure).

Examples

```
spec_v3 <- rjd3x13::x13_spec()
spec_v2 <- as_rjd2(spec_v3)


# Specify the modelling context with user-defined variables
```

```
y <- rjd3toolkit::ABS$X0.2.09.10.M
regs_td <- rjd3toolkit::td(
  s = y,
  groups = c(1, 2, 0, 4, 5, 6, 3),
  contrasts = TRUE
)
colnames(regs_td) <- c("Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday")
context <- rjd3toolkit::modelling_context(variables = list(regs_td))
spec_v3 <- rjd3tramoseats::tramoseats_spec() |>
  rjd3toolkit::set_tradingdays(
    option = "UserDefined",
    uservariable = c("r.Monday", "r.Tuesday", "r.Wednesday",
                     "r.Thursday", "r.Friday", "r.Saturday"),
    test = "None"
  )
spec_v2 <- as_rjd2(spec_v3, context = context)

# For preprocessing models (RegARIMA or TRAMO),
# the input time series `y` should be provided
# to avoid defaulting the start date of the time series to year 1
model_v3 <- rjd3tramoseats::tramo(y)
as_rjd2(model_v3)
as_rjd2(model_v3, y = y)
```

`as_rjd3` *Convert 'RJDemetra' (v2) Objects to 'rjd3' (v3) Format*

Description

Converts seasonal adjustment models, outputs, or specification objects from legacy 'RJDemetra' ('JDemetra+' v2) to the modern 'rjd3' ecosystem (such as packages `rjd3x13` or `rjd3tramoseats` for 'JDemetra+' v3).

Usage

```
as_rjd3(x, new_spec = NULL, ...)

## S3 method for class 'SA_spec'
as_rjd3(x, new_spec = NULL, ...)

## S3 method for class 'regarima'
as_rjd3(x, new_spec = NULL, ...)

## S3 method for class 'SA'
as_rjd3(x, new_spec = NULL, ...)

## S3 method for class 'regarima_spec'
as_rjd3(x, new_spec = NULL, ...)
```

Arguments

x	An object from the 'RJDemetra' package (v2, e.g., of class SA, SA_spec).
new_spec	An optional specification object from the 'rjd3' ecosystem (v3) to be used as a base for conversion. If NULL, a default specification will be created based on the type of model being converted.
...	Additional arguments passed to specific methods.

Details

The parameters ub1 and ub2 are not translated since it seems to have a different convention between versions.

Value

An equivalent object compatible with the 'rjd3' ecosystem (v3 structure).

A list with two elements:

- context: An object of class rjd3toolkit::modelling_context() containing any user-defined variables.
- result: The converted object in the 'rjd3' format.

Examples

```
# Convert only a specification object
spec_v2 <- RJDemetra::regarima_spec_tramoseats()
spec_v3 <- as_rjd3(spec_v2)


# Convert a full seasonal adjustment model
y <- RJDemetra::ipi_c_eu[, "FR"]
model_v2 <- RJDemetra::x13(y, spec = "RSA5c")
model_v3 <- as_rjd3(model_v2)
```

Index

as_rjd2, 2
as_rjd3, 4

rjd3toolkit::modelling_context(), *3*, *5*