

# Package: reportbrick (via r-universe)

August 30, 2024

**Type** Package

**Title** Reporting package for BRICK

**Version** 0.6.0

**Date** 2024-08-29

**Description** This package contains BRICK-specific routines to report model results. The main functionality is to generate a mif-file from a given BRICK model run folder.

**License** LGPL-3

**URL** <https://github.com/pik-piam/reportbrick>

**Imports** dplyr, gamstransfer (>= 3.0.1), ggplot2, knitr, madrat, magclass, mip (>= 0.148.19), piamPlotComparison, piamutils (>= 0.0.9), purrr, rmarkdown, stringr, tidyr, yaml

**Suggests** covr, testthat

**Encoding** UTF-8

**RoxygenNote** 7.3.2

**Repository** <https://pik-piam.r-universe.dev>

**RemoteUrl** <https://github.com/pik-piam/reportbrick>

**RemoteRef** HEAD

**RemoteSha** 69ce84a0a8cd7b2f768125608c07cea2654da8e3

## Contents

.addCarrierDimension . . . . .	2
.agg . . . . .	3
.combinations . . . . .	3
.computeDeviation . . . . .	4
.computeFlowSum . . . . .	4
.computeRelDev . . . . .	5
.computeSumSq . . . . .	5
.constructDimMapping . . . . .	6
.expandDims . . . . .	6

.findInconsistentSetElements . . . . .	7
.findTags . . . . .	7
.missingElements . . . . .	8
.readGdxIter . . . . .	8
.replaceVarName . . . . .	9
.select . . . . .	9
.setNames . . . . .	10
.split . . . . .	10
.tag . . . . .	11
convGDX2MIF . . . . .	11
plotBRICKCalib . . . . .	12
readBrickSets . . . . .	13
readGdxSymbol . . . . .	13
reportAgg . . . . .	14
reportBuildingStock . . . . .	15
reportCalibration . . . . .	15
reportConstruction . . . . .	16
reportDemolition . . . . .	16
reportEnergy . . . . .	17
<b>Index</b>	<b>18</b>

---

*.addCarrierDimension*    *Add carrier dimension based on heating system technology*

---

### **Description**

Add carrier dimension based on heating system technology

### **Usage**

```
.addCarrierDimension(v_stock, hsCarrier)
```

### **Arguments**

v_stock	MagPIE object, BRICK variable
hsCarrier	data.frame, mapping between heating technology and energy carrier

### **Value**

MagPIE object with additional carrier dimension

---

.agg

*Aggregate*

---

**Description**

Aggregate

**Usage**

```
.agg(x, agg, silent = TRUE)
```

**Arguments**

x	MagPIE object, BRICK object
agg	named vector of dimensions to aggregate.
silent	logical, suppress warnings and printing of dimension mapping

**Value**

aggregated MagPIE objects without sub dimensions in dim 3

---

.combinations

*All Combinations of dimension elements*

---

**Description**

All Combinations of dimension elements

**Usage**

```
.combinations(lst)
```

**Arguments**

lst	names list of dimension entries
-----	---------------------------------

**Value**

character vector with all combinations of the dimension elements each separated by .

**Author(s)**

Robin Hasse

---

*.computeDeviation*      *Compute the deviation to historical data*

---

**Description**

Compute the deviation to historical data

**Usage**

```
.computeDeviation(df, dfHist)
```

**Arguments**

df                      data frame, containing data from calibration run  
dfHist                  data frame, containing historical data

**Value**

data frame where the value column contains the deviation from historical data

---

*.computeFlowSum*      *Compute the sum of construction and renovation flow values*

---

**Description**

Compute the sum of construction and renovation flow values

**Usage**

```
.computeFlowSum(con, ren)
```

**Arguments**

con                     data frame, contains construction flow quantities  
ren                     data frame, contains renovation flow quantities

**Value**

data frame with the sum in the value column

---

.computeRelDev      *Compute the relative deviation to historical data as the relative euclidean distance*

---

**Description**

Compute the relative deviation to historical data as the relative euclidean distance

**Usage**

```
.computeRelDev(dfDev, dfHist, tCalib, notInHist = NULL)
```

**Arguments**

dfDev	data frame, containing deviation between calibration and historical data
dfHist	data frame or a list of one or two data frames, containing historical data. The option to pass two data frames is used to combine construction and renovation flows. If two data frames are passed, the first one is assumed to be construction, the second renovation
tCalib	numerical/factor, calibration time periods to filter historical data
notInHist	character vector, pass columns that historical data should not be grouped by when computing the sum of the squares

**Value**

data frame with relative deviation in value column

---

.computeSumSq      *Compute the sum of the squares in a data frame*

---

**Description**

Compute the sum of the squares in a data frame

**Usage**

```
.computeSumSq(df, rpvt = "")
```

**Arguments**

df	data frame, containing the data to be evaluated
rpvt	character, column names for which the sum of the squares should be reported separately

**Value**

data frame sum of squares as value column

---

`.constructDimMapping`    *Construct dimension mapping (either aggregation or reporting) with explicit set elements for each dimension*

---

**Description**

Construct dimension mapping (either aggregation or reporting) with explicit set elements for each dimension

**Usage**

```
.constructDimMapping(agg, rpvt, brickSets, silent)
```

**Arguments**

<code>agg</code>	named vector of dimensions to aggregate
<code>rpvt</code>	named vector of dimensions to report individually
<code>brickSets</code>	named list with definition of common set elements
<code>silent</code>	logical, suppress warnings and printing of dimension mapping

**Value**

nested named list with dimension mapping

---

`.expandDims`    *Extend dimensions of a data frame by adding NA entries, add variable name*

---

**Description**

Extend dimensions of a data frame by adding NA entries, add variable name

**Usage**

```
.expandDims(df, varName, allSets)
```

**Arguments**

<code>df</code>	data frame to be extended
<code>varName</code>	character, variable name to be added
<code>allSets</code>	character, sets that need to be included as column names

**Value**

data frame

---

.findInconsistenSetElements

*Find inconsistencies in set elements between reporting template and gdx*

---

### **Description**

Find inconsistencies in set elements between reporting template and gdx

### **Usage**

.findInconsistenSetElements(brickSets, gdx)

### **Arguments**

brickSets	character, BRICK reporting template
gdx	file path to a BRICK gdx

### **Value**

named list of missing and surplus sets elements

---

.findTags

*Find all tags in name escaped in curly brackets*

---

### **Description**

Find all tags in name escaped in curly brackets

### **Usage**

.findTags(name)

### **Arguments**

name	character, variable name
------	--------------------------

### **Value**

vector of tags in name, NULL if there are none

---

<code>.missingElements</code>	<i>List any missing elements in any given dimension</i>
-------------------------------	---

---

**Description**

List any missing elements in any given dimension

**Usage**

```
.missingElements(x, dimLst)
```

**Arguments**

<code>x</code>	MagPIE object, BRICK object
<code>dimLst</code>	named vector, dimension ans names and elements as values

**Value**

vector of missing dimension elements in x

---

<code>.readGdxIter</code>	<i>Read a symbol from several.gdx files and combine in one data frame</i>
---------------------------	---

---

**Description**

Read a symbol from several.gdx files and combine in one data frame

**Usage**

```
.readGdxIter(
 .gdx,
  .symbol,
  .maxIter,
  .asMagpie = TRUE,
  .ttotFilter = NULL,
  .replaceVar = FALSE
)
```

**Arguments**

<code>gdx</code>	Path to.gdx files; all.gdx file names are assumed to be numbered and adhere to the structure: If <code>gdx = &lt;path/to/file/name.gdx&gt;</code> , the function looks for <code>&lt;path/to/file/name_0.gdx&gt;</code> , <code>&lt;path/to/file/name_1.gdx&gt;</code> , ...
<code>symbol</code>	Symbol to be read from the.gdxes
<code>maxIter</code>	Last iteration to be read



<code>asMagpie</code>	logical, convert to magpie object?
<code>ttotFilter</code>	numeric/factor, time periods to filter for
<code>replaceVar</code>	logical, replace column names bs and hs by bsr and hsr?

**Value**

data frame of all results read in

---

`.replaceVarName`      *Replace column names bs and hs by bsr and hsr*

---

**Description**

Replace column names bs and hs by bsr and hsr

**Usage**

`.replaceVarName(df)`

**Arguments**

`df`                  data frame

**Value**

data frame where column names have been replaced

---

`.select`              *select values from MAgPIE-obect*

---

**Description**

This is a wrapper around `magclass::mselect` that also allows the selection of combinations of multiple dimensions.

**Usage**

`.select(x, ...)`

**Arguments**

<code>x</code>	MAgPIE object
<code>...</code>	entry selection. Combined dimensions have to be seperated with <code>.</code> for both the set names and the set elements.

**Value**

MAgPIE object containing only selected entries

---

`.setNames`                      *Wrapper around setNames*

---

**Description**

Wrapper around setNames

**Usage**

```
.setNames(object, nm)
```

**Arguments**

<code>object</code>	MAGPIE object
<code>nm</code>	a vector of names current names should be replaced with. If only one data element exists you can also set the name to NULL.

**Value**

NULL if object is NULL else object with manipulated names

---

`.split`                              *Split dimension names*

---

**Description**

Split each entry of a character vector and return one unnested character vector.

**Usage**

```
.split(x, split = "\\.")
```

**Arguments**

<code>x</code>	character vector
<code>split</code>	character used to split x

**Value**

character vector with each dimension as an own entry

**Author(s)**

Robin Hasse

---

.tag                      *Create tag*

---

**Description**

Escape dimension name in curly brackets

**Usage**

.tag(dim)

**Arguments**

dim                      character dimension name

**Value**

character, dimension tag

---

convGDX2MIF                      *Read in GDX from BRICK and write \*.mif reporting*

---

**Description**

Read in all information from GDX file that was generated with BRICK and create the \*.mif reporting

**Usage**

```
convGDX2MIF(  
  gdx,  
  tmpl = NULL,  
  file = NULL,  
  scenario = "default",  
  t = NULL,  
  silent = TRUE  
)
```

**Arguments**

gdx                      file path to a BRICK.gdx  
tmpl                      character, BRICK reporting template. There has to be a brickSets mapping named with the same suffix: brickSets\_<tmpl>.yaml  
file                      name of the mif file which will be written, if no name is provided a magpie object containing all the reporting information is returned

scenario	scenario name that is used in the *.mif reporting
t	numeric vector of reporting periods (years)
silent	boolean, suppress warnings and printing of dimension mapping

**Author(s)**

Robin Hasse

---

plotBRICKCalib	<i>Render the BRICK calibration plotting routine</i>
----------------	--

---

**Description**

Renders the file plotCalibration.Rmd to create the plots for the BRICK calibration

**Usage**

```
plotBRICKCalib(
  path = ".",
  cal = "BRICK_calibration_report.csv",
  outName = "",
  scenNames = NULL,
  savePlots = FALSE
)
```

**Arguments**

path	(named) character vector, path(s) to output directories. If several paths are given, the names can be used to pass short scenario names.
cal	character vector, name(s) of file(s) with calibration reporting results
outName	character, string added to the pdf file name and names of additionally saved plots
scenNames	character vector, scenario names for different paths. Needs to be specified if path is unnamed and contains more than one element.
savePlots	logical, whether all plots should additionally be saved as png

**Author(s)**

Ricarda Rosemann

---

readBrickSets	<i>Read brickSets mapping</i>
---------------	-------------------------------

---

**Description**

This function creates an explicit named list with the elements of all BRICK dimensions and corresponding reporting names.

**Usage**

```
readBrickSets(tmp1 = NULL)
```

**Arguments**

tmp1	character, BRICK reporting template
------	-------------------------------------

**Value**

named list with definition of common set elements

**Author(s)**

Robin Hasse

---

readGdxSymbol	<i>Read symbol from gams container</i>
---------------	--

---

**Description**

Read symbol from gams container

**Usage**

```
readGdxSymbol(  
  gdx,  
  symbol,  
  field = "level",  
  asMagpie = TRUE,  
  stringAsFactor = TRUE  
)
```

**Arguments**

gdx	character, file path to GDX file
symbol	character, name of gams object
field	character, field to read (only relevant for variables)
asMagpie	boolean, return Magpie object
stringAsFactor	logical, keep default factors from gams

**Value**

MagPIE object with data of symbol

**Author(s)**

Robin Hasse

---

reportAgg	<i>Report aggregated quantities</i>
-----------	-------------------------------------

---

**Description**

Report aggregated quantities

**Usage**

```
reportAgg(
  x,
  name,
  brickSets = readBrickSets(NULL),
  agg = NULL,
  rpvt = NULL,
  silent = TRUE
)
```

**Arguments**

x	MagPIE object, BRICK object
name	character, name of reporting variable. reported dimensions passed with rpvt have to be escaped with curly brackets.
brickSets	named list, BRICK reporting template
agg	named vector of dimensions to aggregate. Names are dimension names of x and values are either set elements or subsets of set elements to consider.
rpvt	named vector of dimensions to report individually. Names are dimension names of x and values are either set elements or subsets of set elements to report.
silent	boolean, suppress warnings and printing of dimension mapping

**Note**

To consider specific combinations of dimensions in both agg or rprt, the combination can be given just like one dimension separated by ..

**Author(s)**

Robin Hasse

---

reportBuildingStock    *Report building Stock*

---

**Description**

Report quantities describing the stock of buildings

**Usage**

```
reportBuildingStock(gdx, brickSets = NULL, silent = TRUE)
```

**Arguments**

gdx	gams transfer container of the BRICK GDX
brickSets	character, BRICK reporting template
silent	boolean, suppress warnings and printing of dimension mapping

**Author(s)**

Robin Hasse

---

reportCalibration    *Read in model results from calibration for each iteration, calculate deviation*

---

**Description**

Read in model results from calibration for each iteration, calculate deviation

**Usage**

```
reportCalibration(gdx)
```

**Arguments**

gdx	path to a gdx; it is assumed that for each iteration a gdx is present with this path and the iteration number inserted at the end.
-----	--

**Author(s)**

Ricarda Rosemann

---

`reportConstruction`     *Report construction*

---

**Description**

Report quantities describing the construction of new buildings

**Usage**`reportConstruction(gdx, brickSets = NULL, silent = TRUE)`**Arguments**

<code>gdx</code>	gams transfer container of the BRICK GDX
<code>brickSets</code>	character, BRICK reporting template
<code>silent</code>	boolean, suppress warnings and printing of dimension mapping

**Author(s)**

Robin Hasse

---

`reportDemolition`     *Report demolition*

---

**Description**

Report quantities describing the demolition of buildings

**Usage**`reportDemolition(gdx, brickSets = NULL, silent = TRUE)`**Arguments**

<code>gdx</code>	gams transfer container of the BRICK GDX
<code>brickSets</code>	character, BRICK reporting template
<code>silent</code>	boolean, suppress warnings and printing of dimension mapping

**Author(s)**

Robin Hasse



---

reportEnergy	<i>Report energy demand</i>
--------------	-----------------------------

---

**Description**

Report final and useful energy demand for space heating

**Usage**

```
reportEnergy(gdx, brickSets = NULL, silent = TRUE)
```

**Arguments**

gdx	gams transfer container of the BRICK GDX
brickSets	character, BRICK reporting template
silent	boolean, suppress warnings and printing of dimension mapping

**Author(s)**

Robin Hasse

# Index

.addCarrierDimension, 2  
.agg, 3  
.combinations, 3  
.computeDeviation, 4  
.computeFlowSum, 4  
.computeRelDev, 5  
.computeSumSq, 5  
.constructDimMapping, 6  
.expandDims, 6  
.findInconsistenSetElements, 7  
.findTags, 7  
.missingElements, 8  
.readGdxIter, 8  
.replaceVarName, 9  
.select, 9  
.setNames, 10  
.split, 10  
.tag, 11

convGDX2MIF, 11

plotBRICKCalib, 12

readBrickSets, 13  
readGdxSymbol, 13  
reportAgg, 14  
reportBuildingStock, 15  
reportCalibration, 15  
reportConstruction, 16  
reportDemolition, 16  
reportEnergy, 17