

Package: mredgebuidings (via r-universe)

October 1, 2026

Type Package

Title Prepare data to be used by the EDGE-Buildings model

Version 0.15.10

Date 2026-07-01

Description Prepare data to be used by the EDGE-Buildings model.

License MIT + file LICENSE

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

Depends madrat (>= 2.8.1), magclass (>= 6.11.0), mrdriers (>= 4.0.7),
mrenergy (>= 0.1.1), mrcommonsenergy (>= 0.3.1)

Imports brick (>= 0.8.0), climbed (>= 1.1.3), data.table, dplyr,
GDPuc, ipfp, lme4, minpack.lm, openxlsx, purrr, rlang,
quadprog, quitte, readxl, stringr, tibble, tidyr, zoo

Suggests covr, lrcode2, piamutils, pdftools, testthat

Encoding UTF-8

Roxygen list(markdown = TRUE)

Config/roxygen2/version 8.0.0

Config/pak/sysreqs

cmake libfontconfig1-dev libfreetype6-dev libfribidi-dev libgdal-dev gdal-bin libgeos-
dev git libglpk-dev make libharfbuzz-dev libhdf5-dev libgit2-dev libicu-dev libpng-dev libuv1-
dev libxml2-dev libnetcdf-dev libssl-dev pari-gp libproj-dev libsqlite3-dev libx11-dev zlib1g-dev

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

Date/Publication 2026-07-01 14:31:04 UTC

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

RemoteRef HEAD

RemoteSha 889dd9bc6bc47371563303260d36fc6c10ab5a5e

Contents

mredgebldings-package	5
.disaggregate	5
.enforceShares	6
.fillTimeseries	7
aggToLevel	7
calcACOwnershipRates	8
calcACOwnershipRatesFilled	8
calcACOwnershipRegression	9
calcBRG	10
calcBuildingStock	10
calcCarrierPrices	11
calcConstruction	11
calcCostConstruction	12
calcCostDemolition	12
calcCostRenovationBS	13
calcCostRenovationHS	13
calcDensity	14
calcDiscountRate	14
calcEfficiencyRegression	15
calcElecDemandICT	15
calcFEbyEUEC	16
calcFEUEefficiencies	17
calcFloorspacePast	18
calcFloorspacePerCap	18
calcHDDCDD	19
calcHeatingCapacity	20
calcHeatingSystem	20
calcHeatingSystemReplacement	21
calcHeatingSystemSales	21
calcHouseholdSize	22
calcIEA_EEI	22
calcIEAPFU	23
calcLifetimeParams	24
calcLTRS	24
calcMatchingReference	25
calcOdysseeStock	26
calcPFUDB	26
calcPopulationBuildings	27
calcRenDepth	27
calcRenovation	28
calcRenovationCostModel	28
calcShareChina	29
calcShareETP	29
calcShareIEA_EEI	30
calcShareOdyssee	30
calcSharesBuildingDemand	31

calcSharesEU	32
calcShareTCEP	33
calcShareWEO	33
calcStatusQuoShare	34
calcSurface	34
calcTCEP	35
calcUEdemand	35
calcUValue	36
calcUValueParameters	36
calcWeightFeBuildings	37
convertAssotermica	38
convertBDH	38
convertBMK	39
convertCensusHub	39
convertDaioglou	40
convertDeetman2020	40
convertDestatis	41
convertEEAfloorspace	41
convertEHI	42
convertEHPA	42
convertEnergiforsk2016	43
convertEUBuildingsDB	43
convertEurObserver	44
convertEuropeanCommissionRenovation	44
convertEurostatBuildings	45
convertFEGECA	45
convertGDL	46
convertHDDCDDtemp	46
convertHotmaps	47
convertHouseholdSurveys	48
convertIEA_OECD_Prices	48
convertLTRS	49
convertMessageIXBuildings	49
convertOdyssee	50
convertPFUDB	50
convertPRISMA_ICT	51
convertPulsen	51
convertSandberg	52
convertSPIUG	52
convertTCEP	53
convertUNHouseholds	53
convertUniclima	54
convertVHK	54
convertWEO	55
dot2Dash	55
eur2usd	56
extrapolateMissingPeriods	56
fullBRICK	57

fullEDGEBUILDINGS	57
join_all	58
readAssotermica	59
readBDH	59
readBEAM2	60
readBMK	60
readCensusHub	61
readDaiglou	61
readDeetman2020	62
readDestatis	63
readECEMF	63
readEEAfloorspace	64
readEHI	64
readEHPA	65
readEnergiforsk2016	65
readEUBuildingsDB	66
readEurObserver	66
readEuropeanCommissionRenovation	67
readEurostatBuildings	68
readFEGECA	69
readGDL	70
readHDDCDD	71
readHDDCDDtemp	71
readHotmaps	72
readHouseholdSurveys	72
readIEA_acOwnership	73
readIEA_China	73
readIEA_coolingEfficiencies	73
readIEA_OECD_Prices	74
readLTRS	74
readMessageIXBuildings	75
readOdyssee	75
readPFUDB	76
readPRISMA_ICT	76
readPulsen	77
readSandberg	77
readSPIUG	78
readTCEP	78
readUNHouseholds	79
readUniclima	80
readVHK	80
readWEO	81
toolAddThermal	81
toolAggregateBrick	82
toolCalcShares	82
toolCompleteRenFlows	83
toolCountryFillAvg	83
toolDisaggregate	84

<i>mredgebuidings-package</i>	5
toolGetBrickMapping	85
toolUnitConversion	85
Index	87

mredgebuidings-package	
	<i>mredgebuidings: Prepare data to be used by the EDGE-Buildings model</i>

Description

Prepare data to be used by the EDGE-Buildings model.

Author(s)

Maintainer: Robin Hasse <robin.hasse@pik-potsdam.de> ([ORCID](#))

Authors:

- Robin Hasse <robin.hasse@pik-potsdam.de> ([ORCID](#))
- Pascal Sauer <pascal.sauer@pik-potsdam.de>
- Antoine Levesque
- Hagen Tockhorn

See Also

Useful links:

- <https://github.com/pik-piam/mredgebuidings>

<i>.disaggregate</i>	<i>Disaggregate energy demand within an aggregated region</i>
----------------------	---

Description

Disaggregate regional energy demand per carrier by end use while meeting the end use shares in the aggregated region.

Usage

`.disaggregate(subset, key)`

Arguments

subset	data frame for one period and aggregated region
key	named vector with specification of the subset group (not used)

Details

The function first tries to find a solution that satisfies both the regional carrier and the overall end use constraints. If there is no solution, another optimisation is run that tries to also minimise deviations from the end use quantities but removes them from the constraints.

.enforceShares

Enforce Shares into Energy Demand Estimates

Description

Adjusts energy demand estimates to meet specified carrier share targets within end-uses while preserving total carrier demands. This function applies proportional adjustments to align existing shares with target shares while maintaining energy balance constraints. It handles missing combinations, zero values, and special cases where target shares exist for carriers with no historical demand. The adjustments are applied consistently across all time periods with careful rescaling to ensure carrier totals remain unchanged.

Usage

```
.enforceShares(estimate, forceShares)
```

Arguments

estimate	data frame with energy carrier demand estimates
forceShares	data frame with shares to enforce for carrier-enduse combinations

Details

Note: This function requires forceShares data to contain only one period per region-enduse-carrier combination to work correctly.

Value

data frame with adjusted estimates preserving carrier totals

Author(s)

Hagen Tockhorn

<code>.fillTimeseries</code>	<i>Fill missing values in time series data</i>
------------------------------	--

Description

Fills missing values in a time series dataset using interpolation for gaps between known values and growth-based extrapolation for extending the series. The function uses linear interpolation between known values and calculates appropriate growth rates for extrapolation based on available data points. For regions with 3+ data points, it fits a log-linear model, while for regions with 2 data points, it uses a simple growth rate calculation. For regions within the `extGrowthFactors` data frame, the growth rates are used.

Usage

```
.fillTimeseries(df, extGrowthFactors = NULL, periods = 1990:2025)
```

Arguments

<code>df</code>	A data frame containing region, enduse, period, and value columns
<code>extGrowthFactors</code>	A data frame containing external growth factors
<code>periods</code>	Vector of time periods to include in the result (default: 1990:2025)

Value

A complete data frame with filled values for all region-enduse-period combinations

<code>aggToLevel</code>	<i>Aggregate variables with respect to a specified level</i>
-------------------------	--

Description

This function aggregates selected variables at a given categorical level and replaces them with a new summarized entry in the dataset.

Usage

```
aggToLevel(df, variables, level, newname)
```

Arguments

<code>df</code>	A data frame containing the data to be aggregated.
<code>variables</code>	A character vector of variable names that should be aggregated.
<code>level</code>	A string specifying the column in <code>df</code> that defines the level at which aggregation is performed.
<code>newname</code>	A string specifying the name for the newly aggregated category.

Value

A data frame where the selected variables have been aggregated into a single variable with the specified newname.

calcACOwnershipRates *Calculate Air Conditioning Ownership Rates*

Description

Calculates air conditioning ownership rates per household by combining data from household surveys, IEA, and Odyssee. Data sources are prioritized in the following order: Odyssee > IEA > household surveys. Returns all available historical data points without period harmonization.

Usage

```
calcACOwnershipRates()
```

Value

magpie object with all available historical data points per region

Author(s)

Hagen Tockhorn

calcACOwnershipRatesFilled
 *Calculate Extrapolated Air Conditioning Ownership Rates with Re-
 gional Calibration*

Description

This function fills AC ownership rates for all regions from 1990 to endOfHistory using global regression parameters (alpha, beta, gamma, delta) with region-specific GDP shifts. The GDP shift (gdppopShift) is calculated for each region to horizontally shift the global S-curve along the GDP axis so that it passes through the region's most recent historical data point. This calibrates the global curve to match available historical data while maintaining the same functional form across all regions.

Usage

```
calcACOwnershipRatesFilled(endOfHistory = 2025)
```

Arguments

endOfHistory last period of historical data (default: 2025)

Details

The logistic formula used is:

$$penetration = 1 / (1 + \exp(\alpha - \beta \cdot (gdppop - gdppopShift)^\delta \cdot CDD^\gamma))$$

Regions without historical data use the global curve as-is ($gdppopShift = 0$).

Value

magpie object with filled AC ownership rates for all regions (1990-endOfHistory)

Author(s)

Hagen Tockhorn

calcACOwnershipRegression

Calculate Air Conditioner Ownership Regression Parameters

Description

This function estimates global regression parameters for air conditioner (AC) ownership rates using a logistic function with GDP per capita and cooling degree days (CDD) as predictors. The function fits a non-linear regression model to derive parameters for the AC penetration logistic curve.

Usage

```
calcACOwnershipRegression()
```

Details

The regression model follows the logistic formula:

$$penetration = 1 / (1 + \exp(\alpha - \beta \cdot gdppop^\delta \cdot CDD^\gamma))$$

where:

- α is the intercept parameter that sets the lower boundary of AC ownership
- β is the interaction coefficient (negated from fitted model coefficient b)
- γ is the CDD exponent parameter
- δ is the GDP per capita exponent parameter

The function performs a two-stage estimation process:

1. Alpha is set exogenously based on a minimum ownership rate assumption
2. Linear regression on log-transformed data with fixed alpha to obtain starting values for nls()
3. Non-linear regression using nls() with alpha fixed, estimating only beta, gamma, and delta

Value

magpie object with global regression parameters (alpha, beta, gamma, delta)

Author(s)

Hagen Tockhorn

calcBRG	<i>Calculate total boiler sales from BRG</i>
---------	--

Description

BRG has extremely comprehensive (and extremely expensive) sales data. But one can recover total sales of boilers for space heating from the EHPA report.

Usage

calcBRG()

Value

MagPIE object with total boiler sales in European countries

Author(s)

Robin Hasse

calcBuildingStock	<i>calculate the European historic building stock</i>
-------------------	---

Description

Join data from various sources to construct a consistent European building stock with a focus on floor space (million m2).

Usage

calcBuildingStock(subtype = c("residential", "commercial"), granularity = NULL)

Arguments

- | | |
|-------------|--------------------------------------|
| subtype | Character with subsector |
| granularity | character, name of BRICK granularity |

Value

MAGPIE object with historic building stock

Note

if the distribution of building types is missing, it is filled with the Europe-wide distribution. It might be worthwhile to map filling countries instead.

Author(s)

Robin Hasse

calcCarrierPrices	<i>Calculate carrier prices</i>
-------------------	---------------------------------

Description

Final energy carrier prices including energy supply cost, transport and distribution cost(T&D) and taxes (without VAT). CO2 price component and VAT are added later depending on the scenario assumptions.

Usage

```
calcCarrierPrices()
```

Author(s)

Robin Hasse

calcConstruction	<i>Calculate historic construction flows</i>
------------------	--

Description

Calculate historic construction flows

Usage

```
calcConstruction()
```

Author(s)

Robin Hasse

calcCostConstruction	<i>calculate construction cost</i>
----------------------	------------------------------------

Description

floor-space specific construction cost

Usage

```
calcCostConstruction(granularity = NULL)
```

Arguments

granularity	character, name of BRICK granularity
-------------	--------------------------------------

Author(s)

Robin Hasse

calcCostDemolition	<i>Demolition cost</i>
--------------------	------------------------

Description

Demolition cost

Usage

```
calcCostDemolition()
```

Author(s)

Robin Hasse

calcCostRenovationBS	<i>calculate renovation cost</i>
----------------------	----------------------------------

Description

floor-space specific renovation cost

Usage

```
calcCostRenovationBS(granularity = NULL)
```

Arguments

granularity	character, name of BRICK granularity
-------------	--------------------------------------

Value

MagPIE object with floor-space specific renovation cost depending on the initial and final state of the building

Author(s)

Robin Hasse

calcCostRenovationHS	<i>calculate renovation cost</i>
----------------------	----------------------------------

Description

floor-space specific renovation cost

Usage

```
calcCostRenovationHS(energyLadder = FALSE, granularity = NULL)
```

Arguments

energyLadder	logical, should the calculation include renovation transitions that are considered a decline on the energy ladder?
granularity	character, name of BRICK granularity

Value

MagPIE object with floor-space specific renovation cost depending on the initial and final state of the building

Author(s)

Robin Hasse

calcDensity	<i>Get Population Density</i>
-------------	-------------------------------

Description

Get Population Density

Usage

```
calcDensity(endOfHistory = 2025)
```

Arguments

endOfHistory upper temporal boundary for historical data

Value

magpie object

Author(s)

Hagen Tockhorn

calcDiscountRate	<i>Discount rate</i>
------------------	----------------------

Description

Discount rate including implicit discounting

Usage

```
calcDiscountRate(granularity = NULL)
```

Arguments

granularity character, name of BRICK granularity

Author(s)

Robin Hasse

Source

<https://doi.org/10.1016/j.eneco.2011.07.010>

`calcEfficiencyRegression`*Calculation Regression Parameters for FE-UE Efficiency Projection*

Description

Calculate the regression parameters for the projection of final (FE) to useful (UE) energy conversion projection for all combinations of energy enduses and carriers. The regression parameters correspond to an asymptotic regression model and encompass the parameters Asym, R0 and lrc. For electric space cooling, a bounded logistic function is used with parameters k and x0. All parameters are determined using a nonlinear least-squares regression.

Usage

```
calcEfficiencyRegression(gasBioEquality = TRUE)
```

Arguments

`gasBioEquality` Determines if natural gas and modern biomass share the same efficiencies

Details

This approach closely follows the model by De Stercke et al. which is mainly driven by GDP per Capita.

Author(s)

Hagen Tockhorn

References

De Stercke, S. (2014). Dynamics of Energy Systems: A Useful Perspective (Interim Report, p. 68). IIASA. <http://pure.iiasa.ac.at/id/eprint/11254/1/IR-14-013.pdf>

`calcElecDemandICT`*Calculate Electricity Demand for ICT from Data Center Demands*

Description

This function processes historical and scenario electricity demand data for Information and Communication Technology (ICT) and data centers (DC). The initial data comes from exogenous projections using the Digital Transformation Level (DGI) as the primary driver of demand, with IEA data serving as the historical baseline.

Usage

```
calcElecDemandICT(endOfHistory = 2025)
```

Arguments

endOfHistory upper temporal boundary for historical data

Details

The function processes multiple demand estimates:

- Mean and medium estimate
- Lower demand boundary (DGI elasticity from historical efficiency dominant phases)
- Upper demand boundary (DGI elasticity from historical service dominant phases)

Data is downscaled from R5 to country-level using data center counts as weights, with exceptional fixed shares for China within Asia Pacific (73.28%) and for Europe's R12 sub-regions (WEU: 75%, EEU: 14%, FSU: 11%). Network demand is added using global network/DC ratios. Historical data (IEA Base) is separated and harmonized with SSP scenario projections.

Value

A magpie object containing ICT electricity demand data calculated from DC demands across regions, time periods, and scenarios.

Author(s)

Hagen Tockhorn

calcFEbyEUEC

Historic Final Energy Demand by Carrier and Enduse

Description

Takes the historic final energy demand by carriers from IEA and disaggregates it into different end uses.

Usage

calcFEbyEUEC()

Details

The disaggregation is performed such that the aggregated carrier-specific FE values of the IEA IO regions are met as a minimum requirement. In an ideal case, the enduse-specific FE shares are met as well.

Value

data.frame with historic energy demands

Note

For now, existing disaggregated final energy data with respect to carriers and enduses combined is replaced in the final output. However, since the Odyssee data is largely underestimating the real IEA FE targets, these shall serve as lower boundaries for the disaggregation. Such feature has been implemented in toolDisaggregate but is not running smoothly yet.

Author(s)

Hagen Tockhorn, Robin Hasse

calcFEUEfficiencies *Calculation and Projection of Final to Useful Energy*

Description

Calculate Efficiencies of Final (FE) to Useful (UE) Energy Conversion for all combinations of Energy Carriers and Enduses. Missing data points within the time interval of the original data (PFUDB) are filled with estimates from a non-linear regression w.r.t. GDP per capita. The efficiency projections are based on a model by De Stercke et al. which is mainly driven by GDP per Capita. It describes an S-shaped curve approaching assumed efficiency levels. The parameters of that curve are derived by a regression with observations of IEA data.

Usage

```
calcFEUEfficiencies(gasBioEquality = TRUE)
```

Arguments

gasBioEquality Determines if natural gas and modern biomass share the same efficiencies

Author(s)

Hagen Tockhorn

References

De Stercke, S. (2014). Dynamics of Energy Systems: A Useful Perspective (Interim Report, p. 68). IIASA. <http://pure.iiasa.ac.at/id/eprint/11254/1/IR-14-013.pdf>

calcFloorspacePast	<i>Historical residential floor space demand</i>
--------------------	--

Description

Data for many European countries is taken from EEA, India from IEA and other countries from Daioglou et al 2012. The result does not cover all countries and has mixed points in time depending on the region.

Usage

```
calcFloorspacePast(endOfHistory = 2025)
```

Arguments

endOfHistory	upper temporal boundary for historical data
--------------	---

Value

MAGPIE object with historic floor space

Note

RH: In Antoine's EDGE-B, data points associated with an GDP/POP above 70000 USD/cap are dropped here to improve the later regression. This filtering should be moved to getFloorspaceResidential where the regression is performed. Therefore, the high-income data points are kept at this stage.

Author(s)

Robin Hasse, Antoine Levesque, Hagen Tockhorn

calcFloorspacePerCap	<i>Residential floor space per capita</i>
----------------------	---

Description

Residential floor space per capita

Usage

```
calcFloorspacePerCap(scenarios, granularity = NULL)
```

Arguments

scenarios	A string (or vector of strings) designating the scenario(s) to be returned.
granularity	character, name of BRICK granularity

Value

MagPIE object with residential floor space per capita

Author(s)

Robin Hasse

calcHDDCDD

Calculate Heating and Cooling Degree Days (HDD/CDD)

Description

Computes or loads annual heating and cooling degree days (HDDs and CDDs) for given socio-economic scenarios.

Usage

```
calcHDDCDD(scenario = "SSPs", fromSource = TRUE, endOfHistory = 2025)
```

Arguments

scenario	Character vector specifying the socio-economic scenario(s), e.g., "ssp2".
fromSource	Logical. If TRUE, loads HDD/CDD data from source. If FALSE, computes them using the CLIMBED package.
endOfHistory	Integer. Upper temporal boundary for historical data. If it deviates from 2025, fromSource needs to be set to FALSE.

Value

A magclass object with annual degree-day values.

Note

For fromSource = FALSE, the job requires a runtime of >24h and memory > 100G.

Author(s)

Hagen Tockhorn

calcHeatingCapacity *Floor-area specific heating capacity*

Description

Rough assumption based on online reference values scaled with U-values from Hotmaps and heating degree days (HDD).

Usage

```
calcHeatingCapacity(swissFormular = FALSE)
```

Arguments

swissFormular boolean, apply the simplistic swiss formular?

Details

This is a very rough estimation that neglects regional differences and many other effects. It should be replaced by bottom-up calculation based on climate data.

Value

MagPIE object with floor-area specific heating capacity

Author(s)

Robin Hasse

Source

<https://www.heizung.de/ratgeber/diverses/heizleistung-berechnen-gruende-und-ablauf.html>

calcHeatingSystem *Calculate installation cost or efficiency of heating systems*

Description

The data is taken from the EU reference scenario technology assumptions. We map our technologies to the higher resolved PRIMES technologies. For missing commercial technologies, we assume residential values but decrease purchasing cost by 10%.

Usage

```
calcHeatingSystem(
  subtype = c("Purchasing cost", "Efficiency"),
  granularity = NULL
)
```

Arguments

subtype	character, variable type (either 'Purchasing cost' or 'Efficiency')
granularity	character, name of BRICK granularity

Value

MagPIE object with capacity-specific purchasing cost or efficiency of heating systems

Author(s)

Robin Hasse

calcHeatingSystemReplacement

Calculate Matrix of heating system replacement shares

Description

Calculate Matrix of heating system replacement shares

Usage

```
calcHeatingSystemReplacement()
```

Value

MagPIE object with shares

Author(s)

Robin Hasse

calcHeatingSystemSales

Calculate yearly sales of heating systems

Description

We use national data from industry association for the biggest European countries and European data sets from the EHI and EHPA.

Usage

```
calcHeatingSystemSales()
```

Value

MagPIE object with heating system sales

Author(s)

Robin Hasse

calcHouseholdSize	<i>calculate household size</i>
-------------------	---------------------------------

Description

Number of persons per household.

Usage

```
calcHouseholdSize()
```

Value

MAGPIE object with household size

Note

THE EU Buildings Database gives one share of all dwellings with 5 or more persons. This share is weighted with 5 persons which can cause a downward bias.

Author(s)

Robin Hasse

calcIEA_EEI	<i>Process data from IEA End Uses and Efficiency Indicators Database</i>
-------------	--

Description

IEA EEI final energy data is processed and mapped w.r.t. carrier and enduse names.

Usage

```
calcIEA_EEI(subtype = c("buildings"), mixData = FALSE)
```

Arguments

subtype	sector name
mixData	logical indicating whether to mix res/com data points if both are not given

Details

As for the buildings sector, data for residential and commercial ("service") buildings is aggregated and the carrier "biomass" is split into traditional and modern biomass w.r.t. to income per capita. Energy data on "space_cooling" is aggregated over all carriers into "elec".

Value

data.frame containing enduse- and carrier-resoluted energy data

Author(s)

Hagen Tockhorn

calcIEAPFU

IEA database of primary, final and useful energy

Description

Computes data for the Primary Final Useful (PFU) resolution

Usage

```
calcIEAPFU()
```

Value

IEA data as MAGPIE object aggregated to country level

Author(s)

Antoine Levesque, Robin Hasse

See Also

[calcOutput](#)

Examples

```
## Not run: a <- calcOutput("IEAPFU")
```

calcLifetimeParams	<i>Weibull lifetime distribution parameters</i>
--------------------	---

Description

Parameters for the lifetime of heating systems are taken from a very detailed EIA publication for building sector appliances and equipment. The range of the building shell lifetime is taken from Skarning et al. 2017.

Usage

```
calcLifetimeParams(subtype, granularity = NULL)
```

Arguments

subtype	character, type of asset (either 'building', 'heatingSystem' or 'buildingShell')
granularity	character, name of BRICK granularity

Value

MagPIE object with Weibull lifetime distribution parameters

Author(s)

Robin Hasse

Source

<https://www.eia.gov/analysis/studies/buildings/equipcosts/pdf/full.pdf>
<http://dx.doi.org/10.1016/j.enbuild.2017.01.080>

calcLTRS	<i>Calculate data based on Long Term Renovation Strategies</i>
----------	--

Description

Extrapolate the residential heating system distribution in each vintage from DEU to other countries and to commercial buildings.

Usage

```
calcLTRS(refGranularity = c("typ", "sec"))
```

Arguments

refGranularity character, for "typ", you get the most granular data with residential buildings split by SFH and MFH. However, this data is less reliable and incomplete. "sec" will aggregate residential buildings to Res. It should be complete for EU27.

Details

We use iterative proportional fitting to extrapolate the German residential pattern to other countries and subsectors. This data is extremely rough and should be taken with care. It is helpful in the matching nonetheless to represent interactions between vintage and heating systems while other more reliable references assure overall consistency.

Value

MagPIE object

Author(s)

Robin Hasse

calcMatchingReference *Calculate Reference data for input matching*

Description

Calculate Reference data for input matching

Usage

```
calcMatchingReference(subtype)
```

Arguments

subtype character, matching reference

Author(s)

Robin Hasse

calcOdysseeStock	<i>Calculate Odyssee building stock variables</i>
------------------	---

Description

Residential stock variables are given in number of dwellings, flats or houses while commercial stock is given in Floor space. This function computes all stock variables in terms of floor space considering dwelling size.

Usage

```
calcOdysseeStock(interpolate = FALSE)
```

Arguments

interpolate	logical, if TRUE, floor space growth rates from JRC_IDEES are used to inter- and extrapolate the data
-------------	---

Details

The data can be inter- and extrapolated using IDEES floor space data. As we trust the absolute level less than Odyssee, we only consider the growth to fill values.

Value

MagPIE object with building stock data

Author(s)

Robin Hasse

calcPFUDB	<i>Converts Data from Primary Fuel and Useful Energy Data Base (PFUDB)</i>
-----------	--

Description

PFU Data is disaggregated into different thermal and non-thermal Enduses using FE Shares from Odyssee and IEA_ETP.

Usage

```
calcPFUDB()
```

Details

This was adapted from EDGE function 'getPFUDB.R'.

Value

magpie object

Author(s)

Hagen Tockhorn, Robin Hasse

calcPopulationBuildings

Historic and projected population

Description

Population differentiated by location (urban/rural) and building type

Usage

```
calcPopulationBuildings(granularity = NULL)
```

Arguments

granularity character, name of BRICK granularity

Value

MagPIE object with population at BRICK granularity

Author(s)

Robin Hasse

calcRenDepth

Calculate renovation depth

Description

The renovation depth is currently calculated based on the ratio of specific useful energy demand after and before the renovation. This is valid under the assumption that there is no change in behaviour (rebound effect) which we currently make. Once adaptive behaviour is considered, we should use the ratio of U-values.

Usage

```
calcRenDepth()
```

Author(s)

Robin Hasse

calcRenovation	<i>Calculate historic renovation flows</i>
----------------	--

Description

Combine data on heating system sales and replacement to derive full building state transition data

Usage

```
calcRenovation()
```

Value

MagPIE object with historic renovation flows

Author(s)

Robin Hasse

calcRenovationCostModel	<i>Calculate linear renovation cost model</i>
-------------------------	---

Description

Specific renovation cost are estimated with a linear model with intercept w.r.t. to the renovation depth.

Usage

```
calcRenovationCostModel()
```

Details

The specific investment cost for renovation depending on the depth of renovation is calculated using a two-level statistical model fitted with data from a European Commission (EC) report on renovation. In a first step, we fit a simple linear model with intercept to predict specific investment (USD/m²) with relative PE savings. In a second step, this linear model is scaled with one factor for each region. This scaling factor is predicted with GDP/POP using a negative exponential curve. For EU regions that are reported, we correct the result with a region-specific but time-invariant factor to best match the linear model to the data. This way, we get a linear model with intercept for each region that can be extrapolated to other regions (using GDP/POP) which still matches data of EU regions very well. The cost is finally disaggregated across residential building types (SFH, MFH) based on a rough cost factor seen in the ENTRANZE data.

Value

MAgPIE object with specific renovation cost

Author(s)

Robin Hasse

Source

<https://op.europa.eu/s/xnYt>

calcShareChina

Calculate shares from final energy end-use demand for China

Description

The data is obtained from the 2021 IEA report "An Energy Sector Roadmap to Carbon Neutrality in China" from Figure 3.25

Usage

calcShareChina()

Author(s)

Robin Hasse

calcShareETP

Share of energy carriers or end uses in historic buildings demand

Description

Shares are calculated based on energy demands in households and services from the IEA ETP data base. Missing shares that result from missing demand data are filled with the average share across all regions and periods and then normalised to sum up to one again. Biomass is split according to GDP per Capita (see toolSplitBiomass).

Usage

calcShareETP(subtype = c("enduse", "carrier"), feOnly = FALSE)

Arguments

subtype	specifies share
feOnly	specifies if shares or quantities are returned

Value

MAGPIE object with historic shares

Author(s)

Robin Hasse, Antoine Levesque, Hagen Tockhorn

calcShareIEA_EEI	<i>Calculate shares of carriers or enduses in buildings energy demand from IEA EEI data</i>
------------------	---

Description

This function calculates the proportion of energy demand by carrier or enduse in the buildings sector, based on IEA Energy Efficiency Indicators data.

Usage

```
calcShareIEA_EEI(subtype = c("enduse", "carrier"))
```

Arguments

subtype	Character, specifies which type of share to calculate. Must be either "enduse" or "carrier". Default is "enduse".
---------	---

Value

A data.frame with the respective shares

Author(s)

Hagen Tockhorn

calcShareOdyssee	<i>Share of energy carriers or end uses in historic buildings demand</i>
------------------	--

Description

Shares are calculated based on energy demands in households and services from the Odyssee data base (EU member states). In case of multi-level shares, the function gives the share w.r.t. to the last categories but for all categories. E.g. 'enduse_carrier' gives the share of each carrier in the demand from each end use. Missing shares are handled as follows: If all data for a region is missing (no demand data at all), the shares remain NA. If a specific category (e.g., a specific carrier or end use) has no data across all periods for a region, its shares are filled with 0, assuming zero demand for that category. Biomass is split according to GDP per Capita (see toolSplitBiomass).

Usage

```
calcShareOdyssee(
  subtype = c("enduse", "carrier", "enduse_carrier"),
  feOnly = FALSE
)
```

Arguments

subtype	Character, dimension names, level of shares.
feOnly	if TRUE, output is absolute FE values

Value

MAGPIE object with historic shares

Author(s)

Robin Hasse, Antoine Levesque, Hagen Tockhorn

calcSharesBuildingDemand

Share of energy carriers or end uses in historic buildings demand

Description

Shares are calculated based on energy demands from the Odyssee data base for EU member states,

Usage

```
calcSharesBuildingDemand(subtype = c("enduse", "carrier"))
```

Arguments

subtype	Character, dimension names, level of shares.
---------	--

Value

MAGPIE object with historic shares

Author(s)

Robin Hasse, Antoine Levesque

calcSharesEU	<i>Calculate historical energy end-use shares based on final energy demands</i>
--------------	---

Description

Merges and transforms the calculated shares from the datasets: Odyssee 2024 IEA Energy Efficiency Indicators 2024 IEA Tracking Clean Energy Progress IEA World Energy Outlook 2014

Usage

calcSharesEU(thermal = FALSE, endOfHistory = 2025)

Arguments

thermal	logical, if TRUE, the end-use "appliances" shall be transformed into "refrigerators" with appropriate share
endOfHistory	An integer specifying the upper temporal boundary for historical data

Details

End-use shares are extrapolated using a log-linear regression on the existing data points while values in-between existing data are interpolated linearly. Data is merged following a strict priority hierarchy:

1. Odyssee - Provides detailed European data
2. IEA Energy Efficiency Indicators (EEI) - Selected countries, global scope
3. IEA China data
4. IEA World Energy Outlook (WEO) - Only used if EEI has fewer than 2 data points for a particular region-enduse combination
5. IEA Energy Technology Perspectives (ETP) - Only used if EEI has fewer than 2 data points and no WEO data is available

For specific regions (Africa, Middle East), data points from WEO are prioritized due to their regional specificity. These data points are extrapolated using growth factors determined by Tracking Clean Energy Progress (TCEP) to ensure consistent time series extension.

End-use shares are extrapolated using a log-linear regression on existing data points, while values between existing data points are interpolated linearly. The function preserves dataset consistency by avoiding mixing data sources when sufficient data points are available from a single source.

In the thermal case, the enduse "appliances" is transformed to "refrigerators" using the region-specific refrigerator share used in EDGE-B. Higher resolved data was not available.

Value

data.frame with historic end-use shares for each energy carrier

Author(s)

Hagen Tockhorn, Robin Hasse

calcShareTCEP

Calculate Enduse Shares from TCEP data base

Description

Calculate Enduse Shares from TCEP data base

Usage

calcShareTCEP()

Value

magpie object

Author(s)

Hagen Tockhorn

calcShareWEO

Calculate end use shares from WEO 2015

Description

Table 10.2 in the WEO 2015 report lists the share of regional enduses in the global final energy demand. Here, we calculate the share of enduses in the final energy demand for buildings in each region.

Usage

calcShareWEO()

Author(s)

Robin Hasse

calcStatusQuoShare	<i>Calculate Status Quo Share</i>
--------------------	-----------------------------------

Description

The share of heating systems that are replaced with the same technology during heating system replacement for each technology.

Usage

```
calcStatusQuoShare()
```

Details

Quantifying the status quo bias requires detailed data on heating system replacement activities with initial and final technology. There are very few and mostly rough sources on this. We therefore assume technology-specific and temporally and spatially invariant shares of initial systems that are identically replaced. This leaves enough degrees of freedom to the model to reflect region-specific stocks and preferences while representing the status quo bias. It might make sense to model the decision as a two-stage process in which agents first decide whether to just replace identically and thus follow the bias or decide for a new system independent of their old one. This is not yet reflected in this calculation.

Author(s)

Robin Hasse

calcSurface	<i>calcSurface</i>
-------------	--------------------

Description

Calculates country surface

Usage

```
calcSurface()
```

Value

Surface by country

Author(s)

Antoine Levesque

calcTCEP*IEA's Tracking clean energy progress 2014*

Description

IEA's Tracking clean energy progress 2014

Usage

calcTCEP(subtype = "enduse")

Arguments

subtype character, type of data

Author(s)

Robin Hasse

calcUEdemand*Calculate useful energy demand for space heating in buildings*

Description

Calculate useful energy demand for space heating in buildings

Usage

calcUEdemand(granularity = NULL)

Arguments

granularity character, name of BRICK granularity

Author(s)

Robin Hasse

calcUValue	<i>Calculate U-value</i>
------------	--------------------------

Description

Calculate U-value

Usage

```
calcUValue()
```

Author(s)

Robin Hasse

calcUvalueParameters	<i>Calculate fit parameters for U-value projections</i>
----------------------	---

Description

Estimates regional fit parameters for projecting building envelope U-values using mixed-effects regression. Combines data from Hotmaps, MessageIX Buildings, and EU Buildings Observatory to model thermal performance across regions based on climate indicators (HDD/CDD) and economic development (GDP per capita).

Usage

```
calcUvalueParameters(endOfHistory = 2025)
```

Arguments

`endOfHistory` Integer. The last year to consider as historical data (default: 2025)

Details

The function uses a linear mixed-effects model with the formula: $\log(\text{U-value} - \text{minU}) \sim \text{HDD} + \text{CDD} + \log(\text{gdppop} + 1) + (\text{interaction terms}) + (1 \mid \text{region})$ where the interaction terms are interactions between HDD/CDD and $\log(\text{gdppop} + 1)$, respectively, and $(1 \mid \text{region})$ denotes a region-specific offset.

This captures the relationship between U-values and:

- Heating degree days (HDD)
- Cooling degree days (CDD)
- GDP per capita (gdppop)
- Interaction terms: how climate effects vary with economic development

- Regional random effects: region-specific deviations from global pattern

The function handles regional parameter estimation in two ways:

1. For regions with sufficient historical data points, parameters are directly estimated via mixed-effects regression (lmer)
2. For regions with insufficient data, the function:
 - Uses global fixed effects as a baseline
 - Calculates the difference between aggregated regional projections and their reference values
 - Applies empirical Bayes shrinkage to balance between global pattern and regional differences
 - Adjusts intercepts using these shrunk offsets to create region-specific parameters

The resulting parameters can be used to project future U-values based on climate and economic scenarios.

Value

magpie object with regional fit parameters

Author(s)

Hagen Tockhorn

calcWeightFeBuildings *calculate FE demand in buildings*

Description

Simple estimate of total FE demand in different building types

Usage

```
calcWeightFeBuildings()
```

Details

Should only be used as weights not actual demand. The disaggregation to building types is too simplistic.

Author(s)

Robin Hasse

convertAssotermica	<i>Convert heating systems sales in Italy from Assotermica</i>
--------------------	--

Description

Convert heating systems sales in Italy from Assotermica

Usage

convertAssotermica(x)

Arguments

x	MagPIE object with Assotermica data
---	-------------------------------------

Value

MagPIE object with ISO 3 country codes

Author(s)

Robin Hasse

convertBDH	<i>Convert German heating system data from BDH</i>
------------	--

Description

Convert German heating system data from BDH

Usage

convertBDH(x)

Arguments

x	MagPIE object with BDH data
---	-----------------------------

Value

MagPIE object with ISO 3 country codes

Author(s)

Robin Hasse

convertBMK	<i>Convert heating systems sales in Austria from BMK</i>
------------	--

Description

Convert heating systems sales in Austria from BMK

Usage

```
convertBMK(x)
```

Arguments

x	MagPIE object with BMK data
---	-----------------------------

Value

MagPIE object with ISO 3 country codes

Author(s)

Robin Hasse

convertCensusHub	<i>convertCensusHub</i>
------------------	-------------------------

Description

Convert data from Census Hub

Usage

```
convertCensusHub(x, subtype)
```

Arguments

x	raw data
subtype	Eurostat code of data set

Value

MAGPie object with converted data

Author(s)

Robin Hasse

convertDaioglou	<i>convertDaioglou</i>
-----------------	------------------------

Description

Tidy data from Daioglou et al. 2012. Currently, only the specific floor space (m2/cap) is available from household data file. UN sources are dropped, if there is data from other sources for the same region. For the remaining data, if there is data from multiple UN Source for a region and period (i.e. different cities), the average of those sources is considered. This neglects the differences in population size of the cities. Selected data points are removed additionally.

Usage

```
convertDaioglou(x, subtype = "households.specific floor space")
```

Arguments

x	MAGPIE object with data from Daioglou et al. #nolint
subtype	.

Value

clean MAGPIE object with unique data points

Author(s)

Robin Hasse, Antoine Levesque

convertDeetman2020	<i>Convert Weibull lifetime distribution parameters for buildings</i>
--------------------	---

Description

Convert Weibull lifetime distribution parameters for buildings

Usage

```
convertDeetman2020(x, subtype)
```

Arguments

x	MAGPIE object with data from Deetman et al. 2020
subtype	character, building subsector (either 'residential' or 'commercial')

Author(s)

Robin Hasse

convertDestatis	<i>Convert data from the Federal Statistical Office of Germany</i>
-----------------	--

Description

Convert data from the Federal Statistical Office of Germany

Usage

```
convertDestatis(x, subtype)
```

Arguments

x	MagaPie object
subtype	character, data code

Value

MagPIE object

Author(s)

Robin Hasse

convertEEAfloorspace	<i>convert EEAfloorpace</i>
----------------------	-----------------------------

Description

Convert historic floor space data from the European Environment Agency.

Usage

```
convertEEAfloorspace(x)
```

Arguments

x	MAGPIE object with floor space data
---	-------------------------------------

Value

floor national floor space in million m2

Author(s)

Robin Hasse

convertEHI	<i>Convert EHI Heating Market Report</i>
------------	--

Description

Convert EHI Heating Market Report

Usage

convertEHI(x)

Arguments

x	MagPIE object with raw data
---	-----------------------------

Value

MagPIE object with country-level data

Author(s)

Robin Hasse

convertEHPA	<i>Correct heat pump sales data from EHPA</i>
-------------	---

Description

Correct heat pump sales data from EHPA

Usage

convertEHPA(x)

Arguments

x	MagPIE object with EHPA data
---	------------------------------

Value

MagPIE object with ISO 3 country codes

Author(s)

Robin Hasse

`convertEnergiforsk2016`*Convert European district heating prices*

Description

We should consider prper extrapolation

Usage`convertEnergiforsk2016(x)`**Arguments**

`x` MAgPIE object with original district heating prices

Author(s)

Robin Hasse

`convertEUBuildingsDB` *convertEUBuildingsDB*

Description

Convert to common units.

Usage`convertEUBuildingsDB(x, subtype)`**Arguments**

`x` MAgPIE object with data from EU Buildings Database
`subtype` character .

Value

clean MAgPIE object with

Author(s)

Robin Hasse

convertEurObservER	<i>convertEurObservER</i>
--------------------	---------------------------

Description

Each report of the Heat Pumps Barometer covers the two years before. There numerous cases, where reports contradict in overlapping values or there are huge jumps (probably due to changes in the accounting). This function creates smooth trajectories back-casting from the most recent data. If data overlaps, earlier data is scaled to match more recent data. In case of gaps, the trajectories are back-casted maintaining relative growth within each report. Growth rates between reports are interpolated. The Method yields mostly smooth trajectories always hitting the mos recent reported numbers.

Usage

```
convertEurObservER(x, subtype)
```

Arguments

x	raw data
subtype	Eurostat code of data set

Value

MAGPie object with converted data

Author(s)

Robin Hasse

convertEuropeanCommissionRenovation	<i>convertEuropeanCommissionRenovation</i>
-------------------------------------	--

Description

Drop EU28 aggregate and rename regions to ISO3.

Usage

```
convertEuropeanCommissionRenovation(x)
```

Arguments

x	MAGPIE object with data from EC report
---	--

Value

clean MAgPIE object with data from EU Buildings Database

Author(s)

Robin Hasse

convertEurostatBuildings
convertEurostatBuildings

Description

Convert buildings-related data from Eurostat.

Usage

```
convertEurostatBuildings(x, subtype)
```

Arguments

x	raw data
subtype	Eurostat code of data set

Value

MAgPie object with converted data

Author(s)

Robin Hasse

convertFEGECA *Convert heating systems sales in Spain from FEGECA*

Description

Convert heating systems sales in Spain from FEGECA

Usage

```
convertFEGECA(x)
```

Arguments

x	MagPIE object with Assotermica data
---	-------------------------------------

Value

MagPIE object with ISO 3 country codes

Author(s)

Robin Hasse

convertGDL	<i>convertGDL</i>
------------	-------------------

Description

Drop subnational data but keep other subgroups (i.e. rural/urban, income quartiles)

Usage

convertGDL(x, subtype)

Arguments

x MAgPIE object with data from EU Buildings Database
subtype .

Value

clean MAgPIE object with data from EU Buildings Database

Author(s)

Robin Hasse

convertHDDCDDtemp	<i>Convert temporary HDD and CDD data</i>
-------------------	---

Description

The temporary data already is at EDGE-B regional resolution. It is uniformly disaggregated to country level here to comply with the madrat framework before it is aggregated to the identical resolution in fullEDGEBUILDINGS.

Usage

convertHDDCDDtemp(x)

Arguments

x MagPIE object with HDD and CDD data at EDGE-B resolution

Value

HDD CDD data at country resolution

Author(s)

Robin Hasse

convertHotmaps	<i>convert Hotmaps</i>
----------------	------------------------

Description

convert Hotmaps

Usage

convertHotmaps(x)

Arguments

x MagPIE object with raw building stock data

Value

MagPIE object with converted data

Author(s)

Robin Hasse

convertHouseholdSurveys

Convert household survey data on energy carrier use for cooking, lighting and appliances ownership rates

Description

Convert household survey data on energy carrier use for cooking, lighting and appliances ownership rates

Usage

```
convertHouseholdSurveys(x, subtype)
```

Arguments

x	data.frame containing survey data
subtype	specifies data subset

Author(s)

Hagen Tockhorn

References

<https://hdl.handle.net/10419/301069>

convertIEA_OECD_Prices

read energy carrier prices

Description

read energy carrier prices

Usage

```
convertIEA_OECD_Prices(x, subtype)
```

Arguments

x	MagPIE object
subtype	character, data set

Author(s)

Robin Hasse

convertLTRS	<i>Convert Long Term Renovation Strategy</i>
-------------	--

Description

Convert Long Term Renovation Strategy

Usage

convertLTRS(x)

Arguments

x MagPIE object with data read from source

Value

MagPIE object

Author(s)

Robin Hasse

convertMessageIXBuildings	<i>Convert data from MessageIX-Buildings</i>
---------------------------	--

Description

Convert data from MessageIX-Buildings

Usage

convertMessageIXBuildings(x, subtype)

Arguments

x read-in data object
subtype specifies data base category

convertOdysee	<i>convertOdysee</i>
---------------	----------------------

Description

Rename regions and convert unit

Usage

```
convertOdysee(x, subtype = "250109")
```

Arguments

x	MAGPIE object with data from Odysee Database
subtype	dataset version

Value

clean MAGPIE object

Author(s)

Robin Hasse

convertPFUDB	<i>Converts Data from Primary Fuel and Useful Energy Data Base (PFUDB)</i>
--------------	--

Description

This was adapted from EDGE function 'getPFUDB.R'.

Usage

```
convertPFUDB(x)
```

Arguments

x	MAGPIE object with data from PFUDB #nolint
---	--

Value

magpie object

Author(s)

Hagen Tockhorn

convertPRISMA_ICT	<i>Convert PRISMA ICT Data to Standard Format</i>
-------------------	---

Description

Converts PRISMA ICT data to ISO country codes and fills missing countries. Primarily used for data center count data.

Usage

```
convertPRISMA_ICT(x, subtype)
```

Arguments

x	MAGPIE object from readPRISMA_ICT
subtype	Character string specifying data type (same as in readPRISMA_ICT)

Value

A magpie object with converted country codes and filled countries

Author(s)

Hagen Tockhorn

convertPulsen	<i>Convert Pulsen survey on Swedish heat pump market</i>
---------------	--

Description

Convert Pulsen survey on Swedish heat pump market

Usage

```
convertPulsen(x)
```

Arguments

x	MagPIE object with data
---	-------------------------

Value

MagPIE object with corrected region

Author(s)

Robin Hasse

convertSandberg	<i>Convert building life time data from Sandberg et al. 2016</i>
-----------------	--

Description

Convert building life time data from Sandberg et al. 2016

Usage

convertSandberg(x)

Arguments

x	MagPie object
---	---------------

Value

MagPIE object with full region coverage

Author(s)

Robin Hasse

convertSPIUG	<i>Convert heating systems stock and sales in Poland from SPIUG</i>
--------------	---

Description

Convert heating systems stock and sales in Poland from SPIUG

Usage

convertSPIUG(x)

Arguments

x	MagPIE object with PIUG data
---	------------------------------

Value

MagPIE object with ISO 3 country codes

Author(s)

Robin Hasse

convertTCEP	Convert TCEP data base
Description	
Convert TCEP data base	
Usage	
convertTCEP(x, subtype = "enduse")	
Arguments	
x	MAGPIE object with data from TCEP #nolint
subtype	character, type of data
Value	
magpie object	
Author(s)	
Hagen Tockhorn, Robin Hasse	
convertUNHouseholds	convertUNHouseholds

Description

Clean up UN data on household size and composition

Usage

convertUNHouseholds(x, subtype)

Arguments

x MAGPIE object with data from EU Buildings Database
subtype household variable

Value

clean MAGPIE object with

Author(s)

Robin Hasse

convertUniclimate	<i>Convert heating systems sales in France from Uniclimate</i>
-------------------	--

Description

Convert heating systems sales in France from Uniclimate

Usage

convertUniclimate(x)

Arguments

x	MagPIE object with PIUG data
---	------------------------------

Value

MagPIE object with ISO 3 country codes

Author(s)

Robin Hasse

convertVHK	<i>Convert VHK data</i>
------------	-------------------------

Description

Convert VHK data

Usage

convertVHK(x)

Arguments

x	MagPie object
---	---------------

Value

MagPIE object

Author(s)

Robin Hasse

convertWEO	<i>Convert WEO enduse shares w.r.t. to global final energy demand</i>
------------	---

Description

Convert WEO enduse shares w.r.t. to global final energy demand

Usage

```
convertWEO(x)
```

Arguments

x	readWEO object
---	----------------

Author(s)

Hagen Tockhorn

dot2Dash	<i>Replace dots by dashes</i>
----------	-------------------------------

Description

Replace dots by dashes

Usage

```
dot2Dash(x, rev = FALSE)
```

Arguments

x	character vector
rev	logical, if TRUE, replace dashes by dots

Value

character vector with replaced symbols

Author(s)

Robin Hasse

eur2usd	<i>Exchange rate between EUR and USD</i>
---------	--

Description

Convert EUR in given year to internal unit

Usage

```
eur2usd(x = 1, year = 2020, iso3c = "DEU", ...)
```

Arguments

x	value in EUR
year	integer, reference year
iso3c	region
...	further arguments to GDPuc::convertSingle

Value

value in constant 2017 US\$MER

Author(s)

Robin Hasse

extrapolateMissingPeriods	<i>Extrapolate missing values beyond existing periods</i>
---------------------------	---

Description

Extrapolate missing values beyond existing periods

Usage

```
extrapolateMissingPeriods(chunk, key, slopeOfLast = 5)
```

Arguments

chunk	grouped data.frame
key	column holding values for extrapolation
slopeOfLast	number of values for boundary regression

Value

data.frame with extrapolated column key

Author(s)

Robin Hasse, Hagen Tockhorn

fullBRICK

fullBRICK

Description

Compute complete input data set for BRICK

Usage

```
fullBRICK(rev = 0, granularity = NULL)
```

Arguments

rev	data revision which should be used as input (positive numeric).
granularity	character, name of BRICK granularity

Author(s)

Robin Hasse

fullEDGEBUILDINGS

fullEDGEBUILDINGS

Description

Function that produces the complete ISO data set required for the EDGE - Buildings model.

Usage

```
fullEDGEBUILDINGS(rev = 0)
```

Arguments

rev	data revision which should be used as input (positive numeric).
-----	---

Author(s)

Antoine Levesque, Robin Hasse, Hagen Tockhorn

See Also

[readSource, getCalculations, calcOutput](#)

Examples

```
## Not run:
fullEDGEBUILDINGS()

## End(Not run)
```

join_all	<i>Join by all common columns</i>
----------	-----------------------------------

Description

Additional columns to join by that have different names can be give. The kind of joint can also be specified.

Usage

```
join_all(
  x,
  y,
  by = NULL,
  .direction = "left",
  exclude = "value",
  silent = TRUE,
  ...
)
```

Arguments

- | | |
|------------|---|
| x, y | A pair of data frames |
| by | character vector of columns to join by beyond the commonly named |
| .direction | type of dplyr join, 'left' by default |
| exclude | character vector with column names that should be excluded from the search for join columns |
| silent | boolean, whether to print type of joint the join columns |
| ... | additional arguments passed to join function |

Author(s)

Robin Hasse

readAssotermica	<i>Read heating systems sales in Italy from Assotermica</i>
-----------------	---

Description

Read heating systems sales in Italy from Assotermica

Usage

```
readAssotermica(subtype = NULL)
```

Arguments

subtype	Character, report to read. if NULL, the most recent report is used for every data point.
---------	--

Value

magpie object

Author(s)

Robin Hasse

readBDH	<i>Read German heating system data from BDH</i>
---------	---

Description

Read German heating system data from BDH

Usage

```
readBDH()
```

Value

MagPIE object with yearly sales of heating systems in Germany

Author(s)

Robin Hasse

readBEAM2	<i>Read buildings characteristics data from the Built Environment Analysis Model (BEAM^2)</i>
-----------	---

Description

Read buildings characteristics data from the Built Environment Analysis Model (BEAM^2)

Usage

readBEAM2()

Author(s)

Hagen Tockhorn

readBMK	<i>Read heating systems sales in Austria from BMK</i>
---------	---

Description

Read heating systems sales in Austria from BMK

Usage

readBMK()

Value

MagPIE object with yearly sales of heating systems in Germany

Author(s)

Robin Hasse

readCensusHub

*Read data from the European Census Hub***Description**

The Census Hub of the European Statistical System provides census data from 2011 on population and housing of EU member states. 2021 census data is announced to be published there too.

- typeVintage: number of dwellings of different construction periods and building types (single, two, multiple dwellings). Show data on: dwellings, location: nations, Topics: Type of building, Period of construction (select all) -> CSV (Separator: Comma)

Usage

```
readCensusHub(subtype)
```

Arguments

subtype census subset

Value

MAGPIE object with data

Author(s)

Robin Hasse

Source

<https://ec.europa.eu/CensusHub2/>

readDaiglou

*Household and demand data from Daiglou et al. 2012***Description**

Compilation of survey data. Quintile 0 stands for the entire population.

- households: Global data on population, household size, floor space per capita (m2/cap), population density (cap/km2), GINI and household expenditures (PPP2005/cap). Sources are written into a dimension to allow later cleaning based on sources
- shares: Final energy shares of end uses in buildings. Sources are dropped.

Usage

```
readDaiglou(subtype = "households.specific floor space")
```

Arguments

subtype .

Author(s)

Robin Hasse, Antoine Levesque

References

<https://doi.org/10.1016/j.energy.2011.10.044>

readDeetman2020

Read Weibull lifetime distribution parameters for buildings

Description

Read Weibull lifetime distribution parameters for buildings

Usage

```
readDeetman2020(subtype = "residential")
```

Arguments

subtype character, building subsector (either 'residential' or 'commercial')

Value

MagPIE object with Weibull lifetime parameters for different world regions

Author(s)

Robin Hasse

Source

<https://doi.org/10.1016/j.jclepro.2019.118658>

readDestatis	<i>Read data from the Federal Statistical Office of Germany</i>
--------------	---

Description

Read data from the Federal Statistical Office of Germany

Usage

```
readDestatis(subtype)
```

Arguments

subtype	character, data code
---------	----------------------

- "31121-0005": Construction work completed regarding new buildings

Value

MagPIE object

Author(s)

Robin Hasse

readECEMF	<i>Read ECEMF data</i>
-----------	------------------------

Description

Project data: European Climate and Energy Modelling Forum

Usage

```
readECEMF(subtype = "FEPrices")
```

Arguments

subtype	character, type of data
---------	-------------------------

Details

- FEPrices: price components w/o VAT in EUR/MWh

Author(s)

Robin Hasse

readEEAfloorspace	<i>Read national floor space from EEA</i>
-------------------	---

Description

Floor space for many European countries between 1990 and 2009 in thousand m2.

Usage

```
readEEAfloorspace()
```

Value

magpie object with floor space

Author(s)

Robin Hasse

See Also

[readSource](#)

Examples

```
## Not run: a <- readSource(type = "EEAfloorspace")
```

readEHI	<i>Read EHI Heating Market Report</i>
---------	---------------------------------------

Description

Read EHI Heating Market Report

Usage

```
readEHI(subtype = NULL)
```

Arguments

subtype	Character, report to read. if NULL, the most recent report is used for every data point.
---------	--

Value

magpie object

Author(s)

Robin Hasse

readEHPA

Read heat pump sales data from EHPA

Description

This data was provided upon request. It is free for non-profit organisations but we are not allowed to share it.

Usage

```
readEHPA(subtype = "hpSales")
```

Arguments

subtype character, type of data

Value

MagPIE object with HP sales data for EU member states

Author(s)

Robin Hasse

readEnergiforsk2016

Read European district heating prices

Description

Read European district heating prices

Usage

```
readEnergiforsk2016()
```

Author(s)

Robin Hasse

readEUBuildingsDB	<i>Read EU Buildings Database</i>
-------------------	-----------------------------------

Description

Compilation of various data sources from the European Commission. Because of the considerable size of this database, a subtype has to be provided. The subtype represents the category of items. It is possible to append selected variables for further filtering.

Usage

```
readEUBuildingsDB(subtype = "")
```

Arguments

subtype character .

Value

magpie object

Author(s)

Robin Hasse

See Also

[readSource](#)

Examples

```
## Not run: a <- readSource(type = "EUBuildingsDB")
```

readEurObservER	<i>Read EurObservER</i>
-----------------	-------------------------

Description

PDF market reports on Renewables in Europe

Usage

```
readEurObservER(subtype)
```

Arguments

subtype .

Value

magpie object

Note

I download the report and snip a png of the table. Then I extract the data with [this online tool](#) and save it as a tabular csv in the source folder. The Tool is not fully reliable. This function throws an error if values don't sum up. You then have to correct the csv file manually.

Author(s)

Robin Hasse

Source

<https://www.eurobserv-er.org/>

readEuropeanCommissionRenovation
<i>EU report on historic renovation</i>

Description

This function reads data on renovation from a pdf file of an EC report from 2019 (Tables 2 - 13). There is no period dimension. The data represents averages between 2012 and 2016.

Usage

```
readEuropeanCommissionRenovation(subtype = "csv")
```

Arguments

subtype character, source to read from: "csv" (default) or "pdf" (requires the suggested package pdftools)

Value

MAGPIE object with renovation data

Author(s)

Robin Hasse

Source

<https://op.europa.eu/en/publication-detail/-/publication/97d6a4ca-5847-11ea-8b81-01aa75ed71a1>

See Also

[readSource](#)

Examples

```
## Not run: a <- readSource(type = "EuropeanCommissionRenovation")
```

readEurostatBuildings *Read Eurostat data on buildings*

Description

Read various data sets from Eurostat that are used in the modelling of buildings

- nrg_inf_hptc: Heat pumps - technical characteristics by technologies
- nrg_d_hhq: Disaggregated final energy consumption in households
- lfst_r_lfsd2hh: Number of households by degree of urbanisation (2006-2023)
- ilc_hcmh02: Average size of dwelling by household type and degree of urbanisation (2012)
- ilc_lvho31: Average size of dwelling by household composition and degree of urbanisation (2020, 2023)
- cens_21dwbnr_r3: Dwellings by type of building, number of rooms or useful floor space (2021)
- sts_cobp_a: Building permits (1992 - 2024)
- ilc_lvhe02: Persons living in private households by heating system used in the dwelling, household composition and degree of urbanisation (2023)
- ilc_lvhe04: Persons heating their dwelling with district heating by used energy source, household composition and degree of urbanisation (2023)
- ilc_lvhe05: Persons heating their dwelling with other than district heating by used energy source, household composition and degree of urbanisation (2023)
- hbs_car_t315: Household characteristics by degree of urbanisation (1988-2020)
- ilc_lvho01: Distribution of population by degree of urbanisation, dwelling type and income group (2003 - 2024)
- ilc_lvho03: Average number of rooms per person by tenure status and dwelling type (2003 - 2024)

Usage

```
readEurostatBuildings(subtype)
```

Arguments

subtype Eurostat code of data set

Value

MAGPIE object with data

Note

see <https://ec.europa.eu/eurostat/web/energy/data/energy-balances> for definitions of codes

Author(s)

Robin Hasse

readFEGECA

Read heating systems sales in Spain from FEGECA

Description

Read heating systems sales in Spain from FEGECA

Usage

```
readFEGECA(subtype = NULL)
```

Arguments

subtype Character, report to read. if NULL, the most recent report is used for every data point.

Value

magpie object

Author(s)

Robin Hasse

`readGDL`*Read Global Data Lab*

Description

Different databases each comprising the harmonised results of regional/national surveys. Available subtypes are

- SHDI: Subnational Human development Database
- AreaDatabase: Subnational development Indicators (low/medium income countries)

Usage

```
readGDL(subtype)
```

Arguments

subtype .

Value

magpie object

Author(s)

Robin Hasse

Source

<https://globaldatalab.org/>

References

<https://doi.org/10.1038/sdata.2019.38>

<https://globaldatalab.org/asset/286/Smits%20GDL%20Working%20Paper%2016-101%20v360.pdf>

See Also

[readSource](#)

Examples

```
## Not run: a <- readSource(type = "GDL")
```

readHDDCDD	<i>Read historical and projected degree-day data</i>
------------	--

Description

Read historical and projected degree-day data

Usage

```
readHDDCDD(subtype = "ssp2")
```

Arguments

subtype specifies SSP scenarios (e.g. "ssp2" or "ssp1lssp2")

Author(s)

Hagen Tockhorn

readHDDCDDtemp	<i>Temporary reading of HDD and CDD data</i>
----------------	--

Description

Heating and cooling degree days calculated with `mredgebldings::calcHDDCDD`. Calculating this data currently requires manual help. We therefore read the result as a source here. This becomes obsolete once the primary calculation is stable.

Usage

```
readHDDCDDtemp()
```

Author(s)

Robin Hasse

readHotmaps	<i>read EU28 building stock from Hotmaps</i>
-------------	--

Description

Consistent building stock of EU28 residential and commercial sector. The data presents a snapshot, I guess in 2016 but this is not clear to me yet.

Usage

```
readHotmaps()
```

Details

There are a few duplicate rows. Due to inconsistent values we average across duplicates.

Author(s)

Robin Hasse

readHouseholdSurveys	<i>Read household survey data on energy carrier use for cooking, lighting and appliances ownership rates</i>
----------------------	--

Description

Read household survey data on energy carrier use for cooking, lighting and appliances ownership rates

Usage

```
readHouseholdSurveys(subtype = c("cooking", "lighting", "appliances"))
```

Arguments

subtype	Character specifying end-use
---------	------------------------------

Author(s)

Hagen Tockhorn

References

<https://hdl.handle.net/10419/301069>

readIEA_acOwnership	<i>Read AC ownership rates per household from IEA Sources</i>
---------------------	---

Description

Read AC ownership rates per household from IEA Sources

Usage

```
readIEA_acOwnership()
```

Author(s)

Hagen Tockhorn

readIEA_China	<i>Read final energy data by energy end-use for China</i>
---------------	---

Description

The data is obtained from the 2021 IEA report "An Energy Sector Roadmap to Carbon Neutrality in China" from Figure 3.25

Usage

```
readIEA_China()
```

Author(s)

Robin Hasse

readIEA_coolingEfficiencies	<i>Read IEA Global Weighted Average of Air Conditioning Efficiencies</i>
-----------------------------	--

Description

Read IEA Global Weighted Average of Air Conditioning Efficiencies

Usage

```
readIEA_coolingEfficiencies()
```

Author(s)

Hagen Tockhorn

readIEA_OECD_Prices	<i>read energy carrier prices</i>
---------------------	-----------------------------------

Description

read energy carrier prices

Usage

```
readIEA_OECD_Prices(subtype)
```

Arguments

subtype	character, data set
---------	---------------------

Author(s)

Robin Hasse

readLTRS	<i>Read Long Term Renovation Strategy</i>
----------	---

Description

In the LTRS, EU member states have to describe their building stock. Unfortunately, this isn't standardised in any way so it is hard to make use of this data. So far, we only use data from two figures in the German LTRS.

Usage

```
readLTRS()
```

Value

MagPIE object

Author(s)

Robin Hasse

`readMessageIXBuildings`*Read data from MessageIX-Buildings*

Description

Read data from MessageIX-Buildings

Usage

```
readMessageIXBuildings(subtype = "uvalue")
```

Arguments

subtype	specifies data base category
---------	------------------------------

`readOdyssee`*Read Odyssee data base on households and services*

Description

Copied from website: The Odyssee database contains detailed energy consumption by end-use and their drivers as well as energy efficiency and CO2-related indicators. Latest available data is provided by national representatives, such as energy agencies or statistical organization, from all EU countries as well as Norway, Serbia, Switzerland and the United Kingdom.

Usage

```
readOdyssee(subtype = "260305")
```

Arguments

subtype	dataset version
---------	-----------------

Value

magpie object

Note

To download new data, log into the website, select all items of all levels and download the data as CSV (1 observation = 1 row, delimiter: ", "). If this exceeds the download limit, download multiple files, e.g. by splitting along periods.

Variables are labels with the item code but full names can be found in the source data

Author(s)

Pascal Führlich, Robin Hasse

Source

<https://odyssee.enerdata.net/database/>

readPFUDB

Read Data from Primary Fuel and Useful Energy Data Base (PFUDB)

Description

This was adapted from EDGE function 'getPFUDB.R'.

Usage

readPFUDB()

Value

magpie object

Author(s)

Hagen Tockhorn

readPRISMA_ICT

Read PRISMA ICT Data

Description

Reads ICT and data center electricity demand data from PRISMA project Excel file. Supports different regional resolutions and data center counts.

Usage

readPRISMA_ICT(subtype)

Arguments

subtype Character string specifying data type

Value

A magpie object containing the requested ICT/data center data

Author(s)

Hagen Tockhorn

readPulsen	<i>Read Pulsen survey on Swedish heat pump market</i>
------------	---

Description

Read Pulsen survey on Swedish heat pump market

Usage

```
readPulsen()
```

Value

MagPIE object

Author(s)

Robin Hasse

Source

<https://skvp.se/skvpold/statistik/pulsen>

readSandberg	<i>Read building life time data from Sandberg et al. 2016</i>
--------------	---

Description

Read building life time data from Sandberg et al. 2016

Usage

```
readSandberg()
```

Value

MagPIE object with life time parameters of buildings

Author(s)

Robin Hasse

Source

<https://doi.org/10.1016/j.enbuild.2016.05.100>

`readSPIUG`*Read heating systems stock and sales in Poland from SPIUG*

Description

Read heating systems stock and sales in Poland from SPIUG

Usage

```
readSPIUG()
```

Value

magpie object

Author(s)

Robin Hasse

`readTCEP`*Read TCEP data base*

Description

Read TCEP data base

Usage

```
readTCEP(subtype = "enduse")
```

Arguments

subtype character, type of data

Value

magpie object

Author(s)

Hagen Tockhorn, Robin Hasse

readUNHouseholds	<i>Read UN data on Household Size & Composition</i>
------------------	---

Description

Compilation of indicators on household size and membership composition. The estimates are based on 814 unique data points from 172 countries or areas, representing approximately 97 per cent of the world's population in 2019, collected between 1960 to 2018.

Usage

```
readUNHouseholds(subtype)
```

Arguments

subtype	household variable
---------	--------------------

Value

magpie object

Author(s)

Robin Hasse

Source

<https://www.un.org/development/desa/pd/data/household-size-and-composition>

See Also

[readSource](#)

Examples

```
## Not run: a <- readSource(type = "UNHouseholds")
```

readUniclimate	<i>Read heating systems sales in France from Uniclimate</i>
----------------	---

Description

Read heating systems sales in France from Uniclimate

Usage

```
readUniclimate()
```

Value

magpie object

Author(s)

Robin Hasse

readVHK	<i>Read VHK data</i>
---------	----------------------

Description

multiple EU studies mostly on different categories of appliances

Usage

```
readVHK(subtype)
```

Arguments

subtype	character, specifies study and the respective csv file to read: <study>.<file> (without file extension)
---------	---

Value

MagPIE object

Author(s)

Robin Hasse

readWEO	<i>Read WEO enduse shares w.r.t. to global final energy demand</i>
---------	--

Description

Read WEO enduse shares w.r.t. to global final energy demand

Usage

```
readWEO(subtype)
```

Arguments

subtype	variable to define data subset ("Buildings", "Transport", "Industry")
---------	---

Author(s)

Hagen Tockhorn

toolAddThermal	<i>Introduce refrigerators as appliances</i>
----------------	--

Description

Introduce refrigerators as appliances

Usage

```
toolAddThermal(df, mapping, fridgeShare, feOnly = TRUE, shareOf = NULL)
```

Arguments

df	data.frame
mapping	original EDGE-B region mapping
fridgeShare	regional shares of refrigerators
feOnly	are absolute values (TRUE) or shares (FALSE) considered?
shareOf	column name to calculate share of - must be given if feOnly = FALSE

Author(s)

Hagen Tockhorn

toolAggregateBrick	<i>Aggregate BRICK input data to lower granularity</i>
--------------------	--

Description

Aggregate BRICK input data to lower granularity

Usage

```
toolAggregateBrick(x, granularity = NULL, weight = NULL)
```

Arguments

x	MagPIE object at BRICK default resolution
granularity	character, name of BRICK granularity
weight	magPIE object with weights, if NULL (default), an extensive variable is assumed and values are summed instead of averaged

Value

named list with aggregated value and weights

Author(s)

Robin Hasse

toolCalcShares	<i>Calculate Shares</i>
----------------	-------------------------

Description

Calculate Shares

Usage

```
toolCalcShares(df, colShare)
```

Arguments

df	data.frame
colShare	column to be grouped over

Value

data.frame with calculated shares for colShare

Author(s)

Hagen Tockhorn

 toolCompleteRenFlows *Complete Matrix of flows from old to new to cover given items*

Description

Extrapolate from data on selected technologies to all technologies required

Usage

```
toolCompleteRenFlows(x, from, to)
```

Arguments

x	MagPIE object, renovation flows to complete
from	MagPIE object with origin quantities
to	MagPIE object with destination quantities

Details

We assume that there is a high status quo probability that people just do identical replacement. If they don't, they chose the system with probabilities proportional to the general shares.

Value

MagPIE object

Author(s)

Robin Hasse

 toolCountryFillAvg *Tool: CountryFillAvg*

Description

This function expects a MAGPIE object with ISO country codes in the spatial dimension. It applies toolCountryFill but fills with the average across regions.

Usage

```
toolCountryFillAvg(x, ...)
```

Arguments

x	MAGPIE object with ISO country codes in the spatial dimension
...	arguments passed to toolCountryFill

Author(s)

Robin Hasse

toolDisaggregate	<i>Disaggregate energy demand by carriers into end uses by each carrier</i>
------------------	---

Description

This function takes energy demand data that differentiates only by carrier and disaggregates it further such that all relevant combinations of carrier and end use are differentiated. It ensures that given end use shares (across all carriers) are met.

Usage

```
toolDisaggregate(
  data,
  enduseShares,
  outliers = NULL,
  exclude = NULL,
  dataDisagg = NULL,
  regionMapping = NULL,
  forceShares = NULL
)
```

Arguments

data	data.frame with energy demand data differentiated by carriers that is to be disaggregated
enduseShares	data.frame with end use shares that have to be met. These shares might be given at aggregated regional resolution if regionmapping is provided.
outliers	list of regions where naive disaggregation estimate shall be used.
exclude	data.frame with the columns carrier and enduse that should list all combinations of the two that are excluded. All remaining combinations of the carriers in data and the end uses in enduseShares are considered.
dataDisagg	data.frame similar to data but already disaggregated by carriers and end uses. The average distribution of its disaggregation will be used as the target distribution for the minisation.
regionMapping	data.frame with the columns region and regionAgg that maps the regions between data and enduseShares.
forceShares	data frame with shares to enforce for carrier-enduse combinations with single year data points per region.

Details

The function essentially performs a quadratic optimisation. The constraints make sure that both the carrier quantities and the end use shares are met. This generally leaves infinite solutions (in problematic cases none). To get to one disaggregation, the deviation from a given distribution across all relevant combinations of carriers and end uses is minimised.

Author(s)

Hagen Tockhorn, Robin Hasse

toolGetBrickMapping	<i>Get BRICK aggregation mappings</i>
---------------------	---------------------------------------

Description

Provides all mappings needed to aggregate BRICK input data to a lower granularity.

Usage

```
toolGetBrickMapping(granularity)
```

Arguments

granularity	character, name of BRICK granularity
-------------	--------------------------------------

Value

named list of brick dimension aggregation mappings

Author(s)

Robin Hasse

toolUnitConversion	<i>Convert unit</i>
--------------------	---------------------

Description

Takes MAgPIE object with a variable dimension and a unit dimension, and converts the unit according to a unit conversion table.

Usage

```
toolUnitConversion(x, unitConversion, removeUnit = FALSE)
```

Arguments

<code>x</code>	MAGPIE object
<code>unitConversion</code>	data.frame with the columns from, to, factor
<code>removeUnit</code>	boolean remove unit after conversion for clean variable names

Author(s)

Robin Hasse

Index

.disaggregate, 5
.enforceShares, 6
.fillTimeseries, 7

aggToLevel, 7

calcACOwnershipRates, 8
calcACOwnershipRatesFilled, 8
calcACOwnershipRegression, 9
calcBRG, 10
calcBuildingStock, 10
calcCarrierPrices, 11
calcConstruction, 11
calcCostConstruction, 12
calcCostDemolition, 12
calcCostRenovationBS, 13
calcCostRenovationHS, 13
calcDensity, 14
calcDiscountRate, 14
calcEfficiencyRegression, 15
calcElecDemandICT, 15
calcFEbyEUEC, 16
calcFEUEfficiencies, 17
calcFloorspacePast, 18
calcFloorspacePerCap, 18
calcHDDCDD, 19
calcHeatingCapacity, 20
calcHeatingSystem, 20
calcHeatingSystemReplacement, 21
calcHeatingSystemSales, 21
calcHouseholdSize, 22
calcIEA_EEI, 22
calcIEAPFU, 23
calcLifetimeParams, 24
calcLTRS, 24
calcMatchingReference, 25
calcOdysseeStock, 26
calcOutput, 23, 58
calcPFUDB, 26
calcPopulationBuildings, 27

calcRenDepth, 27
calcRenovation, 28
calcRenovationCostModel, 28
calcShareChina, 29
calcShareETP, 29
calcShareIEA_EEI, 30
calcShareOdyssee, 30
calcSharesBuildingDemand, 31
calcSharesEU, 32
calcShareTCEP, 33
calcShareWEO, 33
calcStatusQuoShare, 34
calcSurface, 34
calcTCEP, 35
calcUEdemand, 35
calcUValue, 36
calcUvalueParameters, 36
calcWeightFeBuildings, 37
convertAssotermica, 38
convertBDH, 38
convertBMK, 39
convertCensusHub, 39
convertDaiglou, 40
convertDeetman2020, 40
convertDestatis, 41
convertEEAfloorspace, 41
convertEHI, 42
convertEHPA, 42
convertEnergiforsk2016, 43
convertEUBuildingsDB, 43
convertEurObservER, 44
convertEuropeanCommissionRenovation,
44
convertEurostatBuildings, 45
convertFECECA, 45
convertGDL, 46
convertHDDCDDtemp, 46
convertHotmaps, 47
convertHouseholdSurveys, 48

convertIEA_OECD_Prices, 48
convertLTRS, 49
convertMessageIXBuildings, 49
convertOdyssee, 50
convertPFUDB, 50
convertPRISMA_ICT, 51
convertPulsen, 51
convertSandberg, 52
convertSPIUG, 52
convertTCEP, 53
convertUNHouseholds, 53
convertUniclima, 54
convertVHK, 54
convertWEO, 55

dot2Dash, 55

eur2usd, 56
extrapolateMissingPeriods, 56

fullBRICK, 57
fullEDGEBUILDINGS, 57

GDPuc::convertSingle, 56
getCalculations, 58

join_all, 58

mredgebuidings
 (mredgebuidings-package), 5
mredgebuidings-package, 5

readAssotermica, 59
readBDH, 59
readBEAM2, 60
readBMK, 60
readCensusHub, 61
readDaiglou, 61
readDeetman2020, 62
readDestatis, 63
readECEMF, 63
readEEAfloorspace, 64
readEHI, 64
readEHPA, 65
readEnergiforsk2016, 65
readEUBuildingsDB, 66
readEurObserver, 66
readEuropeanCommissionRenovation, 67
readEurostatBuildings, 68
readFEGECA, 69

readGDL, 70
readHDDCDD, 71
readHDDCDDtemp, 71
readHotmaps, 72
readHouseholdSurveys, 72
readIEA_acOwnership, 73
readIEA_China, 73
readIEA_coolingEfficiencies, 73
readIEA_OECD_Prices, 74
readLTRS, 74
readMessageIXBuildings, 75
readOdyssee, 75
readPFUDB, 76
readPRISMA_ICT, 76
readPulsen, 77
readSandberg, 77
readSource, 58, 64, 66, 68, 70, 79
readSPIUG, 78
readTCEP, 78
readUNHouseholds, 79
readUniclima, 80
readVHK, 80
readWEO, 81

toolAddThermal, 81
toolAggregateBrick, 82
toolCalcShares, 82
toolCompleteRenFlows, 83
toolCountryFillAvg, 83
toolDisaggregate, 84
toolGetBrickMapping, 85
toolUnitConversion, 85