Package: mrremind (via r-universe)

September 5, 2024

Type Package

Title MadRat REMIND Input Data Package

Version 0.187.3

Date 2024-09-05

Description The mrremind packages contains data preprocessing for the REMIND model.

License LGPL-3 | file LICENSE

URL https://github.com/pik-piam/mrremind

Depends edgeTransport (>= 1.5.5), madrat (>= 3.7.1), magclass (>= 6.16.1), mrcommons (>= 1.38.0), mrdrivers (>= 2.0.0), R (>= 2.10.0)

Imports assertr, broom, car, countrycode, data.table, dplyr, ggplot2, Hmisc, luscale, magrittr, nnls, purrr, quitte (>= 0.3105.0), R.utils, readODS, readr, readxl, reshape2, rlang, rmndt, tibble, tidyr, tidyselect, zoo

Suggests covr, knitr, rmarkdown, testthat

VignetteBuilder knitr

Encoding UTF-8

LazyData no

RoxygenNote 7.3.2

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

RemoteUrl https://github.com/pik-piam/mrremind

RemoteRef HEAD

RemoteSha a388dd16ec19199206f7477f6c8f0bbe86ee3751

Contents

`mrremind-package` *mrremind: MadRat REMIND Input Data Package*

Description

The mrremind packages contains data preprocessing for the REMIND model.

Author(s)

Maintainer: Lavinia Baumstark <lavinia@pik-potsdam.de>

Authors:

- Renato Rodrigues
- Antoine Levesque
- Julian Oeser
- Christoph Bertram
- Ioanna Mouratiadou
- Aman Malik
- Felix Schreyer
- Bjoern Soergel
- Marianna Rottoli
- Abhijeet Mishra
- Alois Dirnaichner
- Michaja Pehl
- Anastasis Giannousakis
- David Klein
- Jessica Strefler
- Lukas Feldhaus
- Regina Brecha
- Sebastian Rauner
- Jan Philipp Dietrich
- Stephen Bi
- Falk Benke
- Pascal Weigmann
- Oliver Richters
- Robin Hasse
- Sophie Fuchs
- Rahel Mandaroux
- Johannes Koch

See Also

Useful links:

- https://github.com/pik-piam/mrremind

calcAGEB *Calculate REMIND final energy variables from historical AGEB values*

Description

Calculate REMIND final energy variables from historical AGEB values

Usage

```
calcAGEB(subtype = "balances")
```

Arguments

subtype	data subtype. Either "balances" ("Auswertungstabellen zur Energiebilanz Deutschland") or "electricity" ("Bruttostromerzeugung in Deutschland nach Energieträgern")

Value

A magpie object.

Author(s)

Falk Benke

calcBiomassPrices *read biomass supply curves from Magpie emulator*

Description

read biomass supply curves from Magpie emulator

Usage

```
calcBiomassPrices()
```

Value

Magpie object with two parameters determining linear biomass supply curve

calcBP *Calculate REMIND variables from historical BP values*

Description

Calculate REMIND variables from historical BP values

Usage

```
calcBP()
```

Author(s)

Falk Benke

calcCapacity *calc Capacity*

Description

provides historical capacity values in TW

Usage

```
calcCapacity(subtype)
```

Arguments

subtype	data subtype. Either "capacityByTech" or "capacityByPE"

Value

magpie object of capacity data

Author(s)

Renato Rodrigues, Stephen Bi

Examples

```
## Not run:
calcOutput("Capacity",subtype="capacityByTech")

## End(Not run)
```

calcCapacityFactor *calc Capacity Factor*

Description

provides capacity factor values

Usage

```
calcCapacityFactor()
```

Value

magpie object of the capacity factor data

Author(s)

Renato Rodrigues, Stephen Bi

Examples

```
## Not run:
calcOutput("CapacityFactor")

## End(Not run)
```

calcCapacityFactorHist *calc Capacity Factor*

Description

provides capacity factor values

Usage

```
calcCapacityFactorHist(subtype)
```

Arguments

subtype	data subtype. Either "wind" or "windoff"

Value

magpie object of the capacity factor data

Author(s)

Renato Rodrigues, Stephen Bi

Examples

```
## Not run:
calcOutput("CapacityFactor")

## End(Not run)
```

---

calcCapTarget *Capacity targets from two sources*

Description

The capacity targets (GW) at regional level are produced from two different databases- UNFCCC_NDC database, an update of the Rogelj 2017 paper (see readme in inputdata), and REN21 Global Renewables report The UNFCCC_NDC capacity targets are further broken down to conditional and unconditional targets.

Usage

```
calcCapTarget(sources)
```

Arguments

sources	Database source

Author(s)

Aman Malik, Oliver Richters

---

calcCCSbounds *Calculate CCS bound indicator for 2025 and 2030*

Description

Calculate CCS bound indicator for 2025 and 2030

Usage

```
calcCCSbounds()
```

Author(s)

Jessica Strefler, Lavinia Baumstark

See Also

calcOutput, readSource

`calcCCScapacity` *calc CCS capacity*

Description

Calculate CCS capacity from IEA CCUS data

Usage

```
calcCCScapacity(subtype)
```

Arguments

`subtype`	either 'historical' for data until 2022 or 'projections' for projections in 2020, 2025 and 2030 (including some redistribution on EU/NEU level)

Author(s)

Anne Merfort, Falk Benke

`calcCement` *Calculate Historic Cement Production*

Description

Combines cement production data from `readvanRuijven2016()` and `readUSGS(cement)` into a single data set, using USGS data from 2005 on.

Usage

```
calcCement()
```

Value

A list with a `magpie` object `x` with country-level cement production in tonnes, `weight`, `unit`, `description`, and `min` fields.

Author(s)

Michaja Pehl

See Also

calcOutput

calcCementShare *Calculate Cement Share in NONMET FE Use*

Description

Estimated shares of cement in NONMET final energy use based on OECD and Non-OECD figures from IEA 2017 Energy Technology Perspectives. Shares are weighted by GDP for aggregation and converge towards global values by 2100.

Usage

```
calcCementShare()
```

Value

A list with a magpie object x, weight, unit, description, min, and max.

Author(s)

Michaja Pehl

See Also

calcOutput()

calcChemicalFeedstocksShare *Calculate Chemical Feedstock share projections*

Description

Calculates the share of CHEMICAL in CHEMICAL = NECHEM and converges it towards the maximum value of either OECD or non-OECD countries by 2050.

Usage

```
calcChemicalFeedstocksShare()
```

Value

A list with a magpie object x, weight, unit, description, min, and max.

Author(s)

Michaja Pehl

See Also

calcOutput()

---

calcClinker_to_cement_ratio *Calculate Clinker-to-Cement Ratio*

Description

Calculate Clinker-to-Cement Ratio

Usage

```
calcClinker_to_cement_ratio()
```

Value

A list with a magpie object x, weight, unit, and description.

Author(s)

Michaja Pehl

See Also

calcOutput(), readADVANCE_WP2(), convertADVANCE_WP2()

---

calcCoolingSharesAll *Calculate Cooling Type Shares*

Description

This function merges the output of two other functions that calculate REMIND input data for the shares of cooling types per electricity technology and REMIND region, using as initial information the Davies (2013) data per electricity technology and GCAM region. The two other functions separately calculate data for the base year and for future time steps. The source data provide most required information but some assumptions on missing data are also made.

Usage

```
calcCoolingSharesAll()
```

Value

MAgPIE object on cooling type shares per elecricity technology and REMIND region

Author(s)

Ioanna Mouratiadou

See Also

calcOutput, readDaviesCooling, convertDaviesCooling, calcCoolingSharesBase,calcCoolingSharesFuture

Examples

```
## Not run:

a <- calcOutput("CoolingSharesAll")

## End(Not run)
```

---

calcCoolingSharesBase *Calculate Cooling Type Shares for the Base Year*

Description

This function calculates REMIND input data for the shares of cooling types per electricity technology and REMIND region in 2005, using as initial information the Davies (2013) data per electricity technology and GCAM region. The source data provide most required information but some assumptions on missing data are also made.

Usage

```
calcCoolingSharesBase()
```

Value

MAgPIE object on cooling type shares per elecricity technology and REMIND region

Author(s)

Lavinia Baumstark, Ioanna Mouratiadou

See Also

calcOutput, readDaviesCooling, convertDaviesCooling, calcCoolingSharesAll,calcCoolingSharesFuture

Examples

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

## End(Not run)
```

calcCoolingSharesFuture *Calculate Cooling Type Shares for Future Timesteps*

Description

This function calculates REMIND input data for the shares of cooling types per electricity technology and REMIND region in post-2020, using as initial information the Davies (2013) data per electricity technology and GCAM region. The source data provide most required information but some assumptions on missing data are also made.

Usage

```
calcCoolingSharesFuture()
```

Value

MAgPIE object on cooling type shares per elecricity technology and REMIND region

Author(s)

Ioanna Mouratiadou

See Also

calcOutput, readDaviesCooling, convertDaviesCooling, calcCoolingSharesAll,calcCoolingSharesBase

Examples

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

## End(Not run)
```

calcCostsTrade *Calculate trade costs*

Description

Provides REMIND data for PE tradecosts (energy losses on import).

Usage

```
calcCostsTrade()
```

Value

REMIND data forPE tradecosts (energy losses on import) and corresonding weights (1) as a list of two MAgPIE objects

Author(s)

Lavinia Baumstark

See Also

calcOutput, readSource

Examples

```
## Not run:
calcOutput("calcCostsTrade")

## End(Not run)
```

---

calcCostsTradePeFinancial

Calculate Trade Cost

Description

Provides REMIND data for PE trade cost (energy losses on import, export and use).

Usage

```
calcCostsTradePeFinancial()
```

Author(s)

Regina Brecha, Lavinia Baumstark

See Also

calcOutput, readSource

Examples

```
## Not run:
calcOutput("CostsTradePeFinancial")

## End(Not run)
```

calcCostsWeathering *Calculate costs of transport of enhanced weathering*

Description

Calculate costs of transport of enhanced weathering

Usage

```
calcCostsWeathering()
```

Value

transport costs of spreading rock on the fields

See Also

calcOutput

Examples

```
## Not run:
calcOutput("CostsWeathering")

## End(Not run)
```

calcDiffInvestCosts *Aggregated investment cost data for REMIND regions (based on IEA_WEO)*

Description

Disaggregated investment cost data is aggregated and technologies renamed to REMIND names

Usage

```
calcDiffInvestCosts(subtype)
```

Arguments

subtype	"Invest_Costs", or "Efficiency"

Details

REMIND does not have a classification of coal power plants e.g., sub-critical. Therefore, countries are given coal plant costs assuming what type of coal plants are expected to develop there. For other technologies, certain assumptions are taken to change to REMIND convention.

Value

Magpie object with aggregated but differentiated investment costs for some technologies.

Author(s)

Aman Malik

`calcDspvShare` *Calculates the share of distributed solar pv, wind-onshore/offshore, hydro-small/large from 2015 to 2050. For spv - Only includes grid-connected pv.*

Description

Calculates the share of distributed solar pv, wind-onshore/offshore, hydro-small/large from 2015 to 2050. For spv - Only includes grid-connected pv.

Usage

```
calcDspvShare(subtype)
```

Arguments

`subtype`	Either "current","expert", or "irena". Current are current shares extended until 2050. expert is based on Robert P.'s judgement, and irena are based on IRENA's 2050 global numbers. Hydro case remains same in all cases

Details

Known limitations - source for distributed spv (IEA Renewables 2019) is different than source for total spv (IRENA 2019)

Value

magpie object with REMIND-aggregated regions

Author(s)

Aman Malik

calcEarlyRetirementAdjFactor
calc Early Retirement Adjustment Factor

Description

provides the extra retirement rate to account for relatively old fleet technologies retirement

Usage

```
calcEarlyRetirementAdjFactor(subtype = "none")
```

Arguments

subtype	Some scenarios may require certain regions to increase retirement rate, e.g. PPCA coal phase-out

Value

magpie object of additional adjusment percentage to be added to the fraction of the early retired capital in countries to account for relatively old technologies fleet

Author(s)

Renato Rodrigues

See Also

calcOutput

Examples

```
## Not run:
calcOutput(type = "EarlyRetirementAdjFactor")

## End(Not run)
```

calcEconometricEmiParameter

Calculate baseline emissions of waste

Description

Provides REMIND data for CO2 parameters to calculate baseline emissions of waste from population and investment.

Usage

```
calcEconometricEmiParameter()
```

Value

REMIND data forCO2 parameters to calculate baseline emissions of waste from population and investment and corresonding weights (population) as a list of two MAgPIE objects

Author(s)

Lavinia Baumstark

See Also

calcOutput, readSource

Examples

```
## Not run:
calcOutput("calcEconometricEmiParameter")

## End(Not run)
```

calcEDGAR7Fgases *calcEDGAR7Fgases*

Description

calcEDGAR7Fgases

Usage

```
calcEDGAR7Fgases()
```

Author(s)

Gabriel Abrahao

calcEDGETransport *Prepare EDGETransport inputs*

Description

Prepare EDGETransport inputs

Usage

```
calcEDGETransport(subtype)
```

Arguments

subtype	REMIND/iterative EDGE-T input data subtypes

Value

REMIND/iterative EDGE-T input data for all scenario combinations

Author(s)

Johanna Hoppe

See Also

readSource

Examples

```
## Not run: a <- calcOutput(type = "EDGETransport", subtype = "CAPEXandNonFuelOPEX", aggregate = F)
```

calcEEAGHGProjections *Calculate EEA emission projections from the two projections sources provided by EEA*

Description

Calculate EEA emission projections from the two projections sources provided by EEA

Usage

```
calcEEAGHGProjections()
```

Value

A magpie object.

Author(s)

Falk Benke

`calcEffortSharingRefEmi`
calc Effort Sharing Reference Emissions

Description

provides region specific Effort Sharing Reference Emissions

Usage

```
calcEffortSharingRefEmi(subtype)
```

Arguments

`subtype`	type of reference emissions used to define emission reduction target fo European Effort Sharing Decision: EEA_GHG, Eurostat_GHG, REMIND_GHG (deprecated) or REMIND_CO2.

Value

2005 reference emissions to calculate effort sharing decision targets

Author(s)

Renato Rodrigues

Examples

```
## Not run:
calcOutput("EffortSharingRefEmi",subtype="Eurostat_GHG")

## End(Not run)
```

calcEffortSharingTarget *calc Effort Sharing Target*

Description

provides region specific Effort Sharing Emission target

Usage

```
calcEffortSharingTarget()
```

Value

target data magpie object

Author(s)

Renato Rodrigues

Examples

```
## Not run:
calcOutput("EffortSharingTarget")

## End(Not run)
```

calcEmber *calc Ember*

Description

prepare the yearly Ember electricity data set To use only a part of the Ember data, call calcOutput("Ember", subtype = "...") and convert to TW if you want to use capacities as input data to REMIND.

Usage

```
calcEmber(subtype = "all")
```

Arguments

subtype	data subtype. Either "capacity", "generation" or "all"

Value

A [‘magpie‘][magclass::magclass] object.

Author(s)

Pascal Weigmann

See Also

[‘calcOutput()‘]

`calcEmiAirPollLandUse` *calcEmiAirPoll calculate Air Pollution Emissions*

Description

calcEmiAirPoll calculate Air Pollution Emissions

Usage

```
calcEmiAirPollLandUse()
```

Value

magpie object

Author(s)

Julian Oeser

See Also

`calcOutput`

Examples

```
## Not run:  a <- calcOutput(type="EmiAirPollLandUse")
```

calcEmiCO2LandUse *EmiCO2LandUse calculate co2 emissions from land use change*

Description

EmiCO2LandUse calculate co2 emissions from land use change

Usage

```
calcEmiCO2LandUse()
```

Value

magpie object

Author(s)

Julian Oeser

See Also

calcOutput

Examples

```
## Not run:  a <- calcOutput(type="EmiCO2LandUse")
```

calcEmiLULUCFCountryAcc *calcEmiLULUCFCountryAcc*

Description

hisorical LULUCF emissions following country accounting

Usage

```
calcEmiLULUCFCountryAcc(subtype)
```

Arguments

subtype	Valid subtypes are 'UNFCCC'

Value

Magpie object with historical LULUCF emissions

Author(s)

Felix Schreyer

`calcEmiMac1990` *Calculate baseline emissions for maccs for 1990*

Description

Provides REMIND data for baseline emissions for maccs for 1990.

Usage

```
calcEmiMac1990()
```

Value

REMIND data for baseline emissions for maccs for 1990 and corresonding weights (NULL) as a list of two MAgPIE objects

Author(s)

Lavinia Baumstark

See Also

`calcOutput`, `readSource`

Examples

```
## Not run:
calcOutput("calcEmiMac1990")

## End(Not run)
```

`calcEmiPollutantExo` *calcEmiPollutantExo calculate EmiPollutantExo based on RCP data*

Description

calcEmiPollutantExo calculate EmiPollutantExo based on RCP data

Usage

```
calcEmiPollutantExo(subtype, aviationshippingsource = "RCP")
```

Arguments

`subtype`	Either 'Waste' or 'AviationShipping'
`aviationshippingsource`	Defines source for aviation and shipping emissions. Either 'RCP' or 'Lee-GAINS'.

Value

magpie object

Author(s)

Julian Oeser

See Also

`calcOutput`

Examples

```
## Not run:  a <- calcOutput(type="EmiPollutantExo")
```

calcEmiReference *calc European Reference Emissions*

Description

provides European 2030 emission targets in relation to 1990 and 2005 emissions

Usage

```
calcEmiReference()
```

Value

2030 emission reductions tragets for 40

Author(s)

Falk Benke and Renato Rodrigues

Examples

```
## Not run:
calcOutput("EmiReference")

## End(Not run)
```

calcEmissionFactorsFeedstocks *Calculate emission factors for feedstocks in the chemicals industry using emissions from UNFCCC and energy demands from IEA Energy Balances*

Description

Calculate emission factors for feedstocks in the chemicals industry using emissions from UNFCCC and energy demands from IEA Energy Balances

Usage

```
calcEmissionFactorsFeedstocks()
```

Value

A list with a magpie object x, weight, unit, description.

Author(s)

Falk Benke, Renato Rodrigues, Simón Moreno Leiva

See Also

`calcOutput()`

---

calcEmissions *calcEmissions*

Description

calcEmissions

Usage

```
calcEmissions(datasource = "CEDS16")
```

Arguments

datasource	"CEDS16", "CEDS2REMIND", "CEDS2024", "EDGAR", "EDGAR6", "EDGAR8" "LIMITS", "ECLIPSE", "GFED", "CDIAC"

Value

magpie object with historical emissions

Author(s)

Steve Smith, Pascal Weigmann

---

calcEmiTarget *Output for 2 policy cases*

Description

Output for 2 policy cases

Usage

```
calcEmiTarget(sources, subtype)
```

Arguments

sources	currently only UNFCCC_NDC
subtype	"Ghgshare2005", "Ghgfactor", "Ghghistshare"

Author(s)

Aman Malik, Christoph Bertram, Oliver Richters

`calcETSRefEmi` *calc ETS Reference Emissions*

Description

provides region specific ETS Reference Emissions

Usage

```
calcETSRefEmi(subtype)
```

Arguments

`subtype`	type of reference emissions used to define emission reduction targets for European regulations: EEA_GHG

Value

2005 reference emissions to calculate ETS targets

Author(s)

Renato Rodrigues

Examples

```
## Not run:
calcOutput("ETSRefEmi",subtype="EEA_GHG")

## End(Not run)
```

`calcEuropeanEnergyDatasheets` *Calculate REMIND variables from European Energy Datasheets*

Description

Calculate REMIND variables from European Energy Datasheets

Usage

```
calcEuropeanEnergyDatasheets(subtype)
```

Arguments

`subtype`	data subtype. Either "EU28" (data from June 20 including GBR) or "EU27" (latest data from August 23 without GBR)

Value

A `magpie` object.

Author(s)

Falk Benke

`calcExogDemScen` *calculate exogenuous FE and ES demand pathways*

Description

prepare data for exogenuous FE and ES demand pathways that do not come from EDGE models but from other sources and/or scenario literature. REMIND can be fixed to those demand pathways if the switch cm_exogDem_scen is activated.

Usage

```
calcExogDemScen()
```

Value

A [‘magpie‘][magclass::magclass] object.

Author(s)

Felix Schreyer

calcExpertGuess *Calculate expert guesses*

Description

Calculate expert guesses

Usage

```
calcExpertGuess(subtype)
```

Arguments

subtype	must be 'tradeConstraints' (more to come)

Author(s)

Falk Benke

calcFE *Calculates FE historical from IEA energy balances, projections from EDGE, and historical values from IEA WEO 2019*

Description

Calculates FE historical from IEA energy balances, projections from EDGE, and historical values from IEA WEO 2019

Usage

```
calcFE(source = "IEA", scenario_proj = "SSP2", ieaVersion = "default")
```

Arguments

source	"IEA" or "IEA_WEO"
scenario_proj	"SSP2" by default unless overwritten
ieaVersion	Release version of IEA data, either 'default' (vetted and used in REMIND) or 'latest'.

Author(s)

Lavinia Baumstark, Aman Malik

calcFEdemand *Calculates Final Energy Demand for Industry, Buildings and Transport*

Description

Calculates Final Energy Demand for Industry, Buildings and Transport

Usage

```
calcFEdemand()
```

Author(s)

Falk Benke

calcFeDemandBuildings *Returns the EDGE-Buildings data as REMIND variables*

Description

Returns the EDGE-Buildings data as REMIND variables

Usage

```
calcFeDemandBuildings(subtype)
```

Arguments

subtype	either "FE", "FE_buildings", or "UE_buildings"

Author(s)

Robin Hasse

calcFeDemandIndustry *Calculates FE demand in industry as REMIND variables*

Description

Calculates FE demand in industry as REMIND variables

Usage

```
calcFeDemandIndustry(use_ODYM_RECC = FALSE)
```

Arguments

use_ODYM_RECC	per-capita pathways for 'SDP_xx' scenarios? (Defaults to 'FALSE'.)

Author(s)

Michaja Pehl

calcFeDemandTransport *Calculates FE demand in transport as REMIND variables*

Description

Calculates FE demand in transport as REMIND variables

Usage

```
calcFeDemandTransport()
```

Author(s)

Alois Dirnaicher, Johanna Hoppe

calcFEShares *FE Share parameters used in REMIND*

Description

FE Share parameters used in REMIND

Usage

```
calcFEShares(subtype)
```

Arguments

subtype	'ind_coal' for the share of coal used in industry. 'ind_bio' for the share of biomass used in industry

Author(s)

Antoine Levesque

calcFETaxes *Calculate FETaxes*

Description

Reads in the data of the source IIASA_subs_taxes, by country. and calculate taxes at the final energy delivery level to the end-use sectors (industry, buildings and transport). Regional aggregation is done via the respective energy quantities as weights.

Usage

```
calcFETaxes(subtype = "taxes")
```

Arguments

subtype	choose between tax rates ("taxes") or subsidies rate ("subsidies") output

Value

MAgPIE object

Author(s)

Christoph Bertram and Renato Rodrigues

See Also

calcOutput, readIIASA_subs_taxes, convertIIASA_subs_taxes

Examples

```
## Not run:
calcOutput("FETaxes")


## End(Not run)
```

---

calcFGas *generate F-Gases based on IMAGE data*

Description

generate F-Gases based on IMAGE data

Usage

```
calcFGas(subtype = "interpolate2025")
```

Arguments

subtype	"interpolate2025" will intepolate from EDGAR historical data from 2025-2050 to account for the very old IMAGE scenarios. Any other subtype will ignore this step.

Value

magpie object with F-gases information

Author(s)

Lavinia Baumstark

See Also

calcOutput, readSource

Examples

```
## Not run:
x <- calcOutput("FGas")

## End(Not run)
```

calcFloorspace *Floor space in buildings*

Description

Residential, commercial and total floor space from EDGE-B. Set

Usage

```
calcFloorspace(onlyTotal = FALSE)
```

Arguments

onlyTotal	boolean, only give total instead of sub-sectoral floor space

Value

MAgPIE object with buildings floor space

Author(s)

Antoine Levesque, Robin Hasse

calcFossilExtraction *calc Fossil Extraction*

Description

provides coefficients for fossil fuels (oil, gas and coal) and uranium extraction cost equations.

Usage

```
calcFossilExtraction(subtype = "FossilExtraction")
```

Arguments

subtype	Either 'FossilExtraction' or 'UraniumExtraction'

Value

magpie object of the coefficients for fossil fuels and uranium extraction cost equations

Author(s)

Renato Rodrigues, Felix Schreyer

See Also

calcOutput

Examples

```
## Not run:
calcOutput(type = "FossilExtraction", subtype = "FossilExtraction")

## End(Not run)
```

---

calcGAINS *calcGAINS*

Description

Calculates air pollutant emissions and emission factors (user can choose) based on GAINS emissions and activity data. Result is given on GAINS sector level. User can choose between aggregated and extended sectoral resolution. Results are given for multiple scenarios. Scenario design is partly taken from the GAINS data and partly created in this function (particularly the SSPs).

Usage

```
calcGAINS(subtype = "emission_factors", sectoral_resolution = "extended")
```

Arguments

subtype	decides whether emissions or emission factors are returned
sectoral_resolution	aggreaged or extenden (uses different GAINS input data)

---

calcGAINSEmi *Calculate air pollution emissions and emission factors from GAINS data*

Description

Provides input data for exoGAINSAirpollutants.R

Usage

```
calcGAINSEmi(subtype = "emissions")
```

Arguments

subtype	"emission_factors", "emissions","emissions_starting_values"

Value

Emissions and emission factors

Author(s)

Sebastian Rauner

See Also

calcOutput

Examples

```
## Not run:
calcOutput("calcGAINSEmi")

## End(Not run)
```

`calcGEA2012` *calcGEA2012*

Description

Extracts oil, gas and coal data from the GEA 2012 into a scenario- and time-dependent grade structure

Usage

```
calcGEA2012(subtype, datatype)
```

Arguments

`subtype`	oil, coal, gas, or bounds
`datatype`	extraseed, exportbound, or decoffset for bounds subtype

Value

MAgPIE object containing regionally aggregated GEA 2012 data

Author(s)

Stephen Bi

See Also

calcOutput

Examples

```
## Not run:  a <- calcOutput("GEA2012")
```

`calcGlobalEnergyMonitor` *Calc capacities from Global Energy Monitor*

Description

Calculate near-term expectations of capacities for use in fullVALIDATION.R

Usage

```
calcGlobalEnergyMonitor()
```

Value

A `magpie` object.

Author(s)

Falk Benke

`calcHistorical` *Gather reference data from various sources.*

Description

Gather reference data from various sources.

Usage

```
calcHistorical()
```

calcHRE *Calculate Final Energy for the buildings sector from Heat Roadmap Europe scenarios*

Description

Calculate Final Energy for the buildings sector from Heat Roadmap Europe scenarios

Usage

```
calcHRE()
```

Value

A magpie object.

Author(s)

Pascal Weigmann

calcIEA_ETP *Calculate REMIND emission variables from IEA ETP values*

Description

Calculate REMIND emission variables from IEA ETP values

Usage

```
calcIEA_ETP()
```

Value

A magpie object.

Author(s)

Falk Benke

calcIEA_EVOutlook *Calculate REMIND variables from IEA Global EV Outlook data*

Description

Calculate REMIND variables from IEA Global EV Outlook data

Usage

```
calcIEA_EVOutlook()
```

Author(s)

Falk Benke

calcIEA_WorldEnergyOutlook *Calculate REMIND variables from IEA World Energy Outlook data.*

Description

Calculate REMIND variables from IEA World Energy Outlook data.

Usage

```
calcIEA_WorldEnergyOutlook()
```

Author(s)

Falk Benke

calcIndustry_CCS_limits *Calculate Limits on Industry CCS Capacities*

Description

Calculate Limits on Industry CCS Capacities

Usage

```
calcIndustry_CCS_limits(
  a1 = 0.3,
  a2 = 0.15,
  installation_minimum = 1,
 stage_weight = c(Operational = 1, `In construction` = 1, `Advanced development` = 0.5,
    `Early development` = 0.2),
  facility_subsector = c(Cement = "cement", Chemical = "chemicals",
    `Hydrogen / Ammonia / Fertiliser` = "chemicals", Ethan = "chemicals",
    `Iron and Steel Production` = "steel"),
  region_mapping = NULL
)
```

Arguments

a1, a2	Annual growth factors of CCS capacity limits, for the first ten years and thereafter, default to 0.7 and 0.2 (70 % and 20 %, respectively).
installation_minimum	Minimum emission capacity (in MtCO~2~/year) capacities are rounded up to. Defaults to 0.5 (500 ktCO~2~/year).
stage_weight	A named vector of weight factors for different lifecycle stages. See Details.
facility_subsector	A named vector mapping the "Facility Industry" of CCS projects to REMIND industry subsectors. See Details.
region_mapping	A data frame with columns iso3c and region detailing the regional resolution on which data should be extrapolated. If NULL (the default), extrapolation is done at the country level.

Details

The limits on industry CCS capacities are calculated from data of the Global Status of CCS 2023 report (through readGlobalCCSinstitute(). CCS projects are

- filtered for valid (i.e. not "Under Evaluation") data for "Operation date" and "CO~2~ capture capacity"
- assigned to REMIND industry subsectors according to facility_subsector, which defaults to

Facility Industry	subsector
Cement	cement
Chemical	chemicals
Hydrogen / Ammonia / Fertiliser	chemicals
Ethan	chemicals
Iron and Steel Production	steel

- weighted by lifecycle stage according to stage_weight, which defaults to

Lifecycle stage	weight
Operational	100 %
In construction	100 %
Advanced development	50 %
Early development	20 %

The resulting project capacities constitute the limits on industry subsector CCS capacity for 2025. The limit on CCS capacities for regions (or countries if `region_mapping` is `NULL`) is set to a value of total 2025 subsector CCS capacity, times the regions share in subsector activity (e.g. cement production) of the SSP2EU scenario

- in 2030 if the region as some CCS capacity in 2025 in a different industry subsector, or
- in 2035 if the region has no industry CCS capacity in 2030 at all.

CCS capacities are increased by the annual growth factor `a1` for the ten first years, and by the annual growth factor `a2` afterwards (defaulting to 70 % and 20 %, respectively).

Value

A list with a `magpie` object `x`, `weight`, `unit`, `description`, and `min`.

Author(s)

Michaja Pehl

`calcIndustry_EEK` *Industry Energy Efficiency Capital*

Description

Industry Energy Efficiency Capital

Usage

```
calcIndustry_EEK(kap)
```

Arguments

`kap`	General internal capital stock, as calculated internally by ‘calcCapital()‘.

Value

A list with a [‘magpie‘][magclass::magclass] object ‘x‘, ‘weight‘, ‘unit‘, and ‘description‘ fields.

`calcindustry_max_secondary_steel_share`
Calculate Maximum Secondary Steel Production Share

Description

Reads ExpertGuess/industry_max_secondary_steel_share and expands to all ‘scenarios‘/‘regions‘ using default data. See [‘tool_expand_tibble()‘] for details.

Usage

```
calcindustry_max_secondary_steel_share(scenarios = NULL, regions = NULL)
```

Arguments

`scenarios`	A character vector of scenarios to expand data to.
`regions`	A character vector of regions to expand data to.

Value

A list with a [‘magpie‘][magclass::magclass] object ‘x‘.

`calcindustry_specific_FE_limits`
Thermodynamic Limits for Industry Specific FE Demand

Description

Return `readindustry_subsectors_specific('industry_specific_FE_limits')` in a format usable as a REMIND input.

Usage

```
calcindustry_specific_FE_limits()
```

Value

A `magpie` object.

Author(s)

Michaja Pehl

`calcIO` *Calc Input Output*

Description

Computes IEA-based model data for different "subtypes" by use of raw IEA "Energy Balances" data and a mapping that corresponds to the structure of "products" and "flows" of IEA.

Usage

```
calcIO(
 subtype = c("input", "output", "output_biomass", "trade", "input_Industry_subsectors",
    "output_Industry_subsectors", "IEA_output", "IEA_input"),
  ieaVersion = "default"
)
```

Arguments

`subtype`	Data subtype. See default argument for possible values.
`ieaVersion`	Release version of IEA data, either 'default' (vetted and used in REMIND) or 'latest'.

Details

Mapping structure example: IEA product ANTCOAL used for IEA flow TPATFUEL, contributes via REMIND technology coaltr for generating sesofos from pecoal (REMIND names)

When using subtype `output_Industry_subsectors`, additional corrections are applied to the IEA data in `tool_fix_IEA_data_for_Industry_subsectors`.

Value

IEA data as MAgPIE object aggregated to country level

Author(s)

Anastasis Giannousakis

See Also

`calcOutput`

Examples

```
## Not run:
a <- calcOutput("IO", subtype = "output")

## End(Not run)
```

`calcJRC_IDEES` *Calculate selected REMIND energy and emission variables from historical JRC IDEES values*

Description

Calculate selected REMIND energy and emission variables from historical JRC IDEES values

Usage

```
calcJRC_IDEES(subtype)
```

Arguments

`subtype`	one of • `'Industry'`: calculate REMIND Industry variables • `'Transport'`: calculate REMIND Transport variables • `'ResCom'`: calculate REMIND Residential and Commercial variables

Value

A `magpie` object.

Author(s)

Falk Benke

`calcKLWdamage` *write KLW damage parameters (from Kotz et al. 2024) into input data they are country-specific and should not be aggregated to the regional level at all*

Description

write KLW damage parameters (from Kotz et al. 2024) into input data they are country-specific and should not be aggregated to the regional level at all

Usage

```
calcKLWdamage(subtype)
```

Arguments

`subtype`	"beta1", "beta2", "maxGMT"

Value

MAgPIE object of damage parameters for KLW damage function on country level and for 1000 boostrapping samples

Author(s)

Franziska Piontek

`calcMACCsCO2` *Read in abatement potential for CO2 land-use change derived from MAgPIE*

Description

Rrange of possible abatement between maximum and minimum emission level in a year

Usage

```
calcMACCsCO2()
```

Value

MAgPIE object

Author(s)

David Klein

Examples

```
## Not run:
calcOutput("MACCsCO2")


## End(Not run)
```

calcnonEnergyIndFE *Final energy demand for feedstocks (non-energy use)*

Description

Final energy demand for feedstocks (non-energy use)

Usage

```
calcnonEnergyIndFE()
```

Value

A magpie object.

Author(s)

Renato Rodrigues

See Also

calcOutput().

calcOtherFossilInElectricity *Calculate projected electricity from waste and other fossils using energy demands from IEA Energy Balances.*

Description

This is used in remind2 reporting as input data to calculate additional capacity and secondary energy variables.

Usage

```
calcOtherFossilInElectricity()
```

Details

The projection focuses on a tight mitigation scenario and assumes that all fossil emissions from waste burning / other fossil processes can be reduced to 0 by 2050. Should be replaced in the future by actual modeling of waste / other fossil plants, or at least connected to RCP scenario assumptions.

Value

A list with a magpie object x, weight, unit, description.

Author(s)

Robert Pietzcker, Falk Benke

See Also

calcOutput()

---

calcPE *Computes Primary Energy variables*

Description

Computes Primary Energy variables

Usage

```
calcPE(subtype = "IEA", ieaVersion = "default")
```

Arguments

subtype	source for calculation, either "IEA" or "IEA_WEO"
ieaVersion	Release version of IEA data, either 'default' (vetted and used in REMIND) or 'latest'.

Value

a magclass object

---

calcPETaxes *Calculate SubsStationary*

Description

Reads in the data of the source IIASA_subs_taxes, by country. and calculate taxes at primary energy level. Regional aggregation is done via the respective energy quantities as weights.

Usage

```
calcPETaxes(subtype = "subsidies")
```

Arguments

subtype	subsidies rate ("subsidies") output

Value

MAgPIE object

Author(s)

Christoph Bertram and Renato Rodrigues

See Also

calcOutput, readIIASA_subs_taxes, convertIIASA_subs_taxes

Examples

```
## Not run:
calcOutput("PETaxes")


## End(Not run)
```

`calcPlasticsEoL` *calculates projections for the end of life fate of plastic waste in particular, calculates the share that is incinerated*

Description

calculates projections for the end of life fate of plastic waste in particular, calculates the share that is incinerated

Usage

```
calcPlasticsEoL()
```

Value

A list with a magpie object `x`, `weight`, `unit`, `description`.

Author(s)

Falk Benke, Simón Moreno Leiva

See Also

calcOutput()

calcPotentialGeothermal *Calculate Geothermal potential*

Description

Provides geothermal potential data

Usage

```
calcPotentialGeothermal()
```

Value

geothermal potential data MAgPIE object

Author(s)

Renato Rodrigues

See Also

calcOutput

Examples

```
## Not run:
calcOutput("PotentialGeothermal")


## End(Not run)
```

calcPotentialHydro *Calculate hydro potential*

Description

Provides hydro potential data

Usage

```
calcPotentialHydro()
```

Value

hydro potential data and corresonding weights as a list of two MAgPIE objects

Author(s)

Lavinia Baumstark

See Also

calcOutput, readWGBU, convertWGBU, readSource

Examples

```
## Not run:
calcOutput("PotentialHydro")


## End(Not run)
```

---

calcPotentialWeathering *Calculate hydro potential*

Description

Provides weathering potential data

Usage

```
calcPotentialWeathering()
```

Value

weathering potential data and corresonding weights as a list of two MAgPIE objects

Author(s)

Lavinia Baumstark

See Also

calcOutput

Examples

```
## Not run:
calcOutput("PotentialWeathering")


## End(Not run)
```

calcPotentialWindOff *Calculate wind offshore potential*

Description

Provides wind offshore potential data

Usage

```
calcPotentialWindOff()
```

Value

wind offshore potential data and corresonding weights as a list of two MAgPIE objects

Author(s)

Chen Chris Gong

See Also

calcOutput, readNREL, convertNREL, readSource

Examples

```
## Not run:
calcOutput("PotentialWindOff")


## End(Not run)
```

calcPotentialWindOn *Calculate wind onshore potential*

Description

Provides wind onshore potential data

Usage

```
calcPotentialWindOn()
```

Value

wind onshore potential data and corresonding weights as a list of two MAgPIE objects

Author(s)

Lavinia Baumstark

See Also

calcOutput, readNREL, convertNREL, readSource

Examples

```
## Not run:
calcOutput("PotentialWindOn")


## End(Not run)
```

---

calcProdShares *Manufacture production shares for spc and wind*

Description

Shares of world manufacture for spv modules and wind turbines for 2018 and 2019

Usage

```
calcProdShares()
```

Value

magpie object with REMIND-aggregated region

Author(s)

Aman Malik

---

calcProjectPipelines *calc Project Pipelines*

Description

Calculate the expected near-term deployment of technologies based on projects that are currently either being built or in a planning stage for some technologies multiple sources are available

Usage

```
calcProjectPipelines(subtype)
```

Arguments

subtype	choose technology ‘biomass‘, ‘coal‘, ‘geothermal‘, ‘hydro‘, ‘nuclear‘, ‘solar‘, ‘wind‘ or ‘CCS‘

Author(s)

Pascal Weigmann

---

calcRLDCCoefficients *calc RLDC Coefficients*

Description

provides RLDC coefficients values

Usage

```
calcRLDCCoefficients(subtype = "LoB")
```

Arguments

subtype	Either ’LoB’ or ’Peak’

Value

magpie object of the RLDC coefficients data

Author(s)

Renato Rodrigues

See Also

calcOutput

calcOutput

Examples

```
## Not run:
calcOutput(type="RLDCCoefficients",subtype='LoB')

## End(Not run)
```

calcSharedTarget *calc Shared Target*

Description

provides region specific target

Usage

```
calcSharedTarget(subtype)
```

Arguments

subtype	data subtype. Either "FErenewablesShare", ...

Value

target data magpie object

Author(s)

Renato Rodrigues

Examples

```
## Not run:
calcOutput("SharedTarget",subtype="FErenewablesShare")

## End(Not run)
```

calcShareIndFE *Share of Industry Subsectors in FE Use*

Description

Calculates industry subsector shares in final energy carrier use for the `fixed_shares` realisation of the `industry` module.

Usage

```
calcShareIndFE()
```

Details

For the region mapping `regionmapping_21_EU11.csv`, these are based on IEA data from `calcOutput(type = 'FEdemand')`, for all other region mappings on vintage data which is ultimately based on Enerdata data.

Value

A magpie object.

Note

There is a discrepancy between the shares calculated from these two sources, that will affect REMIND emission reporting.

Author(s)

Lavinia Baumstark

Michaja Pehl

See Also

calcOutput().

---

calcSolar *calcSolar calculate Area, Capacity and Energy for photovoltaics (PV) and contentrated solar power (CSP)*

Description

calcSolar calculate Area, Capacity and Energy for photovoltaics (PV) and contentrated solar power (CSP)

Usage

```
calcSolar()
```

Value

magpie object

Author(s)

Julian Oeser, modified by Renato Rodrigues

See Also

calcOutput

Examples

```
## Not run:  a <- calcOutput(type="Solar")
```

calcSteelStock *Calculate Steel Stock from Mueller steel stock per capita and WDI population*

Description

Calculate Steel Stock from Mueller steel stock per capita and WDI population

Usage

```
calcSteelStock()
```

Value

A magpie object.

Author(s)

Falk Benke

calcSteel_Projections *EDGE-Industry*

Description

Functions for calculating industry activity trajectories.

Usage

```
calcSteel_Projections(
  subtype = "production",
  match.steel.historic.values = TRUE,
  match.steel.estimates = "none",
  save.plots = NULL,
  China_Production = NULL
)

calcIndustry_Value_Added(
  subtype = "physical",
  match.steel.historic.values = TRUE,
  match.steel.estimates = "none",
  save.plots = NULL,
  China_Production = NULL
)
```

Arguments

`subtype` One of

- `production` Returns trajectories of primary and secondary steel production (`calcSteel_Projections()`).
- `secondary.steel.max.share` Returns the maximum share of secondary steel in total steel production (`calcSteel_Projections()`).
- `physical` Returns physical production trajectories for cement (`calcIndustry_Value_Added()`).
- `economic` Returns value added trajectories for all subsectors (`calcIndustry_Value_Added()`).

`match.steel.historic.values`
Should steel production trajectories match historic values?

`match.steel.estimates`
Should steel production trajectories match exogenous estimates? `NULL` or one of

- `IEA_ETP` IEA 2017 Energy Transition Pathways steel production totals for OECD and Non-OECD countries from the *Reference Technologies Scenario* until 2060, and original growth rates after that.

`save.plots` `NULL` (default) if no plots are saved, or the path to save directories to.

`China_Production`
A data frame with columns `period` and `total.production` prescribing total production for China to have, disregarding results from the stock saturation model.

Value

A list with a `magpie` object `x`, `weight`, `unit`, `description`, `min`, and `max`.

Author(s)

Michaja Pehl

See Also

`calcOutput()`

`calcStorageFactor` *calc Capacity Factor*

Description

provides capacity factor values

Usage

```
calcStorageFactor()
```

Value

magpie object of the capacity factor data

Author(s)

Lavinia Baumstark

See Also

calcOutput

Examples

```
## Not run:
calcOutput("StorageFactor")

## End(Not run)
```

---

calcTaxConvergence *calc Tax Convergence*

Description

tax convergence levels for specific regions

Usage

```
calcTaxConvergence()
```

Value

magpie object of the tax convergence levels

Author(s)

Renato Rodrigues

Examples

```
## Not run:
calcOutput("TaxConvergence")

## End(Not run)
```

`calcTaxLimits` *calc Tax Limits*

Description

tax and subsidies maximum levels. The tax limits serve as a work around to avoid excess of subsidy levels that could cause problems on the REMIND model solution. These files should be removed or replaced once a better way to handle this issue is introduced to the REMIND model formulation or once better yearly and country subsidy level data is available for the primary and final energies.

Usage

```
calcTaxLimits(subtype)
```

Arguments

`subtype`	Name of the subsidy data type limit, e.g. "maxFeSubsidy" for maximum final energy subsidy,"maxPeSubsidy" for maximum primary energy subsidy or "propFeSubsidy" for proportional cap for final energy subsidy

Value

magpie object of the subtype tax limit

Author(s)

Renato Rodrigues

Examples

```
## Not run:
calcOutput("TaxLimits")

## End(Not run)
```

`calcTCdamage` *write TC damage parameters into input data they are country-specific and should not be aggregated to the regional level at all*

Description

write TC damage parameters into input data they are country-specific and should not be aggregated to the regional level at all

Usage

```
calcTCdamage(subtype)
```

Arguments

subtype	"const", "tasK"

Value

MAgPIE object of damage parameters for country level tropical cyclone damage function

Author(s)

Franziska Piontek

---

calcTheil *Calculate regional Theil-T index*

Description

To calculate the regional Theil-T index (= correction to welfare function for a lognormal income distribution) we do the following: (1) convert country-level Gini coefficients to Theil (2) calculate contribution to Theil-T index that includes both between-countries and within-country inequality (see e.g. https://en.wikipedia.org/wiki/Theil_index). The latter can then be aggregated with calcOutput().

Usage

```
calcTheil()
```

Details

NB 1: the aggregation depends on the region mapping. It is implemented such that the regionmapping specified in getConfig()$regionmapping is used.

NB 2: the result of calcOutput('Theil', aggregate = FALSE), is NOT the country Theil-T, but the unweighted contribution from a given country to the regional value.

Value

magpie objects of unweighted contribution to Theil, weights (= country shares of regional GDP)

Author(s)

Bjoern Soergel

See Also

calcOutput convertGini,readGini

Examples

```
## Not run:
  calcOutput("Theil")

## End(Not run)
```

---

calcTrade *Computes Trade variables based on latest IEA data available*

Description

Computes Trade variables based on latest IEA data available

Usage

```
calcTrade()
```

Value

a magclass object

---

calcTransportGDPshare *calc transport share in GDP*

Description

provides transport share in GDP to resize edge-t transportation purchase costs

Usage

```
calcTransportGDPshare()
```

Value

magpie object of transport shares in GDP

Author(s)

Renato Rodrigues

Examples

```
## Not run:
calcOutput("TransportGDPshare")

## End(Not run)
```

calcUBA — *Calculate REMIND final energy variables from historical UBA values*

Description

Calculate REMIND final energy variables from historical UBA values

Usage

```
calcUBA()
```

Value

A magpie object.

Author(s)

Falk Benke

calcUNFCCC — *Calculate REMIND emission variables from historical UNFCCC values*

Description

Calculate REMIND emission variables from historical UNFCCC values

Usage

```
calcUNFCCC()
```

Value

A magpie object.

Author(s)

Falk Benke, Pascal Weigmann

calcWasteEnergyUseShares *Calculate waste energy use shares based on IEA World Energy Balances*

Description

The output of this function is used in remind2 for reporting purposes.

Usage

```
calcWasteEnergyUseShares()
```

Author(s)

Robert Pietzcker, Falk Benke

calcWaterConsCoef *Calculate Water Consumption Coefficients*

Description

This function calculates REMIND input data on water consumption coefficients per electricity technology, using as initial information the Macknick (2011) data per electricity technology. The source data provide most required information but some assumptions on missing data are also made.

Usage

```
calcWaterConsCoef()
```

Value

MAgPIE object on water consumption coefficients per electricity technology

Author(s)

Ioanna Mouratiadou

See Also

calcOutput, readMacknickIntensities, calcWaterWithCoef

Examples

```
## Not run:
calcOutput("WaterConsCoef")

## End(Not run)
```

calcWaterWithCoef *Calculate Water Withdrawal Coefficients*

Description

This function calculates REMIND input data on water withdrawal coefficients per electricity technology, using as initial information the Macknick (2011) data per electricity technology. The source data provide most required information but some assumptions on missing data are also made.

Usage

```
calcWaterWithCoef()
```

Value

MAgPIE object on water withdrawal coefficients per elecricity technology

Author(s)

Ioanna Mouratiadou

See Also

calcOutput, readMacknickIntensities, calcWaterConsCoef

Examples

```
## Not run:
calcOutput("WaterWithCoeff")

## End(Not run)
```

convertADVANCE_WP2 *Convert ADVANCE WP2 Data*

Description

Convert ADVANCE WP2 Data

Usage

```
convertADVANCE_WP2(x, subtype)
```

Arguments

x	A magpie object returned by readADVANCE_WP2().
subtype	One of • clinker-to-cement-ratio for the clinker-to-cement ratios from figure 21 of Edelenbosch, O. *Enhancing the representation of energy demand developments in IAM models - A Modeling Guide for the Cement Industry* (2015) zotero://select/items/JP8X2QFK, which is extended from H12 regions to country level.

Value

A magpie object.

Author(s)

Michaja Pehl

See Also

readSource(), readADVANCE_WP2()

---

convertAGEB *Convert AGEB data*

Description

Convert AGEB data

Usage

```
convertAGEB(x)
```

Arguments

x	A magpie object returned from readAGEB().

Value

A magpie object.

Author(s)

Falk Benke

convertAriadneDB *Convert Ariadne DB data*

Description

convert Ariadne database data

Usage

```
convertAriadneDB(x)
```

Arguments

x	A [‘magpie‘][magclass::magclass] object returned from [‘readAriadneDB()‘].

Value

A [‘magpie‘][magclass::magclass] object.

Author(s)

Felix Schreyer

convertBGR *Converts BGR oil, gas, coal and uranium reserves data*

Description

Converts BGR oil, gas, coal and uranium reserves data

Usage

```
convertBGR(x, subtype)
```

Arguments

x	MAgPIE object to be converted
subtype	data subtype. Either "oil", "gas", "coal" or "uranium".

Value

A MAgPIE object containing BGR (Federal Institute for Geosciences and Natural Resources) country reserves disaggregated data of oil, gas, coal and uranium.

Author(s)

Renato Rodrigues

Examples

```
## Not run:
a <- convertBGR(x, subtype = "oil")

## End(Not run)
```

`convertBP` *Disaggregates and cleans BP data.*

Description

Disaggregates and cleans BP data.

Usage

```
convertBP(x, subtype)
```

Arguments

x	MAgPIE object to be converted
subtype	Either "Emission", "Capacity", "Generation", "Production", "Consumption", "Trade Oil", "Trade Gas", "Trade Coal" or "Price"

Value

A [‘magpie‘][magclass::magclass] object.

Author(s)

Aman Malik, Falk Benke

`convertDaviesCooling` *Convert Davies Cooling*

Description

Convert Davies (2013) data on on shares of cooling types using mapping from GCAM regions to ISO country level.

Usage

```
convertDaviesCooling(x)
```

Arguments

x	MAgPIE object containing DaviesCooling data region resolution

Value

MAgPIE object of the Davies (2013) data disaggregated to country level

Author(s)

Lavinia Baumstark, Ioanna Mouratiadou

See Also

readDaviesCooling

Examples

```
## Not run:  a <- convertDaviesCooling(x)
```

---

convertDylanAusGasCost

Converts Dylan's Australian gas cost to magpie

Description

Converts Dylan's Australian gas cost to magpie

Usage

```
convertDylanAusGasCost(x)
```

Arguments

x	MAgPIE object to be converted

Value

magpie object of the CEMO data

Author(s)

Felix Schreyer

See Also

readSource

convertEDGAR7Fgases *convertEDGAR7Fgases*

Description

convertEDGAR7Fgases

Usage

```
convertEDGAR7Fgases(x)
```

Arguments

x	magpie object to be converted

Author(s)

Gabriel Abrahao

convertEdgeBuildings *Convert EDGE Buildings data to data on ISO country level.*

Description

Convert EDGE Buildings data to data on ISO country level.

Usage

```
convertEdgeBuildings(x, subtype = "FE")
```

Arguments

x	MAgPIE object containing EDGE values at ISO country resolution
subtype	either FE or Floorspace

Value

EDGE data as MAgPIE object aggregated to country level

Author(s)

Antoine Levesque, Robin Hasse

convertEDGETransport *Convert EDGEtransport*

Description

Convert EDGEtransport

Usage

```
convertEDGETransport(x, subtype)
```

Arguments

x	MAgPIE object containing EDGE-T values in 21 region resolution
subtype	REMIND/iterative EDGE-T input data subtypes

Value

REMIND/iterative EDGE-T input data as MAgPIE object disaggregated to ISO level

Author(s)

Johanna Hoppe

convertEmber *Convert Ember data*

Description

Convert Ember data

Usage

```
convertEmber(x)
```

Arguments

x	A magpie object returned from readHRE().

Value

A magpie object.

Author(s)

Pascal Weigmann

convertEuropeanEnergyDatasheets *Convert European Energy Datasheets*

Description

Convert European Energy Datasheets

Usage

```
convertEuropeanEnergyDatasheets(x, subtype)
```

Arguments

x	European Energy Datasheets magpie object derived from readEuropeanEnergyDatasheets function
subtype	data subtype. Either "EU28" (data from June 20 including GBR) or "EU27" (latest data from August 23 without GBR)

Value

converted European Energy Datasheets magpie object

Author(s)

Renato Rodrigues and Atreya Shankar

Source

European Energy Datasheets public database https://energy.ec.europa.eu/data-and-analysis/eu-energy-statistical-pocketbook-and-country-datasheets_en

Examples

```
## Not run:
test <- readSource("EuropeanEnergyDatasheets", subtype = "EU27", convert = TRUE)

## End(Not run)
```

convertEurostat_EffortSharing *Converts EU Effort Sharing targets and historical emissions*

Description

Converts EU Effort Sharing targets and historical emissions

Usage

```
convertEurostat_EffortSharing(x, subtype)
```

Arguments

x	MAgPIE object to be converted
subtype	data subtype. Either "target" or "emissions"

Value

A MAgPIE object containing the EU Effort Sharing targets (

Author(s)

Renato Rodrigues

Examples

```
## Not run:  a <- convertEurostat_EffortSharing(x,subtype="target")
```

convertExpertGuess *convertExpertGuess*

Description

Converts data from expert guess

Usage

```
convertExpertGuess(x, subtype)
```

Arguments

x	unconverted magpie object from read-script
subtype	Type of data that are converted.

Value

magpie object with a completed dataset.

See Also

convertExpertGuess

---

convertGEA2012 *convertGEA2012*

Description

Converts oil, gas and coal data from the Global Energy Assessment 2012 to country-level aggregation

Usage

```
convertGEA2012(x, subtype)
```

Arguments

x	MAgPIE object to be disaggregated
subtype	Type of fossil fuel (oil, coal or gas)

Value

MAgPIE object containing country-level disaggregation of GEA 2012 data

Author(s)

Stephen Bi

See Also

readSource

Examples

```
## Not run:
a <- readSource("GEA2012")

## End(Not run)
```

convertGGDC10 *Convert GGDC 10-Sector Database - https://www.rug.nl/ggdc/structuralchange/previous-sector-database/10-sector-2014*

Description

Convert GGDC 10-Sector Database - https://www.rug.nl/ggdc/structuralchange/previous-sector-database/10-sector-2014

Usage

```
convertGGDC10(x)
```

Arguments

x	MAgPIE object to be converted

Value

A MAgPIE object containing GGDC disaggregated data

Author(s)

Renato Rodrigues

Examples

```
 ## Not run:
a <- convertGGDC10(x)

## End(Not run)
```

convertGini *Convert Gini*

Description

Converts Gini data from readGini() to ISO country level. Countries missing in the original data set will have their Gini set to zero (a very small number for numerical reasons to be precise). The original data range is 2011-2100 in one-year steps, here we extend it to 2000-2150 in 5-year steps. Values before (after) the original range are held fixed at 2011 (2100) levels. Gini values for the SDP scenario are taken from the SSP1 scenario

Usage

```
convertGini(x)
```

Arguments

x	MAgPIE object containing Gini data with World Bank codes, 2011-2100, in percent (range 0-100)

Value

MAgPIE object of the Gini data in ISO countries, range 0-1

Author(s)

Bjoern Soergel

See Also

readSource readGini

Examples

```
## Not run:
a <- convertGini(x)

## End(Not run)
```

---

convertGlobalCCSinstitute

Convert Global CCS Institute Project Database

Description

Convert Global CCS Institute Project Database

Usage

```
convertGlobalCCSinstitute(x, subtype = "08-09-2017")
```

Arguments

x	A magpie object returned by readGlobalCCSinstitute().
subtype	Project Database version to read, one of - '08-09-2017': Data apparently from June 2017. - '2023-11': Data from the Global Status of CCS 2023 report.

Value

A magpie object.

convertGlobalEnergyMonitor *Convert Global Energy Monitor data*

Description

Convert Global Energy Monitor data

Usage

```
convertGlobalEnergyMonitor(x)
```

Arguments

x	A magclass object returned from readGlobalEnergyMonitor().

Author(s)

Rahel Mandaroux, Falk Benke

convertHRE *Convert HRE data*

Description

Convert HRE data

Usage

```
convertHRE(x)
```

Arguments

x	A magpie object returned from readHRE().

Value

A magpie object.

Author(s)

Pascal Weigmann

convertIAEA *Nuclear data from world-nuclear.org*

Description

Data on currently operating and under-construction nuclear power plants, reactors planned and proposed, electricity generation from nuclear

Usage

```
convertIAEA(x)
```

Arguments

x	MAgPIE object to be converted

Author(s)

Christoph Bertram

convertIEA_CCUS *Convert IEA CCUS data*

Description

Convert IEA CCUS data

Usage

```
convertIEA_CCUS(x)
```

Arguments

x	A magclass object returned from readIEA_CCUS().

Value

A magclass object.

Author(s)

Anne Merfort, Falk Benke

convertIEA_ETP *Convert IEA ETP projections*

Description

Convert IEA ETP projections

Usage

```
convertIEA_ETP(x, subtype)
```

Arguments

x	IEA ETP projection magpie object derived from readIEA_ETP function
subtype	data subtype. Either "industry", "buildings", "summary", or "transport"

Author(s)

Falk Benke

convertIEA_EVOutlook *Convert IEA EV Outlook*

Description

Convert IEA EV Outlook

Usage

```
convertIEA_EVOutlook(x)
```

Arguments

x	a magclass object returned from ‘readIEA_EVOutlook()‘

Author(s)

Falk Benke

convertIEA_PVPS *Convert IEA PVPS data from REMIND regions to iso countries*

Description

maps to iso countries

Usage

```
convertIEA_PVPS(x, subtype)
```

Arguments

x	MAgPIE object to be converted
subtype	type of data read from IEA PVPS

Value

Magpie object with IEA PVPS investment cost per country

Author(s)

Felix Schreyer

convertIEA_REN *Reads the distributed solar pv capacity from IEA Renewables report (2019).*

Description

Reads the distributed solar pv capacity from IEA Renewables report (2019).

Usage

```
convertIEA_REN(x)
```

Arguments

x	input magpie object

Details

Capacity in GW. Distributed solar, defined in the IEA Renewables (2019), includes rooftop residential (0-10 kW, grid-connected), rooftop and ground-mounted commercial and industrial (10-1000kW, grid-connected), and off-grid (8W - 100 kW)

Value

magpie object with country-wise distributed solar pv capacity

Author(s)

Aman Malik

`convertIEA_WEO` *Converts IEA World Energy Outlook data*

Description

Converts IEA World Energy Outlook data

Usage

```
convertIEA_WEO(x, subtype)
```

Arguments

x	MAgPIE object to be converted
subtype	data subtype. Either "Capacity", "Generation", "Emissions", "Investment Costs", "O&M Costs" or "Efficiency"

Value

magpie object of the WEO data on generation (TWh), capacities (GW), emissions (Mt CO2) or disaggregated investment cost as magpie object

Author(s)

Renato Rodrigues and Aman Malik

See Also

`readSource`

Examples

```
## Not run:
a <- convertWEO(x, subtype = "Capacity")

## End(Not run)
```

convertIEA_WorldEnergyOutlook *Convert IEA World Energy Outlook Data from 2023*

Description

Convert IEA World Energy Outlook Data from 2023

Usage

```
convertIEA_WorldEnergyOutlook(x)
```

Arguments

x	magclass object to be converted

Author(s)

Falk Benke

convertIIASA_subs_taxes *Convert IIASA_subs_taxes data*

Description

Convert IIASA subsidy and taxes data on ISO country level (removes countries not part of 249 oficial ISO countries and fills missing with zeros).

Usage

```
convertIIASA_subs_taxes(x)
```

Arguments

x	MAgPIE object containing IIASA subsidies and taxes data in country resolution

Value

IIASA_subs_taxes data as MAgPIE object aggregated to country level

Author(s)

Christoph Bertram

Examples

```
## Not run:  a <- convertIIASA_subs_taxes(x)
```

---

convertIRENA *Converts IRENA Regional data*

Description

Converts IRENA Regional data

Usage

```
convertIRENA(x, subtype)
```

Arguments

x	MAgPIE object to be converted
subtype	data subtype. Either "Capacity" or "Generation"

Value

A MAgPIE object containing IRENA country disaggregated data with historical electricity renewable capacities (MW) or generation levels (GWh)

Author(s)

Renato Rodrigues

Examples

```
## Not run:
a <- convertIRENA(x, subtype = "Capacity")

## End(Not run)
```

`convertKLWdamage` *convert KLW damage fills in countries for which no damage parameters are provided, setting parameters to zero*

Description

convert KLW damage fills in countries for which no damage parameters are provided, setting parameters to zero

Usage

```
convertKLWdamage(x)
```

Arguments

x	is MAgPIE object containing the damage parameters from KLW

Value

MAgPIE object containing values for all 249 ISO countries

Author(s)

Franziska Piontek

`convertMueller` *Convert Mueller data*

Description

Convert Mueller data

Usage

```
convertMueller(x, subtype)
```

Arguments

x	A `magpie` object returned from `readMueller()`.
subtype	One of:

- `countries`: read table mapping country names use by Müller et al. 2013 to ISO 3166-1 alpha-3 codes.
- `stocks`: read low/medium/high estimates of per-capita steel stocks from Müller et al. 2013 SI2

Value

A magpie object.

Author(s)

Falk Benke

---

convertnonEnergyDemand *Converts Final energy demand for feedstocks (non-energy use)*

Description

Converts Final energy demand for feedstocks (non-energy use)

Usage

```
convertnonEnergyDemand(x)
```

Arguments

x	MAgPIE object to be converted

Value

A MAgPIE object containing country disaggregated data

Author(s)

Renato Rodrigues

Examples

```
## Not run:
a <- convertnonEnergyDemand(x)

## End(Not run)
```

convertNREL *Convert NREL data*

Description

Convert NREL data on ISO country level.

Usage

```
convertNREL(x)
```

Arguments

x	MAgPIE object containing NREL data country-region resolution

Value

NRELWirsenius data as MAgPIE object aggregated to country level

Author(s)

Lavinia Baumstark

Examples

```
## Not run:
a <- convertNREL(x, subtype = "onshore")

## End(Not run)
```

convertOpenmod *Converts Openmod capacities data*

Description

Converts Openmod capacities data

Usage

```
convertOpenmod(x)
```

Arguments

x	MAgPIE object to be converted

Value

A MAgPIE object containing openmod EU country disaggregated data with 2010 and 2015 electricity capacities (GW)

Author(s)

Renato Rodrigues

Examples

```
## Not run:
a <- convertOpenmod(x)

## End(Not run)
```

`convertProdShares` *Reads shares of world manufacture for spv modules and wind turbines.*

Description

Reads shares of world manufacture for spv modules and wind turbines.

Usage

```
convertProdShares(x)
```

Arguments

x	input magpie object

Value

magpie object with shares

Author(s)

Aman Malik

convertRCP *convertRCP convert RCP data*

Description

convertRCP convert RCP data

Usage

```
convertRCP(x, subtype)
```

Arguments

x	Input object obtained by readSource
subtype	Either 'Waste' or 'AviationShipping'

Value

magpie object of the RCP data

Author(s)

Julian Oeser

convertRemindCesPrices *Convert RemindCesPrices*

Description

Converts CES derivatives/prices from former REMIND runs to ISO level

Usage

```
convertRemindCesPrices(x, subtype = "ccd632d33a")
```

Arguments

x	MAgPIE object containing REMIND prices at the REMIND region resolution
subtype	Regional resolution of REMIND data which should be loaded. ccd632d33a corresponds to the REMIND-11, and 690d3718e1 to REMIND-H12

Value

magpie object of REMIND prices

Author(s)

Antoine Levesque

See Also

readSource

---

convertREMIND_11Regi *Converts REMIND regional data*

Description

Converts REMIND regional data

Usage

```
convertREMIND_11Regi(x, subtype)
```

Arguments

x	MAgPIE object to be converted
subtype	Name of the regional data, e.g. "p4", "biomass", "ch4waste", "tradecost", "pe2se", "xpres_tax", "deltacapoffset", capacityFactorRules", "taxConvergence", "maxFeSubsidy", "maxPeSubsidy", "propFeSubsidy", "fossilExtractionCoeff", "uraniumExtractionCoeff", "RLDCCoefficientsLoB", "RLDCCoefficientsPeak", "earlyRetirementAdjFactor"

Value

A MAgPIE object containing country disaggregated data

Author(s)

original: not defined - capacity factor, tax, fossil and RLDC changes: Renato Rodrigues

Examples

```
## Not run:  a <- convertREMIND_11Regi(x,subtype="capacityFactorGlobal")
```

convertREN21 *Policy targets for REN21*

Description

This code aggregates and homogenises different types of renewable energy targets into total installed capacity targets (in GW).

Usage

```
convertREN21(x, subtype)
```

Arguments

x	MAgPIE object to be converted
subtype	Only "Capacity" asof now

Details

Policy database accessible in "inputdata/sources/REN21/README"

Value

Magpie object with Total Installed Capacity targets. The target years differ depending upon the database.

Author(s)

Aman Malik

convertStationary *Convert Stationary data to data on ISO country level.*

Description

Convert Stationary data to data on ISO country level.

Usage

```
convertStationary(x)
```

Arguments

x	MAgPIE object to be converted

Author(s)

Antoine Levesque, Robin Hasse

convertStegmann2022 *convertStegmann2022*

Description

Converts data from Stegmann2022

Usage

```
convertStegmann2022(x)
```

Arguments

x	unconverted magpie object from read-script

Value

magpie object with a completed dataset.

convertStrefler *convertStrefler*

Description

Converts data on enhanced weathering

Usage

```
convertStrefler(x, subtype)
```

Arguments

x	unconverted magpie object from read-script
subtype	data subtype. Either "weathering_graderegi", or "weathering_costs"

Value

magpie object with a completed dataset

convertTCdamageKrichene

Convert TCdamage fills in countries not affected by tropical cyclones (TC), setting parameters to zero

Description

Convert TCdamage fills in countries not affected by tropical cyclones (TC), setting parameters to zero

Usage

```
convertTCdamageKrichene(x)
```

Arguments

x	is MAgPIE object containing the damage parameters for the TC-prone countries

Value

MAgPIE object containing values for all 249 ISO countries

Author(s)

Franziska Piontek

convertTransportSubsidies

Converts transport subsidies data

Description

Converts transport subsidies data

Usage

```
convertTransportSubsidies(x)
```

Arguments

x	MAgPIE object to be converted

Value

A MAgPIE object containing transport subsidies per technology

Author(s)

Renato Rodrigues

Examples

```
## Not run:
a <- convertTransportSubsidies(x)

## End(Not run)
```

---

convertUBA *Convert UBA data*

Description

Convert UBA data

Usage

```
convertUBA(x)
```

Arguments

x	A magpie object returned from readUBA().

Value

A magpie object.

Author(s)

Falk Benke

---

convertUNFCCC *Convert UNFCCC data*

Description

Convert UNFCCC data

Usage

```
convertUNFCCC(x)
```

Arguments

x	A magpie object returned from readUNFCCC().

Value

A magpie object.

Author(s)

Falk Benke

---

convertUNFCCC_NDC *Policy targets for NDCs from UNFCCC_NDC*

Description

Converts conditional and unconditional capacity targets into total capacity (GW) in target year the Generation targets are similar to the capacity targets but include the capacity factors, the Emissions targets are the total (except land CO2) emissions in the target year

Usage

```
convertUNFCCC_NDC(x, subtype)
```

Arguments

x	MAgPIE object to be converted
subtype	Capacity_YYYY_cond or Capacity_YYYY_uncond for Capacity Targets, Emissions_YYYY_cond or Emissions_YYYY_uncond for Emissions targets, with YYYY NDC version year

Value

Magpie object with Total Installed Capacity (GW) targets, target years differ depending upon the database.

Author(s)

Aman Malik, Christoph Bertram, Oliver Richters

convertWGBU *Convert WGBU data*

Description

Convert WGBU data on ISO country level.

Usage

```
convertWGBU(x)
```

Arguments

x	MAgPIE object containing WGBU data country-region resolution

Value

WGBU data as MAgPIE object aggregated to country level

Author(s)

Lavinia Baumstark

Examples

```
## Not run:
a <- convertWGBU(x)

## End(Not run)
```

exportThresholds *export validation thresholds*

Description

assemble near-term thresholds from project pipelines and potentially other data sources and export them to a file

Usage

```
exportThresholds(type = "config", years = NULL)
```

Arguments

type	choose either "config" to export thresholds as used in the validationConfig or "full" to export all pipeline data
years	choose years to include, currenty only 2025 and 2030 are available, if NULL uses all available years

Author(s)

Pascal Weigmann

---

filter_historical_mif *Filter improperly aggregated regional data from historical.mif*

Description

Load a `historical.mif` file, remove designated data, write back to file.

Usage

```
filter_historical_mif(path = NULL, filter_table = NULL)
```

Arguments

path	Path to `historical.mif` (or any mif for that matter), defaults to `historical.mif` in the currently configured `getConfig('outputfolder')`.
filter_table	Data frame with columns `scenario`, `model`, `variable`, `include_regions`, and `exclude_regions`. Defaults to the built-in `./inst/extdata/historical_mif_filter_table.csv` of the `mrremind` package.

Details

For all combinations of scenario, model, and variable (and all years), either all regions listed in `exclude_regions` in the `filter_table` are removed, or all regions *except* those listed in `include_regions` in the `filter_table` are removed. Defining both `include_regions` and `exclude_regions` on the same row of `filter_table` will throw an error. Scenarios, models, and variables are matched precisely, not partially or via regular expressions.

fullDECENT *fullDECENT*

Description

Function that produces the complete regional data set required for the DECENT model.

Usage

```
fullDECENT(rev = 0)
```

Arguments

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

Author(s)

Lavinia Baumstark, Lukas Feldhaus

See Also

readSource,getCalculations,calcOutput

Examples

```
## Not run:
fullDECENT()

## End(Not run)
```

fullREMIND *fullREMIND*

Description

Function that produces the complete regional data set required for the REMIND model.

Usage

```
fullREMIND()
```

Author(s)

Lavinia Baumstark

See Also

readSource,getCalculations,calcOutput

Examples

```
## Not run:
fullREMIND()

## End(Not run)
```

---

fullVALIDATIONREMIND *Generate Validation Data for REMIND*

Description

Function that generates the historical regional dataset against which the REMIND model results can be compared.

Usage

```
fullVALIDATIONREMIND(rev = 0)
```

Arguments

rev	Unused parameter here for the pleasure of madrat.

Author(s)

David Klein, Falk Benke

See Also

fullREMIND(), readSource(), getCalculations(), calcOutput()

Examples

```
## Not run:
fullVALIDATIONREMIND()

## End(Not run)
```

readADVANCE_WP2 *Read ADVANCE WP2 Data*

Description

Read ADVANCE WP2 Data

Usage

```
readADVANCE_WP2(subtype)
```

Arguments

subtype	One of • `clinker-to-cement-ratio` for the clinker-to-cement ratios from figure 21 of Edelenbosch, O. *Enhancing the representation of energy demand developments in IAM models - A Modeling Guide for the Cement Industry* (2015) zotero://select/items/JP8X2QFK

Value

A magpie object.

Author(s)

Michaja Pehl

See Also

readSource(), convertADVANCE_WP2()

readAGEB *Read AGEB*

Description

Read AGEB

Usage

```
readAGEB(subtype = "balances")
```

Arguments

subtype	data subtype. Either "balances" ("Auswertungstabellen zur Energiebilanz Deutschland") or "electricity" ("Bruttostromerzeugung in Deutschland nach Energieträgern")

Value

A magpie object.

Author(s)

Falk Benke

---

readAR6GWP *Read GWP (or other metrics) from the AR6 WGIII Table SM7 per GHG species*

Description

Read GWP (or other metrics) from the AR6 WGIII Table SM7 per GHG species

Usage

```
readAR6GWP(subtype = "GWP100")
```

Arguments

subtype	data subtype. Currently just "GWP100", but other metrics are also available in the input data

Value

A data.frame with two columns, "Gas", with the common name of the GHG species, and "GWP", with the selected GWP

Author(s)

Gabriel Abrahao

---

readAriadneDB *Ariadne database scenario data*

Description

Scenario data from the Ariadne modeling intercomparison project for Germany. See README in input file for more details.

Usage

```
readAriadneDB()
```

Value

A [‘magpie‘][magclass::magclass] object.

Author(s)

Felix Schreyer

`readBGR` *Read BGR oil, gas, coal and uranium reserves data*

Description

Read-in BGR csv files as magclass object

Usage

```
readBGR(subtype)
```

Arguments

`subtype`	data subtype. Either "oil", "gas", "coal" or "uranium".

Value

magpie object of the BGR (Federal Institute for Geosciences and Natural Resources) data of reserves of oil, gas, coal and uranium per country.

Author(s)

Renato Rodrigues

See Also

`readSource`

Examples

```
## Not run:
a <- readSource(type = "BGR", subtype = "oil")

## End(Not run)
```

readBP *BP Capacity and Generation Data*

Description

BP Capacity and Generation Data

Usage

```
readBP(subtype)
```

Arguments

subtype	Either "Emission", Capacity", "Generation", "Production", "Consumption", "Trade Oil", "Trade Gas", "Trade Coal" or "Price"

Value

A [‘magpie‘][magclass::magclass] object.

Author(s)

Aman Malik, Falk Benke

readCEEW *Read Employment factors and cumulative jobs for RE techs (for India)*

Description

Read Employment factors and cumulative jobs for RE techs (for India)

Usage

```
readCEEW(subtype)
```

Arguments

subtype	data subtype. Either "Employment factors" or "Employment"

Details

Reports published by CEEW et al. See README.txt in the source folder for more information.

Author(s)

Aman Malik

Examples

```
 ## Not run:
a <- readSource("CEEW",convert=F,subtype="Employment")

## End(Not run)
```

`readDaviesCooling` *Read Davies Cooling*

Description

Read in Davies (2013) data on shares of cooling types per electricity technology and GCAM region

Usage

```
readDaviesCooling(subtype)
```

Arguments

`subtype`	Type of Davies data that should be read. Available types are: • `dataBase`: The Davies source data for the base year • `dataFuture`: The Daves source data for the future

Value

MAgPIE object of the Davies (2013) data

Author(s)

Lavinia Baumstark, Ioanna Mouratiadou

See Also

`readSource`

Examples

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

## End(Not run)
```

readDylanAusGasCost *read-in power Australian gas extraction cost curve based on Dylan's data Australian contact: Dylan McConnell, dylan.mcconnell(at)unimelb.edu.au*

Description

read-in power Australian gas extraction cost curve based on Dylan's data Australian contact: Dylan McConnell, dylan.mcconnell(at)unimelb.edu.au

Usage

```
readDylanAusGasCost()
```

Value

magpie object of the cemo database data

Author(s)

Felix Schreyer

See Also

readSource

readEDGAR7Fgases *Read EDGAR7 emissions data for F-gases per species, in kt of each gas*

Description

Read EDGAR7 emissions data for F-gases per species, in kt of each gas

Usage

```
readEDGAR7Fgases()
```

Value

A magpie object with F-gases emissions per gas species and per country

Author(s)

Gabriel Abrahao

readEdgeBuildings *Load an EDGE Buildings file as magclass object.*

Description

Load an EDGE Buildings file as magclass object.

Usage

```
readEdgeBuildings(subtype = c("FE", "Floorspace"))
```

Arguments

subtype	One of the possible subtypes, see default argument.

Value

magclass object

Author(s)

Antoine Levesque, Robin Hasse

readEDGETransport *Read REMIND/EDGE-T iterative input data*

Description

Run EDGE-Transport Standalone in all used scenario combinations to supply input data to REMIND and the iterative EDGE-T script

Usage

```
readEDGETransport(subtype)
```

Arguments

subtype	REMIND/iterative EDGE-T input data subtypes

Value

magpie object of EDGEtransport iterative inputs

Author(s)

Johanna Hoppe

See Also

readSource

Examples

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

`readEmber` *Read Ember Yearly Electricity Data*

Description

Read Ember Yearly Electricity Data

Usage

```
readEmber()
```

Value

A [‘magpie‘][magclass::magclass] object.

Author(s)

Pascal Weigmann

Source

https://ember-climate.org/data-catalogue/yearly-electricity-data/

See Also

[‘readSource()‘]

readEuropeanEnergyDatasheets *Read European Energy Datasheets*

Description

Read European Energy Datasheets .xlsx file as magpie object.

Usage

```
readEuropeanEnergyDatasheets(subtype)
```

Arguments

subtype	data subtype. Either "EU28" (data from June 20 including GBR) or "EU27" (latest data from August 23 without GBR)

Value

magpie object of European Energy Datasheets

Author(s)

Renato Rodrigues, Atreya Shankar, Falk Benke

Source

European Energy Datasheets public database https://energy.ec.europa.eu/data-and-analysis/eu-energy-statistical-pocketbook-and-country-datasheets_en

Examples

```
## Not run:
test <- readSource("EuropeanEnergyDatasheet", subtype = "EU27", convert = FALSE)

## End(Not run)
```

readEurostat_EffortSharing *Read EU Effort Sharing targets and historical emissions*

Description

Read-in EU Effort Sharing targets and historical emissions csv files as magclass object

Usage

```
readEurostat_EffortSharing(subtype)
```

Arguments

subtype	data subtype. Either "target" or "emissions"

Value

magpie object of the EU Effort Sharing targets (

Author(s)

Renato Rodrigues

See Also

readSource

Examples

```
## Not run:
a <- readSource(type = "Eurostat_EffortSharing", subtype = "target")

## End(Not run)
```

readExpertGuess *Read ExpertGuess*

Description

Read-in data that are based on expert guess

Usage

```
readExpertGuess(subtype)
```

Arguments

subtype — Type of data that should be read. One of

- `Chinese_Steel_Production`: "Smooth" production estimates by Robert Pietzcker (2022).
- `industry_max_secondary_steel_share`: Maximum share of secondary steel production in total steel production and years between which a linear convergence from historic to target shares is to be applied.
- `cement_production_convergence_parameters`: convergence year and level (relative to global average) to which per-capita cement demand converges
- `ies`
- `prtp`
- `CCSbounds`
- `costsTradePeFinancial`
- `tradeContsraints`: parameter by Nicolas Bauer (2024) for the region specific trade constraints, values different to 1 activate constraints and the value is used as effectiveness to varying degrees such as percentage numbers

Value

magpie object of the data

Author(s)

Lavinia Baumstark

See Also

`readSource`

Examples

```
## Not run:
a <- readSource(type = "ExpertGuess", subtype = "ies")

## End(Not run)
```

---

readGCPT — *Data from the Global Coal Plant Tracker January 2021 release by Global Energy Monitor (formerly EndCoal/CoalSwarm)*

Description

Historical data of operating, under-construction, planned and announced Coal Plants by country (in MW) from the Global Energy Monitor's Global Coal Plant Tracker, and extrapolations for 2025 capacity scenarios

Usage

```
readGCPT(subtype)
```

Arguments

subtype	Options are status, historical, future, lifespans, comp_rates and emissions

Author(s)

Stephen Bi

---

readGEA2012 *read GEA 2012*

Description

Read in datafiles comprising fossil fuel data from the Global Energy Assessment 2012

Usage

```
readGEA2012(subtype)
```

Arguments

subtype	Type of fossil fuel and type of data (oil, coal, or gas + costs, qtys, or dec)

Value

MAgPIE object of the GEA data

Author(s)

Stephen Bi

See Also

readSource

Examples

```
## Not run:
a <- readSource("GEA2012", "coal")

## End(Not run)
```

readGGDC10 *Read GGDC 10-Sector Database - https://www.rug.nl/ggdc/structuralchange/previous-sector-database/10-sector-2014*

Description

Read GGDC 10-Sector Database - https://www.rug.nl/ggdc/structuralchange/previous-sector-database/10-sector-2014

Usage

```
readGGDC10()
```

Author(s)

Renato Rodrigues

Examples

```
 ## Not run:
a <- readSource("GGDC10",convert=F)

## End(Not run)
```

readGini *Read Gini*

Description

Read Gini coefficients for SSP scenarios from Rao et al., Futures, 2018. Data has been provided by the authors, but will be made publicly available as well. This contains data for 184 countries and from 2011 onwards.

Usage

```
readGini()
```

Details

Copied from the documentation provided by the authors: This sheet contains the original Gini projections for 43 countries from the underlying empirical model (See reference to RSP 2016 in the main paper) and the extrapolations to all countries using the methodology described in the article. The country codes are the World Bank codes.

Value

magpie object of the Gini data

Author(s)

Bjoern Soergel

See Also

readSource convertGini

Examples

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

---

readGlobalCCSinstitute

Read Global CCS Institute Project Database

Description

Read Global CCS Institute Project Database

Usage

```
readGlobalCCSinstitute(subtype = "08-09-2017")
```

Arguments

subtype	Project Database version to read, one of - '08-09-2017': Data apparently from June 2017. - '2023-11': Data from the Global Status of CCS 2023 report.

Value

A magpie object.

readGlobalEnergyMonitor *Read Global Energy Monitor data*

Description

read GEM data for all available technologies and relevant statuses

Usage

```
readGlobalEnergyMonitor()
```

Author(s)

Rahel Mandaroux, Falk Benke, Pascal Weigmann

readHRE *Read Heat Roadmap Europe data*

Description

Read Heat Roadmap Europe data

Usage

```
readHRE()
```

Value

A [‘magpie‘][magclass::magclass] object.

Author(s)

Pascal Weigmann

Source

https://heatroadmap.eu/roadmaps/

See Also

[‘readSource()‘]

readIAEA *Nuclear data from world-nuclear.org*

Description

Data on currently operating and under-construction nuclear power plants, reactors planned and proposed, electricity generation from nuclear

Usage

```
readIAEA()
```

Author(s)

Christoph Bertram

readIEA_CCUS *Read IEA CCUS data*

Description

Reads in capacities from projects in IEA CCUS database

Usage

```
readIEA_CCUS(subtype)
```

Arguments

subtype	either ‘historical‘ for data until 2023, ‘projections‘ for "high" and "low" projections up to 2030 used as input-data or ‘pipeline‘ separated by status for use in formulating near-term bounds

Author(s)

Anne Merfort, Falk Benke

readIEA_ETP *Read IEA ETP projections*

Description

Read IEA ETP projections

Usage

```
readIEA_ETP(subtype)
```

Arguments

subtype	data subtype. Either "industry", "buildings", "summary", or "transport"

Author(s)

Falk Benke

readIEA_EVOutlook *Read IEA EV Outlook*

Description

Read IEA EV Outlook

Usage

```
readIEA_EVOutlook()
```

Author(s)

Falk Benke

readIEA_PVPS *Reads PV investment cost data for 2020 which are based on 2018 data from IEA PVPS report*

Description

reads excel sheet with PV investment cost data

Usage

```
readIEA_PVPS(subtype)
```

Arguments

subtype	type of data read from IEA PVPS

Value

magpie object with PV investment cost data

Author(s)

Felix Schreyer

readIEA_REN *Reads the distributed solar pv capacity from IEA Renewables report (2019).*

Description

Reads the distributed solar pv capacity from IEA Renewables report (2019).

Usage

```
readIEA_REN()
```

Details

Capacity in GW. Distributed solar, defined in the IEA Renewables (2019), includes rooftop residential (0-10 kW, grid-connected), tooftop and ground-mounted commercial and industrial (10-1000kW, grid-connected), and off-grid (8W - 100 kW)

Author(s)

Aman Malik

readIEA_WEIO_2014 *IEA World Energy Investment Outlook (2014)*

Description

Read projected 2014-20 investments into industry energy efficiency from the [IEA World Energy Investment Outlook (2014)](http://www.iea.org/publications/freepublications/publication/weo-2014-special-report—investment.html)

Usage

```
readIEA_WEIO_2014()
```

Value

A [madrat_mule()] with a list containing the [tibble] ‘data‘ with 2014–20 average annual investments into ‘Energy intensive‘ and ‘Non-energy intensive‘ industry, in $bn 2012, and the [tibble] ‘country_groups‘ with ‘IEA region‘s and corresponding ‘iso3c‘ country codes.

readIEA_WEO *Read IEA World Energy Outlook data*

Description

Read-in IEA WEO 2016 data for investment costs, O&M costs and Efficiency of different technologies, and WEO 2017 data for historical electricity capacities (GW), generation (TWh) or emissions (Mt CO2). WEO 2019 data for PE and FE (Mtoe).

Usage

```
readIEA_WEO(subtype)
```

Arguments

subtype	data subtype. Either "Capacity", "Generation", "Emissions", "Investment Costs", "O&M Costs" or "Efficiency"

Value

magpie object of the WEO data on generation (TWh), capacities (GW), emissions (Mt CO2) or disaggregated investment cost as magpie object

Author(s)

Renato Rodrigues, Aman Malik, and Jerome Hilaire

See Also

readSource

Examples

```
## Not run:
a <- readSource(type = "WEO", subtype = "Capacity")

## End(Not run)
```

`readIEA_WorldEnergyOutlook` *Read in IEA World Energy Outlook Data from 2023*

Description

Read in IEA World Energy Outlook Data from 2023

Usage

```
readIEA_WorldEnergyOutlook()
```

Author(s)

Falk Benke

`readIIASA_subs_taxes` *Read IIASA subsidies and taxes*

Description

Read-in country level data on final energy taxes and subsidies as provided from IIASA from .csv file as magclass object

Usage

```
readIIASA_subs_taxes(subtype)
```

Arguments

`subtype`	Type of country level data as compiled by IIASA that should be read in. Available types are: • `tax_rate`: tax rate pre final energy category • `sub_rate`: subsidy rate per final energy category • `energy`: final energy quantities per category

Value

magpie object of the IIASA_subs_taxes data

Author(s)

Christoph Bertram

See Also

readSource

Examples

```
## Not run:
a <- readSource(type = "IIASA_subs_taxes", "tax_rate")

## End(Not run)
```

---

readindustry_subsectors_specific

industry/subsector change factors

Description

Change factors of specific FE and material demand for the industry/subsector realisation of REMIND.

Usage

```
readindustry_subsectors_specific(subtype = NULL)

calcindustry_subsectors_specific(
  subtype = NULL,
  scenarios = NULL,
  regions = NULL,
  direct = NULL
)
```

Arguments

subtype	One of • FE for specific final energy demand change factors • material_alpha for alpha factors and convergence time of specific material demand decreases relative to the SSP2EU scenario • material_relative for scaling factors of specific material demand relative to baseline scenarios

- material_relative_change for scaling factors of specific material demand *change* relative to baseline scenarios

scenarios	A vector of scenarios for which factors are to be returned.
regions	A vector of regions for which factors are to be returned.
direct	A data frame as returned by readindustry_subsectors_specific() to load debugging/developing data directly instead of from file.

Details

Factors are read from the files specific_FE.csv, specific_material_alpha.csv, specific_material_relative.csv, and specific_material_relative_change.csv, respectively. NA is used to mark defaults for the scenario and region columns, and specified values will overwrite these defaults.

So

- NA,NA,cement,1 will be extended to all scenarios and regions
- scen1,NA,cement,2 will overwrite this default for all regions in scen1
- NA,regi1,cement,3 will overwrite this again for all scenarios (including scen1) for regi1
- scen1,regi1,cement,4 will lastly overwrite the value for the scen1, regi1 combination

Replacements occure in this fixed order (NA/NA, scenario/NA, NA/region, scenario/region).

Lastly, output is filtered for scenarios and regions.

For debugging and development, instead of modifying the .csv files in sources/industry_subsectors_specific/ and interfering with production runs, modify the calling code (e.g. calcFEdemand.R) to use direct data (entered verbatim or loaded from somewhere else.)

Value

A magpie object.

Author(s)

Michaja Pehl

---

readINNOPATHS *Read INNOPATHS*

Description

Read INNOPATHS

Usage

```
readINNOPATHS()
```

readIRENA *Read IRENA*

Description

Read-in an IRENA csv file as magclass object

Usage

```
readIRENA(subtype)
```

Arguments

subtype	data subtype. Either "Capacity" or "Generation"

Value

magpie object of the IRENA data with historical electricity renewable capacities (MW) or generation levels (GWh)

Author(s)

Renato Rodrigues

See Also

readSource

Examples

```
## Not run:
a <- readSource(type = "IRENA", subtype = "Capacity")

## End(Not run)
```

readKLWdamage *Reads country-specific damage coefficients for the damage function presented in Kotz et al. (2024). Data has been provided by the authors. This contains data for all countries and for 1000 boostrapping realizations per country, capturing uncertainty from climate and empirical modeling. Subtypes are the temperature and temperature^2 coefficients and the maximum temperature per country for which the function is defined.*

Description

Reads country-specific damage coefficients for the damage function presented in Kotz et al. (2024). Data has been provided by the authors. This contains data for all countries and for 1000 boostrapping realizations per country, capturing uncertainty from climate and empirical modeling. Subtypes are the temperature and temperature^2 coefficients and the maximum temperature per country for which the function is defined.

Usage

```
readKLWdamage(subtype)
```

Arguments

subtype	data subtype. Either "beta1", "beta2" or "maxGMT"

Value

KLW damage coefficients

Author(s)

Franziska Piontek

`readLee` *readLee Read in Aviation emission data from Lee*

Description

readLee Read in Aviation emission data from Lee

Usage

```
readLee(subtype)
```

Arguments

subtype	Either 'emi' or 'ef'

Value

magpie object of Aviation emission / emission factors data

Author(s)

Julian Oeser

See Also

readSource

Examples

```
## Not run:
a <- readSource(type = "", subtype = "Waste")

## End(Not run)
```

`readMacknickIntensities` *Read Macknic Intensities*

Description

Read in Macknick (2011) data on water consumption and withdrawal coefficients per electricity technology

Usage

```
readMacknickIntensities(subtype)
```

Arguments

`subtype`	Type of Macknick data that should be read. Available types are: • `data`: The original Macknick source data • `missingAssumed`: Additional data to fill gaps

Value

MAgPIE object of the Macknick (2011) data

Author(s)

Ioanna Mouratiadou

See Also

`readSource`

Examples

```
## Not run:
a <- readSource(type = "MacknickIntensities", convert = FALSE)

## End(Not run)
```

readMueller *Read Müller et al. 2013 data.*

Description

Read data from Müller et al. 2013 (http://dx.doi.org/10.1021/es402618m).

Usage

```
readMueller(subtype)
```

Arguments

subtype	One of: • countries: read table mapping country names use by Müller et al. 2013 to ISO 3166-1 alpha-3 codes. • stocks: read low/medium/high estimates of per-capita steel stocks from Müller et al. 2013 SI2

Value

A magpie object.

Author(s)

Michaja Pehl

See Also

readSource()

readnonEnergyDemand *Read Final energy demand for feedstocks (non-energy use)*

Description

Read Final energy demand for feedstocks (non-energy use)

Usage

```
readnonEnergyDemand()
```

Value

magpie object of region dependent data

Author(s)

Renato Rodrigues

See Also

readSource

Examples

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

## End(Not run)
```

---

readNREL *Read NREL*

Description

Read-in NREL xlsx file as magclass object

Usage

```
readNREL(subtype)
```

Arguments

subtype	type either "onshore" or "offshore"

Value

magpie object of NREL

Author(s)

Lavinia Baumstark

See Also

readSource

Examples

```
## Not run:
a <- readSource(type = "NREL", subtype = "onshore")

## End(Not run)
```

readODYM_RECC *Read ODYM_RECC data from the SHAPE Project*

Description

Read ODYM_RECC data from the SHAPE Project

Usage

```
readODYM_RECC(subtype, smooth = TRUE)

calcODYM_RECC(subtype, smooth = TRUE)
```

Arguments

subtype	One of • 'REMIND_industry_trends': Trends in per-capita production of industry subsectors cement, chemicals, steel_primary, steel_secondary, and otherInd. Trends for chemicals and otherInd are averages of the other three trends, which are provided by NTNU.
smooth	Smooth REMIND_industry_trends (default) or not.

Value

A magpie object.

Author(s)

Michaja Pehl

readOECD *Read OECD*

Description

Read-in risk premium

Usage

```
readOECD()

convertOECD(x)
```

Arguments

x	MAgPIE object returned from readOECD

Value

The read-in data, usually a magpie object. If supplementary is TRUE a list including the data and metadata is returned instead. The temporal and data dimensionality should match the source data. The spatial dimension should either match the source data or, if the convert argument is set to TRUE, should be on ISO code country level.

See Also

[madrat::readSource()]

Examples

```
## Not run:
readSource("OECD")

## End(Not run)
```

`readOpenmod` *Read Openmod capacities data*

Description

Read-in an modified openmod capacities data file as magclass object

Usage

```
readOpenmod()
```

Value

magpie object of the LIMES team updated Openmod data on capacities (GW)

See Also

`readSource`

Examples

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

## End(Not run)
```

readPauliuk *Read Pauliuk et al. 2013 data*

Description

Read data from Pauliuk et al. 2013 (https://dx.doi.org/10.1016/j.resconrec.2012.11.008).

Usage

```
readPauliuk(subtype = "lifetime")
```

Arguments

subtype	One of: • `lifetime`: Read estimated lifetime of overall steel stocks (approach b) in years.

Value

A `magpie` object.

Author(s)

Michaja Pehl

See Also

`readSource()`

readProdShares *Reads shares of world manufacture for spv modules and wind turbines.*

Description

Reads shares of world manufacture for spv modules and wind turbines.

Usage

```
readProdShares()
```

Author(s)

Aman Malik

readPWT *Read PWT*

Description

Read-in PWT data as magclass object

Usage

```
readPWT()

convertPWT(x)
```

Arguments

x	MAgPIE object returned by readPWT

Value

The read-in data, usually a magpie object. If supplementary is TRUE a list including the data and metadata is returned instead. The temporal and data dimensionality should match the source data. The spatial dimension should either match the source data or, if the convert argument is set to TRUE, should be on ISO code country level.

See Also

[madrat::readSource()]

Examples

```
## Not run:
readSource("PWT")

## End(Not run)
```

readRCP *Read RCP Read in RCP data*

Description

Read RCP Read in RCP data

Usage

```
readRCP(subtype)
```

Arguments

subtype	Either 'Waste' or 'AviationShipping'

Value

magpie object of the RCP data

Author(s)

Julian Oeser

See Also

readSource

Examples

```
## Not run:  a <- readSource(type="RCP", subtype="Waste")
```

---

readREMIND_11Regi *Read REMIND region dependent data*

Description

Read-in an csv files that contains regional data

Usage

```
readREMIND_11Regi(subtype)
```

Arguments

subtype	Name of the regional data, e.g. "p4", "biomass", "ch4waste", "tradecost", "pe2se", "xpres_tax", "deltacapoffset", "capacityFactorGlobal", "capacityFactorRules", "residuesShare", "taxConvergence", "maxFeSubsidy", "maxPeSubsidy", "propFeSubsidy", "fossilExtractionCoeff", "uraniumExtractionCoeff", "RLDCCoefficientsLoB", "RLDCCoefficientsPeak", "earlyRetirementAdjFactor"

Value

magpie object of region dependent data

Author(s)

original: not defined, capacity factor, tax, fossil and RLDC changes: Renato Rodrigues

See Also

readSource

Examples

```
## Not run:
a <- readSource(type = "REMIND_11Regi", subtype = "capacityFactorGlobal")

## End(Not run)
```

---

readREN21 *Reads policy database from REN21 2017 with capacity targets or regional technology costs*

Description

Reads excel sheet with data on proposed policies, on Renewable energy capacity targets (which are broken down into Total Installed Capacity (TIC-Absolute), Additional Installed Capacity (AC-Absolute), and Production Absolute targets) or regional technology costs

Usage

```
readREN21(subtype)
```

Arguments

subtype	Capacity Generation Emissions Share

Details

Country name is ISO coded. Capacity/Additional Capacity targets are in GW. Generation/Production targets are in GWh.

Value

magpie object with Total Installed Capacity targets in GW for different target years

Author(s)

Aman Malik, Lavinia Baumstark

readStationary *Load Stationary File as magclass object*

Description

Load Stationary File as magclass object

Usage

```
readStationary()
```

Value

magclass object

Author(s)

Antoine Levesque, Robin Hasse

readStegmann2022 *Read PlasticsEoL*

Description

Read-in data for the End-of-Life fate of plastics from 1.Stegmann, P., Daioglou, V., Londo, M., van Vuuren, D. P. & Junginger, M. Plastic futures and their CO2 emissions. Nature 612, 272–276 (2022). https://www.nature.com/articles/s41586-022-05422-5 Link to SI: https://static-content.springer.com/esm/art%3A10.
022-05422-5/MediaObjects/41586_2022_5422_MOESM1_ESM.xlsx #nolint

Usage

```
readStegmann2022()
```

Value

magpie object of the data

Author(s)

Falk Benke, Simón Moreno

See Also

readSource

Examples

```
## Not run:
a <- readSource(type = "Stegmann2022")

## End(Not run)
```

`readStrefler` *Get data on enhanced weathering*

Description

Get data on enhanced weathering

Usage

```
readStrefler(subtype)
```

Arguments

`subtype`	type of data, one of "weathering_graderegi", "weathering_costs"

Value

magpie object of region dependent data

See Also

`readSource`

Examples

```
## Not run:
a <- readSource(type="Strefler", subtype="weathering_graderegi")

## End(Not run)
```

`readTCdamageKrichene` *Reads country-specific damage coefficients for tropical cyclones from Krichene et al. (in prep.). Data has been provided by the authors, but will be made publicly available as well. This contains data for 41 countries (those exposed to tropical cyclones), and two coefficients (constant and linear temperature)*

Description

Reads country-specific damage coefficients for tropical cyclones from Krichene et al. (in prep.). Data has been provided by the authors, but will be made publicly available as well. This contains data for 41 countries (those exposed to tropical cyclones), and two coefficients (constant and linear temperature)

Usage

```
readTCdamageKrichene(subtype)
```

Arguments

`subtype`	data subtype. Either "const" or "tasK"

Value

TC damage coefficients

Author(s)

Franziska Piontek

`readTransportSubsidies` *Read transport subsidies data*

Description

Read-in transport subsidies csv files as magclass object

Usage

```
readTransportSubsidies()
```

Value

magpie object of the transport subsidies for BEV, FCEV and PHEV (euros/car) for private and legal entities

Author(s)

Renato Rodrigues

See Also

readSource

Examples

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

---

readUBA *Read UBA*

Description

Read UBA

Usage

```
readUBA()
```

Value

A magpie object.

Author(s)

Falk Benke

---

readUNFCCC *Read UNFCCC data*

Description

Read UNFCCC data

Usage

```
readUNFCCC()
```

Value

A [‘magpie‘][magclass::magclass] object.

Author(s)

Falk Benke

See Also

['readSource()']

`readUNFCCC_NDC` *Reads NDC policy database with capacity, emission, and share targets, originally based on Rogelj et al. 2017*

Description

Reads excel sheet with NDC (Nationally Determined Contributions) data on different policy targets (capacity, emission, and share targets) with different variations

Usage

```
readUNFCCC_NDC(subtype)
```

Arguments

`subtype`	Capacity_2023_cond (or 2018/2021/2022 or uncond) for capacity target, Emissions_2023_cond (or 2018/2021/2022 or uncond) for Emissions targets

Details

Country name is ISO coded. Capacity/Additional Capacity targets are in GW. Generation/Production targets are in GWh.

Value

magpie object

Author(s)

Aman Malik, Christoph Bertram, Oliver Richters, Sophie Fuchs, Rahel Mandaroux

readUNIDO *UNIDO data*

Description

Read and convert data from United Nations Industrial Organisation.

Usage

```
readUNIDO(subtype = "INDSTAT2")

convertUNIDO(x, subtype = "INDSTAT2")

calcUNIDO(subtype = "INDSTAT2")
```

Arguments

subtype	one of - INDSTAT2: read INDSTAT2 data
x	result from readUNIDO() as passed to convertUNIDO()

Value

A magpie object.

readUNIDO returns raw INDSTAT2 data. convertUNIDO converts to iso3c country codes, selects industry subsectors value added data according to this table

subsector	ISIC	ctable	utable
manufacturing	D	20	17–20
cement	20	20	17–20
chemicals	24	20	17–20
steel	27	20	17–20

and filters data that is either unreasonable or would unduly bias regional regressions according to this table

subsector	iso3c	years
manufacturing	BIH	1990–91
manufacturing	CHN	1963–97
manufacturing	HKG	1963–2015
manufacturing	IRQ	1994–98
manufacturing	MAC	1963–2015
manufacturing	MDV	1963–2015
cement	BDI	1980–2010
cement	CIV	1990–93
cement	HKG	1973–79

cement	IRQ	1992–97
cement	NAM	2007–10
cement	RUS	1970–90
chemicals	CIV	1989
chemicals	HKG	1973–79, 2008–15
chemicals	MAC	1978–79
chemicals	NER	1999–2002
steel	BGD	2011
steel	CHE	1995–96
steel	CHL	2008
steel	HKG	1973–79
steel	HRV	2012
steel	IRL	1980
steel	LKA	2006
steel	MAR	1989–2004
steel	MKD	1996
steel	PAK	1981–82
steel	TUN	2003–06

`calcUNIDO()` calculates `otherInd` subsector values as the difference between `manufacturing` and `cement`, `chemicals`, and `steel` values and is intended to be called through `calcOutput()`, which will aggregate regions.

Author(s)

Michaja Pehl

See Also

`readSource()`, `calcOutput()`

`readUSGS` *Read U.S. Geological Survey data*

Description

Read U.S. Geological Survey data

Usage

```
readUSGS(subtype = "cement")

convertUSGS(x, subtype = "cement")
```

Arguments

subtype	One of • 'cement': read cement production data from U.S. Geological Survey Minerals Yearbook (unit: tonnes per year)
x	Data returned by `readUSGS()`.

Value

A `magpie` object.

Author(s)

Michaja Pehl

---

readvanRuijven2016 *Read van Ruijven et al. (2016) data.*

Description

Read data from van Ruijven et al. 2016, (http://dx.doi.org/10.1016/j.resconrec.2016.04.016, https://www.zotero.org/groups/5
obtained through personal communication (e-mail to Michaja Pehl). Units are tonnes per year.

Usage

```
readvanRuijven2016()
```

Value

A `magpie` object.

Author(s)

Michaja Pehl

See Also

`readSource()`

readWGBU *Read WGBU*

Description

Read-in an WGBU xlsx file as magclass object

Usage

```
readWGBU()
```

Value

magpie object of WGBU

Author(s)

Lavinia Baumstark

See Also

readSource

Examples

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

## End(Not run)
```

readworldsteel *Read World Steel Statistical Yearbook Data*

Description

Read combined data of World Steel Association statistical yearbooks (https://www.worldsteel.org/steel-by-topic/statistics/steel-statistical-yearbook.html).

Usage

```
readworldsteel(subtype = "detailed")
```

Arguments

subtype	One of - ‘detailed‘ returning data for the worksheets - ‘Pig Iron Production‘ - ‘DRI Production‘ - ‘Total Production of Crude Steel‘ - ‘Production in Oxygen-Blown Converters‘ - ‘Production in Open Hearth Furnaces‘ - ‘Production in Electric Arc Furnaces‘ - ‘Apparent Steel Use (Crude Steel Equivalent)‘ from 1991 on or - ‘long‘ returning total production data from 1967 on

Value

A [‘magpie‘][magclass::magclass] object.

Author(s)

Michaja Pehl

See Also

[‘readSource()‘]

---

toolAddDimensions — *Wrapper around magclass::add_dimension supporting more than one value for the new dimension. For each value, the input magclass object is copied, extended by the new dimension and appended to the output.*

Description

Wrapper around magclass::add_dimension supporting more than one value for the new dimension. For each value, the input magclass object is copied, extended by the new dimension and appended to the output.

Usage

```
toolAddDimensions(x, dimVals, dimName, dimCode)
```

Arguments

x	a magclass object
dimVals	list of values for the new dimension to be added
dimName	name of the new dimension
dimCode	dimension number of the new dimension (e.g. 3.1)

Value

the extended magclass object

`toolAggregateTimeSteps` *Aggregate values to n-year averages to suppress volatility*

Description

Aggregate values to n-year averages to suppress volatility

Usage

```
toolAggregateTimeSteps(x, nYears = 5)
```

Arguments

x	a magclass object
nYears	time steps to be used for averaging, defaults to 5

Value

magclass object with averages

Author(s)

Robin Hasse

`toolBiomassSupplyAggregate` *toolBiomassSupplyAggregate The function aggregates biomass supply curves to regionmapping different from H12. It only works if all regions are subregions of H12 regions. The offset parameter (a) is taken from the H12 region. The slope parameter (b) is multiplied by a weight. The weight is the inverse of the share of agricultural area of the subregion in the H12 region.*

Description

toolBiomassSupplyAggregate The function aggregates biomass supply curves to regionmapping different from H12. It only works if all regions are subregions of H12 regions. The offset parameter (a) is taken from the H12 region. The slope parameter (b) is multiplied by a weight. The weight is the inverse of the share of agricultural area of the subregion in the H12 region.

Usage

```
toolBiomassSupplyAggregate(
  x,
  rel = NULL,
 weight = calcOutput("FAOLand", aggregate = F)[, , "6610", pmatch = TRUE][, "y2010", ]
)
```

Arguments

x	magclass object that should be aggregated
rel	relation matrix containing a region mapping.
weight	aggregation weight

Value

return: returns region aggregated biomass supply curve data

Author(s)

Felix Schreyer

---

toolCubicFunctionAggregate

toolCubicFunctionAggregate

Description

Estimates the function that represents the sum of cubic function inverses (sum in the x-axis)

Usage

```
toolCubicFunctionAggregate(
  x,
  rel = NULL,
  xLowerBound = 0,
  xUpperBound = 100,
  returnMagpie = TRUE,
  returnCoeff = TRUE,
  returnChart = FALSE,
  returnSample = FALSE,
  numberOfSamples = 1000,
  unirootLowerBound = -10,
  unirootUpperBound = 1e+100,
  colourPallete = FALSE,
  label = list(x = "x", y = "y", legend = "legend"),
  steepCurve = list()
)
```

Arguments

x	magclass object that should be aggregated or data frame with coefficients as columns.
rel	relation matrix containing a region mapping. A mapping object should contain 2 columns in which each element of x is mapped to the category it should belong to after (dis-)aggregation
xLowerBound	numeric. Lower bound for x sampling (default=0).
xUpperBound	numeric. Upper bound for x sampling (default=100).
returnMagpie	boolean. if true, the function will return a single data table with all the countries in MagPie format. returnChart and returnSample are set to FALSE automatically if this option is active (default=TRUE).
returnCoeff	boolean. Return estimated coefficients (default=TRUE).
returnChart	boolean. Return chart (default=FALSE).
returnSample	boolean. Return samples used on estimation (default=FALSE).
numberOfSamples	numeric. NUmber of y-axis samples used on estimation (default=1e3).
unirootLowerBound	numeric. Lower bound to search for inverse solution in the initial bounds (default = -10).
unirootUpperBound	numeric. Upper bound to search for inverse solution in the initial bounds (default = 1e100).
colourPallete	vector. colour pallete to use on chart (default=FALSE).
label	list. List of chart labels (default=list(x = "x", y = "y", legend = "legend")).
steepCurve	list. List with coefficients for a very "vertical" function for the case with all countries with upper bound zero in an specific region aggregation (default= empty list, list()).

Details

Use case: aggregate country cubic cost functions to a single function that represents the entire region.

input: coefficients of the n-th country level cubic cost function.

Description of the problem: the aggregation of functions that represent unit costs, or prices in the y-axis, and quantities in the x-axis require operations with the inverse of the original functions. As complex functions present analytically challenging inverse function derivations, we adopt a sampling method to derive the function that corresponds to the sum of cubic function inverses.

Further extensions: the R function can be extended to support more complex curve estimations (beyonf third degree), whenever the mathematical function have a well defined inverse function in the selected boundaries.

Value

return: returns a list of magpie objects containing the coefficients for the aggregate function. If returnMagpie is FALSE, returns a list containing the coefficients for the aggregate function (returnCoeff=TRUE), charts (returnChart=FALSE) and/or samples used in the estimation (returnSample=FALSE).

Author(s)

Renato Rodrigues

See Also

toolCubicFunctionDisaggregate

Examples

```
# Example
# data
EUR <- setNames(data.frame(30, 50, 0.123432, 2), c("c1", "c2", "c3", "c4"))
NEU <- setNames(data.frame(30, 50, 1.650330, 2), c("c1", "c2", "c3", "c4"))
df <- rbind(EUR, NEU)
row.names(df) <- c("EUR", "NEU")
# maxExtraction (upper limit for function estimation)
maxExtraction <- 23
# output
output <- toolCubicFunctionAggregate(df,
  xUpperBound = maxExtraction,
  returnMagpie = FALSE, returnChart = TRUE, returnSample = TRUE,
  label = list(x = "Cumulated Extraction", y = "Cost", legend = "Region Fuel Functions")
)
output$coeff
output$chart
```

---

toolCubicFunctionDisaggregate

toolCubicFunctionDisaggregate

Description

Estimates cubic function inverses based on a weight factor that sum up to the original cubic function (sum in the x-axis)

Usage

```
toolCubicFunctionDisaggregate(
  x,
  weight,
  rel = NULL,
```

```
  xLowerBound = 0,
  xUpperBound = 100,
  returnMagpie = TRUE,
  returnCoeff = TRUE,
  returnChart = FALSE,
  returnSample = FALSE,
  numberOfSamples = 1000,
  unirootLowerBound = -10,
  unirootUpperBound = 1e+100,
  colourPallete = FALSE,
  label = list(x = "x", y = "y", legend = "legend")
)
```

Arguments

x	magclass object that should be aggregated or data frame with coefficients as columns.
weight	magclass object containing weights which should be considered for a weighted aggregation. The provided weight should only contain positive values, but does not need to be normalized (any positive number>=0 is allowed).
rel	relation matrix containing a region mapping. A mapping object should contain 2 columns in which each element of x is mapped to the category it should belong to after (dis-)aggregation
xLowerBound	numeric. Lower bound for x sampling (default=0).
xUpperBound	numeric. Upper bound for x sampling (default=100).
returnMagpie	boolean. if true, the function will return a single data table with all the countries in MagPie format. returnChart and returnSample are set to FALSE automatically if this option is active (default=TRUE).
returnCoeff	boolean. Return estimated coefficients (default=TRUE).
returnChart	boolean. Return chart (default=FALSE).
returnSample	boolean. Return samples used on estimation (default=FALSE).
numberOfSamples	numeric. NUmber of y-axis samples used on estimation (default=1e3).
unirootLowerBound	numeric. Lower bound to search for inverse solution in the initial bounds (default = -10).
unirootUpperBound	numeric. Upper bound to search for inverse solution in the initial bounds (default = 1e100).
colourPallete	vector. colour pallete to use on chart (default=FALSE).
label	list. List of chart labels (default=list(x = "x", y = "y", legend = "legend")).

Details

Use case: disaggregate a single region cubic cost function to multiple country cubic functions weighted by a contribution factor. The sum of the countries function output is equal to the original regional function.

input: coefficients of the n-th country level cubic cost function.

Description of the problem: the disaggregation of functions that represent unit costs (or prices) in the y-axis and quantities in the x-axis require operations with the inverse of the original functions. As complex functions present analytically challenging inverse function derivations, we adopt a sampling method to derive the function that corresponds to the sum of cubic function inverses.

Further extensions: the R function can be extended to support more complex curve estimations (beyond third degree), whenever the mathematical function have a well defined inverse function in the selected boundaries.

Value

return: returns a list of magpie objects containing the coefficients for the aggregate function. If returnMagpie is FALSE, returns a list containing the coefficients for the aggregate function (returnCoeff=TRUE), charts (returnChart=FALSE) and/or samples used in the estimation (returnSample=FALSE).

Author(s)

Renato Rodrigues

See Also

toolCubicFunctionAggregate

Examples

```
# Example
# LAM coefficients
df <- setNames(data.frame(30, 50, 0.34369, 2), c("c1", "c2", "c3", "c4"))
row.names(df) <- "LAM"
# weight
weight <- setNames(c(21, 0, 579, 3, 228), c("ARG", "BOL", "BRA", "CHL", "COL"))
# maxExtraction (upper limit for function estimation)
maxExtraction <- 100
# output
output <- toolCubicFunctionDisaggregate(df, weight,
  xUpperBound = maxExtraction,
  returnMagpie = FALSE, returnChart = TRUE, returnSample = TRUE,
  label = list(x = "Cumulated Extraction", y = "Cost", legend = "Region Fuel Functions")
) #' output$chart
output$coeff
output$chart
```

toolFillEU34Countries *Sets values for 6 EU countries not belonging to EU 28 but EU 34 to zero if they are NA. Used to avoid EUR region yielding NA because of these countries.*

Description

Sets values for 6 EU countries not belonging to EU 28 but EU 34 to zero if they are NA. Used to avoid EUR region yielding NA because of these countries.

Usage

```
toolFillEU34Countries(x)
```

Arguments

x	magpie object with 249 ISO country codes in the spatial dimension

Author(s)

Falk Benke

toolSolarFunctionAggregate *toolSolarFunctionAggregate*

Description

Aggregate Solar data into regions

Usage

```
toolSolarFunctionAggregate(
  x,
  rel = NULL,
 weight = calcOutput("FE", aggregate = FALSE)[, "y2015", "FE|Electricity (EJ/yr)"]
)
```

Arguments

x	magclass object that should be aggregated
rel	relation matrix containing a region mapping. A mapping object should contain 2 columns in which each element of x is mapped to the category it should belong to after (dis-)aggregation
weight	aggregation weight (should be FE|Electricity (EJ/yr) in 2015)

Value

return: returns region aggregated solar data

Author(s)

Felix Schreyer, Renato Rodrigues, Julian Oeser

`tool_expand_tibble` *Expand tibble across scenarios and regions with default values*

Description

The data.frame ‘d‘ is expanded in such a manner that all rows with ‘NA‘ in either the ‘scenario‘ or ‘region‘ columns are extended to repeat for all scenarios and regions listed in ‘scenarios‘ and ‘regions‘. Rows with specified scenarios and/or regions will overwrite extended ones. Regions are expanded before scenarios.

Usage

```
tool_expand_tibble(d, scenarios, regions, structure.columns = NULL)
```

Arguments

`d`	A data.frame with columns ‘scenario‘ and ‘region‘.
`scenarios`	A character vector of scenario names.
`regions`	A character vector of region names.
`structure.columns`	A character vector of column names to be carried along.

Value

A ‘tibble‘.

Examples

```
## Not run:
tribble(
  ~scenario,   ~region,   ~value,
  NA,          NA,        0,
  NA,          'CHA',     1,
  'SSP1',      NA,        2,
  'SSP2EU',    'DEU',     3) %>%
  tool_expand_tibble(scenarios = c('SSP1', 'SSP2EU', 'SSP5'),
                     regions = c('CHA', 'DEU', 'USA')) %>%
  pivot_wider(names_from = 'region')

tribble(
```

```
  ~scenario,   ~region,   ~name,   ~value,
  NA,          NA,        'A',     0,
  NA,          'CHA',     'B',     1,
  'SSP1',      NA,        'A',     2,
  'SSP2EU',    'DEU',     'B',     3) %>%
  tool_expand_tibble(scenarios = c('SSP1', 'SSP2EU', 'SSP5'),
                     regions = c('CHA', 'DEU', 'USA'),
                     structure.columns = 'name')

## End(Not run)
```

`tool_fix_IEA_data_for_Industry_subsectors`
Apply corrections to IEA data needed for Industry subsectors

Description

Apply corrections to IEA data to cope with fragmentary time series and replace outputs from blast furnaces and coke ovens, that are inputs into industry subsectors, by their respective inputs. The corrections done by this function are rather rudimentary and crude. This gets smoothed away in regional aggregation. But do not use the resulting country-level data without additional scrutiny.

Usage

```
tool_fix_IEA_data_for_Industry_subsectors(data, ieamatch, threshold = 0.01)
```

Arguments

`data`	MAgPIE object containing the IEA Energy Balances data
`ieamatch`	mapping of IEA product/flow combinations to REMIND `sety`/`fety`/`te` combinations as used in `calcIO()`
`threshold`	minimum share each industry subsector uses of each product. Defaults to 1 %.

Details

Use regional or global averages if IEA industry data lists energy use only as "non-specified". Outputs from blast furnaces (`BLFURGS`, `OGASES`) and coke ovens (`OVENCOKE`, `COKEOVGS`, `COALTAR`, `NONCRUDE`), that are inputs into industry subsectors. Used internally in `calcIO()` for subtype `output_Industry_subsectors`.

Value

a MAgPIE object

Author(s)

Michaja Pehl

Index