Package 'cer'

November 2, 2026

Title Download and Tidy Australian Clean Energy Regulator Data

Version 0.2.0

Description Fetch Australian Clean Energy Regulator data on carbon credits, safeguard mechanism facilities, renewable energy certificates, and greenhouse gas reporting. Provides tidy access to the Australian Carbon Credit Unit (ACCU) Scheme project register, Safeguard Mechanism baselines and covered emissions, Large-scale Renewable Energy Target (LRET) power station accreditations, Small-scale Renewable Energy Scheme (SRES) installation data, the National Greenhouse and Energy Reporting (NGER) scheme, and Quarterly Carbon Market Reports <https://cer.gov.au/markets/reports-and-data>. Includes a post-Chubb ACCU integrity layer (Chubb 2022 Independent Review), Safeguard reform handling (declining industry baselines from July 2023), National Greenhouse and Energy Reporting scope discipline (Scope 1 / Scope 2 market vs location / Climate Active), reconciliation against the Quarterly Carbon Market Report, and reproducibility helpers (snapshot pinning, SHA-256 cache integrity, session manifest, optional Zenodo deposit). Data is published by the Clean Energy Regulator under a Creative Commons Attribution 4.0 International licence.

Depends R (>= 4.1.0)

License MIT + file LICENSE

Encoding UTF-8

Language en-US

Imports cli (>= 3.6.0), httr2 (>= 1.0.0), readxl (>= 1.4.0), tools, utils

Suggests digest, jsonlite, knitr, openssl, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

VignetteBuilder knitr

URL https://charlescoverdale.github.io/cer/, https://github.com/charlescoverdale/cer

BugReports https://github.com/charlescoverdale/cer/issues

Config/roxygen2/version 8.1.0

NeedsCompilation no

Author Charles Coverdale [aut, cre]

Maintainer Charles Coverdale <charlesfcoverdale@gmail.com>

Repository CRAN

Date/Publication 2026-11-01 23:20:08 UTC

Contents

cer_accu_aggregate . . . 3
cer_accu_contracts . . . 4
cer_accu_issuances . . . 4
cer_accu_methods . . . 5
cer_accu_projects . . . 6
cer_accu_relinquishments . . . 8
cer_cache_info . . . 9
cer_cite . . . 9
cer_clear_cache . . . 10
cer_deposit_zenodo . . . 11
cer_international_comparator . . . 12
cer_legislative_instrument . . . 13
cer_lgc_power_stations . . . 14
cer_manifest . . . 15
cer_manifest_clear . . . 16
cer_manifest_write . . . 16
cer_method_determination . . . 17
cer_method_integrity . . . 18
cer_nger_climate_active . . . 19
cer_nger_corporate . . . 20
cer_nger_electricity . . . 21
cer_nger_scope . . . 22
cer_qcmr . . . 23
cer_qcmr_reference . . . 24
cer_reconcile . . . 25
cer_safeguard_baseline_trajectory . . . 26
cer_safeguard_facilities . . . 27
cer_safeguard_industry_baselines . . . 28
cer_safeguard_teba_facilities . . . 29
cer_sha256 . . . 29
cer_snapshot . . . 30
cer_sres_installations . . . 31
cer_to_carbondata . . . 32
print.cer_tbl . . . 33

Index **34**

`cer_accu_aggregate` *Integrity-weighted ACCU aggregation*

Description

Takes an `ato_tbl` or data frame of ACCU project records (from `cer_accu_projects()`), joins the method integrity scorecard, and aggregates by integrity tier. Emits a warning if contested or under-review methods contribute materially to the total so the analyst cannot silently aggregate them into a single headline figure.

Usage

```
cer_accu_aggregate(x, value_col = NULL, warn_contested_pct = 0.05)
```

Arguments

`x`	A `cer_tbl` or data frame with columns `method` and a numeric issuance column.
`value_col`	Column to sum. Default `NULL` uses the register's `accus_total_units_issued` column (or `accus_issued`, if that is the one `x` has).
`warn_contested_pct`	Warn if the contested tier exceeds this share of the total. Default `0.05` (5%).

Value

A data frame with one row per integrity tier and columns `integrity_tier`, `projects`, `<value_col>_sum` and `share_pct`. Register methods are assigned to a tier through the scorecard's `register_pattern`; methods it does not cover are `"unclassified"`.

See Also

Other accu: `cer_accu_contracts()`, `cer_accu_issuances()`, `cer_accu_methods()`, `cer_accu_projects()`, `cer_accu_relinquishments()`, `cer_method_determination()`, `cer_method_integrity()`

Examples

```
op <- options(cer.cache_dir = tempdir())
try({
  p <- cer_accu_projects()
  cer_accu_aggregate(p)
})
options(op)
```

cer_accu_contracts *ACCU Carbon Abatement Contract register*

Description

Government purchase contracts under Emissions Reduction Fund auctions administered by the CER.

Usage

```
cer_accu_contracts(auction = NULL)
```

Arguments

auction	Optional character vector of auction identifiers (e.g. "ERF Auction 18").

Value

A cer_tbl with one row per contract.

Source

Clean Energy Regulator Carbon Abatement Contract register (CC BY 4.0).

See Also

Other accu: cer_accu_aggregate(), cer_accu_issuances(), cer_accu_methods(), cer_accu_projects(), cer_accu_relinquishments(), cer_method_determination(), cer_method_integrity()

Examples

```
op <- options(cer.cache_dir = tempdir())
cer_accu_contracts()
options(op)
```

cer_accu_issuances *ACCU issuances by project*

Description

Derived from the ACCU Scheme project register: returns one row per project with accus_total_units_issued, project metadata, and total issuance net of relinquishments. For method- or period-level issuance tracking, query cer_accu_projects() directly.

Usage

```
cer_accu_issuances(method = NULL, project_id = NULL, from = NULL, to = NULL)
```

Arguments

method	Optional character vector of method substrings.
project_id	Optional character vector of ERF / EOP project identifiers.
from, to	Optional Date (or YYYY-MM-DD string) for crediting-period overlap filter. NULL returns all projects.

Value

A cer_tbl with one row per project.

Source

Clean Energy Regulator ACCU project register (CC BY 4.0).

See Also

Other accu: cer_accu_aggregate(), cer_accu_contracts(), cer_accu_methods(), cer_accu_projects(), cer_accu_relinquishments(), cer_method_determination(), cer_method_integrity()

Examples

```
op <- options(cer.cache_dir = tempdir())
i <- cer_accu_issuances(method = "savanna")
head(i)
options(op)
```

---

cer_accu_methods *Approved ACCU Scheme methods*

Description

A static table of ACCU Scheme methods grouped by sector (Agriculture, Energy Efficiency, Landfill and Waste, Mining/Oil/Gas, Vegetation, Savanna, Blue Carbon, Transport). Built from the CER methods index at build time.

Usage

```
cer_accu_methods()
```

Details

Because the CER publishes methods as per-method PDFs on the Federal Register of Legislation rather than a machine-readable index, this function returns a package-embedded lookup. The `status` column captures post-Chubb Review changes: methods are flagged as `"active"`, `"expired"`, `"suspended"`, or `"under_review"`. Run `cer_accu_projects()` then `unique(x$method)` to cross-check against currently-registered projects.

Value

A `cer_tbl` with columns `method`, `sector`, `short_name`, `status`, `commenced`, and `source_url`.

Source

Clean Energy Regulator ACCU Scheme methods index https://cer.gov.au/schemes/australian-carbon-credit-uni accu-scheme-methods.

See Also

Other accu: `cer_accu_aggregate()`, `cer_accu_contracts()`, `cer_accu_issuances()`, `cer_accu_projects()`, `cer_accu_relinquishments()`, `cer_method_determination()`, `cer_method_integrity()`

Examples

```
m <- cer_accu_methods()
head(m)
```

`cer_accu_projects` *ACCU Scheme project register*

Description

Download and parse the Clean Energy Regulator's Australian Carbon Credit Unit ('ACCU') Scheme project register. Returns one row per project with proponent, method, state, crediting period, permanence period, and cumulative ACCUs issued and relinquished.

Usage

```
cer_accu_projects(
  state = NULL,
  method = NULL,
  status = c("active", "revoked", "completed", "all")
)
```

Arguments

`state`	Optional character vector of Australian state codes (e.g. `"NSW"`, `c("VIC", "QLD")`) to filter on. `NULL` returns projects across all states.
`method`	Optional character vector of method name substrings (case-insensitive, e.g. `"savanna"`, `"landfill"`, `"vegetation"`).
`status`	Character. One of `"active"` (default), `"revoked"`, `"completed"`, or `"all"`.

Details

The register is updated monthly by the CER, usually around the third week. Downloaded once per R session (or invalidate with `cer_clear_cache()`).

Value

A `cer_tbl` (data frame) with one row per ACCU project.

Source

Clean Energy Regulator, ACCU project and contract register: https://cer.gov.au/markets/reports-and-data/accu-project-and-contract-register. Licensed under CC BY 4.0.

References

Commonwealth of Australia. *Carbon Credits (Carbon Farming Initiative) Act 2011*. Enabling legislation for the ACCU Scheme.

Chubb, I. (2022). *Independent Review of Australian Carbon Credit Units*. Commonwealth of Australia, Department of Climate Change, Energy, the Environment and Water.

Climate Change Authority (2024). *2026 Review of the ACCU Scheme: Issues paper*.

Macintosh, A., Butler, D., Larraondo, P., Evans, M.C., Ansell, D., Gibbons, P., Lindenmayer, D., Waschka, M. and Fisher, R. (2022). "The emperor's new clothes: assessing the integrity of ACCUs." Australian National University working paper series.

See Also

Other accu: `cer_accu_aggregate()`, `cer_accu_contracts()`, `cer_accu_issuances()`, `cer_accu_methods()`, `cer_accu_relinquishments()`, `cer_method_determination()`, `cer_method_integrity()`

Examples

```
op <- options(cer.cache_dir = tempdir())
p <- cer_accu_projects(state = "NSW")
head(p)
options(op)
```

`cer_accu_relinquishments` *ACCU relinquishments by project*

Description

Returns projects with non-zero ACCU relinquishments. Useful for tracking voluntary cancellations and compliance obligations.

Usage

```
cer_accu_relinquishments(project_id = NULL)
```

Arguments

`project_id`	Optional character vector of project IDs.

Value

A `cer_tbl` with one row per project with `accus_relinquished > 0`.

Source

Clean Energy Regulator ACCU project register (CC BY 4.0).

See Also

Other accu: `cer_accu_aggregate()`, `cer_accu_contracts()`, `cer_accu_issuances()`, `cer_accu_methods()`, `cer_accu_projects()`, `cer_method_determination()`, `cer_method_integrity()`

Examples

```
op <- options(cer.cache_dir = tempdir())
cer_accu_relinquishments()
options(op)
```

cer_cache_info *Inspect the local cer cache*

Description

Returns the cache directory, number of files, total size, and a per-file summary. Useful for deciding whether to call `cer_clear_cache()`.

Usage

```
cer_cache_info()
```

Value

A list with elements `dir`, `n_files`, `size_bytes`, `size_human`, and `files` (a data frame with `name`, `size_bytes`, `modified`).

See Also

Other configuration: `cer_clear_cache()`

Examples

```
op <- options(cer.cache_dir = tempdir())
cer_cache_info()
options(op)
```

cer_cite *Cite a cer_tbl or URL in BibTeX, plain-text, or APA form*

Description

Returns a citation suitable for footnotes, papers, and carbon market analysis reports. Uses the provenance attributes attached to every `cer_tbl`: source URL, licence, retrieval date, title, snapshot pin (from `cer_snapshot()`), and SHA-256 digest.

Usage

```
cer_cite(x, style = c("text", "bibtex", "apa"), doi = NULL)
```

Arguments

`x`	A `cer_tbl` (as returned by any `cer_*` data function) or a character URL.
`style`	One of `"text"` (default), `"bibtex"`, or `"apa"`.
`doi`	Optional DOI (e.g. from `cer_deposit_zenodo()`) to include in BibTeX output as a `doi` field and APA suffix.

Details

For carbon market research, the BibTeX `note` field includes the snapshot date and first 12 hex characters of the SHA-256, which are the minimum provenance fields needed to defend a published figure against a reviewer.

Value

A character string.

See Also

Other reproducibility: `cer_deposit_zenodo()`, `cer_legislative_instrument()`, `cer_manifest()`, `cer_manifest_clear()`, `cer_manifest_write()`, `cer_sha256()`, `cer_snapshot()`

Examples

```
x <- data.frame(a = 1)
x <- structure(x,
 cer_source = "https://cer.gov.au/markets/reports-and-data/accu-project-and-contract-register",
  cer_licence = "CC BY 4.0",
  cer_retrieved = as.POSIXct("2026-04-24 00:00:00", tz = "UTC"),
  cer_title = "ACCU project register",
  cer_sha256 = "abc123def456",
  cer_snapshot_date = "2026-04-24",
  class = c("cer_tbl", "data.frame"))
cer_cite(x)
cer_cite(x, style = "bibtex")
# DOI style: supply any minted DOI (Zenodo, DataCite, etc.).
# The placeholder below is illustrative only.
cer_cite(x, style = "apa", doi = "10.5281/zenodo.XXXXXXXX")
```

`cer_clear_cache` *Clear the cer cache*

Description

Deletes all locally cached files. The next call to any data function will re-download from the Clean Energy Regulator.

Usage

```
cer_clear_cache()
```

Value

Invisibly returns `NULL`.

See Also

Other configuration: `cer_cache_info()`

Examples

```
op <- options(cer.cache_dir = tempdir())
cer_clear_cache()
options(op)
```

`cer_deposit_zenodo` *Prepare a Zenodo deposit payload for the session manifest*

Description

Builds Zenodo-shaped metadata for a data deposit using the current `cer_manifest()` and the snapshot pin set via `cer_snapshot()`. Dry run by default; pass `upload = TRUE` and a token to actually deposit.

Usage

```
cer_deposit_zenodo(
  title = NULL,
  description = NULL,
  creators = list(list(name = "Anonymous")),
  keywords = c("Clean Energy Regulator", "carbon markets", "Australia", "ACCU",
    "Safeguard", "reproducibility"),
  upload = FALSE,
  sandbox = FALSE,
  token = Sys.getenv("ZENODO_TOKEN")
)
```

Arguments

`title`	Deposit title. Defaults to a snapshot-date string.
`description`	Free-text description. Defaults to a short auto-generated summary of datasets fetched.
`creators`	List of creator records.
`keywords`	Character vector of keywords.
`upload`	Logical. If `TRUE`, POSTs to Zenodo.
`sandbox`	Logical. If `TRUE`, uses Zenodo Sandbox for tests.
`token`	Zenodo personal access token. Defaults to `Sys.getenv("ZENODO_TOKEN")`.

Details

Because CER data is revised retroactively, Zenodo DOIs are the strongest reproducibility artefact available: they freeze the exact bytes of the ACCU project register, Safeguard facility file, or QCMR workbook at the date of analysis.

Value

A list with `payload`, `manifest_path`, and (on upload) `deposit_id`, `doi_prereserve`, `url`.

See Also

Other reproducibility: `cer_cite()`, `cer_legislative_instrument()`, `cer_manifest()`, `cer_manifest_clear()`, `cer_manifest_write()`, `cer_sha256()`, `cer_snapshot()`

Examples

```
cer_snapshot("2026-04-24")
cer_deposit_zenodo(
  title = "CER data snapshot for working paper v1",
  creators = list(list(name = "Author, A.")),
  upload = FALSE
)
```

---

cer_international_comparator

Bundled international carbon market comparator

Description

Returns headline price/volume figures for comparison between ACCU (Australia) and major international markets: EU ETS, New Zealand ETS, California-Quebec (WCI), Verra VCUs, and Gold Standard. Values are indicative snapshots. For live data use the companion `carbondata` package.

Usage

```
cer_international_comparator()
```

Value

A `cer_tbl` of indicative comparator data.

References

International Carbon Action Partnership (annual). *Emissions Trading Worldwide: Status Report*. https://icapcarbonaction.com/en

World Bank (annual). *State and Trends of Carbon Pricing*.

See Also

Other interop: cer_to_carbondata()

Examples

```
cer_international_comparator()
```

`cer_legislative_instrument` *Build a Federal Register of Legislation URL*

Description

Given a Legislative Instrument identifier (e.g. "F2024L01292" for the Environmental Plantings 2024 method), return the stable URL on the Federal Register of Legislation. This is the authoritative source of truth for ACCU methods, Safeguard Rule amendments, and NGER Determinations.

Usage

```
cer_legislative_instrument(instrument)
```

Arguments

instrument	Character identifier (e.g. "F2024L01292"). Case-insensitive.

Value

A length-1 character URL string.

References

Commonwealth of Australia. *Legislation Act 2003* establishes the Federal Register of Legislation as the authoritative source for Legislative Instruments.

See Also

Other reproducibility: cer_cite(), cer_deposit_zenodo(), cer_manifest(), cer_manifest_clear(), cer_manifest_write(), cer_sha256(), cer_snapshot()

Examples

```
cer_legislative_instrument("F2024L01292")
cer_legislative_instrument("F2015L00398")
```

`cer_lgc_power_stations` *LRET accredited power stations*

Description

Power stations accredited under the Large-scale Renewable Energy Target ('LRET') and eligible to create Large-scale Generation Certificates ('LGCs').

Usage

```
cer_lgc_power_stations(technology = NULL, state = NULL)
```

Arguments

`technology`	Optional character vector of technology substrings (e.g. `"solar"`, `"wind"`, `"hydro"`, `"bioenergy"`).
`state`	Optional character vector of Australian state codes (e.g. `c("VIC", "SA")`).

Value

A `cer_tbl` with one row per accredited power station, including accreditation date, technology, capacity (MW), and state.

Source

Clean Energy Regulator, Historical large-scale renewable energy supply data: https://cer.gov.au/markets/reports-and-data/large-scale-renewable-energy-data/historical-large-scale-renewable-ene
Licensed under CC BY 4.0.

References

Commonwealth of Australia. *Renewable Energy (Electricity) Act 2000*. Establishes the Renewable Energy Target (RET), the LGC (Large-scale Generation Certificate) and the shortfall charge mechanism.

Clean Energy Regulator (annual). *LRET: Accreditation and eligibility guidelines for power stations*.

Australian Energy Market Commission (2023). *Integrated System Plan 2024* reference on renewable capacity expansion.

Examples

```
op <- options(cer.cache_dir = tempdir())
s <- cer_lgc_power_stations(technology = "solar", state = "VIC")
head(s)
options(op)
```

`cer_manifest` *Return the session manifest of fetched CER datasets*

Description

Every call to a data function appends one row to the session manifest, recording URL, dataset title, SHA-256 of the cached file, size, retrieval timestamp, and the snapshot pin set via `cer_snapshot()`. Duplicate URLs within a session are deduplicated (last fetch wins).

Usage

```
cer_manifest(format = c("df", "yaml", "json"))
```

Arguments

`format`	One of `"df"` (default), `"yaml"`, or `"json"`.

Details

For carbon market research this is the minimum artefact needed for reproducibility: ACCU project registers and Safeguard releases change retroactively, so a citation that points to a URL alone is not enough. The manifest plus SHA-256 fixes the exact bytes analysed.

Value

A data frame, YAML string, or JSON string.

See Also

Other reproducibility: `cer_cite()`, `cer_deposit_zenodo()`, `cer_legislative_instrument()`, `cer_manifest_clear()`, `cer_manifest_write()`, `cer_sha256()`, `cer_snapshot()`

Examples

```
op <- options(cer.cache_dir = tempdir())
cer_manifest_clear()
cer_snapshot("2026-04-24")
cer_manifest()
options(op)
```

cer_manifest_clear *Clear the session manifest*

Description

Clear the session manifest

Usage

```
cer_manifest_clear()
```

Value

Invisibly `NULL`.

See Also

Other reproducibility: `cer_cite()`, `cer_deposit_zenodo()`, `cer_legislative_instrument()`, `cer_manifest()`, `cer_manifest_write()`, `cer_sha256()`, `cer_snapshot()`

Examples

```
cer_manifest_clear()
```

cer_manifest_write *Write the session manifest to a file*

Description

Write the session manifest to a file

Usage

```
cer_manifest_write(path, format = c("auto", "csv", "yaml", "json"))
```

Arguments

`path`	Output file path. Extension determines format when `format = "auto"`.
`format`	One of `"auto"`, `"csv"`, `"yaml"`, `"json"`.

Value

Invisibly, the absolute path written.

See Also

Other reproducibility: cer_cite(), cer_deposit_zenodo(), cer_legislative_instrument(), cer_manifest(), cer_manifest_clear(), cer_sha256(), cer_snapshot()

Examples

```
p <- tempfile(fileext = ".csv")
cer_manifest_clear()
cer_manifest_write(p)
```

---

cer_method_determination

Federal Register URL for an ACCU method determination

Description

Given a method short name (e.g. "HIR", "Savanna_2026"), return the authoritative legislative instrument URL on the Federal Register of Legislation. This is the technical determination document that defines measurement, monitoring, baseline, and permanence rules for the method.

Usage

```
cer_method_determination(method_short)
```

Arguments

method_short	Character scalar matching a method_short value from cer_method_integrity().

Value

A length-1 character URL, or NA_character_ if not found.

See Also

Other accu: cer_accu_aggregate(), cer_accu_contracts(), cer_accu_issuances(), cer_accu_methods(), cer_accu_projects(), cer_accu_relinquishments(), cer_method_integrity()

Examples

```
cer_method_determination("HIR")
cer_method_determination("Savanna_2026")
```

`cer_method_integrity` *ACCU method integrity scorecard*

Description

Returns a method-level integrity scorecard for all ACCU methods. Columns include `integrity_tier` (high / standard / contested), `chubb_affected` (whether the Chubb 2022 Independent Review flagged the method), `chubb_recommendation_applies` (whether a specific Chubb recommendation targets the method), and `federal_register_instrument` (authoritative legislative instrument ID).

Usage

```
cer_method_integrity(tier = NULL, status = NULL)
```

Arguments

`tier`	Optional filter on `integrity_tier`. One or more of `"high"`, `"standard"`, `"contested"`.
`status`	Optional filter on method `status`. One or more of `"active"`, `"suspended"`, `"under_review"`, `"superseded"`, `"expired"`.

Details

Use this to filter or weight ACCU aggregates, or to annotate a chart with integrity context. For example, an analyst reporting "185 million ACCUs issued to date" should, at minimum, break the total down by `integrity_tier` and flag the contested share.

Value

A `cer_tbl` with one row per method (19 rows at v0.2.0). The `register_pattern` column is the regular expression that assigns the ACCU project register's full legal method names (for example, "Carbon Credits (Carbon Farming Initiative-Landfill Gas) Methodology Determination 2015") to each row; `cer_accu_aggregate()` and the `chubb_flag` column of `cer_accu_projects()` use it. Register methods that no pattern matches are reported as `"unclassified"`.

References

Chubb, I. (2022). *Independent Review of Australian Carbon Credit Units*. Commonwealth of Australia, Department of Climate Change, Energy, the Environment and Water.

Macintosh, A., Butler, D., Larraondo, P., Evans, M.C., Ansell, D., Gibbons, P., Lindenmayer, D., Waschka, M. and Fisher, R. (2022). "The emperor's new clothes: assessing the integrity of ACCUs." Australian National University working paper series.

Butler, D., Macintosh, A. and Pouliot, M. (2023). "Response to the Chubb Review." Australian National University.

Ansell, D. *et al.* (2020). "Contemporary fire regimes of northern Australia: a review of current patterns, impacts and challenges." *Environmental Research Reviews*.

See Also

Other accu: cer_accu_aggregate(), cer_accu_contracts(), cer_accu_issuances(), cer_accu_methods(), cer_accu_projects(), cer_accu_relinquishments(), cer_method_determination()

Examples

```
cer_method_integrity()
cer_method_integrity(tier = "contested")
cer_method_integrity(status = "active")
```

---

cer_nger_climate_active

Translate NGER scope columns to Climate Active categories

Description

Climate Active (Australia's voluntary carbon-neutral certification) uses a Scope 1 / Scope 2 (market-based and location-based) / Scope 3 taxonomy. NGER publishes Scope 1 and Scope 2 only (Scope 3 is not NGER-reportable). This helper renames columns so an NGER output flows into a Climate Active inventory template.

Usage

```
cer_nger_climate_active(x)
```

Arguments

x	A cer_tbl or data frame from cer_nger_corporate().

Value

A data frame with Climate Active-style columns: operational_scope_1_t_co2e, operational_scope_2_location_t_co operational_scope_2_market_t_co2e.

References

Department of Climate Change, Energy, the Environment and Water (annual). *Climate Active Carbon Neutral Standard for Organisations*. Commonwealth of Australia.

Greenhouse Gas Protocol (2004). *A Corporate Accounting and Reporting Standard (Revised Edition)*. World Resources Institute.

See Also

Other nger: cer_nger_corporate(), cer_nger_electricity(), cer_nger_scope()

Examples

```
op <- options(cer.cache_dir = tempdir())
try({
  x <- cer_nger_corporate(2025)
  cer_nger_climate_active(x)
})
options(op)
```

`cer_nger_corporate` *NGER corporate emissions and energy data*

Description

Controlling-corporation-level Scope 1 and Scope 2 emissions and energy use from the National Greenhouse and Energy Reporting ('NGER') scheme.

Usage

```
cer_nger_corporate(year = 2025)
```

Arguments

`year`	Integer. Reporting year ending 30 June.

Details

Published annually by the CER on 28 February. Facility-level Scope 1 and 2 data is only published for the electricity sector; non-electricity sectors report at this controlling-corporation level only.

Value

A `cer_tbl` with one row per reporting corporation.

Source

Clean Energy Regulator, NGER reporting data and registers: https://cer.gov.au/markets/reports-and-data/nger-reporting-data-and-registers. Licensed under CC BY 4.0.

References

Commonwealth of Australia. *National Greenhouse and Energy Reporting Act 2007*; *National Greenhouse and Energy Reporting (Measurement) Determination 2008* (as amended annually).

Greenhouse Gas Protocol (2004). *A Corporate Accounting and Reporting Standard (Revised Edition)*. World Resources Institute.

Department of Climate Change, Energy, the Environment and Water (annual). *NGER Technical Guidelines*. Methodology for Scope 1 and Scope 2 calculations.

See Also

Other nger: cer_nger_climate_active(), cer_nger_electricity(), cer_nger_scope()

Examples

```
op <- options(cer.cache_dir = tempdir())
try({
  x <- cer_nger_corporate(year = 2025)
  head(x)
})
options(op)
```

---

cer_nger_electricity *NGER electricity sector facility emissions*

Description

Facility-level Scope 1 emissions and generated electricity for facilities in the electricity generation sector that met the NGER reporting threshold.

Usage

```
cer_nger_electricity(year = 2025, facility = NULL)
```

Arguments

year	Integer. Reporting year ending 30 June.
facility	Optional character vector of facility name substrings (case-insensitive).

Value

A cer_tbl with one row per generator facility.

Source

Clean Energy Regulator, National Greenhouse and Energy Register: https://cer.gov.au/markets/reports-and-data/nger-reporting-data-and-registers. Licensed under CC BY 4.0.

See Also

Other nger: cer_nger_climate_active(), cer_nger_corporate(), cer_nger_scope()

Examples

```
op <- options(cer.cache_dir = tempdir())
try({
  e <- cer_nger_electricity(year = 2025, facility = "Bayswater")
  head(e)
})
options(op)
```

`cer_nger_scope` *Select NGER columns by scope with explicit errors*

Description

Given an NGER corporate-level data frame from `cer_nger_corporate()`, select the requested scope columns and error if they are not present. Avoids the silent-misaggregation failure mode where a user sums a column that is actually Scope 1+2 combined when they intended Scope 1 only.

Usage

```
cer_nger_scope(
  x,
  scope = c("1_plus_2", "1", "2_market", "2_location"),
  col_pattern = NULL
)
```

Arguments

`x`	A `cer_tbl` or data frame from `cer_nger_corporate()`.
`scope`	One of `"1_plus_2"` (default), `"1"`, `"2_market"`, or `"2_location"`.
`col_pattern`	Optional regex override for the target column.

Details

Column-name detection is heuristic because NGER renames columns year to year. Pass `col_pattern` to override.

Value

A data frame with an `emissions_t_co2e` column populated by the requested scope and a `scope` column recording which scope was selected.

See Also

Other nger: `cer_nger_climate_active()`, `cer_nger_corporate()`, `cer_nger_electricity()`

Examples

```
op <- options(cer.cache_dir = tempdir())
try({
  x <- cer_nger_corporate(2025)
  cer_nger_scope(x, scope = "1")
})
options(op)
```

`cer_qcmr` *Quarterly Carbon Market Report data workbook*

Description

Aggregate carbon market statistics from the CER's Quarterly Carbon Market Report, including ACCU issuances, LGC creations, STC volumes, unit surrenders, and Safeguard Mechanism developments.

Usage

```
cer_qcmr(quarter = "latest", sheet = "Contents")
```

Arguments

`quarter`	Character. Quarter identifier in `"YYYYqN"` form (e.g. `"2025q4"` for the December 2025 quarter). Defaults to `"latest"`, which resolves to the most recent release.
`sheet`	Sheet name or integer index. The CER QCMR workbook contains a `Contents` sheet (default) plus one sheet per figure (e.g. `"Figure 1.1"`, `"Figure 3.14"`). Pass a figure name to extract that figure's data.

Value

A `cer_tbl` with the requested sheet's contents.

Source

Clean Energy Regulator, Quarterly Carbon Market Reports: https://cer.gov.au/markets/reports-and-data/quarterly-carbon-market-reports. Licensed under CC BY 4.0.

References

Clean Energy Regulator (quarterly). *Quarterly Carbon Market Report*. Methodology notes accompany each release.

Clean Energy Regulator (2024). *Carbon Market Indicators: methodology and data dictionary*.

Examples

```
op <- options(cer.cache_dir = tempdir())
try({
  # Contents sheet lists all available figures
  toc <- cer_qcmr("latest")
  head(toc)
  # Fetch a specific figure
  fig <- cer_qcmr("latest", sheet = "Figure 1.1")
})
options(op)
```

---

cer_qcmr_reference *QCMR headline reference totals*

Description

Bundled quarterly Clean Energy Regulator QCMR headline figures for cross-validation of live-fetched aggregates.

Usage

```
cer_qcmr_reference(quarter = NULL, measure = NULL)
```

Arguments

quarter	Optional character filter (e.g. "2024-Q4").
measure	Optional character filter (e.g. "accu_cumulative_issuances").

Details

Every value is read from the QCMR data workbook by data-raw/qcmr_headline_totals.R, and the source column names the workbook vintage, figure and column. Measures: accu_issuances, accu_cumulative_issuances, accu_projects_registered, accu_cumulative_projects_registered, safeguard_accu_surrenders, safeguard_smc_surrenders, lgc_validations, lgc_cumulative_validations, sres_solar_installations and sres_cumulative_solar_installations. Cumulative measures are running sums of the quarterly figures over the history the workbook covers.

Value

A cer_tbl of reference totals.

See Also

Other reconciliation: cer_reconcile()

Examples

```
cer_qcmr_reference()
cer_qcmr_reference(quarter = "2024-Q4")
```

`cer_reconcile` *Reconcile an aggregate against the QCMR headline figure*

Description

Compares a scalar or data-frame sum against the CER Quarterly Carbon Market Report headline for the same quarter and measure. Warns on absolute percentage gap > 2% (default). Small gaps (sub-1%) are expected because ACCU project registers are retroactively revised between QCMR releases.

Usage

```
cer_reconcile(value, quarter, measure, sum_column = NULL, warn_pct = 0.02)
```

Arguments

`value`	Numeric scalar or data frame. If a data frame, pass `sum_column` to pick which numeric column to sum.
`quarter`	Character, e.g. `"2024-Q4"` or `"2023-24"`.
`measure`	Character, matching a `measure` value in `cer_qcmr_reference()`. For example `"accu_cumulative_issuances"`, `"lgc_validations"`, `"safeguard_accu_surrenders"`.
`sum_column`	Column name to sum when `value` is a data frame.
`warn_pct`	Warn if |pct_diff| > this threshold. Default 0.02.

Value

A one-row data frame: `measure`, `quarter`, `value`, `reference`, `diff`, `pct_diff`, `unit`, `source`.

References

Clean Energy Regulator (quarterly). *Quarterly Carbon Market Report*. https://cer.gov.au/markets/reports-and-data/quarterly-carbon-market-reports

See Also

Other reconciliation: `cer_qcmr_reference()`

Examples

```
cer_reconcile(value = 158.4e6,
              quarter = "2024-Q4",
              measure = "accu_cumulative_issuances")
```

`cer_safeguard_baseline_trajectory` *Default Safeguard baseline emissions intensity trajectory*

Description

Returns the baseline emissions intensity, per unit of a production variable, for a facility whose baseline is set at the default emissions intensity in Schedule 1 of the Safeguard Mechanism Rule 2015 and the default emissions reduction contribution (ERC) in section 31:

Usage

```
cer_safeguard_baseline_trajectory(industry, from_year = 2023, to_year = 2030)
```

Arguments

`industry`	Character scalar: a `production_variable` from `cer_safeguard_industry_baselines()` (case-insensitive), or its Schedule 1 section number, e.g. `"7"` for aluminium.
`from_year`	Start year (default 2023 for FY 2023-24).
`to_year`	End year (default 2030 for FY 2030-31).

Details

$$B_t = EI \times ERC_t$$

where `ERC` is 0.951 for FY 2023-24 and falls by 0.049 a year (a linear decline of 4.9 percent of the default intensity) to 0.657 for FY 2029-30, then by 0.03285 a year, never below zero. Facilities using site-specific emissions intensities, and trade-exposed baseline-adjusted facilities (see `cer_safeguard_teba_facilities()`), have different baselines.

Value

A data frame with `year`, `financial_year`, `default_emissions_intensity`, `decline_factor` (the default ERC) and `baseline` (t CO2-e per unit of the production variable).

References

Commonwealth of Australia. *National Greenhouse and Energy Reporting (Safeguard Mechanism) Rule 2015* (F2015L01637), compilation of 31 August 2024: sections 31 to 33 and Schedule 1.

See Also

Other safeguard: `cer_safeguard_facilities()`, `cer_safeguard_industry_baselines()`, `cer_safeguard_teba_facil`

Examples

```
cer_safeguard_baseline_trajectory("Aluminium")
cer_safeguard_baseline_trajectory("63", from_year = 2023, to_year = 2035)
```

`cer_safeguard_facilities` *Safeguard Mechanism facility-level data*

Description

Covered emissions, baselines, Safeguard Mechanism Credits (SMCs) issued or surrendered, and abatement information for facilities above the 100,000 tonne CO2e threshold.

Usage

```
cer_safeguard_facilities(year = 2025)
```

Arguments

`year`	Integer. Reporting year ending 30 June (e.g. `2025` for 2024-25). Defaults to the most recently released year.

Details

Published annually by the CER on 15 April following each reporting year. The download URL is resolved dynamically from the CER's Safeguard landing page, so no slug changes here are required when new years publish.

Value

A `cer_tbl` with one row per facility.

Source

Clean Energy Regulator, Safeguard data: https://cer.gov.au/markets/reports-and-data/safeguard-data. Licensed under CC BY 4.0.

References

Commonwealth of Australia. *National Greenhouse and Energy Reporting Act 2007*.

Commonwealth of Australia. *National Greenhouse and Energy Reporting (Safeguard Mechanism) Rule 2015* (as amended).

Commonwealth of Australia (2023). *Safeguard Mechanism (Crediting) Amendment Act 2023*. Introduced declining industry baselines (4.9% p.a. nominal) and Safeguard Mechanism Credits (SMCs).

Climate Change Authority (2023). *Review of the Safeguard Mechanism*.

See Also

Other safeguard: `cer_safeguard_baseline_trajectory()`, `cer_safeguard_industry_baselines()`, `cer_safeguard_teba_facilities()`

Examples

```
op <- options(cer.cache_dir = tempdir())
try({
  f <- cer_safeguard_facilities(year = 2025)
  head(f)
})
options(op)
```

`cer_safeguard_industry_baselines` *Safeguard default emissions intensities by production variable*

Description

The default (industry-average) emissions intensity for every production variable in Schedule 1 of the Safeguard Mechanism Rule 2015 that has one, read from the Rule by `data-raw/safeguard_industry_baselines.R`. Production variables whose default is yet to be specified are omitted.

Usage

```
cer_safeguard_industry_baselines()
```

Value

A `cer_tbl` with columns `section`, `production_variable`, `default_emissions_intensity`, `unit` (the Rule's wording) and `source` (section and compilation date).

See Also

Other safeguard: `cer_safeguard_baseline_trajectory()`, `cer_safeguard_facilities()`, `cer_safeguard_teba_faci`

Examples

```
b <- cer_safeguard_industry_baselines()
b[b$production_variable %in% c("Aluminium", "Alumina"), ]
```

cer_safeguard_teba_facilities *Trade-exposed baseline-adjusted (TEBA) facility list*

Description

Returns the Clean Energy Regulator's table of trade-exposed baseline-adjusted (TEBA) determinations, bundled by `data-raw/safeguard_teba_facilities.R`. A TEBA determination sets a slower decline in a facility's emissions reduction contribution (ERC) than the default, for the financial years it covers. A facility appears once per determination, so a revoked and replaced determination gives two rows.

Usage

```
cer_safeguard_teba_facilities()
```

Value

A `cer_tbl` with columns `facility_name`, `responsible_emitter`, `start_date`, `end_date` (the financial years covered), `facility_type`, `decline_rate`, `erc_2023_24` to `erc_2026_27` (`NA` where the determination does not apply), `status` (`"current"` or `"historical"`) and `historical_reason`.

References

Clean Energy Regulator. *Trade-exposed baseline-adjusted facility data*. https://cer.gov.au/markets/reports-and-data/safeguard-data/trade-exposed-baseline-adjusted-facility-data

See Also

Other safeguard: `cer_safeguard_baseline_trajectory()`, `cer_safeguard_facilities()`, `cer_safeguard_industry_`

Examples

```
cer_safeguard_teba_facilities()
```

cer_sha256 *Compute the SHA-256 digest of a file*

Description

Compute the SHA-256 digest of a file

Usage

```
cer_sha256(file)
```

Arguments

file	Path to a local file.

Value

A length-1 character string (hex digest), or NA if the file does not exist. Uses digest::digest() when the digest package is available (recommended for research work), falling back to tools::md5sum() otherwise.

See Also

Other reproducibility: cer_cite(), cer_deposit_zenodo(), cer_legislative_instrument(), cer_manifest(), cer_manifest_clear(), cer_manifest_write(), cer_snapshot()

Examples

```
f <- tempfile()
writeLines("hello", f)
cer_sha256(f)
```

---

cer_snapshot *Pin or inspect the session snapshot date*

Description

Call once at the top of an analysis script to declare the vintage of CER data you intend to use. Every subsequent cer_* fetch records this date in the cer_tbl provenance header, in cer_manifest() entries, and in cer_cite() output.

Usage

```
cer_snapshot(date)
```

Arguments

date	ISO "YYYY-MM-DD" character, Date, or POSIXct. Pass NULL to clear.

Details

This matters more for CER data than for most government data: projects relinquish ACCUs retroactively, the CER transfers projects between proponents, methods are reclassified (e.g. post-Chubb-Review status changes), and the Quarterly Carbon Market Report occasionally restates prior quarters. A snapshot pin is the minimum evidence a reviewer needs to verify a published carbon-market claim.

Value

Invisibly, the new pinned date (as Date), or NULL.

See Also

Other reproducibility: cer_cite(), cer_deposit_zenodo(), cer_legislative_instrument(), cer_manifest(), cer_manifest_clear(), cer_manifest_write(), cer_sha256()

Examples

```
cer_snapshot("2026-04-24")
cer_snapshot()
cer_snapshot(NULL)
```

---

cer_sres_installations

SRES small-scale installations by postcode

Description

Monthly Small-scale Renewable Energy Scheme ('SRES') installation counts or capacity by postcode. Covers rooftop solar PV, solar water heaters, air-source heat pumps, and small batteries.

Usage

```
cer_sres_installations(
  technology = c("solar_pv", "swh", "heat_pump", "battery"),
  postcode = NULL,
  measure = c("installations", "capacity")
)
```

Arguments

technology	Character. One of "solar_pv" (default), "swh" (solar water heater), "heat_pump", or "battery".
postcode	Optional character vector of 4-digit Australian postcodes to filter on.
measure	Character. "installations" (default, counts) or "capacity" (total kW installed).

Value

A cer_tbl. For measure = "installations" columns include postcode and per-month install counts. For measure = "capacity", per-month kW capacity.

Source

Clean Energy Regulator, Small-scale installation postcode data: https://cer.gov.au/markets/reports-and-data/small-scale-installation-postcode-data. Licensed under CC BY 4.0.

References

Commonwealth of Australia. *Renewable Energy (Electricity) Act 2000*, Small-scale Renewable Energy Scheme provisions.

Australian Energy Market Operator (2024). *Distributed PV Forecasting Methodology*. AEMO uses SRES installation data as input to ISP and ESOO forecasts.

Best, R., Burke, P.J. and Nishitateno, S. (2019). "Understanding the determinants of rooftop solar installation: evidence from household surveys in Australia." *Australian Journal of Agricultural and Resource Economics*, 63(4), 922-939. doi:10.1111/14678489.12319

Examples

```
op <- options(cer.cache_dir = tempdir())
inst <- cer_sres_installations(postcode = c("2000", "3000"))
head(inst)
options(op)
```

---

cer_to_carbondata *Remap an ACCU project data frame to the carbondata voluntary-project schema*

Description

The `carbondata` R package exposes Verra, Gold Standard, ACR, and CAR voluntary projects under a unified schema. This function renames ACCU register columns to match, so an analyst can compare ACCU issuance patterns against international voluntary markets on a consistent basis.

Usage

```
cer_to_carbondata(x)
```

Arguments

`x`	A `cer_tbl` from `cer_accu_projects()`.

Details

Schema mapping is lossy: ACCU-specific fields (Chubb flag, permanence period, crediting-period dates) pass through with original names.

Value

A data frame with carbondata-style columns: `project_id`, `project_name`, `methodology`, `project_type`, `country`, `registry`, `issued_credits_to_date`, `retired_credits_to_date`.

See Also

Other interop: `cer_international_comparator()`

Examples

```
op <- options(cer.cache_dir = tempdir())
try({
  ccy <- cer_accu_projects()
  carb <- cer_to_carbondata(ccy)
  head(carb)
})
options(op)
```

`print.cer_tbl` *Print a cer_tbl*

Description

Prints a provenance header (title, source URL, licence, retrieval time, dimensions) followed by the data frame.

Usage

```
## S3 method for class 'cer_tbl'
print(x, ...)
```

Arguments

`x`	A `cer_tbl` object.
`...`	Passed to the next method.

Value

Invisibly returns `x`.

Examples

```
x <- data.frame(project_id = "ERF123", accus_issued = 4200)
x <- structure(x, cer_title = "Demo", cer_source = "https://cer.gov.au",
               cer_licence = "CC BY 4.0", cer_retrieved = Sys.time(),
               class = c("cer_tbl", "data.frame"))
print(x)
```

Index

∗ **accu**
cer_accu_aggregate, 3
cer_accu_contracts, 4
cer_accu_issuances, 4
cer_accu_methods, 5
cer_accu_projects, 6
cer_accu_relinquishments, 8
cer_method_determination, 17
cer_method_integrity, 18
∗ **configuration**
cer_cache_info, 9
cer_clear_cache, 10
∗ **interop**
cer_international_comparator, 12
cer_to_carbondata, 32
∗ **lret**
cer_lgc_power_stations, 14
∗ **nger**
cer_nger_climate_active, 19
cer_nger_corporate, 20
cer_nger_electricity, 21
cer_nger_scope, 22
∗ **qcmr**
cer_qcmr, 23
∗ **reconciliation**
cer_qcmr_reference, 24
cer_reconcile, 25
∗ **reproducibility**
cer_cite, 9
cer_deposit_zenodo, 11
cer_legislative_instrument, 13
cer_manifest, 15
cer_manifest_clear, 16
cer_manifest_write, 16
cer_sha256, 29
cer_snapshot, 30
∗ **safeguard**
cer_safeguard_baseline_trajectory, 26
cer_safeguard_facilities, 27
cer_safeguard_industry_baselines, 28
cer_safeguard_teba_facilities, 29
∗ **sres**
cer_sres_installations, 31

cer_accu_aggregate, 3
cer_accu_aggregate(), *4–8*, *17–19*
cer_accu_contracts, 4
cer_accu_contracts(), *3*, *5–8*, *17*, *19*
cer_accu_issuances, 4
cer_accu_issuances(), *3, 4*, *6–8*, *17*, *19*
cer_accu_methods, 5
cer_accu_methods(), *3–5*, *7, 8*, *17*, *19*
cer_accu_projects, 6
cer_accu_projects(), *3–6*, *8*, *17–19*, *32*
cer_accu_relinquishments, 8
cer_accu_relinquishments(), *3–7*, *17*, *19*
cer_cache_info, 9
cer_cache_info(), *11*
cer_cite, 9
cer_cite(), *12, 13*, *15–17*, *30, 31*
cer_clear_cache, 10
cer_clear_cache(), *7*, *9*
cer_deposit_zenodo, 11
cer_deposit_zenodo(), *9, 10*, *13*, *15–17*, *30, 31*
cer_international_comparator, 12
cer_international_comparator(), *32*
cer_legislative_instrument, 13
cer_legislative_instrument(), *10*, *12*, *15–17*, *30, 31*
cer_lgc_power_stations, 14
cer_manifest, 15
cer_manifest(), *10–13*, *16, 17*, *30, 31*
cer_manifest_clear, 16
cer_manifest_clear(), *10*, *12, 13*, *15*, *17*, *30, 31*
cer_manifest_write, 16

cer_manifest_write(), *10*, *12, 13*, *15, 16*, *30, 31*
cer_method_determination, 17
cer_method_determination(), *3–8*, *19*
cer_method_integrity, 18
cer_method_integrity(), *3–8*, *17*
cer_nger_climate_active, 19
cer_nger_climate_active(), *21, 22*
cer_nger_corporate, 20
cer_nger_corporate(), *19*, *21, 22*
cer_nger_electricity, 21
cer_nger_electricity(), *19*, *21, 22*
cer_nger_scope, 22
cer_nger_scope(), *19*, *21*
cer_qcmr, 23
cer_qcmr_reference, 24
cer_qcmr_reference(), *25*
cer_reconcile, 25
cer_reconcile(), *24*
cer_safeguard_baseline_trajectory, 26
cer_safeguard_baseline_trajectory(), *27–29*
cer_safeguard_facilities, 27
cer_safeguard_facilities(), *26*, *28, 29*
cer_safeguard_industry_baselines, 28
cer_safeguard_industry_baselines(), *26, 27*, *29*
cer_safeguard_teba_facilities, 29
cer_safeguard_teba_facilities(), *26–28*
cer_sha256, 29
cer_sha256(), *10*, *12, 13*, *15–17*, *31*
cer_snapshot, 30
cer_snapshot(), *9–13*, *15–17*, *30*
cer_sres_installations, 31
cer_to_carbondata, 32
cer_to_carbondata(), *13*

print.cer_tbl, 33