Package ‘okBATHTUB’

June 24, 2026

Title Empirical Reservoir Eutrophication Modelling with Oklahoma Calibration

Version 0.1.11

Description Empirical reservoir water quality modelling using Walker's 'BATHTUB' Model 1 (second-order available-phosphorus sedimentation) from Walker (1985) <https://hdl.handle.net/11681/13884> and Walker (1996) <https://hdl.handle.net/11681/4353> as the default retention model. The Vollenweider (1976) hydraulic-residence form and the equivalent formulation of Larsen and Mercier (1976) are available as alternatives. Predicts in-lake total phosphorus, total nitrogen, chlorophyll-a, and Secchi depth from tributary nutrient and hydraulic loading inputs, and computes Carlson (1977) <doi:10.4319/lo.1977.22.2.0361> Trophic State Indices. Optional Oklahoma-specific chlorophyll and Secchi regression coefficients are provided, calibrated from publicly available state lake monitoring data. Supports single-segment and multi-segment reservoir configurations and load-reduction scenario analysis. Designed to complement watershed loading models such as the Soil and Water Assessment Tool ('SWAT'; <https://swat.tamu.edu>) and the U.S. EPA Hydrologic and Water Quality System ('HAWQS'; <https://hawqs.tamu.edu>) in a two-model nutrient management workflow.

License MIT + file LICENSE

Encoding UTF-8

Depends R (>= 4.1.0)

Imports utils

Suggests ggplot2 (>= 3.4.0), dplyr (>= 1.1.0), tidyr (>= 1.3.0), stringr (>= 1.5.0), knitr, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

VignetteBuilder knitr

LazyData true

URL https://0011235813.github.io/Oklahoma-BATHTUB/, https://github.com/0011235813/Oklahoma-BATHTUB

BugReports https://github.com/0011235813/Oklahoma-BATHTUB/issues

Config/roxygen2/version 8.0.0

NeedsCompilation no

Author Jordon Henderson [aut, cre, cph]

Maintainer Jordon Henderson <jordon.henderson@owrb.ok.gov>

Repository CRAN

Date/Publication 2026-06-24 18:50:02 UTC

Contents

`ok_hydraulics` *Compute reservoir hydraulic characteristics*

Description

`ok_hydraulics()` takes the inflow volume from `ok_load()` and the reservoir's morphometric parameters to compute two quantities that drive nutrient retention in all subsequent steps:

- **Hydraulic residence time** (`tau`, yr): reservoir volume divided by annual outflow, where volume is approximated as `mean_depth * surface_area`.
- **Areal water load** (`q_s`, m/yr): inflow volume divided by surface area. The primary driver of settling-velocity based retention.

Usage

```
ok_hydraulics(x, surface_area_ha, mean_depth_m, outflow_m3yr = NULL)
```

Arguments

`x`	An `okBATHTUB` object produced by `ok_load()`.
`surface_area_ha`	Numeric. Reservoir surface area at normal pool (ha). Must be positive.
`mean_depth_m`	Numeric. Mean reservoir depth at normal pool (m). Must be positive. For Oklahoma reservoirs this is typically 2-10 m.
`outflow_m3yr`	Numeric. Annual outflow volume (m^3/yr). If `NULL` (default), outflow is assumed equal to inflow (steady-state water balance). Supply an explicit value when significant evaporation, diversion, or storage change alters the water balance.

Value

An `okBATHTUB` object at pipeline step `"hydraulics"`.

Volume approximation

Reservoir volume is computed as `mean_depth * surface_area`, treating the reservoir as a right rectangular prism with flat bottom. This is a simplification: real reservoirs have varying bathymetry, and the relationship `V = Z * A` is exact only when `Z` is the volume-weighted mean depth, which is what bathymetric surveys typically report. If you only have maximum depth or a depth-area regression estimate, expect roughly a factor-of-1.5 uncertainty in `tau` and proportional downstream uncertainty in predicted in-lake concentrations.

See Also

`ok_load()`, `ok_retention()`

Examples

```
result <- ok_load(
  inflow_m3yr   = 45e6,
  tp_inflow_ugl = 120,
  tn_inflow_ugl = 1800
) |>
  ok_hydraulics(surface_area_ha = 890, mean_depth_m = 4.2)
print(result)
```

---

ok_inlake — *Predict in-lake water quality concentrations*

Description

`ok_inlake()` applies nutrient retention coefficients from `ok_retention()` to predict in-lake total phosphorus (TP) and total nitrogen (TN) concentrations via the mass balance

$$C_{lake} = C_{in} \times (1 - R)$$

then uses empirical log-log regression to predict chlorophyll-a from in-lake TP and Secchi depth from chlorophyll-a.

Note on retention identity: when `coefficients = "walker"` (Walker BATHTUB Model 1), the retention coefficient stored by `ok_retention()` is back-calculated from Walker's quadratic mass balance solution so that `C_lake = C_in * (1 - R)` exactly reproduces the Model 1 result.

Usage

```
ok_inlake(x, predict_chla = TRUE, predict_secchi = TRUE)
```

Arguments

`x`	An `okBATHTUB` object produced by `ok_retention()`.
`predict_chla`	Logical. Whether to predict chlorophyll-a from in-lake TP. Default `TRUE`.
`predict_secchi`	Logical. Whether to predict Secchi depth from chlorophyll-a. Requires `predict_chla = TRUE`. Default `TRUE`.

Value

An `okBATHTUB` object at pipeline step `"inlake"`.

Chlorophyll-a from TP

Log-log linear regression:

$$\log_{10}(\text{Chl-a}) = a + b \times \log_{10}(\text{TP}_{lake})$$

Default coefficients are Walker's nationally-derived values ($a = -1.136$, $b = 1.449$); Oklahoma ecoregion-specific values are applied when `coefficients = "oklahoma"`.

Secchi depth from chlorophyll-a

$$\log_{10}(\text{Secchi}) = a + b \times \log_{10}(\text{Chl-a})$$

Default Walker national: $a = 0.616$, $b = -0.473$.

In high-turbidity Oklahoma reservoirs, Secchi depth is often controlled more by inorganic suspended sediment than by algal biomass. This is partly captured by the Oklahoma ecoregion-specific Secchi regressions, but for reservoirs with very high non-algal turbidity (e.g. central and western Oklahoma), Secchi predictions should be interpreted with caution.

See Also

`ok_retention()`, `ok_tsi()`

Examples

```
result <- ok_load(
  inflow_m3yr   = 45e6,
  tp_inflow_ugl = 120,
  tn_inflow_ugl = 1800
) |>
  ok_hydraulics(surface_area_ha = 890, mean_depth_m = 4.2) |>
  ok_retention() |>
  ok_inlake()
print(result)
```

---

ok_lake_ecoregion — *Look up an Oklahoma lake's EPA Level III ecoregion*

Description

Convenience wrapper around `ok_lake_ecoregions` for retrieving the ecoregion assignment for one or more lakes. Always returns a data frame so that the return type is stable across single-match and multi-match results.

To get a bare ecoregion name string for use in `ok_load()`, use:

```
eco <- ok_lake_ecoregion("Arcadia Lake", exact = TRUE)$eco_l3_name
```

Usage

```
ok_lake_ecoregion(lake_name, exact = FALSE, simplify = NULL)
```

Arguments

- `lake_name` — Character. One or more lake names to look up. Partial matching is supported (case-insensitive) unless `exact` is `TRUE`.
- `exact` — Logical. If `TRUE`, requires an exact name match (case-insensitive). Default `FALSE`.
- `simplify` — Deprecated. Retained for backward compatibility with pre-0.1.3 callers; ignored with a deprecation warning. The function always returns a data frame.

Value

A data frame of zero or more rows from `ok_lake_ecoregions` with all columns. An unmatched lookup returns an empty data frame (with the correct column structure) plus a warning.

See Also

`ok_lake_ecoregions`, `ok_reservoir()`

Examples

```
# Standard usage: exact match, extract ecoregion name
eco <- ok_lake_ecoregion("Arcadia Lake", exact = TRUE)$eco_l3_name
eco

# Partial match: returns a data frame of all matches
ok_lake_ecoregion("Tenkiller")

# Use in pipeline
eco <- ok_lake_ecoregion("Arcadia Lake", exact = TRUE)$eco_l3_name
if (length(eco) == 1L && !is.na(eco)) {
  ok_load(inflow_m3yr   = 45e6,
          tp_inflow_ugl = 120,
          coefficients  = "oklahoma",
          ecoregion     = eco)
}
```

`ok_lake_ecoregions` *Oklahoma lakes -> EPA Level III ecoregion lookup*

Description

A data frame mapping 138 Oklahoma lakes to their EPA Level III ecoregion, with monitoring coverage statistics from a 1998-2023 snapshot of publicly available state lake monitoring records. Useful for quickly resolving an ecoregion for use with `coefficients = "oklahoma"`, or for understanding which lakes have sufficient monitoring history to support empirical analysis.

Usage

```
ok_lake_ecoregions
```

Format

A data frame with one row per lake and the following columns:

lake_name Character. Lake name as it appeared in source monitoring records.

eco_l3_code Character. EPA Level III ecoregion numeric code (as string, `US_L3CODE`), e.g. `"29"` for Cross Timbers.

eco_l3_name Character. EPA Level III ecoregion name (`US_L3NAME`).

n_sites_total Integer. Total number of monitoring stations on this lake in the source dataset.

n_sites_tier1 Integer. Number of stations with at least 3 years each of TP, chlorophyll-a, and Secchi data (the core calibration parameters). Always <= `n_sites_total`.

latitude Numeric. Approximate lake centroid latitude (WGS84 decimal degrees), computed as the mean of monitoring station coordinates.

longitude Numeric. Approximate lake centroid longitude.

max_yrs_tp Integer. Maximum number of calendar years (in the 1998-2023 window) with any reported total phosphorus data at any station on the lake.

max_yrs_chla Integer. Same, for chlorophyll-a.

max_yrs_secchi Integer. Same, for Secchi depth.

max_yrs_tn Integer. Same, for total nitrogen.

Coverage statistics caveats

The `n_sites_*` and `max_yrs_*` columns reflect a 1998-2023 snapshot taken when the calibration was performed. They are **not** updated automatically with new monitoring data. Treat them as a useful starting point for assessing data availability, not as a current inventory. For up-to-date monitoring coverage, query the source monitoring system directly.

Ecoregion assignment

Ecoregion assignment uses EPA Level III boundaries (Griffith et al. 2004) applied by spatial join of each lake's monitoring-station coordinates against the bundled ecoregion polygons; each lake takes the ecoregion of the majority of its stations, with a nearest-feature fallback for stations just outside a polygon edge. All lakes in the current dataset are assigned an ecoregion. Lakes in ecoregions without an ecoregion-specific coefficient set use the statewide pooled regressions when modelling with `coefficients = "oklahoma"`.

Source

Compiled from publicly available Oklahoma lake monitoring records (1998-2023). Ecoregion polygons from Griffith, G.E. et al. (2004), *Ecoregions of Oklahoma*, U.S. Geological Survey, Reston, Virginia. See `data-raw/ok_ecoregion_assignment.R` in the package source for the assignment script.

See Also

`ok_lake_ecoregion()`, `ok_reservoirs`

Examples

```
# All lakes in a specific ecoregion
head(ok_lake_ecoregions[ok_lake_ecoregions$eco_l3_name == "Cross Timbers", ])

# Lakes with the longest monitoring history
top_monitored <- ok_lake_ecoregions[
  order(-ok_lake_ecoregions$max_yrs_tp), ][1:10, ]
top_monitored[, c("lake_name", "eco_l3_name", "max_yrs_tp")]
```

---

ok_load *Assemble tributary load inputs*

Description

ok_load() is the entry point for the okBATHTUB pipeline. It accepts tributary hydraulic and nutrient loading data, validates inputs, resolves the coefficient set, and returns an okBATHTUB object ready to pass into ok_hydraulics().

All concentration inputs are volume-flow-weighted means representing the period of analysis (typically an annual or seasonal average). If multiple tributaries contribute to the reservoir, either aggregate them manually before calling ok_load(), or use ok_load_multi() to supply tributary data as a data frame.

Usage

```
ok_load(
  inflow_m3yr,
  tp_inflow_ugl,
  tn_inflow_ugl = NULL,
  tss_inflow_mgl = NULL,
  segment_label = "main",
  coefficients = "walker",
  ecoregion = NULL
)
```

Arguments

inflow_m3yr Numeric. Total annual tributary inflow volume (m^3/yr). Must be positive.

tp_inflow_ugl Numeric. Flow-weighted mean total phosphorus concentration of tributary inflow (ug/L). Must be non-negative.

tn_inflow_ugl Numeric. Flow-weighted mean total nitrogen concentration of tributary inflow (ug/L). Optional. Default NULL.

tss_inflow_mgl Numeric. Flow-weighted mean total suspended solids concentration (mg/L). Optional. Default NULL.

segment_label Character. Optional label for this reservoir segment (e.g. "riverine", "lacustrine"). Default "main".

`coefficients`	One of `"walker"` (default, Walker BATHTUB Model 1), `"vollenweider"` (Vollenweider 1976 / Larsen-Mercier 1976), `"oklahoma"` (Walker Model 1 retention plus Oklahoma-specific Chl-a / Secchi regressions), or a named list of custom coefficients.
`ecoregion`	Character. EPA Level III ecoregion name (e.g. `"Cross Timbers"`). Used only when `coefficients = "oklahoma"`; silently ignored otherwise (with a message if the combination is suspicious). Default `NULL`.

Value

An `okBATHTUB` object at pipeline step `"load"`.

See Also

`ok_load_multi()`, `ok_hydraulics()`

Examples

```
# Minimum required inputs (TP only)
result <- ok_load(inflow_m3yr = 45e6, tp_inflow_ugl = 120)
print(result)

# Full inputs with TN and TSS
result <- ok_load(
  inflow_m3yr    = 45e6,
  tp_inflow_ugl  = 120,
  tn_inflow_ugl  = 1800,
  tss_inflow_mgl = 35,
  segment_label  = "lacustrine"
)

# Oklahoma ecoregion-specific coefficients
result <- ok_load(
  inflow_m3yr   = 45e6,
  tp_inflow_ugl = 120,
  coefficients  = "oklahoma",
  ecoregion     = "Cross Timbers"
)
```

`ok_load_multi` *Assemble tributary loads from multiple tributaries*

Description

A convenience wrapper around `ok_load()` for reservoirs with more than one tributary. Accepts a data frame of tributary data, computes flow-weighted mean concentrations, sums inflow volumes, and calls `ok_load()` with the aggregated values.

Usage

```
ok_load_multi(
  tributaries,
  segment_label = "main",
  coefficients = "walker",
  ecoregion = NULL
)
```

Arguments

`tributaries`	A data frame with one row per tributary and these columns: `inflow_m3yr` Annual inflow volume (m^3/yr). Required. `tp_inflow_ugl` Flow-weighted mean TP (ug/L). Required. `tn_inflow_ugl` Flow-weighted mean TN (ug/L). Optional. `tss_inflow_mgl` Flow-weighted mean TSS (mg/L). Optional.
`segment_label`	Character. Segment label passed to `ok_load()`. Default `"main"`.
`coefficients`	Coefficient set. Default `"walker"`.
`ecoregion`	EPA Level III ecoregion name. Default `NULL`. Passed through to `ok_load()`.

Value

An okBATHTUB object at pipeline step `"load"`.

See Also

`ok_load()`

Examples

```
tribs <- data.frame(
  inflow_m3yr   = c(30e6, 15e6),
  tp_inflow_ugl = c(110,  145),
  tn_inflow_ugl = c(1600, 2100)
)
result <- ok_load_multi(tribs)
```

`ok_plot_response` *Plot a load-response curve for a reservoir*

Description

`ok_plot_response()` generates a load-response curve showing how predicted in-lake chlorophyll-a, Secchi depth, or mean TSI responds to a range of inflow total phosphorus concentrations. This is the core planning tool for answering "what inflow TP concentration achieves a given trophic state target?"

The curve is generated by running the full okBATHTUB pipeline across a sequence of TP concentrations while holding all other parameters constant at the baseline values.

Usage

```
ok_plot_response(
  baseline,
  response = c("chla", "secchi", "tsi", "tp_inlake"),
  tp_range = c(10, 300),
  n_points = 100L,
  target_tsi = NULL,
  target_class = NULL,
  show_trophic_bands = NULL,
  current_tp = NULL,
  lake_name = NULL
)
```

Arguments

baseline	An okBATHTUB object that has been run through at least ok_hydraulics(). Morphometry and coefficient set are taken from this object.
response	One of "chla" (chlorophyll-a, ug/L), "secchi" (Secchi depth, m), "tsi" (mean Carlson TSI), or "tp_inlake" (in-lake TP, ug/L). Default "chla".
tp_range	Numeric vector of length 2. Range of inflow TP concentrations to plot (ug/L). Default c(10, 300).
n_points	Integer. Number of points along the curve. Default 100.
target_tsi	Numeric. Optional horizontal reference line showing a TSI target. Only shown when response = "tsi".
target_class	Character. Trophic class target. Sets target_tsi automatically. One of "oligotrophic", "mesotrophic", "eutrophic".
show_trophic_bands	Logical. Show coloured trophic state background bands on TSI plots. Default TRUE when response is "tsi".
current_tp	Numeric. Optional vertical reference line showing the current inflow TP concentration.
lake_name	Character. Lake name for plot title. Default NULL.

Value

A ggplot2 object.

See Also

ok_scenario, ok_plot_scenario

Examples

```
baseline <- ok_load(
  inflow_m3yr   = 45e6,
  tp_inflow_ugl = 120,
  coefficients  = "oklahoma",
```

```
  ecoregion     = "Cross Timbers"
) |>
ok_hydraulics(surface_area_ha = 890, mean_depth_m = 4.2)

ok_plot_response(baseline, response = "tsi",
                 target_class = "mesotrophic",
                 current_tp   = 120,
                 lake_name    = "Arcadia Lake")
```

---

ok_plot_scenario — *Plot scenario comparison from ok_scenario() output*

Description

ok_plot_scenario() visualises the output of ok_scenario() or ok_scenario_sweep() as a faceted dot-and-line chart, showing how key water quality metrics change across loading scenarios.

Usage

```
ok_plot_scenario(
  scenario_result,
  metrics = c("tp_inlake_ugl", "chla_ugl", "secchi_m", "tsi_mean"),
  highlight_target = TRUE,
  lake_name = NULL
)
```

Arguments

scenario_result	A data frame returned by ok_scenario() or ok_scenario_sweep().
metrics	Character vector. Which metrics to plot. Any subset of c("tp_inlake_ugl", "chla_ugl", "secchi_m", "tsi_mean"). Default: all four.
highlight_target	Logical. If the scenario result contains a meets_target column, highlight scenarios that meet the target in green. Default TRUE.
lake_name	Character. Lake name for the plot title.

Value

A ggplot2 object.

See Also

ok_scenario, ok_scenario_sweep

Examples

```
baseline <- ok_load(inflow_m3yr = 45e6, tp_inflow_ugl = 120,
                     coefficients = "oklahoma",
                     ecoregion    = "Cross Timbers") |>
  ok_hydraulics(surface_area_ha = 890, mean_depth_m = 4.2)

sweep <- ok_scenario_sweep(baseline, target_class = "mesotrophic")
ok_plot_scenario(sweep, lake_name = "Arcadia Lake")
```

---

ok_plot_segments *Plot a longitudinal water quality profile across reservoir segments*

Description

`ok_plot_segments()` visualises the spatial gradient in water quality from riverine to lacustrine zones of a multi-segment reservoir, using the output of `ok_segment_summary()` or `ok_segment_chain()`.

Usage

```
ok_plot_segments(
  segment_data,
  metrics = c("tp_inlake_ugl", "chla_ugl", "secchi_m", "tsi_mean"),
  lake_name = NULL
)
```

Arguments

`segment_data`	Either a named list of `okBATHTUB` objects from `ok_segment_chain()`, or a data frame from `ok_segment_summary()`.
`metrics`	Character vector. Which metrics to plot. Any subset of `c("tp_inlake_ugl", "chla_ugl", "secchi_m", "tsi_mean","tp_retention_pct")`. Default: first four.
`lake_name`	Character. Lake name for the plot title.

Value

A `ggplot2` object.

See Also

`ok_segment_chain`, `ok_segment_summary`

Examples

```
segs <- list(
  list(label = "Riverine",     surface_area_ha = 280, mean_depth_m = 3.1),
  list(label = "Transitional", surface_area_ha = 410, mean_depth_m = 4.5),
  list(label = "Lacustrine",   surface_area_ha = 610, mean_depth_m = 5.8)
)
chain <- ok_segment_chain(45e6, 150, tn_inflow_ugl = 2200, segments = segs)
ok_plot_segments(chain, lake_name = "Example Reservoir")
```

---

ok_plot_tsi *Plot a Carlson TSI deviation diagram*

Description

ok_plot_tsi() produces a Carlson (1977) TSI deviation diagram, plotting TSI(TP) on the x-axis against TSI(Chl-a) and TSI(Secchi) on the y-axis with trophic state background bands. Deviations from the 1:1 line indicate light limitation (Chl-a below TP line) or non-algal turbidity (Secchi below Chl-a line).

Can accept either a single okBATHTUB result, or a data frame containing the required TSI columns (e.g. from ok_scenario() or a user-supplied observed-data summary).

Usage

```
ok_plot_tsi(x, color_by = NULL, lake_name = NULL, show_diagonal = TRUE)
```

Arguments

x	An okBATHTUB object, or a data frame containing columns tsi_tp, tsi_chla, and optionally tsi_secchi. If a data frame, it may also contain lake_name and eco_l3_name for grouping.
color_by	Character. Column name in a data frame to color points by. Common choices: "eco_l3_name", "trophic_state", "lake_name". Ignored when x is an okBATHTUB object.
lake_name	Character. Lake name for the plot title when x is an okBATHTUB object.
show_diagonal	Logical. Show the 1:1 reference line. Default TRUE.

Value

A ggplot2 object.

References

Carlson, R.E. (1977). A trophic state index for lakes. *Limnology and Oceanography*, 22(2), 361-369.

See Also

ok_tsi, ok_tsi_observed

Examples

```
# From a single pipeline result
result <- ok_load(inflow_m3yr = 45e6, tp_inflow_ugl = 120,
                  coefficients = "oklahoma",
                  ecoregion    = "Cross Timbers") |>
  ok_hydraulics(surface_area_ha = 890, mean_depth_m = 4.2) |>
  ok_retention() |>
  ok_inlake()    |>
  ok_tsi()
ok_plot_tsi(result, lake_name = "Arcadia Lake")
```

---

ok_reservoir *Look up Oklahoma reservoir morphometry*

Description

ok_reservoir() retrieves morphometric parameters for one or more Oklahoma reservoirs from the bundled ok_reservoirs dataset. Returns a data frame that can be used directly with ok_hydraulics().

Usage

```
ok_reservoir(
  lake_name = NULL,
  exact = FALSE,
  ecoregion = NULL,
  data_quality = c("A", "B")
)
```

Arguments

lake_name	Character. One or more lake names to look up. Partial matching is supported - "Arcadia" will match "Arcadia Lake". Case-insensitive. If NULL, returns all reservoirs (subject to other filters).
exact	Logical. If TRUE, requires exact name match (case-insensitive). Default FALSE (partial matching).
ecoregion	Character. Filter results to a specific EPA Level III ecoregion name. Default NULL (no filter).
data_quality	Character vector. Filter to specific data quality codes. Default c("A", "B") (all).

Value

A data frame with one row per matched reservoir containing all columns from ok_reservoirs.

See Also

ok_reservoirs, ok_hydraulics()

Examples

```
# Look up a single lake (partial match)
ok_reservoir("Arcadia")

# Exact match
ok_reservoir("Arcadia Lake", exact = TRUE)

# Use in pipeline
res <- ok_reservoir("Arcadia Lake")
if (nrow(res) > 0) {
  ok_load(inflow_m3yr = 45e6, tp_inflow_ugl = 120) |>
    ok_hydraulics(
      surface_area_ha = res$surface_area_ha[1],
      mean_depth_m    = res$mean_depth_m[1]
    )
}

# All Cross Timbers lakes with quality A data
ok_reservoir(ecoregion = "Cross Timbers", data_quality = "A")
```

`ok_reservoirs` *Oklahoma reservoir morphometry dataset*

Description

A data frame containing morphometric and geographic characteristics for major reservoirs in Oklahoma, compiled from publicly available sources to enable rapid setup of okBATHTUB pipelines without manual morphometric lookup.

Usage

```
ok_reservoirs
```

Format

A data frame with one row per reservoir and the following columns:

lake_name Character. Canonical lake name.

alt_name Character. Common alternate name, if any.

county Character. Primary Oklahoma county.

managing_agency Character. Primary managing agency.

primary_use Character. Primary designated use.

surface_area_ha Numeric. Normal pool surface area (hectares).

mean_depth_m Numeric. Mean depth at normal pool (metres).

max_depth_m Numeric. Maximum depth at normal pool (metres).

volume_m3 Numeric. Total storage at normal pool (m^3).

watershed_area_km2 Numeric. Contributing watershed area (km^2).

eco_l3_code Character. EPA Level III ecoregion code.

eco_l3_name Character. EPA Level III ecoregion name.

latitude Numeric. Approximate dam latitude (WGS84 decimal degrees).

longitude Numeric. Approximate dam longitude (WGS84 decimal degrees).

year_completed Integer. Year dam was completed.

data_quality Character. Data quality code: `"A"` = direct from authoritative source (USACE design memoranda, published bathymetric surveys, or National Inventory of Dams); `"B"` = mean depth estimated from Oklahoma regional regression.

notes Character. Data source or caveat.

Data quality

Mean depth is the most critical morphometric parameter for okBATHTUB modelling (it drives hydraulic residence time). For reservoirs coded `"B"`, mean depth was estimated using an Oklahoma regional log-log regression fitted to reservoirs with known bathymetry:

$$\log_{10}(\text{mean depth}) = 0.28 \times \log_{10}(\text{area in ha}) - 0.34$$

This regression has substantial residual scatter; the resulting depth carries roughly a factor-of-1.5 prediction interval. Users with access to authoritative bathymetric data for specific reservoirs are encouraged to supply those values directly to `ok_hydraulics()` rather than relying on the estimated values here.

Source

Compiled from publicly available sources including U.S. Army Corps of Engineers (USACE) Tulsa District design memoranda, the National Inventory of Dams (NID), U.S. Bureau of Reclamation (BOR) design data, and published Oklahoma Water Resources Board reports. This dataset is provided as a convenience starting point and should be verified against the most current authoritative source for any decision-relevant application.

See Also

`ok_reservoir()`

Examples

```
# View all reservoirs
head(ok_reservoirs)

# Filter to a specific ecoregion
ok_reservoirs[ok_reservoirs$eco_l3_name == "Cross Timbers", ]
```

`ok_reservoir_summary` *Summarise the bundled reservoir dataset coverage*

Description

Prints a summary of the `ok_reservoirs` dataset by ecoregion, showing the number of lakes, surface area range, and data quality breakdown. Useful for understanding dataset coverage before modelling.

Usage

```
ok_reservoir_summary()
```

Value

A data frame (invisibly) summarising coverage by ecoregion.

`ok_retention` *Estimate nutrient retention coefficients*

Description

`ok_retention()` computes the fraction of incoming total phosphorus (TP) and total nitrogen (TN) that is retained within the reservoir. The retention coefficient is defined as

$$R = 1 - C_{lake}/C_{in}$$

and is applied in `ok_inlake()` to predict in-lake nutrient concentrations.

Three retention model families are supported, selected via the `coefficients` argument to `ok_load()`:

`"walker_model1"` **(Walker BATHTUB Model 1, default)** Second-order available-phosphorus sedimentation (Walker 1985, 1996). The mass balance solution is

$$C_{lake} = \frac{-1 + \sqrt{1 + 4KA_1C_{in}\tau}}{2KA_1\tau}$$

where $A_1 = 0.17 \cdot Q_s/(Q_s + 13.3)$ for TP and $B_1 = 0.0045 \cdot Q_s/(Q_s + 7.2)$ for TN, with $Q_s = \max(Z/T, 4)$.

`"vollenweider"` **(Vollenweider 1976 / Larsen-Mercier 1976)** First-order hydraulic-residence model:

$$R_{TP} = \frac{1}{1 + 1/\sqrt{\tau}}$$

Mathematically equivalent to Walker BATHTUB Model 5 (Northern Lakes). Single parameter (residence time), no ortho-P data required. Walker (1996) notes this form is not calibrated to Corps of Engineers reservoir data.

`"settling_velocity"` **(used as TN companion to vollenweider)**

$$R = v_s/(v_s + q_s)$$

where v_s is an apparent settling velocity (m/yr).

Usage

```
ok_retention(x, tp_retention_override = NULL, tn_retention_override = NULL)
```

Arguments

`x` An `okBATHTUB` object produced by `ok_hydraulics()`.

`tp_retention_override` Numeric between 0 and 1. If supplied, bypasses the equation and uses this value directly. Useful when observed retention is available from paired inflow/outflow monitoring. Default `NULL`.

`tn_retention_override` Numeric between 0 and 1. Same as above for TN. Default `NULL`.

Value

An `okBATHTUB` object at pipeline step `"retention"` with the following fields added to `$data`:

`tp_retention_coeff` TP retention coefficient (0-1).

`tn_retention_coeff` TN retention coefficient (0-1), or `NULL` if TN was not supplied.

`tp_retention_form`, `tn_retention_form` Character. Which retention equation was applied.

See Also

`ok_hydraulics()`, `ok_inlake()`

Examples

```
result <- ok_load(
  inflow_m3yr   = 45e6,
  tp_inflow_ugl = 120,
  tn_inflow_ugl = 1800
) |>
  ok_hydraulics(surface_area_ha = 890, mean_depth_m = 4.2) |>
  ok_retention()
print(result)
```

```
# Vollenweider / Larsen-Mercier retention
result_v <- ok_load(
  inflow_m3yr   = 45e6,
  tp_inflow_ugl = 120,
  coefficients  = "vollenweider"
) |>
  ok_hydraulics(surface_area_ha = 890, mean_depth_m = 4.2) |>
  ok_retention()

# Observed retention coefficient override
result_obs <- ok_load(inflow_m3yr = 45e6, tp_inflow_ugl = 120) |>
  ok_hydraulics(surface_area_ha = 890, mean_depth_m = 4.2) |>
  ok_retention(tp_retention_override = 0.42)
```

---

ok_scenario — *Run load reduction scenarios and compare predicted water quality responses*

Description

`ok_scenario()` takes a baseline `ok_load()` result and runs the full pipeline under one or more alternative loading scenarios, returning a tidy comparison table of predicted water quality across all scenarios. This is the primary tool for nutrient management planning - answering "how much does TP need to be reduced to move this lake from eutrophic to mesotrophic?"

Usage

```
ok_scenario(
  baseline,
  scenarios,
  include_baseline = TRUE,
  target_tsi = NULL,
  target_class = NULL
)
```

Arguments

`baseline`	An `okBATHTUB` object that has been run through at least `ok_hydraulics()`. All morphometry is taken from this object.
`scenarios`	A list of named lists, one per scenario. Each list must have a `label` (character, required). Optional fields: `tp_reduction` (numeric 0-1, fractional TP reduction), `tn_reduction` (numeric 0-1, fractional TN reduction), `tp_inflow_ugl` (numeric, absolute inflow TP in ug/L, overrides tp_reduction), `tn_inflow_ugl` (numeric, absolute inflow TN in ug/L), `flow_change` (numeric, fractional change in inflow volume, e.g. `-0.20` = 20% flow reduction).

include_baseline
: Logical. Whether to include the baseline run as the first row in the output. Default TRUE.

target_tsi
: Numeric. Optional TSI target. If supplied, a meets_target column is added. Default NULL.

target_class
: Character. One of "oligotrophic", "mesotrophic", or "eutrophic". Sets target_tsi to the upper bound of that class automatically.

Value

A data frame with one row per scenario (plus baseline if requested). Columns include: scenario (label), tp_inflow_ugl (inflow TP in ug/L), tp_reduction_pct (reduction from baseline in percent), tp_inlake_ugl (predicted in-lake TP), chla_ugl (predicted chlorophyll-a), secchi_m (predicted Secchi depth), tsi_tp, tsi_chla, tsi_mean (Carlson TSI values), trophic_state (classification), and optionally meets_target (logical, present when target_tsi is supplied).

How scenarios work

Each scenario modifies one or more inflow parameters relative to the baseline. Reductions are expressed as fractions (0-1), where tp_reduction = 0.30 means a 30% reduction in inflow TP load. Scenarios can also specify absolute concentrations directly via tp_inflow_ugl, which overrides the reduction fraction.

The morphometric parameters (surface area, mean depth) from the baseline ok_hydraulics() call are applied to every scenario.

See Also

ok_load, ok_scenario_sweep

Examples

```
# Baseline: Arcadia Lake with estimated inflow loading
baseline <- ok_load(
  inflow_m3yr   = 45e6,
  tp_inflow_ugl = 120,
  tn_inflow_ugl = 1800,
  segment_label = "lacustrine",
  coefficients  = "oklahoma",
  ecoregion     = "Cross Timbers"
) |>
ok_hydraulics(surface_area_ha = 890, mean_depth_m = 4.2)

# Scenario analysis: what TP reduction gets us to mesotrophic?
scenarios <- list(
  list(label = "10% TP reduction",  tp_reduction = 0.10),
  list(label = "20% TP reduction",  tp_reduction = 0.20),
  list(label = "30% TP reduction",  tp_reduction = 0.30),
  list(label = "40% TP reduction",  tp_reduction = 0.40),
  list(label = "50% TP reduction",  tp_reduction = 0.50)
)
```

```
results <- ok_scenario(
  baseline      = baseline,
  scenarios     = scenarios,
  target_class  = "mesotrophic"
)
print(results)
```

ok_scenario_sweep — *Sweep TP reduction scenarios automatically*

Description

A convenience wrapper around `ok_scenario()` that automatically generates a sequence of TP reduction scenarios from 0 to max_reduction_pct percent in steps of step_pct percent. Useful for generating load-response curves and finding the minimum reduction needed to achieve a trophic state target.

Usage

```
ok_scenario_sweep(
  baseline,
  max_reduction_pct = 70,
  step_pct = 5,
  target_tsi = NULL,
  target_class = NULL
)
```

Arguments

`baseline`	An `okBATHTUB` object through `ok_hydraulics()`.
`max_reduction_pct`	Numeric. Maximum TP reduction to evaluate (percent). Default 70.
`step_pct`	Numeric. Step size between scenarios (percent). Default 5.
`target_tsi`	Numeric. Optional TSI target passed to `ok_scenario()`. Default `NULL`.
`target_class`	Character. Optional trophic class target (`"oligotrophic"`, `"mesotrophic"`, or `"eutrophic"`) passed to `ok_scenario()`. Default `NULL`.

Value

A data frame as returned by `ok_scenario()`, with one row per reduction step plus the baseline.

See Also

`ok_scenario`

Examples

```
baseline <- ok_load(
  inflow_m3yr   = 45e6,
  tp_inflow_ugl = 120,
  ecoregion     = "Cross Timbers",
  coefficients  = "oklahoma"
) |>
ok_hydraulics(surface_area_ha = 890, mean_depth_m = 4.2)

sweep <- ok_scenario_sweep(baseline, target_class = "mesotrophic")
print(sweep[, c("scenario", "tp_inflow_ugl", "tsi_mean",
                "trophic_state", "meets_target")])
```

---

ok_segment — *Chain multiple reservoir segments sequentially*

Description

ok_segment() links two reservoir segments in series, passing the outflow water quality of an upstream segment as the inflow to the next downstream segment. This reflects the longitudinal zonation common in Oklahoma reservoirs - riverine, transitional, and lacustrine segments each behave differently and should be modelled separately when data support it.

The function takes a completed upstream segment result (through at least ok_inlake()) and returns a new okBATHTUB object at the "load" step, pre-populated with the upstream outflow concentrations as the downstream inflow inputs. The downstream segment can then be run through the full pipeline normally.

Usage

```
ok_segment(
  upstream,
  segment_label = "downstream",
  coefficients = NULL,
  ecoregion = NULL
)
```

Arguments

upstream	An okBATHTUB object that has been run through at least ok_inlake(). The in-lake TP, TN, and inflow volume from this result become the downstream segment's inflow.
segment_label	Character. Label for the downstream segment. Default "downstream".
coefficients	Coefficient set for the downstream segment. Defaults to the same coefficient set used in the upstream segment. Can be overridden to apply different ecoregion coefficients to different segments.
ecoregion	Character. EPA Level III ecoregion for the downstream segment. If NULL, inherits from the upstream segment.

Value

An okBATHTUB object at pipeline step "load", ready to pipe into ok_hydraulics() for the downstream segment.

Mass balance at the segment boundary

The outflow TP concentration from the upstream segment becomes the inflow TP concentration for the downstream segment:

$$C_{in,down} = C_{lake,up} = C_{in,up} \times (1 - R_{up})$$

Inflow volume is passed through unchanged under the steady-state assumption. If the downstream segment has a different surface area or morphometry, supply those via ok_hydraulics() after this call.

See Also

ok_load, ok_segment_chain

Examples

```
# Two-segment reservoir: riverine -> lacustrine
riverine <- ok_load(
  inflow_m3yr   = 45e6,
  tp_inflow_ugl = 150,
  tn_inflow_ugl = 2200,
  segment_label = "riverine"
) |>
ok_hydraulics(surface_area_ha = 280, mean_depth_m = 3.1) |>
ok_retention() |>
ok_inlake()

lacustrine <- ok_segment(riverine, segment_label = "lacustrine") |>
  ok_hydraulics(surface_area_ha = 610, mean_depth_m = 5.8) |>
  ok_retention() |>
  ok_inlake() |>
  ok_tsi()

summary(lacustrine)
```

---

ok_segment_chain *Chain multiple reservoir segments from a list*

Description

A convenience wrapper around ok_segment() for reservoirs with more than two segments. Accepts a list of segment morphometry specifications and runs them sequentially, passing each segment's outflow into the next.

Usage

```
ok_segment_chain(
  inflow_m3yr,
  tp_inflow_ugl,
  tn_inflow_ugl = NULL,
  segments,
  coefficients = "walker",
  ecoregion = NULL
)
```

Arguments

`inflow_m3yr` Numeric. Total annual inflow (m^3/yr).

`tp_inflow_ugl` Numeric. Inflow TP (ug/L).

`tn_inflow_ugl` Numeric. Inflow TN (ug/L). Optional.

`segments` A list of named lists, one per segment, each containing:

- `label` Character. Segment name.
- `surface_area_ha` Numeric. Surface area (ha).
- `mean_depth_m` Numeric. Mean depth (m).
- `outflow_m3yr` Numeric. Outflow volume (m^3/yr). Optional; defaults to inflow.

`coefficients` Coefficient set applied to all segments. Default `"walker"`.

`ecoregion` EPA Level III ecoregion. Applied to all segments.

Value

A named list of okBATHTUB objects at step `"tsi"`, one per segment, in downstream order. Names match the `label` field of each segment specification.

See Also

`ok_segment`

Examples

```
segments <- list(
  list(label = "riverine",     surface_area_ha = 280, mean_depth_m = 3.1),
  list(label = "transitional", surface_area_ha = 410, mean_depth_m = 4.5),
  list(label = "lacustrine",   surface_area_ha = 610, mean_depth_m = 5.8)
)

results <- ok_segment_chain(
  inflow_m3yr   = 45e6,
  tp_inflow_ugl = 150,
  tn_inflow_ugl = 2200,
  segments      = segments
)

# View trophic state of each segment
```

```
lapply(results, function(r) r$data$trophic_state)
```

`ok_segment_summary` *Summarise a multi-segment chain as a data frame*

Description

Converts the list output of `ok_segment_chain()` into a tidy data frame with one row per segment, suitable for plotting or reporting.

Usage

```
ok_segment_summary(chain_result)
```

Arguments

`chain_result` Named list of `okBATHTUB` objects returned by `ok_segment_chain()`.

Value

A data frame with one row per segment containing key water quality predictions and TSI values.

`ok_theme` *Oklahoma water-themed ggplot2 theme*

Description

A clean, minimal ggplot2 theme used consistently across all okBATHTUB visualization functions. Based on `theme_minimal()` with clean typography and grid settings.

Usage

```
ok_theme(base_size = 11, legend_position = "bottom")
```

Arguments

`base_size` Numeric. Base font size. Default 11.

`legend_position` Character. Legend position. Default `"bottom"`.

Value

A `ggplot2::theme` object.

ok_trophic_colors — *Oklahoma water-themed ggplot2 palette*

Description

Named color vector for trophic state classifications used consistently across all okBATHTUB plots.

Usage

```
ok_trophic_colors()
```

Value

Named character vector of hex colors.

ok_tsi — *Compute Carlson Trophic State Indices*

Description

`ok_tsi()` computes Carlson (1977) Trophic State Index (TSI) values from in-lake water quality predictions and assigns an overall trophic state classification.

Usage

```
ok_tsi(
  x,
  observed_tp_ugl = NULL,
  observed_chla_ugl = NULL,
  observed_secchi_m = NULL
)
```

Arguments

- `x` — An `okBATHTUB` object produced by `ok_inlake()`.
- `observed_tp_ugl` — Numeric. If supplied, computes TSI(TP) from this observed value instead of the model-predicted in-lake TP. Default `NULL`.
- `observed_chla_ugl` — Numeric. Observed chlorophyll-a (ug/L) to use instead of predicted. Default `NULL`.
- `observed_secchi_m` — Numeric. Observed Secchi depth (m) to use instead of predicted. Default `NULL`.

Value

An okBATHTUB object at pipeline step "tsi".

TSI equations (Carlson 1977)

$$TSI(TP) = 14.42 \times \ln(TP) + 4.15$$
$$TSI(Chl\text{-}a) = 9.81 \times \ln(Chl\text{-}a) + 30.6$$
$$TSI(Secchi) = 60.0 - 14.41 \times \ln(Secchi)$$

where TP is in ug/L, chlorophyll-a is in ug/L, and Secchi depth is in metres. The Chl-a coefficient 9.81 matches the original Carlson (1977) paper; Walker's BATHTUB documentation uses 9.84 (likely a rounding artifact). The package uses 9.81, consistent with the primary limnological literature.

Trophic state classification

Based on the mean TSI across available indices:

- TSI < 40 -> Oligotrophic
- 40 <= TSI < 50 -> Mesotrophic
- 50 <= TSI < 70 -> Eutrophic
- TSI >= 70 -> Hypereutrophic

Note on partial indices

When only one or two of the three TSI components are available (e.g. because Secchi depth could not be predicted), `tsi_mean` is the arithmetic mean of the available components and `tsi_n` reports how many were used. Carlson's deviation analysis assumes all three are available; interpret `tsi_mean` with caution when `tsi_n` < 3.

References

Carlson, R.E. (1977). A trophic state index for lakes. *Limnology and Oceanography*, 22(2), 361-369.

See Also

`ok_inlake()`, `ok_tsi_observed()`

Examples

```
result <- ok_load(
  inflow_m3yr   = 45e6,
  tp_inflow_ugl = 120,
  tn_inflow_ugl = 1800
) |>
  ok_hydraulics(surface_area_ha = 890, mean_depth_m = 4.2) |>
  ok_retention() |>
```

```
  ok_inlake() |>
  ok_tsi()
summary(result)
```

`ok_tsi_observed` *Compute Carlson TSI from observed values only*

Description

A standalone helper for computing Carlson TSI directly from observed water quality measurements, without running the full prediction pipeline. Useful for computing observed trophic state from grab sample data for comparison against modelled predictions.

Usage

```
ok_tsi_observed(tp_ugl = NULL, chla_ugl = NULL, secchi_m = NULL)
```

Arguments

`tp_ugl`	Numeric. Observed in-lake total phosphorus (ug/L). Optional.
`chla_ugl`	Numeric. Observed chlorophyll-a (ug/L). Optional.
`secchi_m`	Numeric. Observed Secchi depth (m). Optional.

Value

A named list with elements `tsi_tp`, `tsi_chla`, `tsi_secchi`, `tsi_n`, `tsi_mean`, and `trophic_state`.

References

Carlson, R.E. (1977). A trophic state index for lakes. *Limnology and Oceanography*, 22(2), 361-369.

Examples

```
ok_tsi_observed(tp_ugl = 85, chla_ugl = 22, secchi_m = 0.8)
```

print.okBATHTUB *Print method for okBATHTUB objects*

Description

Prints a compact summary of an okBATHTUB pipeline result to the console, including the pipeline step, segment label (if any), the coefficient set in use, and a list of single-value numeric or character fields stored in the result.

Usage

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

Arguments

x	An okBATHTUB object.
...	Ignored.

Value

The okBATHTUB object x, returned invisibly. Called primarily for the side effect of printing a formatted summary to the console via cat().

summary.okBATHTUB *Summary method for okBATHTUB objects*

Description

Prints a formatted summary of all water quality predictions accumulated through the pipeline. Most informative after ok_tsi() has been run.

Usage

```
## S3 method for class 'okBATHTUB'
summary(object, ...)
```

Arguments

object	An okBATHTUB object.
...	Ignored.

Value

The okBATHTUB object object, returned invisibly. Called primarily for the side effect of printing a multi-line formatted water quality summary to the console. The summary block includes segment metadata, hydraulic and load fields, in-lake water quality predictions, and Carlson Trophic State Indices, depending on which pipeline steps have been completed on the input object.

Index