Appendix B. Manual for Functions and Datasets in the R-Package ‘wrv’

This vignette is a reference manual for functions and datasets in the **wrv** package (version 1.0.0). A short description and R usage information (such as examples of how to use a function) are given for all functions and datasets. Dataset documentation include sections describing the format of a dataset (such as a vector or matrix) and details on the original or secondary data source. Functions and datasets are named according to the following conventions. The form for function names is no separator between words and initial capital letters (such as `AddScaleBar`); dataset names have words separated with dots and all lower case letters (such as `alluvium.extent`).

Contents

AddBubbles	*Add Bubble Map to Plot*

Description

This function can be used to add a bubble map to a plot. Proportional circle symbols are used to represent spatial point data, where symbol area varies in proportion to an attribute variable.

Usage

```
AddBubbles(x, y = NULL, z, zlim = NULL, inches = c(0, 0.2),
           scaling = c("perceptual", "mathematical"),
           bg.pos = "red", bg.neg = "blue", fg = NA, lwd = 0.25,
           cex = 0.7, format = NULL, draw.legend = TRUE,
           loc = c("bottomleft", "topleft", "topright", "bottomright"),
           inset = 0.02, breaks = NULL, break.labels = NULL,
           quantile.breaks = FALSE, make.intervals = FALSE,
           title = NULL, subtitle = NULL, add = TRUE)
```

Arguments

`x, y`	`numeric`; the x and y coordinates for the centers of the circle symbols. They can be specified in any way which is accepted by `xy.coords`.
`z`	`numeric`; is the attribute variable.
`zlim`	`numeric`; the minimum and maximum `z` values that circle symbols are plotted; defaults to the range of the finite values of `z`.
`inches`	`numeric`; a vector of length 2 specifying the radii limits for the drawn circle symbol.
`scaling`	`character`; selects the proportional symbol mapping algorithm to be used; either `"perceptual"` or `"mathematical"` scaling (Tanimura and others, 2006).
`bg.pos, bg.neg`	`character` or `function`; the fill color(s) for circle symbols corresponding to positive and negative `z` values, respectively. A color palette also may be specified.
`fg`	`character`; the outer-line color for circle symbols. Specify an `NA` value to remove the symbols outer line, and a `NULL` value to match the outer-line color with the symbols fill color.
`lwd`	`numeric`; is the line width for drawing circle symbols.
`cex`	`numeric`; the character expansion factor for legend labels.
`format`	`character`; the formatting for legend values, see `formatC` for options.
`draw.legend`	`logical`; if `TRUE`, a legend is drawn.
`loc`	`character`; the position of the legend in the main plot region: `"bottomleft"`, `"topleft"`, `"topright"`, or `"bottomright"` to denote scale location.
`inset`	`numeric`; the inset distance of the legend from the margins as a fraction of the main plot region. Defaults to 2 percent of the axis range.
`breaks`	`numeric`; a set of finite breakpoints for the legend circle symbols.
`break.labels`	`character`; a vector of break labels with length equal to `breaks`.
`quantile.breaks`	`logical`; if `TRUE`, `breaks` are set to the sample quantiles of `z`.
`make.intervals`	`logical`; if `TRUE`, represent `z` within intervals. See `findInterval` function for details.
`title`	`character`; the main title to be placed at the top of the legend.
`subtitle`	`character`; a legend subtitle to be placed below the main title.
`add`	`logical`; if `TRUE`, circle symbols (and an optional legend) are added to an existing plot.

Details

Symbols are sequentially drawn in decreasing order of circle diameter.

Value

Primarily used for the side-effect of a bubble map drawn on the current graphics device.

Author(s)

J.C. Fisher, U.S. Geological Survey, Idaho Water Science Center

References

Tanimura, S., Kuroiwa, C., and Mizota, T., 2006, Proportional Symbol Mapping in R: Journal of Statistical Software, v. 15, no. 5, 7 p.

See Also

`symbols`

Examples

```
n <- 50L
x <- cbind(runif(n, 1, 10), runif(n, 1, 10))
z <- runif(n, -5000, 10000)
AddBubbles(x, z = z, fg = "green", lwd = 2, title = "Title", loc = "topright",
           breaks=pretty(z, n = 8), add = FALSE)

Pal1 <- colorRampPalette(c("#F4A582", "#CA0020"))
Pal2 <- colorRampPalette(c("#92C5DE", "#0571B0"))
AddBubbles(x, z = z, bg.pos = Pal1, bg.neg = Pal2, add = FALSE)

AddBubbles(x, z = z, bg.pos = Pal1, bg.neg = Pal2, add = FALSE,
           make.intervals = TRUE)

AddBubbles(x, z = runif(n, 10, 10000), title = "Quantiles", bg.pos = topo.colors,
           quantile.breaks = TRUE, fg = NULL, add = FALSE)
```

`AddColorKey` *Add Color Key to Plot*

Description

This function can be used to add a color key to a plot.

Usage

```
AddColorKey(mai, is.categorical, breaks, col, at = NULL, labels = TRUE,
            scientific = FALSE, explanation = NULL, padx = 0.2)
```

Arguments

`mai` — `numeric`; a numerical vector of the form `c(bottom, left, top, right)` which gives the margin size specified in inches (optional).

`is.categorical` — `logical`; if `TRUE`, color-key values represent categorical data; otherwise, these data values are assumed continuous.

`breaks` — `numeric`; a set of finite numeric breakpoints for the colors: must have one more breakpoint than color and be in increasing order.

`col` — `character`; a vector of colors to be used in the plot. This argument requires `breaks` specification for continuous data. For continuous data there should be one less color than breaks; whereas, categorical data require a color for each category.

`at` — `numeric`; the points at which tick-marks and labels are to be drawn, only applicable for continuous data. The tick-marks will be located at the color breaks if the length of `at` is greater than or equal to one minus the length of `breaks`.

`labels` — `logical` or `character`; this can either be a logical value specifying whether (numerical) annotations are to be made at the tickmarks, or a character or expression vector of labels to be placed at the tickpoints.

`scientific` — `logical`; indicates if axes labels should be formatted for scientific notation, see `ToScientific` for details.

`explanation` — `character`; a label that describes the data values.

`padx` — `numeric`; the inner padding for the left and right margins specified in inches.

Value

Used for the side-effect of a color key drawn on the current graphics device.

Author(s)

J.C. Fisher, U.S. Geological Survey, Idaho Water Science Center

See Also

`PlotCrossSection`, `PlotMap`

Examples

```
dev.new(width = 7, height = 2)

AddColorKey(is.categorical = FALSE, breaks = 0:10, scientific = TRUE,
            explanation = "Example description for data variables in meters.")
AddColorKey(is.categorical = FALSE, breaks = 0:10, at = pretty(0:10))
AddColorKey(is.categorical = FALSE, breaks = seq(0.5, 10.5, by = 1), at = 1:10)

AddColorKey(is.categorical = TRUE, labels = LETTERS[1:5])
AddColorKey(is.categorical = TRUE, col = terrain.colors(5))

graphics.off()
```

`AddInsetMap` *Add Inset Map to Plot*

Description

This function can be used to add an inset map to a plot.

Usage

```
AddInsetMap(p, col = c("#D8D8D8", "#BFA76F"),
            main.label = list(label = NA, adj = NULL),
            sub.label = list(label = NA, adj = NULL),
            loc = c("bottomleft", "topleft", "topright", "bottomright"),
            inset = 0.02, width = NULL)
```

Arguments

`p`	`SpatialPolygons`; the polygon describing the large map.
`col`	`character`; a vector of length 2 giving the colors for filling the large map polygon `p` and the smaller plot extent rectangle.
`main.label`	`list`; a list with components `label` and `adj`. The text label and position (`x` and `y` adjustment of the label) for the large map, respectively.
`sub.label`	`list`; identical to the `main.label` argument but for the plot extent rectangle.
`loc`	`character`; the position of the inset map in the main plot region: `"bottomleft"`, `"topleft"`, `"topright"`, or `"bottomright"` to denote scale location.
`inset`	`numeric`; the inset distance from the margins as a fraction of the main plot region. Defaults to 2 percent of the axis range.
`width`	`numeric`; the width of the inset map, in inches.

Details

The smaller axis-aligned rectangle (relative to the larger map polygon) is defined by the user coordinate extent of the main plot region, see `par("usr")`.

Value

Used for the side-effect of a inset map drawn on the current graphics device.

Author(s)

J.C. Fisher, U.S. Geological Survey, Idaho Water Science Center

See Also

`PlotMap`

Examples

```
PlotMap(alluvium.thickness@crs, bg.image = hill.shading, reg.axs = FALSE)
AddInsetMap(idaho, width = 1, main.label = list("IDAHO", adj = c(-0.4, -4.9)),
            sub.label = list("Map area", adj = c(0.5, 2.5)), loc = "topright")

graphics.off()
```

AddScaleBar	*Add Scale Bar to Plot*

Description

This function can be used to add a scale bar to a plot.

Usage

```
AddScaleBar(asp = 1, unit = NULL, is.lonlat = FALSE,
            loc = c("bottomleft", "topleft", "topright", "bottomright"),
            offset = c(0, 0), lab.vert.exag = NULL)
```

Arguments

`asp`	`numeric`; the *y/x* aspect ratio for spatial axes.
`unit`	`character`; axis unit of measurement, for example `"METERS"`.
`is.lonlat`	`logical`; if `TRUE`, plot coordinates are in longitude and latitude.
`loc`	`character`; the position of the scale bar in the plot region: `"bottomleft"`, `"topleft"`, `"topright"`, or `"bottomright"` to denote scale location.
`offset`	`numeric`; the `x` and `y` adjustments of the scale bar, in inches.
`lab.vert.exag`	`logical`; if `TRUE`, a label is drawn specifying the vertical exaggeration.

Value

Used for the side-effect of a scale bar drawn on the current graphics device.

Author(s)

J.C. Fisher, U.S. Geological Survey, Idaho Water Science Center

See Also

`PlotCrossSection`, `PlotMap`

Examples

```
plot(-100:100, -100:100, type = "n", xlab = "x", ylab = "y", asp = 2)
AddScaleBar(2, unit = "FEET", loc = "topleft")
AddScaleBar(2, unit = "METERS", loc = "bottomright", offset = c(-0.2, 0))
```

alluvium.extent	*Extent of Alluvium Unit*

Description

The estimated extent of alluvium unit in the Wood River Valley, south-central Idaho.

Usage

```
alluvium.extent
```

Format

An object of `SpatialPolygonsDataFrame` class containing 1 `Polygons`. Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

Extent defined by Bartollino and Adkins (2012, Plate 1).

References

Bartolino, J.R., and Adkins, C.B., 2012, Hydrogeologic framework of the Wood River Valley aquifer system, south-central Idaho: U.S. Geological Survey Scientific Investigations Report 2012-5053, 46 p., available at http://pubs.usgs.gov/sir/2012/5053/.

Examples

```
plot(alluvium.extent, col = "#BFA76F")
str(alluvium.extent)
```

`alluvium.thickness` *Thickness of the Quaternary Sediment*

Description

The estimated thickness of the Quaternary sediment in the Wood River Valley aquifer system, South-Central Idaho.

Usage

```
alluvium.thickness
```

Format

An object of `RasterLayer` class. Each cell on the surface grid represents a depth measured from land surface in meters. Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM). The spatial grid is composed of 565 rows and 429 columns, and has cell sizes that are constant at 100 meters by 100 meters.

Source

This dataset is a revised version of Plate 1 in Bartolino and Adkins (2012).

References

Bartolino, J.R., and Adkins, C.B., 2012, Hydrogeologic framework of the Wood River Valley aquifer system, south-central Idaho: U.S. Geological Survey Scientific Investigations Report 2012-5053, 46 p., available at http://pubs.usgs.gov/sir/2012/5053/.

Examples

```
col <- rainbow(255, start = 0.0, end = 0.8)
image(alluvium.thickness, col = col, asp = 1, axes = FALSE, xlab = "", ylab = "")
summary(alluvium.thickness)
```

`basalt.extent` *Extent of Basalt Unit*

Description

The estimated extent of the basalt unit underlying the alluvial Wood River Valley aquifer system.

Usage

```
basalt.extent
```

Format

An object of `SpatialPolygonsDataFrame` class containing 1 `Polygons`. Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

Extent defined by Bartolino and Adkins (2012, Plate 1).

References

Bartolino, J.R., and Adkins, C.B., 2012, Hydrogeologic framework of the Wood River Valley aquifer system, south-central Idaho: U.S. Geological Survey Scientific Investigations Report 2012-5053, 46 p., available at `http://pubs.usgs.gov/sir/2012/5053/`.

Examples

```
plot(basalt.extent, col = "#BEAED4", border = NA)
plot(alluvium.extent, add = TRUE)
str(basalt.extent)
```

`bellevue.wwtp.ponds` *Bellevue Waste Water Treatment Plant Ponds*

Description

The location of the Bellevue Waste Water Treatment Plant ponds.

Usage

```
bellevue.wwtp.ponds
```

Format

An object of `SpatialPolygons` class containing 1 `Polygons`. Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

Idaho Department of Water Resources, accessed on December 11, 2014

Examples

```
plot(bellevue.wwtp.ponds)
```

`BumpDisconnectCells` *Adjustment for Vertically Disconnected Cells*

Description

This function decreases model cell values (such as, land-surface elevations) in the lower raster layer if they violate a minimum vertical overlap between adjacent cells.

Usage

```
BumpDisconnectCells(rs, min.overlap = 2, bump.by = 0.1, max.itr = 1e+04)
```

Arguments

`rs`	`RasterStack`; a collection of two raster layers, the first and second layers represent the top and bottom of a model layer.
`min.overlap`	`numeric`; the minimum vertical overlap between adjacent cells.
`bump.by`	`numeric`; the amount to decrease a cell value by during each iteration of the algorithm.
`max.itr`	`numeric`; the maximum number of iterations.

Details

During each iteration of the algorithm: (1) Cells are identified that violate the minimum vertical overlap between adjacent cells; that is, the bottom of cell `i` is greater than or equal to the top of an adjacent cell `j` minus the minimum overlap specified by the `min.overlap` argument. (2) For cells violating the minimum vertical overlap, lower raster layer (`rs[[2]]`) values are decreased by the value specified in the `bump.by` argument.

Value

Returns a `RasterLayer` that can be added to `rs[[2]]` to ensure connectivity between cells. Cell values in the returned raster grid represent vertical adjustments.

Author(s)

J.C. Fisher, U.S. Geological Survey, Idaho Water Science Center

Examples

```
set.seed(0)
r.top <- raster(ncols = 10, nrows = 10)
r.bot <- raster(ncols = 10, nrows = 10)
r.top[] <- rnorm(ncell(r.top), mean = 12)
r.bot[] <- rnorm(ncell(r.bot), mean = 10)
summary(r.top - r.bot)

r <- BumpDisconnectCells(stack(r.top, r.bot), min.overlap = 0.1)
plot(r.bot + r)
```

`BumpRiverStage` *Adjustment for Implausible River Stage*

Description

This function decreases stage values in river cells if they are implausible with respect to water always flowing downhill.

Usage

```
BumpRiverStage(r, outlets, min.drop = 1e-06)
```

Arguments

`r`	`RasterLayer`; each cell on the surface grid represents a river stage.
`outlets`	`SpatialPoints*`, `SpatialLines*`, `SpatialPolygons*` or `Extent`; the location of discharge outlets. The `rasterize` function is used to locate outlet cells in the raster grid `r`.
`min.drop`	`numeric`; the minimum drop in stage between adjacent river cells.

Details

The Lee algorithm (Lee, 1961) is used to identify flow paths among the modeled river cells. An analysis of river cell stage values along a flow path identifies any problematic cells that are obstructing downhill surface-water flow. Stage values for these problematic cells are then lowered to an acceptable elevation.

Value

Returns a `RasterLayer` with cell values representing the vertical change in stream stage. These changes can be added to `r` to ensure that water always flows downhill.

Author(s)

J.C. Fisher, U.S. Geological Survey, Idaho Water Science Center

References

Lee, C.Y., 1961, An algorithm for path connections and its applications: IRE Transactions on Electronic Computers, v. EC-10, no. 2, p. 346–365.

Examples

```
## Not run: # see uncalibrated-model vignette
```

`bypass.canal` *Bypass Canal*

Description

The location of the Bypass Canal.

Usage

```
bypass.canal
```

Format

An object of `SpatialLines` class containing 4 `Lines`. Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

Idaho Department of Water Resources, accessed on January 15, 2015

Examples

```
plot(bypass.canal)
```

`canal.seep` *Canal Seepage*

Description

Canal seepage as a fraction of diversions for irrigation entities in the Wood River Valley.

Usage

```
canal.seep
```

Format

A `data.frame` object with 19 records and the following variables:

EntityName is the name of the irrigation entity served by the canal system.

SeepFrac is the estimated canal seepage as a fraction of diversions.

Source

Idaho Department of Water Resources, accessed on November 4, 2015

See Also

`canals`

Examples

```
str(canal.seep)

d <- canal.seep[order(canal.seep$SeepFrac, decreasing=TRUE), ]
par(mar = c(4.1, 8.1, 0.1, 0.6))
barplot(d$SeepFrac, names.arg = d$EntityName, horiz = TRUE, cex.names = 0.7,
        cex.axis = 0.7, cex.lab = 0.7, las = 1, xlab = "Seepage fraction")

graphics.off()
```

`canals` *Canal Systems*

Description

The canal systems in the Wood River Valley and surrounding areas.

Usage

```
canals
```

Format

An object of `SpatialLinesDataFrame` class containing 113 `Lines` and a `data.frame` with the following variable:

EntityName the name of the irrigation entity served by the canal system.

Name the local canal name.

Source

Idaho Department of Water Resources, accessed on November 29, 2014

See Also

`r.canals`, `canal.seep`

Examples

```
plot(canals, col = "#3399CC")
str(canals@data)
```

`cities` *Cities and Towns*

Description

Cities and towns in the Wood River Valley and surrounding areas.

Usage

```
cities
```

Format

An object of `SpatialPointsDataFrame` class containing 11 points. Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

Idaho Department of Water Resources (IDWR), accessed on April 15, 2015

Examples

```
str(cities)

col <- "#333333"
plot(cities, pch = 15, cex = 0.8, col = col)
text(cities, labels = cities@data$FEATURE_NA, col = col, cex = 0.5, pos = 1, offset = 0.4)
```

`clay.extent` *Extent of Clay Unit*

Description

The estimated extent of the clay confining unit (aquitard) separating the unconfined aquifer from the underlying confined aquifer in the Wood River Valley.

Usage

```
clay.extent
```

Format

An object of `SpatialPolygonsDataFrame` class containing 2 `Polygons`. Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

Extent defined by Moreland (1977, fig. 3 in USGS Open-File report). Moreland (1977) shows an outlier by Picabo that is assumed to indicate confined conditions in the basalt and not the lake sediments.

References

Moreland, J.A., 1977, Ground water-surface water relations in the Silver Creek area, Blaine County, Idaho: Boise, Idaho Department of Water Resources, Water Information Bulletin 44, 42 p., 5 plates in pocket, accessed January 31, 2012. Also published as U.S. Geological Survey Open-File report 77-456, 66 p., available at http://pubs.er.usgs.gov/pubs/ofr/ofr77456.

Examples

```
plot(clay.extent, col = "#FDC086", border = NA)
plot(alluvium.extent, add = TRUE)
str(clay.extent)
```

`comb.sw.irr` *Combined Surface-Water Irrigation Diversions*

Description

Supplemental groundwater rights and associated surface-water rights.

Usage

```
comb.sw.irr
```

Format

A `data.frame` object with 1,213 records and the following variables:

WaterRight is the name of the supplemental groundwater right.

CombWaterRight is the name of the surface-water right that shares a combined limit with the groundwater right.

Source is the river or stream source name for the surface-water right.

WaterUse is the authorized beneficial use for the surface-water right.

MaxDivRate is the authorized maximum diversion rate for the surface-water right, in cubic meters per day.

Pdate is the priority date of the surface-water right.

Source

Idaho Department of Water Resources (IDWR), accessed on April 25, 2014; derived from combined limit comments in IDWR water rights database.

Examples

```
str(comb.sw.irr)
```

`div.gw` *Groundwater Diversions*

Description

Groundwater diversions recorded by Water District 37 or municipal water providers. Groundwater is diverted from the aquifer by means of either pumping wells or flowing-artesian wells.

Usage

```
div.gw
```

Format

A `data.frame` object with 7,292 records and the following variables:

YearMonth is the year and month during which diversions were recorded, with a required date format of `YYYYMM`.

Diversion is the name of the well.

Reach is the name of the river subreach into which the well water is discharged; only applicable to exchange wells.

BigReach is the name of the river reach into which the well water is discharged; only applicable to exchange wells.

EntityName is the name of the irrigation entity which the well supplies water.

WMISNumber is the well number in the Idaho Department of Water Resources (IDWR) Water Measurement Information System.

GWDiv is the volume of water diverted during the month, in cubic meters.

Source

IDWR, accessed on December 11, 2014; compiled data records from Water District 37 and 37M, City of Ketchum, Sun Valley Water and Sewer District, City of Hailey, and City of Bellevue.

Examples

```
str(div.gw)
```

`div.ret.exch` *Diversions, Returns, and Exchange Wells*

Description

The location of streamflow diversions, irrigation canal or pond returns, and exchange well returns.

Usage

```
div.ret.exch
```

Format

An object of `SpatialPointsDataFrame` class containing 117 points with the following variables:

Name a local name for the diversion/return site.

Type the data type: “Diversion”, “Return”, and “Exchange well inflow”.

LocSource the data source.

Big the corresponding river reach.

Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

Idaho Department of Water Resources, accessed on June 5, 2015

Examples

```
plot(div.ret.exch)
str(div.ret.exch@data)
```

`div.sw` *Surface-Water Diversions*

Description

Surface-water diversions recorded by Water District 37 or municipal water providers.

Usage

```
div.sw
```

Format

A `data.frame` object with 15,550 records and the following variables:

YearMonth is the year and month during which diversions were recorded, with a required date format of `YYYYMM`.

Diversion is the name of the surface-water diversion.

Reach is the river subreach from which the water is diverted.

BigReach is the river reach from which the water is diverted.

EntityName is the name of the irrigation entity which the diversion supplies water.

SWDiv is the volume of water diverted during the month, in cubic meters.

Source

Idaho Department of Water Resources, accessed on December 11, 2014; compiled data records from Water District 37 and 37M, City of Hailey, City of Bellevue, City of Ketchum, and Sun Valley Water and Sewer District.

Examples

```
str(div.sw)
```

---

div.ww — *Wastewater Treatment Plant Diversions*

Description

Discharge from wastewater treatment plants.

Usage

```
div.ww
```

Format

A `data.frame` object with 1,182 records and the following variables:

YearMonth is the year and month during which diversions were recorded, with a required date format of `YYYYMM`.

Return is the name of the wastewater treatment plant.

Reach is the name of the river subreach to which treated effluent is discharged; only applicable to wastewater treatment plants that discharge to the river.

BigReach is the name of the river reach to which treated effluent is discharged; only applicable to wastewater treatment plants that discharge to the river.

EntityName is the name of the irrigation entity served by the wastewater treatment plant.

WWDiv is the volume of wastewater discharged during the month, in cubic meters.

Source

Idaho Department of Water Resources and U.S. Geological Survey, accessed on August 11, 2014; compiled data records from the U.S. Environmental Protection Agency for plants that discharge to the river, and from records of the Idaho Department of Environmental Quality for plants that discharge to land application.

Examples

```
str(div.ww)
```

`DownloadFile` *Download File from the Internet*

Description

This function downloads a file from the Internet.

Usage

```
DownloadFile(url, dest.dir = tempdir(), mode = NULL, extract = TRUE,
             max.attempts = 10L, wait.time = 30)
```

Arguments

`url`	`character`; the URL (or FTP) of a resource to be downloaded.
`dest.dir`	`character`; the directory where the downloaded file is saved.
`mode`	`character`; the mode with which to write the file, such as `"w"`, `"wb"` (binary), `"a"` (append) and `"ab"`.
`extract`	`logical`; if `TRUE`, an attempt is made to extract files from the file archive.
`max.attempts`	`integer`; the maximum number of attempts to download a file.
`wait.time`	`numeric`; the time to wait between download attempts, in seconds.

Details

This function requires package **RCurl**.

Value

Returns the file path(s) to the downloaded file (or uncompressed files).

Author(s)

J.C. Fisher, U.S. Geological Survey, Idaho Water Science Center

See Also

`CFILE`, `curlPerform`

Examples

```
## Not run:
url <- paste0("https://raw.githubusercontent.com/USGS-R/",
              "wrv/master/inst/extdata/alluvium.extent.zip")
files <- DownloadFile(url)
unlink(files)
## End(Not run)
```

`drains` *Drain Boundaries at Stanton Crossing and Silver Creek*

Description

Polygons used to define the locations of drain boundaries in the model domain. The polygons clip the line segments along the aquifer boundary (see `alluvium.extent`), and model cells intersecting these clipped-line segments are defined as boundary cells.

Usage

```
drains
```

Format

An object of `SpatialPolygonsDataFrame` class containing a set of 2 `Polygons` and a `data.frame` with the following variable:

Name is an identifier for the polygon.

cond is the drain conductance in square meters per day.

elev is the drain threshold elevation in meters above the North American Vertical Datum of 1988 (NAVD 88).

Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

U.S. Geological Survey, accessed on March 27, 2015; a Keyhole Markup Language (KML) file created in Google Earth with polygons drawn by hand in areas of known drains.

Examples

```
str(drains)

plot(drains, border = "red")
plot(alluvium.extent, add = TRUE)
```

`drybed` *Dry River Bed and Stream Fed Creek Conditions*

Description

A summary of dry river bed and stream fed conditions in the Wood River Valley, Idaho. Stream reaches on the Big Wood River between Glendale and Wood River Ranch are episodically dry; these dry periods are specified for calendar months when water diversions to the Bypass Canal begins before the 16th of the month and ends after the 15th of the month.

Usage

```
drybed
```

Format

A `data.frame` object with 12 records and the following variables:

Reach is the stream reach name.

199501, . . . , 201012 are logical values indicating whether the stream reach exhibits dry-bed conditions during a stress period.

Source

Idaho Department of Water Resources, accessed on January 6, 2016; compiled from Water District 37 records.

Examples

```
str(drybed)
```

---

efficiency *Irrigation Efficiency*

Description

The irrigation efficiency of irrigation entities.

Usage

```
efficiency
```

Format

A `data.frame` object with 88 records and the following variables:

EntityName is the name of the irrigation entity which the irrigation efficiency is applied.

Eff is the estimated irrigation efficiency, the ratio of the amount of water consumed by the crop to the amount of water supplied through irrigation.

Comment a brief comment on irrigation conditions.

Source

Idaho Department of Water Resources, accessed on July 9, 2015

Examples

```
str(efficiency)
```

`entity.components` *Irrigation Entity Components*

Description

Irrigation entities and their components in the Wood River Valley and surrounding areas. An irrigation entity is defined as an area served by a group of surface-water and/or groundwater diversion(s).

Usage

```
entity.components
```

Format

A `list` object with components of `SpatialPolygonsDataFrame-class`. There are a total of 192 components, one for each month in the 1995–2010 time period. Linked `data.frame` objects have the following variables:

EntitySrce a concatenation of the `EntityName` and `Source` character strings.

mean.et the mean evapotranspiration (ET) on irrigated and semi-irrigated lands in meters.

area the area of irrigated and semi-irrigated lands in square meters.

PrecipZone the name of the precipitation zone. See `precip.zones` dataset for details.

et.vol the volume of ET on irrigated and semi-irrigated lands in cubic meters.

precip.vol the volume of precipitation on irrigated and semi-irrigated lands in cubic meters.

cir.vol the volume of crop irrigation requirement in cubic meters (ET minus precipitation).

EntityName is the name of the irrigation entity.

Source is the water source: “Mixed” for a mixture of surface water and groundwater, “SW Only” for surface water only, and “GW Only” for groundwater only.

Source

Calculated from the `irr.entities`, `wetlands`, `public.parcels`, `irr.lands.year`, `et`, and `precipitation` datasets; see the ‘`package-datasets`’ vignette for the R code used in this calculation.

Examples

```
names(entity.components)
plot(entity.components[["199506"]])
print(entity.components[["199506"]])
```

`et` *Evapotranspiration*

Description

Evapotranspiration (ET) in the Wood River Valley and surrounding areas. Defined as the amount of water lost to the atmosphere via direct evaporation, transpiration by vegetation, or sublimation from snow covered areas.

Usage

```
et
```

Format

An object of `RasterStack` class containing 192 `RasterLayer` objects, one layer for each month in the 1995-2010 time period. Each cell on a layers surface grid represents the monthly depth of ET in meters. Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

Idaho Department of Water Resources, accessed on November 17, 2014

See Also

`et.method`

Examples

```
print(et)
plot(et[["199505"]])
```

`et.method` *Method Used to Calculate Evapotranspiration*

Description

The methods used to estimate monthly distributions of evapotranspiration (ET) rate.

Usage

```
et.method
```

Format

A `data.frame` object with 122 records with the following variables:

YearMonth The year and month during which the method was applied, with a required date format of `YYYYMM`.

ETMethod An identifier that indicates the method used to estimate ET values. Identifiers include: “Allen-Robison”, the Allen and Robison method (Allen and Robison, 2007); “METRIC”, the Mapping Evapotranspiration at high Resolution and with Internalized Calibration (METRIC) model (Allen and others, 2010a); “NDVI”, the Normalized Difference Vegetation Index (NDVI) method (Allen and others, 2010b); “Interpolation”, interpolation from known ET data; and “METRIC-NDVI”, a combination of METRIC and NDVI methods.

Source

Idaho Department of Water Resources, accessed on April 27, 2015

References

Allen, R., and Robison, C.W., 2007, Evapotranspiration and consumptive water requirements for Idaho, University of Idaho, Kimberly, Idaho.

Allen, R., Tasumi, M., Trezza, R., and Kjaersgaard, J., 2010a, METRIC mapping evapotranspiration at high resolution applications manual for Landsat satellite imagery version 2.07, University of Idaho, Kimberly, ID.

Allen, R., Robison, C.W., Garcia, M., Trezza, R., Tasumi, M., and Kjaersgaard, J., 2010b, ETrF vs NDVI relationships for southern Idaho for rapid estimation of evapotranspiration, University of Idaho, Kimberly, ID.

ET Idaho: http://data.kimberly.uidaho.edu/ETIdaho/

Examples

```
str(et.method)
```

ExportRasterStack *Export Raster Stack*

Description

This function writes a raster-stack, a collection of raster layers, to local directories using multiple file formats.

Usage

```
ExportRasterStack(rs, path, zip = "", col = rainbow(250, start = 0.0, end = 0.8))
```

Arguments

rs	RasterStack; a collection of RasterLayer objects with the same extent and resolution.
path	character; path name to write raster stack.
zip	character; if there is no zip program on your path (on windows), you can supply the full path to a 'zip.exe' here, in order to make a KMZ file.
col	character; a vector of colors.

Details

Five local directories are created under path and named after their intended file formats: Comma-Separated Values ('csv'), Portable Network Graphics ('png'), georeferenced TIFF ('tif'), R Data ('rda'), and Keyhole Markup Language ('kml'). For its reference system, 'kml' uses geographic coordinates: longitude and latitude components as defined by the World Geodetic System of 1984. Therefore, the conversion of gridded data between cartographic projections may introduce a new source of error.

To install 'zip.exe' on windows, download the latest binary version from the Info-ZIP website: select one of the given FTP locations, enter directory 'win32', download 'zip300xn.zip', and extract.

Value

None. Used for the side-effect files written to disk.

Author(s)

J.C. Fisher, U.S. Geological Survey, Idaho Water Science Center

See Also

writeRaster

Examples

```
## Not run:
f <- file.path(getwd(), "SIR2016-5080/ancillary/uncalibrated/data/rda/rasters.rda")
load(file = f)
ExportRasterStack(rs, tempdir())
## End(Not run)
```

`ExtractAlongTransect` *Extract Raster Values Along Transect Line*

Description

This function extracts values from raster layer(s) along a user defined transect line.

Usage

```
ExtractAlongTransect(transect, r)
```

Arguments

`transect`	`SpatialPoints` or `SpatialLines`; transect line or its vertices.
`r`	`RasterLayer`, `RasterStack`, or `RasterBrick`; the raster layer(s)

Details

The transect line is described using a simple polygonal chain. The transect line and raster layer(s) must be specified in a coordinate reference system.

Value

A `list` is returned with components of class `SpatialPointsDataFrame`. These components represent continuous piecewise line segments along the transect. The following variables are specified for each coordinate point in the line segment:

`dist`	`numeric`; the distance along the transect line.
`2, ..., n`	`numeric`; the extracted value for each raster layer in `r`, where column names match their respective raster layer name.

Author(s)

J.C. Fisher, U.S. Geological Survey, Idaho Water Science Center

See Also

`PlotCrossSection`

Examples

```
coords <- rbind(c(-100, -90), c(80, 90), c(80, 0), c(40, -40))
crs <- CRS("+proj=longlat +datum=WGS84")
transect <- SpatialPoints(coords, proj4string = crs)
r <- raster(nrows = 10, ncols = 10, ymn = -80, ymx = 80, crs = crs)
names(r) <- "value"
set.seed(0)
r[] <- runif(ncell(r))
r[4, 6] <- NA
plot(r, xlab = "x", ylab = "y")
lines(SpatialLines(list(Lines(list(Line(coords)), ID = "Transect")), proj4string = crs))
points(transect, pch = 21, bg = "red")
segs <- ExtractAlongTransect(transect, r)
for (i in 1:length(segs)) points(segs[[i]], col = "blue")

dev.new()
xlab <- "Distance along transect"
ylab <- "Raster value"
```

```
xlim <- range(vapply(segs, function(seg) range(seg@data[, "dist"]), c(0, 0)))
ylim <- range(vapply(segs, function(seg) range(seg@data[, "value"], na.rm = TRUE),
                     c(0, 0)))
plot(NA, type = "n", xlab = xlab, ylab = ylab, xlim = xlim, ylim = ylim)
for (i in 1:length(segs))
  lines(segs[[i]]@data[, c("dist", "value")], col = rainbow(length(segs))[i])
coords <- coordinates(transect)
n <- length(transect)
d <- cumsum(c(0, as.matrix(dist((coords)))[cbind(1:(n - 1), 2:n)]))
abline(v = d, col = "grey", lty = 2)
mtext(paste0("(", paste(head(coords, 1), collapse = ", "), ")"), adj = 0)
mtext(paste0("(", paste(tail(coords, 1), collapse = ", "), ")"), adj = 1)
```

`gage.disch` *Daily Mean Discharge at Streamgages*

Description

The daily mean discharge at streamgages in the Big Wood River Valley, Idaho. Discharge records bracket the 1992-2014 time period and are based on records with quality assurance code of approved ('A').

Usage

```
gage.disch
```

Format

A `data.frame` object with 8,315 records and the following variables:

Date is the date during which discharge was averaged.

13135500 is the daily mean discharge in cubic meters per day, recorded at the USGS 13135500 Big Wood River near Ketchum streamgage.

13139510 is the daily mean discharge in cubic meters per day, recorded at the USGS 13139510 Big Wood River at Hailey streamgage.

13140800 is the daily mean discharge in cubic meters per day, recorded at the USGS 13140800 Big Wood River at Stanton Crossing near Bellevue streamgage.

Source

National Water Information System (NWIS), accessed on January 8, 2015

Examples

```
str(gage.disch)

col <- c("red", "blue", "green")
ylab <- paste("Discharge in cubic", c("meters per day", "acre-foot per year"))
PlotGraph(gage.disch, ylab = ylab, col = col, lty = 1:3, conversion.factor = 0.29611)
leg <- sprintf("USGS %s", names(gage.disch)[-1])
legend("topright", leg, col = col, lty = 1:3, inset = 0.02, cex = 0.7,
       box.lty = 1, bg = "#FFFFFFE7")

graphics.off()
```

`gage.height` *Daily Mean Gage Height at Streamgages*

Description

The daily mean gage height at streamgages in the Big Wood River Valley, Idaho. Gage height records bracket the 1987-2014 and are based on records with quality assurance codes of working ('W'), in review ('R'), and approved ('A').

Usage

```
gage.height
```

Format

A `data.frame` object with 9,980 records and the following variables:

Date is the date during which gage height was averaged.

13135500 is the daily mean gage height in meters, recorded at the USGS 13135500 Big Wood River near Ketchum streamgage.

13139510 is the daily mean gage height in meters, recorded at the USGS 13139510 Big Wood River at Hailey streamgage.

13140800 is the daily mean gage height in meters, recorded at the USGS 13140800 Big Wood River at Stanton Crossing near Bellevue streamgages.

Source

Data queried from the National Water Information System (NWIS) database on December 15, 2014, by Ross Dickinson (USGS). Records recorded on May 26-28, 1991 and March 15-22, 1995 were reassigned quality assurance codes of 'I' because of assumed ice build-up. Missing data at the Big Wood River near Ketchum gage was estimated using a linear regression model based on recorded gage-height data at the Big Wood River at Hailey and Near Ketchum streamgages. Missing data at the Stanton Crossing near Bellevue gage was replaced with average gage-height values recorded at this gage.

Examples

```
str(gage.height)

col <- c("red", "blue", "green")
ylab <- paste("Gage height in", c("meters", "feet"))
PlotGraph(gage.height, ylab = ylab, col = col, lty = 1:3, conversion.factor = 3.28084)
leg <- sprintf("USGS %s", names(gage.height)[-1])
legend("topright", leg, col = col, lty = 1:3, inset = 0.02, cex = 0.7,
       box.lty = 1, bg = "#FFFFFFE7")

graphics.off()
```

`GetDaysInMonth` *Get Number of Days in a Year and Month*

Description

This function determines the number of days in a year and month.

Usage

```
GetDaysInMonth(x)
```

Arguments

`x`	`character` or `integer`; a vector of year and month values, with a required date format of `YYYYMM`.

Value

Returns an integer vector indicating the number of days in each year and month value specified in `x`.

Author(s)

J.C. Fisher, U.S. Geological Survey, Idaho Water Science Center

Examples

```
GetDaysInMonth(c("199802", "199804", "200412"))
```

`GetSeasonalMult` *Get Seasonal Multiplier*

Description

This function determines the seasonal fraction of the long-term mean value.

Usage

```
GetSeasonalMult(x, reduction, d.in.mv.ave, tr.stress.periods)
```

Arguments

`x`	`data.frame`; a time series with `Date` and `numeric` components.
`reduction`	`numeric`; is the signal amplitude reduction, a dimensionless quantity. Its magnitude should be greater than or equal to 1, where a value of 1 indicates no reduction in the signal amplitude.
`d.in.mv.ave`	`numeric`; is the number of days in the moving average subset.
`tr.stress.periods`	`Date`; a vector giving the start and end dates for each model stress period.

Details

A simple moving average is first calculated for each month using the previous data (such as the previous 9-months of stage data recorded at a streamgage). The seasonal average of the monthly moving average is then passed through a signal amplitude reduction algorithm. The reduced values are then divided by the mean of the seasonal reduced data to give the seasonal fraction of the mean (seasonal multiplier).

Value

An object of `data.frame` class with `Date` and `numeric` components; that is, the starting date and multiplier for each season.

Author(s)

J.C. Fisher and J.R. Bartolino, U.S. Geological Survey, Idaho Water Science Center

A.H. Wylie and J. Sukow, Idaho Department of Water Resources

Examples

```
tr.interval <- as.Date(c("1995-01-01", "2011-01-01"))
tr.stress.periods <- seq(tr.interval[1] , tr.interval[2], "1 month")
m <- GetSeasonalMult(gage.disch[, c("Date", "13139510")], 2, 273.932, tr.stress.periods)
f <- vapply(tributaries$Flow, function(i) m$multiplier * i, rep(0, nrow(m)))
colnames(f) <- tributaries$ID
d <- cbind(m, f)
str(d)
```

`GetWellConfig` *Get Well Completion and Pumping Rate in Model Space*

Description

This function determines well completions and pumping rates in model space. The pumping rate is specified for each model cell intersecting a well's open interval(s) and calculated by multiplying the estimated pumping demand by the cell's transmissivity fraction. The transmissivity fraction is calculated by dividing the cell's aquifer transmissivity by the sum of all transmissivity values for cells belonging to the same well. The transmissivity fraction calculation assumes saturated conditions in the model cell.

Usage

```
GetWellConfig(rs.model, wells, well.col, rate.col = NULL, lay2.hk.tol = 1e-02)
```

Arguments

`rs.model`	`RasterStack`; is composed of raster layers describing the model grid and hydraulic conductivity distribution: `lay1.top`, `lay1.bot`, `lay2.bot`, `lay3.bot`, `lay1.top`, `lay1.hk`, `lay2.hk`, and `lay3.hk`.
`wells`	`SpatialPointsDataFrame`; is the average pumping rate for each well during various times.
`well.col`	`character`; is the column name of the well identifier field.
`rate.col`	`character`; is a vector of column names for the pumping rate fields.
`lay2.hk.tol`	`numeric`; is the hydraulic conductivity tolerance for model cells in layer 2. Used to prevent pumping in the aquitard layer of the aquifer system. Pumping is prohibited in model layer 2 cells with hydraulic conductivity values less than `lay2.hk.tol` and a well opening isolated to layer 2; for these cases, pumping is allocated to the adjacent layer 1 cell.

Value

An object of `data.frame` class with the following components:

`...`	`numeric`; a unique identifier assigned to a well, its name is specified by `well.col`.
`lay, row, col`	`integer`; is the layer, row, and column number of a model cell, respectively.
`hk`	`numeric`; is the hydraulic conductivity of the model cell, in meters per day.
`thk`	`numeric`; is the vertical length of the well opening (open borehole or screen) in the model cell, in meters. A value of zero indicates that the well opening is unknown or below the modeled bedrock surface.
`frac`	`numeric`; is the transmissivity fraction for a model cell, where transmissivity is defined as `hk` multiplied by `thk`.
`...`	`numeric`; is the pumping rate allocated to the model cell for each time period specified by `rate.col`, in cubic meters per day. The pumping rate is calculated by multiplying the pumping demand for a well (specified in `wells`) by `frac`.

Author(s)

J.C. Fisher, U.S. Geological Survey, Idaho Water Science Center

A.H. Wylie, Idaho Department of Water Resources

Examples

```
## Not run: # see uncalibrated-model vignette
```

`hill.shading` *Land Surface Hill Shading*

Description

Hill shading of the Wood River Valley and surrounding area.

Usage

```
hill.shading
```

Format

An object of `RasterLayer` class. Each cell on the surface grid represents the hill shade. Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM). The spatial grid is composed of 3,108 rows and 2,360 columns, and has cell sizes that are constant at 20 meters by 20 meters.

Source

Calculated from the slope and aspect of the `land.surface` dataset using the `terrain` and `hillShade` functions; see the '`package-datasets`' vignette for the R code used in this calculation.

Examples

```
image(hill.shading, length(hill.shading), col = grey(0:255 / 255), asp = 1,
      axes = FALSE, xlab = "", ylab = "")
```

`idaho` *U.S. State of Idaho*

Description

The boundary of Idaho, a state in the northwestern region of the United States.

Usage

```
idaho
```

Format

An object of `SpatialPolygons` class containing 1 `Polygons`. Cartographic boundary at 5m (1:5,000,000) resolution.

Source

U.S. Department of Commerce, U.S. Census Bureau, Geography Division/Cartographic Products Branch. A simplified representation of the State of Idaho from the 2014 Census Bureau's MAF/TIGER geographic database.

Examples

```
plot(idaho, col = "#EAE2CF", border = "#BFA76F", lwd = 0.5)
print(idaho)
```

---

irr.entities *Irrigation Entities*

Description

The delineation of areas served by a group of surface-water and (or) groundwater diversions.

Usage

```
irr.entities
```

Format

An object of `SpatialPolygonsDataFrame` class containing 235 `Polygons` and a `data.frame` with the following variables:

EntityName is the name of the irrigation entity served by a group of diversions.

Source is the water source: "Mixed" for a mixture of surface water and groundwater, "SW Only" for surface-water only, and "GW Only" for groundwater only.

EntitySrce is a concatenation of the `EntityName` and `Source` character strings.

PrecipZone is the name of the precipitation zone. See `precip.zones` dataset for details.

Source

Idaho Department of Water Resources (IDWR), accessed on December 11, 2014; derived from IDWR water rights database, Blaine County tax lot data, and IDWR irrigated land classification files.

Examples

```
plot(irr.entities)
print(irr.entities)
```

---

irr.lands *Irrigated Lands*

Description

The irrigation classification of land area in the Wood River Valley and surrounding areas; available for years 1996, 2000, 2002, 2006, 2008, 2009, and 2010.

Usage

```
irr.lands
```

Format

A `list object` of length 7 with components of `SpatialPolygonsDataFrame-class`. The `data.frame` associated with each of the `SpatialPolygons` objects has the following variable:

Status is the status of land during the year reviewed, may be "irrigated", "semi-irrigated", or "non-irrigated".

Source

Idaho Department of Water Resources, accessed on November 17, 2014; polygons derived from U.S. Department of Agriculture Common Land Unit polygons with some refinement of polygons. Irrigation status interpreted using satellite imagery and aerial photography.

See Also

irr.lands.year

Examples

```
spplot(irr.lands[["2010"]], "Status")
print(irr.lands)
```

---

irr.lands.year *Irrigation Lands for a Given Year*

Description

The annual land classification for irrigation practices is only available for select years. For missing years, this dataset provides substitute years when land-classification was available (see irr.lands).

Usage

```
irr.lands.year
```

Format

A `data.frame` object with 16 records and the following variables:

Year is the year with a required date format of `YYYY`.

IL_Year is the substitute year with a required date format of `YYYY`.

Source

Idaho Department of Water Resources, accessed on April 25, 2014

Examples

```
str(irr.lands.year)
```

kriging.zones *Kriging Zones*

Description

The location of kriging zones in the Wood River Valley, used in parameter estimation.

Usage

```
kriging.zones
```

Format

An object of `SpatialPolygonsDataFrame` class containing 18 `Polygons` and a `data.frame` with the following variables:

Zone is the kriging zone, interpolation in this zone is based on the parameter values of pilot points located within this zone.

Layer is the model layer.

Name is the local name.

Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

Idaho Department of Water Resources

See Also

`pilot.points`

Examples

```
str(kriging.zones@data)
```

lakes *Lakes and Reservoirs*

Description

Lakes and reservoirs of the Wood River Valley and surrounding areas.

Usage

```
lakes
```

Format

An object of `SpatialPolygonsDataFrame` class containing 55 `Polygons`. Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

Idaho Department of Water Resources (IDWR), accessed on April 2, 2014

Examples

```
plot(lakes, col = "#CCFFFF", border = "#3399CC", lwd = 0.5)
str(lakes@data)
```

---

land.surface *Topography of Land Surface*

Description

The Wood River Valley (WRV) is a geologic feature located in south-central Idaho. This dataset gives the topography of the land surface in the WRV and vicinity.

Usage

```
land.surface
```

Format

An object of SpatialGridDataFrame class. Each cell on the surface grid represents an elevation in meters above the North American Vertical Datum of 1988 (NAVD 88). Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM). The spatial grid is composed of 565 rows and 429 columns, and has cell sizes that are constant at 100 meters by 100 meters.

Source

The National Elevation Dataset (NED) 1/3-arc-second raster (Gesch, 2007; Gesch and others, 2002), accessed on December 1, 2015. This dataset can be downloaded in a Esri ArcGRID format using the National Map Viewer. NED data are distributed in geographic coordinates in units of decimal degrees, and in conformance with the NAD 83. Elevation values are in meters above the NAVD 88. The west, east, south, and north bounding coordinates for this dataset are -115, -114, 43, and 44 decimal degrees, respectively. Post-processing includes: (1) project the values of the NED dataset into the `alluvium.thickness` spatial grid using bilinear interpolation, and (2) set values in cells where the elevation of the alluvium bottom is missing to `NA`.

References

Gesch, D.B., 2007, The National Elevation Dataset, in Maune, D., ed., Digital Elevation Model Technologies and Applications: The DEM Users Manual, 2nd Edition: Bethesda, Maryland, American Society for Photogrammetry and Remote Sensing, p. 99-118.

Gesch, D., Oimoen, M., Greenlee, S., Nelson, C., Steuck, M., and Tyler, D., 2002, The National Elevation Dataset: Photogrammetric Engineering and Remote Sensing, v. 68, no. 1, p. 5-11.

Examples

```
image(land.surface)
summary(land.surface)
```

`major.roads` *Major Roads*

Description

Major roads in the Wood River Valley and surrounding areas.

Usage

```
major.roads
```

Format

An object of `SpatialLinesDataFrame` class containing 475 `Lines`. Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

Idaho Department of Water Resources (IDWR), accessed on October 20, 2015

Examples

```
plot(major.roads)
str(major.roads@data)
```

`map.labels` *Map Labels*

Description

Map labels in the Wood River Valley and surrounding areas.

Usage

```
map.labels
```

Format

An object of `SpatialPointsDataFrame` class containing 40 points with the following variables:

label is the text to be written.

cex is the character expansion factor.

col, font is the color and font to be used.

srt is the string rotation in degrees.

Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

Best estimates of map label locations.

Examples

```
plot(map.labels, col = "red")
lab <- cbind(map.labels@coords, map.labels@data)
for (i in seq_len(nrow(lab))) {
  text(lab$x[i], lab$y[i], labels = lab$label[i], cex = lab$cex[i],
       col = lab$col[i], font = lab$font[i], srt = lab$srt[i])
}
```

`misc.locations` *Miscellaneous Locations*

Description

Miscellaneous locations in the Bellevue triangle area.

Usage

```
misc.locations
```

Format

An object of `SpatialPointsDataFrame` class containing 3 points with the following variable:

label is a description of the point location.

Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

Idaho Department of Water Resources (IDWR), accessed on December 23, 2015

Examples

```
plot(misc.locations, pch = 20, col = "red")
text(misc.locations, labels = misc.locations@data$label, pos = 3, cex = 0.6)
```

`misc.seepage` *Recharge from Miscellaneous Seepage Sites*

Description

Recharge from miscellaneous seepage sites in the Wood River Valley, Idaho.

Usage

```
misc.seepage
```

Format

A `data.frame` object with 2 records and the following variables:

RechSite is the name of the recharge site, see `bellevue.wwtp.ponds` and `bypass.canal` datasets.

199501, . . . , 201012 is the monthly volume of recharge during a stress period, in cubic meters. The variable name is specified as year and month.

Source

Idaho Department of Water Resources, accessed on January 15, 2015

Examples

```
str(misc.seepage)
```

`obs.wells` *Observation Wells*

Description

Observation wells in the Wood River Valley aquifer system.

Usage

```
obs.wells
```

Format

An object of `SpatialPointsDataFrame` class containing 776 points with the following variables:

id a unique well identifier used in this study.

SiteNo a unique well identifier within the National Water Information System (NWIS).

SITEIDIDWR a unique well identifier within the Idaho Department of Water Resources (IDWR) hydrologic database.

WELLNUMBER the USGS or IDWR site name for the well.

PESTNAME a unique well identifier for PEST.

METHODDRIL the drilling method.

TOTALDEPTH the depth at which drilling stopped, in feet.

OPENINGMIN the top of the screened interval, in feet.

OPENINGMAX the bottom of the screened interval, in feet.

COMPLETION the date on which the well drilling and construction stopped.

WCWELLID the well construction well identifier.

ALTITUDE the land surface elevation, in feet.

ALTMETHOD the method for obtaining the land surface elevation.

XYMETHOD the method of obtaining the spatial coordinates.

BASINNO the basin number.

COUNTYNAME the Idaho county name.

TWPRGE the township and range the well is located in.

SITENAME a local name for the well.

desc is a description of the well type.

TopOpen1 is the depth to the top of the first open interval in a groundwater well, in meters below land surface.

BotOpen1 is the depth to the bottom of the first open interval in a groundwater well, in meters below land surface.

TopOpen2 is not applicable.

BotOpen2 is not applicable.

Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

IDWR well construction database, accessed on June 29, 2015

See Also

obs.wells.head

Examples

```
plot(obs.wells)
str(obs.wells@data)
```

`obs.wells.head` *Hydraulic Heads in Observation Wells*

Description

Hydraulic-head (groundwater-level) measurements recorded in observation wells in the Wood River Valley aquifer system. Values are used as observations during the parameter estimation process.

Usage

```
obs.wells.head
```

Format

A `data.frame` object with 3,477 records and the following variables:

PESTNAME is a unique well identifier for PEST.

DateTime is the date and time when the measurement was recorded.

Head is the groundwater-level measurement (hydraulic head) in meters above NAVD 88.

Source

Idaho Department of Water Resources, accessed on June 26, 2015

See Also

obs.wells

Examples

```
str(obs.wells.head)
```

pilot.points *Pilot Points*

Description

Location of pilot points in the model domain.

Usage

```
pilot.points
```

Format

An object of `SpatialPointsDataFrame` class containing 106 points with the following variables:

Layer is the model layer.

Zone is the kriging zone.

Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

Idaho Department of Water Resources, accessed on June 11, 2015

See Also

`kriging.zones`

Examples

```
plot(pilot.points)
str(pilot.points@data)
```

PlotCrossSection *Plot Method for Cross Sections*

Description

This function creates a cross-section view of raster data. A key showing how the colors map to raster values is shown below the map.

Usage

```
PlotCrossSection(transect, rs, geo.lays = names(rs), val.lays = NULL,
                 wt.lay = NULL, asp = 1, ylim = NULL, max.dev.dim = c(43, 56),
                 n = NULL, breaks = NULL, pal = NULL, col = NULL, ylab = NULL,
                 unit = NULL, id = c("A", "A'"), labels = NULL,
                 explanation = NULL, features = NULL, max.feature.dist = Inf,
                 draw.key = TRUE, draw.sep = TRUE, is.categorical = FALSE,
                 contour.lines = NULL, bg.col = "#E1E1E1", wt.col = "#FFFFFFD8")
```

Arguments

`transect` — `SpatialLines`; the piecewise linear transect line.

`rs` — `RasterStack`; a collection of `RasterLayer` objects with the same extent and resolution.

`geo.lays` — `character`; a vector of names in `rs` that specify the geometry raster layers; these must be given in decreasing order, that is, from the upper most (such as land surface) to the lowest (such as a bedrock surface).

`val.lays` — `character`; a vector of names in `rs` that specify the value raster layers (optional). Values from the first layer are mapped as colors to the area between the first and second geometry layers; the second layer mapped between the second and third geometry layers, and so on.

`wt.lay` — `character`; the name in `rs` that specifies the water-table raster layer (optional).

`asp` — `numeric`; the *y/x* aspect ratio for spatial axes.

`ylim` — `numeric`; a vector of length 2 giving the minimum and maximum values for the *y*-axis.

`max.dev.dim` — `numeric`; a vector of length 2 giving the maximum width and height for the graphics device in picas, respectively. Suggested dimensions for single-column, double-column, and sidetitle figures are `c(21, 56)`, `c(43, 56)`, and `c(56, 43)`, respectively.

`n` — `integer`; the desired number of intervals to partition the range of raster values (optional).

`breaks` — `numeric`; a vector of break points used to partition the colors representing numeric raster values (optional).

`pal` — `function`; a color palette to be used to assign colors in the plot, `rainbow` by default.

`col` — `character`; a vector of colors to be used in the plot. This argument requires `breaks` specification for numeric raster values and overrides any palette function specification. For numeric values there should be one less color than breaks. Categorical data require a color for each category.

`ylab` — `character`; a label for the *y* axis.

`unit` — `character`; a label for the measurement unit of the *x*- and *y*-axes.

`id` — `character`; a vector of length 2 giving the labels for the end points of the transect line, defaults to *A–A'*.

`labels` — `list`; describes the location and values of labels in the color key. This list may include components `at` and `labels`, numeric and character vectors, respectively.

`explanation` — `character`; a label that describes the cell values.

`features` — `SpatialGridDataFrame`; point features adjacent to the transect line that are used as reference labels for the upper geometry layer.

`max.feature.dist` — `numeric`; the maximum distance from a point feature to the transect line, specified in the units of the `rs` projection.

`draw.key` — `logical`; if `FALSE`, a color key is not drawn.

`draw.sep` — `logical`; if `TRUE`, lines separating geometry layers are drawn.

`is.categorical` — `logical`; if `TRUE`, cell values in `val.lays` represent categorical data; otherwise, these data values are assumed continuous.

`contour.lines` — `list`; if specified, contour lines are drawn. The contours are described using a list of arguments supplied to `contour`. Passed arguments include `"drawlables"`, `"method"`, and `"col"`.

`bg.col` — `character`; the color used for the background of the area below the upper geometry raster layer.

`wt.col` — `character`; the color used for the water-table line.

Details

The dimensions of a new graphics device is dependent on the argument values of `max.dev.dim` and `asp`.

Value

Used for the side-effect of a new plot generated. Returns a `list` object with the following graphical parameters:

- `din` — `numeric`; the device dimensions `(width, height)`, in inches.
- `usr` — `numeric`; the extremes of the coordinates of the plotting region `(x1, x2, y1, y2)`.
- `heights` — `numeric`; the relative heights on the device `(upper, lower)` for the map and color-key plots.

Author(s)

J.C. Fisher, U.S. Geological Survey, Idaho Water Science Center

See Also

ExtractAlongTransect, AddScaleBar, AddColorKey

Examples

```
data(volcano)
x <- seq(from = 2667405, length.out = 61, by = 10)
y <- seq(from = 6478705, length.out = 87, by = 10)
r1 <- raster(volcano, xmn = min(x), xmx = max(x), ymn = min(y), ymx = max(y),
             crs = CRS("+init=epsg:27200"))
r2 <- min(r1[]) - r1 / 10
r3 <- r1 - r2
rs <- stack(r1, r2, r3)
names(rs) <- c("r1", "r2", "r3")

xy <- rbind(c(2667508, 6479501), c(2667803, 6479214), c(2667508, 6478749))
transect <- SpatialLines(list(Lines(list(Line(xy)), ID = "Transect")),
                         proj4string = crs(rs))

plot(r1)
lines(transect)
text(as(transect, "SpatialPoints"), labels = c("A", "BEND", "A'"), cex = 0.7,
     pos = c(3, 4, 1), offset = 0.1, font = 4)

graphics.off()
PlotCrossSection(transect, rs, geo.lays = c("r1", "r2"), val.lays = "r3",
                 ylab="Elevation", asp = 5, unit = "METERS",
                 explanation = "Vertical thickness between layers, in meters.")

graphics.off()
```

PlotGraph *Plot Method for Graphs*

Description

This function draws a sequence of points, lines, or box-and-whiskers using specified coordinates.

Usage

```
PlotGraph(x, y, xlab, ylab, asp = NA, xlim = NULL, ylim = NULL,
          xn = 5L, yn = 5L, ylog = FALSE, type = "s", lty = 1, lwd = 1,
          pch = NULL, col = NULL, bg = NA, fill = NULL, pt.cex = 1,
          seq.date.by = "year", scientific = NA, conversion.factor = NULL,
          boxwex = 0.8, center.date.labels = FALSE, bg.polygon = NULL)
```

Arguments

`x, y`	`Date`, `numeric`, `matrix`, or `data.frame`; vectors or matrices of data for plotting. The vector length or number of rows should match. If `y` is missing, then `x = x[, 1]` and `y = x[, 2:n]`.
`xlab`	`character`; title for *x* and axis.
`ylab`	`character`; a vector of length 2 giving the title for the 1st and 2nd *y* axes. The title for the 2nd *y* axis is optional and requires `conversion.factor` be specified.
`asp`	`numeric`; the *y/x* aspect ratio.
`xlim`	`numeric` or `Date`; the `x` limits `(x1, x2)` of the plot.
`ylim`	`numeric`; the `y` limits `(y1, y2)` of the plot.
`xn, yn`	`integer`; the desired number of intervals between tick-marks on the *x*- and *y*-axis, respectively.
`ylog`	`logical`; if `TRUE`, a logarithm scale is used for the *y* axis.
`type`	`character`; is the type of plot for each column of `y`, see `plot` for possible types. A box-and-whisker plot is drawn when `type = "box"`, with whiskers extending to the data extremes.
`lty`	`integer`; is the line type, see `par` for all possible types. Line types are used cyclically.
`lwd`	`numeric`; is the line width.
`pch`	`integer`; is the point type, see `points` for all possible types. Point types are used cyclically.
`col`	`character` or `function`; is the point or line color, see `par` for all possible ways this can be specified. Colors are used cyclically.
`bg`	`character`; a vector of background colors for the open plot symbols given by `pch = 21:25` as in `points`.
`fill`	`character`; a vector of fill colors for areas beneath (or above, direction towards 0) lines of type `"l"` or `"s"`.
`pt.cex`	`numeric`; expansion factor for the points.
`seq.date.by`	`character`, `numeric`, or `difftime`; is the increment of the date sequence, see `seq.Date` for all possible ways this can be specified.
`scientific`	`logical`; a vector of length 3 that indicates if axes labels should be encoded in nice scientific format. Vector elements correspond to the `x`-axis, `y`-axis, and second `y`-axis labels. Values are recycled as necessary.
`conversion.factor`	`numeric`; a conversion factor for the 2nd *y* axis.
`boxwex`	`numeric`; a scale factor to be applied to all boxes, only applicable for box-and-whisker plots.
`center.date.labels`	`logical`; if `TRUE`, date labels are horizontally centered between *x*-axis tickmarks.
`bg.polygon`	`list`; if specified, a background polygon is drawn. The polygon is described using a list of arguments supplied to the `polygon` function. Passed arguments include `"x"` and `"col"`.

Value

Used for the side-effect of a new plot generated.

Author(s)

J.C. Fisher, U.S. Geological Survey, Idaho Water Science Center

See Also

`matplot`, `boxplot`

Examples

```
n <- 50L

x <- as.Date("2008-07-12") + 1:n
y <- sample.int(n, replace = TRUE)
PlotGraph(x, y, ylab = paste("Random number in", c("meters", "feet")), type = "p",
          pch = 16, seq.date.by = "weeks", scientific = FALSE, conversion.factor = 3.28)

graphics.off()
y <- data.frame(lapply(1:3, function(i) sample(n, replace = TRUE)))
PlotGraph(x, y, ylab = "Random number", type = "s", pch = 1, seq.date.by = "days",
          scientific=TRUE)

graphics.off()
y <- sapply(1:3, function(i) sample((1:100) + i * 100, n, replace = TRUE))
m <- cbind(as.numeric(x), y)
col <- c("red", "gold", "green")
PlotGraph(m, xlab = "Number", ylab = "Random number", type = "b", pch = 15:17,
          col = col, pt.cex = 0.9)
legend("topright", LETTERS[1:3], inset = 0.05, col = col, lty = 1, pch = 15:17,
       pt.cex = 0.9, cex = 0.8, bg = "white")

graphics.off()
```

PlotMap *Plot Method for Maps*

Description

This function maps raster layer values. A key showing how the colors map to raster values is shown below the map.

Usage

```
PlotMap(r, layer = 1, att = NULL, n, breaks, xlim = NULL, ylim = NULL,
        zlim = NULL, asp = 1, extend.xy = FALSE, extend.z = FALSE,
        reg.axs = TRUE, trim.r = TRUE, dms.tick = FALSE, bg.lines = FALSE,
        bg.image = NULL, bg.image.alpha = 1, pal = NULL, col = NULL,
        max.dev.dim = c(43, 56), labels = NULL, scale.loc = "bottomleft",
        arrow.loc = NULL, explanation = NULL, credit = proj4string(r),
        shade = NULL, contour.lines = NULL, rivers = NULL, lakes = NULL,
        roads = NULL, draw.key = NULL, draw.raster = TRUE, useRaster)
```

Arguments

`r`	`RasterLayer`, `SpatialGridDataFrame`, or `CRS`; a raster layer with values to be plotted or a coordinate reference system (CRS).
`layer`	`integer`; the column to use in the `SpatialGridDataFrame`.
`att`	`numeric` or `character`; the variable identifying the levels attribute to use in the Raster Attribute Table (RAT). This argument requires `r` values that are of class `factor`.
`n`	`integer`; the desired number of intervals to partition the range of raster values (or `zlim` if specified) (optional).
`breaks`	`numeric`; a vector of break points used to partition the colors representing numeric raster values (optional).
`xlim`	`numeric`; a vector of length 2 giving the minimum and maximum values for the *x*-axis.

`ylim` — `numeric`; a vector of length 2 giving the minimum and maximum values for the *y*-axis.

`zlim` — `numeric`; a vector of length 2 giving the minimum and maximum raster values for which colors should be plotted.

`asp` — `numeric`; the *y/x* aspect ratio for spatial axes.

`extend.xy` — `logical`; if `TRUE`, the spatial limits will be extended to the next tick mark on the axes beyond the grid extent.

`extend.z` — `logical`; if `TRUE`, the raster value limits will be extended to the next tick mark on the color key beyond the measured range.

`reg.axs` — `logical`; if `TRUE`, the spatial data range is extended.

`trim.r` — `logical`; if `TRUE`, the outer rows and columns that consist of all `NA` values will be removed.

`dms.tick` — `logical`; if `TRUE`, the axes tickmarks are specified in degrees, minutes, and decimal seconds.

`bg.lines` — `logical`; if `TRUE`, the graticule is drawn in back of the raster layer using white lines and a grey background.

`bg.image` — `RasterLayer`; an image to drawn in back of the main raster layer `r`.

`bg.image.alpha` — `numeric`; the opacity of the background image from 0 to 1.

`pal` — `function`; a color palette to be used to assign colors in the plot, `rainbow` by default.

`col` — `character`; a vector of colors to be used in the plot. This argument requires `breaks` specification for numeric values of `r` and overrides any palette function specification. For numeric values there should be one less color than breaks. Factors require a color for each level.

`max.dev.dim` — `numeric`; a vector of length 2 giving the maximum width and height for the graphics device in picas, respectively. Suggested dimensions for single-column, double-column, and sidetitle figures are `c(21, 56)`, `c(43, 56)`, and `c(56, 43)`, respectively.

`labels` — `list`; describes the location and values of labels in the color key. This list may include components `at` and `labels`.

`scale.loc` — `character`; the position of the scale bar: `"bottomleft"`, `"topleft"`, `"topright"`, or `"bottomright"` to denote scale location.

`arrow.loc` — `character`; the position of the north arrow: `"bottomleft"`, `"topleft"`, `"topright"`, or `"bottomright"` to denote arrow location.

`explanation` — `character`; a label explaining the raster value.

`credit` — `character`; a label crediting the base map.

`shade` — `list`; if specified, a semi-transparent shade layer is drawn on top of the raster layer. This layer is described using a list of arguments supplied to `hillShade`. Passed arguments include `"angle"` and `"direction"`. Additional arguments also may be passed that control the vertical aspect ratio (`"z.factor"`) and color opacity (`"alpha"`).

`contour.lines` — `list`; if specified, contour lines are drawn. The contours are described using a list of arguments supplied to `contour`. Passed arguments include `"drawlables"`, `"method"`, and `"col"`.

`rivers` — `list`; if specified, lines are drawn. The lines are described using a list of arguments supplied to the plot method for `SpatialLines`. Passed arguments include `"x"`, `"col"`, and `"lwd"`.

`lakes` — `list`; if specified, polygons are drawn. The polygons are described using a list of arguments supplied to the plot method for `SpatialPolygons`. Passed arguments include `"x"`, `"col"`, `"border"`, and `"lwd"`. Bitmap images require a regular grid.

`roads` — `list`; if specified, lines are drawn. The lines are described using a list of arguments supplied to the plot method for `SpatialLines`. Passed arguments include `"x"`, `"col"`, and `"lwd"`.

`draw.key` — `logical`; indicates if a color key should be drawn.

`draw.raster` — `logical`; if `FALSE`, the raster image is not drawn.

`useRaster` — `logical`; if `TRUE`, a bitmap raster is used to plot `r` instead of polygons. If `UseRaster` is not specified, raster images are used when the `getOption("preferRaster")` is true.

Details

The dimensions of a new graphics device is dependent on the argument values of `max.dev.dim` and `asp`.

Value

Used for the side-effect of a new plot generated. Returns a `list` object with the following graphical parameters:

din	`numeric`; the device dimensions `(width, height)`, in inches.
usr	`numeric`; the extremes of the coordinates of the plotting region `(x1, x2, y1, y2)`.
heights	`numeric`; the relative heights on the device `(upper, lower)` for the map and color-key plots.

Author(s)

J.C. Fisher, U.S. Geological Survey, Idaho Water Science Center

See Also

AddScaleBar, AddColorKey

Examples

```
r <- raster(system.file("external/test.grd", package="raster"))
PlotMap(r, scale.loc = "topleft", dms.tick = TRUE, trim.r = TRUE)

graphics.off()
r <- raster(nrow = 10, ncol = 10)
r[] <- 1L
r[51:100] <- 2L
r[3:6, 1:5] <- 8L
r <- ratify(r)
rat <- levels(r)[[1]]
rat$land.cover <- c("Pine", "Oak", "Meadow")
rat$code <- c(12, 25, 30)
levels(r) <- rat
PlotMap(r, att = "land.cover", col = c("grey", "orange", "purple"))
PlotMap(r, att = "code")

graphics.off()
r <- alluvium.thickness
PlotMap(r@crs, bg.image = hill.shading, reg.axs = FALSE)
plot(alluvium.extent, border = "red", add = TRUE)
PlotMap(r, bg.image = hill.shading, bg.image.alpha = 0.6)
PlotMap(r, n = 10, extend.xy = TRUE)

graphics.off()
PlotMap(r, ylim = c(NA, 1360000), max.dev.dim = c(56, 43), n = 10, extend.z = TRUE,
        contour.lines = list(col = "#A9A9A9"))
plot(alluvium.extent, add = TRUE)
shade <- list(z.factor = 15, alpha = 0.4)
txt <- "Land surface elevation in meters above National Geodetic Vertical Datum of 1929."
ans <- PlotMap(r, ylim = c(NA, 1360000), max.dev.dim = c(56, 43), bg.lines = TRUE,
               shade = shade, arrow.loc = "topright", explanation = txt)

graphics.off()
```

pod.gw *Points of Diversion for Groundwater*

Description

Points of diversion for groundwater within the Wood River Valley model study area.

Usage

```
pod.gw
```

Format

A `data.frame` object with 1,755 records and the following variables:

WMISNumber is a unique number assigned to a water right point of diversion.

WaterRight is a number identifying a specific authorization to use water in a prescribed manner.

EntityName is the name of the irrigation entity the point of diversion is assigned to.

EntitySrce is the source of water for an irrigation entity. Possible sources of water include surface water, groundwater and mixed. Mixed source entities derive water from both groundwater and surface water.

Pdate is the priority date, the date the water right was established.

IrrRate is the irrigation rate in cubic meters per day, the maximum permitted water use rate associated with a water right.

Source

Idaho Department of Water Resources water rights database, accessed on December 11, 2014

See Also

`pod.wells`

Examples

```
summary(pod.gw)
```

pod.wells *Well Completions*

Description

Well completions for pumping wells in the Wood River Valley aquifer system.

Usage

```
pod.wells
```

Format

An object of SpatialPointsDataFrame class containing 1,243 points with the following variables:

WMISNumber is a unique number assigned to a water right point of diversion.

WellUse is the permitted use(s) for a groundwater well.

TopOpen1 is the depth to the top of the first open interval in a groundwater well, in meters below land surface.

BotOpen1 is the depth to the bottom of the first open interval in a groundwater well, in meters below land surface.

TopOpen2 is the depth to the top of the second open interval in a groundwater well, in meters below land surface.

BotOpen2 is the depth to the bottom of the second open interval in a groundwater well, in meters below land surface.

Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

Idaho Department of Water Resources water rights database, accessed on November 29, 2014

See Also

pod.gw

Examples

```
plot(pod.wells)
str(pod.wells@data)
```

---

precip.zones *Precipitation Zones*

Description

Precipitation zones specified for the Wood River Valley and surrounding areas. There are three precipitation zones, each containing a single weather station. Precipitation zones were distributed to maintain the geographic similarity between weather stations and zones.

Usage

```
precip.zones
```

Format

An object of SpatialPolygonsDataFrame class containing 3 Polygons and a data.frame with the following variables:

ID a numeric identifier assigned to the polygon.

PrecipZone the name of the precipitation zone: “Ketchum”, the northernmost zone with data from the Ketchum National Weather Service coop weather station. “Hailey”, the central zone with data from the Hailey 3NNW National Weather Service coop weather station. “Picabo”, the southernmost zone with data from the Picabo AgriMet weather station.

Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

Created using the northing midpoint between weather stations, see weather.stations dataset.

See Also

precipitation

Examples

```
col <- c("#D1F2A5", "#FFC48C", "#F56991")
plot(precip.zones, col = col)
legend("topright", legend = precip.zones@data$PrecipZone, fill = col, bty = "n")
plot(alluvium.extent, add = TRUE)

print(precip.zones)
```

---

precipitation — *Precipitation Rate*

Description

Precipitation rates in the Wood River Valley and surrounding areas.

Usage

```
precipitation
```

Format

A `data.frame` object with 582 records and the following variables:

YearMonth is the year and month during which precipitation were recorded, with a required date format of `YYYYMM`.

PrecipZone the name of the precipitation zone, see `precip.zones` dataset for details.

Precip is the monthly depth of precipitation accounting for spring melt, in meters.

Precip.raw is the monthly depth of precipitation recorded at the weather station, in meters.

Source

Idaho Department of Water Resources, accessed on April 24, 2015

References

National Oceanic and Atmospheric Administration's National Weather Service (NWS) Cooperative Observer Program

U.S. Bureau of Reclamation's Cooperative Agricultural Weather Network (AgriMet)

See Also

precip.zones, swe

Examples

```
str(precipitation)

d <- precipitation
d <- data.frame(Date = as.Date(paste0(d$YearMonth, "15"), format = "%Y%m%d"),
                Precip = d$Precip)
zones <- c("Hailey", "Ketchum", "Picabo")
d1 <- d[precipitation$PrecipZone == zones[1], ]
d2 <- d[precipitation$PrecipZone == zones[2], ]
d3 <- d[precipitation$PrecipZone == zones[3], ]
```

```
d <- merge(merge(d1, d2, by = "Date"), d3, by = "Date")

col <- c("red", "blue", "green")
ylab <- paste("Precipitation in", c("meters", "feet"))
PlotGraph(d, ylab = ylab, col = col, lty = 1:3, conversion.factor = 3.28084)
legend("topright", zones, col = col, lty = 1:3, inset = 0.02, cex = 0.7,
       box.lty = 1, bg = "#FFFFFFE7")

graphics.off()
```

`priority.cuts` *Priority Cuts*

Description

Priority cut dates applied to Big Wood River above Magic Reservoir and Silver Creek by Water District 37 and 37M at the end of each month.

Usage

```
priority.cuts
```

Format

A `data.frame` object with 112 records and the following variables:

YearMonth is the year and month during of the priority cut date, with a required date format of `YYYYMM`.

Pdate_BWR is the date of the priority cut applied to Big Wood River above Magic Reservoir by Water District 37.

Pdate_SC is the date of the priority cut applied to Silver Creek by Water District 37M.

Source

Idaho Department of Water Resources, accessed on November 17, 2014; compiled priority cut dates in effect at the end of each month, derived from Water District 37 and 37M records

Examples

```
str(priority.cuts)
```

`public.parcels` *Public Land Parcels*

Description

Non-irrigated public land parcels in the Wood River Valley and surrounding areas.

Usage

```
public.parcels
```

Format

An object of `SpatialPolygons` class containing 669 `Polygons`.

Source

Idaho Department of Water Resources, accessed on November 29, 2014; derived from Blaine County tax lots and aerial photography

Examples

```
plot(public.parcels)
print(public.parcels)
```

`r.canals` *Rasterized Canals*

Description

Canal systems of the Wood River Valley and surrounding areas transferred to raster cells.

Usage

```
r.canals
```

Format

An object of `RasterLayer` class with indexed cell values linked to a raster attribute table (RAT). The RAT is a `data.frame` with the following components:

ID the integer cell index.

COUNT the frequency of the cell index in the raster grid.

EntityName the name of the irrigation entity served by the canal system.

Source

Calculated by transferring the `canals` dataset to grid cells in the `land.surface` dataset using the `rasterize` function; see the '`package-datasets`' vignette for the R code used in this calculation.

Examples

```
plot(r.canals)
print(levels(r.canals)[[1]])
```

`reach.recharge` *Stream-Aquifer Flow Exchange Along River Reaches*

Description

Stream-aquifer flow exchange along river reaches specified as aquifer recharge. Values used as observations in parameter estimation.

Usage

```
reach.recharge
```

Format

A `data.frame` object with 192 records and the following variables:

YearMonth is the year and month of the measurement record, with a required date format of `YYYYMM`.

nKet_Hai the stream-aquifer flow exchange in the Big Wood River, near Ketchum to Hailey river reach, in cubic meters per day.

Hai_StC the stream-aquifer flow exchange in the Big Wood River, Hailey to Stanton Crossing river reach, in cubic meters per day.

WillowCr the stream-aquifer flow exchange in the Willow Creek river reach, in cubic meters per day.

SilverAbv the stream-aquifer flow exchange in Silver Creek, above Sportsman Access river reach, in cubic meters per day.

SilverBlw the stream-aquifer flow exchange in Silver Creek, Sportsman Access to near Picabo river reach, in cubic meters per day.

Details

A positive stream-aquifer flow exchange indicates aquifer recharge (a losing river reach).

Source

Calculated from continuous stream flow measurements, diversion data, return flow data, and exchange well data using a flow difference method to estimate groundwater inflows and outflows along a river reach, accessed on September 1, 2015. Derived from U.S. Geological Survey, Idaho Power Company, and Water District 37 and 37M records.

Examples

```
str(reach.recharge)
```

`ReadModflowBinary` *Read MODFLOW Binary File*

Description

This function reads binary output data from a MODFLOW run.

Usage

```
ReadModflowBinary(f, data.type = c("array", "flow"), rm.totim.0 = FALSE)
```

Arguments

`f`	`character`; the name of the binary file.
`data.type`	`character`; a description of how the data are saved.
`rm.totim.0`	`logical`; if `TRUE`, stress-periods at time zero are removed.

Details

This function reads binary head ('`.hds`'), drawdown ('`.ddn`'), and budget ('`.bud`') files generated from a MODFLOW run.

Value

Returns a `list` object of length equal to the number of times the data type is written to the binary file. List components are `list` objects with the following components:

`d`	`matrix` or `data.frame`; the data values.
`kstp`	`integer`; the time step.
`kper`	`integer`; the stress period.
`desc`	`character`; the variable name.
`ilay`	`integer`; the model-grid layer.
`delt`	`numeric`; the length of the current time step.
`pertim`	`numeric`; the time in the stress period.
`totim`	`numeric`; the total elapsed time.

Author(s)

J.C. Fisher, U.S. Geological Survey, Idaho Water Science Center

See Also

`SummariseBudget`

Examples

```
## Not run:
path <- file.path(getwd(), "SIR2016-5080/output/output.model1")
hds <- ReadModflowBinary(file.path(path, "wrv_mfusg.hds"), "array")
bud <- ReadModflowBinary(file.path(path, "wrv_mfusg.bud"), "flow")
## End(Not run)
```

`ReplaceInTemplate` *Replace Values in a Template Text*

Description

This function replaces keys within special markups in a template text with specified values. Pieces of R code can be put into the markups of the template text, and are evaluated during the replacement.

Usage

```
ReplaceInTemplate(text, replacement)
```

Arguments

`text`	`character`; a vector of character strings, the template text.
`replacement`	`list`; the values to replace in `text`.

Details

Keys are enclosed into markups of the form `$(KEY)` and `@{CODE}`.

Value

A vector of character strings after key replacement.

Author(s)

J.C. Fisher, U.S. Geological Survey, Idaho Water Science Center

References

This code was derived from the sensitivity`::template.replace` function.

Examples

```
text <- c("Hello $(name)!", "$(a) + $(b) = @{$(a) + $(b)}",
          "pi = @{format(pi, digits = 5)}")
replacement <- list(name = "world", a = 1, b = 2)
cat(ReplaceInTemplate(text, replacement), sep = "\n")
```

`river.reaches` *Major River Reaches*

Description

The major river reaches of the Wood River Valley, Idaho.

Usage

```
river.reaches
```

Format

An object of `SpatialLinesDataFrame` class containing 22 `Lines` and a `data.frame` with the following variables:

Reach is the name of the subreaches measured in U.S. Geological Survey (USGS) seepage survey.

BigReach is the name of the reaches for which time series targets are available for part or all of the calibration period.

DrainRiver is the model boundary assignment, either “drain” or “river”.

RchAvg is the estimated average reach gain in cubic meters per day for 1995-2010 based on a combination of gage data and the USGS seepage survey.

BigRAv is the estimated average reach gain in cubic meters per day for 1995-2010 based on gage data.

ReachNo is the reach number identifier.

Depth is the estimated average depth in meters of water in reach, measured from the air-water interface to the top of the riverbed sediments.

BedThk is the estimated thickness in meters of the saturated riverbed sediments.

Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

Idaho Department of Water Resources, accessed on June 6, 2015

Examples

```
plot(river.reaches)
str(river.reaches@data)
```

`RmSmallCellChunks` *Remove Small Cell Chunks*

Description

This function identifies cell chunks in a single raster grid layer, where a cell chunk is defined as a group of connected cells with non-missing values. The cell chunk with the largest surface area is preserved and all others removed.

Usage

```
RmSmallCellChunks(r)
```

Arguments

`r`	`RasterLayer`; a raster grid layer with cell values.

Value

The raster grid layer `r` with cell values in the smaller cell chunks set to `NA`.

Author(s)

J.C. Fisher, U.S. Geological Survey, Idaho Water Science Center

See Also

`clump`

Examples

```
set.seed(0)
r <- raster(ncols = 10, nrows = 10)
r[] <- round(runif(ncell(r)) * 0.7)
r <- clump(r)
plot(r)

r.new <- RmSmallCellChunks(r)
plot(r.new, zlim = range(r[], na.rm = TRUE))
```

`rs.entities` *Rasterized Monthly Irrigation Entities*

Description

Irrigation entities of the Wood River Valley and surrounding areas transferred to raster cells.

Usage

```
rs.entities
```

Format

An object of class `RasterStack` class containing a 192 `RasterLayer` objects, one layer for each month in the 1995-2010 time period. For each raster layer, indexed cell values are linked to a raster attribute table (RAT). The RAT is a `data.frame` with the following components:

ID the integer cell index.

COUNT the frequency of the cell index in the raster grid.

EntityName the name of the irrigation entity served by a group of diversions.

Source

Calculated by transferring the `entity.components` dataset to grid cells in the `land.surface` dataset using the `rasterize` function; see the 'package-datasets' vignette for the R code used in this calculation.

Examples

```
names(rs.entities)
plot(rs.entities[["199507"]])
print(levels(rs.entities[["199507"]])[[1]])
```

`rs.rech.non.irr` *Rasterized Monthly Recharge on Non-Irrigated Lands*

Description

Aerial recharge and discharge on non-irrigated lands of the Wood River Valley and surrounding areas transferred to raster cells.

Usage

```
rs.rech.non.irr
```

Format

An object of `RasterStack` class containing 192 `RasterLayer` objects, one layer for each month in the 1995-2010 time period. Each cell on a layers surface grid represents the monthly recharge in meters.

Source

Calculated from the `et`, `precipitation`, `precip.zones`, and `soils` datasets; see the 'package-datasets' vignette for the R code used in this calculation.

Examples

```
names(rs.rech.non.irr)
plot(rs.rech.non.irr[["199507"]])
```

`RunWaterBalance` *Run Water Balance*

Description

This function estimates areal recharge, and pumping demand at production wells. A water-balance approach is used to calculate these volumetric flow rate estimates, where positive values are flow into the aqufer system (groundwater recharge), and negative values are flow out of the system (groundwater discharge).

Usage

```
RunWaterBalance(tr.stress.periods, r.grid, eff, seep, ss.stress.periods = NULL,
                verbose = FALSE)
```

Arguments

`tr.stress.periods`
: `Date`; a vector of start and end dates for each stress period in the simulation.

`r.grid`
: `RasterLayer`; a raster of numeric values where `NA` indicates an ‘inactive’ cell in the top layer of the model.

`eff`
: `data.frame`; see `efficiency` dataset for details.

`seep`
: `data.frame`; see `canal.seep` dataset for details.

`ss.stress.periods`
: `Date`; a vector of start and end dates for stress periods used to create steady-state conditions.

`verbose`
: `logical`; indicates whether to return summary tables `natural.rech`, `inciden.rech`, `pumping.rech`; see ‘Value’ section for table formats.

Value

Returns a `list` object with the following components:

(1) Water-table flow data (combines natural and incidental groundwater recharge and discharge) are stored in `areal.rech`, an object of `RasterStack` class with raster layers for each model stress period; cell values are specified as volumetric flow rates in cubic meters per day.

(2) Production well pumping data are stored in `pod.rech`, an object of `data.frame` class with the following components:

`WMISNumber`
: `numeric`; a unique number assigned to a water right point of diversion.

`ss, 199501, ..., 201012`
: `numeric`; is the volumetric flow rate, specified for each stress period, in cubic meters per day.

(3) Natural groundwater recharge and discharge data are stored in `natural.rech`, an object of `data.frame` class with the following components:

`YearMonth`
: `factor`; is the calendar year and month `YYYYMM`.

`Area`
: `numeric`; the land-surface area of non-irrigated lands, in square meters.

`ET`
: `numeric`; evapotranspiration on non-irrigated lands, in cubic meters per month.

`Rech`
: `numeric`; is the volumetric flow rate, in cubic meters per month.

(4) Incidental groundwater recharge data are stored in `inciden.rech`, an object of `data.frame` class with the following components:

`EntityName`
: `character`; is the name of the irrigation entity.

`YearMonth`
: `factor`; is the calendar year and month `YYYYMM`.

`SWDiv`
: `numeric`; surface-water diversions, in cubic meters per month.

`SeepFrac`	`numeric`; canal seepage as a fraction of diversions, a dimensionless quantity.
`CanalSeep`	`numeric`; canal seepage, in cubic meters per month.
`SWDel`	`numeric`; surface-water delivered to field headgates, in cubic meters per month.
`area.sw`	`numeric`; area irrigated by only surface water, in square meters.
`et.sw`	`numeric`; evapotranspiration on lands irrigated by only surface water, in cubic meters per month.
`precip.sw`	`numeric`; precipitation on lands irrigated by only surface water, in cubic meters per month.
`cir.sw`	`numeric`; crop irrigation requirement on lands irrigated by only surface water, in cubic meters per month.
`area.mix`	`numeric`; area irrigated by both surface and groundwater, in square meters.
`et.mix`	`numeric`; evapotranspiration on lands irrigated by both surface and groundwater, in cubic meters per month.
`precip.mix`	`numeric`; precipitation on lands irrigated by both surface and groundwater, in cubic meters per month.
`cir.mix`	`numeric`; crop irrigation requirement on lands irrigated by both surface and groundwater, in cubic meters per month.
`area.gw`	`numeric`; area irrigated by only groundwater, in square meters.
`et.gw`	`numeric`; evapotranspiration on lands irrigated by only groundwater, in cubic meters per month.
`precip.gw`	`numeric`; precipitation on lands irrigated by only groundwater, in cubic meters per month.
`cir.gw`	`numeric`; crop irrigation requirement on lands irrigated by only groundwater, in cubic meters per month.
`Eff`	`numeric`; irrigation efficiency, a dimensionless quantity.
`GWDiv`	`numeric`; recorded groundwater diversions, in cubic meters per month.
`WWDiv`	`numeric`; inflow to municipal wastewater treatment plants, in cubic meters per month.
`hg.sw`	`numeric`; surface-water delivered to field headgates on lands irrigated by only surface water, in cubic meters per month.
`hg.mix`	`numeric`; surface-water delivered to field headgates on lands irrigation by both surface and groundwater, in cubic meters per month.
`rech.sw`	`numeric`; incidental groundwater recharge beneath lands irrigated by only surface water, in cubic meters per month.
`gw.dem.mix`	`numeric`; groundwater demand on lands irrigated by both surface and groundwater, in cubic meters per month.
`gw.div.est`	`numeric`; calculated groundwater diversions, in cubic meters per month.
`rech.mix`	`numeric`; incidental groundwater recharge beneath lands irrigated by both surface and groundwater, in cubic meters per month.
`gw.only`	`numeric`; groundwater demand on lands irrigated by only groundwater in entities with lands also irrigated by both surface and groundwater, in cubic meters per month.
`rech.muni`	`numeric`; incidental groundwater recharge beneath entities with lands irrigated by only groundwater and lands irrigated by both surface and groundwater, in cubic meters per month.
`gw.dem.gw`	`numeric`; groundwater demand on lands irrigated by only groundwater in entities without surface-water irrigation, in cubic meters per month.
`rech.gw`	`numeric`; incidental groundwater recharge beneath lands irrigated by only groundwater, in cubic meters per month.
`area.model`	`numeric`; area of the irrigation entity that is located in the model domain, in square meters.

Volumetric flow rates are calculated for their respective area in the irrigation entity—not just that part overlying the model area. Flow rate values are given this way in order to facilitate with quality assurance of the water-budget calculation. To calculate a simulated volumetric-flow rate: divide the flow rate by the affected area, and then multiply this value by the area of the irrigation entity that is located in the model domain (`area.model`).

(5) Well pumping data are also stored in `pumping.rech` (see `pod.rech` component), an object of `data.frame` class with the following components:

`WMISNumber`	`numeric`; a unique number assigned to a water right point of diversion.
`YearMonth`	`factor`; is the calendar year and month `YYYYMM`.
`Pumping`	`numeric`; is the volumetric rate of pumping, in cubic meters per month.

Author(s)

J.C. Fisher, U.S. Geological Survey, Idaho Water Science Center

J. Sukow and M. McVay, Idaho Department of Water Resources

See Also

UpdateWaterBudget

Examples

```
## Not run: # see wrv-introduction vignette
```

---

seepage.study — *Stream Seepage Study*

Description

A Wood River Valley stream seepage study with streamflow measurements made during the months of August 2012, October 2012, and March 2013.

Usage

```
seepage.study
```

Format

An object of `SpatialPointsDataFrame` class containing 82 points with the following variables:

Order an index used to preserve the downstream order of measurement sites.

Name a local name for the measurement site.

SiteNo a unique identifier for the measurement site within the National Water Information System (NWIS).

Type the type of measurement site: “Big Wood River”, “Willow Creek”, “Spring fed creeks”, “Silver Creek”, “Diversion”, “Exchange well inflow”, “Return”, and “Tributary”.

Comments an abbreviated site name.

Aug the volumetric flow rate measured during August 2012, in cubic meters per day.

Oct the volumetric flow rate measured during October 2012, in cubic meters per day.

Mar the volumetric flow rate measured during March 2013, in cubic meters per day.

Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

Derived from Bartolino (2014) seepage study, Idaho Department of Water Resources, Water District 37 and 37M flow records.

References

Bartolino, J.R., 2014, Stream seepage and groundwater levels, Wood River Valley, south-central Idaho, 2012–13: U.S. Geological Survey Scientific Investigations Report 2014-5151, 34 p., http://dx.doi.org/10.3133/sir20145151.

Examples

```
plot(seepage.study)
str(seepage.study@data)
```

`sensitivity` *PEST Sensitivity*

Description

Calibrated parameter values and composite sensitivities generated by PEST.

Usage

```
sensitivity
```

Format

A `data.frame` object with 336 records and the following variables:

parameter.desc is a description of the parameter.

ID is a unique identifier within the parameter type, such as an identifier for a pilot point or irrigation entity.

units is the parameter units.

start.value is the starting parameter value prior to model calibration.

lower.bound is the lower bound placed on the parameter value during the model-calibration process.

upper.bound is the upper bound placed on the parameter value during the model-calibration process.

parameter.name is the PEST parameter name.

group is the PEST parameter group.

value is the calibrated parameter value estimated by PEST.

comp.sens is the composite sensitivity generated during the final iteration of PEST.

rel.comp.sens is the relative composite sensitivity.

Source

Idaho Department of Water Resources, accessed on January 15, 2016

See Also

`pilot.points`, `irr.entities`, `river.reaches`, `drains`, `tributaries`

Examples

```
str(sensitivity)
```

`SetPolygons` *Analysis of Multi-Polygon Objects*

Description

Determines the intersection or difference between two multi-polygon objects.

Usage

```
SetPolygons(x, y, cmd = c("gIntersection", "gDifference"), buffer.width = NA)
```

Arguments

`x`	`SpatialPolygons*`; a multi-polygon object.
`y`	`SpatialPolygons*` or `Extent`; a multi-polygon object.
`cmd`	`character`; specifying `"gIntersection"`, the default, cuts out portions of the `x` polygons that overlay the `y` polygons. If `"gDifference"` is specified, only those portions of the `x` polygons falling outside the `y` polygons are copied to the output polygons.
`buffer.width`	`numeric`; expands or contracts the geometry of `y` to include the area within the specified width, see `gBuffer`. Specifying `NA`, the default, indicates no buffer.

Details

This function tests if the resulting geometry is valid, see `gIsValid`.

Value

Returns an object of `SpatialPolygons*` class.

Author(s)

J.C. Fisher, U.S. Geological Survey, Idaho Water Science Center

See Also

`gIntersection`, `gDifference`

Examples

```
library(sp)

m1a <- matrix(c(17.5, 24.7, 22.6, 16.5, 55.1, 55.0, 61.1, 59.7), nrow = 4, ncol = 2)
m1b <- m1a
m1b[, 1] <- m1b[, 1] + 11
p1 <- SpatialPolygons(list(Polygons(list(Polygon(m1a, FALSE), Polygon(m1b, FALSE)), 1)))
plot(p1, col = "blue")

m2a <- matrix(c(19.6, 35.7, 28.2, 60.0, 58.8, 64.4), nrow = 3, ncol = 2)
m2b <- matrix(c(20.6, 30.9, 27.3, 56.2, 53.8, 51.4), nrow = 3, ncol = 2)
p2 <- SpatialPolygons(list(Polygons(list(Polygon(m2a, FALSE), Polygon(m2b, FALSE)), 2)))
plot(p2, col = "red", add = TRUE)

p <- SetPolygons(p1, p2, "gIntersection")
plot(p, col = "green", add = TRUE)

p <- SetPolygons(p2, p1, "gDifference")
plot(p, col = "purple", add = TRUE)
```

soils *Soil Units*

Description

Representation of mapped surficial soil units created by the Idaho Office of the National Resource Conservation Service (NRCS). Soils have been assigned a percolation rate based on the average, saturated hydraulic conductivity of the soils as classified using the Unified Soil Classification System (USCS).

Usage

```
soils
```

Format

An object of `SpatialPolygonsDataFrame` class containing 718 `Polygons` and a `data.frame` with the following variables:

GroupSymbol is a soil class identifier.

SoilLayer is an identifier used to differentiate the soil data source used to create the soils map. Data sources are either 'USCS' or 'STATSGO', the NRCS State Soil Geographic Data Base.

SoilClass is a description of the soil class.

MinRate is the lower percolation rate limit for the soil class, in meters per month.

MaxRate is the upper percolation rate limit for the soil class, in meters per month.

PercolationRate is the percolation rate in meters per month.

Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

Idaho Department of Water Resources, accessed on April 22, 2015

Examples

```
spplot(soils, "PercolationRate")
str(soils@data)
```

streamgages *Streamgages*

Description

Select streamgages in the Wood River Valley.

Usage

```
streamgages
```

Format

An object of `SpatialPointsDataFrame` class containing 9 points and a `data.frame` with the following variable:

SiteNo the unique site number for the streamgage.

SiteName the official name of the streamgage.

Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

National Water Information System (NWIS), accessed on May 29, 2015.

Examples

```
str(streamgages)
```

`streams.rivers` *Streams and Rivers*

Description

Streams and rivers of the Wood River Valley and surrounding areas.

Usage

```
streams.rivers
```

Format

An object of `SpatialLinesDataFrame` class containing 581 `Lines`. Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

Idaho Department of Water Resources (IDWR), accessed on April 2, 2014

Examples

```
plot(streams.rivers, col = "#3399CC")
str(streams.rivers@data)
```

subreach.recharge *Stream-Aquifer Flow Exchange Along River Subreaches*

Description

Stream-aquifer flow exchange along river subreaches specified as aquifer recharge. Values used as observations in parameter estimation.

Usage

```
subreach.recharge
```

Format

A `data.frame` object with 19 records and the following variables:

ReachNo is the subreach number identifier.

Reach is the name of the subreach.

BigReachNo is the reach number identifier.

BigReach is the name of the reach.

Aug the estimated volumetric flow rate measured during August 2012, in cubic meters per day.

Oct the estimated volumetric flow rate measured during October 2012, in cubic meters per day.

Mar the estimated volumetric flow rate measured during March 2013, in cubic meters per day.

Details

A positive stream-aquifer flow exchange indicates aquifer recharge (also know as a losing river subreach).

Source

Flow values calculated from `seepage.study` data.

Examples

```
str(subreach.recharge)
```

SummariseBudget *Summarize Volumetric Water Budget*

Description

Summarizes volumetric flow rates for boundary condition types. Splits the budget data into subsets, computes summary statistics for each, and returns the result in a summary table.

Usage

```
SummariseBudget(budget, desc = c("wells", "drains", "river leakage"))
```

Arguments

`budget`	`character` or `list`; either a description of the path to the MODFLOW Budget File or the returned results from a call to the `ReadModflowBinary` function.
`desc`	`character`; a vector of MODFLOW package identifiers. Data of this package type is included in the summary table.

Details

The budget[[i]]$d data table component must contain a numeric id field, see WriteModflowInput for variable description. Subsets are grouped by the MODFLOW package identifier (desc), stress period (kper), time step (kstp), and location identifier (id).

Value

Returns an object of data.frame class with the following components:

desc	factor; is the MODFLOW package identifier.
kper	integer; is the stress period.
kstp	integer; is the time step.
id	integer; is a location identifier.
delt	numeric; is the length of the current time step.
pertim	numeric; is the time in the stress period.
totim	numeric; is the total elapsed time.
count	integer; is the number of cells in each subset.
flow.sum	numeric; is the total volumetric flow rate.
flow.mean	numeric; is the mean volumetric flow rate.
flow.median	numeric; is the median volumetric flow rate.
flow.sd	numeric; is the standard deviation of the volumetric flow rate.
flow.dir	factor; is the flow direction where "in" and "out" indicate water entering and leaving the groundwater system, respectively.

Author(s)

J.C. Fisher, U.S. Geological Survey, Idaho Water Science Center

Examples

```
## Not run:
f <- file.path(getwd(), "SIR2016-5080/output/output.model1/wrv_mfusg.bud")
d <- SummariseBudget(f)
str(d)
## End(Not run)
```

---

swe *Snow Water Equivalent*

Description

Average daily snow water equivalent (SWE) at weather stations in the Wood River Valley and surrounding areas.

Usage

```
swe
```

Format

A `data.frame` object with 366 records and the following variables:

MonthDay is the month and day, with a required date format of `MMDD`.

Choco is the daily SWE recorded at the Chocolate Gulch snow telemetry (SNOTEL) weather station.

Hailey is the daily SWE recorded at the Hailey Ranger Station at Hailey hydrometeorological automated data system (HADS) weather station.

Picabo is the daily SWE recorded at the Picabo PICI HADS weather station.

Source

Idaho Department of Water Resources, accessed on April 24, 2015

See Also

`weather.stations`, `precip.zones`, `precipitation`

Examples

```
str(swe)
```

`ToScientific` *Format for Scientific Notation*

Description

This function formats numbers in scientific notation $m \times 10^n$.

Usage

```
ToScientific(x, digits = format.info(as.numeric(x))[2],
             lab.type = c("latex", "plotmath"))
```

Arguments

`x`	`numeric`; a vector of numbers.
`digits`	`integer`; the number of digits after the decimal point for the mantissa.
`lab.type`	`character`; by default, LaTeX formatted strings for labels are returned. Alternatively, `lab.type = "plotmath"` returns `plotmath`-compatible expressions.

Value

For the default `lab.type = "latex"`, a `character` vector of the same length as argument `x`. And for `lab.type = "plotmath"`, an `expression` of the same length as `x`, typically with elements of the form `m x 10^n`. In order to comply with Section 508, an "x" is used as the label separator for the `plotmath` type—rather than the more common "`%*%`" seperator.

Author(s)

J.C. Fisher, U.S. Geological Survey, Idaho Water Science Center

Examples

```
x <- c(-1e+09, 0, NA, pi * 10^(-5:5))
ToScientific(x, digits = 2)
ToScientific(x, digits = 2, lab.type = "plotmath")
```

tributaries *Tributary Basin Underflow*

Description

The location and average flow conditions for model boundaries in the major tributary canyons and upper part of the Wood River Valley, south-central Idaho.

Usage

```
tributaries
```

Format

An object of `SpatialPolygonsDataFrame` class containing a set of 22 `Polygons` and a `data.frame` with the following variable:

Name is the tributary name.

MinLSD is the minimum land-surface datum (elevation) along the transect, in meters above the North American Vertical Datum of 1988 (NAVD 88).

BdrkDepth is the mean saturated thickness along the transect line, in meters; estimated as the distance between the estimated water table and bedrock elevations.

TribWidth is the width of the tributary canyon, or length of the transect line, in meters.

LandGrad is the land surface elevation gradient perpendicular to the cross-sectional transect line, a dimensionless quantity.

K is the hydraulic conductivity, in meters per day.

SatArea is the estimated saturated cross-sectional area, in square meters; its geometry is represented as the lower-half of an ellipse with width and height equal to `TribWidth` and `BdrkDepth`, respectively.

DarcyFlow is the groundwater flow rate, in cubic meters per day, calculated using a Darcian analysis.

BasinArea is the land-surface area defined by the basin above the cross-sectional transect line.

BasinAreaType is a label that describes the relative basin size; where `"big"` indicates a basin area greater than 10 square miles (25.9 square kilometers), and `"small"` indicates a basin area that is less than this breakpoint value.

PrecipRate is the mean precipitation rate within the basin area, in meters per day.

PrecipFlow is the mean precipitation flow rate, in cubic meters per day, calculated by multiplying `PrecipRate` by `BasinArea`.

FlowRatio is the ratio of darcy flow rate to precipitation flow rate, or `DarcyFlow` divided by `PrecipFlow`, a dimensionless quantity.

Flow is the estimated average volumetric flow rate, in cubic meters per day.

Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

U.S. Geological Survey, accessed on July 2, 2015; a Keyhole Markup Language (KML) file created in Google Earth with polygons drawn by hand in areas of known specified flow boundaries. Transect lines are defined by the polygon boundaries within the extent of alluvium aquifer (see `alluvium.extent` dataset). The land surface gradient (`LandGrad`) was estimated from the `land.surface` dataset. Hydraulic conductivity (`K`) is the average of two geometric means of hydraulic conductivity in the unconfined aquifer taken from table 2 in Bartolino and Adkins (2012). The U.S. Geologic Survey StreamStats tool (Ries and others, 2004) was used to delineate the basin area (`BasinArea`) and estimate the precipitation rate (`PrecipRate`). See the '`package-datasets`' vignette for the R code used to calculate the flow estimates (`Flow`).

References

Bartolino, J.R., and Adkins, C.B., 2012, Hydrogeologic framework of the Wood River Valley aquifer system, south-central Idaho: U.S. Geological Survey Scientific Investigations Report 2012-5053, 46 p., available at `http://pubs.usgs.gov/sir/2012/5053/`.

Ries, K.G., Steeves, P.A., Coles, J.D., Rea, A.H., and Stewart, D.W., 2004, StreamStats–A U.S. Geological Survey web application for stream information: U.S. Geological Survey Fact Sheet FS-2004-3115, 4 p., available at `http://pubs.er.usgs.gov/usgspubs/fs/fs20043115`.

Examples

```
plot(tributaries, border = "red")
plot(alluvium.extent, add = TRUE)
str(tributaries@data)
```

`tributary.streams` *Streams and Rivers*

Description

Tributary streams of the upper Wood River Valley and surrounding areas.

Usage

```
tributary.streams
```

Format

An object of `SpatialLinesDataFrame` class containing 88 `Lines`. Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

Idaho Department of Water Resources, accessed on June 1, 2015

Examples

```
plot(tributary.streams, col = "#3399CC")
str(tributary.streams@data)
```

`UpdateWaterBudget` *Update Water Budget*

Description

This function runs the water budget calculation and updates the MODFLOW Well Package file. It is executed during each iteration of PEST and my be run in an interactive R session to initialize the parameter estimation files.

Usage

```
UpdateWaterBudget(dir.run, id, qa.tables = c("none", "si", "english"),
                  ss.interval = NULL, iwelcb = 0L)
```

Arguments

`dir.run` character; the path name of the directory to read/write model files.

`id` character; a short identifier (file name) for model files.

`qa.tables` character; indicates if quality assurance tables are written to disk; by default "none" of these tables are written. Values of "si" and "english" indicate that table values are written in metric and English units, respectively.

`ss.interval` Date or character; a vector of length 2 specifying the start and end dates for the period used to represent steady-state boundary conditions. That is, recharge values for stress periods coinciding with this time period are averaged and used as steady-state boundary conditions. The required date format is YYYY-MM-DD. This argument overrides the ss.stress.periods object in the 'model.rda' file, see 'Details' section for additional information.

`iwelcb` integer; is a flag and unit number. If equal to zero, the default, cell-by-cell flow terms resulting from conditions in the MODFLOW Well Package will not be written to disk. This default value is appropriate for model calibration, where MODFLOW run times are kept as short as possible. If greater than zero, the cell-by-cell flow terms are written to disk. See the MODFLOW Name File ('*.nam') for the unit number associated with the budget file ('*.bud').

Details

Files read during execution, and located within the dir.run directory, include the MODFLOW hydraulic conductivity reference files 'hk1.ref', 'hk2.ref', and 'hk3.ref' corresponding to model layers 1, 2, and 3, respectively. Hydraulic conductivity values are read from a two-dimensional array in matrix format with 'white-space' delimited fields. And a binary data file 'model.rda' containing the following serialized R objects: rs, misc, trib, tr.stress.periods, and ss.stress.periods.

rs is an object of RasterStack class with raster layers "lay1.top", "lay1.bot", "lay2.bot", and "lay3.bot". These raster layers describe the geometry of the model grid; that is, the upper and lower elevation of model layer 1, and the bottom elevations of model layers 2 and 3. In addition to these layers, rs includes ancillary raster layers "lay1.zones", "lay2.zones", and "lay3.zones" describing the distribution of hydrogeologic zones in the model grid. Missing cell values (equal to NA) indicate inactive model cells lying outside of the model domain.

misc is a data.frame object with miscellaneous seepage, such as from the 'Bellevue Waste Water Treatment Plant ponds' and the 'Bypass Canal'. This object is comprised of the following components: **lay**, **row**, **col** are integer values specifying a model cell's layer, row, and column index, respectively; and **ss**, **199501**, **199502**, . . . , **201012** are numeric values of elevation during each stress period, respectively, in meters above the North American Vertical Datum of 1988.

trib is a data.frame object with default values for the long-term mean underflows in each of the tributary basins. The object is comprised of the following components: **Name** is a unique identifier for the tributary basin; **lay**, **row**, **col** are integer values of a model cell's layer, row, and column index, respectively; and **ss**, **199501**, **199502**, . . . , **201012** are numeric values of underflow during each stress period, respectively, in cubic meters per day.

tr.stress.periods is a vector of Date values giving the start and end dates for stress periods in the model simulation period (1995–2010).

ss.stress.periods is a vector of Date values giving the start and end dates for stress periods used to define steady-state conditions.

reduction is a numeric default value for the signal amplitude reduction algorithm, a dimensionless quantity.

d.in.mv.ave is a numeric default value for the number of days in the moving average subset.

Value

Returns an object of difftime class, the runtime for this function. Used for the side-effect of files written to disk.

A MODFLOW Well Package file '<id>.wel' is always written to disk; whereas, parameter estimation files 'seep.csv', 'eff.csv', and 'trib.csv', and a script file 'UpdateBudget.bat', are only written if they do not already exist. The script file may be used to automate the execution of this function from a file manager (such as, Windows Explorer).

The 'seep.csv' file stores as tabular data the canal seepage fraction for each of the irrigation entities. Its character and numeric data fields are delimited by commas (a comma-separated-value [CSV] file). The first line is reserved for field names EntityName and SeepFrac.

The 'eff.csv' file stores as tabular data the irrigation efficiency for each of the irrigation entities. Its character and numeric data fields are delimited by commas. The first line is reserved for field names EntityName and Eff.

The 'trib.csv' file stores as tabular data the underflow boundary conditions for each tributary basin. Its character and numeric data fields are delimited by commas. The first line is reserved for field names Name and Value. Data records include a long-term mean flow multiplier for each of the tributary basins (name is the unique identifier for the tributary), a record for the amplitude reduction (reduction), and a record for the number of days in the moving average (d.in.mv.ave).

If the qa.tables argument is specified as either "si" or "english", quality assurance tables are written to disk as CSV files ('qa-*.csv'). Volumetric flow rate data within these tables is described in the 'Value' section of the RunWaterBalance function; see returned list components natural.rech, inciden.rech, and pumping.rech. The well configuration data are described in the 'Value' section of the GetWellConfig function.

Author(s)

J.C. Fisher, U.S. Geological Survey, Idaho Water Science Center

See Also

RunWaterBalance, GetSeasonalMult

Examples

```
## Not run:
dir.run <- "C:/Users/jfisher/Documents/SIR2016-5080/model/model1"
UpdateWaterBudget(dir.run, "wrv_mfusg", qa.tables = "si",
                  ss.interval = c("1998-01-01", "2011-01-01"))
## End(Not run)
```

---

weather.stations *Weather Stations*

Description

Weather stations in the WRV and surrounding areas.

Usage

```
weather.stations
```

Format

An object of SpatialPointsDataFrame class containing 5 points and the following variables:

name is the name of the weather station.

id is a unique identifier for the weather station.

type is the type of weather stations: "HADS", a Hydrometeorological Automated Data System operated by the National Weather Service Office of Dissemination; "AgriMet", a satellite-telemetry network of automated agricultural weather stations operated and maintained by the Bureau of Reclamation; and "SNOTEL", an automated system of snowpack and related climate sensors operated by the Natural Resources Conservation Service.

organization is the managing organization.

elevation is the elevation of the weather station in meters above the North American Vertical Datum of 1988 (NAVD 88).

Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

National Oceanic and Atmospheric Administration (NOAA), Bureau of Reclamation, Natural Resources Conservation Service (NRCS), accessed on May 1, 2015

Examples

```
plot(alluvium.extent)
plot(weather.stations, col = "red", add = TRUE)
str(weather.stations@data)
```

`wetlands` *Wetlands*

Description

Wetlands in the Wood River Valley and surrounding areas.

Usage

```
wetlands
```

Format

An object of `SpatialPolygons` class containing 3,024 `Polygons`.

Source

U.S. Fish and Wildlife Service National Wetlands Inventory, accessed on April 2, 2014

Examples

```
plot(wetlands, col = "#CCFFFF", border = "#3399CC", lwd = 0.5)
print(wetlands)
```

`wl.200610` *Groundwater-Level Contours for October 2006*

Description

Groundwater-level contours with a 20 foot (6.096 meter) contour interval for the unconfined aquifer in the Wood River Valley, south-central Idaho, representing conditions during October 2006.

Usage

```
wl.200610
```

Format

An object of `SpatialLinesDataFrame` class containing 265 `Lines` and a `data.frame` with the following variables:

CONTOUR is the groundwater elevation contour value in meters above the North American Vertical Datum of 1988 (NAVD 88).

certainty is the certainty of the groundwater-level contour based on data position and density, specified as `"good"` or `"poor"`.

Geographic coordinates are in units of meters, in conformance with the North American Datum of 1983 (NAD 83), and placed in the Idaho Transverse Mercator projection (IDTM).

Source

This dataset is from Plate 1 in Skinner and others (2007), and available on the WRD NSDI Node.

References

Skinner, K.D., Bartolino, J.R., and Tranmer, A.W., 2007, Water-resource trends and comparisons between partial development and October 2006 hydrologic conditions, Wood River Valley, south-central, Idaho: U.S. Geological Survey Scientific Investigations Report 2007-5258, 30 p., available at http://pubs.usgs.gov/sir/2007/5258/

Examples

```
is.good <- wl.200610@data$certainty == "good"
plot(wl.200610[is.good, ], col = "blue")
plot(wl.200610[!is.good, ], col = "red", lty = 2, add = TRUE)
str(wl.200610@data)
```

WriteModflowInput *Write MODFLOW Input Files*

Description

This function generates and writes input files for a MODFLOW simulation of groundwater flow in the Wood River Valley (WRV) aquifer system.

Usage

```
WriteModflowInput(rs.model, rech, well, trib, misc, river, drain, id, dir.run,
                  is.convertible = FALSE, ss.perlen = 0L,
                  tr.stress.periods = NULL, ntime.steps = 4L,
                  mv.flag = 1e+09, auto.flow.reduce = FALSE, verbose = TRUE)
```

Arguments

`rs.model`	`RasterStack`; a collection of `RasterLayer` objects with the same extent and resolution, see 'Details' for required raster layers.
`rech`	`data.frame`; is the areal recharge rate, in cubic meters per day. Variables describe the model cell location (`lay`, `row`, `col`) and volumetric rate during each stress period (`ss`, `199501`, `199502`, ..., `201012`).
`well`	`data.frame`; is the well pumping at point locations in cubic meters per day. Variables describe the model cell location and volumetric rate during each stress period.
`trib`	`data.frame`; is the incoming flows from the major tributary canyons. Variables describe the model cell location and volumetric rate during each stress period.
`misc`	`data.frame`; is recharge from miscellaneous seepage sites in cubic meters per day. Variables describe the model cell location and volumetric rate during each stress period.
`river`	`data.frame`; is the river conditions. Variables describe the model cell location, river conductance (`cond`) in square meters per day, river bottom elevation (`bottom`) in meters above the North American Vertical Datum of 1988 (NAVD 88), and a numeric river reach identifier (`id`).
`drain`	`data.frame`; is the drain conditions for groundwater outlet boundaries. Variables describe the model cell location, drain threshold elevation (`elev`) in meters above the NAVD 88, drain conductance (`cond`) in square meters per day, and a numeric identifier (`id`) indicating the drains general location.
`id`	`character`; a short identifier for the model run.
`dir.run`	`character`; the path name of the directory to write model input files.

`is.convertible`
: `logical`; if `TRUE`, indicates model layers are 'convertible', with transmissivity computed using upstream water-table depth. Otherwise, model layers are 'confined' and transmissivity is constant over time.

`ss.perlen` `integer` or `difftime`; the length of the steady-state stress period in days.

`tr.stress.periods`
: `Date`; a vector of start times for each stress period in the transient simulation. If missing, only steady-state conditions are simulated.

`ntime.steps` `integer`; the number of uniform time steps in a stress period.

`mv.flag` `numeric`; default `NA`, missing value flag for output reference data files.

`auto.flow.reduce`
: `logical`; if `TRUE`, a simulated well will adjust pumping according to supply under bottom-hole conditions. Pumping rates that have been automatically reduced will be written to a model output file ('`.afr`').

`verbose` `logical`; if `TRUE`, additional information is written to the listing file ('`.lst`') and budget file ('`.bud`').

Details

Groundwater flow in the WRV aquifer system is simulated using the MODFLOW-USG groundwater-flow model. This numerical model was chosen for its ability to solve complex unconfined groundwater flow simulations. The solver implemented in MODFLOW-USG incorporates the Newton-Raphson formulation for improving solution convergence and avoiding problems with the drying and rewetting of cells (Niswonger and others, 2011). A structured finite-difference grid is implemented in the model to (1) simplify discretization, (2) keep formats and structures for the MODFLOW-USG packages identical to those of MODFLOW-2005, and (3) allow any MODFLOW post-processor to be used to analyze the results of the MODFLOW-USG simulation (such as Model Viewer).

Model input files are written to `dir.run` and include the following MODFLOW Package files: Name ('`.nam`'), Basic ('`.ba6`'), Discretization ('`.dis`'), Layer-Property Flow ('`.lpf`'), Drain ('`.drn`'), River ('`.riv`'), Well ('`.wel`'), Sparse Matrix Solver ('`.sms`'), and Output Control ('`.oc`'). See the users guide (*Description of Model Input and Output*) included with the MODFLOW-USG software for details on input file formats and structures.

Data within the `rech`, `well`, `trib`, and `misc` arguments are combined in the MODFLOW Well Package and identifiable with added `id` values of 1, 2, 3, and 4, respectively.

The Layer-Property Flow file includes options for the calculation of vertical flow in partially dewatered cells. For the WRV model, where there is no indication that perched conditions exist, `CONSTANTCV` and `NOVFC` options are used to create the most stable solution (Panday and others, 2013, p. 15-16). Options for the Sparse Matrix Solver were set for unconfined simulations by implementing an upstream-weighting scheme with Newton-Raphson linearization, Delta-Bar-Delta underrelaxation, and the χMD solver of Ibaraki (2005).

The raster stack `rs.model` includes the following layers:

lay1.top is the elevation at the top of model layer 1 (land surface), in meters above the NAVD 88.

lay1.bot is the elevation at the bottom of model layer 1, in meters above the NAVD 88.

lay2.bot is the elevation at the bottom of model layer 2.

lay3.bot is the elevation at the bottom of model layer 3.

lay1.strt is the initial (starting) hydraulic head in model layer 1, in meters above the NAVD 88.

lay2.strt is the initial hydraulic head in model layer 2.

lay3.strt is the initial hydraulic head in model layer 3.

lay1.zones is the hydrogeologic zones in model layer 1 where values `= 1` is unconfined alluvium, `= 2` is basalt, `= 3` is clay, and `= 4` is confined alluvium.

lay2.zones is the hydrogeologic zones in model layer 2.

lay3.zones is the hydrogeologic zones in model layer 3.

lay1.hk is the horizontal hydraulic conductivity in model layer 1, in meters per day.

lay2.hk is the horizontal hydraulic conductivity in model layer 2.

lay3.hk is the horizontal hydraulic conductivity in model layer 3.

Value

None. Used for the side-effect of files written to disk.

Author(s)

J.C. Fisher, U.S. Geological Survey, Idaho Water Science Center

References

Ibaraki, M., 2005, χMD User's guide-An efficient sparse matrix solver library, version 1.30: Columbus, Ohio State University School of Earth Sciences.

Niswonger, R.G., Panday, Sorab, and Ibaraki, Motomu, 2011, MODFLOW-NWT, A Newton formulation for MODFLOW-2005: U.S. Geological Survey Techniques and Methods 6-A37, 44 p., available at http://pubs.usgs.gov/tm/tm6a37/.

Panday, Sorab, Langevin, C.D., Niswonger, R.G., Ibaraki, Motomu, and Hughes, J.D., 2013, MODFLOW-USG version 1: An unstructured grid version of MODFLOW for simulating groundwater flow and tightly coupled processes using a control volume finite-difference formulation: U.S. Geological Survey Techniques and Methods, book 6, chap. A45, 66 p., available at http://pubs.usgs.gov/tm/06/a45/.

Examples

```
## Not run: # see uncalibrated-model vignette
```

`zone.properties` *Hydraulic Properties of Hydrogeologic Zones*

Description

Estimates of the hydraulic properties for each hydrogeologic zone.

Usage

```
zone.properties
```

Format

A `data.frame` object with the following variables:

ID is a numeric identifier for the hydrogeologic zone.

name is the name of the hydrogeologic zone.

vani is the vertical anisotropy, a dimensionless quantity.

sc is the storage coefficient, a dimensionless quantity.

sy is the specific yield, a dimensionless quantity.

hk is the horizontal hydraulic conductivity in meters per day.

ss is the specific storage, in inverse meter.

Source

Bartolino, J.R., and Adkins, C.B., 2012, Hydrogeologic framework of the Wood River Valley aquifer system, south-central Idaho: U.S. Geological Survey Scientific Investigations Report 2012-5053, 46 p., available at http://pubs.usgs.gov/sir/2012/5053/.

Examples

```
str(zone.properties)
```