

Processing Instructions for the Groundwater Flow Model of the Wood River Valley, Idaho

By Jason C. Fisher

Contents

Introduction	3
Software	3
Hydrogeologic Framework	4
Space-Time Model Grid	4
Model Grid Conceptualization	4
Spatial Discretization	15
Temporal Discretization	16
Hydraulic Properties	16
Hydrologic Boundaries	21
Groundwater Fluxes from the Tributary Canyons and Upper Big Wood River Valley	21
Groundwater Flow Beneath Silver Creek and Stanton Crossing	23
Stream-Aquifer Flow Exchange in the Big Wood River and Silver Creek	25
Starting Hydraulic Head Distribution	26
Aerial Recharge and Well Pumping	26
Data Archive	26
Model Run	29
Simulated Hydraulic Heads	30
Reproducibility	33

Figures

1	Land surface topography and extent of the aquifer system	5
2	Thickness of Quaternary sediment in the aquifer system	6
3	Extent of basalt in the aquifer system	7
4	Thickness of the aquifer system	9
5	Extent of aquitard in the aquifer system	10
6	Adjustment to bedrock elevations to account for vertically disconnected cells	11
7	Location of specified flow cells in the aquifer system	13
8	Schematic cross-section representation of hydrogeologic units and three-layer model grid. . .	14
9	Hydrogeologic zones in model layer 1	18

10	Hydrogeologic zones in model layer 2	19
11	Hydrogeologic zones in model layer 3	20
12	Seasonal flow rate in the tributary canyons and upper Big Wood River valley	22
13	Location of drain cells in model layer 1	24
14	Stream-reach types in the Big Wood River and Silver Creek	27
15	Steady-state aerial recharge in the aquifer system	28
16	Simulated hydraulic head in model layer 1	31
17	Saturated and partially-saturated cells in model layer 1	32

Tables

1	Summary description of the model grid attributes.	16
2	Hydraulic properties allocated to each hydrogeologic zone in the model.	17
3	Tributary canyons and upper Big Wood River valley specified flows.	23
4	Drain conductance and elevation threshold for drain cell boundaries.	23
5	Description of stream reaches in the Big Wood River and Silver Creek.	25
6	Volumetric budget for entire model at end of time step 1 stress period 1.	30

Conversion Factors and Datums

Conversion Factors

Multiply	By	To obtain
Length		
meter (m)	3.28084	foot (ft)
meter (m)	1.09361	yard (yd)
kilometer (km)	0.621371	mile (mi)
Area		
square meter (m ²)	0.0002471	acre
square meter (m ²)	10.7639	square foot (ft ²)
square kilometer (km ²)	247.104	acre
square kilometer (km ²)	0.386102	square mile (mi ²)
Volume		
cubic meter (m ³)	0.0008107	acre foot (acre-ft)
cubic meter (m ³)	35.3147	cubic foot (ft ³)
Volume per unit time		
cubic meter per day (m ³ /d)	0.296107	acre-foot per year (acre-ft/yr)

Datums

Vertical coordinate information is referenced to the North American Vertical Datum of 1988 ([NAVD 88](#)).

Elevation, as used in this document, refers to distance above vertical datum.

Horizontal coordinate information is referenced to the North American Datum of 1983 ([NAD 83](#)).

Maps are based on the Idaho Transverse Mercator projection ([IDTM](#)).

Introduction

The **wrv** package is a pre- and post-processing program for the numerical groundwater flow model of the Wood River Valley (WRV) aquifer system, south-central Idaho. This document (also known as a *package vignette*) explains steps taken to process the model; its contents should be viewed as provisional until model and report completion in 2015 (Fisher *et al.*, 2015). It is assumed that the readers of this vignette are familiar with the R-programming language and have read the help pages for functions and data sets in the **wrv** package.

This package vignette integrates R code in a \LaTeX document. The code is run when the vignette is built, and all data analysis output (figures, tables, etc.) is created on the fly and inserted into the final document. Small chunks of stylized code are shown throughout the vignette and are intended to be used interactively. Commands that comprise these *code chunks* are essential for processing the groundwater flow model, and describe approaches to model development and analysis decisions. Note that embedded code can be extracted from this vignette, which allows for truly reproducible research; see ‘[Reproducibility](#)’ section for details.

Software

Software items needed to run the processing instructions for the groundwater flow model include R and MODFLOW-USG. R is a language and environment for statistical computing and graphics (R Core Team, 2014). If R (version ≥ 3.1) is not already installed on your computer, download and install the latest binary distribution from the Comprehensive R Archive Network (CRAN). Extend the capabilities of R by installing user-contributed packages available online. Start an R session and type the following commands in your R Console window:

```
repos <- c("<SPECIFIED WHEN REPORT IS PUBLISHED>", "http://cran.us.r-project.org")
install.packages("wrv", repos = repos, dependencies = TRUE, type = "both")
```

MODFLOW-USG is a computer program for simulating three-dimensional, steady-state and transient groundwater flow using a control volume finite-difference formulation (Panday *et al.*, 2013). If MODFLOW-USG (version ≥ 1.2) is not already installed on your computer, download and decompress the latest file archive. The archive contains an executable file for Windows. Note that users of a [Unix-like](#) operating system will need to compile MODFLOW-USG. Specify the full path name to the executable file using the following R command (change path as needed):

```
file.exe <- "C:/WRDAPP/mfusg.1_2/bin/mfusg_x64.exe" # path specified with forward slashes
```

To gain access to the contents of the **wrv** package in R, use the following command:

```
library(wrv)
```

To open help pages for functions and data sets in the **wrv** package, type:

```
help(package = "wrv")
```

Hydrogeologic Framework

The WRV aquifer system is composed of a single unconfined aquifer that underlies the entire valley, an underlying confined aquifer that is present only in the southern part of the valley, and a confining unit separating the two aquifers (Bartolino and Adkins, 2012, pg. 3). The land-surface topography and spatial extent of the aquifer system are shown in figure 1. The aquifer system primarily consists of Quaternary deposits that can be divided into three hydrogeologic units: a coarse-grained sand and gravel unit (alluvium unit), a fine-grained silt and clay unit (clay unit), and a basalt unit (Bartolino and Adkins, 2012, pg. 3).

Space-Time Model Grid

Model Grid Conceptualization

The creation of the model grid is the first step in developing the groundwater flow model, because all model inputs including hydraulic properties and boundary conditions are assigned to the model cells. The three-dimensional model grid is rectilinear (square cells) horizontally, distorted vertically, and not rotated. The decision to use a structured grid, rather than exploit the unstructured grid capabilities of MODFLOW-USG, was based on a desire to avoid the added complexities of designing pre- and post-processing algorithms for an unstructured grid. A preliminary sensitivity analysis to changes in grid resolution indicated that a 100 m (330 feet [ft]) resolution provides the optimal tradeoff between the inherent spatial variability of the data and the ability to get continuous grid coverage in the narrow and steep tributary canyons of the WRV.

Solid-boundary representations of the land surface and pre-Quaternary bedrock surface and top of Quaternary basalt (alluvium bottom) are used to generate the basic structure of the model grid.

```
rs.data <- stack() # initialize a raster stack, a collection of raster layers
rs.data <- stack(rs.data, land.surface) # add raster layer to raster stack
rs.data <- stack(rs.data, alluvium.bottom)
```

Thickness of the Quaternary sediment is calculated by subtracting alluvium-bottom elevations from land surface elevations (fig. 2). Cells which are too thin can lead to numerical instability in the model; therefore, cells less than 1 m (3.3 ft) thick are made inactive.

```
r <- rs.data[["land.surface"]] - rs.data[["alluvium.bottom"]]
min.thickness <- 1 # in meters
is.too.thin <- r[] < min.thickness
rs.data[["alluvium.bottom"]][is.too.thin] <- NA
r[is.too.thin] <- NA
names(r) <- "alluvium.thickness"
rs.data <- stack(rs.data, r)
```

The total number of cells removed because they were too thin is 2.

The estimated aerial extent of basalt in the WRV aquifer system is shown in figure 3.

```
r <- rasterize(basalt.extent, rs.data, getCover = TRUE, silent = TRUE)
r[r > 0] <- 1
r[r < 1] <- NA
r <- ratify(r) # add a raster attribute table
levels(r) <- cbind(levels(r)[[1]], att = "basalt")
names(r) <- "basalt.extent"
rs.data <- stack(rs.data, r)
```

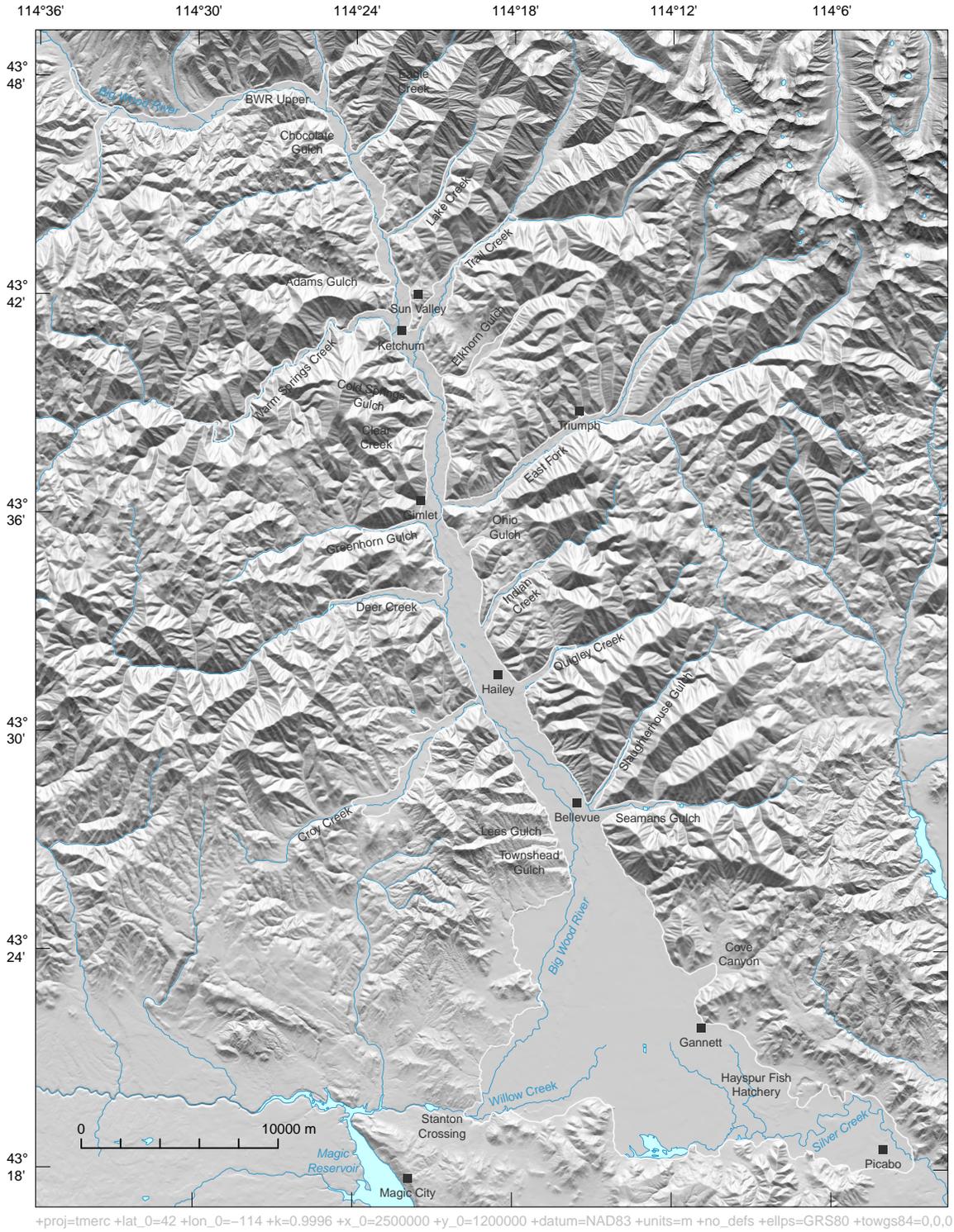


Figure 1: Land surface topography and extent of the aquifer system.

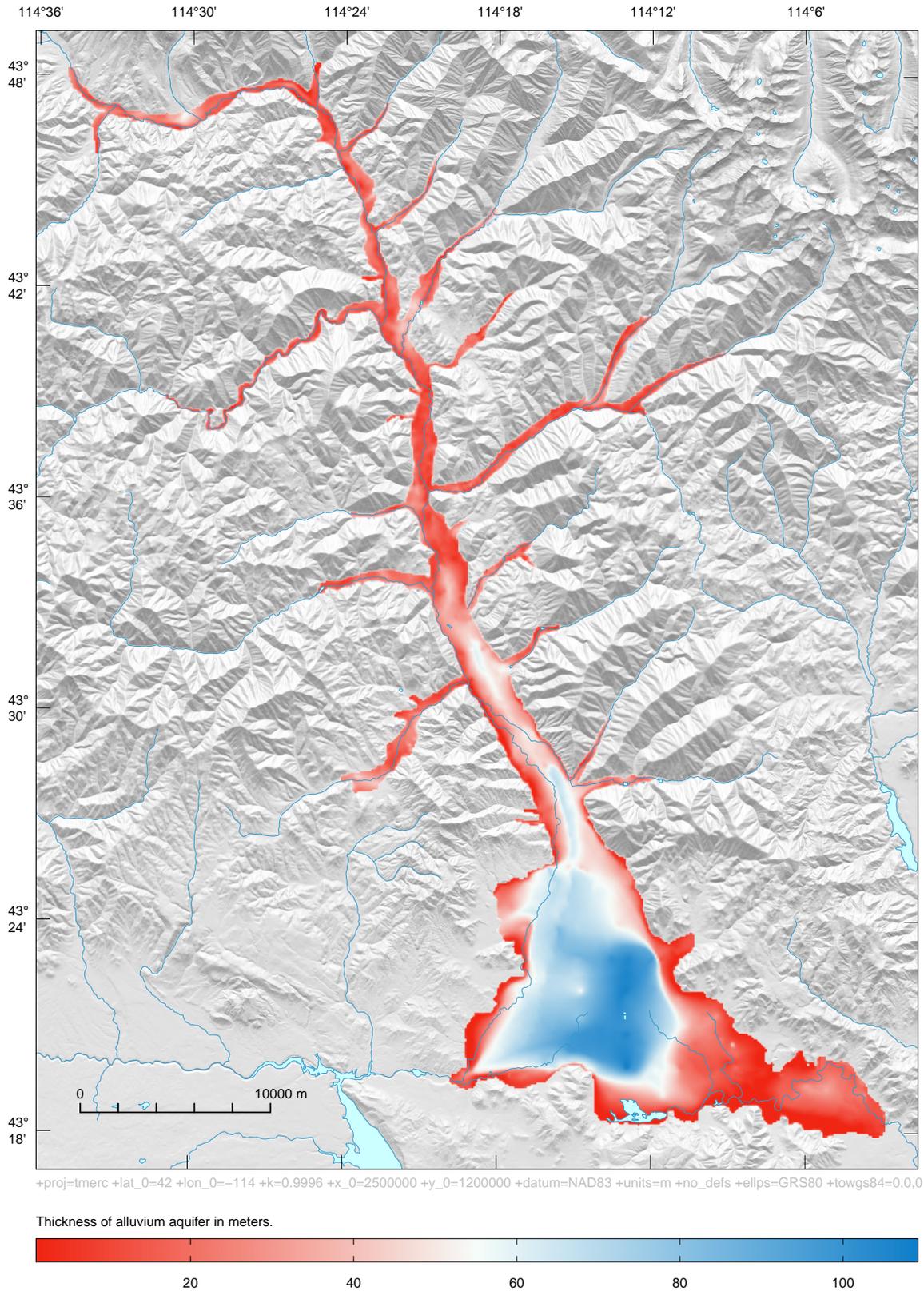
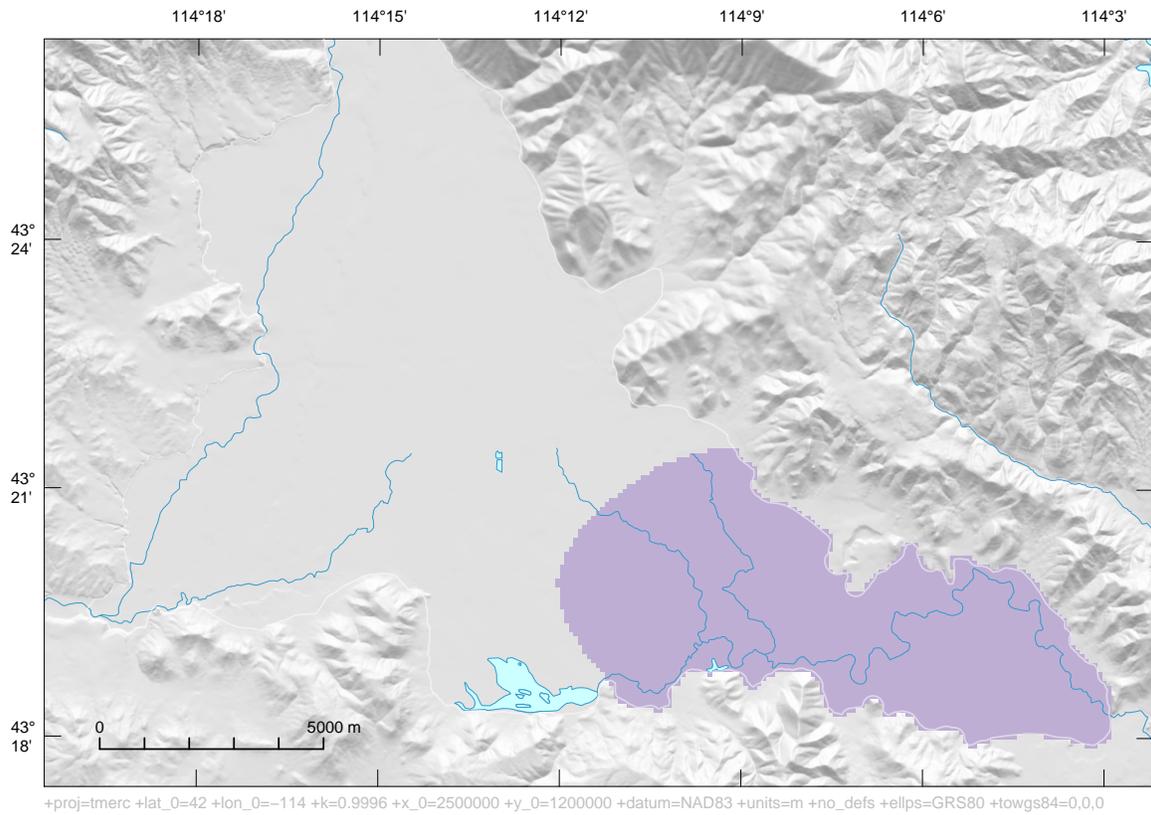


Figure 2: Thickness of Quaternary sediment in the aquifer system.



basalt

Figure 3: Extent of basalt in the aquifer system.

Basalt underlies the Quaternary sediment; however, very little data is available to describe the unit thickness of basalt. The few wells that penetrate the basalt unit are located at the Hayspur Fish Hatchery (fig. 1) and describe consistent unit thicknesses among wells of about 15 m (49 ft) for alluvium and 37 m (121 ft) for basalt. Summing these unit thicknesses gives the estimated depth, measured as the distance below land surface, to the bottom of the basalt unit at 52 m (170 ft). Note that this depth is assumed constant throughout the extent of the basalt unit. Transmissive materials that may be present beneath the basalt unit are neglected due to insufficient data to describe these materials. The bedrock surface elevation for the aquifer system is then calculated by integrating units.

```
depth.to.basalt.bottom <- 52 # in meters
r <- rs.data[["land.surface"]] - depth.to.basalt.bottom
r[r > rs.data[["alluvium.bottom"]] | is.na(rs.data[["basalt.extent"]])] <- NA
basalt.bottom <- r
r <- rs.data[["alluvium.bottom"]]
is.basalt.cell <- !is.na(basalt.bottom)
r[is.basalt.cell] <- basalt.bottom[is.basalt.cell]
names(r) <- "bedrock"
rs.data <- stack(rs.data, r)
```

Subtracting bedrock surface elevations from land surface elevations gives the thickness of the WRV aquifer system (fig. 4).

```
r <- rs.data[["land.surface"]] - rs.data[["bedrock"]]
names(r) <- "aquifer.thickness"
rs.data <- stack(rs.data, r)
```

The aquitard separating the unconfined aquifer from the underlying confined aquifer is represented with the clay unit. The estimated extent of the aquitard in the WRV aquifer system is shown in figure 5.

```
r <- rasterize(gUnaryUnion(aquitard.extent), rs.data, getCover = TRUE, silent = TRUE)
r[r > 0] <- 1
r[r < 1] <- NA
r <- ratify(r)
levels(r) <- cbind(levels(r)[[1]], att = "clay")
names(r) <- "aquitard.extent"
rs.data <- stack(rs.data, r)
```

Well driller reports and geophysical surveys describe the clay unit as about 5 m (16 ft) thick, and generally lying at a depth of about 30 m (98 ft).

```
aquitard.thickness <- 5 # in meters
depth.to.aquitard.top <- 30 # in meters
r <- rs.data[["land.surface"]] - depth.to.aquitard.top
r[r < rs.data[["alluvium.bottom"]] | is.na(rs.data[["aquitard.extent"]])] <- NA
names(r) <- "aquitard.top"
rs.data <- stack(rs.data, r)
```

Vertical connectivity among cells is ensured by setting a minimum vertical overlap between adjacent cells. Cells having less than 2 m (6.6 ft) of overlap are adjusted by incrementally dropping the cells bottom elevation until the minimum vertical overlap is attained (fig. 6).

```
min.overlap <- 2 # minimum vertical overlap between adjacent cells, in meters
r <- BumpDisconnectedCells(subset(rs.data, c("land.surface", "bedrock")), min.overlap)
rs.data[["bedrock"]] <- rs.data[["bedrock"]] + r
names(r) <- "disconnect.adjustment"
rs.data <- stack(rs.data, r)
```

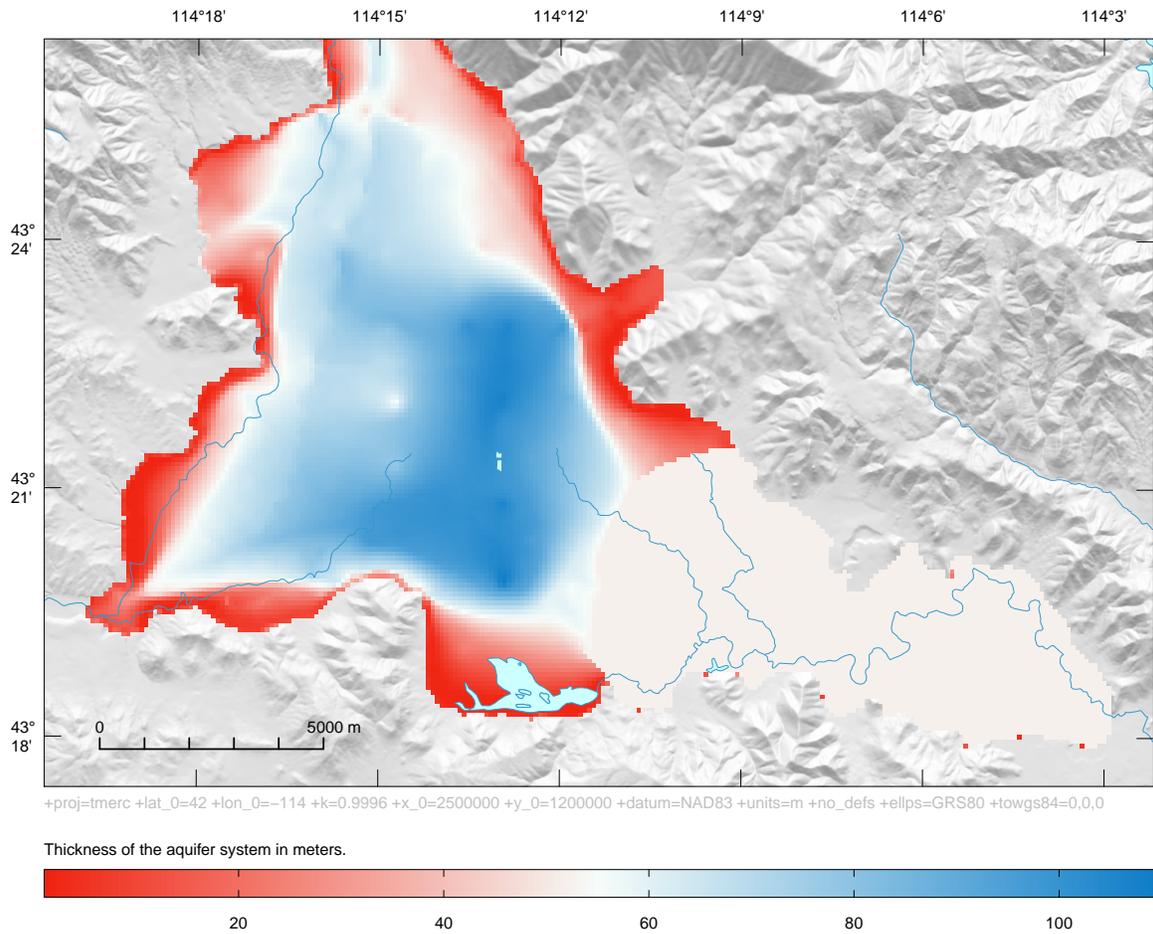
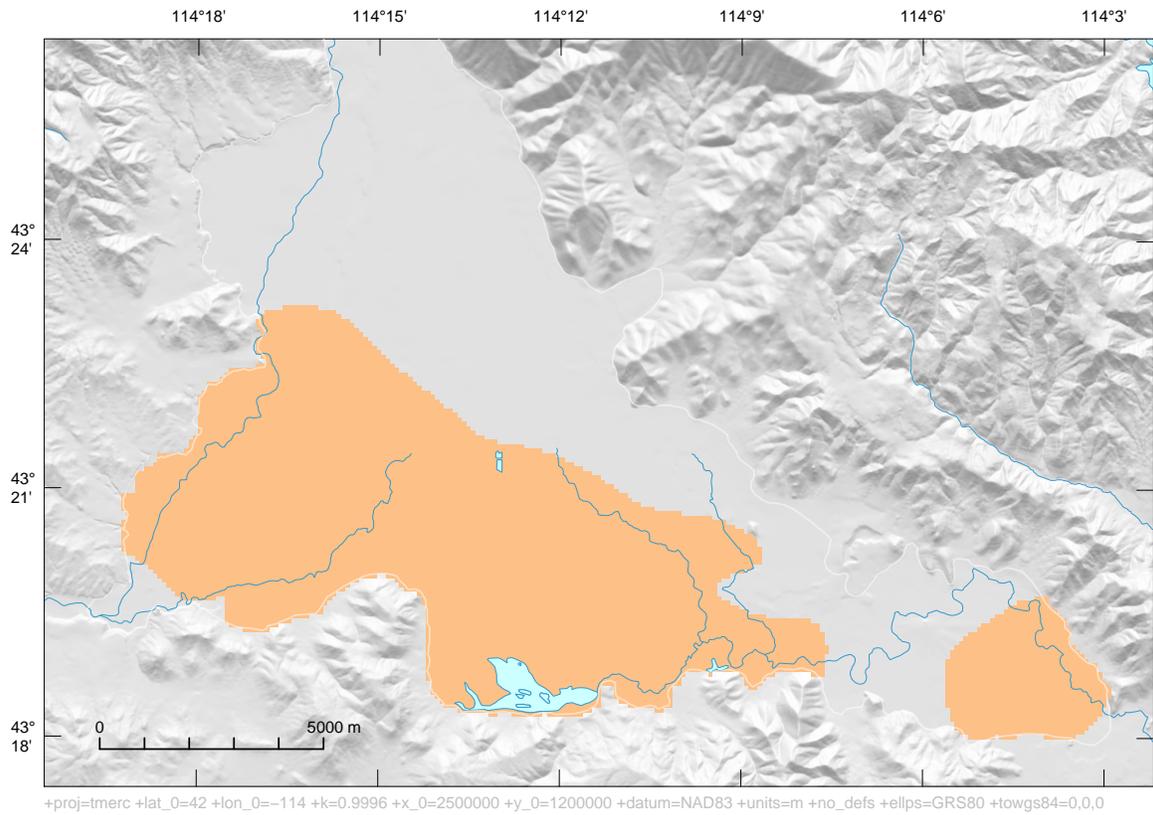


Figure 4: Thickness of the aquifer system.



clay

Figure 5: Extent of aquitard in the aquifer system.

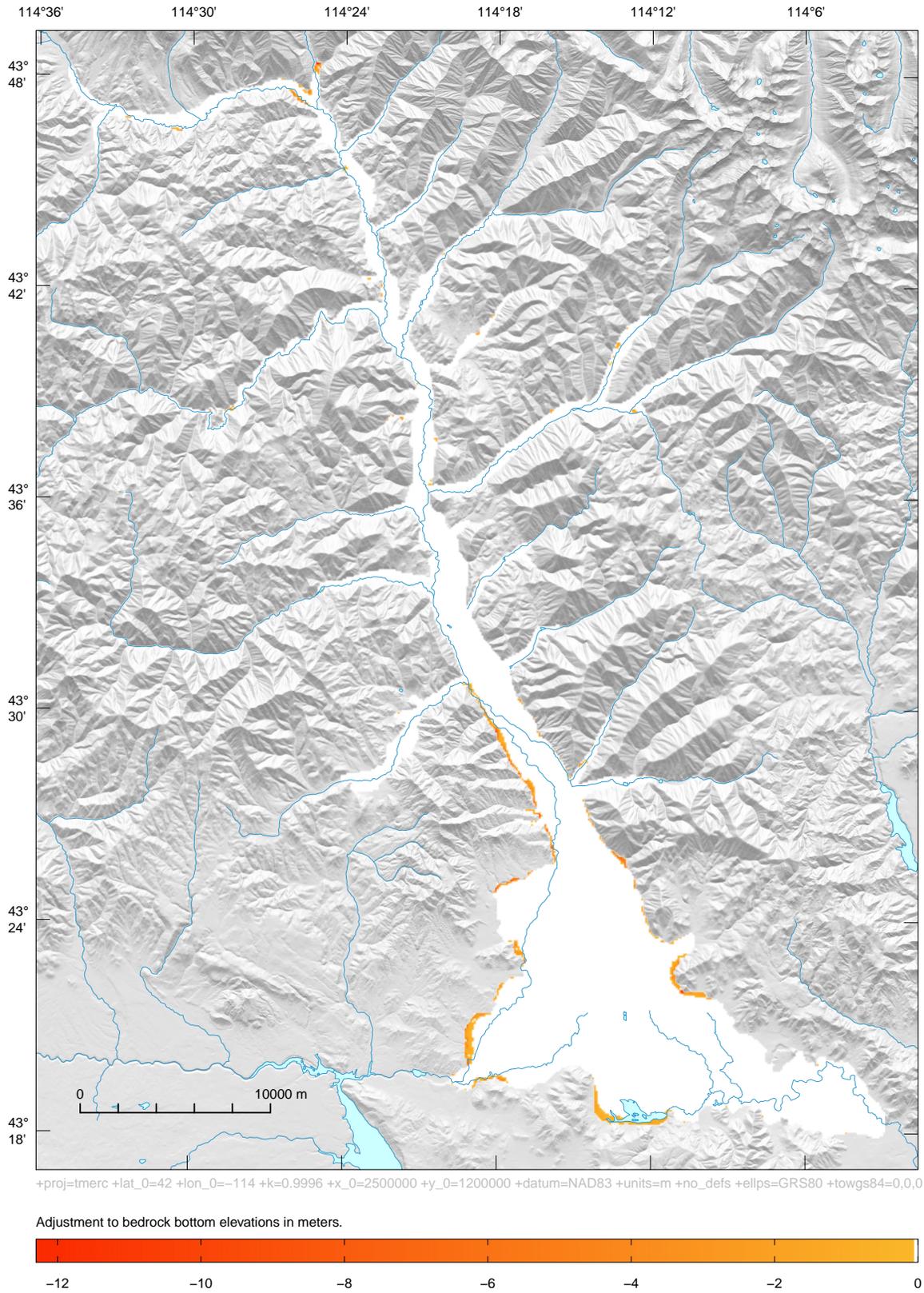


Figure 6: Adjustment to bedrock elevations to account for vertically disconnected cells.

The total number of vertically adjusted cells is 886, with a median and standard deviation of -1.2 m and 1.8 m, respectively.

Groundwater enters the model domain through specified flow cells located in the major tributary canyons and beneath the valley floor at the confluence of the Big Wood River and the North Fork Big Wood River ('BWR Upper' in fig. 1). A sparsity of field observations in the major tributary canyons indicates large uncertainty in the historic flow contribution from each of the tributary canyons. Therefore, specified flow cells are placed in the upper part of the tributary canyons to help contain errors that may propagate into the model from these boundaries. Simulated hydraulic heads in the tributary canyons should be considered less reliable than in the WRV. Maintaining continuous grid coverage in the narrow tributary canyons and leveraging existing observation wells was also taken into consideration when designating specified flow cells. An intended consequence of these boundaries is a reduction in the extent of the modeled aquifer system. Specified flow cells are identified using horizontal polygons with a single polygon allocated to each of the 22 boundaries. Active cells intersecting a polygon line segment are defined as specified flow cells, and cells located within the body of a polygon made inactive (fig. 7).

```
l <- gIntersection(as(specified.flows, "SpatialLinesDataFrame"), aquifer.extent, TRUE)
spec.flow.lines <- SpatialLinesDataFrame(l, data = specified.flows@data, match.ID = FALSE)
r <- rs.data[["alluvium.bottom"]]
is.in.aquifer <- !is.na(r)
is.in.poly <- !is.na(rasterize(specified.flows, rs.data, silent = TRUE))
is.on.line <- !is.na(rasterize(spec.flow.lines, rs.data))
r[is.in.aquifer & is.in.poly] <- 0 # inactive cells
r[is.in.aquifer & !is.in.poly] <- 1 # active cells
r[is.in.aquifer & is.on.line] <- 2 # specified flow cells
cells <- which(r[] == 2)
adj.cells <- adjacent(r, cells, directions = 4)
is.valid <- adj.cells[, 2] %in% which(r[] == 1)
rm.cells <- cells[!(cells %in% unique(adj.cells[is.valid, 1]))]
r[rm.cells] <- 0
r <- ratify(r)
att <- c("inactive cells", "active cells", "specified flow cells")
levels(r) <- cbind(levels(r)[[1]], att = att)
names(r) <- "specified.flows"
rs.data <- stack(rs.data, r)
```

Flow through the low-permeability aquitard that separates the alluvium aquifers may significantly influence groundwater pressure responses, necessitating a multi-layer model. Model layering was designed to allow accurate representation of the aquitard. Figure 8 shows a schematic cross-section representation of the hydrogeologic units and the three-layer model grid. Embedded clay within the basalt unit is assumed to have a negligible effect on groundwater flow. Model cells in layers 2 and 3 become inactive in the vicinity north of Hailey (fig. 1).

The bottom elevation of model layer 1 is calculated by subtracting the depth to the top of the aquitard (30 m) from land surface. Cell values lying beneath the pre-Quaternary bedrock surface and top of Quaternary basalt are replaced with alluvium bottom elevations.

```
rs.model <- stack() # initialize a raster stack for model input
r <- rs.data[["land.surface"]] - depth.to.aquitard.top
is.below <- rs.data[["alluvium.bottom"]] > r
r[is.below] <- rs.data[["alluvium.bottom"]][is.below]
r[(rs.data[["land.surface"]] - r) < min.thickness] <- NA # enforce min. layer thickness
r[rs.data[["specified.flows"]] == 0] <- NA
r <- ExcludeSmallCellChunks(r) # ensure horizontal connectivity among cells
names(r) <- "lay1.bot"
rs.model <- stack(rs.model, r)
```

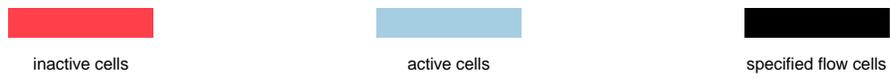
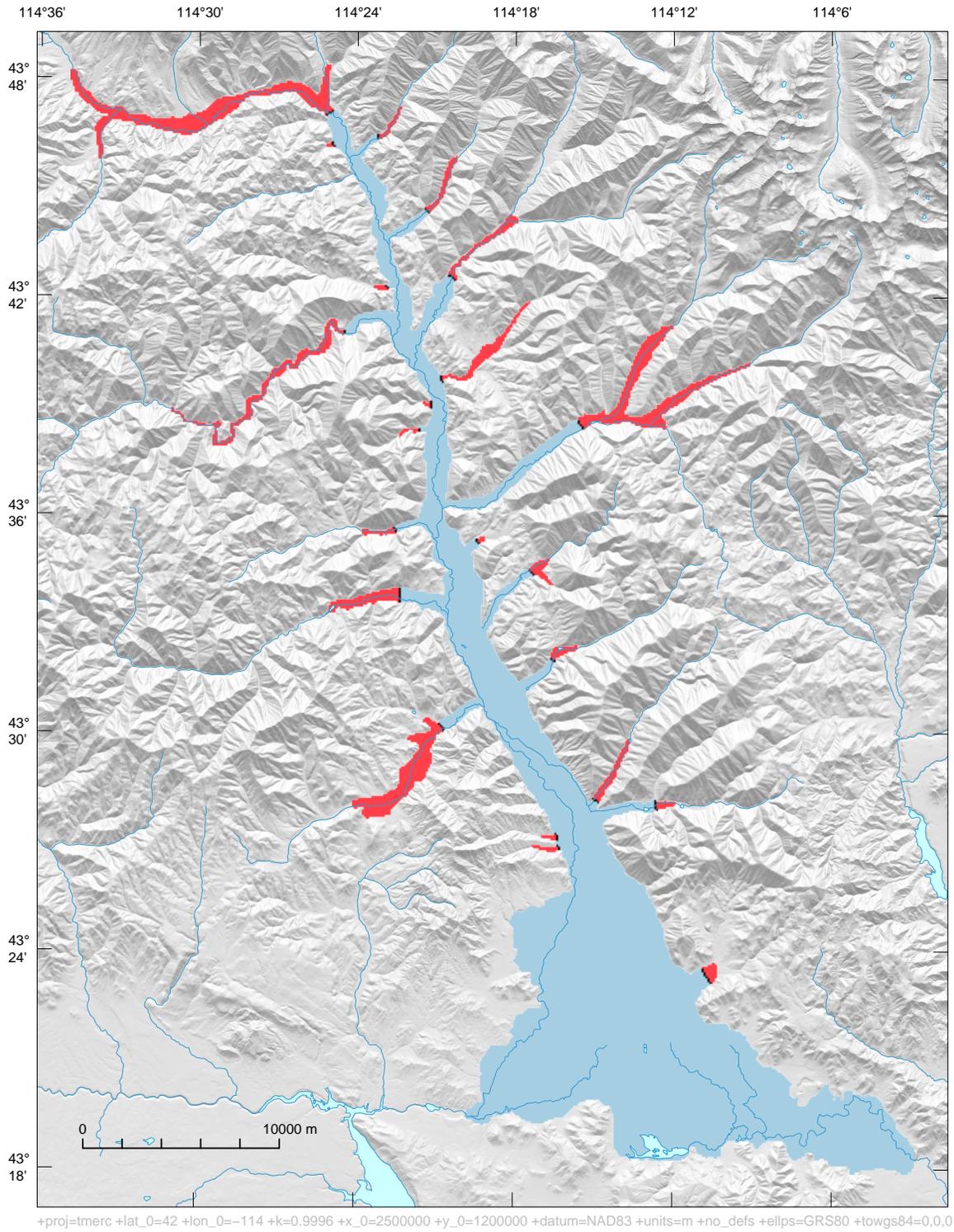


Figure 7: Location of specified flow cells in the aquifer system.

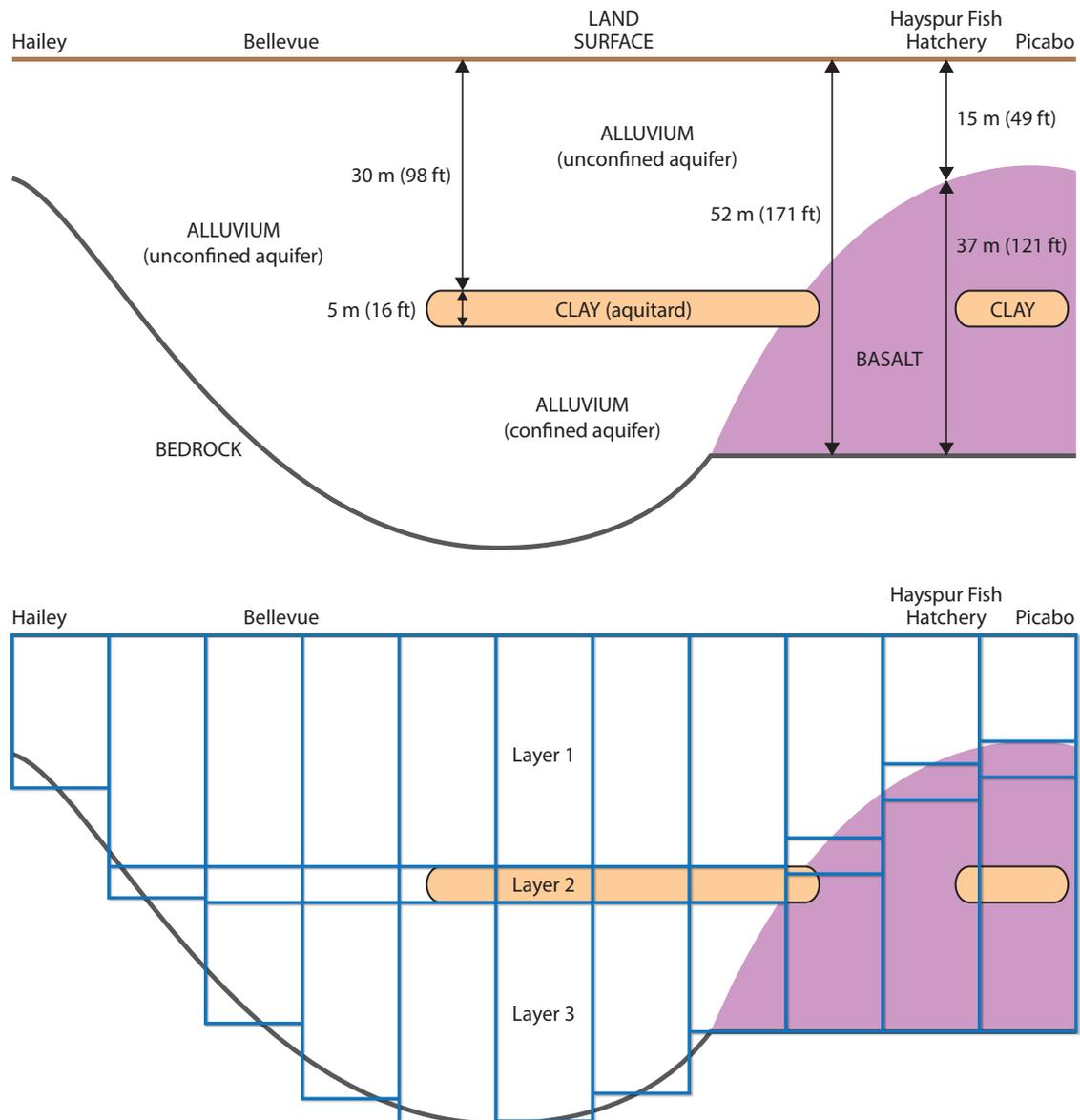


Figure 8: Schematic cross-section representation of hydrogeologic units and three-layer model grid.

Subtracting the aquitard thickness (5 m) from the bottom of model layer 1 gives the bottom elevation of model layer 2. Cell values lying beneath the bedrock surface are replaced with bedrock elevations.

```
r <- rs.model[["lay1.bot"]] - aquitard.thickness
is.below <- rs.data[["bedrock"]] > r
r[is.below] <- rs.data[["bedrock"]][is.below]
r[(rs.model[["lay1.bot"]] - r) < min.thickness] <- NA # enforce minimum thickness
r <- ExcludeSmallCellChunks(r)
names(r) <- "lay2.bot"
rs.model <- stack(rs.model, r)
```

The bottom elevation of model layer 3 is at bedrock.

```
r <- rs.data[["bedrock"]]
r[is.na(rs.model[["lay2.bot"]])] <- NA
```

```
r[(rs.model[["lay2.bot"]] - r) < min.thickness] <- NA # enforce minimum thickness
r <- ExcludeSmallCellChunks(r)
names(r) <- "lay3.bot"
rs.model <- stack(rs.model, r)
```

Bottom elevations of model layer 1 are adjusted to bedrock where the cell value is above bedrock and its vertically adjacent cell is inactive in model layer 2.

```
r <- rs.model[["lay1.bot"]]
is.adjusted <- r > rs.data[["bedrock"]] & is.na(rs.model[["lay2.bot"]])
r[is.adjusted] <- rs.data[["bedrock"]][is.adjusted]
r <- ExcludeSmallCellChunks(r)
rs.model[["lay1.bot"]] <- r
```

The top elevation of model layer 1 is at land surface.

```
r <- rs.data[["land.surface"]]
r[is.na(rs.model[["lay1.bot"]])] <- NA
names(r) <- "lay1.top"
rs.model <- stack(rs.model, r)
```

The thickness of each model layer is calculated by subtracting the layers bottom elevations from its top elevations.

```
r <- rs.data[["land.surface"]] - rs.model[["lay1.bot"]]
names(r) <- "lay1.thickness"
rs.model <- stack(rs.model, r)
r <- rs.model[["lay1.bot"]] - rs.model[["lay2.bot"]]
names(r) <- "lay2.thickness"
rs.model <- stack(rs.model, r)
r <- rs.model[["lay2.bot"]] - rs.model[["lay3.bot"]]
names(r) <- "lay3.thickness"
rs.model <- stack(rs.model, r)
```

Spatial Discretization

Removing outer rows and columns that are all inactive results in the horizontal model grid. A summary of the model grid attributes is shown in table 1.

```
model.extent <- trim(rs.model[["lay1.bot"]])
FUN <- function(i) {
  return(crop(rs.model[[i]], model.extent))
}
rs.model <- stack(lapply(names(rs.model), FUN), quick = TRUE)
```

Active and inactive cells are located in the model grid.

```
r <- rs.model[["lay1.bot"]]
r[] <- as.integer(!is.na(r[]))
names(r) <- "lay1.ibound"
rs.model <- stack(rs.model, r)
r <- rs.model[["lay2.bot"]]
r[] <- as.integer(!is.na(r[]))
names(r) <- "lay2.ibound"
rs.model <- stack(rs.model, r)
r <- rs.model[["lay3.bot"]]
```

Table 1: Summary description of the model grid attributes.

Attribute	Value
Number of rows	542
Number of columns	299
Number of layers	3
Number of active model cells	54,886
Uniform spacing in the easting direction (m)	100
Uniform spacing in the northing direction (m)	100
Easting coordinate of model origin (m)	2,466,200
Northing coordinate of model origin (m)	1,344,139

```
r[] <- as.integer(!is.na(r[]))
names(r) <- "lay3.ibound"
rs.model <- stack(rs.model, r)
```

Temporal Discretization

The interval of discretization for time is the *time step*. A uniform time step of 1 day is specified for the groundwater flow model. Time steps are grouped into *stress periods*, where time dependent input data can be changed every stress period (Harbaugh *et al.*, 2000, pg. 8). Individual stress periods in a single simulation can either be steady-state or transient. The transient groundwater flow model is assumed to start from a period when the aquifer system was in steady-state equilibrium. The first stress period is specified as steady state and all subsequent stress periods as transient; that is, the initial value of transient analysis is the first stress period solution. Steady-state flow was simulated to represent conditions in 1995 with average recharge from April 2004 through March 2005, a period that included a relatively large number of water-level measurements and conditions similar to 1995. The transient stress periods simulate groundwater flow between 1995 and 2010, a 16 year duration. A uniform transient stress period of 1 month is specified for the model.

```
ss.interval <- as.Date(c("2004-04-01", "2005-03-31")) # steady state
tr.interval <- as.Date(c("1995-01-01", "2010-12-31")) # transient
ss.stress.periods <- seq(ss.interval[1], ss.interval[2], "month")
tr.stress.periods <- seq(tr.interval[1], tr.interval[2], "month")
```

Hydraulic Properties

Prior to model calibration, the distribution of hydraulic properties (such as hydraulic conductivity) is based on hydrogeologic zones, groups of model cells with uniform hydraulic properties that compose part or all of a hydrogeologic unit. The model consists of four hydrogeologic zones described as follows:

Zone 1: composed of the alluvium unit under unconfined conditions and located in all three model layers;

Zone 2: composed of the basalt and clay units and located in model layers 2 and 3;

Zone 3: composed of the clay unit and located in model layer 2; and

Zone 4: composed of the alluvium unit under confined conditions and located in model layer 3.

Horizontal hydraulic conductivity values for the hydrogeologic zones are based on previous estimates by Bartolino and Adkins (2012, table 2, pg. 25-26). Parameter values should be viewed as preliminary and subject to change during model calibration (table 2).

```
att.table <- as.data.frame(list(ID = 1:4, name = paste("Zone", 1:4),
                               hk = c(21, 15.2, 8.6e-07, 12.8), vni = 50,
                               ss = c(2.5, 0.2, 1.3, 2.5), sy = c(0.3, 0.2, 0.1, 0.3)))
```

Table 2: Hydraulic properties allocated to each hydrogeologic zone in the model.

Name	Horizontal hydraulic conductivity (m/d)	Vertical anisotropy	Specific storage (1/m)	Specific yield
Zone 1	2.1e+01	50	2.5	0.3
Zone 2	1.5e+01	50	0.2	0.2
Zone 3	8.6e-07	50	1.3	0.1
Zone 4	1.3e+01	50	2.5	0.3

The delineation of hydrogeologic zones in model layer 1 is shown in figure 9.

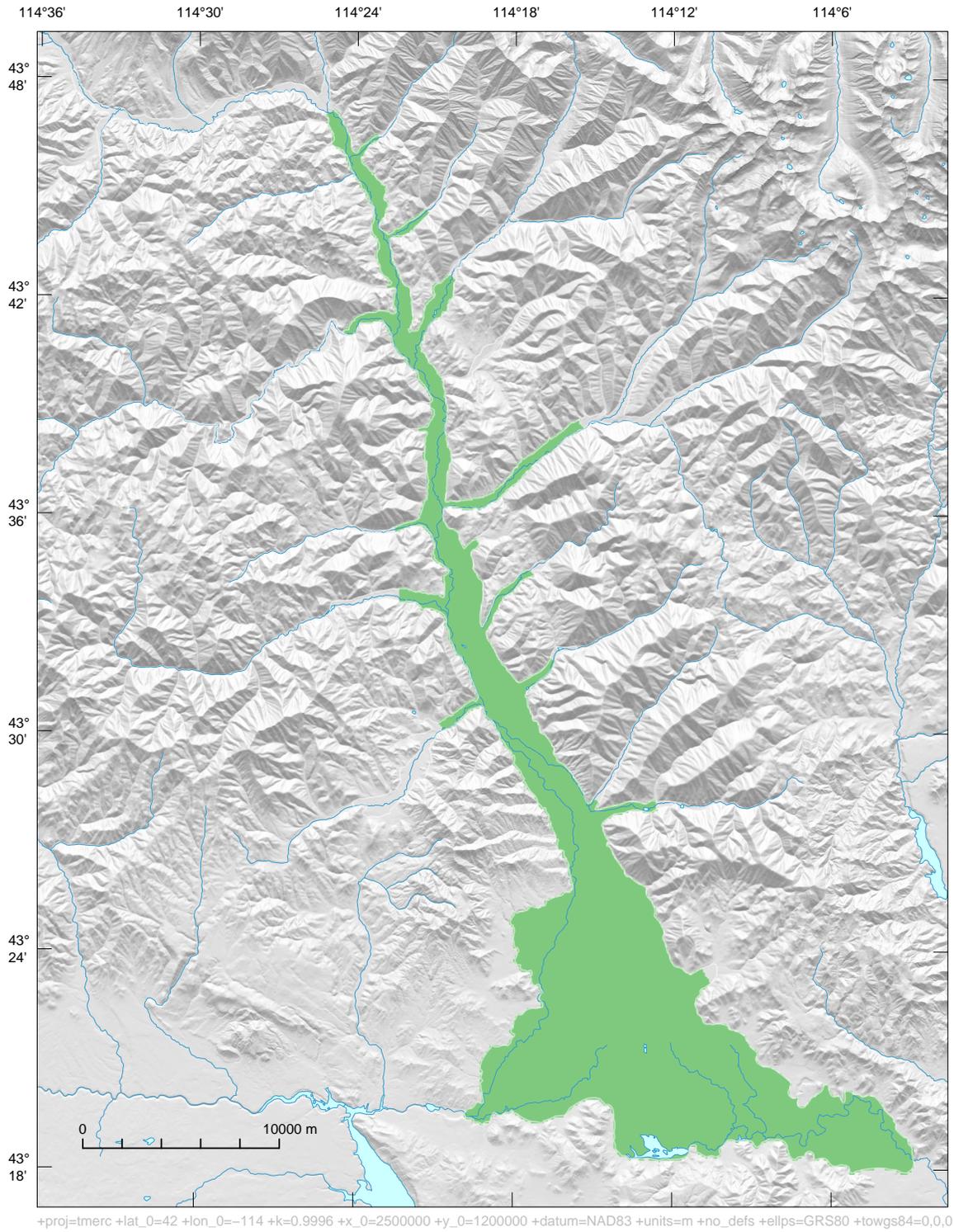
```
r <- rs.model[["lay1.bot"]]
r[!is.na(r)] <- 1
r <- ratify(r)
levels(r) <- cbind(merge(levels(r)[[1]], att.table))
names(r) <- "lay1.zones"
rs.model <- stack(rs.model, r)
```

The delineation of hydrogeologic zones in model layer 2 is shown in figure 10.

```
r <- rs.model[["lay2.bot"]]
r[!is.na(r)] <- 1
r[!is.na(r) & !is.na(crop(rs.data[["aquitard.extent"]], model.extent))] <- 3
r[rs.model[["lay2.bot"]] < crop(rs.data[["alluvium.bottom"]], model.extent)] <- 2
r <- ratify(r)
levels(r) <- cbind(merge(levels(r)[[1]], att.table))
names(r) <- "lay2.zones"
rs.model <- stack(rs.model, r)
```

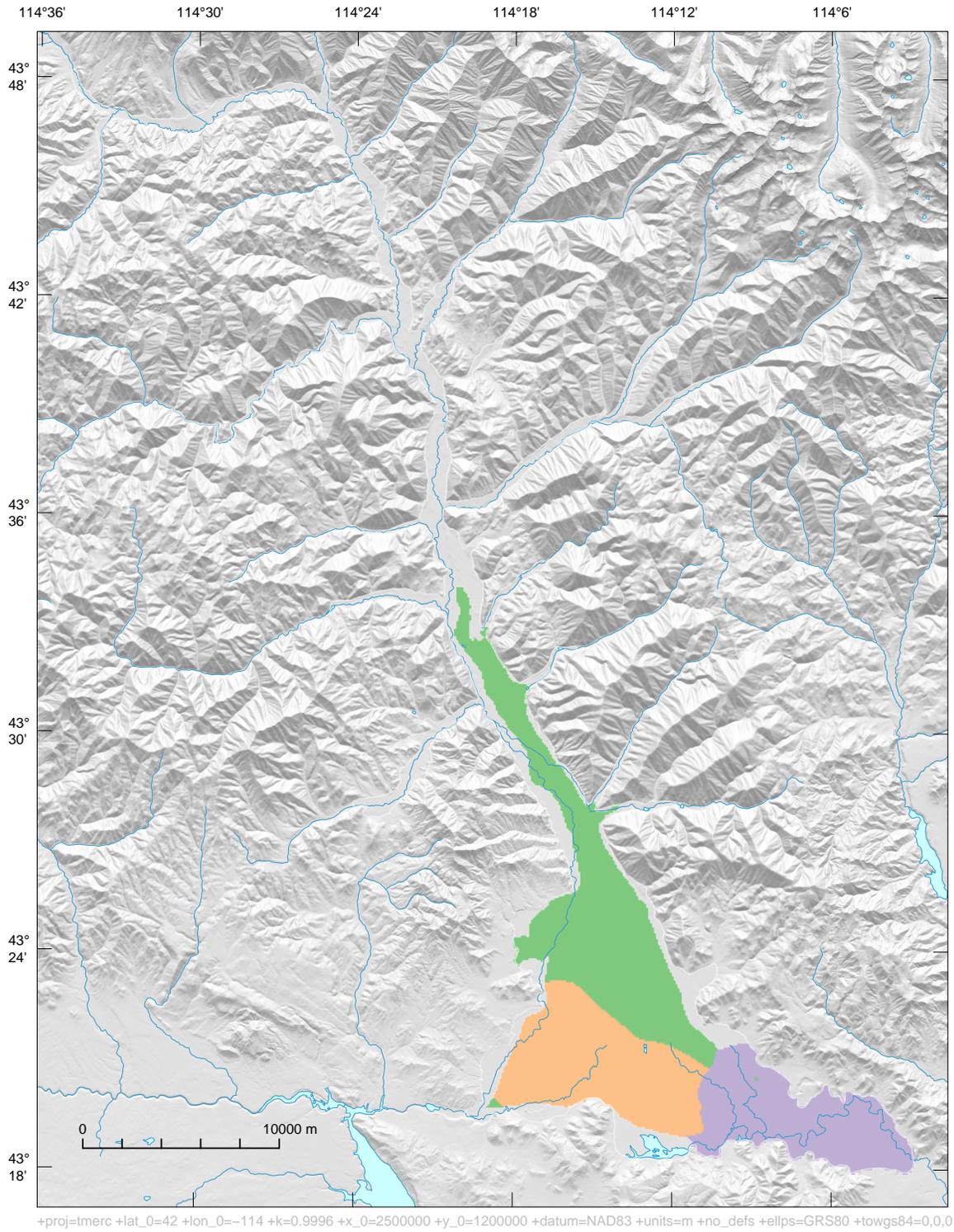
The delineation of hydrogeologic zones in model layer 3 is shown in figure 11.

```
r <- rs.model[["lay3.bot"]]
r[!is.na(r)] <- 1
r[!is.na(r) & rs.model[["lay2.zones"]] == 3] <- 4
r[rs.model[["lay3.bot"]] < crop(rs.data[["alluvium.bottom"]], model.extent)] <- 2
r <- ratify(r)
levels(r) <- cbind(merge(levels(r)[[1]], att.table))
names(r) <- "lay3.zones"
rs.model <- stack(rs.model, r)
```



Zone 1

Figure 9: Hydrogeologic zones in model layer 1.



Zone 1

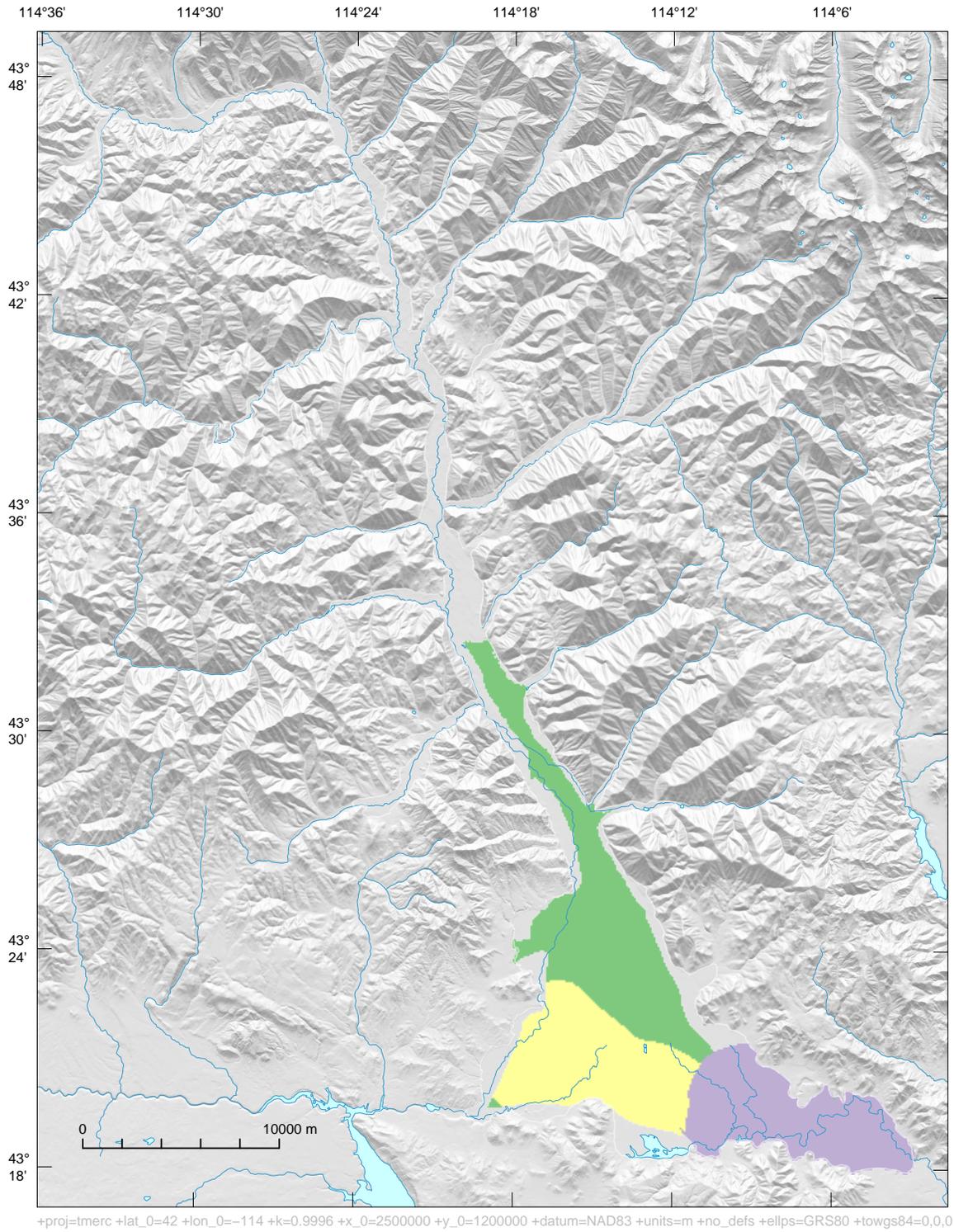


Zone 2



Zone 3

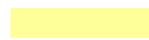
Figure 10: Hydrogeologic zones in model layer 2.



Zone 1



Zone 2



Zone 4

Figure 11: Hydrogeologic zones in model layer 3.

Hydrologic Boundaries

Groundwater Fluxes from the Tributary Canyons and Upper Big Wood River Valley

Groundwater entering the aquifer system through the major tributary canyons and upper Big Wood River Valley is simulated using the MODFLOW Flow and Head Boundary Package (Leake and Michael, 1997), a specified flow boundary condition. Figure 7 shows the location of these boundaries in the model. The average volumetric flux for each boundary is calculated using Darcy's law and shown in table 3. A scaling index is used to represent temporal variation in the volumetric fluxes (fig. 12).

```
mult <- GetSeasonalMultiplier(hailey.discharge, 2, 273.932, tr.interval)
mult <- data.frame(Date = tr.stress.periods, multiplier = rep(mult$multiplier, each = 3))
flow <- vapply(tributaries$Flow, function(i) mult$multiplier * i, rep(0, nrow(mult)))
colnames(flow) <- tributaries$ID
rownames(flow) <- format(mult$Date, format = "%Y%m")
```

The volumetric fluxes are combined with the specified flow cells.

```
r <- rasterize(spec.flow.lines, rs.model)
r[crop(rs.data[["specified.flows"]], model.extent) != 2] <- NA
rat <- levels(r)[[1]]
rat <- merge(rat, freq(r), by.x="ID", by.y="value", all.x=TRUE)
d <- as.data.frame(t(flow[, match(tributaries$Name, rat$Name)]))
rat <- cbind(ID = rat$ID, COUNT = rat$count, d)
rat$ss <- apply(d[, format(ss.stress.periods, "%Y%m")], 1, mean)
cells <- lapply(rat$ID, function(i) which(r[] == i))
names(cells) <- rownames(rat)
levels(r) <- rat
names(r) <- "tributaries"
rs.model <- stack(rs.model, r)
```

For each boundary, the volumetric flux is uniformly distributed among its specified flow cells.

```
att.table <- as.data.frame(matrix(NA, nrow = sum(rat$COUNT), ncol = ncol(rat)))
colnames(att.table) <- colnames(rat)
att.table$ID <- seq_len(nrow(att.table))
att.table$COUNT <- NULL
r <- raster(rs.model)
r[unlist(cells)] <- seq_len(length(unlist(cells)))
count <- rep(rat$COUNT, times = rat$COUNT)
for (i in colnames(att.table)[-1]) {
  att.table[, i] <- rep(rat[, i], times = rat$COUNT) / count
}
r <- ratify(r)
levels(r) <- cbind(merge(levels(r)[[1]], att.table))
names(r) <- "fhh"
rs.model <- stack(rs.model, r)
```

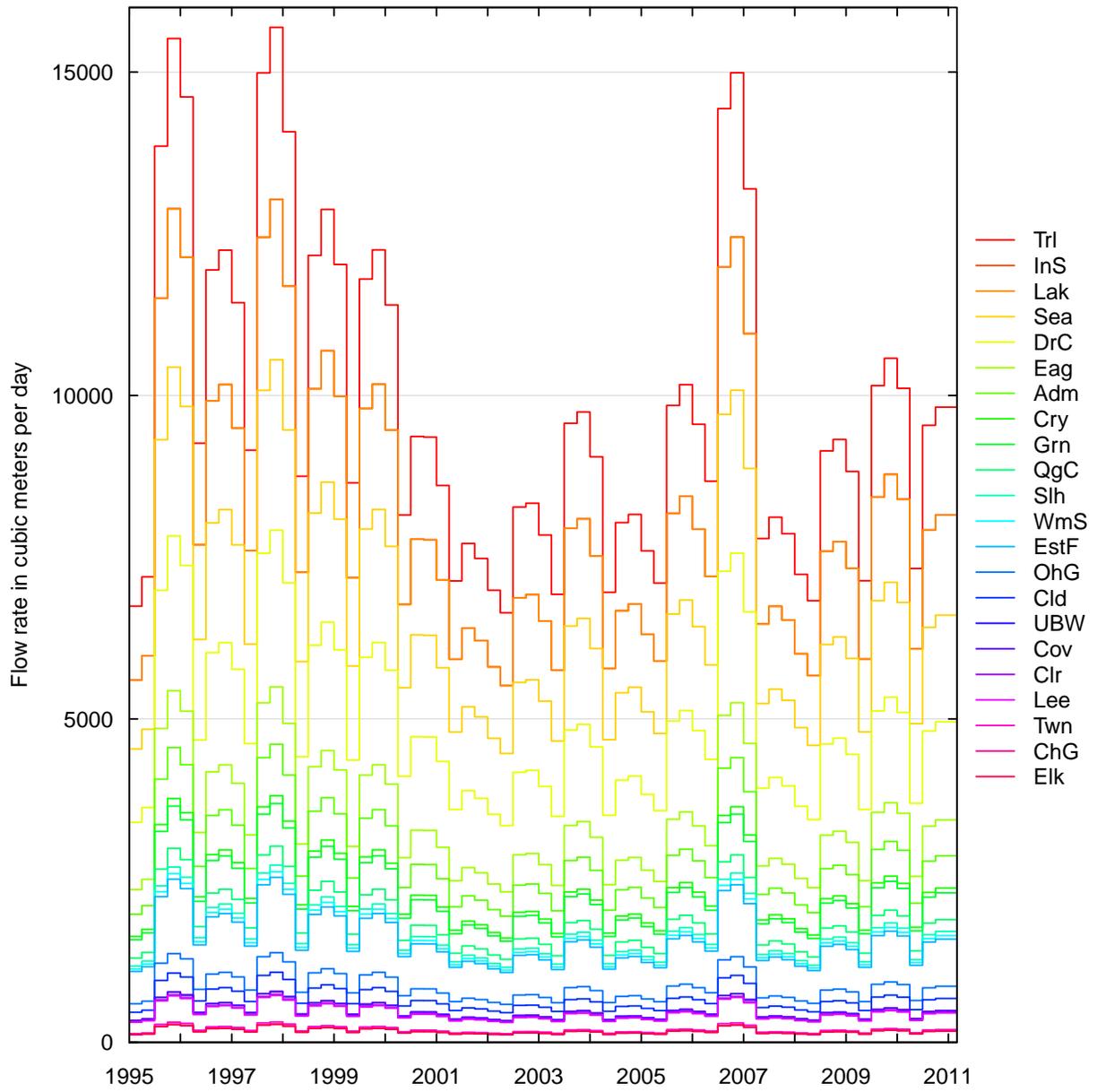


Figure 12: Seasonal flow rate in the tributary canyons and upper Big Wood River valley.

Table 3: Tributary canyons and upper Big Wood River valley specified flows.

Name	Identifier	Basin area (km ²)	Flow rate (m ³ /d)
Adams Gulch	Adm	28	2,874
BWR Upper	UBW	461	490
Chocolate Gulch	ChG	2	176
Clear Creek	Clr	6	463
Cold Springs Gulch	Cld	8	675
Cove Canyon	Cov	36	490
Croy Creek	Cry	73	2,377
Deer Creek	DrC	142	4,937
Eagle Creek	Eag	28	3,428
East Fork	EstF	223	1,591
Elkhorn Gulch	Elk	34	172
Greenhorn Gulch	Grn	54	2,303
Indian Creek	InS	28	8,128
Lake Creek	Lak	31	8,125
Lees Gulch	Lee	7	453
Ohio Gulch	OhG	13	865
Quigley Creek	QgC	44	1,891
Seamans Gulch	Sea	60	6,582
Slaughterhouse Gulch	Slh	34	1,709
Townshhead Gulch	TwN	3	196
Trail Creek	Trl	166	9,787
Warm Springs Creek	WmS	249	1,645

Groundwater Flow Beneath Silver Creek and Stanton Crossing

Groundwater leaving the aquifer system beneath Silver Creek and Stanton Crossing (fig. 1) is simulated using the MODFLOW Drain Package (Harbaugh *et al.*, 2000), a head-dependent flux boundary condition. If the head in a model cell falls below a certain threshold, the flux drops to zero; therefore, these model cells will only allow groundwater to leave the aquifer system. The drain conductance and elevation threshold at Silver Creek and Stanton Crossing are shown in table 4.

```
att.table <- as.data.frame(list(ID = 1:2, name = c("Silver Creek", "Stanton Crossing"),
                                cond = c(152, 210), elev = c(1450, 1461)))
```

Table 4: Drain conductance and elevation threshold for drain cell boundaries.

Name	Conductance (m ² /d)	Elevation (m)
Silver Creek	152	1,450
Stanton Crossing	210	1,461

The location of drain cells in model layer 1 is shown in figure 13. Note that Silver Creek drain cells also reside in model layers 2 and 3, and mirror the configuration of drain cells in layer 1.

```
l <- gIntersection(drains, as(aquifer.extent, "SpatialLinesDataFrame"), TRUE)
drain.lines <- SpatialLinesDataFrame(l, data = drains@data, match.ID = FALSE)
r <- rasterize(drain.lines, rs.model)
r[!is.na(r) & is.na(rs.model[["lay1.bot"]])] <- NA
r <- ratify(r)
```

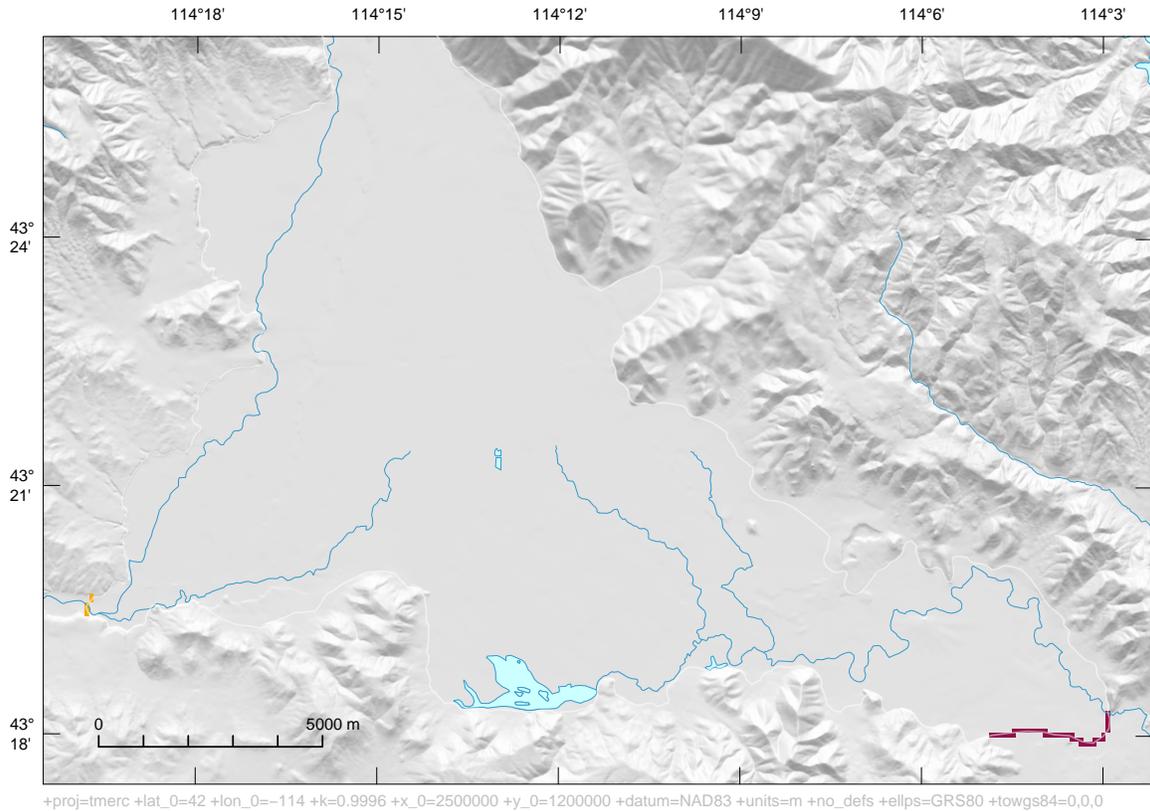


Figure 13: Location of drain cells in model layer 1.

```

levels(r) <- cbind(merge(levels(r)[[1]], att.table))
names(r) <- "lay1.bdry.drains"
rs.model <- stack(rs.model, r)
r <- rasterize(drain.lines, rs.model)
r[!is.na(r) & is.na(rs.model[["lay2.bot"]])] <- NA
r <- ratify(r)
levels(r) <- cbind(merge(levels(r)[[1]], att.table))
names(r) <- "lay2.bdry.drains"
rs.model <- stack(rs.model, r)
r <- rasterize(drain.lines, rs.model)
r[!is.na(r) & is.na(rs.model[["lay3.bot"]])] <- NA
r <- ratify(r)
levels(r) <- cbind(merge(levels(r)[[1]], att.table))
names(r) <- "lay3.bdry.drains"
rs.model <- stack(rs.model, r)

```

Stream-Aquifer Flow Exchange in the Big Wood River and Silver Creek

Stream-aquifer flow exchange in the Big Wood River and Silver Creek is simulated using the MODFLOW River Package (Harbaugh *et al.*, 2000), a head-dependent flux boundary condition. Note that the River Package does not account for the amount of flow in streams. Use of a more sophisticated package that accounts for streamflow, such as the MODFLOW Streamflow-Routing Package (Niswonger and Prudic, 2005), is infeasible due to insufficient data to describe these flows. To simplify the structural complexity of the rivers, major stream reaches were identified. A stream reach is defined as a section of a stream that has (1) uniform depth, (2) uniform riverbed thickness, and (3) uniform riverbed conductance (table 5). Riverbed conductance is initially specified at 850 m²/d for all stream reaches.

```
d <- bwr.sc@data[order(bwr.sc@data$ReachNo), c("Reach", "DrainRiver", "Depth", "BedThk")]
att.table <- cbind(ID = seq_len(nrow(d)), d, cond = 850)
```

Table 5: Description of stream reaches in the Big Wood River and Silver Creek.

Name	Type	Water depth (m)	Riverbed thickness (m)	Riverbed Conductance (m ² /d)
Big Wood, Nr Ketchum to Hulen Rd	river	0.6	0.3	850
Big Wood, Hulen Rd to Ketchum	river	0.6	0.3	850
Big Wood, Ketchum to Gimlet	river	0.6	0.3	850
Big Wood, Gimlet to Hailey	river	0.6	0.3	850
Big Wood, Hailey to N Broadford	river	0.6	0.3	850
Big Wood, N Broadford to S Broadford	river	0.6	0.3	850
Big Wood, S Broadford to Glendale	river	0.6	0.3	850
Big Wood, Glendale to Sluder	river	0.6	0.3	850
Big Wood, Sluder to Wood River Ranch	river	0.6	0.3	850
Big Wood, Wood River Ranch to Stanton Crossing	drain	0.6	0.3	850
Willow Creek	drain	0.3	0.9	850
Buhler Drain abv Hwy 20	drain	0.3	0.9	850
Patton Creek abv Hwy 20	drain	0.3	0.9	850
Cain Creek abv Hwy 20	drain	0.3	0.9	850
Chaney Creek abv Hwy 20	drain	0.3	0.9	850
Mud Creek abv Hwy 20	drain	0.3	0.9	850
Wilson Creek abv Hwy 20	drain	0.3	0.9	850
Grove Creek abv Hwy 20	drain	0.3	0.9	850
Loving Creek abv Hwy 20	drain	0.3	0.9	850
Spring creeks blw Hwy 20	river	0.6	0.9	850
Silver Creek, Sportsman Access to Nr Picabo	river	0.6	0.9	850

River cells are identified using horizontal polylines with a single polyline allocated to each of the stream reaches.

```
r <- rasterize(bwr.sc, rs.model, field = "ReachNo")
r[is.na(rs.model[["lay1.bot"]]) | !is.na(rs.model[["lay1.bdry.drains"]])] <- NA
r <- ratify(r)
levels(r) <- cbind(merge(levels(r)[[1]], att.table))
names(r) <- "riv.reach"
rs.model <- stack(rs.model, r)
```

Calculate the elevation of the riverbed bottom.

```
r <- rs.model[["lay1.top"]] - deratify(rs.model[["riv.reach"]], "Depth") -
  deratify(rs.model[["riv.reach"]], "BedThk")
```

```
names(r) <- "riv.bottom"
rs.model <- stack(rs.model, r)
```

Stream stage is based on the stream reach type (table 5). For stream reaches of type 'river', the stream depth is assumed at land surface; whereas for type 'drain', the depth is specified at the riverbed bottom. Figure 14 shows the delineation of stream reach types.

```
r <- mask(rs.model[["lay1.top"]], rs.model[["riv.reach"]])
is.riv.drain <- deratify(rs.model[["riv.reach"]], "DrainRiver")[] == 2L
r[is.riv.drain] <- rs.model[["riv.bottom"]][is.riv.drain]
names(r) <- "riv.stage"
rs.model <- stack(rs.model, r)
```

Starting Hydraulic Head Distribution

The starting head distribution for the steady-state stress period is specified at 1 meter below land surface.

```
starting.head.depth <- 1 # in meters
r <- rs.model[["lay1.top"]] - starting.head.depth
names(r) <- "lay1.strt"
rs.model <- stack(rs.model, r)
r <- rs.model[["lay1.strt"]]
r[is.na(rs.model[["lay2.bot"]])] <- NA
names(r) <- "lay2.strt"
rs.model <- stack(rs.model, r)
r <- rs.model[["lay1.strt"]]
r[is.na(rs.model[["lay3.bot"]])] <- NA
names(r) <- "lay3.strt"
rs.model <- stack(rs.model, r)
```

Aerial Recharge and Well Pumping

Something... in figure 15.

```
ans <- ProcessRecharge(tr.stress.periods, rs.model)
r <- calc(subset(ans[["aerial.rech"]], format(ss.stress.periods, "%Y%m")), fun = mean)
names(r) <- "ss"
rech <- stack(ans[["aerial.rech"]], r, quick = TRUE)
```

Something...

```
well <- GetWellConfig(ans[["pod.rech"]], rs.model)
well$ss <- rowMeans(well[, format(ss.stress.periods, "%Y%m")], na.rm = TRUE)
```

Data Archive

Export raster stacks as georeferenced image files; place files in a subdirectory of the working directory.

```
dir.out <- file.path(getwd(), paste0("wrv_", format(Sys.time(), "%Y%m%d%H%M%S")))
dir.create(path = dir.out)
ExportRasterStack(file.path(dir.out, "Data"), rs.data)
ExportRasterStack(file.path(dir.out, "Model"), rs.model)
```

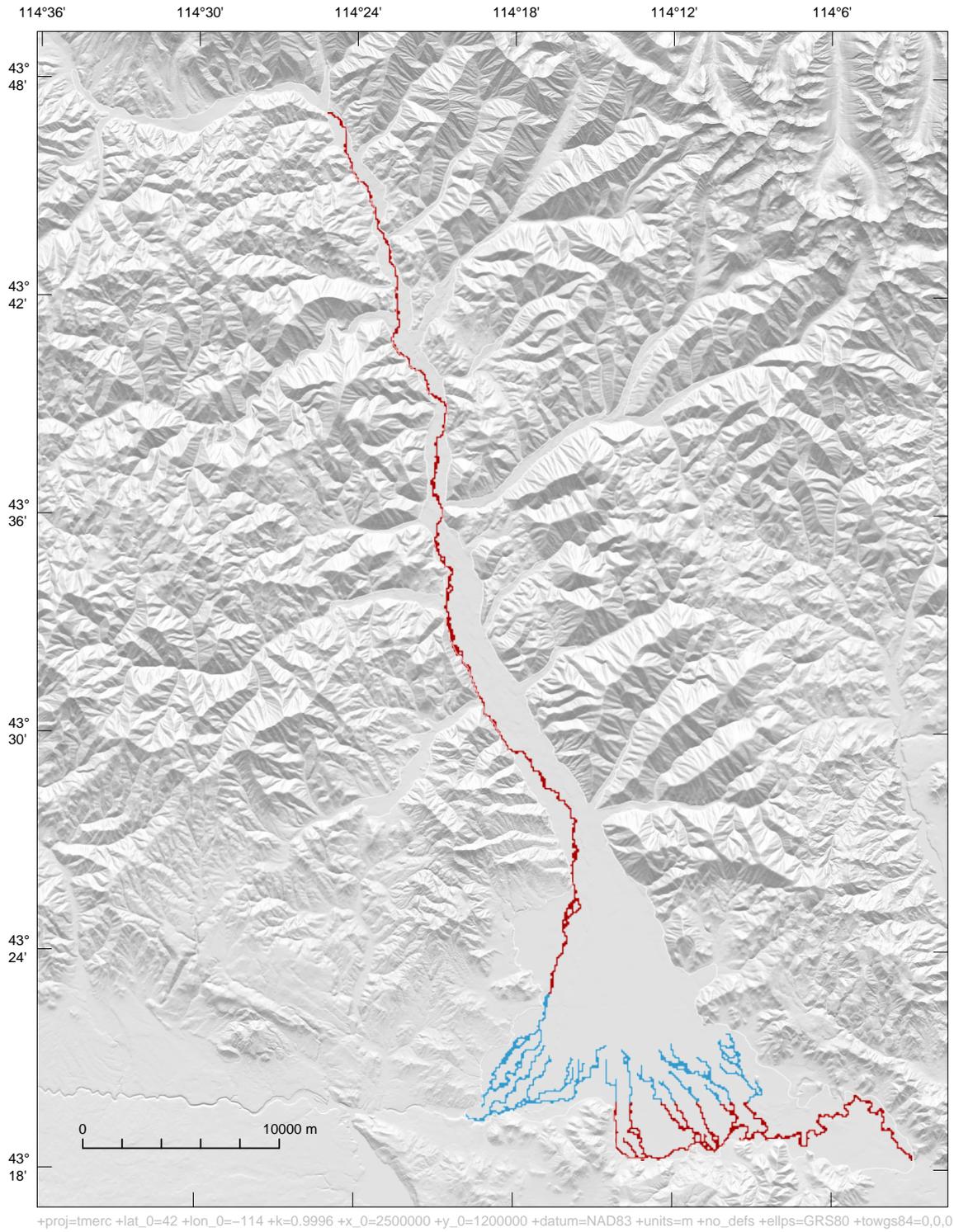


Figure 14: Stream-reach types in the Big Wood River and Silver Creek.

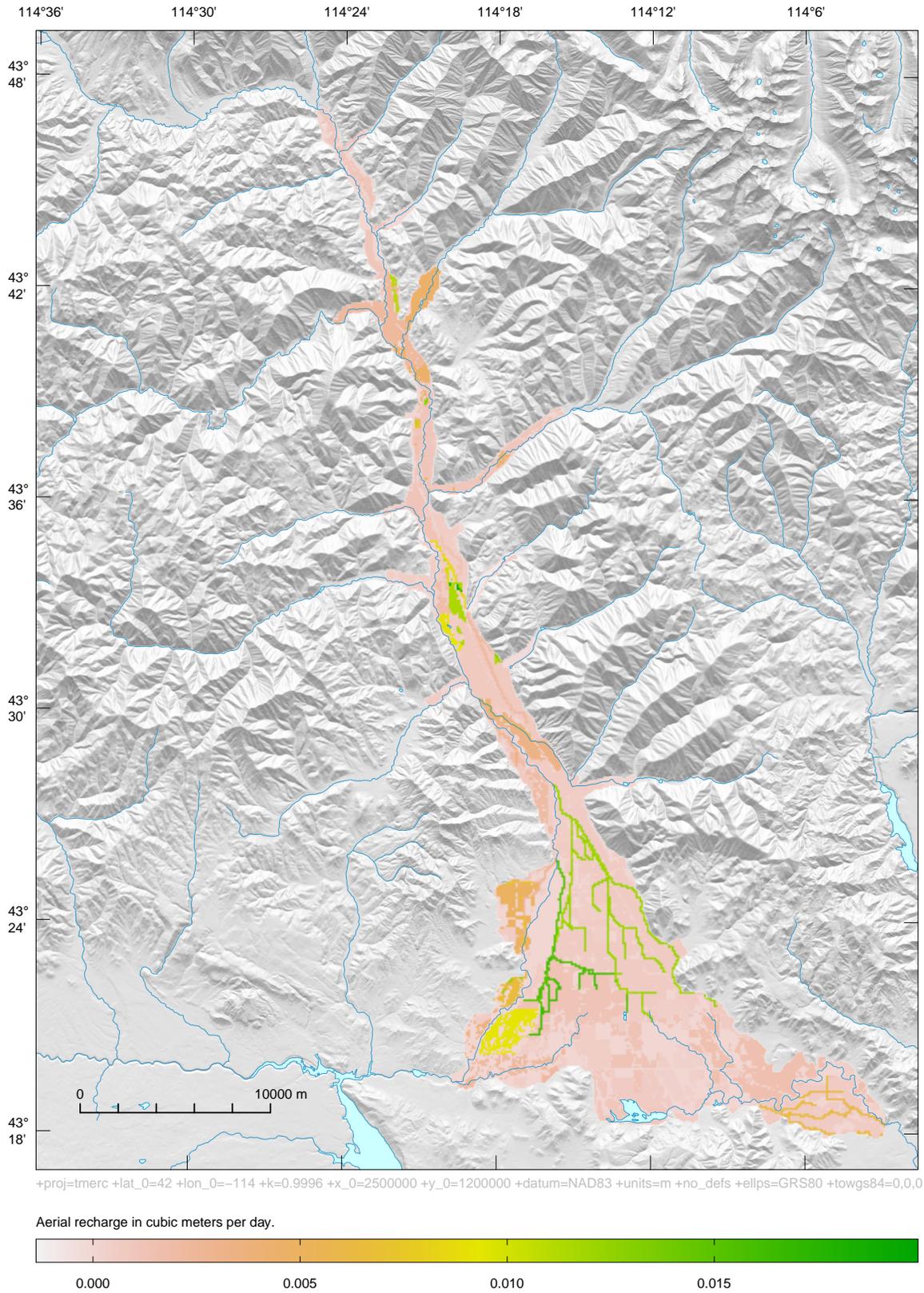


Figure 15: Steady-state aerial recharge in the aquifer system.

Create a horizontal polygon for each model cell and assign location and elevation attributes; place shapefile in a subdirectory of the working directory.

```
r <- raster(rs.model)
r[] <- seq_len(ncell(r))
names(r) <- "cell"
p <- rasterToPolygons(r)
p@data <- cbind(p@data, rowColFromCell(r, p@data$cell),
               lay1.top = rs.model[["lay1.top"]][p@data$cell],
               lay1.bot = rs.model[["lay1.bot"]][p@data$cell],
               lay2.bot = rs.model[["lay2.bot"]][p@data$cell],
               lay3.bot = rs.model[["lay3.bot"]][p@data$cell])
writeOGR(p, dsn = file.path(dir.out, "Model"), layer = "grid", driver = "ESRI Shapefile")
```

Model Run

Steady-state 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.

```
id <- "wrv_ss_mfug" # model run identifier
dir.run <- file.path(dir.out, "Run")
perlen <- as.integer(diff(ss.interval))
arguments <- list(rs.model = rs.model, rech = rech, well = well,
                 id = id, dir.run = dir.run, perlen = perlen)
do.call(CreateModflowInputFiles, arguments)
```

Create and execute a *batch file* containing commands that run MODFLOW-USG.

```
cmd <- c(paste("cd", shQuote(dir.run)),
        paste(shQuote(file.exe), shQuote(paste0(id, ".nam"))))
file.bat <- file.path(dir.run, "Run.bat")
cat(cmd, file = file.bat, sep = "\n")
Sys.chmod(file.bat, mode = "755")
output <- system(shQuote(file.bat), intern = TRUE)
```

Captured output from running the model is provided below.

```
#
#                                MODFLOW-USG
#    U.S. GEOLOGICAL SURVEY MODULAR FINITE-DIFFERENCE GROUNDWATER FLOW MODEL
#                                Version 1.2.00 03/21/2014
#
# Using NAME file: wrv_ss_mfug.nam
# Run start date and time (yyyy/mm/dd hh:mm:ss): 2014/09/09 21:47:53
#
# Solving:  Stress period:      1    Time step:      1    Groundwater Flow Eqn.
# Run end date and time (yyyy/mm/dd hh:mm:ss): 2014/09/09 21:57:32
# Elapsed run time:  9 Minutes, 38.891 Seconds
#
# Normal termination of simulation
```

The volumetric budget at the end of the steady-state simulation is shown in table 6.

Table 6: Volumetric budget for entire model at end of time step 1 stress period 1.

	Vol. (m ³)	Vol. (acre-ft)	Rate (m ³ /d)	Rate (acre-ft/yr)
Storage in	0	0	0	0
Constant head in	0	0	0	0
Wells in	0	0	0	0
Drains in	0	0	0	0
River leakage in	80,620,955	65,360	221,486	65,584
Recharge in	138,984,789	112,676	381,826	113,061
Specified flows in	16,928,171	13,724	46,506	13,771
Total in	236,533,916	191,760	649,818	192,416
Storage out	0	0	0	0
Constant head out	0	0	0	0
Wells out	70,291,648	56,986	193,109	57,181
Drains out	15,387,959	12,475	42,275	12,518
River leakage out	150,273,851	121,828	412,840	122,245
Recharge out	580,453	471	1,595	472
Specified flows out	0	0	0	0
Total out	236,533,910	191,760	649,818	192,416
In minus out	5	0	0	0
Percent discrepancy	0	0	0	0

Simulated Hydraulic Heads

Read simulated hydraulic heads (heads) for each model layer into a raster stack (fig. 16).

```
heads <- ReadModflowBinaryFile(file.path(dir.run, paste0(id, ".hds")))
rs.head <- stack() # initialize a raster stack for simulated heads
r <- raster(rs.model)
r[] <- heads[[1]]$d
r[!rs.model[["lay1.ibound"]]] <- NA
names(r) <- "lay1.head"
rs.head <- stack(rs.head, r)
r[] <- heads[[2]]$d
r[!rs.model[["lay2.ibound"]]] <- NA
names(r) <- "lay2.head"
rs.head <- stack(rs.head, r)
r[] <- heads[[3]]$d
r[!rs.model[["lay3.ibound"]]] <- NA
names(r) <- "lay3.head"
rs.head <- stack(rs.head, r)
```

Determine which model cells are saturated in model layer 1 (fig. 17).

```
r <- rs.head[["lay1.head"]] > rs.model[["lay1.top"]]
r <- ratify(r)
levels(r) <- cbind(levels(r)[[1]], att = c("partially saturated", "saturated"))
names(r) <- "lay1.saturated"
rs.head <- stack(rs.head, r)
```

Write simulated heads to georeferenced image files.

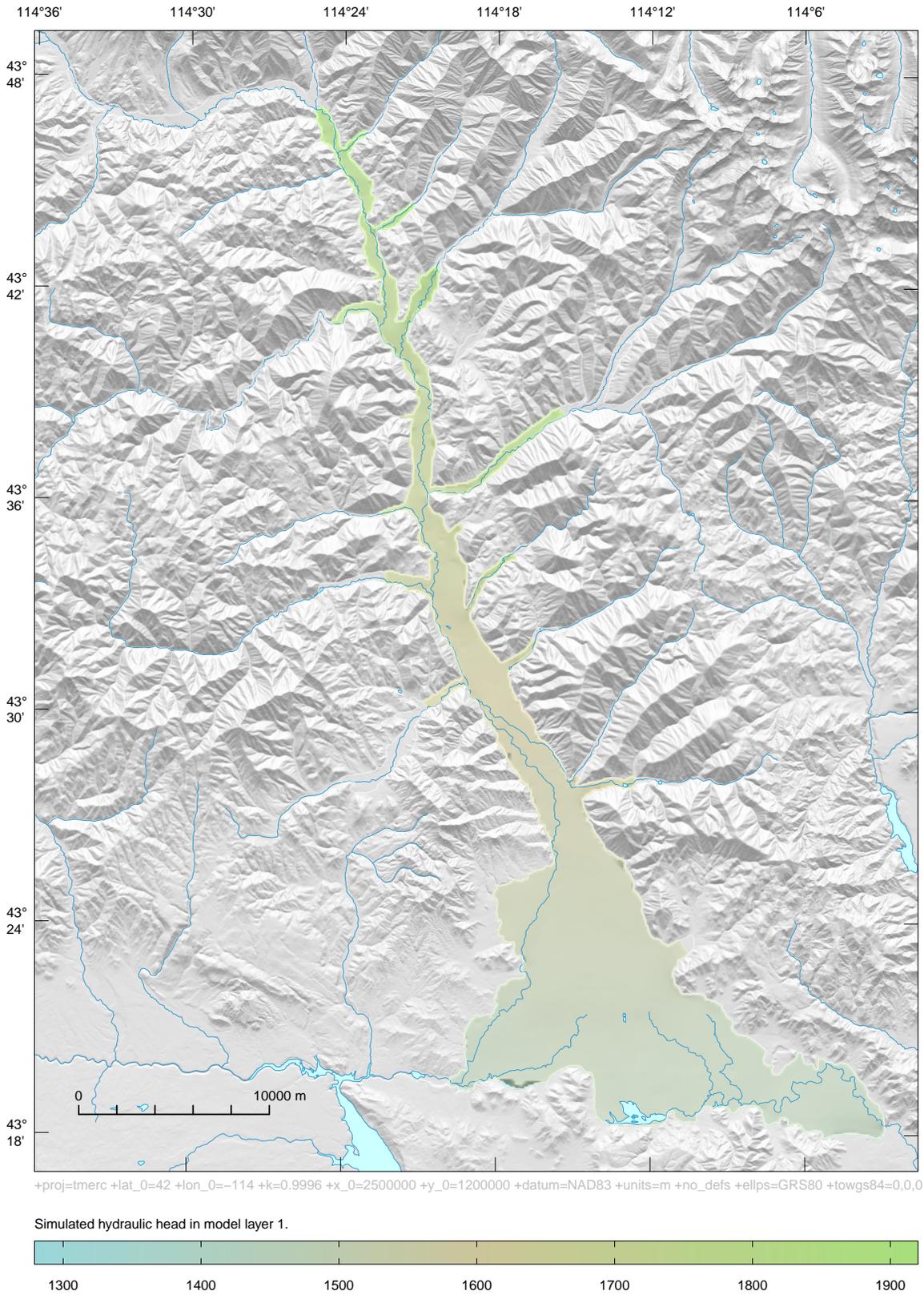


Figure 16: Simulated hydraulic head in model layer 1.

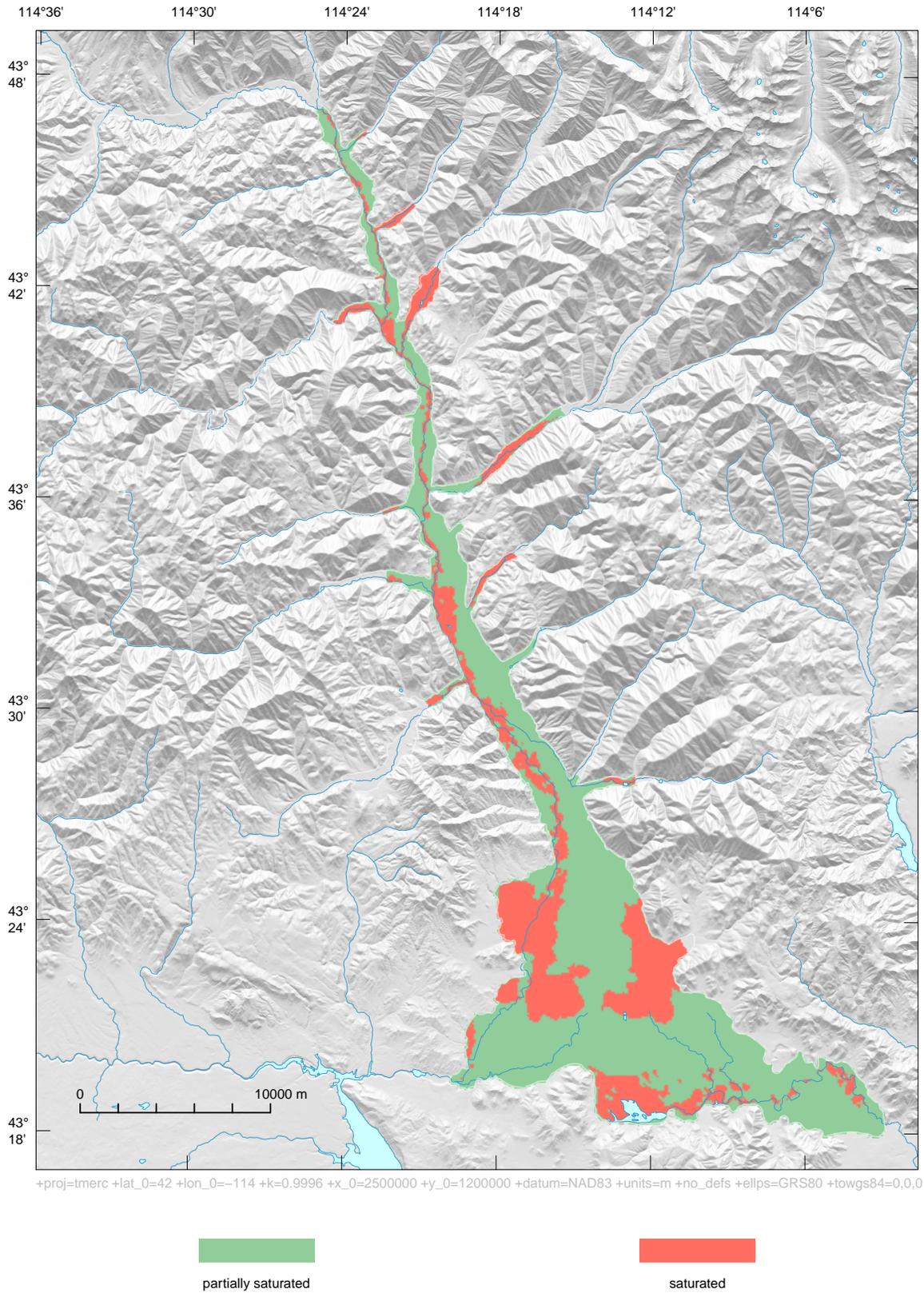


Figure 17: Saturated and partially-saturated cells in model layer 1.

```
ExportRasterStack(dir.run, rs.head)
```

Reproducibility

To reprocess the groundwater flow model, evaluate R code extracted from this vignette using the following command:

```
source(system.file("doc", "wrv-process.R", package = "wrv"), echo = TRUE)
list.files(dir.out, full.names = TRUE, recursive = TRUE) # path names of output files
```

Version information about R and attached or loaded packages is as follows:

- R version 3.1.1 (2014-07-10), x86_64-w64-mingw32
- Base packages: base, datasets, grDevices, graphics, methods, stats, utils
- Other packages: RCurl 1.95-4.3, bitops 1.0-6, igraph 0.7.1, raster 2.3-0, rgdal 0.8-16, rgeos 0.3-6, sp 1.0-15, wrv 0.1-5, xtable 1.7-3
- Loaded via a namespace (and not attached): evaluate 0.5.5, formatR 1.0, grid 3.1.1, highr 0.3, knitr 1.6, lattice 0.20-29, stringr 0.6.2, tools 3.1.1

Total processing time for this vignette was 19 minutes, built on September 09, 2014.

References

- Bartolino J, Adkins C (2012). "Hydrogeologic framework of the Wood River Valley aquifer system, south-central Idaho." *Scientific Investigations Report 2012-5053*, U.S. Geological Survey. URL <http://pubs.usgs.gov/sir/2012/5053/>.
- Fisher J, Bartolino J, Wylie A, Sukow J, McVay M, Vincent S (2015). "Groundwater flow model of the Wood River Valley aquifer system, south-central Idaho." U.S. Geological Survey: Scientific Investigations Report.
- Harbaugh A, Banta E, Hill M, McDonald M (2000). "MODFLOW-2000, the U.S. Geological Survey Modular Ground-Water Model-User guide to modularization concepts and the Ground-Water Flow Process." *Open-File Report 00-92*, U.S. Geological Survey. URL <http://pubs.er.usgs.gov/publication/ofr200092>.
- Leake S, Michael R (1997). "Documentation of a computer program (FHB1) for assignment of transient specified-flow and specified-head boundaries in applications of the Modular Finite-Difference Ground-Water Flow Model (MODFLOW)." *Open-File Report 97-571*, U.S. Geological Survey. URL <http://pubs.er.usgs.gov/publication/ofr97571>.
- Niswonger R, Prudic D (2005). "Documentation of the Streamflow-Routing (SFR2) Package to include unsaturated flow beneath streams-A modification to SFR1." In *Techniques and Methods*, chapter A13, p. 50. U.S. Geological Survey. URL <http://pubs.usgs.gov/tm/2005/tm6A16/>.
- Panday S, Langevin C, Niswonger R, Ibaraki M, Hughes J (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." In *Techniques and Methods*, chapter A45, p. 66. U.S. Geological Survey. URL <http://pubs.usgs.gov/tm/06/a45>.
- R Core Team (2014). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL <http://www.R-project.org/>.