CW3M Digital Handbook

Information about the Community Willamette Whole Watershed Model

4/7/22

CW3M is maintained by Oregon Freshwater Simulations Inc.

contact: [david.conklin@freshwatersim.com](mailto:david.conklin@freshwatersim.com)

Changes since the 1.2.2 (2/18/22) version

Contents

[What is a digital handbook? 4](#_Toc99004988)

[CW3M concept 5](#_Toc99004989)

[A different kind of community model 5](#_Toc99004990)

[What CW3M is for 5](#_Toc99004991)

[Management and maintenance 5](#_Toc99004992)

[CW3M – A “first principles” model with tuning knobs 6](#_Toc99004993)

[The tuning knobs 6](#_Toc99004994)

[ET\_MULT – forest evapotranspiration 6](#_Toc99004995)

[SOLAR\_RADIATION\_MULTIPLIER – reservoir insolation 7](#_Toc99004996)

[FLOW\_CMS – Discharge from High Cascade springs 7](#_Toc99004997)

[The Envision framework and the GIS Layers 7](#_Toc99004998)

[The Reach Layer 7](#_Toc99004999)

[Time steps, autonomous processes, and global methods 8](#_Toc99005000)

[Some global method details 9](#_Toc99005001)

[Study areas 9](#_Toc99005002)

[DataCW3M directory structure 10](#_Toc99005003)

[DataCW3M\ScenarioData directory structure 10](#_Toc99005004)

[DataCW3M\ScenarioData\<scenario> folder contents 11](#_Toc99005005)

[DataCW3M\<study area> folder contents 11](#_Toc99005006)

[Calendar conventions 17](#_Toc99005007)

[Water years 17](#_Toc99005008)

[Climate data 17](#_Toc99005009)

[Numbered climate scenarios in the model 17](#_Toc99005010)

[Monthly and seasonal weather data 18](#_Toc99005011)

[Climate data grids 19](#_Toc99005012)

[The WW2100 climate grid 19](#_Toc99005013)

[The v2 climate grid 20](#_Toc99005014)

[How climate data is looked up 20](#_Toc99005015)

[What is in a climate dataset 21](#_Toc99005016)

[Water rights 22](#_Toc99005017)

[Water rights data 22](#_Toc99005018)

[The wr\_pods.csv file 22](#_Toc99005019)

[The wr\_pous.csv file 23](#_Toc99005020)

[Allocating surface water rights to irrigation 24](#_Toc99005021)

[Adding a water right 24](#_Toc99005022)

[Initial conditions 25](#_Toc99005023)

[State variables 25](#_Toc99005024)

[Reproducibility and the use of random number generators 25](#_Toc99005025)

[Initializing ID numbers: coldstart 26](#_Toc99005026)

[Initializing Flow and HBV model state variables: spinup 26](#_Toc99005027)

[Initializing HBVCALIB attributes: spinup 26](#_Toc99005028)

[Initial conditions for Flow: the IC file and the IDU, HRU, and Reach shapefiles 27](#_Toc99005029)

[What is in the IC file 27](#_Toc99005030)

[State variables in the IDU shapefile 27](#_Toc99005031)

[Initial conditions for HBV: the IC file and the IDU shapefile 28](#_Toc99005032)

[Default values used when the data is incomplete 28](#_Toc99005033)

[Stream flow and stream temperature 29](#_Toc99005034)

[Water parcels 29](#_Toc99005035)

[Daily water mass and energy balance 29](#_Toc99005036)

[Estimating the rate of flow in a stream reach 31](#_Toc99005037)

[Estimating the surface area of the water in a subreach 33](#_Toc99005038)

[Water temperature from thermal energy 34](#_Toc99005039)

[Boundary conditions for stream water temperature 34](#_Toc99005040)

[Thermal stratification in reservoirs 35](#_Toc99005041)

[Thermal loading 35](#_Toc99005042)

[Reservoirs 36](#_Toc99005043)

[Reservoir data 37](#_Toc99005044)

[Reservoir data sources 39](#_Toc99005045)

[Creating a new study area from a watershed within the WRB 39](#_Toc99005046)

[Limitations of subbasin simulations 41](#_Toc99005047)

[Calibration procedure 41](#_Toc99005048)

[Using PEST to determine HBV parameter values 42](#_Toc99005049)

[Calibrating streamflows using stream gage measurements 42](#_Toc99005050)

[Calibrating stream temperatures 43](#_Toc99005051)

[North Santiam study area 44](#_Toc99005052)

[Instream water rights 44](#_Toc99005053)

[Municipal water rights 45](#_Toc99005054)

[Reality check 45](#_Toc99005055)

[Inflows from springs 47](#_Toc99005056)

[Calibration and simulation skill 47](#_Toc99005057)

[Analysis of tuning knob values and locations where the model has low skill 50](#_Toc99005058)

[Simulated North Santiam basin water budget and thermal energy budget 50](#_Toc99005059)

[McKenzie Basin Wetlands Study 53](#_Toc99005060)

[Project Overview 53](#_Toc99005061)

[Model and Simulation Overview 55](#_Toc99005062)

[McKenzie wetlands in the initial versions of the IDU, HRU, and Reach data layers 56](#_Toc99005063)

[Data changes for better representation of wetlands 56](#_Toc99005064)

[Projection, Calendar, and Units 57](#_Toc99005065)

[Simulation of changes in wetlands over time 57](#_Toc99005066)

[How wetlands are represented in the model 57](#_Toc99005067)

[The Wetland class and Wetland objects in the C++ code 59](#_Toc99005068)

[Topology of wetlands, HRUs, IDUs, and reaches 59](#_Toc99005069)

[Attributes of interest in the wetlands study 59](#_Toc99005070)

[A WETNESS attribute 59](#_Toc99005071)

[Calculating the exchange of water between the wetland and the reach 60](#_Toc99005072)

[Wetland IDU parameters 61](#_Toc99005073)

[Reach parameters 61](#_Toc99005074)

[Flooding 61](#_Toc99005075)

[Implementation of the exchange of water between the wetland and the reach 61](#_Toc99005076)

[Standing water temperature 64](#_Toc99005077)

[Evapotranspiration from wetlands 64](#_Toc99005078)

[Loss (or gain) of wetlands 64](#_Toc99005079)

[Wetland state and transition model 65](#_Toc99005080)

[Activating a wetland in the wetland model 66](#_Toc99005081)

[Reality check 66](#_Toc99005082)

[High Cascades Springs 68](#_Toc99005083)

[Calibration and simulation skill 70](#_Toc99005084)

[Analysis of locations where the model has low skill 73](#_Toc99005085)

[Simulated McKenzie basin water budget and thermal energy budget 73](#_Toc99005086)

[CW3M v. Shade-a-lator 75](#_Toc99005087)

[Comparison of CW3M results to Shade-a-lator results 77](#_Toc99005088)

[The Clackamas Basin 79](#_Toc99005089)

[Autonomous process details 80](#_Toc99005090)

[Prescribed LULCs 80](#_Toc99005091)

[Directory structure and file names 81](#_Toc99005092)

[C++ Source Code Conventions 82](#_Toc99005093)

[Conklin’s conventions for variable names 82](#_Toc99005094)

[Conklin’s conventions for the IDU layer and its attributes 82](#_Toc99005095)

[Release Log 83](#_Toc99005096)

[References 88](#_Toc99005097)

[Acronyms and Abbreviations 90](#_Toc99005098)

Figures

[Figure 1. Willamette River basin study area; embedded study areas are colored other than gray. The 2 lighter blues together make up the upper Willamette basin study area. 12](#_Toc87080447)

[Figure 2. Tualatin basin study area; Chicken Creek watershed is highlighted. 13](#_Toc87080448)

[Figure 3. North Santiam watershed study area. 14](#_Toc87080449)

[Figure 4. Upper Willamette study area. 15](#_Toc87080450)

[Figure 5. Chicken Creek watershed study area. 16](#_Toc87080451)

[Figure 6. Instream water rights in the North Santiam watershed 45](#_Toc87080452)

[Figure 7. McKenzie basin in CW3M screen capture 54](#_Toc87080453)

[Figure 8. Wetlands in the lower McKenzie basin, with their WETL\_ID attribute values 55](#_Toc87080454)

[Figure 9. Clackamas basin in CW3M screen capture 79](#_Toc87080455)

Tables

[Table 1. Numbered climate scenarios 18](#_Toc99341137)

# What is a digital handbook?

We intend the digital handbook to collect all our information about CW3M in a single place. “CW3MdigitalHandbook” is the name of a folder on a Windows file system which contains multiple files, including the MS Word file of this document (CW3MdigitalHandbook.docx) and an accompanying Excel file (CW3Mhandbook.xlsx).

# CW3M concept

## A different kind of community model

CW3M stands for Community Willamette Whole Watershed Model. The “Willamette Whole Watershed Model” part is a generic term for the series of whole watershed models developed by the WW2100, OUWIN, and Willamette INFEWs projects. The “Community” part is inspired by the community models maintained by the National Center for Atmospheric Research: the Community Earth System Model (CESM), and the Whole Atmosphere Community Climate Model (WACCM). NCAR’s “Community” is the community of scientists who develop and use the NCAR models. The “Community” in CW3M is made up of the analogous group of scientists from the research projects, but includes also the community of policy makers, land managers, reservoir operators, NGOs, government agencies, utility operators, farmers, and others who can make use of the model outputs - the stakeholder community. CW3M is for that larger community. It retains, however, some of the key features of NCAR’s community models: open access to source code, the opportunity for qualified people to contribute to model development and data, and rigorous version control.

## What CW3M is for

The intent for CW3M is to flesh out the Willamette basin regional results with more details at the scale of the Willamette's tributary rivers. As of mid 2020, model data files have been set up for 13 study areas: the entire Willamette basin; the smaller basins drained by the Clackamas, Long Tom, Luckiamute, Marys, McKenzie, Molalla, North Santiam, Pudding, and Tualatin rivers; the Chicken Creek watershed in the Tualatin basin; and the upper Willamette basin, composed of the areas drained by the Coast Fork, the Middle Fork, and the McKenzie rivers. Ultimately CW3M would include detailed models of all the major subregions in the Willamette River basin.

A second objective of the CW3M project is to improve the model's skill through reconciliation with available independent datasets. This is a continuation of work already started at the regional scale, comparing simulated streamflows with USGS gage records for the actual weather of recent decades; results of that work were reported at the 2018 NW Scientific Association conference. This effort is expected to improve the calibration of many of the constituent submodels: not just streamflow but also urban water use, irrigation, population, crop choice, land use change, expansion of urban growth areas, and so on. Opportunities for better calibration increase as the spatial focus narrows from the whole WRB to tributary subbasins.

The goal is to produce an open source model and an accompanying collection of datasets which are useful to agencies and organizations involved in planning and management of lands in the Willamette River basin, especially in the context of the stresses and uncertainties arising from climate change and population growth. The model, if successful, will provide process-based and data-driven conjectures in response to some of the essential what-if questions.

## Management and maintenance

Unlike for NCAR’s community models, there is no large organization behind CW3M. A CW3M Steering Committee has been set up and is composed of 8 individuals who have agreed to a memorandum of understanding (MOU). The text of the MOU is a separate file in the digital handbook. The steering committee meets quarterly over the Internet. Rebecca McCoun is the current steering committee chairperson. She can be reached by email at northsantiamwc@gmail.com.

As of mid-2020, Oregon Freshwater Simulations (Freshwater) is continuing as the “principal developer” of CW3M, a role described in the MOU. Freshwater is a tiny consultancy in Portland. It was the keeper of the Envision-based models for the WW2100 and OUWIN project, and for the initial years of the Willamette INFEWS project, acting as a vendor to Oregon State University.

# CW3M – A “first principles” model with tuning knobs

CW3M is a process model at heart; it uses the physical relationships between real world objects and energies to simulate changes across time. The sun shines; the sunshine falls on a stream; the water in the stream warms up. To a useful extent, the first principles are understood and can be represented in computer algorithms to estimate how much warmer the water will get.

The model is “correct” if it embodies the first principles accurately. If the modeler codes accurately the equation for water temperature change as a function of solar energy, then the model’s estimate of the amount of warming, *given a specified amount of solar energy*, will be correct.

The model is “skillful” to the extent that, when driven with real world input data, it produces answers that approximate what actually happens in the real world. We test its skill by driving it with historical real world input data, and comparing the model’s answers with what actually happened historically.

The model is useful when it is skillful enough to produce plausible, defensible projections of what may happen in the future as the real world inputs change. How much warmer might the stream water be, assuming the climate warms? Will it be too warm for salmon?

To be skillful, the model must first be correct, and in addition must be supplied with reasonably accurate data about the real world itself. To estimate the temperature in a headwater stream in late spring, it is important to simulate the evolution of the snowpack during the spring thaw. To get that right requires good daily weather data – temperature and precipitation, at a minimum. Happily, pretty good historical weather data really is available, and CW3M uses it.

Some data about the real world, although necessary for skillful simulation, simply isn’t available. For example, a sizable fraction (perhaps a third) of the precipitation that falls in the Willamette basin is returned to the atmosphere as evaptranspiration (ET), rather than ending up in the stream network. The model does a creditable job of estimating from first principles how much water is transpired by the upland forests, *as a function of how much leaf area* is in the forests. But the amount of leaf area itself is poorly constrained. The GIS data layer showing forest stands by type and age was built from remote sensing data taken in the early 2000s. Since then, parts of the forests have grown up, parts have been harvested, and parts have burned down. We don’t have accurate data about upland forest leaf area.

CW3M accomodates such situations using tuning knobs. For evapotranspiration, the tuning knob is called “ET\_MULT”. It is applied as a multiplicative fudge factor applied to the model’s initial estimate of ET, to make the simulated overall water budget more consistent with historical records. If the model is simulating 5% too much water flowing out of the mouth of a river compared to actual gage measurements, the simulated amount may be tuned to the gage measurements by increasing ET\_MULT by some amount. Adjusting ET\_MULT is a trial-and-error process.

# The tuning knobs

## ET\_MULT – forest evapotranspiration

The model’s initial estimate of evapotranspiration in forested IDUs is multiplied by ET\_MULT. Values of ET\_MULT are specified by calibrated drainage (HBVCALIB) in the HBV.csv file. As of 12/12/20, values range from 0.5 (Blue River reservoir drainage) to 1.93 (Mohawk River basin).

## SOLAR\_RADIATION\_MULTIPLIER – reservoir insolation

The amount of sunshine absorbed by the water in reservoirs depends on topographic shading, among other things. The extent of topographic shading itself depends not only on the topography surrounding the reservoir, but also on the season and on the amount of water in the reservoir. As of 12/12/20, the model does not have an algorithm for estimating the decrease in solar radiation absorbed by the water in a reservoir due to topographic shading. Instead, a value of SOLAR\_RADIATION\_MULTIPLIER may be specified in the <reservoir> block for a given reservoir, in the Flow XML file. As of 12/12/20 this has been done only for the Cougar (0.45) and Blue River (0.15) reservoirs.

## FLOW\_CMS – Discharge from High Cascade springs

More water, and colder water, flows into the Willamette mainstem from its tributaries in the western Cascades than would be expected from the amount of precipitation in their drainages. Springs of uncertain origin and amount contribute water at multiple locations. To produce simulated flows more consistent with gage measurements, spring discharge locations and flow rates are specified in the Flow XML file. Here is an example for spring water added to reach 23773429 above the Blue River reservoir, used to make the simulated flow into the reservoir more consistent with measurements from USGS gage 14161100 located on that reach.

<Spring name="Springs feeding Blue R at and above Tidbit Creek"

COMID="23773429"

flow\_cms="1.60"

temp\_C="4.9"

/>

As of 12/12/20, seven <Spring> entries have been added to the Flow XML file for the McKenzie basin study area. For more information, see the “High Cascades Springs” section, later in this document.

# The Envision framework and the GIS Layers

CW3M uses the Envision modeling framework created by John Bolte and colleagues at Oregon State University. The particular version of Envision used in CW3M evolved from the version used by the WW2100 project, splitting off from the main line of Envision model development at OSU in about 2013 or 2014.

Central to the Envision framework as used in CW3M are three ArcGIS shapefiles: the IDU, HRU, and Reach layers. IDU stands for “independent decision unit” and is the spatial unit of computation. IDUs are arbitrary polygons which tile the watershed under study. The Reach layer represents the stream network which drains the watershed. HRU stands for “hydrologic resource unit” and ties the IDU and Reach layers together by HRU polygons composed of IDUs which all drain to the same reach. The use of shapefiles (as opposed to grids) underlies an important feature of the Envision framework: the landscape can be represented using a heterogeneous spatial scale. In uniform parts of the landscape, the IDU polygons can be large, while smaller polygons can be used in areas where higher information density is required.

## The Reach Layer

Conceptually, the Reach layer is a set of lines structured as a tree, with flow downstream from “leaves” (headwater reaches) to a single “root” (the watershed outlet, a.k.a. “pourpoint” in ArcGIS jargon). By convention, each reach can be connected to only a single downstream reach and no more than two upstream reaches.

Reaches may be of arbitrary length but should be relatively uniform in the characteristics which matter to the simulation, for example direction, width, streamside vegetation, and so on. For mathematical convenience, reaches longer than 1 kilometer are further subdivided into equal-length subreaches. For example, a reach of length 1.3 km is subdivided into two 650 m subreaches, a 2.1 km long reach would be composed of three 700 m subreaches, and so on. By convention, the division into subreaches does not reflect information about the actual reach; it is merely a mathematical convenience.

As of August 2021, CW3M uses the Reach layer from the WW2100 project, with only a few corrections. This is not ideal for modeling stream temperature, since many reaches are curved to such a degree that insolation varies significantly along their lengths. When implementing the stream temperature model, consideration was given to using subreaches as the unit of computation, but that approach was ultimately abandoned. To address the issue of undesirable heterogeneity within individual reaches, it will be necessary to come up with a more detailed reach layer. The reach layer inherited from the WW2100 project was based on the National Hydrography Dataset ca. 2010, but may not have used the highest resolution version of that dataset currently available in 2021.

## Time steps, autonomous processes, and global methods

Envision was originally designed to run at a yearly timestep. Autonomous processes (APs) are executed at the yearly timestep; the order in which the APs execute is important, and is specified in the ENVX file.

The Flow model is an autonomous process and executes once per yearly timestep. Within the Flow model there is a daily timestep. The Flow XML file determines what happens in the daily timestep, among other things, by specifying “global methods” which execute at the daily timestep.

It is helpful to remember the APs as yearly processes and the global methods as daily processes. The yearly and daily processes used in CW3M release 1.0.0 9/9/21) in the McKenzie Wetlands project are listed below. The processes are grouped in the executable code elements which implement them.

Yearly processes, as in CW3M\_McKenzie.envx (release 1.0.0)

Modeler.dll – ResetIRRIGATION, ResetFireDISTURB, ResetCCdisturb, ResetHarvestDISTURB, ResetVEGTRNTYPE, ResetCropChoice, PVTdisturb

APs.dll – GetWeather, PrescribedLULCs, PopGrowth, IrrigationDecision, CropChoice, UrbanWater, FarmlandRent, LandUseTransitions, UGAexpansion, FishModel

Flow.dll – Flow

SpatialAllocator.dll – StandDisturbance

VegSTM.dll – VegSTM, WetlandSTM

Reporter.dll – YearlyReports

Daily processes, as in Flow.xml (release 1.0.0)

Flow.dll – reach routing, evap\_trans (ForestET, AgET, WetlandET), urban\_water\_demand, allocation (WaterRights), Spring (18 springs)

HBV.dll – HBVdailyProcess

MCfire.dll – CW3Mfire\_DailyProcess

## Some global method details

Near the beginning of the Flow XML file, there is a <global\_methods> block. Within that block, a variety of global methods are recognized; Flow.xml in CW3M ver. 335 specifies these global methods:

<reach\_routing>

<lateral\_exchange>

<hru\_vertical\_exchange>

<external>, used for HBV and MCfire

<evap\_trans>, for ForestET, AgET, and WetlandET

<urban\_water\_demand>

<allocation>

<Spring>, for 18 springs in the High Cascades

The source code for all the global methods except <external> ones is part of the Flow project in Visual Studios (VS). There are separate VS projects for HBV and MCfire.

Within the individual global method specification blocks, there is always a *name* field and often but not always *method* and *query* fields. A value of “none” for the *method* field indicates that the global method is not used; that’s the case for the <lateral\_exchange> and <hru\_vertical\_exchange> methods in Flow.xml in CW3M ver. 335.

The expression in the *query* field identifies the IDUs to which the method will be applied. The query is evaluated at the beginning of each simulation year. An integer array with one element for each IDU is used to record, for each global method, whether the IDU satisfies the global method’s query. Each bit in the integer corresponds to a different global method. Since integers are 32 bits wide, there is a limit of 32 global methods.

The fact that the query is evaluated just once per simulation year means that it is insensitive to day-to-day changes in attribute values. LULC attributes are commonly used in the queries. It would not make sense to use an attribute like PRECIP in a global method query, but PRECIP\_YR would make sense.

# Study areas

As of mid-2020, CW3M simulations can be executed for 13 watersheds:

Willamette River (Figure 1)

Tualatin River (Figure 2)

Chicken Creek (part of the Tualatin basin; Figure 5)

North Santiam River (Figure 3)

the upper Willamette basin (McKenzie + Middle Fork + Coast Fork; Figure 4)

Calapooia River

Clackamas River

Long Tom River

Luckiamute River

Marys River

McKenzie River

Molalla River

Pudding River

## DataCW3M directory structure

Installation of CW3M using a turnkey installer package such as CW3M\_0.1.4\_Installer.exe or CW3M\_McKenzie\_0.4.3.exe will result in the creation of a similarly named folder, e.g. “C:\DataCW3M\_0.1.4” or “C:\DataCW3M\_McKenzie\_0.4.3”, containing the data files necessary for execution of the “Demo” scenario by the model. The data folder contains the ENVX files for some or all of the study areas, a subdirectory for each such study area, and in addition some other files and folders common to all the study areas.

Each ENVX file specifies how to carry out the simulation of the particular study area, and the corresponding folder contains input and output files pertaining to the particular study area. For example, C:\DataCW3M\_1.0.0 may contain these files and folders:

Folders

Calapooia

ChickenCreek

Clackamas

CW3MdigitalHandbook

GriddedRecentWeather

LongTom

Luckiamute

Marys

McKenzie

Molalla

NSantiam

Observations

PEST\_BLU9

PEST\_Marys

Pudding

Reservoirs

ScenarioData

Tualatin

UpperWRB

Files

CW3M\_Calapooia.envx

CW3M\_ChickenCreek.envx

…

CW3M\_NSantiam.envx

CW3M\_PEST\_BLU9.envx

CW3M\_PEST\_Marys.envx

CW3M\_Pudding.envx

...

CW3M\_WRB.envx

Files and folders with PEST in their names pertain to simulations using PEST, the Model-Independent Parameter Estimation and Uncertainty Analysis used for finding values for some of the tunable model parameters.

## DataCW3M\ScenarioData directory structure

The ScenarioData directory contains folders for each scenario and some files which are common to multiple scenarios:

Folders for each scenario

Baseline

Demo

Files common to multiple scenarios

APs.xml

cooling\_cost.csv

cropchoice.csv

Crops.csv

deterministic\_transition\_lookup.csv

HBV.csv

HRU.xml

IDU.xml

Reach.xml

Reporter.xml

SpatialAllocator.xml

VegSTM.xml

wr\_pods.csv

wr\_pous.csv

## DataCW3M\ScenarioData\<scenario> folder contents

Scenario folders contain data files which are specific to individual scenarios.

## DataCW3M\<study area> folder contents

The folder for a particular study area contains the IDU, HRU, and Reach shapefiles for that study area and may contain other folders and files specific to the study area. Generally, a file specific to a study area will use the study area name as a suffix, e.g. Flow\_NSantiam.xml. The suffix comes before the file type.

## 

![A close up of a map

Description automatically generated]()

Figure 1. Willamette River basin study area; embedded study areas are colored other than gray. The 2 lighter blues together make up the upper Willamette basin study area.

![A screenshot of a video game

Description automatically generated]()

Figure 2. Tualatin basin study area; Chicken Creek watershed is highlighted.

![A picture containing screenshot

Description automatically generated]()

Figure 3. North Santiam watershed study area.

![A screenshot of a map

Description automatically generated]()

Figure 4. Upper Willamette study area.

![A close up of a map

Description automatically generated]()

Figure 5. Chicken Creek watershed study area.

# Calendar conventions

CW3M supports the standard calendar with leap years, and in addition supports a calendar consisting of uniform 365-day years with no leap years or leap days. The uniform 365-day calendar convention is included as an option for compatibility with older climate datasets lacking leapdays.

CW3M defaults to the standard calendar. The uniform 365-day option can be invoked by including

maxDaysInYear='365'

in the <settings> block of the ENVX file. The default setting is

maxDaysInYear='366'

Hadley Climate Center in the U.K. produced climate datasets with uniform 360-day years in the past, but has now switched to more standard calendar conventions. Hadley’s 360-day climate datasets are not supported by CW3M.

For debugging purposes, CW3M allows maxDaysInYear to be set to 1 or 0.

## Water years

CW3M supports reporting some output data on a water year basis, instead of a calendar year basis. Water years start on October 1st and end on September 30th; they are named the same as the calendar year in which they end.

Reporting by water year is controlled by a field in the <settings> block of the ENVX file:

useWaterYears='1'

The default setting is

useWaterYears='0'

Reports specified in the Flow XML file are affected by the useWaterYears setting.

# Climate data

## Numbered climate scenarios in the model

The Flow.xml file contains a numbered list of climate datasets available within the model. As of October 2020, the list has 13 entries which are consistent with the corresponding lists in the WW2100, OUWIN, and CW3M projects. 1 identifies the numbered climate scenarios.

Table 1. Numbered climate scenarios

|  |  |  |  |
| --- | --- | --- | --- |
| climate scenario number | date range | name; source of data | project, simulation scenario first used for |
| 0 | 1950-2100 | MIROC; MIROC5 from MACA v1 | WW2100 Reference |
| 1 | 1950-2100 | GFDL; MACA v1 | WW2100 LowClim |
| 2 | 1950-2099 | HadGEM; MACA v1 | WW2100 HighClim |
| 3 | 1950-2100 | WW2100StationaryClimate; MIROC5 from MACA v1 | WW2100 Stationary Climate |
| 4 | 1979-2011 | MACAtrainingData; MACA v1 | WW2100 PEST |
| 5 | 2000-2017 | BaselineGrid; MACA training data | INFEWS development |
| 6 | 2000-2017 | BaselinePoly; BaselineGrid interpolated to IDU centroids by SnowModel | INFEWS Baseline |
| 7 | 2010-2017 | GriddedRecentWeatherForDemos; subset of Baseline\_2019versionOfGridMET\_2010-curren | INFEWS Demo |
| 8 | 2010-2019 | Baseline\_2019versionOfGridMET\_2010-current – from 2019 version of gridMET, representing daily PRISM bias-corrected to monthly PRISM, but on a slightly different 4-km grid than PRISM | CW3M Baseline from PRISM |
| 9 | 1950-2005 | MIROC5\_macav2; macav2metdata MIROC5 r1i1p1 20th century | CW3M MIROC5\_20th\_century |
| 10 | 2006-2099 | MIROC5\_rcp85; macav2metdata  MIROC5\_r1i1p1\_rcp85 | CW3M MIROC5\_rcp85 |
| 11 | 1950-2005 | HadGEM2-ES\_macav2; HadGEM2-ES365\_r1i1p1\_20th century | CW3M HadGEM-ES\_20th\_century |
| 12 | 2006-2099 | HadGEM2-ES\_rcp85; macav2metdata\_HadGEM2-ES365\_r1i1p1\_rcp85 | CW3M HadGEM-ES\_rcp85 |
| 13 | 2000-2009 | "Baseline\_2019versionOfGridMET\_2000-09" from 2019 version of gridMET, representing daily PRISM bias-corrected to monthly PRISM, but on a slightly different 4-km grid than PRISM | CW3M Baseline from PRISM |
| 14 | 1979-2018 | BaselineGridMultiyearFiles; old gridMET historical data from before 2019 (aka METDATA) which was used as training data for the MACA bias correction | CW3M Baseline from old gridMET |

## Monthly and seasonal weather data

Several model components pertaining to the agricultural basin use monthly and seasonal weather data (e.g. crop choice and farm rent). The monthly and seasonal weather values are aggregated up from the daily weather values in the input climate dataset. The monthly and seasonal weather values are held as IDU attributes.

There are 7 attributes which represent values for months or seasons in the current year: TMIN\_GROW, PRCP\_GROW, PRCPWINTER, PRCPSPRING, PRCP\_JUN, PRCP\_JUL, PRCP\_AUG. For these calculations, the growing season is defined as April through October. Winter is January through March, and spring is April through June.

There are 5 attributes which represent multi-year averages: TMINGROAVG, PRCPGROAVG, PRCPJUNAVG, PRCPJULAVG, PRCPAUGAVG. Multi-year averages are calculated using exponential averaging:

multi-year avg. = prev. avg. \* e-1/tau + curr. yr. \* (1 – e-1/tau)

where tau defaults to 10 but can be specified in the ENVX file with the tag, “yearsInStartingClimateAverages” in the <settings> block.

The initial condition previous average values used at the beginning of simulation in the exponential average calculation are themselves calculated when the simulation is run for zero years (i.e. in the FlowModel::InitRun() subroutine), using climate data for the “tau” number of years just prior to the “reference start year”. The reference start year defaults to 2010, but can be set explicitly using the “referenceStartYear” tag in the <settings> block of the ENVX file.

There is no explicit definition of the boundaries of the agricultural basin. The monthly and seasonal weather data is calculated for the entire study area.

## Climate data grids

The climate data grid has a lattice spacing of 2’ 30” of arc in both latitude and longitude. It consists of 54 columns from west to east, and 66 rows from south to north. Two slightly different versions of this grid have been used. Climate scenarios 0 through 7 use climate datasets from the WW2100 project, structured in many relatively small NetCDF files each containing daily data for a single climate field for a single year, e.g. daily precipitation for 2010. Climate scenarios 8 through 12 use “v2” datasets, structured in a small number of much larger NetCDF files each containing data for a single climate field for a range of years, e.g. daily precipitation for the years 2006-2099.

Note that, although the climate grid is in units of arc (degrees of latitude and longitude), the GIS layers (IDU, HRU, and Reach shapefiles) use UTM Zone 10N coordinates in units of distance (meters).

### The WW2100 climate grid

The WW2100 grid locates the row 0, column 0 gridcell in the northwest corner of the grid. In the WW2100 grid, the centers of the corner grid cells are located at

center of northwest corner gridcell 46 deg 01’ 15” N, 123 deg 48’ 45” W

center of southwest corner gridcell 43 deg 18’ 45” N, 123 deg 48’ 45” W

center of northeast corner gridcell 46 deg 01’ 15” N, 121 deg 36’ 15” W

center of southeast corner gridcell 43 deg 18’ 45” N, 121 deg 36’ 15” W

Data for each day begins with the data for the NW gridcell, designated as row 0, column 0, the grid origin. Column numbers increase from west to east, from 0 through 53, before the row number increases from 0 to 1. Row numbers increase from north to south.

This WW2100 grid is a subgrid of a grid whose southern edge is the equator, northern “edge” is 90 deg N, eastern edge is 0 deg of longitude, and western edge is at 180 deg of longitude. The edges of the subgrid are located at:

northern edge 46 deg 02’ 30” N

southern edge 43 deg 17’ 30” N

eastern edge 121 deg 35’ 00” W

western edge 123 deg 50’ 00” W

### The v2 climate grid

The v2 grid locates the row 0, column 0 gridcell in the southwest corner of the grid, and is offset from the WW2100 grid by 15 arc-seconds to the north and 15 arc-seconds to the east. In the v2 grid, the centers of the corner grid cells are located at

center of northwest corner gridcell 46 deg 01’ 30” N, 123 deg 48’ 30” W

center of southwest corner gridcell 43 deg 19’ 00” N, 123 deg 48’ 30” W

center of northeast corner gridcell 46 deg 01’ 30” N, 121 deg 36’ 00” W

center of southeast corner gridcell 43 deg 19’ 00” N, 121 deg 36’ 00” W

Data for each day begins with the data for the SW gridcell, designated as row 0, column 0, the grid origin. Column numbers increase from west to east, from 0 through 53, before the row number increases from 0 to 1. Row numbers increase from south to north.

## How climate data is looked up

Each IDU is associated with the climate grid cell in which the IDU’s centroid is located, and similarly with HRUs. The IDU layer has an attribute named GRID\_INDEX which identifies the climate grid cell associated with the IDU polygon. The IDU, HRU, and Reach layers all have attributes for the weather fields: precipitation, tmin, tmax, humidity, etc. Data is read from the gridded climate data files in each daily time step and used to populate the IDU attributes for the weather fields. The HRU weather field values are the area-weighted averages of the IDUs in the HRU. The Reach weather fields are the values for the IDU which contains the reach’s vertex 0, the upstream end of the reach.

C++ methods involved in looking up climate data include:

FlowModel::OpenClimateDataFiles(year), called at the beginning of each simulated year

FlowModel::GetDailyWeatherField(day, year, field)

ClimateDataInfo::GetTimeIndex(day, year) calculates the location in the file of the day’s data

FlowModel::InitClimateMeanValues(), called in InitRun(), but only when yearsToRun = 0.

FlowModel::InitGridIndex(), called in InitRun(), but only yearsToRun = 0.

## What is in a climate dataset

Climate data is supplied at a temporal resolution of 1 day. The spatial resolution is one climate grid cell. An alternative spatial resolution of IDU polygons is also available; it requires preprocessing the gridded climate data to interpolate spatially to IDU centroids. The required and optional weather fields are identified below. Within parentheses, common field variable names are shown in lower case, and associated attribute names are shown in upper case.

These fields are required:

daily precipitation, mm H2O (pr, precip, PRECIP)

diurnal minimum temperature, deg C or deg K (tasmin, tmin, TMIN)

diurnal maximum temperature, deg C or deg K (tasmax, tmax, TMAX)

solar radiation, W/m2 (rsds, solrad, RAD\_SW)

Humidity is also required, but can be supplied either as

specific humidity, kg H2O/kg air (sph, humidity, SPHUMIDITY)

or

relative humidity, %

If both are supplied, specific humidity is used. As of May 2019, the ability to use relative humidity climate data instead of specific humidity data has not yet been implemented.

Relative humidity may be supplied either as a daily value or as diurnal extremes:

diurnal minimum relative humidity, % (rhmin)

diurnal maximum relative humidty, % (rhmax)

Daily mean temperature may either be supplied or calculated.

daily mean temperature, deg C or deg K (TEMP)

If daily mean temperature is not supplied, then it is calculated as needed as the average of the diurnal extreme temperatures.

Windspeed is used, but if not supplied is set to a constant 1 m/s. Windspeed may be supplied directly as

windspeed, m/s (ws, windspd, WINDSPEED)

or may be calculated as needed from the components of the wind vector if they are supplied:

eastward wind, m/s (uas)

northward wind, m/s (vas)

# Water rights

## Water rights data

The water rights module reads two .csv files at initialization. One, wr\_pods.csv, is a list of points of diversion (PODs). The other, wr\_pous.csv, is a list of points of use (POUs). These files were created for the WW2100 project in the summer of 2015 by Andrea Laliberte, a consultant working for Bill Jaeger (OSU Applied Economics Dept.). The OWRD water rights database is the original source data from which the POD and POU files were derived. A number of small changes have since been made to the files to improve their accuracy and correct typos. The two .csv files are read and interpreted by the AltWaterMaster::LoadWRDatabase() method. With minor exceptions (e.g. when PODRATE is zero), LoadWRDatabase() creates one record in the internal water rights array for each record in the wr\_pods.csv file. As in the OWRD database itself, there can be multiple records in the wr\_pods.csv file and in the internal water rights array with the same value of the WATERRIGHTID field, representing multiple points of diversion associated with a single WATERRIGHTID value.

Points of diversion for surface water sources are associated in the model with stream reaches (REACHCOMID in the wr\_pods.csv file). Points of diversion for groundwater sources are specified by their coordinates (X and Y in the wr\_pods.csv file), but groundwater POD locations don’t matter to the model, since the model treats groundwater as physically unlimited. Water rights to groundwater used for municipal and irrigation purposes are included in the two .csv files. The amount of water used under those water rights is tracked by the model.

The OWRD database uses a variety of permit codes to characterize points of diversion, but the model only recognizes water rights with the WRP\_SURFACE or WRP\_GROUNDWATER permit codes. The WRP\_RESERVOIR permit code is defined in the source code (it has the value ‘8’), but is never referenced.

CW3M assumes that the watermaster will never cut off a municipal water supply. To track events which might otherwise lead to cutting off a municipal water supply, imaginary backup water rights for municipal water systems in the 8 largest urban growth areas (Metro, Eugene-Springfield, Salem-Keiser, Corvallis-Philomath, Albany, Newberg, Woodburn) are represented in the wr\_pods.csv file. These rights have WATERRIGHTID values of >= 200000 and PODRATE value of 1000 cfs, with priority dates of 1/1/2010. Use of these backup water rights is recorded in the “ALTWM Annual Muni Backup Water Right Use” report.

Source code which reads and interprets the wr\_pods.csv and wr\_pous.csv files is located in two subroutines (LoadWRDatabase() and CalculateWRattributes()), which are both methods of the AltWaterMaster class and located in the WaterRights.cpp source file.

## The wr\_pods.csv file

wr\_pods.csv is a comma-separated-values text file with the columns listed below. The model code requires the file to be sorted in order of priority date, from earliest date to latest date. During the simulation, rights having priority dates later than the current simulation date are ignored. For code values in the PERMITCODE, USECODE, AND SPECIAL columns, see the “Water Right Code Values” tab in CW3Mhandbook.xlsx.

WATERRIGHTID – “WaterRightId” in OWRD’s Water Right Information System

X – easting of point of diversion

Y – northing of point of diversion

PODID – point of diversion ID number assigned by CW3M

POUID – if > 0, ID number of point of use of water from this point of diversion; if < 0 and > -99, water from this point of diversion is used in a municipal water system, and the absolute value of POD is the urban growth area ID number; if = -99, this is an instream water right.

PERMITCODE – code for type of diversion. As of August 2021, the model recognizes only 2 permit codes: surface water (2) and groundwater (4).

PODRATE – maximum diversion rate, cfs.

USECODE – code for type of use (municipal, irrigation, instream, …)

from FlowInterface.h in CW3M repository version 461

enum WR\_USE { ... WRU\_UNKNOWN=1, WRU\_IRRIGATION=16, WRU\_MUNICIPAL=1024, WRU\_INSTREAM=2048, ... };

PRIORITYDOY – priority date day of year (Jan 1 = 1)

YEAR – priority date year

BEGINDOY – day of year on which the right goes into effect

ENDDOY – last day of year on which the right is in effect

REACHCOMID – for surface water rights COMID of reach from water is diverted

LENGTH\_OR\_COMID – for instream water rights, either the length of stream reach to which the right applies, or the COMID of the downstream end of the portion of the stream to which the right applies

SPECIAL – special codes used to identify how certain water rights should be handled in different scenarios (e.g. a value of 1 indicates an unconverted instream water right)

from WaterMaster.h in CW3M repository version 461

enum WR\_SPECIALCODE { ... WRSC\_NEWINSTREAM = 1, WRSC\_MUNIBACKUP = 2, WRSC\_NOT\_NEWINSTREAM\_ONLY = 4 ...}

WRSC\_NOT\_NEWINSTREAM\_ONLY indicates a water right that only applies when the new instream water rights (the ones with WRSC\_NEWINSTREAM) are not in effect. Whether or not to use the new instream water rights is an input variable which can be turned on or off in the SimulationScenarios.xml file.

CERTIFICATE – For certificated water rights, the certificate number, or other note if the entry does not represent a certificated water right, or -99 if the data has not yet been entered.

## The wr\_pous.csv file

wr\_pous.csv is a comma-separated-values text file with the columns listed below.

POU\_INDEX – index of data record in wr\_pous.csv (0, 1, 2, …)

POU\_ID – same as for wr\_pods.csv

IDU\_ID – identifies the IDU in which the point of use is located

WRIS\_ID – “WaterRightId” in OWRD’s Water Right Information System

AREA\_POU – area of the POU, m2

PERCENT\_POU – per cent of the POU overlapped by the IDU

AREA\_IDU – area of the IDU, m2

PERCENT\_IDU – per cent of the IDU overlapped by the POU

USECODE – same as for wr\_pods.csv

PERMITCODE – same as for wr\_pods.csv

XCOORD - easting of IDU centroid

YCOORD – northing of IDU centroid

CERTIFICATE – For certificated water rights, the certificate number, or other note if the entry does not represent a certificated water right, or -99 if the data has not yet been entered.

Note that the POU\_INDEX values are sequential and unique, but POUID values are not necessarily unique, reflecting the fact that water from a single point of diversion may go to multiple points of use. Before the POU\_INDEX values are added, the file is sorted by POUID and IDU\_ID.

The POU file includes an IDU\_ID field, which is used to identify the IDU in which the point of use is located. In future, this issue could be handled by adding two columns containing the x,y coordinates (easting, northing) of the point of use to the POU file. Logic could then be added to the cold start process in OUWIN to populate the IDU\_ID column based on the location of the POU.

## Allocating surface water rights to irrigation

Surface water is allocated for irrigation by water right in a method of the AltWaterMaster class. The method is called AllocateIrrigationWater() and has 4 inputs: a pointer to the water right record, the index of the IDU for which the water is being allocated, the fraction of the POU which overlaps the IDU, and the available flow at the point of diversion. A target irrigation rate for the IDU for the current day has already been calculated elsewhere in the model when AllocateIrrigationWater() is called, and is passed into the method in AltWaterMaster::m\_iduLocalIrrRequestArray[iduNdx]. AllocateIrrigationWater() returns the amount allocated and the remaining amount available in cms. The source code is in the WaterRights.cpp file. Allocation of groundwater for irrigation is handled in a different method, AllocateWellWR(), with somewhat different logic, and is not covered in this section.

An upper limit to the allocation is read from the PODRATE field of the water right record (obtained from the PODRATE column of the wr\_pods.csv file). The limit is adjusted downward by subtracting off whatever has already been allocated in the current day from the water right record.

If the water right has already been used this year to allocate an amount equal to the IDU’s maximum duty, then no further water is allocated under that water right to that IDU in the current year. The IDU’s maximum duty is the product of the IDU attribute XIRR\_DUTY and the model’s overall maximum duty. The overall maximum duty defaults to 2.5 feet per year, but can be set explicitly as an input variable to the Flow autonomous process.

The model also has an overall maximum irrigation rate in cfs per acre, which defaults to 0.0125 cfs per acre (= 1/80 cfs per acre = 7.5 mm/day). It too can be set explicitly as an input variable to Flow. However, the code which uses the overall maximum irrigation rate is commented out, so this limit is not currently applied by the model.

## Adding a water right

To add a water right to the wr\_pods.csv and wr\_pous.csv files, follow these steps.

1. Determine the highest value in the PODID field of the wr\_pods.csv file; assign the next higher value as the PODID value for the new water right.

2. If the new water right is an instream water right, set its POUID to -99. If it is for a municipal water system, set its POUID = - UGB. Otherwise, determine the highest value in the POUID field of the wr\_pous.csv file; assign POUIDs to the new water right starting with one greater than that highest value.

3. If the new water right does not correspond to an entry in the OWRD database, assign a WATERRIGHTID to it equal to the next unassigned number in the series of WATERRIGHTIDs which begin with 200000. For example, a water right corresponding to a 1945 judicial decree, which is not represented in the OWRD database, is assigned a WATERRIGHTID value of 200084 in the CW3M PODS file.

4. Add an appropriate point-of-diversion record to the end of the wr\_pods.csv file. Note that for groundwater rights, REACHCOMID and LENGTH\_OR\_COMID should be set to -99. For an instream water right, COMID should be set to the reach at the upstream end of the water right; LENGTH\_OR\_COMID should be set either to the length along the stream in feet to which the right applies or to the COMID of the reach at the downstream end of the right; the X and Y coordinates, if unknown, should be set to a point on or near the upstream reach.

5. Insert the new point-of-diversion record(s) into the wr\_pods.csv file in sorted order, or sort the entire wr\_pods.csv file by priority year and day and the portion of the year for which the right is in effect (sort by YEAR, PRIORITYDOY, BEGINDOY, ENDDOY, WATERRIGHTID, X, Y, PODID). This completes the changes to the wr\_pods.csv file. If the new water right is for a municipal water system, no changes are necessary to the wr\_pous.csv file. Otherwise, continue with the next step for additions to the wr\_pous.csv file.

6. At the end of the wr\_pous.csv file, add one record for each point of use. Sort by POUID. Recalculate the POU\_INDEX column. This completes the changes to the wr\_pous.csv file.

# Initial conditions

## State variables

CW3M is a space-before-time model. The simulation is advanced one day at a time by calculating the changes across the space of the study area for that particular day. Inputs to those calculations include the states of the IDUs, HRUs, and reaches in the previous day plus the weather for the current day plus the outputs from various submodels, for example, the amount of water withdrawn from stream reaches for municipal use.

## Reproducibility and the use of random number generators

CW3M results are reproducible, provided care is taken to begin the simulations with the same initial conditions. Given the goal of reproducibility, It may seem counterintuitive that CW3M makes use of random number generators in some of its submodels, for example in the Spatial Allocator process as when placing wildfires on the landscape. The apparent conflict between reproducibility and the use of random numbers is partially resolved by control over the value of the seed for the random number generator. The random number generator itself produces a fixed but effectively infinite sequence of “random” numbers; the seed determines where in the fixed sequence the first value is chosen. By default, CW3M always starts from the same seed, but it also provides a way for the user to set the value of the seed. As of May 2021, current work with CW3M has not made use of the option of specifying the value of the random number seed.

Since the random number seed can affect the results of the simulation, the value of the seed is, properly considered, a state variable. However, as of May 2021, there is no mechanism for saving the value of the random number seed; the model’s reproducibility relies on the default behavior of always starting from the same value of the seed.

## Initializing ID numbers: coldstart

Each spatial unit of computation – IDUs, HRUs, and reaches – has an identification number associated with it. The ID numbers are not, in general, the same as the index to the element in it GIS shapefile. For reaches, the ID number is the COMID of the reach in the National Hydrography Dataset, an 8-digit integer. CW3M itself constructs the ID numbers of IDUs and HRUs.

ID numbers are intended to remain the same across all subsets of the Willamette River Basin study area. For example, the ID number of the IDU where the outfall from the Detroit reservoir is located is 84542, both in the full WRB IDU shapefile and in the smaller IDU shapefile for the North Santiam River watershed. In order for that to be the case, the ID numbers must be assigned by CW3M at the time it runs for the full WRB rather than for any smaller region within it. The process of assigning the ID numbers is part of “coldstart”, i.e. starting up CW3M with default values for all its state variables. There is a coldstart=’0’ field in the <settings> block of the CW3M\_WRB.envx file; changing it to coldstart=’1’ causes CW3M to perform a coldstart.

## Initializing Flow and HBV model state variables: spinup

Daily output from the flow model, which includes reach discharges from reaches, reservoir water volumes, and soil moisture conditions, depend both on the current day’s weather and on the flows and water volumes on the previous day, collectively known as the Flow and HBV model state variables. Spinup starts all these state variables out at prescribed default values. The values of the state variables then evolve from one daily time step to the next. Current practice (as of May 2021) with CW3M is to run a spinup simulation using actual weather data over the ten years 2000-09, and then save the state variables for use in starting additional simulations in 2010.

There is a spinup=’0’ field in the <settings> block of all the CW3M ENVX files, both for the entire WRB and for all its smaller study area watersheds. Changing it to spinup=’1’ causes CW3M to perform a warmstart.

## Initializing HBVCALIB attributes: spinup

HBVCALIB parameter values, held in the HBV CSV file, are used to calibrate the HBV precipitation:runoff submodel, the runoff water temperature to air temperature submodel, and the evapotranspiration fraction of precipitation.

HBVCALIB, essentially a row number in the HBV CSV file, is an attribute of all three GIS layers (IDU, HRU, and Reach). The HBVCALIB value associates a specific row of parameter values in the HBV CSV file with the particular spatial unit. The process of setting the HBVCALIB values is, like the initialization of the Flow and HBV state variables, part of spinup. HBVCALIB areas are defined by the HBVcalibPts array in the WaterRights.cpp source file, by specifying the location of their pourpoints. The HBVCALIB attribute values in the IDU, HRU, and Reach layers are calculated and stored by CW3M during spinup.

The value of HBVCALIB, a small positive integer, determines which row of the HBV.csv file holds the parameter values for the spatial unit. Spatial units with the same HBVCALIB value drain to the same pourpoint. Areas and portions of the stream network which share a single HBVCALIB value are assigned first at the headwaters and then downward in the stream network to the WRB outlet. Each HBVCALIB area is either a complete a watershed in itself or is the downstream end of a watershed formed by itself together with one or more upstream HBVCALIB areas. For example, the DET12 HBVCALIB area plus the Blowout51 HBVCALIB area forms a complete watershed with its pourpoint at the reach (COMID 23780511) where the Niagara gage (USGS 14181500) is located, downstream of the Detroit dam. DET12 has HBVCALIB value 12. Blowout51 is the complete Blowout Creek watershed, which has HBVCALIB value 51.

## Initial conditions for Flow: the IC file and the IDU, HRU, and Reach shapefiles

The Flow model has the ability to read initial values for the water in the stream network from an IC file. The IC file is named “flow<year>.ic” where <year>, e.g. “2010”, is the calendar year to which the initial conditions apply. The IC file is located in the same folder as the IDU layer. When the value of the spinup field in the ENVX file is 1, the IC file is ignored.

Except during a spinup, when the IC file is not specified or cannot be accessed, a warning message is issued. At the beginning of a spinup and whenever the IC file is inaccessible, Flow initializes with nominal water amounts and flows; customarily, a 10-year simulation from that point is used for spinup. On completion of every simulation run, CW3M saves a new IC file with the values from the end of the run, to the user’s Documents folder. The name of the newly saved file is simply the calendar year to which it applies, for example, “2010”. To be used in a subsequent simulation, the file must be moved or copied to the folder where the IDU layer is located and renamed in the required format: “2010” would be renamed as “flow2010.ic”.

As of May 2021, the Flow and HBV models have some state variables which are not saved in the IC file; these extra state variables are, however, present as attributes in the IDU, HRU, and Reach shapefiles. For that reason, to save the state of the simulation at the end of a spinup, it is necessary to save the IDI, HRU, and Reach shapefiles in addition to the IC file.

## What is in the IC file

The IC file is a binary file consisting of a long series of C++ *double* values, i.e. double precision floating point numbers, conceptually representing the state variables of the Flow model as applied to the current IDU, HRU, and Reach shapefiles. The first value is the number of state variables, *modelStateVarCountWP*, the sum of

*hruCountWP \* hruLayerCount + subreach\_countWP + reservoir\_countWP*

The “WP” suffix in these C++ variable names indicates that they apply to model versions which use *WaterParcel* objects to track water temperature as well as water volume. The various ...*CountWP* variables take into account how many values are saved for each kind of object. For instance, for each subreach, 3 values are saved: discharge (cms), volume (m3), and temperature (°C). For each reservoir, 2 values are saved: volume (m3), and temperature (°C). HRUs have 6 “layers”, i.e. compartments in the HBV precip/runoff model. For each layer of each HRU, 2 values are saved: volume (m3), and temperature (°C).

## State variables in the IDU shapefile

The HBV model has 6 compartments for soil and surface water. One compartment is used for 2 purposes: to hold the volume of liquid water contained in a snowpack, and to hold the volume of water standing on the surface of a wetland. An HRU may have both wetland IDUs with standing water and non-wetland IDUs with snow, but any single IDU can only have standing water or snow, or neither; an IDU cannot have both snow and standing water at the same time. The LULC\_A attribute determines whether an IDU is a wetland or not. Wetland IDUs may have standing water or not; the WETNESS attribute value if positive represents the depth of the standing water. Non-wetland IDUs may have snow or not; the H2O\_MELT attribute represents the amount of liquid water in the snow, expressed as a depth. The LULC\_A, WETNESS, and H2O\_MELT attributes are state variables of the simulation.

## Initial conditions for HBV: the IC file and the IDU shapefile

HRUs which include cropland or wetland IDUs will vary in the amount of water from IDU to IDU. Cropland IDUs use water at different rates depending on which crop is being grown. Wetland IDUs may be inundated to different depths, or may have standing water at the same time upland IDUs have snow cover. Some IDU attributes are used to keep track of these different states: SNOWPACK, H2O\_MELT, WETNESS, and SM\_DAY. The values of those attributes at the time a simulation begins also count as initial conditions, even though they are not included in the IC file. Those IDU values and the HRU layer values are combined to set the initial state of the HBV model.

SNOWPACK (mm) is the liquid water equivalent depth of the frozen part of the snow on the ground. H2O\_MELT (mm) is the liquid water present in the snowpack. SM\_DAY (mm) is the soil moisture in the part of the soil accessible to plant roots. WETNESS (mm) is an attribute defined for wetlands. When WETNESS is zero or positive, it signifies that the soil is saturated and may be inundated; the numeric value is the depth in mm of standing water. When WETNESS is negative, its absolute value is the amount of water in mm which would have to be added to the soil to bring it back up to its field capacity.

As of April 2021, CW3M ignores the difference between field capacity and saturation. This simplification is necessitated by the absence of data on saturation from the HBV parameter set. However, since the extra water held by the soil at saturation is just the amount expected to drain out by gravity flow in 1 day, and given that the timestep of the model is also 1 day, we expect that glossing over the difference between field capacity and saturation will introduce an error in the depth of inundation, as represented by positive values of WETNESS, limited to no more than saturation water capacity minus field capacity.

When CW3M first initializes, if an IC file is not available, CW3M initializes the compartments in the HBV model and the corresponding IDU attributes to a value specified in the init\_water\_content field of the <catchments> block of the Flow XML file. The current setting for the init\_water\_content is the single value 0.1, which results in values of 0.0 for SNOWPACK and H2O\_MELT, a value in m3 equivalent to 40 mm for SM\_DAY, and a negative value of WETNESS equal to 40 mm minus the field capacity of the soil, obtained from the HBV parameter file.

If on the other hand an IC file is available, then CW3M constructs the HRU values for snowpack, surface water, and root-accessible compartments of the HBV model by aggregating the values of the corresponding IDU attributes across the IDUs which make up each HRU. Warning messages are produced when the HRU values in the IC file are inconsistent with those aggregated from the IDU attributes.

## Default values used when the data is incomplete

Ideally, the Reach layer would contain information about the characteristic width and summer low flow of each reach, and about the height and density of the streamside vegetation. These Reach attributes are there to hold that information: WIDTHGIVEN, Q\_MIN, VEGHTGIVEN, and VGDNSGIVEN. For portions of the stream network for which there is Shade-a-lator data available, CW3M can estimate the values of WIDTHGIVEN, VEGHTGIVEN, and VGDNSGIVEN from the Shade-a-lator data. As of May 2021, Shade-a-lator data is only available in CW3M for a small part of the McKenzie basin. Everywhere else, nominal default values are calculated and used.

|  |  |  |
| --- | --- | --- |
| attribute used in the simulation | attribute for observed value | attribute for default value |
| WIDTHREACH | WIDTHGIVEN | WIDTH\_MIN |
| Q\_MIN |  | Q\_MIN |
| VEGHTREACH | VEGHTGIVEN | VEGHT\_CALC |
| VGDNSREACH | VGDNSGIVEN | LAI |

The default value for WIDTH\_MIN in meters is the square of the order of the reach (headwater reaches have order 1, and order increases going downstream).

There is a symbol defined for the minimum flow in the smallest reaches:

#define NOMINAL\_LOW\_FLOW\_CMS 0.010 /\* 10 liters of water per sec \*/

This value was chosen partly so as to prevent mathematical issues in the flow calculations. For reaches of order 2 and larger, the Q\_MIN value in cms defaults to (order – 1)3.

VEGHT\_CALC is an estimate of streamside vegetation height based on the VEGCLASS attributes of streamside IDUs.

LAI represents the leaf area index of the streamside vegetation based on the LAI attributes of the streamside IDUs. Beers Law is used to calculate vegetation density from the leaf area index. The Beers Law coefficient is defined as a symbol:

#define BEERS\_LAW\_K 0.5

As of May 2021, there are 11 Reach attributes which are initialized during spinup to a non-physical token value (-1), to indicate that observational data is not available: DEPTH\_MIN, WIDTH\_MIN, WIDTHGIVEN, VGDNSGIVEN, VEGHTGIVEN, VEG\_HT\_L, VEG\_HT\_R, TOPOELEV\_E, TOPOELEV\_S, TOPOELEV\_W, and RADSWGIVEN.

# Stream flow and stream temperature

## Water parcels

Water flows continuously. Digital computers use discrete quantities to represent, approximately, the continuous elements and properties of the real world. CW3M accounts for the water on a landscape as “parcels”, represented in the C++ code as objects of the class WaterParcel. The properties of a water parcel, represented in C++ as members of the WaterParcel class, are

volume, expressed in units of cubic meters

temperature, deg C

The volume property is used as a surrogate for mass, at a constant density of 998.2 kg per cubic meter. This approximation reflects the fact that observational data for water in streams and water bodies is always in units of volume, and the convenient reality that the density of liquid water is relatively constant over the range of conditions encountered in natural environments. In the future, the WaterParcel class could be extended to include other properties, such as sediment load and dissolved pollutants.

In CW3M, spatial units have water parcels associated with them. Each subreach in the stream network has a parcel of water in it, whose properties change from one daily timestep to the next. The total volume of a reach is the sum of the subreach volumes. Landscape polygons (“IDUs”) and bodies of water also have associated water parcels.

## Daily water mass and energy balance

Thermal energy algorithms are being added to CW3M in 2020 as part of the McKenzie Basin Wetlands Study, described in a later chapter of this document. Inspiration and technical information for the thermal energy code is from the 204-page manual for Heat Source Model Version 7.0, prepared by Matthew Boyd and Brian Kasper in 2003 and 2007 for the Oregon Department of Environmental Quality (Boyd & Kasper 2003).

CW3M estimates the properties of water parcels at a daily timestep. The basic equation for daily subreach volume is straightforward

Vt = Vt-1 + Vup + Vlateral - Vdown - Vevap + Vprcp

where (all quantities in m3)

Vt = volume in day t

Vt-1 = volume in the previous day

Vup = volume flowing in from upstream

Vlateral = volume entering (+) or leaving (-) the subreach through the stream banks or streambed, or from withdrawals for irrigation or municipal use, or discharges from municipal points of central discharge

Vdown = volume flowing out to downstream

Vevap = volume lost to evaporation

Vprcp = volume gained from precipitation falling on the water surface

The equation for subreach thermal energy has additional terms

Et = Et-1 + Eup + Elateral – Edown + ESW – ELW - Eevap + Eprcp

where (all quantities in kJ)

Et = thermal energy in day t

Et-1 = thermal energy in the previous day

Eup = thermal energy in the water flowing in from upstream

Elateral = thermal energy entering or leaving in water from stream banks, withdrawals for irrigation or municipal use, groundwater exchange, etc.

Edown = thermal energy leaving in water flowing out of the reach

ESW = incoming shortwave solar radiation

ELW = outgoing longwave radiation

Eevap = energy carried away from the reach through evaporation

Eprcp = energy entering via precipitation

There are corresponding equations with somewhat different terms for landscape polygons and bodies of water. The equations attempt to account for first order effects; they neglect effects which are usually (but not always) of lesser magnitude: the conversion of mechanical energy to thermal energy, convective heat exchange with the air and the streambed, and so on.

Thermal energy density (i.e. temperature) affects the rate of evaporation and the outgoing longwave radiation. As a result, the mass and energy balance calculations are interrelated; the estimate of thermal energy is made in the same subroutine which calculates the flow in the reach, so that the flow to downstream can be corrected for the loss of water to the atmosphere by evaporation. Precipitation on the surface of the stream also affects volume and thermal energy. The amount of evaporation lost and the amount of precipitation received are proportional to the surface area of the stream reach or water body.

Note in connection with evaporation from stream reaches that the model tiles the land surface with IDU polygons, and treats the stream network as a set of lines with no area of their own. Water bodies – reservoirs and lakes – are represented in the IDU layer. This makes it difficult to track precipitation which falls on stream reaches, because that precipitation has already been accounted for in the amount simulated as falling on the IDUs that the reach traverses. In effect, Vprcp is included in Vlateral. Eprcp contributes to Elateral, but an error is introduced inasmuch as in reality the thermal energy of precipitation falling directly on the stream water surface is more closely correlated to the temperature of the air than to the temperature of the soil.

## Estimating the rate of flow in a stream reach

CW3M uses a kinematic wave algorithm to estimate the average daily flow rate in each subreach and reach. The reaches themselves are organized as a tree: 1) headwater reaches are “leaves”; 2) all other reaches may have either one or two upstream reaches (“branches”) which drain into them; 3) each reach drains into a single next reach just downstream. Internally, the model holds an array of pointers to reach objects (class FlowModel, member m\_reachArray[]) which is ordered such that for any given reach all the reaches which drain directly or indirectly into that reach come before it in the array, that is, the array starts with the leaves and ends with the root.

Each individual reach is represented as an ordered set of subreaches (class Reach, member m\_subnodeArray[]). The ordered set starts with the upstream end of the reach and ends with the downstream end. The kinematic wave algorithm is used to estimate the average daily outflow from each subreach. The daily flow rate for the reach is taken as the average daily flow rate of the most downstream subreach of the reach.

The core of the kinematic wave code is a function in the ReachRouting.cpp source file called *KinematicWave()*. The explicit inputs to the function are the outflow rate from the previous day, the current day’s inflow rate from upstream, and the current day’s lateral inflow rate. The use of the current day’s inflow rate as an input necessitates taking the reaches in order from headwater reaches down to the pour point; the outflow from one reach cannot be calculated until the outflows from all the other reaches upstream have been calculated. The function uses an equation for the depth of the water in the reach as a function of the outflow rate which is derived from the Manning equation for the outflow rate. The Manning equation itself is:

Q = A \* R2/3 \* √S / *n*

(<https://www.openchannelflow.com/blog/manning-formula-for-determining-open-channel-flows>)

where

Q = volumetric flow rate, m3/s

A = cross-sectional area of flow, m2

p = wetted perimeter, m

R = A/p hydraulic radius, m

S = slope, as rise/run

*n* = Manning roughness coefficient, a function of channel material and condition, s/m1/3

Manning roughness coefficients are usually in s/ft1/3, with typical values 0.001 s/ft1/3 (smoothest) – 0.050 s/ft1/3 (roughest). To convert from a table value of *n* in s/ft1/3 to a value in s/m1/3, multiply the table value by (1 m / 3.28084 ft)1/3 (= 0.673).

By algebraic manipulation and by assuming that the channel has a rectangular cross-section with a fixed width:depth ratio, the Manning equation can be rewritten as an equation for depth d as a function of the flow Q, the channel width:depth ratio wdr, and the slope S. The formula used for depth as a function of Q in CW3M makes depth proportional to the 3/8 power of Q. The assumption that the channel cross-section is both rectangular and has a fixed width:depth ratio is mathematically convenient but physically questionable. In reality, holding the width:depth ratio constant with changing flow would necessitate changes to the channel width in proportion to the channel depth as Q changes. Doing so while maintaining a rectangular cross-section would necessitate displacing the streambed and stream banks.

The actual C++ code in the *GetManningDepthFromQ()* function has this form

d = (0.3 Q / (√S \* wdrterm))3/8

where

wdr = the width:depth ratio, a prescribed constant (CW3M default value = 10)

wdrterm = wdr \* (wdr / (2 + wdr))2/3

float wdterm = (float)pow((wdRatio / (2 + wdRatio)), 2.0f / 3.0f) \* wdRatio;

= (wdr / (2 + wdr))2/3 \* wdr

float depth = (float)pow(((Q \* pReach->m\_n) / ((float)sqrt(pReach->m\_slope) \* wdterm)), 3.0f / / 8.0f);

= ((Q \* m\_n) / (√S \* wdterm))3/8

Consolidating the constant factors into a single term

kk = (0.3 / √S)3/8

the C++ code could have been written as

d = kk \* Q3/8 / wdrterm3/8

Here is the derivation of the C++ form from the Manning equation.

Q = A \* R2/3 \* √S / *n* original form of Manning equation for volumetric flow

Using the assumption of a rectangular channel cross-section, express A, R, p and w in terms of d and wdr.

w = d \* wdr width, m

A = w \* d cross-sectional area of flow, m2

= d2 \* wdr

p = w + 2d wetted perimeter, m

= d \* wdr + 2d

= d \* (wdr + 2)

R = A / p

= (d2 \* wdr) / (d \* (wdr + 2))

= d \* wdr / (wdr + 2)

Substitute those expressions for A and R back into the original form and solve for d in terms of Q and wdr.

Q = (d2 \* wdr) \* (d \* wdr / (wdr + 2))2/3 \* √S / *n*

Q = (√S / *n*) \* d8/3 \* wdr \* ( wdr / (wdr + 2))2/3

d8/3 = (*n* / √S) \* Q \* (1 / (wdr \* ( wdr / (wdr + 2))2/3))

d8/3 = (*n* / √S) \* Q \* (1 / wdrterm)

d8/3 = (*n* / √S) \* Q / wdrterm

(d8/3)3/8 = ((*n* / √S) \* Q / wdrterm)3/8

d = (*n* / √S)3/8 \* Q3/8 / wdrterm3/8

This is similar to the form used in the C++ code

d = (0.3 Q / (√S \* wdrterm))3/8

which implies

(0.3 / √S)3/8 = (*n* / (K \* √S))3/8

0.3 = *n* / K

*n* = 0.3 K

and, for *n* in s/ft1/3, K is 1 / 1.4859 ft1/3/m1/3, and

*n* = 0.3 / 0.673

*n*, s/ft1/3 = 0.446

Table 9-6 “Values of Manning’s *n* for Channels of Various Types” (p. 428) of Dingman (2002), which is attributed to Chow (1959), gives minimum, normal, and maximum values of *n* for a larger variety of channels. Maximum *n* values range up to 0.200 (“dense willows, summer, straight”) and include 0.150 (max) and 0.100 (normal) for “very weedy reaches, deep pools, ir floodways with heavy stand of timber and underbrush”. In CW3M, the use of a value for *n* much higher than the table values is inherited from the WW2100 project. As of 12/13/20, the default value for Manning’s *n* in CW3M has been changed to 0.045 s/m1/3 (= 0.030 s/ft1/3), which is the normal value given in the table for "Minor streams (top width at flood stage < 100 ft) / Streams on plain / 1. Clean, straight, full stage, no riffles or deep pools". CW3M’s simulated flows seem to be relatively insensitive to the value of *n*, at least at the basin level.

At the beginning of the *KinematicWave()* function, the previous day’s outflow rate is used in a call to the function *GetManningDepthFromQ()* to calculate the “Manning depth” of the stream. Then that depth is used together with the length of the subreach and the current day’s upstream and lateral inflows to calculate the current day’s outflow rate. The current day’s outflow rate is subsequently used in a second call to *GetManningtDepthFromQ()* to calculate a new Manning depth for the water in the reach.

Note that there are two ways to calculate the width of the subreach. Both ways use the depth from the *GetDepthFromQ()* function. The simpler way is to multiply the depth by the parameterized width-to-depth ratio. The other way is to use that depth, together with the length of the subreach and the volume of water in the subreach to calculate the width. The *SetSubreachGeometry()* function uses the second way, which results in a width which is consistent with the length, depth, and volume, but is generally *not* equal to what would be expected from the parameterized width-to-depth value. Getting a plausible value for the stream width is important because it determines the surface area of the water in the stream, which in turn partially determines the radiative energy fluxes: wide shallow streams are more influenced by insolation and longwave radiation than narrow, deep streams.

As implemented*,* *KinematicWave()* makes the rate of flow out of a subreach more immediately responsive to the inflow from upstream than to the lateral flow. In the original WW2100 implementation, this characteristic led to unrealistic accumulation of water in headwater reaches, since by definition there is no inflow from upstream into the upstream end of a headwater reach. A workaround was later adopted to compensate for this malfunction, in the form of logic which treats any volume in a subreach in excess of the sum of the current day’s actual inflow from upstream and the current day’s lateral flow as if it were part of the current day’s inflow from upstream.

## Estimating the surface area of the water in a subreach

For simulation of stream temperatures, a plausible estimate of the water surface area of the subreach is important, because the radiative and evaporative energy exchanges are proportional to the surface area. The surface area of the water is approximated by the product of the subreach length, which is fixed, and the width of the stream. In principle, the width of the stream varies with the volume of water in the subreach, and varies in a different way with the rate of flow. The volume of water itself is determined by the daily mass balance, after determination of the flows into and out of the subreach. CW3M has a built-in, default method for estimating stream width; the estimate goes into the Reach attribute WIDTH\_CALC.

In some places, the width of the reach is relatively constant. Where the width of the reach is known, it may be specified explicitly in the Reach attribute WIDTHGIVEN. Values greater than zero in WIDTHGIVEN override the value in WIDTH\_CALC. The operative width value is held in the attribute WIDTHREACH.

The width estimate that goes into WIDTH\_CALC is the average of the calculated widths for the subreaches that make up the reach. Subreach widths are estimated as a function of the length of the subreach, the volume of water it contains, and a prescribed width-to-depth ratio = 10.

## Water temperature from thermal energy

The amount of thermal energy in the water in a reach or reservoir, referenced to zero energy for liquid water at the freezing point of water, is estimated and tracked within CW3M’s Flow model. The average temperature of the water in a reach can then be calculated for output purposes from the thermal energy per unit volume

temperature = thermal energy / (volume \*water density \* specific heat of water)

where

temperature = average temperature of the water in the reach (degC)

thermal energy = the thermal energy of the water in the reach, rel to the freezing pt. (KJ)

volume = volume of water (m3)

water density = 998.2 kg/m3

specific heat of water = 4.187 kJ/(kg degC)

## Boundary conditions for stream water temperature

Water enters stream reaches from upstream and from lateral flow. Water entering from upstream already has a temperature associated with it. The temperature of water entering laterally as runoff from the land is estimated as a function of the air temperature:

H2O\_TEMP = W2A\_SLP[hbvcalib] \* AIR\_TEMP + W2A\_INT[hbvcalib]

hbvcalib is the index to the row in the HBV calibration data for the subbasin where the reach is located. W2A\_SLP and W2A\_INT are column headings in the table of HBV calibration data (the HBV.csv file). Basing the temperature of the water on the temperature of the air introduces seasonality to the water temperatures.

Where sufficient measured data of stream temperatures is available, the W2A\_SLP and W2A\_INT values are obtained from a linear fit of measured monthly stream temperatures to air temperatures from the baseline weather data. If measured stream temperatures are not available, the   
W2A\_SLP and W2A\_INT values may be copied from those for a similarly situated subbasin. If there are no comparable calibrated subbasins, W2A\_SLP may be set to zero and W2A\_INT set to DEFAULT\_SOIL\_H2O\_TEMP\_DEGC, which has the effect of setting the temperature of the runoff when it enters the reach to a constant, year around.

As of 12/8/20, two symbols are used in the source code to parameterize the temperature of water entering the streams:

#define DEFAULT\_SOIL\_H2O\_TEMP\_DEGC 5.

#define DEFAULT\_REACH\_H2O\_TEMP\_DEGC 8.

The reach water temperature is used when initializing flow data in the absence of an IC file.

Further elaboration of water temperature boundary conditions is necessary. Water returns to stream reaches from municipal water systems at points of central discharge. Municipal discharge water is typically warmer than stream water, and its temperature may vary seasonally. The temperature of the water reaching the streams via the stream banks also varies seasonally, but not in strict proportion to air temperature (e.g. cold streams in spring from snow melt). A 2019 article by Leach and Moore in Water Resources Research provides some clues and further references for improving the representation of the temperature of water entering via the stream banks.

As of 12/8/20, stream temperatures have only been fitted to air temperatures for 3 drainages in the McKenzie basin: the Cougar reservoir drainage (HBVCALIB = 8), the Blue River reservoir drainage (HBVCALIB = 9), and the McKenzie outlet (HBVCALIB = 16). The fit for the Cougar drainage is also being used for the Clear Lake drainage (HBVCALIB = 46) and for the upper McKenzie above Walterville (HBVCALIB = 34). The fit for the McKenzie outlet is being used for the Mohawk subbasin (HBVCALIB = 25). Here are the values:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HBVCALIB | subbasin | mean elevation, m ASL | W2A\_SLP deg H2O/deg air | W2A\_INT deg C |
| 16 | McKenzie outlet | 314 | 0.60 | 3.37 |
| 9 | Blue River reservoir | 932 | 0.64 | 2.70 |
| 8 | Cougar reservoir | 1222 | 0.38 | 3.61 |

## Thermal stratification in reservoirs

Initially, for the purpose of simulating water temperature, reservoirs will be represented as extremely large stream reaches characterized by a single uniform temperature. Discharges wil be at that temperature. This is an unfortunate oversimplification, because in reality temperature is a function of depth during the summer, and the Army Corps of Engineers can manage downstream water temperatures to some extant by choosing how much water to release from outlets at different depths.

The initial oversimplification is necessary due to the lack of adequate (and adequately simple) algorithms for simulating thermal stratification and for simulating USACE’s daily operating decisions about how much water to release from different available outlets.

## Thermal loading

A thermal loading algorithm will be implemented in CW3M as part of the McKenzie Basin Wetlands Study. The scope for the study calls for replication in CW3M, to the extent that it is practical, of the methods used in the Shade-a-lator model, part of the Heat Source model. Heat Source and Shade-a-lator are maintained by the Oregon Department of Environmental Quality.

For CW3M, the relevant output of Shade-a-lator is the thermal load in kcal/day of insolation on specific stream segments. Shade-a-lator calculates the kcal/day figure for each “node” along a stream; nodes are typically 25 meters apart. A single node is characterized by elevation, aspect, stream width, right and left bank vegetation height and density, and topographic shading.

CW3M divides streams up into reaches from the National Hydrography Dataset, and subdivides the reaches into subreaches. Within a reach, subreaches are of equal length and no longer than 1000 meters. Reaches are represented by a series of points; conceptually, the stream flows from point to point. Importantly, the points are not equally spaced and are not collinear. Some reaches are curved, and are represented by relatively many closely-spaced points. Other reaches are straighter, and are represented by fewer points spaced farther apart.

CW3M represents the McKenzie system with 1,047 reaches of average length 1,781 meters. The longest reach is 9,010 meters; the shortest is just 6 meters. The Reach\_McKenzie.shp file which stores the points which define the reaches is 6,361,228 bytes long, an average of 6,076 bytes per reach. Even if each point required as many as 100 bytes of storage, that would be an average of 61 points per reach and a total of 63 thousand points, representing more than 60 thousand straight stream segments averaging about 30 meters in length. Double precision floating point numbers require 8 bytes of storage, so 3 coordinates, if stored as double precision floating point numbers, would require only 24 bytes. So it is likely that each point takes up much less than 100 bytes, and that hence there are many more than 63 thousand points altogether, with an average distance between them of much less than 30 meters.

Short, straight stream segments correspond to the nodes used in Shade-a-lator, as they can be characterized by single values for elevation, aspect, stream width, and vegetation characteristics. Elevation and aspect can be calculated from the 3D coordinates of the points themselves. Other values, e.g. vegetation characteristics, are available from the IDUs within which the stream segments are situated. Since there are only 16,883 IDUs, the values derived from the IDUs would necessarily apply, on average, to about 4 contiguous stream segments. On sharply curved reaches with closely spaced points, the same IDU-derived values would apply to many such contiguous segments.

Since the aim is to reproduce the Shade-a-lator results as much as is practical, we propose to create an additional GIS data layer, similar to the reach layer but more detailed spatially. Since the terms “subreach” and “subnode” have already been used in the CW3M code base to refer to the equal length pieces of each reach used in the flow calculations, we’ll call this new layer the “stream segment” layer. Its characteristics are:

* segments are subdivisions of reaches
* segments are straight and defined by just 2 points, the endpoints, which are themselves points from the reach of which the segment is a part
* segments are not less than 25 meters in length, except when the reach itself is less than 25 m long
* segment attributes include midpoint elevation, aspect, slope, and topographic shading, all derived from the reach file and digital elevation map
* each of the two stream banks for each segment is associated with an IDU, which may be the same IDU for both banks; since segments may cross more than one IDU, a rule will have to be devised to pick the IDU most characteristic of the stream bank, when there is more than one
* additional segment attributes are derived from the IDUs which the segment traverses, and include vegetation type, height, and density or some other characterization of vegetative shading
* additional attributes may be necessary to satisfy the data requirements of the Shade-a-lator algorithms (e.g. to calculate bank shading), for example channel width under base flow conditions

The stream segment layer would thus consist of simple straight line segment, generally at least 25 meters in length, defined by points already present in the reach layer. Reaches would decompose into one or more continuous segments. Segments would be in a many-to-one relationship with reaches. Segments would be the unit of computation for the Shade-a-lator—like thermal loading calculations.

Some of the data needed for the thermal loading calculations will be constant and can be precalculated offline, for example topographic shading characteristics. Other data may change seasonally (vegetation density) or interannually (tree height). Insolation is also dependent on the season and the weather. Accordingly, the thermal loading calculation will be added to the daily calculation of stream flow for each reach.

# Reservoirs

CW3M simulates the 13 reservoirs in the U.S. Army Corps of Engineers (USACE) Willamette Project. There are some other non-USACE dams and reservoirs in the Willamette basin as well, but they are not (as of November 2020) represented in CW3M.

The USACE reservoirs appear in the IDU layer as polygons with LULC\_A = 7 Water/Snow/Ice, LULC\_B = 71 Open Water, VEGCLASS = 111 Open Water, and COMID = the COMID of a reach associated with the reservoir. The Reach layer has a RES\_ID attribute which is zero for most reaches, but is equal to the reservoir’s RES\_ID value for reaches which overlap with the reservoir’s area. Here are the RES\_ID values:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| USACE  code | RES\_ID | name | subbasin | reservoir type |
| HCR | 1 | Hills Creek | Middle Fork | flood control, hydropower |
| LOP | 2 | Lookout Point | Middle Fork | flood control, hydropower |
| DEX | 3 | Dexter | Middle Fork | re-regulating, hydropower |
| FAL | 4 | Fall Creek | Middle Fork | flood control |
| DOR | 5 | Dorena | Coast Fork | flood control |
| COT | 6 | Cottage Grove | Coast Fork | flood control |
| FRN | 7 | Fern Ridge | Long Tom | flood control |
| CGR | 8 | Cougar | McKenzie | flood control, hydropower |
| BLU | 9 | Blue Lake | McKenzie | flood control |
| GPR | 10 | Green Peter | South Santiam | flood control, hydropower |
| FOS | 11 | Foster | South Santiam | flood control, hydropower |
| DET | 12 | Detroit | North Santiam | flood control, hydropower |
| BCL | 13 | Big Cliff | North Santiam | re-regulating, hydropower |

Each reservoir is associated with a single reach which receives its outfall. The outfall reaches are identified in the Reach layer by having the reservoir’s RES\_ID value in the Reach attribute RESOUTFALL. Conceptually, the reservoir is represented as a water body inserted into the reach tree just upstream of the outfall reach and downstream of all the reaches which overlap the reservoir’s actual area.

## Reservoir data

The USACE operates the 13 reservoirs in the Willamette Project as an integrated system. The Corps manages the releases of water from the reservoirs, using a set of rules and data which are represented in CW3M’s reservoir submodel. The CW3M code is sometimes referred to as “RESSIM-lite”, after the Corps’ own ResSim model.

Reservoir rules and data are stored in the CW3M repository at DataCW3M/Reservoirs. Data and rules for specific reservoirs are identified in their individual <reservoir> blocks in the Flow XML file. Here is an example.

<reservoir id="9" name="Blue River Reservoir (USACE)"

reservoir\_type="FloodControl”

td\_elev="415.0" fc1\_elev="414.0" inactive\_elev="359.66"

maxVolume="168123574" initVolume="5063443"

minOutflow="1.417" maxPowerFlow="0"

path="Reservoirs\"

av\_dir="Area\_Capacity\_Curves\" area\_vol\_curve="blue\_river\_area\_capacity.csv"

rc\_dir="Rule\_Curves\{ScenarioName}" rule\_curve="blue\_river\_rule\_curve.csv"

buffer\_zone="BR\_buffer.csv"

re\_dir="Rel\_Cap\" composite\_rc="BR\_composite\_rc.csv"

RO\_rc="BR\_RO\_capacity.csv"

spillway\_rc="BR\_spillway\_capacity.csv"

cp\_dir="ControlPoints\{ScenarioName}"

rp\_dir="Rules\_BR\" rule\_priorities="blue\_river\_rule\_priorities.csv"

ressim\_output\_f="BR\_ressim\_flow\_1935\_2008.csv"

ressim\_output\_r="BR\_ressim\_rule\_1935\_2008.csv"

/>

Five subdirectories of the Reservoirs directory and nine CSV text data files in the DataCW3M\Reservoirs directory tree are identified for the single reservoir in this example. Some paths include a {ScenarioName} field, which allows CW3M to incorporate the name of a simulation scenario into the path at run time, when {ScenarioName} is defined elsewhere in the Flow XML file.

Area\_Capacity\_Curves\blue\_river\_area\_capacity.csv – The Area\_Capacity\_Curves subdirectory has one CSV file for each reservoir. Each such file has three columns: Elevation\_m, Storage\_m3, and Area\_ha. The one for BLU has 400 rows of data, with elevations in sorted order from 396.2 to 517.9.

ControlPoints\{ScenarioName} – The ControlPoints subdirectory has 20+ files, each containing data for a particular control point under a particular set of circumstances. For example, there is a file in the ControlPoint subdirectory named cp\_Max\_bf\_Goshen\_23759228.csv. Column names vary from file to file.

Rule\_Curves\{ScenarioName}blue\_river\_rule\_curve.csv and ...}BR\_buffer.csv – The Rule\_Curves subdirectory has 2 files for each of the reservoirs except the 2 re-regulating reservoirs, BCL and DEX. The 2 files for BLU are named Blue\_River\_rule\_curve.csv and BR\_buffer.csv. Each has 2 columns; the first column is titled Date and contains a Julian day number (1 = Jan 1; 365 = Dec 31). The second column in the Blue\_River\_rule\_curve.csv file is titled Cons\_Pool\_elev\_m, and there are 29 rows of data. The second column in the ...buffer.csv file is titled Pool\_Elevation\_m, and there are 19 rows of data.

Rules\_BR\blue\_river\_rule\_priorities.csv – There is a separate Rules\_<reservoir name> subdirectory for each reservoir except the 2 re-regulating reservoirs, BCL and DEX. blue\_river\_rule\_priorities.csv has 5 columns labeled simply 0,1,...,4 and 7 rows. Each row, column position contains either a file name (16 positions, but only 12 unique file names) or the word Missing (18 positions), or is empty (1 position). The 12 file names are

cp\_Max\_Vida\_Flood\_23772903.csv

cp\_Min\_Flow\_at\_Albany\_23762845.csv

cp\_Min\_flow\_at\_Salem\_23791083.csv

Max\_con\_flow\_blue\_river.csv

Max\_EvaculationRelease.csv

MaxD\_Daily\_BiOP\_MaxD.csv

MaxD\_FloodDcrsRate\_Blue.csv

MaxD\_MaxFloodDcrsRate\_Blue.csv

MaxI\_FloodIncrsRate\_Blue.csv

MaxI\_MaxFloodIncrsRate\_Blue.csv

Min\_Flow\_50\_cfs.csv

Min\_Flow\_at\_Blue\_River.csv

All the files beginning with cp\_ are in the Reservoirs/ControlPoints subdirectory. The other files are in the Rules\_BR subdirectory itself.

Rel\_Cap\BR\_composite\_rc.csv, ...\BR\_RO\_capacity.csv, and ...\BR\_spillway\_capacity.csv – The release capacity subdirectory (Rel\_Cap) contains 39 files, 3 for each of the 13 reservoirs. The prefix of each file name indicates the reservoir, and the suffixes are composite\_rc.csv, RO\_capacity.csv, and spillway\_capacity.csv. For BLU, the three files have the same structure: 2 columns labeled pool\_elev\_m and release\_cap\_cms, and multiple rows of data.

BR\_ressim\_flow\_1935\_2008.csv and BR\_ressim\_rule\_1935\_2008.csv – These files are in Reservoirs\Output\_from\_ResSim.

## Reservoir data sources

Reservoir data in CW3M as of version 113 (11/18/20) is from the WW2100 project, and was probably compiled around 2012. A newer source for at least some this data would be Appendix E, ResSim Analysis for 2008 Baseline Flow Dataset (June 2018), part of the Willamette Basin Review Feasibility Study/Final Integrated Feasibility Report and Environmental Assessment, issued in December 2019 by the Corps’ Portland District (USACE 2019). This report is a product of the “Willamette Project Reallocation Study”, undertaken by the USACE together with state agencies. The reallocation study was originally authorized in 1988 and proceeded in fits and starts over a period of three decades.

In Appendix E of the report, Sec. 3.3 “ResSim Inputs for Physical Parameters of Each Dam” (p. 18) says

“The physical parameters of the dams in ResSim for the Baseline will remain the same for all alternatives evaluated – alternatives will only have operational rule changes from the Baseline.”

This gives us confidence that the reservoir data representing physical parameters of the dams, which was assembled for the original WW2100, is still correct.

Comparison of the data in the Rule\_Curves\blue\_river\_rule\_curve.csv file with Table 5.1 Blue River Zone Specifications makes visible that the WW2100 data is expressed in meters with no fractional digits, while Table 5.1 uses feet with one or more digits after the decimal point. Table 5.1 also has the data for the Rule\_Curves\BR\_buffer.csv file.

# Creating a new study area from a watershed within the WRB

Any complete watershed, whether the entire Willamette River basin, a tributary subbasin such as that of the North Santiam River, or a smaller drainage like the Chicken Creek watershed, may be used as a study area. Moreover, a study area may be formed from the combination of several named tributary basins, so long as they combine to a single pour point. For example, the upper Willamette basin study area incorporates the areas drained by the Coast and Middle Forks of the Willamette River together with the McKenzie River watershed.

The spatial extent of the study area is defined by the IDU shapefile. Reach and HRU shapefiles are also required, and must be consistent with the IDU shapefile. Once a particular drainage has been selected as a study area, the necessary shapefiles are made from the larger WRB shapefiles using ArcMap.

ArcMap Tools/Analysis/Extract/Select is used to produce the IDU file for the target area from the WRB IDU file. Selections may be made on the SUB\_AREA attribute or the HBVCALIB attribute. The query expression must be chosen so as to produce a whole watershed. The SUB\_AREA attribute generally identifies the areas drained by the major tributaries of the Willamette River. Each single value of SUB\_AREA from 1 through 11 selects a whole watershed:

|  |  |  |
| --- | --- | --- |
| SUB\_AREA | Tributary | COMID of outlet reach |
| 1 | McKenzie | 23765583 |
| 2 | Molalla and Pudding | 23800560 |
| 3 | Clackamas | 23809000 |
| 4 | Long Tom | 23763071 |
| 5 | Marys | 23762881 |
| 6 | North Santiam | 23780877 |
| 7 | South Santiam | 23785607 |
| 8 | Tualatin | 23792815 |
| 9 | Coast Fork | 23759222 |
| 10 | Middle Fork | 23751752 |
| 11 | Upper Yamhill above McMinville | 23791899? |
| 12 | Upper Willamette mainstem | 23763395 |
| 13 | Lower Santiam | 23780405 |

Selections of whole watersheds may also be made using the HBVCALIB attribute, but unlike SUB\_AREA, several values of HBVCALIB may be necessary to select for a complete drainage because some HBVCALIB drainages are nested within others. For example, the ones for the Cougar and Blue River reservoirs are nested within the one for the McKenzie above Walterville. In such cases, the selection should be made on all the HBVCALIB values, e.g. HBVCALIB = 34 OR HBVCALIB = 8 OR HBVCALIB = 9 would select for the portion of the McKenzie basin which drains to the gage at Walterville. The HBVCALIB tab in CW3Mhandbook.xlsx lists the HBVCALIB drainages, and Figure 1 illustrates many of their locations within the WRB.

A third attribute, Sub\_Area\_C (“calibration subareas”), identifies 19 small watersheds which were used in an early WW2100 HBV calibration. They are listed on the Sub\_Area\_C tab of CW3Mhandbook.xlsx.

To prepare the necessary subbasin shapefiles, the ArcMap Selection and Clip tools are used (in ArcMap, Tools/Analysis/Extract/Select and …/Clip).

The Select tool is used to create the subbasin Reach and IDU layers from the WRB Reach and IDU layers. The selection is usually made on the HBVCALIB attribute. For example, HBVCALIB = 9 would select the Blue River reservoir drainage. Note that some HBVCALIB drainages are nested within others, e.g. for the North Santiam basin, the expression would be HBVCALIB=12 OR HBVCALIB=37 OR HBVCALIB = 44.

Selections may be made on other attributes, but the “whole watershed model” concept requires that the selected area should form a watershed draining to a single point. Single values of the SUB\_AREA attribute select for major subbasins (e.g. SUB\_AREA = 3 selects for the Clackamas River drainage). Single values of the Sub\_Area\_C attribute select for smaller watersheds used in the original HBV calibration work.

The Clip tool is used to clip the WRB HRU layer to the extent of the newly created subbasin IDU layer, to create the subbasin HRU layer.

Subbasins which are not currently defined by any value of SUB\_AREA or Sub\_Area\_C or any combination of HBVCALIB values may be defined by adding to the HBVcalibPt array in WaterRights.cpp, and then running a model coldstart. The HBVcalibPt entry specifies the location or COMID of a pour point; the coldstart logic sets the HBVCALIB attribute to the specified value for all the reaches and IDUs which drain to that pour point, and which have not already been assigned an HBVCALIB value. To activate coldstart, set the coldstart parameter to 1 in the settings block of the .envx file. Since the coldstart function generates indices which are preserved in subbasins, it should only be used when simulating the complete Willamette River basin study area.

The newly created subbasin shapefiles should be placed in a folder in the DataCW3M directory; that folder should be named for the target subbasin. For example, the shapefiles for the North Santiam watershed are in DataCW3M\NSantiam. Any other files specific to the subbasin should also be placed in the subbasin folder, except for the subbasin ENVX file itself. By convention, the name of the subbasin ENVX file is composed as <model name>\_<subbasin name>.envx. For example, the ENVX file for the North Santiam subbasin is named “CW3M\_NSantiam.envx”. ENVX files should be placed in the DataCW3M directory.

Here is an example of the names and locations of all the files specific to a particular subbasin.

DataCW3M (folder)

CW3M\_McKenzie.envx

McKenzie (folder)

Flow\_McKenzie.xml

HRU\_McKenzie shapefile

IDU\_McKenzie shapefile

Reach\_McKenzie shapefile

## Limitations of subbasin simulations

Anthropogenic effects cross subbasin boundaries, so simulations of single subbasins may omit effects which would be present if the whole Willamette basin were simulated.

The USACE operates the Willamette Project dams and reservoirs as a system, modifying reservoir outflows in upstream subbasins as necessary to observe target flow limits at downstream control points (e.g. Albany and Salem). The feedback from the downstream control points cannot be simulated when individual upstream subbasins are simulated. For example, when simulating the McKenzie basin, the Salem control point rule cannot be applied in the logic to determine the release rate from the Blue River and Cougar dams, since the flow at Salem is outside the McKenzie basin is not simulated.

Municipal water demands are simulated for complete urban growth areas (UGAs) and are correlated with the populations of the UGAs. Where a UGA lies partly within a subbasin which is being simulated by itself, the population data for the UGA must be adjusted in order to scale the municipal water demand. For example, the Portland Metro UGA overlaps the Chicken Creek watershed, but Chicken Creek includes only a small fraction of the population of the Metro UGA.

# Calibration procedure

Calibration of the model is done by simulating past years, comparing the simulation output to observations taken during those years, and adjusting various aspects of the model to improve the correlation of simulated output with observations. Simulation of past years (“hindcasts”) are carried out using input data representing actual weather, population, and other actual conditions during those years, to the extent it is practical.

## Using PEST to determine HBV parameter values

The first step in the calibration process is to choose HBV parameter values. HBV is the precipitation-runoff submodel; the values of its parameters determine how fast rain and snow drain into the stream reaches. We use an automated parameter estimation program called PEST (Model-Independent Parameter Estimation and Uncertainty Analysis, https://pesthomepage.org/) to select values for ~8 parameters. PEST is run on some of the smaller drainages and tributary subbasins, a subset of the ~40 watersheds identified by the HBVCALIB index. Parameter values for watersheds not calibrated with PEST are copied from similarly situated watersheds that have been calibrated with PEST. The use of PEST is limited by the time it takes to run; choosing parameter values for a single watershed may require simulating the watershed hundreds of times. Running PEST for a watershed can take several days of computer time.

PEST calibrations were carried out twice for the WW2100 project. PEST was subsequently run on a single small subbasin (Chicken Creek, which drains into the Tualatin River) during the OUWIN project. Results from the second WW2100 calibration and from the OUWIN Chicken Creek calibration are reported in Conklin (2016) and Conklin (2019).

Values of the HBV parameters are held in the HBV.csv file, which is identified at 3 places in the Flow XML file: in the <evap\_trans> block named “ForestET”, in the <evap\_trans> block named “AgET”, and in the <tables> block as a table named “HBV”. These 3 references should all point to the same file. Note that not all of the columns in the HBV.csv file contain values from PEST. In the most recent use of PEST, 4 parameters were given constant values (TT, CFMAX, CFR, and WP). Three columns have been added to HBV.csv since the most recent work with PEST: ET\_MULT, W2A\_SLP, and W2A\_INT. Those 3 parameters will be described later in this section.

The second PEST calibration for WW2100 (the one described in Conklin 2016) was carried out using “NRNI” data for the fifteen years 1980-1994 from the USGS. “NRNI” stands for “no reservoirs no irrigation”. NRNI data is observations from USGS stream gages adjusted (by the USGS) to remove anthropogenic effects. The later OUWIN Chicken Creek PEST calibration used actual gage readings for the 7 water years 2009-15, rather than NRNI data. As of 12/28/20, calibration work for CW3M uses actual gage data, not NRNI data, for the 9 calendar years 2010-18 as the base period. Simulations for 2010 are begun using initial conditions produced by a 10-year spinup for 2000-09.

## Calibrating streamflows using stream gage measurements

After the results of any PEST calibrations have been incorporated into the HBV.csv file, then the next step is to use the available tuning knobs to bring the simulated monthly streamflows as close as practical to the observed monthly streamflows. Wherever USGS gage data exists, we compare the simulated streamflows to the gage measurements, after aggregating to monthly timesteps. Generally, when the %BIAS statistic has a magnitude of 1% or less, and the other Moriasi statistics get grades of S, G, or VG, then we consider the calibration for the portion of the study area which drains to that location as satisfactory or better. The Moriasi statistics are described in a later section “McKenzie basin wetlands study / Calibration and simulation skill”.

Calibration of upstream reaches affects the flows further downstream, so the calibration process begins with the headwater basins and proceeds downstream. Some reaches receive water from springs as well as from runoff. Springs are specified in the Flow XML file; each spring is associated with a specific reach, a constant flow rate in m3/s, and a constant temperature in deg C. The average flow in any reach fed directly or indirectly from a spring may be adjusted up or down by changing the specified flow rate for the spring. Since spring flow rates are constant all year long in the model, the adjustments apply uniformly across the year. Adjustments to the spring flow rates do not serve to make the simulated seasonal pattern of flows conform better to the observed seasonal pattern. The springs which feed Clear Lake in McKenzie basin are a special case, which is discussed in the section on the McKenzie basin wetlands study.

ET\_MULT is a tuning knob which affects all the drainages, whether or not they have springs. Separate values of ET\_MULT are specified in the HBV.csv file for each of the drainages identified by the HBVCALIB index. ET\_MULT is a multiplier on the simulated evapotranspiration (ET). The water which returns to the atmosphere via ET would otherwise end up in the stream, so increasing ET has the effect of decreasing streamflow and vice versa. The effect is not strictly linear in ET\_MULT, because ET is also affected by soil moisture, leaf area, and so on. Since ET itself varies seasonally, adjusting ET\_MULT, unlike changing spring discharge rates, can affect the correlation between the seasonal pattern of simulated flows and the seasonal pattern of the observations.

## Calibrating stream temperatures

Stream temperatures are simulated by keeping track of the thermal energy, relative to 0 deg C, of each ”parcel” of water tracked by the model in the stream network. Each subreach has an associated water parcel, as does each reservoir (and, eventually, each wetland IDU). Using this energy balance approach requires supplying a water temperature as well as a water volume for each water parcel entering or leaving the stream network. Specifying the temperature of water entering the streams is challenging. As of 12/28/20, the temperature of water entering a reach on a given day as runoff or through the streambanks is modeled as a linear function of the air temperature on that day at the location of the reach. Doing so requires specifying the slope and intercept of the linear fit, which is supplied in the W2A\_SLP and W2A\_INT columns of the HBV.csv file. Treating runoff temperature on a day as directly proportional to the air temperature on the same day is not ideal (e.g. spring melt water temperature will not bear the same relationship to air temperature as does summer thunderstorm rain does). But there is precedent in the literature (Leach & Moore 2019 discuss how others have done what we’re doing, and then go on to propose a better way) and it seems to work well enough until we have time to evaluate other alternatives.

The stream temperature in reaches fed directly or indirectly by water from springs is also affected by the specified temperature of the spring water. The spring water temperature can be used as a tuning knob, although this may lead to an inconsistency between published values, if there are any, of the spring water temperature and the value actually used in the simulation.

Stream temperature is also affected by insolation, which itself is a complicated function of local conditions (vegetation height and density, stream width and orientation, topographic shading, ...). As of 12/28/20, simulated insolation of stream reaches is not yet taking the local conditions into account.

The water temperature in reservoirs impacts the stream temperature in reaches below the reservoirs. Reservoir water temperature is raised by insolation, especially in summer. The amount of insolation is affected by the topographic shading on the reservoir water surface. As of 12/28/20, the model does not yet have any topographic shading data for either reservoirs or reaches. As a work-around, and possibly as a permanent tuning knob, a parameter “solar\_radiation\_multiplier” has been added to the <reservoir> block in the Flow XML file. This parameter should be set to a value between 0 and 1 to allow for reduction of insolation of the reservoir water surface due to topographic shading. It can then be dialed up or down to adjust downstream water temperatures to get them closer to observed values.

# North Santiam study area

The North Santiam (NSantiam) study area (Figure 3) consists of the watershed of the North Santiam River down to the point where the North and South Santiam Rivers come together to form the Santiam River itself. The outlet reach of the study area has COMID 23780877. Its area is 469,061 acres; it is comprised of 11,001 IDUs, 686 HRUs, and 710 reaches. Various parameter values for the NSantiam are recorded in the “NSantiam” tab of the NSantiam.xlsx file in the CW3Mhandbook folder. In addition, there is a separate file devoted to data for the North Santiam study area, called “NSantiam.xlsx”. There are tabs in NSantiam.xlsx labeled

Municipal Populations

Municipal Water

Muni WR details

Instream Water Rights

existing instream WR details

unconverted WR details

Water Right Codes

## Instream water rights

Per Joel Plahn at OWRD on 12/13/18, there are 6 instream water rights and 3 unconverted applications for instream water rights in the North Santiam basin. All are owned by OWRD. There is also a judicial decree from 1945. In the list below, the number in the leftmost column is the water right ID for those currently included in CW3M. Prescribed minimum flow rates are stated in cubic feet per second. Some prescribed instream water right minimum flow rates vary seasonally; where that is the case, the smallest and largest flow rates are given.

North Santiam R.

- 1945 decree from Gardner Bennett diversion to the mouth 50.0 cfs

200081 appl# MF141 at USGS gage 14-1841 near Jefferson at Greens Bridge 6/22/64 430.0 cfs

reach 23780833; one order 3 reach

200082 appl# MF142 at USGS gage 14-1830 at Mehama 6/22/64 580.0 cfs

reach 23780481; one order 5 reach

200083 appl# MF143 at USGS gage 14-1815 at Niagara 6/22/64 500.0 cfs

reach 23780511; one order 5 reach

118093 cert# 65756 at USGS gage 14-1780 near Detroit 6/22/64 345.0 cfs

reach 23780591; one order 4 reach

Little North Santiam R.

118092 cert# 65755 Little North Santiam R. at USGS gage 14-1825 near Mehama 6/22/64 40.0 cfs

reach 23780805; one order 3 reach

124853 cert# 72598 Little North Santiam R. from Battle Ax Cr. to the mouth 10/18/90 40.0-180.0 cfs

reaches 12780851, …0853, …0855, …0857, …0859, …0861, …0863, …0865, …0867, …0869,

…0871, …0873; one order 2 reach and 11 order 3 reaches

Creeks

124850 cert# 72595 Stout Creek from Shellburg Creek to the mouth 10/18/90 1.75-20.0 cfs

reach 23780991; a headwater (order 1) reach

124851 cert# 72596 Rock Cr. from East Fork Cr. to the mouth 10/18/90 3.0-50.0 cfs

reaches 23781441, …1453, …1473, …1485; three order 2 reaches and one order 3 reach

124852 cert# 72597 Mad Cr. from the headwaters to the mouth 10/18/90 2.0-22.0 cfs

reach 23781465; a headwater reach

Previously there had been a different unconverted right at the mouth of the N. Santiam:

1930 at the mouth of the N. Santiam R. 1000-1500 cfs

reaches 23780877, …0879, …0881, …0883; 4 order 5 reaches



Figure 6. Instream water rights in the North Santiam watershed

## Municipal water rights

The CW3M water rights data includes actual municipal water rights for withdrawals from the North Santiam River for 5 urban growth areas (Salem-Keizer, Stayton, Lyons, Gates, and Detroit). Details of these water rights are in the “Muni WR details” tab of the NSantiam.xls file. Maximum diversion rates vary from 62 cfs for Salem down to 1.7 cfs for Gates. Normalized by average 2010-17 simulated population, the maximum diversion rates vary from 164 gal/day/person for Salem-Keizer (pop. 244,141) up to 6,853 gal/day/person for Detroit (pop. 212) (see the “Municipal Water” tab of NSantiam.xlsx).

## Reality check

Observational data for flows and stream temperatures at a number of gage locations in the North Santiam basin is available on a website maintained by the USGS. We have selected these records for use in calibration (7 flow gages and 6 temperature sensors):

|  |  |  |
| --- | --- | --- |
| USGS flow gage | COMID |  |
| 14178000 | 23780591 | NO SANTIAM R BLW BOULDER CRK NR DETROIT |
| 14179000 | 23780701 | BREITENBUSH R ABV FRENCH CR NR DETROIT |
| 14180300 | 23780557 | BLOWOUT CREEK NEAR DETROIT |
| 14181500 | 23780511 | NORTH SANTIAM RIVER AT NIAGARA |
| 14182500 | 23780805 | LITTLE NORTH SANTIAM RIVER NEAR MEHAMA |
| 14183000 | 23780481 | NORTH SANTIAM RIVER AT MEHAMA |
| 14184100 | 23780883 | NORTH SANTIAM R AT GREENS BRIDGE NR JEFFERSON |
|  |  |  |
|  |  |  |
| temperature sensor | COMID |  |
| 14178000 | 23780591 | NO SANTIAM R BLW BOULDER CRK NR DETROIT |
| 14179000 | 23780701 | BREITENBUSH R ABV FRENCH CR NR DETROIT |
| 14180300 | 23780557 | BLOWOUT CREEK NEAR DETROIT |
| 14181500 | 23780511 | NORTH SANTIAM RIVER AT NIAGARA |
| 14182500 | 23780805 | LITTLE NORTH SANTIAM RIVER NEAR MEHAMA |
| 14184100 | 23780883 | NORTH SANTIAM R AT GREENS BRIDGE NR JEFFERSON |

Chart, histogram

Description automatically generated

Chart

Description automatically generated

## Inflows from springs

Spring water inputs have been added to the North Santiam basin simulation at 10 locations. For a discussion of spring water inputs, see the section McKenzie Basin Wetlands Study/High Cascades Springs.

Specifications for springs in CW3M ver. 409

<Spring name="Marion Cr." COMID="23780673" flow\_cfs="41" temp\_C="4.9" />

<Spring name="Pamelia Cr." COMID="23780957" flow\_cfs="23" temp\_C="4.9" />

<Spring name="Whitewater Cr." COMID="23781597" flow\_cfs="15" temp\_C="4.9" />

<Spring name="Horn Cr." COMID="23781739" flow\_cfs="31" temp\_C="4.9" />

<Spring name="Boulder Cr." COMID="23780901" flow\_cfs="120" />

<Spring name="Blowout Cr." COMID="23780557" flow\_cfs="8" />

<Spring name="Breitenbush" COMID="23780701" flow\_cfs="139" />

<Spring name="Breitenbush Hot Spring" COMID="23781273" flow\_cfs="0.42" temp\_C="84." />

<Spring name="French Cr." COMID="23780793" flow\_cfs="155" temp\_C="4.9" />

<Spring name="Little N Santiam" COMID="23780805" flow\_cfs="173" />

## Calibration and simulation skill

Calibration of CW3M for the North Santiam basin, done through June 2021 and implemented in model version 409 has yielded good to very good skill statistics for all 7 flow locations and 5 of 6 stream temperature locations.

We assign grades to the values of the statistics – VG very good, G good, S satisfactory, NS not satisfactory - using the Moriasi skill criteria (Moriasi et al. 2007, 2015). These “Moriasi statistics” are NSE, PBIAS, RSR, and R2. Note that Moriasi’s skill criteria were actually developed for flows, not for temperatures; they are applied here to temperatures for lack of any better well-defined temperature simulation skill criteria. For stream temperatures, we add a fifth statistic, MAE, the mean absolute error. For more information on the Moriasi criteria, see the section McKenzie Basin Wetlands Study/Calibration and simulation skill.

Graphical user interface, table, Excel

Description automatically generated

Four sets of “tuning knobs” have been used to improve CW3M’s simulation skill for flows and stream temperatures in the North Santiam basin: 1) the ET\_MULT parameter in the HBV.csv parameter set used by the HBV precipitation-runoff submodel, 2) the placement, flow rates, and water temperatures of the springs, 3) the W2A\_SLP and W2A\_INT parameters in the HBV.csv parameter set used to infer the temperature of runoff into streams from the air temperature, and 4) the size of the topographic shading effect on the insolation of water in the reservoirs.

The ET\_MULT parameter is a multiplier on evapotranspiration to allow for the fact that the average leaf area is poorly known in the forested uplands. Real and simulated evapotranspiration varies seasonally, so adjustments to ET\_MULT affect the seasonal pattern of flows as well as their annual averages.

Discharges from springs are represented as constant – the same amount of water is added at the same temperature every day of the year, without any seasonal variation. The amount of flow is initially set to approximate the summer low flow in the associated reach, but may be tuned to a different value during the calibration process. Locations, water temperature, and flow rates of simulated springs in the North Santiam basin were given in a previous section.

Tuning knob values in CW3M version 409 for watersheds and reservoirs in the North Santiam basin

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HBVCALIB | ET\_MULT | solar\_radiation\_multiplier | W2A\_SLP | W2A\_INT |
| 10 DET | 1.23 | 0.9 | 0.45 | 4.77 |
| 37 Mehama | 0.15 |  | 0.66 | 3.34 |
| 44 Lower North Santiam | 1.63 |  | 0.77 | 2.08 |
| 50 Little North Santiam | 0.35 |  | 1.00 | 0.96 |
| 51 Blowout Creek | 0.1 |  | 0.67 | 2.56 |

## Analysis of tuning knob values and locations where the model has low skill

As of CW3M version 409 (June 2021), Moriasi skill grades were satisfactory to very good for simulations of flows at all of the 7 reaches in the North Santiam which were analyzed. Corresponding grades were good to very good at 5 of the 6 reaches which were analyzed. There is a spreadsheet on a preceding page with the details.

The skill grade was NS (not satisfactory) for %bias of stream temperature at Greens Bridge location near Jefferson on the lower North Santiam. Simulated temperatures there averaged 2.0 degrees C lower than measured temperatures for 2010-18.

A number of the tuning knob values are not plausible. For example, ET\_MULT for the Blowout Creek watershed is only 0.1, meaning that the model’s simulated values for evapotranspiration have to be reduced by 90% in order to increase the watershed discharge to improve the simulation relative to the gage record. A second example is the W2A\_SLP and W2A\_INT values (1.00 and 0.96, respectively) for the Little North Santiam, meaning that the temperature of the runoff into the Little North Santiam is generally about one degree C higher than the air temperature.

## Simulated North Santiam basin water budget and thermal energy budget

Entries for “added by model” in these budgets reflect heuristics in the model code to deal with mathematical anomalies (e.g. division by zero). The model is built to handle free-flowing perennial streams. The math in the model encounters anomalies when stream reaches dry up or freeze up. Under those conditions, the model adds water or thermal energy as necessary to maintain minimum flow rates of liquid water.

The water and energy budgets are said to “close” when the “total in” value matches the “total out” value in each budget. As of CW3M version 409 (7/1/21), the water budget closes nicely but the thermal energy budget is still out by an appreciable amount..

In the water budget, the average yearly total out is 14 mm more than the average yearly total in, a difference of 0.57%. However, a closer look at the underlying data reveals that at the beginning of 2010, the total amount of water in the reaches, reservoirs, and soil tracked by the model was equivalent to 538.1 mm over the area of the basin, while 9 years later at the end of 2018 the analogous amount was 524.4 mm, a decrease of 13.8 mm over the 9 year period, or 1.5 mm/year. Taking that change in the total volume of water into account increases the discrepancy from 13.8 mm (0.57%) to 115.4 mm

(0.73%).

As of 7/2/21 CW3M ver. 409, for the period 2010-18, CW3M simulates this water budget for the North Santiam basin.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Inputs** | annual mmH2O | % of total inputs | cms | % of basin discharge |
| precipitation | 2094 | 85.8% |  |  |
| springs | 332 | 13.6% | 20.0 | 18.8 % |
| wells | 6 | 0.3% |  |  |
| added by model | 8 | 0.3% |  |  |
| total in | 2441 | 100.0% |  |  |
|  |  |  |  |  |
| **Outputs** | annual mmH2O | % of precipitation | cms | % of basin discharge |
| basin discharge | 1767 | 84.4% | 106.3 | 100.0 % |
| evapotranspiration | 597 | 28.5% |  | 33.8% |
| snow “evaporation” | 92 | 4.4% |  | 5.2% |
| piped out | 0 | 0.0% |  | 0.0% |
| total out | 2455 | 117.2% |  |  |
| discrepancy, mm | 14 |  |  |  |
| discrepancy, % | 0.6% |  |  |  |

The energy budget fails to close by about +0.5%, i.e there is 0.5% more modeled thermal energy leaving the surface waters of the basin and in the basin discharge than there is modeled thermal energy reaching the surface waters.

As of 7/9/21 CW3M ver. 414, for the 9 years 2010-18, CW3M simulates this yearly thermal energy budget for the stream network, including reservoirs, in the North Santiam basin.

|  |  |  |  |
| --- | --- | --- | --- |
| **Inputs** | volume, m3 | average temperature, °C | thermal energy, MJ |
| runoff from land surface | 2.737E9 | 7.1 | 78.38e9 |
| springs | 0.631e9 | 5.0 | 13.04e9 |
| insolation of reaches |  |  | 35.66e9 |
| insolation of reservoirs |  |  | 61.40e9 |
| added by model | 0.016e9 | 12.8 | 0.832e9 |
| total in | 3.384e9 |  | 189.31e9 |
|  |  |  |  |
| **Outputs** | volume, m3 | average temperature, °C | thermal energy, MJ |
| basin discharge | 3.353e9 | 7.3 | 101.72e9 |
| municipal withdrawals from surface waters | 0.0018e9 | 8.4 | 0.06e9 |
| irrigation withdrawals from surface waters | .0094e9 | 12.0 | 0.47e9 |
| longwave radiation from reaches |  |  | 13.27e9 |
| evaporation from reaches | 0.0085e9 |  | 21.01e9 |
| longwave radiation from reservoirs |  |  | 17.83e9 |
| evaporation from reservoirs | 0.0145e9 |  | 35.93e9 |
| total out | 3.387e9 |  | 190.29e9 |
| discrepancy | +0.003e9 |  | +0.98e9 |
| discrepancy, % | +0.1% |  | +0.5% |

# McKenzie Basin Wetlands Study

## Project Overview

The EPA has funded the Lane Council of Governments (LCOG) to carry out a study which includes use of CW3M to simulate the McKenzie River basin (Figure 7). The purpose of using CW3M in this study is to simulate changes in the extent of McKenzie basin wetlands and changes in their nature, over the next four decades as climate changes. One effect of wetlands which is of particular interest is on the temperature of water returning to the stream network from the downstream end of the wetlands. As of the beginning of the project, CW3M did not have a functional stream temperature model. The stream temperature model was added in late 2020. The project scope includes re-implementing in CW3M some of the Heat Source/Shade-a-lator model (Boyd & Kasper 2003) used by the Oregon Water Resources Department, to provide simulations of daily stream temperatures along with daily flows. That work was completed in February 2021, but as of that time the Reach layer had not yet been updated to support the Shade-a-lator logic more appropriately.

At the start of project in June 2020, LCOG subcontracted the work to Land Craft, LLC, which is operated by David Richey. During the first year of the project, Oregon Freshwater was a subcontractor to Land Craft. David Conklin did most of the work for Freshwater. The Freshwater-Land Craft contract stipulated that the additions and refinements to CW3M which Freshwater developed for this study would be placed in the public domain as part of CW3M. Kyla Zaret, a wetland ecologist at the Institute for Natural Resources, is providing expertise about wetlands to the project.

The subcontract between Land Craft and Oregon Freshwater was terminated at the end of the first year of the project by Oregon Freshwater. As of 7/14/21, it is expected that Oregon Freshwater will continue its work on the project as a contractor to LCOG. The McKenzie wetlands study is expected to be completed in 2022.

Kyla Zaret’s 1/23/20 report on another project (Zaret 2020) lists six types of wetlands benefits: flood attenuation, late season flow provision, habitat for at-risk species, temperature, nitrogen and phosphorus reduction, and groundwater recharge. The LCOG McKenzie Wetlands project addresses stream temperature, and may perhaps make use of the model of flood attenuation described in Zaret’s report.

![A screenshot of a map

Description automatically generated]()

Figure . McKenzie basin in CW3M screen capture

![A close up of a map

Description automatically generated]()

Figure . Wetlands in the lower McKenzie basin, with their WETL\_ID attribute values

## Model and Simulation Overview

2010-18 is being used as a calibration period. Observational data for 2019-20 is being compared to simulation output for those years, to assess model skill. Future simulations will be made for 2021-2060.

CW3M’s IDU layer for the McKenzie basin currently has 16,883 polygons with a total area of 330,708 hectares. Its reach layer has 1047 reaches. There are 954 HRUs, representing contiguous smaller areas which drain to a single reach or to several adjacent contiguous reaches. HRUs are made up of IDUs; HRU boundaries are coincident with IDU boundaries.

In older versions of CW3M, water comes into reaches three ways: 1) from upstream reaches, 2) as lateral flow from streambanks, or 3) from sources identified explicitly in the Flow XML file (e.g. high Cascades groundwater). This project adds a fourth way: 4) flow from inundated wetland IDUs into adjacent reaches.

Every reach is associated with a single HRU (Reach layer attribute HRU\_ID). Some or all of the drainage from the associated HRU flows into the reach. The water draining from a single HRU may be divided up among several reaches. Reach layer attribute HRU\_FRAC specifies how much of the drainage from the HRU associated with the reach goes into the reach.

The amount of wall clock time that it takes to execute CW3M is roughly proportional to the numbers of IDUs and reaches in the study area. At the current sizes, simulation of the McKenzie basin for 2010-60 takes about four and a half hours on a moderately fast desktop PC.

We anticipate that new, more detailed IDU, reach, and HRU layers may be constructed during the course of the project. If the new reach layer were based on the most detailed version of the National Hydrography Dataset (NHD), it would contain about 13,000 reaches. We expect to identify hundreds to thousands of wetlands in the McKenzie basin. We think the new IDU layer will have less than 100,000 polygons. For comparison, the WW2100 IDU layer had 180,000 polygons, and each full length (90-year) simulation of a single WW2100 scenario took several days to execute.

CW3M has a 3-layer land use/land cover (LULC) hierarchy. The top level has just 8 categories (Unknown/Developed/Agriculture/Other veg./Forest/Barren/Wetlands/Water snow ice). At the beginning of the project, the top and middle levels in the hierarchy each had only a single Wetlands category (LULC\_A = 6 and LULC\_B = 61), and the bottom level had just two, Woody Wetlands (VEGCLASS = 190) and Herbaceous Wetlands (VEGCLASS = 195). Sixty-five IDUs totaling 519 hectares were classified as wetlands. All but four of the wetland IDUs were classified as woody wetlands; those totaled 500 hectares. The other four IDUs, classified as herbaceous wetlands, were near the point where the McKenzie flows into the Willamette.

As of July 2021, we intend to replace the single top-level Wetlands classification (LULC\_A 6) with a more specific classification, “Palustrine non-isolated wetland”, LULC\_A 62. This will leave open the possibility of adding other wetland classes at the LULC\_A level in the future (e.g. isolated wetlands, lacustrine wetlands, riparian wetlands). The palustrine non-isolated wetland class will be subdivided into 8 LULC\_B categories such as “aquatic bed”, “emergent”, and “deciduous scrub-shrub. The 8 LULC\_B categories will in turn be subdivided into 57 VEGCLASS categories. This wetland classification is described in the spreadsheet file DataCW3M/CW3MdigitalHandbook/CW3M\_LULC\_NWI\_crosswalk.xlsx.

## McKenzie wetlands in the initial versions of the IDU, HRU, and Reach data layers

There is an Excel spreadsheet named “Wetlands.xslx” in the CW3M Digital Handbook folder. It has data for wetlands in both the McKenzie and Clackamas basins. Initial data for the McKenzie shows 18 wetlands (WETL\_ID 1 through 18; Figure 8) ranging in size from 7 to 162 ha and totaling 519 ha. For comparison, Clackamas data has 12 wetlands (WETL\_ID 19 through 30), ranging in size from 5 to 38 ha and totaling 204 ha.

## Data changes for better representation of wetlands

Note that CW3M has a convention that the LULC attributes of an IDU may change only from one year to the next, not multiple times within a single year.

1. Apply the wetland LULC\_A, LULC\_B, and VEGCLASS hierarchy specified in the file CW3M\_LULC\_NWI\_crosswalk.xlsx in the \trunk\DataCW3M\CW3MdigitalHandbook folder of the CW3M repository.

2. Develop a shapefile which represents historic, current, and potential future wetlands, divided into polygons by LULC\_B values.  Intersect that shapefile with the existing IDU shapefile to produce a new IDU shapefile with more IDUs, where the boundaries of the IDUs in the original IDU shapefile are coincident with the outer boundaries of contiguous sets of 1 or more IDUs in the new IDU shapefile.

3. Populate initial condition data attributes in the new IDU layer with data for those attributes in the original IDU layer, except assign LULC\_A, LULC\_B, and VEGCLASS attribute values for wetland IDUs using the new classes developed in the first step.

## Projection, Calendar, and Units

CW3M’s spatial data files use the NAD\_1983\_UTM\_Zone\_10N projected coordinate system.

CW3M can aggregate results from daily to yearly on either a calendar year basis or a water year basis. For this study, a water year basis seems more appropriate. Climate datasets begin on January 1st and end on December 31st. Simulations also begin on January 1st and end on December 31st, regardless of whether the simulation results are aggregated on a calendar basis or water year basis. This has the consequence that when a 51-year simulation run is carried out, water year results are presented for only 50 complete water years, starting with October of the first calendar year and ending with September of the final calendar year. For example, when CW3M is run for the 41 (calendar) years 2020-2050, it produces results for only 40 complete water years, 2021-2050. Water years are named by the calendar year in which they end.

CW3M’s native units are metric. In the CW3M output files, areas will usually be presented in hectares, depths in meters, volumes in cubic meters, flows in cubic meters per second (cms), temperatures in Celsius, and precipitation in millimeters of water. Some exceptions will be made to facilitate comparisons with data from other sources, e.g. USGS gage data in cubic feet per second (cfs).

## Simulation of changes in wetlands over time

CW3M will be used in this study to address the question of how McKenzie basin wetlands might change over the next half-century. In order to produce results which are relevant, the model must be able to produce changes in the wetlands in response to other changes, both natural and anthropogenic. Natural changes as used here are changes to the precipitation regime, seasonal temperatures, and so on. Anthropogenic changes would include development encroaching on wetland, changes in the upstream flow regime due to changes in reservoir operations or irrigation withdrawals, and so on. As of the beginning of the study, CW3M treats IDU wetlands as static: once a wetland, always a wetland. Development in wetlands is not represented in CW3M’s land use model, nor is the loss of wetlands due to drainage, drought, or conversion to agriculture. At the very least, we will need to implement logic to tie the areal extent of a wetland to the quantity of water available to nourish it, as precipitation and from upstream. And we will need new logic to tie the temperature and volume of the water flowing out to the temperature and volume of the water flowing in.

## How wetlands are represented in the model

In CW3M, the areas and shapes of IDUs are fixed in the model’s initial data and do not change during the simulation. This has the consequence that, in a given simulation year, an IDU may be classified (the LULC\_A attribute) as a wetland in its entirety or as not-a-wetland (e.g. agricultural or forested) in its entirety, but not as a combination of wetland and not-a-wetland. From one simulation year to the next, the classification of an IDU can be changed, for example from wetland to agricultural or vice versa. As noted previously, as of the beginning of this project, in CW3M wetlands are static; there is no logic to change an IDU’s LULC\_A attribute from wetland (LULC\_A=6) to anything else, nor any logic to change from something else to the wetland value. New logic to do so is discussed later in the section titled “Loss (or gain) of wetlands”.

The condition of a real wetland changes both seasonally and interannually. We’ll add a WETNESS attribute to represent how wet a wetland IDU is on a given day. The value of the WETNESS attribute will be set in the daily simulation loop. The WETNESS attribute is discussed below in the section titled “A WETNESS attribute”.

The representation of a wetland IDU most consistent with the overall CW3M design is as a flat-bottomed pool with soil at the bottom and vertical sides of uniform height. The vegetation type and water depth are taken as uniform across the area of the IDU. This representation can support outputs for water temperature and water depth whenever the wetland is inundated, and the degree to which the soil is saturated when it is not inundated. A more elaborate representation of a wetland IDU would be as a bowl with a curved bottom, so that increasing fractions of the IDU’s area are inundated as the water level rises, but we won’t attempt that representation in this project.

Conceptually, each wetland IDU adjacent to a reach may be visualized as a pool adjacent to a channel. When the wetland is wet and flow in the channel is low, water moves out of the pool into the channel by overflowing the side of the pool, and it can also move through the soil to the channel. When the wetland is dry and flow in the channel is higher, water can move from the channel into the pool, over the banks of the channel. CW3M does not have a mechanism for moving water laterally out of the channel directly into the soil of the adjacent IDU.

The pool has a capacity (idu WETL\_CAP, in units of depth). In this simple wetland model, we parameterize the reach with two values, a threshold flow (reach Q\_CAP, in cms), and a fraction (reach QSPILL\_FRC). Water always flows longitudinally down the channel, but at flow rates above Q\_CAP, the portion of the flow above Q\_CAP is divided between flow down-channel and flow laterally over the sides of the channel into the adjacent pool.

A wetland IDU separated from a stream by another wetland IDU may be visualized as a second pool adjacent to the first but on the other side from the channel. The rim of the second pool is at the same height as the first pool, but the bottom of the second pool may be higher than the bottom of the first pool, so that the capacity per unit area of the second pool is smaller than the capacity per unit area of the first pool. The two pools are connected at the level of the bottom of the second pool, so the water level is the same in the two pools, as long as it is high enough in the first pool to reach the level of the bottom of the second pool.

The conceptualization described so far allows for nourishment of the wetland idus from a nearby reach, but it does not account for lateral flow into the wetlands from the soil of adjacent non-wetland IDUs. Nor does it account directly for a wetland slowing the runoff from upland precipitation. CW3M uses a version of the HBV precipitation-runoff model. HBV, at least as in CW3M, represents only lateral flow from the land into the stream, not from one parcel of land to an adjacent parcel. Even with these omissions, the conceptualization described so far may result in reducing the hydrograph peaks, since some of the water will detour into the adjacent pools on its way downstream.

Elaboration of CW3M’s precipitation-runoff submodel to include a representation of lateral runoff or subsurface flow between adjacent IDUs would entail changing the principal unit of computation from HRUs to the IDUs of which the HRUS are composed. This would amount to a major extension (or replacement) of the HBV submodel, and is outside the scope of the current project.

Another consequence of the current conceptualization is the inability to represent wetlands which are not adjacent to a reach. A single real wetland may be represented by multiple wetland IDUs forming a contiguous area, so long as at least one IDU is adjacent to a reach. Wetland #18, consisting of a single IDU (IDU\_ID=149851) in an HRU (HRU\_ID=2344) which drains to a reach of McGowan Creek (COMID=23773619), is an example of a wetland IDU not adjacent to a reach nor connected by adjacency to other wetland IDUs.

## The Wetland class and Wetland objects in the C++ code

For coding convenience, a Wetland class has been defined in the model’s C++ code. A Wetland object represents a single contiguous wetland area on the landscape. Each Wetland object has a list of the IDUs which make up the wetland area that it represents; in that sense, a Wetland is composed of the Wetland IDUs which make it up. The Wetland class has a method, Q2Wetland(water\_parcel), which distributes water coming to the wetland from the associated reach to the IDUs in the wetland.

## Topology of wetlands, HRUs, IDUs, and reaches

The IDUs which make up a wetland must be contiguous. A wetland may extend over multiple IDUs and HRUs. A given IDU can only be a part of a single wetland. Moreover, all the wetland IDUs in a given HRU must be part of a single wetland; that is, a given HRU may contain all or part of at most one wetland. An HRU must always contain at least one non-wetland IDU.

HRUs drain to one or more contiguous reaches. A given reach is associated with only one HRU. A given IDU drains to only one reach. One or more IDUs may drain to a reach, but all the IDUs which drain to a given reach must be part of the same HRU. All the wetland IDUs which drain to a given reach must be part of the same wetland.

## Attributes of interest in the wetlands study

A number of attributes of the IDU, HRU, and Reach data layers may be used as inputs and outputs in the wetland simulations. An initial set of such attributes is:

WETL\_ID, WETNESS, WETL\_CAP, WETL2Q in the IDU layer (new)

Q\_CAP, QSPILL\_FRC, Q2WETL, Q\_DISCHARG in the reach layer (new)

Q in the Reach layer, the flow in the outlet reach of the wetland

REACH\_H2O in the Reach layer, the volume of water in the reach

TEMP\_H2O in the Reach layer (new), reach water temperature

PRECIP in the IDU and HRU layers, precipitation

TEMP in the IDU and HRU layers, air temperature

AWS, SM\_DAY, and SOILH2OEST in the IDU layer, discussed below

## A WETNESS attribute

A new IDU attribute, WETNESS, is proposed as a generalization of the attributes for soil moisture. The calculation of WETNESS would likely make use of existing attributes for soil water holding capacity (AWS) and soil moisture (SM\_DAY and/or SOILH2OEST). Positive values of WETNESS are simply average water depth when the surface is covered by water. A value of zero indicates fully saturated soil with no standing water. The magnitude of negative values of WETNESS indicates how much water would have to be added to the soil to fully saturate it.

The idea of a WETNESS attribute was suggested by the use of negative “water depths” by Poiani and Johnson in their prairie wetland model (Poiani & Johnson 1993). Here is their description:

…negative “water depths” were calculated for certain cells in the GIS. This approximated depth to groundwater for vegetation types that were not permanently flooded or saturated. For example, suppose the water elevation over the entire wetland basin was 557.7 metres above sea level (m.a.s.l.). A cell in the center of the basin with a ground elevation of 557.0 m.a.s.l. would have a water depth of 0.7 m. In contrast, a cell at the edge of the wetland with an elevation of 558.0 m.a.s.l. would have a “water depth” (or depth to groundwater) of -0.3 m.

The proposed WETNESS attribute is the same as water depth for positive values, but differs from Poiani’s negative water depth in value. Poiani’s negative water depth is referenced to the elevation of groundwater, which is presumably the elevation of the water level in the nearest portion of their study area that has standing water. Negative values of the WETNESS attribute would instead be calculated from soil water holding capacity and an estimate of current soil moisture, probably using existing IDU attributes AWS, SM\_DAY, and SOILH2OEST.

Existing IDU attribute AWS is described in the CW3M data dictionary as “average soil water holding capacity” from SSURGO, in units of cmH2O. We can adjust the AWS values for wetland IDUs to tune our wetland model.

Existing IDU attribute SM\_DAY is the amount of water held in the soil, in mm, as tracked by the CW3M’s precipitation/infiltration/runoff submodel. It can change daily. Conceptually it applies to the soil from the surface down to a point too deep to be accessible by plants and too deep for water to flow laterally into the nearest downhill stream reach. SM\_DAY does not include aquifer water.

Existing IDU attribute SOILH2OEST is an estimate of plant available soil moisture, in mm of water, calculated during the growing season separately for the irrigated and unirrigated parts of HRUs containing agricultural IDUs.

The proposed WETNESS attribute is a floating point variable with meaningful values in the range of about +3000 to -1000, with 0 indicating mud, i.e. fully saturated soil with no standing water and little oxygen. Positive values of WETNESS represent the average depth of water covering the soil, in mm. Negative values of WETNESS in principle could be derived from the SM\_DAY attribute and the soil water holding capacity (AWS):

when there is no standing water, WETNESS = SM\_DAY - AWS

This formulation allows interpretation of the magnitude of negative WETNESS values as a measure of how much more water the soil will absorb before water begins to accumulate on the surface.

## Calculating the exchange of water between the wetland and the reach

Conceptually, flow out of the reach into adjacent wetlands is modeled as a diversion from just upstream of the downstream end of the reach. The flow from the reach to the wetland is not included in the value of the discharge at the downstream end of the reach. Flow from subreach to subreach within the reach is modeled without recognition that in reality the discharge from a subreach is diminished by the overbank spillage within the subreach. Flow back into the reach from the wetland over the top of the streambank is modeled like other lateral flows.

### Wetland IDU parameters

WETNESS – a variable representing how wet the wetland (mmH2O)

WETL\_CAP – a parameter representing the depth of water (mmH2O) at which the water overflows back to the associated reach, assuming Q < Q\_CAP

WETL2Q – a variable representing the overflow from the wetland back to the reach, cms

### Reach parameters

Q – total flow out of the reach both downstream and over the banks into adjacent wetlands, cms

Q\_DISCHARG – flow downstream at the downstream end of the reach to the upstream end of the adjacent reach, cms

Q\_CAP – below this flow rate, all the water exits downstream; above this rate, a fraction spills into the adjacent wetland (cms)

QSPILL\_FRC – the fraction of Q above Q\_CAP which spills into the adjacent wetland

Q2WETL – a variable representing the flow over the banks of the reach into the wetland, calculated at the end of the reach loop in ReachRouting::SolveReachKinematicWave()

Under all conditions, Q = Q\_DISCHARG + Q2WETL.

When Q > Q\_CAP and WETNESS < WETL\_CAP, then

WETL2Q = 0 and

Q2WETL = (Q – Q\_CAP) \* QSPILL\_FRC

Q\_DISCHARG = Q – Q2WETL

When both Q > Q\_CAP and WETNESS>WETL\_CAP, a flood condition exists. See the “Flooding” section below.

When Q < Q\_CAP and WETNESS > WETL\_CAP, then WETL2Q is > 0 and Q2WETL = 0.

WETL2Q = WETNESS – WETL\_CAP

### Flooding

When the flow in the reach goes over its banks (Q > Q\_CAP) and the wetland is already at capacity (WETNESS > WETL\_CAP), a flood condition exists. As of March 2022, Q\_CAP is only specified for reaches adjacent to wetlands, and CW3M does not have a general representation of flooding. Flooding of wetlands is represented by allowing water to overfill the wetland at GMT\_REACH time in the latter part of a daily timestep, and then draining the wetland back to WETL\_CAP in the initial part of the next daily timestep. Such floodwater is simulated as leaving the reach on the first day at the stream temperature on the first day, and returning to the reach at the temperature of runoff water the next day, so flooding affects the thermal energy in the stream in a crude way.

## Implementation of the exchange of water between the wetland and the reach

There is a mismatch between the scale at which HBV in CW3M tracks water (HRUs) and the scale at which the CW3M wetland model represents wetlands (IDUs). In the WW2100 project, a similar scale mismatch between HBV and irrigation was accomodated by adding an “irrigated soil water” compartment (BOX\_IRRIG\_SOIL) to the soil “layers” recognized by HBV. This addition allowed some IDUs within a given HRU to be irrigated, and others not irrigated. The wetland model accomodates both wetland and non-wetland IDUs within a single HRU, and tracks the different conditions from one wetland IDU to another, through the use of IDU attributes: LULC\_B, VEGCLASS, WETNESS, and WETL\_CAP. Maintaining overall water balance, however, requires tracking water at the HRU level in HBV. To fit within the existing CW3M/HBV/HRU structure, standing water in wetland IDUs is aggregated into the “meltwater” (BOX\_MELT) compartment CW3M’s HBV implementation. Since the amounts in that compartment may be a mixture of meltwater at some IDUs and standing water at other IDUs, it has been renamed as the “surface water” compartment (BOX\_SURFACE\_H2O), distinguishing it both from the snowpack’s frozen water and the water in the soil. In the source code, BOX\_MELT and BOX\_STANDING\_H2O both refer to the BOX\_SURFACE\_H2O compartment.

CW3M’s algorithms already make use of simplifying assumptions about the flowing water in the reaches: that the reaches never freeze and never stop flowing. The wetland implementation will make use of additional simplifying assumptions: 1) any given IDU may have standing water or may have snow on the surface, but not both at the same time, and 2) when snow falls on standing water, it melts. A consequence of these assumptions is that all the standing water must be absorbed into the soil before a snowpack can form, in a wetland IDU. Liquid standing water in CW3M is represented in computational units of complete IDUs only – wetlands or water bodies; there is no representation of puddles or ponds smaller than an IDU. The situation is similar for snow; there is no representation of snowdrifts or patches of snow smaller than an IDU. Finer spatial resolution could conceivably be achieved by constructing a new IDU layer with more and smaller IDUs.

Two boolean variables, *m\_standingH2Oflag* and *m\_snowpackFlag,* have been added to the HRU class to indicate when any IDUs in the HRU have standing water or snow on the ground. While a single IDU can’t have both standing water and snow at the same time, an HRU may contain IDUs with snow, different IDUs with standing water, and/or IDUs with neither. Thus at the HRU level, any combination of states of *m\_standingH2Oflag* and *m\_snowpackFlag* may occur during a simulation. The HRU’s snow compartment (BOX\_SNOWPACK) accounts for the aboveground frozen water in all the IDUs in the HRU. The HRU’s surface water compartment (was BOX\_MELT, now BOX\_SURFACE\_H2O) accounts for the aboveground liquid water in all the IDUs other than those representing reservoirs or other substantial bodies of water. The model has no representation of ice in the soil, nor of surface ice on standing or flowing water.

Logic has been added to HBV::HBVdailyProcess() for wetland IDUs somewhat analogous to that for SOILH2OEST for IDUs in agricultural use. The new logic maintains the value of the WETNESS attribute and calculates WETL2Q for wetland IDUs when calculating the flow through the soil from the wetland into the reach. Flow in the other direction, through the soil from the reach to the wetland soil, is not represented or accounted for in the model. In effect, the wetland gets its water only from precipitation and from overflow of the banks of the reach, and it loses water through overflow back to the reach and from flow through the soil back to the reach as calculated in the HBV submodel.

As of mid 2021, HBV in CW3M does not have any explicit provisions for what happens to standing water on the soil surface; rather, it assumes that excess water will run off immediately to the streams. The wetland model incorporates the assumption that whenever there is standing water, the topsoil is at field capacity, and calculates drainage through the soil accordingly. Each day an amount of water will be withdrawn from the standing water sufficient to restore the soil to field capacity. In addition, standing water above WETL\_CAP will flow back to the stream as WETL2Q.

A calculation of evaporation from the surface of any standing water in the wetland IDUs has been added to the existing ET calculations. The new logic may take advantage of LAI information associated with the wetland states.

Within the Flow daily loop, the order of calculation is:

time daily process

GMT\_START\_STEP

GMT\_CATCHMENT

m\_hruBlock.Integrate(GetCatchmentDerivatives)

IntRK4(GetCatchmentDerivatives)

GetCatchmentDerivatives()

GlobalMethodManager::Step()

HBVdailyProcess()

HRU::WetlSurfH2Ofluxes()

Set FLOODDEPTH to zero and calculate WETNESS

with floodwater included.

Set SM\_DAY.

ApplyDeltaArray()

[EvapTrans::Step(); ApplyDeltaArray()] x 3

RunAltWM(); ApplyDeltaArray()

[Springs(); ApplyDeltaArray()] x ...

Add HRU fluxes to HRU water compartments

(all water compartments are >= 0 here)

ResetHRUfluxes()

Update IDU and HRU attributes based on HRU water compartments.

Copy HBV box values into HRU attributes HruSNOW\_BOX, HruBOXSURF\_M3, HruNAT\_SOIL,

HruIRRIG\_SOIL, HruGW\_FASTBOX, and HruGW\_SLOWBOX

Calculate and store IDU attribute WETL2Q

Calculate and store HRU attributes HruH2OMELT\_M3 and HruH2OSTNDGM3

Calculate and store IDU attributes SM\_DAY, SNOW\_SWE, and H2O\_MELT

CheckSurfaceH2O(pHRU)

GMT\_RESERVOIR Reservoir updates flows in reservoir outlet reaches

GMT\_REACH

ApplyWETL2Q()

pReach->AccumWPadditions() for each IDU

pBOX\_SURFACE\_H2O->AccumWithdrawalsFromHRUlayer() for each IDU

ReachRouting::Step()

SolveReachKinematicWave()

calculate Q2WETL

ResetReachFluxes()

ApplyQ2WETL

ReachH2OtoWetlandIDU()

Wetland logic in FlowModel::InitHRULayers()

* Clear *m\_standingH2Oflag* to false for each HRU except when there is no snowpack, there is water in the meltwater/standing water compartment, and the HRU contains a wetland IDU with WETNESS > 0; in that case, set *m\_standingH2Oflag* to true.

Wetland logic in FlowModel::Run() at GMT\_CATCHMENT time

Call m\_hruBlock.Integrate(), which calls GetCatchmentDerivatives(), which calls GlobalMethod::Run(), which calls HBVdailyProcess().

Wetland logic in HBVdailyProcess() for HRUs and IDUs (GMT\_CATCHMENT time)

* For each HRU, infiltrate wetland standing water from the surface water compartment as if soil water is at field capacity.
* For each wetland IDU with standing water (WETNESS > 0), calculate WETL2Q and call AddFluxFromGlobalHandler() to move surface water to the associated reach.
* For each wetland IDU, update WETNESS. When WETNESS goes to zero or negative for every wetland IDU in a given HRU, clear *m\_standingH2Oflag* for that HRU.

Wetland logic in FlowModel::Run() after m\_hruBlock.Integrate() is called

* For each HRU for which m\_standingH2Oflag is set, use the IDU WETNESS attribute to find and add up the standing water in the wetland IDUs (standing water exists in the IDUs for which LULC\_A = LULCA\_WETLAND and WETNESS > 0). Use that standing water total and the total area of the associated IDUs to partition the water in the HBV surface water compartment between wetland standing water and melt water. Update the SM\_DAY and H2O\_MELT IDU attributes.

Wetland logic in ReachRouting for each reach (GMT\_REACH time)

* Incorporate WETL2Q via the global flux handler.
* Calculate Q2WETL.

Wetland logic in FlowModel::Run() at GMT\_REACH time

* Call GlobalMethodManager::Step(), which calls EvapTrans::Step().
* After GlobalMethodManager::Step() is called, call ApplyQ2WETL().

Wetland logic in EvapTrans for each wetland IDU (GMT\_REACH time)

* Calculate ET from standing water.

Wetland logic in ApplyQ2WETL (GMT\_REACH time)

* Update the surface water compartments of the HRUs.
* For each HRU, set or clear *m\_standingH2Oflag* as appropriate.

## Standing water temperature

At some times, i.e. when WETNESS > 0, wetland IDUs have standing water. As of 8/10/2021, we are modeling the standing water as being at a temperature half way between the daily average air temperature and the temperature at which soil water enters the stream, but not less than the soil water temperature.

## Evapotranspiration from wetlands

When there is no standing water, i.e. when WETNESS <= 0, ET from wetlands is calculated in the same way as ET from upland natural vegetation, using the Penman-Monteith method. When there is standing water, additional evaporation is calculated, in the same way as evaporation from reservoirs, except with the insolation reduced by vegetative shading as a function of LAI, and without any allowance for topographic shading.

## Loss (or gain) of wetlands

In the real world, a wetland which dries up and remains dry for a decade or so can transition to a dryland vegetation type. Wetlands may also be lost through conversion to agricultural use or developed use. Less frequently, wetlands are sometimes restored; the wetland at the junction of Chicken Creek with the Tualatin River is an example. For simulation of anthropogenic land use changes involving IDUs changing into or out of wetland land cover, the “PrescribedLULCs” autonomous process was added in Fall, 2021. PrescribedLULCs is described in the “Autonomous Process Details” chapter of this document. Transitions from wetland natural vegetation states to dryland natural vegetation states can be accomodated by the wetland state and transition model.

## Wetland state and transition model

Pending more-informed definition of wetland transitions by wetland scientists, placeholder transition tables for the wetland STM were implemented in September and October, 2021. The STM used in CW3M is driven by two transition tables, one for deterministic transitions based on stand age alone, and a second one which incorporates probabilities and time since disturbance. An example of a deterministic transition from one wetland state to another would be the growth of woody vegetation over time, resulting in a transition from the PSS7C state to the PFO7C state when the average vegetation height goes from below 6 meters (scrub-shrub) to above 6 meters (forest). For each state, only a single deterministic transition may specified. It is helpful to think of the deterministic transition as what will eventually happen if all the conditions remain the same. Some states (e.g. “climax forest” states) are stable and transition back to themselves when their “ending age” parameter is exceeded.

Examples of upland probabilistic transitions would be a stand-replacing wildfire or mortality from a beetle infestation. However, for the wetland STM, it seems more useful for the likelihood of transition to be affected by the interannual variation in the weather. During an extended drought, wetlands may dry up for more of the year than they normally do, promoting the growth of shrubs or trees that cannot tolerate constant, year-around immersion. Other examples of state changes caused by climate change are easy to imagine. To simulate such climate-dependent transitions, a facility for basing the likelihood of transitions on measures of the wetland’s hydrologic condition will be added to the STM engine.

Three annual measures of a wetland’s hydrologic condition are under consideration, all based on the sequence of daily values of the WETNESS variable:

* the fraction of days in the year when WETNESS >= 0, i.e. the soil surface is saturated or covered by standing water (“WET\_FRAC”)
* the length of the longest sequence of consecutive days when WETNESS >= 0 (“WETLONGEST”)
* the average depth of the standing water on the days when the soil surface is saturated or covered by standing water, the average of all the WETNESS values greater than or equal to zero (“WETAVGDPTH”)

Numeric values for these measures may be calculated at the end of the simulated water year and stored as IDU attributes.

For the purpose of code development, we plan to use a set of transitions for a wetland as it dries up over a period of years. Using the abbreviations for the wetland states from the wetland model spec (Conklin & Zaret 2021), this process can be represented as

PUBF -> PABF -> PEMA -> PSS7A -> PFO7C

where

PUBF is for an open water pond that’s inundated most of the time, but may draw down by the end of some growing seasons, a shallow pond without submerged or floating vegetation.

PABF is semi-permanently flooded wetland, as for a shallow pond with submerged or floating vegetation. The PUBF->PABF transition will occur when a wetland in the PUBF state stays in that state for 3 consecutive years (a deterministic transition).

PEMA is emergent temporarily flooded wetland, that’s inundated briefly at the start of the growing season, as for a marsh or wet meadow. The PABF->PEMA transition will occur when the PABF wetland experiences 2 consecutive years with WET\_FRAC < 1/4, WETLONGEST < 60 days, and WETAVGDEPTH < 1 m.

PSS7A is temporarily flooded wetland, such as a floodplain-based wetland dominated by evergreen shrubs (woody veg < 6 m tall) that’s inundated briefly at the start of the growing season. The PEMA->PSS7A transistion occurs when a PEMA wetland experiences 3 consecutive years with WET\_FRAC < 1/6, WETLONGEST < 45 days, and WETAVGDEPTH < 0.5 m.

PFO7C is seasonally flooded wetland dominated evergreen trees, such as a spruce swamp. The PSS7A->PFO7C transition occurs when a PSS7A wetland stays in that state for 10 consecutive years (a deterministic transition).

The nominal transition threshold values for WET\_FRAC, WETLONGEST, and WETAVGDPTH may be adjusted during development for convenience. Transitions may be added to reflect increasingly wet conditions from year to year as well, if necessary when debugging.

## Activating a wetland in the wetland model

The original LULC\_A value for wetlands, 6, does not activate the wetland model. Wetland IDUs may exist with meaningful WETL\_ID values but with LULC\_A = 6. To activate the wetland model for a given WETL\_ID value, these fields must be populated:

* for each IDU in the wetland, IDU attributes VEGCLASS (i.e. LULC\_C) and WETL\_CAP (mm) must be given values appropriate for the wetland model
* the VEGCLASS values must be propagated up to LULC\_B and LULC\_A; the resulting LULC\_A value should be 62
* for reaches which can overflow into the wetland, the reach attributes Q\_CAP (cms) and QSPILL\_FRC must be assigned appropriate values

Standing up a new wetland will upset the consistency between the water values in the IDU layer and those for the corresponding HRU(s), resulting in warning messages at Init time. The inconsistencies are repaired as part of Init. Saving the IDU, HRU, and Reach layers after a successful Init will eliminate the warning messages in subsequent Inits.

## Reality check

We need a way to assess our model’s skill at simulating the real world. We rely on hindcasting – driving the model with actual weather data and other historical records – and comparing the simulation results with temporally correlated historical observations. A working assumption for our wetland model development is that we will have historical data available for some wetlands, such as water depth as it varies seasonally and interannually, that we can compare to our simulation results. As of 7/15/21, this data has not yet been acquired.

Per Kyla Zaret on 7/17/20:

Regarding historic data:  There's a real dearth of quantitative information pertaining to palustrine wetlands in Oregon (especially those that don't include an open waterbody like a pond, lake or reservoir).  The [National Wetland Condition Assessment](https://www.epa.gov/national-aquatic-resource-surveys/what-national-wetland-condition-assessment) (NWCA) is one program (via the EPA) through which data on water depth, etc. are collected at such sites, but there are very few sites sampled per state relative to the total number of wetlands…  I would expect DEQ and the USGS to be the owners of data pertaining to water flow, temperature and quality (at least of riverine systems).  Around Portland, I'm told that some of the special stormwater districts may collect data at palustrine systems if they own or manage properties containing those types of wetlands.  Could EWEB have such data?

Observational data for flows and stream temperatures at a number of gage locations in the McKenzie basin is available on a website maintained by the USGS. We have selected these records for use in calibration (13 flow gages and 6 temperature sensors):

|  |  |  |  |
| --- | --- | --- | --- |
| USGS flow gage | COMID |  | |
| 14158500 | 23773373 | MCKENZIE RIVER AT OUTLET OF CLEAR LAKE, OR | |
| 14158790 | 23773393 | SMITH RIVER ABV SMITH R RESV NR BELKNAP SPRNGS | |
| 14158850 | 23773359 | MCKENZIE R BLW TRAIL BR DAM NR BELKNAP SPRINGS | |
| 14159200 | 23773037 | SO FK MCKENZIE RIVER ABV COUGAR LAKE NR RAINBOW |
| 14159500 | 23773009 | SOUTH FORK MCKENZIE RIVER NEAR RAINBOW |
| 14161500 | 23773411 | LOOKOUT CREEK NEAR BLUE RIVER | |
| 14162200 | 23773405 | BLUE RIVER AT BLUE RIVER | |
| 14162500 | 23772909 | MCKENZIE RIVER NEAR VIDA | |
| 14163150 | 23772857 | MCKENZIE RIVER BLW LEABURG DAM, NR LEABURG OR | |
| 14163900 | 23772801 | MCKENZIE RIVER NEAR WALTERVILLE | |
| 14164700 | 23774369 | CEDAR CREEK AT SPRINGFIELD | |
| 14164900 | 23772751 | MCKENZIE RIVER ABV HAYDEN BR, AT SPRINGFIELD, OR | |
| 14165000 | 23773513 | MOHAWK RIVER NEAR SPRINGFIELD | |
|  |  |  | |
| temperature sensor | COMID |  | |
| 14159200 | 23773037 | SO FK MCKENZIE RIVER ABOVE COUGAR LAKE NR RAINBO | |
| 14159500 | 23773009 | SOUTH FORK MCKENZIE RIVER NEAR RAINBOW | |
| 14161100 | 23773429 | BLUE RIVER BELOW TIDBITS CREEK NR BLUE RIVER OR | |
| 14162200 | 23773405 | BLUE RIVER AT BLUE RIVER | |
| 14162500 | 23772909 | MCKENZIE RIVER NEAR VIDA | |
| 14164900 | 23772751 | MCKENZIE RIVER ABV HAYDEN BR AT SPRINGFIELD OR | |

Chart, line chart

Description automatically generated

Chart, line chart

Description automatically generated

## High Cascades Springs

More water drains out of portions of the upper McKenzie basin than would be expected from the amount of precipitation which falls (Jefferson et al. 2006). The extra water is conceptualized as coming from springs fed by aquifers which originate from percolation over areas different from the current drainages. In fact, there are substantial springs in the High Cascades, as for example those which feed Clear Lake. However, the available information about the number, locations, flow rates, and possible seasonality of the springs is not detailed enough to attempt direct representation of all the actual springs in CW3M. As a workaround, we prescribe average spring flows at the some of the same reaches where we have USGS gage data. Initially, the flow rate is taken as the average summer low flow rate for 2010-18. The spring flow rate may be further adjusted to reduce the bias in the simulated flows at the gage locations relative to the USGS gage data.

It may be possible in some drainages to run “PEST” calibrations for the HBV precipitation-runoff model, extending the parameter set to include the value for the spring flow. PEST stands for Parameter Estimation. Earlier PEST calibrations for portions of the Willamette basin are described in documents from the WW2100 and OUWIN projects (Conklin 2016, 2019). The numbers in the HBV.csv data file were produced from the various PEST calibrations for different watersheds in the Willamette basin, stretching back to early in the WW2100 project and from time to time since.

Spring water inputs have been added to the McKenzie basin simulation at 8 locations, shown in the table below.

Specifications for springs in CW3M ver. 741

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| gage | COMID | location | temperature, deg C | sim rate, cms | gage low flow, cms |
| 14158500 | 23773373 | Clear Lake | 4.9 | 5.70 | ~6.60 |
| 14161500 | 23773411 | Lookout Cr near Blue R | 4.9 | 0.14 | ~0.35 |
| 14158850 | 23773359 | Below Trail Bridge Dam | 4.9 | 9.67 | ~19.39 |
|  | 23773239 | Foley Springs on W. Fork | 4.9 | 3.31 | ungauged |
| 14161100 | 23773429 | Blue R at Tidbits Cr | same as runoff | 1.60 | not available |
| 14158790 | 23773393 | Smith R above Smith R Reservoir | 4.9 | 0.76 | ~0.15 |
| 14159200 | 23773037 | Above Cougar Reservoir | same as runoff | 6.06 | ~6.40 |
| 14159500 | 23773009 | S. Fork near Rainbow | same as runoff | 1.10 | ~12.41 |
|  |  | total |  | 31.40 cms | ~45-50 |

## Calibration and simulation skill

Calibration of CW3M for the McKenzie basin, done through March 2021 for the EPA-funded study, has yielded satisfactory to very good skill statistics for 7 of 13 flow locations and 3 of 6 stream temperature locations. The flow location exceptions are discussed in a later section of this document. As of March 2021, the temperature location exceptions remain to be analyzed. USGS gage and stream temperature data for the 9 years 2010-18 were used for calibration. Skill statistics were calculated both for the 2010-18 calibration period and for a two-year validation period, 2019-20. A table of the skill statistics is included on a following page.

We assign grades to the values of the statistics – VG very good, G good, S satisfactory, NS not satisfactory - using the Moriasi skill criteria (Moriasi et al. 2007, 2015). Note that Moriasi’s skill criteria were actually developed for flows, not for temperatures; they are applied here to temperatures for lack of any better well-defined temperature simulation skill criteria.

The four statistics recommended by Moriasi et al. (2007, 2015; see especially Table 5, pp 1771-2 in the 2015 pub) are described just below. These “Moriasi statistics” are NSE, PBIAS, RSR, and R2. For stream temperatures, we add a fifth statistic, MAE, the mean absolute error.

Statistical performance measures

* NSE – Nash Sutcliffe Efficiency. Range +1 to –infinity. Bigger is better, +1 is a perfect match. 0 is equivalent to using the mean to predict the individual values. Moriasi describes NSE as “a normalized statistic that determines the relative magnitude of the residual variance (‘noise’) compared to the measured data variance (‘information’)” and cites Nash & Sutcliffe (1970). Moriasi goes on to say that “NSE indicates how well the plot of observed versus simulated data fits the 1:1 line.
* PBIAS – Percent bias. Smaller absolute value is better; 0 is perfect, range is +infinity to –infinity. +10% means the average of the observed values is 10% larger than the average of the simulated values. PBIAS = 100 \* ∑(observed – simulated) / ∑observed. Moriasi’s description is that PBIAS “measures the average tendency of the simulated data to be larger or smaller than their observed counterparts”.
* RSR – Ratio of the root-mean-square-error to the standard deviation of the observations. Smaller is better; 0 is perfect. Range 0 to + infinity. Since the error term is normalized by the standard deviation, RSR is useful for comparing the accuracy of simulation of flows which vary over orders of magnitude, i.e comparing simulation of headwater and upper basin flows with large flows in the valley bottom below the reservoirs, and for comparing the accuracy of simulation between subbasins of different sizes.
* R2 – coefficient of determination. Bigger is better; 1 is perfect. Range is 0 to 1. Moriasi (2007) says “R2 describes the proportion of the variance in measured data explained by the model” and goes on to note that “although R and R2 have been widely used for model evaluation, these statistics are overly sensitive high extreme values (outliers) and insensitive to additive and proportional differences between model predictions and measured data”, citing Legates and McCabe (1999).
* MAE – Mean absolute error. This is average of the absolute values of the differences between the simulated values and the corresponding observed values, expressed in the units of the observations. As such it is useful to inform one’s expectation of the accuracy of the simulation at a given time.

Moriasi et al. descriptive terms (Moriasi et al. 2007, 2015)



Table

Description automatically generated

From “Hindcast skill – Calibration v Validation.xlsx” in the SkillAssessment folder.

As of 7/7/21, two types of “tuning knobs” have been used to improve CW3M’s simulation skill for flows in the McKenzie basin: 1) the ET\_MULT parameter in the HBV.csv parameter set used by the HBV precipitation-runoff submodel, and 2) the placement and flow rates of the springs. There are additional tuning knobs for stream temperatures.

The ET\_MULT parameter was originally added to HBV.csv as a multiplier on evapotranspiration to account for the fact that the average leaf area is poorly known in the forested uplands. Real and simulated evapotranspiration varies seasonally, so adjustments to ET\_MULT affect the seasonal pattern of flows as well as their annual averages.

Generally, discharges from springs are represented as constant – the same amount of water is added at the same temperature every day of the year, without any seasonal variation. The amount of flow from the spring is initially set to approximate the summer low flow in the associated reach, but may be tuned to a different value during the calibration process.

As of 7/7/21, three tuning knobs are available for calibrating stream temperatures: 1) the temperature of water running off the land into the stream reaches, 2) the temperature of water from springs, and 3) the size of the topographic shading effect on the insolation of water in the reservoirs.

Tuning knob values for the McKenzie basin in CW3M version 741

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HBVCALIB | ET\_MULT | solar\_radiation\_multiplier | W2A\_SLP | W2A\_INT |
| 8 CGR | 0.83 | 1.0 | 0.42 | 4.43 |
| 9 BLU | 0.36 | 0.3 | 0.64 | 4.90 |
| 16 McKenzie outlet | 1.91 |  | 0.60 | 4.97 |
| 25 Mohawk | 1.52 |  | 0.60 | 3.37 |
| 34 McKenzie above Walterville | 0.42 |  | 0.46 | 5.23 |
| 46 Clear Lake | 2.07 |  | 0.38 | 3.61 |

## Analysis of locations where the model has low skill

As of 1/3/21 and CW3M version 219, five flow gage locations have unsatisfactory skill grades. Two of the flow locations are downstream of dams; the poor skill is presumably a result of poor representation of the dam operations. That is almost certainly the case at reach 23773359 below Trail Bridge dam, since the Trail Bridge dam is not represented in the model at all. Reach 23773009 on the South Fork near Rainbow is downstream of the Cougar dam. Cougar, one of the USACE Willamette Project dams, is represented explicitly in the model, using current operating curves. Further analysis may allow us to improve the skill at that location.

Two more of the five flow locations with unsatisfactory skill (reaches 23772857 and 23772801) are affected by the Leaburg and Walterville canals, which divert water from the river to produce hydroelectric power, and return it further downstream. CW3M does not include a representation of the canals.

Low simulation skill for Cedar Creek at Springfield (reach 23774369, average gaged flow 0.78 cms for 2010-18) may be due to urbanization of its drainage area. Further analysis could lead to improvements, but the creek contributes less than 1% of the basin discharge, so any improvements are likely to make only a small difference to the picture of the basin as a whole.

## Simulated McKenzie basin water budget and thermal energy budget

Entries for “added by model” in these budgets reflect heuristics in the model code to deal with mathematical anomalies (e.g. division by zero). The model is built to handle free-flowing perennial streams. The math in the model encounters anomalies when stream reaches dry up or freeze up. Under those conditions, the model adds water or thermal energy as necessary to maintain minimum flow rates of liquid water.

The water and energy budgets are said to “close” when the “total in” value matches the “total out” value in each budget. As of CW3M version 409 (7/1/21), the water budget closes nicely but the thermal energy budget is still out by an appreciable amount..

In the water budget, the average yearly total out is 2.5 mm less than the average yearly total in, a difference of 0.11%. However, a closer look at the underlying data reveals that at the beginning of 2010, the total amount of water in the reaches, reservoirs, and soil tracked by the model was equivalent to 844.2 mm over the area of the basin, while 9 years later at the end of 2018 the analogous amount was 862.6 mm, an increase of 18.4 mm over the 9 year period, or 2.0 mm/year. Taking that change in the total volume of water into account reduces the discrepancy from -2.5 mm (-0.11%) to -0.44 mm

(-0.02%).

As of 7/2/21 CW3M ver. 409, for the period 2010-18, CW3M simulates this water budget for the McKenzie basin.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Inputs** | annual mmH2O | % of total inputs | cms | % of basin discharge |
| precipitation | 1890 | 86.8% |  |  |
| springs | 270 | 12.4% | 31.5 | 19.3 % |
| wells | 11 | 0.5% |  |  |
| added by model | 7 | 0.3% |  |  |
| total in | 2179 | 100.0% |  |  |
|  |  |  |  |  |
| **Outputs** | annual mmH2O | % of precipitation | cms | % of basin discharge |
| basin discharge | 1419 | 75.1% | 148.7 | 100.0 % |
| evapotranspiration | 669 | 35.4% |  | 47.2% |
| snow “evaporation” | 80 | 4.2% |  | 5.7% |
| piped out | 8 | 0.4% |  | 0.6% |
| total out | 2176 | 115.1% |  |  |
| discrepancy, mm | -2 |  |  |  |
| discrepancy, % | 0.1% |  |  |  |

The energy budget fails to close by -6.4% or more, i.e. 6.4% or more of the modeled thermal energy reaching the surface waters of the basin is still unaccounted for in modeled thermal energy leaving those waters and in the basin discharge.

As of 7/1/21 CW3M ver. 409, for the 9 years 2010-18, CW3M simulates this yearly thermal energy budget for the stream network, including reservoirs, in the McKenzie basin.

|  |  |  |  |
| --- | --- | --- | --- |
| **Inputs** | volume, m3 | average temperature, °C | thermal energy, MJ |
| runoff from land surface | 3.799e9 | 9.0 | 143.1e9 |
| springs | 0.893e9 | 4.9 | 18.3e9 |
| insolation of reaches |  |  | 56.2e9 |
| insolation of reservoirs |  |  | 32.1e9 |
| added by model | 0.023e9 |  | negligible |
| total in | 4.715e9 |  | 249.7e9 |
|  |  |  |  |
| **Outputs** | volume, m3 | average temperature, °C | thermal energy, MJ |
| basin discharge | 4.693e9 | 7.8 | 153.7e9 |
| municipal and irrigation withdrawals from surface waters | 0.005e9 | <= 15 | ~<= 0.3e9 |
| longwave radiation from reaches |  |  | 17.3e9 |
| evaporation from reaches | 0.013e9 |  | 33.4e9 |
| longwave radiation from reservoirs |  |  | 8.7e9 |
| evaporation from reservoirs | 0.008e9 |  | 20.3e9 |
| total out | 4.713e9 |  | < 233.7e9 |
| discrepancy | -0.002e9 |  | ~-13.7e9 |
| discrepancy, % | -0.04% |  | ~-6.4%% |

## CW3M v. Shade-a-lator

Shade-a-lator output was obtained from Tommy Franzen at The Freshwater Trust (TFT) for comparison to CW3M output. (Despite the similarity in their names, there is no connection between TFT and Oregon Freshwater Simulations.) The Shade-a-lator study area is the final 56.275 km of the McKenzie River, before its confluence with the Willamette River. This stretch of the river was divided into 25 m segments, 2,251 segments in all. Shade-a-lator output was provided for 4 cases:

* current conditions, 11/1/19-3/31/20, 36% canopy density, representative of leaf off
* current conditions, 4/1/20-10/31/20, 75% canopy density, representative of leaf on
* future conditions, 11/1-3/31, 36% canopy density
* future conditions, 4/1-10/31, 75% canopy density

An email dated 12/9/20 from Olivia Duren at TFT described the Shade-a-lator study:

“...TFT has modeled potential shade using Shade-a-lator for the mainstem McKenzie and major tributaries below Leaburg Dam and compared a current conditions scenario to future conditions to estimate solar radiation that would be blocked from reaching the stream as new tree canopy grows. We’ve modeled the same for the Middle Fork and Coast Fork of the Willamette. Both scenarios look at shade produced by a forest buffer strip adjacent to the river that is 60 ft wide. The ‘current conditions’ scenario is based on canopy heights from 2016 LiDAR data... The ‘future conditions’ scenario assumes that any canopy 5m or shorter will grow to 100 ft.”

Shade-a-lator works at finer spatial and temporal scales than CW3M. CW3M’s temporal unit of computation is one day, while the temporal unit for TFT’s Shade-a-lator study was 30 minutes. In its stream thermal energy submodel, CW3M’s spatial unit of computation is the subreach. For each subreach, in CW3M the available data pertinent to the insolation estimate is limited to:

* subreach length, width, and aspect
* a representative VEGCLASS and AGECLASS value for each stream bank, for the reach to which the subreach belongs
* topographic elevation E, S, and W for the reach, in degrees

The LIDAR data provides a more detailed spatial dataset for Shade-a-lator. For each 25 m segment, these data values are present in Shade-a-lator:

* vegetation heights and ground elevations at 9 distances from the stream center in each of 7 directions (NE, E, SE, S, SW, W, NW)
* bottom width, elevation, and gradient

The original intent in the McKenzie wetlands project was to replace the McKenzie Reach GIS layer, inherited from the WW2100 project, with a more detailed version in which the division of reaches into subreaches reflected the shape of the stream on the landscape. The idea was for each subreach to be relatively straight, and suitable to be characterized by single values for direction and for the height of vegetation on each bank. Subreaches would necessarily be of variable length, shorter in curved sections of the stream, and potentially much longer in straighter sections. Preparation of the more detailed Reach layer is outside the scope of Oregon Freshwater’s part of the project and, as of February 2021, still remains to be done. In order to continue development of the thermal loading model, Oregon Freshwater adopted the stream reaches inherited from the WW2100 project as the spatial unit of computation. The TFT Shade-a-lator data for the lower McKenzie has 2,251 stream segments each 25 meters long. The 2,251 Shade-a-lator segments correspond to 86 reaches in CW3M, averaging 654 meters long, a 26:1 difference in spatial scale.

Shade-a-lator calculates the intensity of the incoming solar shortwave radiation at each half hour of the day as it varies seasonally. The shortwave flux is attenuated by its passage through the canopy; the attenuation varies over the course of the day with the sun’s compass direction (azimuth) and height above the horizon (elevation). Shade-a-lator has the ability to take cloudiness into account, but it seems that no cloudiness data was available for use with TFT’s Shade-a-lator study (the Cloudiness column in the Continuous Data tab of the McKenzieRiver\_cc.xlsm spreadsheet is empty).

CW3M gets its shortwave radiation data from the climate dataset (e.g. from the file UofIMETDATA\_surface\_downwelling\_shortwave\_flux\_in\_air\_2010-2019.nc in climate scenario 8 "Baseline\_2019versionOfGridMET\_2010-current"). The values, in W/m2, have already been attenuated by atmospheric conditions, and represent 24-hour averages for each calendar date, of radiation falling on a flat surface without topographic shading. CW3M reduces the shortwave radiation value by taking into account topographic shading. Note however, that the ability to account for topographic shading is dependent on the availability of topographic elevations to the east, south, and west. Reach attributes for the topographic elevations have not been populated with actual data. Doing so requires ArcGIS work which is outside the scope of Oregon Freshwater’s part of the project. For code development purposes, the topographic elevations of reaches corresponding to Shade-a-lator segments were calculated as the averages of the topographic elevations in the Shade-a-lator data.

The information about streamside vegetation is much more limited in CW3M than in Shade-a-lator. CW3M averages the TREE\_HT attributes of the representative streamside IDUs for each bank, and infers the canopy density from the LAI attributes. Attenuation by streamside conditions is estimated as a function of stream width, vegetation height, and canopy density.

Actual thermal loading of a stream by the sun depends not only on the average intensity of the shortwave radiation, but also on the amount of surface area of the water. Shallow streams warm up faster than deeper streams. The stream width (“bottom width”) is part of the Shade-a-lator input data, but is not generally available as input data to CW3M. CW3M estimates the width of each subreach daily as a function of stream order and flow rate.

## Comparison of CW3M results to Shade-a-lator results

Shade-a-lator data in files named “cc75\_kcal.csv” and “fc75\_kcal.csv” was compared to output data from CW3M. “cc75” means “current coditions with canopy density of 75%”; “fc75” represents future conditions with that canopy density. The study period was from 4/1/20 through 10/31/20. Data for the McKenzie River from km 0 up to km 55.285 was used in the comparison, representing reaches 23765583 upstream through reach 23772865. Shade-a-lator data was for 25 m river segments, spanning the distance from km 0 through km 55.300.

CW3M simulations to generate output data for comparison with Shade-a-lator data made use of as much of the Shade-a-lator input data as possible. Reach widths were taken as the average of corresponding segment widths. Similar averages were made of topographic elevations and vegetation heights. LAI in CW3M was inferred from Shade-a-lator’s canopy density using Beer’s law with k = 0.5.

CW3M data in the figures below comparing thermal loading for current and future conditions is from CW3M version 301 on 2/21/21. The figure titled “Daily kCal 75CC CW3M v Shade-a-lator” shows the thermal energy added to the river between km 0 and km 55.3 for each day of the study period, as simulated by CW3M (blue) and from the Shade-a-lator cc75\_kcal.csv data (orange). The CW3M line is irregular because it reflects the reduction of direct sunshine on cloudy and rainy days. Summed over the April 1 – October 31 period, the thermal load calculated by Shade-a-lator on this 55 km section of river is 6% higher than the thermal load calculated by CW3M.

Taller streamside vegetation decreases the thermal load on the stream from direct sunshine. In the “future conditions” simulations, canopy under 5 m tall is replaced with canopy 30.48 m tall (100 ft tall). The figure titled “Daily kCal CW3M CC v FC” shows the thermal energy added to the river each day when shaded by the original vegetation heights (CC) and when shaded by the taller vegetation (FC), as simulated by CW3M. CW3M simulates a reduction of the thermal load by 16%. Finally, the figure titled “Daily kCal Shade-a-lator CC v FC” shows the same comparison using data from the cc75\_kcal.csv and fc75\_kcal.csv Shade-a-lator files. In this case, the reduction in thermal load is smaller, only 4%.







![A screenshot of a map

Description automatically generated]()

Figure . Clackamas basin in CW3M screen capture

# The Clackamas Basin

Data files for the Clackamas basin have been set up to allow its use as a separate study area (Figure 9). Simulations of the Clackamas basin are being run as part of the McKenzie Basin Wetlands Study, for comparative purposes. The Wetlands.xslx spreadsheet, part of the CW3M Digital Handbook, identifies 12 wetlands in the Clackamas basin, in the 0.3.3 release of CW3M.

# Autonomous process details

## Prescribed LULCs

The Prescribed LULCs autonomous process provides a way to prescribe year-to-year changes in land-use-land-cover categories for individual IDUs. As of October 2021, the specific cases handled by the Prescribed LULCs process are:

* loss of part of a wetland, i.e. a wetland IDU (but not the last remaining IDU in the wetland) is converted to a non-wetland IDU
* addition of an IDU to an existing wetland
* addition of an IDU to an urban growth area

These changes are specified in a CSV file, the “LULCtrans” file. Each row of the file specifies an LULC change for a single IDU at the beginning of a certain calendar year. The field names (i.e. column headers) in the LULCtrans file are:

* YEAR, the calendar year of the transition
* IDU\_ID, the ID of the affected IDU
* LULC, the new LULC code, at the most detailed level of the LULC hierarchy, which is LULC\_C (a.k.a. VEGCLASS) for CW3M
* UGB, the new UGB code
* WETL\_ID, the new wetland ID
* DESCRIPTION, a comment string

Integer data is used for all the fields except the DESCRIPTION field, which is a string. By convention, YEAR is the leftmost field and DESCRIPTION is the rightmost field. Rows must appear sorted from earliest YEAR to latest YEAR. Within a given year, entries should be sorted from smallest IDU\_ID to largest IDU\_ID. Values in the LULC, UGB, and WETL\_ID fields may be the new target values or, for no change, a token (-1).

As an example, here is the LULCtrans file which was used in the development of the Prescribed LULCs code:

; Prescribed LULC changes

; LULC values are for the most detailed LULC category, i.e. LULC\_C aka VEGCLASS for WW2100.

; Entries should be in order from earliest year to latest year, and

; within a given year, should be in order from lowest to highest IDU\_ID.

YEAR,IDU\_ID,LULC,UGB,WETL\_ID,DESCRIPTION

2010,33806,6246,22,10,Add one IDU to wetland 10.

2011,33827,122,22,0,One of the IDUs in wetland 10 goes to Developed Low Density.

2013,33827,6246,22,10,Add the IDU back into wetland 10.

Note the use of comments in the initial part of the file. The Prescribed LULCs code uses a method named “ReadCSVwithLeadingComments()” to read the LULCtrans file. That method skips over lines beginning with a semicolon in column 1 at the beginning of the file, and interprets the first line without a semicolon in column 1 as the column headers.

LULCtrans files are specific to simulation scenarios, and are placed in DataCW3M/ScenarioData/<simulation scenario name>. For example, as of October 2021, the LULCtrans file for the Demo\_Baseline scenario is

DataCW3M/ScenarioData/Demo\_Baseline/prescribedLULCs.csv

The code for the Prescribed LULCs process is in the APs Visual Studio project. At runtime, the location and name of the LULCtrans file is specified in the APs.xml file, as for example

<prescribed\_LULCs id='25'

prescribed\_LULCs\_file='ScenarioData/SCENARIO\_NAME/prescribedLULCs.csv' />

In this example, SCENARIO\_NAME is a string recognized by APs::LoadXml() which is replaced at simulation runtime by the name of the simulation scenario. The <prescribed\_LULCs> block in the APs.xml file comes after the <carbon> block and before the <land\_use\_transitions> block. The value for the ID field in the <prescribed\_LULCs> block must match the ID value in the <autonomous\_process> block for the Prescribed LULCs process in the ENVX file.

# Directory structure and file names

The directory structure of the CW3M Subversion repository follows the pattern of many Subversion repositories:

CW3M

branches

tags

trunk

Within the trunk folder, CW3M has 4 subdirectories (a.k.a. folders):

trunk

DataCW3M

GDAL

Installer

SourceCode

The DataCW3M folder in turn has both subdirectories and data files of various kinds:

<study area> folders, e.g. NSantiam, McKenzie, Marys, …

CW3MdigitalHandbook folder

Documents folder

GriddedRecentWeather folder

MonthlyDataOnPRISMgrid folder

Observations folder

RegressionTesting folder

Reservoirs folder

<CSV data files> common to all the study areas

<XML files> common to all the study areas

CW3M\_<study area>.envx files for each study area

Individual study area folders hold folders and files specific to a single study area. For example, here is what is in the NSantiam study area folder:

Observations folder

Outputs folder (not part of the repository, but present on disk)

Flow\_2010.ic initial conditions file

Flow\_NSantiam.xml file

IDU\_NSantiam.shp and associated files for the IDU layer

HRU\_NSantiam.shp and associated files for the HRU layer

Reach\_NSantiam.shp and associated files for the Reach layer

# C++ Source Code Conventions

Source code styles, like natural language writing styles, are idiosyncratic to the author. A large computer program like CW3M is necessarily the work of many programmers over many years, and it is not feasible to constrain the code to a uniform style. Nevertheless, it is useful to identify coding conventions used by major contributors in order to facilitate reading the code.

## Conklin’s conventions for variable names

Prefixes are used to impart information about the variable:

* ‘p’ followed by an upper case letter signifies a local variable which is a pointer to another variable
* “m\_” signifies a member of a class
* “m\_p” followed by an upper case letter signifies a member of a class which is a pointer to another variable
* ‘g’ followed by an upper case letter signifies a variable which is global to the Visual Studios project and which is also a pointer to another variable or object
* “gIDUs” is the name of a global variable in autonomous process DLLs which contains the address of the object representing the IDU layer
* “m\_col” followed by an IDU attribute name signifies the index of that attribute in the IDU layer, i.e. its column number in the DBF component of the IDU shapefile
* “m\_colReach” followed by a Reach attribute name signifies the index of that attribute in the Reach layer
* “m\_colHru” followed by an HRU attribute name signifies the index of that attribute in the HRU layer

## Conklin’s conventions for the IDU layer and its attributes

An *IDUlayer* class is defined as a child of class *MapLayer.* The IDUlayer class contains knowledge of IDU attributes and facilitates reference to individual attributes in the source code by their names. Attribute names are upper case and no longer than 10 characters. The limitation to 10 characters is for compatibility with ArcGIS. Attribute values are of type *double, float,* or *int*. Doubles are preferred over floats.

The *IDUlayer* class definition includes integer member variables which hold the indices (i.e. column numbers) of the attributes in the DBF component of the IDU shapefile. For example, the *IDUlayer* class has an integer member named *m\_colAREA* which holds the index to the AREA attribute.

Visual Studio projects which generate DLLs for CW3M must include a global variable *gIDUs* which holds a pointer to the IDUlayer object representing the IDU shapefile for the current CW3M model instance. The value of *gIDUs* must be explicitly set at Init() time.

The *IDUlayer* class definition includes #defines for IDU attribute names which expand to strings of the form *gIDUs->m\_col*<attribute name>. For example,

#define AREA gIDUs->m\_colAREA

The *IDUlayer* class includes 3 methods for getting attribute values:

double IDUlayer::Att(iduPolyNdx, attributeName)

float IDUlayer::AttFloat(iduPolyNdx, attributeName)

int IDUlayer::AttInt(iduPolyNdx, attributeName)

The *IDUlayer* class includes 3 methods for setting attribute values:

bool IDUlayer::SetAtt(iduPolyNdx, attributeName, double attributeValue)

bool IDUlayer::SetAttFloat(iduPolyNdx, attributeName, float attributeValue)

bool IDUlayer::SetAttInt(iduPolyNdx, attributeName, int attributeValue)

Visual Studios projects which generate DLLs for CW3M may include inline routines with the same names as the 6 IDUlayer:: attribute access routines, to shorten the code. For example, the HBV DLL includes

inline double HBV::Att(int iduPolyNdx, int col) { return(gIDUs->Att(iduPolyNdx, col)); }

These conventions support fetching the value of an attribute with a statement like

double idu\_wetl2q = Att(idu\_poly\_ndx, WETL2Q);

which is equivalent to

double idu\_wetl2q = gIDUs->Att(idu\_poly\_ndx, gIDUs->m\_colWETL2Q);

# Release Log

CW3M\_ChickenCreekInstaller\_0.1.0.exe 12/6/18 82 MB from CW3M ver. 98. Uses HBV parameter values for Chicken Creek from John Dalyrmple’s recent calibration of INFEWS using WillametteINFEWSdemo\_0.1.2 and uniform weighting, together with ET\_MULTIPLIER = 1.7.

CW3M\_ChickenCreekInstaller\_0.1.1.exe 12/12/18 84 MB made from CW3M ver. 105, saved in ver. 106. Renames ET\_MULTIPLIER to ET\_MULT, to facilitate use of the PEST program for calibration. PEST limits variable names to 12 characters. Includes CW3M Digital Handbook folder.

CW3M\_NSantiamInstaller\_0.1.2.exe 12/16/18 110 MB from CW3M ver. 113. Revise N. Santiam instream water rights per Joel Plahn at OWRD.

CW3M\_0.1.3\_Installer.exe 1/28/19 97 MB from CW3M ver. 129. Add Marys River watershed.

CW3M\_Installer\_0.2.0.exe 4/19/19 from CW3M ver. 155. Add options for including leapdays and for reporting on a water year basis. Accommodate multiyear NetCDF climate data files. Eliminate the need for monthly climate data. Add routines for accessing weather data by reach. Skip over reservoirs which are outside the study area when calculating the mass balance. Turn off the UpdateDGVMvegtype autonomous process. Incorporate the version of HBV.csv used in INFEWS. Add climate scenario 8, BaselineGridMultiyearFiles. Add logic for 5-column format for observation files which references the day number to 1/1/1900, to be used for observation files which include leapdays. Add support for running the parameter estimation program PEST on the Marys River study area.

CW3M\_Installer\_0.2.1.exe 4/19/19 from CW3M ver. 156. Add Marys\PEST folder with files to support use of the PEST parameter estimation program on the Marys River basin.

CW3M\_Installer\_0.2.2.exe 5/15/19 from CW3M ver. 160. Fix bugs in the logic which reads the climate data files. Add the new multiyear MIROC5 climate dataset as climate scenario 9 for the Marys basin study area. Add a new “MIROC5\_macav2” simulation scenario. These new scenarios aren’t working yet.

CW3M\_Installer\_0.2.3.exe 5/17/19 from CW3M ver. 162. Fix more bugs in the logic which reads the climate data files.

CW3M\_Installer\_0.2.4.exe 5/17/19 from CW3M ver. 172. Overhaul logic for reading multiyear climate data files, to speed up access to larger files. Add scenarios for macav2metdata MIROC5 and HadGEM-ES 1950-2005 and RCP 8.5 2006-99.

CW3M\_Installer\_0.2.5.exe 5/24/19 from CW3M ver. 176. Correct a bug which causes the Baseline scenario to fail.

CW3M\_Installer\_0.2.6.exe 6/3/19 from CW3M ver. 182. Fix bugs in IGet(), GetTimeIndex(), and GetDailyWeatherfield(). Get the MIROC5 and HadGEM-ES scenarios working. Add the 365dayBaseline scenario, which uses the old single-year MACA training data climate. Baseline uses the new multi-year v2metdata climate dataset, but there are still issues with it.

CW3M\_Installer\_0.2.7.exe 6/4/19 from CW3M ver. 184. Fix a bug in GetJulianDay() which was causing the 365dayBaseline scenario to crash in 2005. The Baseline and 365dayBaseline scenarios are both working now, but precip from the multi-year v2metdata climate files used by Baseline is higher than precip in the original single-year MACA training data climate files, so Baseline gives higher flows in the Marys River basin than 365dayBaseline does.

CW3M\_Installer\_0.2.8.exe 6/8/19 from CW3M ver. 186. Put in a workaround for the fact that the origin of the climate data grid in the multi-year v2 climate datasets is in the SW corner of the grid, but GDAL still calculates gridcell indices as if the origin were still in the NW corner as in the original single-year climate files.

CW3M\_Installer\_0.2.9.exe 6/14/19 from CW3M ver. 190. Enter HBV parameter values for HBVCALIB=23 Marys from the most recent calibration, but with all available digits of precision. Manually change K2 for HBVCALIB 23 from 0.000124 to 0.0006 to avoid a long term buildup of water in the slow flow groundwater compartment of the HBV model. Add files to allow for simulation of the complete Willamette River basin.

CW3M\_Installer\_0.2.10.exe 6/17/19 from CW3M ver. 194. Update Flow initial conditions files. The new files were made by spinning up the Marys and WRB with the BaselineGridMultiyearFiles climate dataset for 31 years (1979-2009), and by spinning up just the Marys River basin using the MIROC5\_20th\_century and HadGEM-ES\_20th\_century climate datasets for 56 years (1950-2005).

CW3M\_Installer\_0.2.11.exe was withdrawn because of errors in the script which created it.

CW3M\_Installer\_0.2.12.exe 12/10/19 from CW3M ver. 6 on GitHub. Includes the LOG\_Q attribute in the reach layer.

CW3M\_Installer\_0.3.0.exe 6/24/20 from CW3M ver. 16 on GitHub. Logic has been added to create the LOG\_Q and RH attributes in the reach layer, the GRID\_INDEX and RH attributes in the IDU layer, and the RH attribute in the HRU layer. Includes the data files for the McKenzie River basin, which have been updated to include the new attributes. Fix a bug in the “multirun” feature in the logic for running from a command line; this fix was contributed by Maria Wright on the OUWIN project. Comment out the maxDaysInYear specification in the CW3M\_McKenzie.envx file, and update the climate scenario specification in Flow\_McKenzie.xml. This version is the starting version for the Lane Council of Governments Wetlands project funded by the EPA.

CW3M\_Installer\_McKenzie\_0.3.1.exe 6/24/20 from CW3M ver. 17 on GitHub. An installer script has been added which includes the data for the McKenzie basin, but leaves out the data for the Willamette River basin outside the McKenzie basin.

CW3M\_Installer\_McKenzie\_0.3.2.exe 7/7/20 from CW3M ver. 25 on GitHub. Add the Clackamas basin. Get the McKenzie working with future climates. Add a Required Input page to the Data Dictionary.

CW3M\_Installer\_McKenzie\_0.3.3.exe 7/23/20 from CW3M ver. 27 on GitHub. Add and populate the WETL\_ID attribute in the McKenzie and Clackamas IDU layers. Add Wetlands.xlsx and the WW2100 PNAS paper to the CW3M Digital Handbook folder.

CW3M\_Installer\_McKenzie\_0.3.4.exe 8/1/20 from CW3M ver. 31 on GitHub. Add reports on wetland outflows and at the Hayden Bridge gage location. Populate the HBV.csv table for the lower McKenzie and for the Mohawk basin.

CW3M\_Installer\_McKenzie\_0.4.0.exe 10/30/20 from CW3M ver. 87 on GitHub. First draft of stream temperature code based on HeatSource 7.0.

CW3M\_Installer\_McKenzie\_0.4.1.exe 11/23/20 from CW3M ver. 124 on GitHub. Stream temperatures are being simulated, although not yet with the effects of vegetation or streambank shading. A coarse calibration has been done. Reservoir data has been updated for Blue River and Cougar.

CW3M\_Installer\_McKenzie\_0.4.2.exe 12/8/20 from CW3M ver. 156 on GitHub. Stream temperatures and reservoir temperatures are being simulated, although not yet with the effects of vegetation, streambank, or terrain shading. The Flow initial conditions file now saves water temperatures as well as water volumes. The temperature of water running off the land into the stream reaches is being estimated as a function of air temperature. More calibration has been done.

CW3M\_Installer\_McKenzie\_0.4.3.exe 12/27/20 from CW3M ver. 206 on GitHub. McKenzie stream flows and temperatures have been calibrated to USGS gage data for 2010-18. Special logic was added for the springs which feed Clear Lake. The special logic provides for a constant spring flow of 5.7 cms plus a seasonally-varying additional spring flow of 0.8351 times the amount simulated as flowing into Clear from upstream. The Demo and Baseline scenarios have been combined into a single Demo\_Baseline scenario. This release includes files for the Clackamas and North Santiam basins as well as for the McKenzie.

CW3M\_Installer\_McKenzie\_0.4.4.exe 1/8/21 from CW3M ver. ~232 on GitHub. More of the stream temperature logic is working now. A lot of details about the flow and stream temperature models have been added to the digital handbook. The McKenzie flow and stream temperature models were calibrated one more time. Preliminary estimates of the stream flow and thermal energy budgets for the McKenzie basin have been made, and added to the digital handbook. Files and a folder related to running PEST on the BLU drainage have been added. Files for the McKenzie and North Santiam basins are in this release, but the files for the Clackamas basin have been left out.

CW3M\_Installer\_McKenzie\_0.4.5.exe from CW3M ver. 245 1/20/21. Most of the work since the 0.4.4 release was in support of running PEST calibrations. One functional improvement was moving the specification of climate scenarios into its own separate file, now identified in the <scenarios> block of the ENVX file.

0.4.6 was cancelled.

CW3M\_Installer\_McKenzie-0.4.7.exe from CW3M ver. 267 1/30/21. This release was made to facilitate PEST calibration work for Lookout Creek, South Fork above Cougar, and the Mohawk basin. This release also incorporates the addition of a “studyAreaName” field to the ENVX file, and the separation of FLOW report specs from the Flow XML file into a another file.

CW3M\_Installer\_McKenzie-0.4.8.exe from CW3M ver. 295 2/19/21. Add code to reproduce Shade-a-lator output as much as is practical. Also, add the files for the Marys River basin as a courtesy to David Rupp for his work with the City of Corvallis.

CW3M\_Installer\_McKenzie-0.4.9.exe from CW3M ver. 302 2/22/21. Convert a non-essential but inconvenient warning message into a log message.

CW3M\_Installer\_McKenzie-0.4.10.exe from CW3M ver. 305 3/4/21. For David Rupp running Marys River, fix data errors in the specification for climate scenario 9 MIROC5 20th century. For David Holman, turn on Land Use Transitions in PEST\_Smith47.

CW3M\_Installer\_McKenzie-0.4.11.exe from CW3M ver. 306 3/4/21. Incorporate new PEST-calibrated parameter values for Smith47.

CW3M\_Installer\_McKenzie-0.4.12.exe from CW3M ver. 353 4/14/21. For David Holman running PEST to recalibrate the NSantiam basin. Has logic for wetland ET, but doesn’t yet have the code working for flows between the reaches and the wetlands.

CW3M\_Installer\_McKenzie-1.0.0.exe from CW3M ver. 488 9/9/21. First release package since making CW3M officially publicly accessible.

CW3M\_Installer\_McKenzie-1.1.0.exe from CW3M ver. ~564 9/26/21. Add the WetlandSTM yearly and daily processes.

CW3M\_Installer\_McKenzie-1.2.0.exe (209 MB) from CW3M ver. 612 10/21/21. Add the Prescribed LULCs yearly process.

CW3M\_Installer\_1.2.0.exe (333 MB) from CW3M ver 613 10/21/21. This installer has shapefiles, etc, for Clackamas, Marys, McKenzie, NSantiam, and WRB.

CW3M\_Installer\_McKenzie\_1.2.1.exe (209 MB) from CW3M ver. 651 11/6/21. This release includes a new calibration of the Wiley Creek watershed in the South Santiam basin. Code was added to support leading comments in CSV files; the new Prescribed LULCs process uses this code. Added logic to populate the WETL\_CAP attribute automatically when initializing wetland objects, if WETL\_CAP values are not already present. Activated wetland 15 in the wetland model with Q\_CAP = 300 cms and QSPILL\_FRC = 0.9 for its outlet reach (COMID 23772753). Added a No\_wetlands\_demo scenario.

CW3M\_Installer\_1.2.1.exe (320 MB) from CW3M ver 651 11/6/21. Like CW3M\_Installer\_McKenzie\_1.2.1.exe, except includes shapefiles, etc, for Clackamas, Marys, McKenzie, NSantiam, and WRB.

CW3M\_Installer\_1.2.2.exe (322 MB) from CW3M ver 744 2/17/22. Update HBV.csv parameter values for HillsCreek1, FallCreek4, COT6, DOR5, FRN7, the UpperWRB, Marys, BLU9, and McKenzie. Add logic to accommodate basin-specific water rights files. Incorporate new water rights data for the McKenzie basin from David Richey. Reduce nominal low flow from 10 liters per sec to 1 liter per second to support David Rupp’s recalibration of the Marys basin, still in progress.

# References

Abatzoglou JT (2012) Development of gridded surface meteorological data for ecological applications and modelling. *Int J Climatol* 33:121–131.

Allen RG, Pereira LS, Raes D, Smith M (1998) Crop evapotranspiration – guidelines for computing crop water requirements – FAO irrigation and drainage paper 56. *FAO Rome* 300:6541*.*

Annear R, McKillip M, Khan SJ, Berger C, Wells S (2004). Willamette River Basin Temperature TMDL Model: Boundary Conditions and Model Setup. Portland State University School of Engineering and Applied Science, Department of Civil and Environmental Engineering. Technical Report EWR-01-04. 522 pages.

Boyd M and Kasper B (2003). Analytical methods for dynamic open channel heat and mass transfer: Methodology for heat source model Version 7.0. <https://www.oregon.gov/deq/FilterDocs/heatsourcemanual.pdf>

Chow VT (1959). Open Channel Hydraulics. McGraw-Hill, New York, NY.

Chow VT (1964). Handbook of Hydrology.

Also <https://cedb.asce.org/CEDBsearch/record.jsp?dockey=0020129>

and <https://ascelibrary.org/doi/pdf/10.1061/%28ASCE%290733-9429%282002%29128%3A1%2820%29>

https://book4sales.com/book-details.php?book=58f6

Conklin DR (2016). WW2100 HBV Parameter Calibration, 9/15/2016. Oregon Freshwater Simulations, unpublished. In the CW3M repository at trunk/DataCW3M/CW3MdigitalHandbook/WW2100 HBV Parameter Calibration.docx.

Conklin DR (2019). OUWIN Chicken Creek Calibration, 8/29/2019. Oregon Freshwater Simulations, unpublished. In the CW3M repository at trunk/DataCW3M/CW3MdigitalHandbook/OUWIN Chicken Creek Calibration.docx.

Conklin DR and Zaret K (2021). Specification for Wetland Persistence Model, 7/30/21. Unpublished. In the CW3M repository at trunk/DataCW3M/CW3MdigitalHandbook/Specification for Wetland Persistence Model.docx.

Cowardin LM and Golet FC (1995). US Fish and Wildlife Service 1979 wetland classification: A review. Vegetatio 118: 139. <https://doi.org/10.1007/BF00045196>

Daly, C., Halbleib, M., Smith, J.I., Gibson, W.P., Doggett, M.K., Taylor, G.H., Curtis, J. and Pasteris, P.P. (2008), Physiographically sensitive mapping of climatological temperature and precipitation across the conterminous United States. Int. J. Climatol., 28: 2031-2064. <https://doi.org/10.1002/joc.1688>

Jefferson A, Grant G, Rose T (2006). Influence of volcanic history on groundwater patterns on the west slope of the Oregon High Cascades. Water Resources Research 42:W12411. <https://doi.org/10.1029/2005WR004812>

Leach JA and Moore RD (2019). Empirical Stream Thermal Sensitivities May Underestimate Stream Temperature Response to Climate Warming. Water Resources Research, 55. <https://doi.org/10.1029/2018WR024236>

Legates DR and McCabe GJ (1999). Evaluating the use of “goodness-of-fit” measures in hydrologic and hydroclimatic model validation. Water Resources Research 35(1):233-241.

Moriasi DN, Arnold JG, Van Liew MW, Bingner RL, Harmel RD, and Veith TL (2007). Model Evaluation Guidelines for Systematic Quantification of Accuracy in Watershed Simulations. Transactions of the ASABE, 50(3): 885-900.

Moriasi DN, Gitau MW, Pai N, Daggupati P (2015). Hydrologic and Water Quality Models: Performance Measures and Evaluation Criteria. Transactions of the ASABE, 58(6): 1763-1785.

Nash JE and Sutcliffe JV (1970). River flow forecasting through conceptual models: Part1. Adiscussion of principles. J. Hydrology 10(3):282-290.

Poiani KA and Johnson WC (1993). A Spatial Simulation Model of Hydrology and Vegetation Dynamics in Semi-Permanent Prairie Wetlands. Ecological Applications 3(2): 279-293 (May 1993). <http://www.jstor.com/stable/1941831>

Seibert J (1997). Estimation of Parameter Uncertainty in the HBV Model. Nordic Hydrology, 28 (4/5): 247-262.

USACE (2019) Willamette Basin Review Feasibility Study/Final Integrated Feasibility Report and Environmental Assessment, USACE Portland District, December 2019. <https://usace.contentdm.oclc.org/digital/collection/p16021coll7/id/13273>

Zaret K (2020). Flood Attenuation Benefits of Wetlands In the Tualatin Basin. MMTWetlands Benefits Project, Institute for Natural Resources, Portland, OR.

# Acronyms and Abbreviations

**AP** – autonomous process

**cms** – cubic meters per second

**CW3M** – Community Willamette Whole Watershed Model

**EPA** – Environmental Protection Agency

**IDU** – Integrated Decision Unit, CW3M’s smallest 2-dimensial spatial unit of computation. IDU polygons tile the study area.

**gridMET** – “gridMET is a dataset of daily high-spatial resolution (~4-km, 1/24th degree) surface meteorological data covering the contiguous US from 1979-yesterday” <http://www.climatologylab.org/gridmet.html>

**HBV** – Hydrologiska Byråns Vattenbalansavdelning model (Seibert 1997)

**HRU** – Hydrologic Response Unit

**LCOG** – Lane Council of Governments

**LULC** – land use land cover

**MOU** – Memorandum of Understanding

**NRNI** – No Reservoirs, No Irrigation

**PEST** – Model-Independent Parameter Estimation and Uncertainty Analysis

Parameter Estimation Program (https://pesthomepage.org/)

**SSURGO** – Soil Survey Geographic Database, from the Natural Resources Conservation Service - National Cartography and Geospatial Center

**USACE** – United States Army Corps of Engineers

**VS** – Visual Studios

From [www.dictionary.com](http://www.dictionary.com)

lacustrine - of or relating to a lake.

palustrine - Relating to a system of inland, nontidal wetlands characterized by the presence of trees, shrubs, and emergent vegetation (vegetation that is rooted below water but grows above the surface). Palustrine wetlands range from permanently saturated or flooded land (as in marshes, swamps, and lake shores) to land that is wet only seasonally (as in vernal pools).

riparian - of, relating to, or situated or dwelling on the bank of a river or other body of water

vernal pool - A seasonal body of standing water that typically forms in the spring from melting snow and other runoff, dries out completely in the hotter months of summer, and often refills in the autumn. Vernal pools range from broad, heavily vegetated lowland bodies to smaller, isolated upland bodies with little permanent vegetation. They are free of fish and provide important breeding habitat for many terrestrial or semiaquatic species such as frogs, salamanders, and turtles.