Changes from PRMS version 5.2.1.1 for 5.2.2

Changed module version dates for modified modules

Removed GSFLOW related code from basin.f90, climateflow.f90, soilzone.f90, subbasin.f90, call\_modules.f90, and setup\_cont.c

**MMF**

setup\_cont.c – removed **gsflow\_output\_file**, **gsflow\_csv\_file**, **creator\_email**, **cbh\_binary\_file**, musroute\_flag, snow\_cbh\_flag, gwflow\_cbh\_flag, dprst\_area\_dynamic, jhcoef\_dynamic, capillary\_module, added **forcing\_check\_flag, parameter\_check\_flag default changed to 0, added no\_snow\_flag, cbh\_active\_flag, snarea\_thresh\_dynamic, dynamic\_soil\_param\_log\_file,**

**changed imperv\_stor\_dynamic default to dynimpervstor**

**(removed dyn\_ag\_flag**

**Parker updates to address compile time warnings and string sizes:**

batch\_run\_functions, declparam.c, declvar.c, defs.h, protos.h, read\_control, read\_params,

removed unused mmf functions interfaces in protos.h:

matinv, matind, snort, dprint, dpstr, dpint4, dplong, dpreal, dpfloat, dpdble, oprint, opint4, oplong, oplong, opdble, opreal, rosopt, opinit, bdry, param, coropt, sub1, tcale, unscal, sub3, stats, uprint, closeuserfiles, ustr, upint4, uplong, upreal, upfloat, updble, sensitivity, save\_control, save\_vars, esp\_batch\_run, rosenbrock\_batch\_run,

No changes: strmflow, frost\_date, ddsolrad, potet\_hamon, potet\_jh, convert\_params, temp\_dist2

consistent external and intrinsic declarations, add missing: potet\_pan, transp\_tindex, temp\_map, basin\_summary, obs, temp\_1sta\_laps, prms\_summary, nsegment\_summary, soltab, precip\_1sta\_laps, nhru\_summary, basin, ide\_dist, basin\_sum, xyz\_dist, water\_use\_read, muskingum\_lake, routing, glacr\_melt, cascade, call\_modules, utils\_prms, stream\_temp, gwflow, srunoff, water\_balance

spelling corrections: basin\_summary, nsegment\_summary, utils\_prms, gwflow

remove unused USE variables: nsub\_summary, routing

Use “:” instead of 1 to pass two-dimensional arrays which is more portable; affected potet\_pt, potet\_pm, soltab, nsub\_summary, nsegment\_summary, map\_results, xyz\_dist, glacr\_melt, snowcomp

remove initialization of unnecessary variables: prms\_summary, muskingum, obs, subbasin, intcp, routing, climate\_hru, climateflow, gwflow, srunoff, soilzone, snowcomp, water\_balance

hru\_impervstor moved from srunoff to flowvars: subbasin

dprst\_stor\_hru moved from srunoff to flowvars: subbasin

soil\_moist\_tot moved from soilzone to flowvars: subbasin

soil\_zone\_max moved from soilzone to flowvars: subbasin

temp\_set routine in climateflow.f90 – Added **forcing\_check\_flag** to optionally check *tmax* < *tmin*, *tminf* < -150 and *tmaxf* < 200. This change affects temp\_1sta, temp\_laps, temp\_sta, xyz\_dist, ide\_dist, temp\_map, temp\_dist2, and climate\_hru.

find\_header\_end routine modified to remove code specific to reading CBH Files; it reads input files to the first line that starts with ####. This routine is in utils\_prms.f90 and is used in dynamic\_param\_read.f90, dynamic\_soil\_param\_read.f90, nhru\_summary.f90, precip\_map.f90, temp\_map.f90, and water\_use\_read.f90.

find\_current\_time routine in utils\_prms.f90 modified to remove code related to **cbh\_binary\_flag**. This affects climate\_hru.f90, precip\_map.f90, and temp\_map.f90

nhru\_summary, glacr\_melt, and dynamic\_param\_read

basin: always allocate dprst\_frac, check hru\_frac\_perv <0.00001 instead of < 0.00099, error message changed IF ( Hru\_frac\_perv(i)<0.00001 ) THEN

PRINT \*, 'ERROR, pervious fraction must be >= 0.00001 for HRU:', i

PRINT \*, ' pervious fraction is 1.0 - hru\_percent\_imperv - dprst\_frac'

precip\_dist2: rain\_mon: Monthly (January to December) factor applied to rain on each HRU; snow\_mon: Monthly (January to December) factor applied to snow on each HRU; psta\_mon: 'Monthly (January to December) factor applied to precipitation

climateflow: add pref\_flow\_stor, soil\_lower\_stor\_max, soil\_moist\_tot, soil\_zone\_max, hru\_impervstor, dprst\_stor\_hru, glacrb\_melt, ssstor\_init\_frac, soil\_moist\_init\_frac, soil\_rechr\_init\_frac, soil\_rechr\_max\_frac, PRMS\_IT0\_VARS: It0\_basin\_ssstor, It0\_basin\_soil\_moist, It0\_dprst\_stor\_hru, It0\_soil\_moist, It0\_hru\_impervstor, It0\_pkwater\_equiv, and It0\_ssres\_stor to address circular dependencies, only declare basin\_lakeevap when nlake>0; use ‘glacr\_melt’ instead of MODNAME for declaring glacier variables, similarly for some soilzone module and intcp variables; only declparam for humidity\_percent for potet\_pm, potet\_pt and humidity\_cbh\_flag = 0; adjmix\_rain: Monthly (January to December) multiplicative factor to adjust rain proportion in a mixed rain/snow event; basin\_tsta min from 1 to 0; radj\_sppt: Multiplicative adjustment factor for computed solar radiation for summer day with greater than; radj\_wppt Multiplicative adjustment factor for computed solar radiation for winter day with greater than; bug declparam for hru\_pansta when nevap>0, not just when potet\_pan; will use in intcp now; soil\_moist\_max: max from 20 to 30; soil\_rechr\_max from 20 to 30; soil\_moist\_init\_frac: Initial fraction of available water in the capillary reservoir (fraction of soil\_moist\_max) for each HRU; declare, get then deallocate parameter variables: soil\_rechr\_max\_frac, soil\_moist\_init\_frac, soil\_rechr\_init\_frac, ssstor\_init\_frac instead of using arrays that they initialize; compute basin\_soil\_moist and basin\_ssstor as need values to set It0 variables; only get humidity\_percent for potet\_pt, potet\_pm and humidity\_cbh\_flag = 0; if glacier\_flag = 1 initialize glacrb\_melt, glacier\_frac, alt\_above\_ela, and glrette\_frac. Add code to temp\_set for forcing\_check\_flag = 1; only print error and stop for tmin<MINTEMP or TMAX>MAXTEMP when forcing\_check\_flag = 1; add argument Ihru and code for forcing\_check\_flag = 1; if tmaxf-tminf <0.0001 set to 0.0001, instead of < NEARZERO

Potet\_pm\_sta: changed comment say Humidity instead of Hru\_humidity\_sta

muskingum\_lake.f90 – BUG FIX: lake\_transfer was subtracted instead of added to lake\_outflow (water use issue).

Glacr\_melt: only set function value to 0 at start of function, rather than to 1 at start and 0 at end

Cascade: set default cascade\_tol to 0

Gwflow: when gwstor < 0 (maybe water use issue), set gwstor = 0 with message and then do other computations, used to set gwflow and gwres\_sink = 0 and not set gwstor = 0

Ide\_dist and xyz\_dist: adjust\_snow: Monthly (January to December) multiplicative snow'//

+ ' downscaling adjustment; adjust\_rain: Monthly (January to December) multiplicative rain'//

+ ' downscaling adjustment

Precip\_1sta\_laps: rain\_adj : min 0.5 to 0.2, Monthly (January to December) multiplicative factor to adjust measured rain on each HRU to account for differences in elevation, and so forth; snow\_adj, min from 0.5 to 0.2, max from 2.5 to 5 Monthly (January to December) multiplicative factor to adjust measured snow on each HRU to account for differences in elevation, and so forth

Soltab: cleaned up warning message about solt<0, deallocate hru\_lat if glacr\_melt inactive

Potet\_hs – hs\_krs: & 'Monthly (January to December) multiplicative adjustment factor used in Hargreaves-Samani potential ET computations for each HRU

Potet\_pt: pt\_alpha: Monthly (January to December) multiplicative adjustment factor used in Priestly-Taylor potential ET computations for each HRU

Transp\_frost – add initialize transp\_on and basin\_transp\_on

Utils\_prms: remove cbh\_binary\_flag from find\_current\_time, find\_header\_end, add find\_cbh\_header\_end; removed checking for Iret/=2 as this can't happen

\*\*\*

Call\_modules: remove USE ISO\_FORTRAN\_ENV, use all of PRMS\_CONSTANTS, increased EQULS strig to 74 “=”;, add dynamic\_soil\_flag, control: forcing\_check\_flag, no\_snow\_flag, seg2hru\_flag, remove snow\_cbh\_flag and gwflow\_cbh\_flag, modules dynamic\_soil\_param\_read, segment\_to\_hru, compute execution\_time\_start and end converting integers to real, print github commit hash, print available modules tables only if print\_debug > -1; don’t check module function values for /= 0 as they always equal 0. Explicitly specify module defaults (not needed); add error check for endday < startday. For special model modes CLIMATE, TRANSPIRE, WRITE\_CLIMATE, POTET, FROST, always return, i.e., don’t call modules later in sequence for clean, decl, init and still call summary\_output, list of summary variables needs to be correct in control file, don’t call summary\_output for FROST; add error check for nhru=nssr=ngw; add print of “ERROR READING RESTART FILE” instead of just stopping; if snarea\_curve\_flag = 1, set ndepl to nhru and ndeplval = ndepl\*11; remove GSFLOW code;don’t preset module names, as they are set to default in setup\_cont;

routing.f90 – do mod of segment\_type,100 only when segment\_type > 99., only compute Cfs2acft in init, removed from run; seg\_length and segment\_slope and value checks for muskingum\_mann and stream\_temp here, removed from stream\_temp; set special\_seg\_type\_flag to 1 if any seg\_type > 0; added back initializing special segment type variables

climate\_hru: add dimension ncbh(Number of values in each CBH File (active HRUs)), control cbh\_active\_flag, parameter cbh\_hru\_id (HRU identification number associated with each value in CBH File) to provide capability for CBH files to only specify active HRUs; values for inactive HRUs are set to -999.0, check precip units against constant MM instead of CELSIUS. BUG FIX: removed **chb\_binary\_flag** as use of binary CBH Files does not work. Identify climate\_hru as module for all processes specified by CBH file, added print\_module for humidity\_hru, albedo\_hru, cloud\_cover\_hru, and windspeed\_hru, had others. an error occurred in reading CBH file(s). Add check for *tmin* < *tmax* as input in CBH Files and adjusted *tmin* < *tmax* when **forcing\_check\_flag** = 1. climate\_hru is printed as module used when using CBH Files of *humidity*, *albedo*, *cloud\_cover*, and/or *windspeed*. Add routine find\_cbh\_header\_end that reads a CBH to the first line that starts with ####, which is a modification of routine find\_header\_end. If error reading, print ERROR in climate\_hru with ierr then stop with ERROR\_cbh, if humidity\_cbh\_flag is 1, always read CBH values and don’t use parameter values, add up errors using variable ierr and print message and stop if > 0;

Snow\_cbh\_adj: Monthly (January to December) multiplicative adjustment factor to

Rain\_cbh\_adj: Monthly (January to December) multiplicative adjustment factor to

Potet\_cbh\_adj: Monthly (January to December) multiplicative adjustment factor to

Tmax\_cbh\_adj: Monthly (January to December) additive adjustment factor to maximum air temperature for each HRU

Tmin\_cbh\_adj: Monthly (January to December) additive adjustment factor to minimum air temperature for each HRU

climate\_hru: add subroutine read\_cbh\_values to replace duplicate code reading tmax, tmin, hru\_ppt, potet, humidity\_hru, albedo\_hru, and cloud\_cover\_cbh; add implicit none to routines that didn't have it; cbh\_hru\_id description cbh\_hru\_id changed to HRU identification number associated with each value in CBH File;

Prms\_time: add setting of antecedent It0 variables for It0\_soil\_moist, It0\_ssres\_stor, It0\_basin\_ssstor, It0\_basin\_soil\_moist, It0\_dprst\_stor\_hru, It0\_hru\_impervstor, It0\_pkwater\_equiv; use constants YEAR, MONTH, DAY instead of 1, 2, 3 for startday. USE PRMS\_FLOWVARS, ONLY: Soil\_moist, Ssres\_stor, Basin\_ssstor, &

Basin\_soil\_moist, Dprst\_stor\_hru, Hru\_impervstor, Pkwater\_equiv

USE PRMS\_IT0\_VARS, ONLY: It0\_soil\_moist, It0\_ssres\_stor, &

It0\_basin\_ssstor, It0\_basin\_soil\_moist, It0\_dprst\_stor\_hru, &

It0\_hru\_impervstor, It0\_pkwater\_equiv

precip\_map.f90 – removed declaring read\_cbh\_date as EXTERNAL as not used, and INTRINSIC SNGL

Strmflow\_in\_out, muskingum, muskingum\_lake – only check for segment\_type to set special flow variables when any segment\_type > 0; check made in routing to set special\_seg\_type\_file

Write\_climate\_hru – output variables based on temp\_units and precip\_units instead of always F and inches. Values written as F10.5 instead of E12.4

prms\_summary:, array initialize poi\_gage\_id instead of loop

Setup\_param: hru\_lon min -180, max 180, parameter is not used

Prms\_constants: USE ISO\_FORTRAN\_ENV, ONLY: REAL64 and REAL32, set REALsize and DOUBLEsize for use, add ZERO\_SNOWPACK = EPSILON(0.0D0) and DCLOSEZERO = 1.0D-12

Strmflow\_character: Add variable: seg\_res\_time (Mean residence time of water in each segment), remove CFS\_TO\_CMS and use CFS2CMS\_CONV from PRMS\_CONSTANTS, use DNEARZERO instead of NEARZERO since seg\_outflow is double. Use CLOSEZERO to check seg\_area > 0, instead of NEARZERO, use seg\_length as needed to compute seg\_res\_time

ccsolrad.f90 – BUG FIX: check for divide by zero when *soltab\_horad\_potsw* or *hru\_cossl* = 0.0, needed for Alaska model

Makefile – added segment\_to\_hru and dynamic\_soil\_param\_read, clean up dependencies

obs.f90 – added dimension **nstream\_temp** and observed variable *stream\_temp* (Stream temperature at each measurement station) for use as stream temperature replacement.

temp\_1sta\_laps.f90 – BUG FIX: didn't print the module name correctly when **temp\_module** = temp\_sta.

intcp.f90, check for 0 hru\_ppt and netsnow < CLOSEZERO instead of NEARZERO; and pkwater\_equiv <= 0 instead of < DNEARZERO intcp: set net\_apply and use like net\_rain in srunoff for all irrigation types, was irrigation\_type = 1 or covden = 0, divided net\_apply by hru\_perv

IF ( .not.(It0\_pkwater\_equiv(i)>0.0D0) .AND. netsnow<CLOSEZERO ) THEN ! changed from NEARZERO to CLOSEZERO 11/24/2023

IF ( netsnow<CLOSEZERO ) THEN !rsr, added 3/9/2006, changed from NEARZERO to CLOSEZERO 11/24/2023

IF ( Hru\_ppt(i)<CLOSEZERO ) THEN ! changed from NEARZERO to CLOSEZERO 11/24/2023

basin\_sum.f90 – glacier variables *basin\_glacrb\_melt*, *basin\_glacrevap*, *basin\_gl\_to\_melt*, and lake variable *basin\_2ndstflow* added to computations

**PRMS**

**New modules: segment\_to\_hru, dynamic\_soil\_param\_read**

precip\_form routine in climateflow.f90 – added HRU ID as argument so that the ID is identified in an error message. Added **forcing\_check\_flag** to optionally check *hru\_ppt* < 0, *hru\_rain* < 0, and *hru\_snow* < 0. This change affects precip\_1sta, precip\_laps, xyz\_dist, ide\_dist, precip\_map, and climate\_hru.

call\_modules.f90 - call strmflow\_character only when stream\_temp is active, version set to 5.2.2 11/11/2022, new control parameter **forcing\_check\_flag** and new dimension **nstreamtemp**, call new module dynamic\_soil\_param\_read for soilzone, dprst, and impervious dynamic parameters, removed checking for ierr return code from module functions if the return code can only be 0. When model\_mode equal CLIMATE, TRANSPIRE, POTET, or FROST, check for **model\_mode** CLEAN and return; add check to be sure **start\_time** is before **end\_time**, when **snarea\_curve\_flag** is 1, **ndepl** is set to **nhru** and **ndeplval** is set to **nhru**\*11

climateflow.f90 – added variables dprst\_total\_open, dprst\_total\_open\_out, dprst\_total\_close\_in, dprst\_total\_close\_out, moved pref\_flow\_stor from soilzone module so it can be updated used in dynamic parameters.

ddsolrad.f90 – BUG FIX: check for divide by zero when *soltab\_horad\_potsw* or *hru\_cossl* = 0.0, needed for Alaska model

dynamic\_param\_read.f90 – moved code related to impervious, dprst, and soilzone dynamic parameters to new function dynamic\_soil\_param\_read.f90. Added check for errors reading dynamic parameter files. Check values of active HRUs for changed values instead of all HRUs. This module does not adjust any states.

dynamic\_soil\_param\_read.f90 – new module that reads and checks impervious, dprst, and soilzone dynamic parameters and adjusts associated storage. The log file for this module is named using control parameter **dynamic\_soil\_param\_log\_file**. If storage on impervious fraction with change making impervious fraction = 0, that storage is added to the gravity reservoir instead of capillary reservoir. If storage on dprst fraction with change making dprst fraction = 0, that storage is added to the gravity reservoir instead of capillary reservoir;. Note: error messages are triggered if impervious plus dprst fraction add up to > 0.999; if dprst\_frac > 0 and dprst\_depth\_avg = 0; if **soil\_rechr\_max\_frac** > 1.0 it is set to so that **soil\_rechr\_max** >= 0.00001; if **soil\_moist\_max** < 0.00001 or **soil\_rechr\_max** < 0.00001 they are set to 0.00001. BUG FIXES: a) if variable *soil\_moist* is updated, *basin\_soil\_moist* is reset so that water balance is computed correctly; b) if variable *slow\_stor* is updated, *ssres\_stor* and *basin\_ssstor* are reset so that water balance is computed correctly; c) if variable *slow\_stor* is updated, *ssres\_stor* and *basin\_ssstor* are reset so that water balance is computed correctly. c) if dprst volumes are updated, *dprst\_stor\_hru* is reset so that water balance is computed correctly.

snowcomp.f90 – intrinsic function SNGL used when it shouldn't be to compute pk\_den; use local variable *hruarea\_dble* set to **hru\_area\_dble**, instead of referencing array hru\_area multiple times computing basin variables to be more efficient. Set double variable snarea\_thresh\_dble, variables Pk\_temp, Pk\_dev, Tcal, Glacr\_pk\_def, Glacr\_pk\_temp; check for Ai>0.0D0 instead of > DNEARZERO; add routine snow\_states\_to\_zero so pk is initialized consistently; add intcp\_changeover as part of net\_rain area available water; cals and icals now double; set pptmix\_nopack outside main do loop; use ZERO\_SNOWPACK instead of DNEARZERO for checking Pkwater\_equiv <= 0.0D0, they are same value EPSILON(0.0D0); if new snow and pkwater\_equiv <= 0.0D0, instead of < DNEARZERO; at end of do loop check for pkwater <ZERO\_SNOWPACK instead of <=0.0D0. Check for glacier pkwater > ZERO\_NEARZERO instead of > 0.0D0. double local variables for new double varialbes (caln, calpr, calps, calnd, dif (in caloss)); add more explicit mixed precision computations; at end of calin, if pkwater\_equiv<=0.0D0 instead of only setting pk\_den to 0.0, call snow\_states\_to\_zero and set glacier\_states\_to\_zero for glacier HRUs; in snowball double ts, pk\_defsub, pkt, pks, and qcond; in snowevap double cal, ez; bug fix found by James; add Pk\_ice to Freeh2o if Pk\_ice < 0.0, check for freehs0 < 0.0D0; bug: if snow\_evap < 0, set Pkwater\_equiv to 0 outside check print\_debug > -1; check Ai>0.0D0 instead of DNEARZERO in snowcov; ; in init set Ai = 0.0D0 only for not using initial conditions file;

soilzone.f90 - added variable *hru\_perv\_actet*, added checks to be sure **soil\_rechr\_max** > 0 to compute *basin\_soil\_rechr\_stor\_frac*, initialize fluxes that weren't already initialized to 0 outside HRU loop instead of inside. GSFLOW related code removed. Variable *pref\_flow\_stor* moved to climateflow.f90 so it can be used in dynamic\_soil\_param\_read.f90. In descriptions for **soil2gw\_max**, **ssr2gw\_rate**, and **ssr2gw\_exp** change GWR to groundwater reservoir storage. Maximum value for **fastcoef\_lin** changed to 1.5. *ssres\_stor* is set to *slow\_stor* + *pref\_flow\_stor* in INIT procedure as a precaution. Print warning if *avail\_potet* is less than -CLOSEZERO and if *avail\_potet* < 0 set *hru\_actet* to *potet*. Pass HRU ID to compute\_szactet so the ID can be printed in the warning message for *perv\_actet* > *avail\_actet*; simulation date also printed. Added check to be sure **soil\_rechr\_max** > 0 before being used in division in compute\_actet. BUG FIX: for frozen ground the water that should have gone into the capillary reservoir is add to *sroff*, *hru\_sroffp*, *hortonian\_flow*, *basin\_hortonian*, *basin\_sroff*, and *basin\_sroffp*. BUG FIX: for frozen ground flow Dunnian flow due to infiltration from the land surface is left in preferential flow reservoir and Dunnian flow due to discharge from the gravity reservoir is left in slow storage. Removed commented out code for intermediate state/flux variables not used. Remove basin\_cap\_up\_max; more info in warning for not specifying pref\_flow\_infil\_frac; use pref\_flow\_thrsh to check if preferential flow storage instead of pref\_flow\_den as it is possible to not have pfr storage with pref\_flow\_den > 0; ssres\_stor was not set on initialize, but was set in run; check for soil\_rechr\_max > 0 before using as divisor as it is possible to have no soil recharge zone; bug fixes for frozen ground; initialize several fluxes outside main loop rather than inside; print warning about lake\_evap > potet only when print\_debug > -1; in compute\_szactet print more info if perv\_et warning; if avail\_potet<0, set avail\_potet to 0 AND hru\_actet = potet; only compute preferential flow and storage is pref\_flow\_infil\_frac and pref\_flow\_thrsh > 0; bug if frozen ground, add capwater\_in to sroff, capwater\_maxin\*perv\_frac to hru\_sroffp, and to Hortonian\_flow; if frozen ground, leave any flow from gravity reservoir in pref storage assuming it can’t get to surface, this can mean pref storage can exceed pref\_flow\_thrsh and excess released once when soil thaws; compute cascading interflow and Dunnian flow only if interflow+dunnianflw>CLOSEZERO to reduce computing tiny cascading flow; if frozen ground recompute basin\_hortonian, basin\_sroff, and basin\_sroffp; if soil\_rechr\_max = 0 set pctr = 0; if soil\_rechr < 0 set to 0; allow swales to be frozen, if frozen add available to pref and cap, still no et from soil if frozen; soilzone: fix recompute Basin\_sroff if Dunnian flow as dunnian\_flow is added to sroff; if frozen and not swale add capwater\_maxin to sroff, if swale add capwaterin\_maxin to soil\_moist and soil\_rechr; if frozen and swale and pref\_flow\_maxin to pref\_flow\_stor; swale HRUs can be frozen instead of error

srunoff.f90 – move hru\_impervstor and dprst\_stor\_hru to flowvars, use It0 variables instead of Imperv\_stor\_ante and dpsrt\_stor\_ante, add Sra as module variable; make dprst\_vol\_frac, dprst\_vol\_open\_frac, and dprst\_vol\_clos\_frac double; initialize dprst\_seep\_hru, dprst\_sroff\_hru, dprst\_evap\_hru, dprst\_in, dprst\_stor\_hru outside run main do as dynamic parameters could have changed the HRUs with dprst; initialize hru\_sroffi, hru\_sroffp, imperv\_evap, hru\_impervevap, and contrib\_fraction outside main loop as array assignment instead of by HRU in main do loop; uncomment check for insensitive smidx\_coef and smidx\_exp > 200%, so carea\_max controls ignoring smidx coefs done when parameter\_check\_flag > 0; remove extra check of glacier\_flag; address issue where water use irrigation for the canopy is only applied in srunoff when covden = 0 and irr\_type = 1 (was doing in intcp and srunoff); bug fix; net\_rain was added twice for very rare condition where there was an antecedent snowpack, that evaporates the current snowpack with no snowmelt; this bug affected impervious, dprst runoff, and infiltration; code for availh2o\_tot done outside dprst\_comp; impervious runoff moved to main from compute\_infil; if frozen ground, now assume impervious is frozen and water runs off instead of added to storage, except added to storage if swale; dprst\_in is set to 0 outside main do loop in case dynamic parameter change which HRUs have dprst; check for ca\_fraction < 0 that can happen with round off error and set to 0 if true; if frozen add availh2o\_total\*dprst\_area\_max to runoff; make sure srunoff not < 0, set to 0, if imprev\_stor < 0, set to 0; check net\_ppt-netsnow<=0, not < NEARZERO; check net\_snow<= 0, instead of < NEARZERO; if pervious srunoff < 0 set to 0; check dprst volumes and areas <= 0 instead of < NEARZERO; don’t print if srp and sri <0 in dprst\_init, just set to 0; use glacier\_free (1-glacier\_frac) to limit water available for infiltration and runoff; irrigation water runs off for non-swale HRUs, else added to swale infil comp; add any net\_apply for input to dprst; if frozen add available water to storage, allow seepage, but don’t compute runoff; don’t compute cascading runoff if frozen; set negative runoff, impervious storage, depression storage and area due to round-off error to 0; evaporate from impervious if frozen, didn’t used to; for glacier HRUs compute infiltration for glacier free fraction; don’t print warning if srp<-nearzero (commented); bug don’t compute dprst runoff if swale or frozen; fix add hru\_cfgi\_sroff as surface runoff when frozen soil; set active\_glacier = 0 instead of -1 for default; add setting glacier\_frac variable = 1.0 – glacier\_frac and use to limit availh2o\_total; use It0\_pkwater\_equiv instead of Pkwater\_ante; remove check for apply\_irr\_in\_sroff and always use net\_apply for all HRU fractions;

srunoff: ; fix add hru\_cfgi\_sroff as surface runoff when frozen soil; set active\_glacier = 0 instead of -1 for default; add setting glacier\_frac variable = 1.0 – glacier\_frac and use to limit availh2o\_total and infil; use It0\_pkwater\_equiv instead of Pkwater\_ante; remove check for apply\_irr\_in\_sroff and always use net\_apply for all HRU fractions; assume cascading flow is not active with glaciers; check net\_snow<= 0, instead of < NEARZERO; set Hru\_sroffi and basin\_sroffi after checking for frozen; fix net\_apply is applied to whole HRU instead of just pervious area; fix if frozen Srp is set to availh2o\_total + Infil and infil set to 0; compute evaporation from impervious even when frozen; remove commented out sanity check in cascade calculation; frac\_op\_ar and frac\_cl\_ar put back to single; compute dprst seepage even if frozen; don't compute surface runoff if hru\_type = 3 and not frozen

added code to compute new variables *dprst\_total\_open*, *dprst\_total\_open\_out*, *dprst\_total\_close\_in*, *dprst\_total\_close\_out*; removed extra check of **glacier\_flag**. Change comment text from inch-acres to acre-inches. Add declare of read\_error as EXTERNAL in dprst\_init. For frozen ground dprst storage is assumed to be frozen, thus *dprst\_seep\_hru*, *dprst\_sroff\_hru*, and *dprst\_evap\_hru* are set to 0 and all input water is added to *sroff*. As a precaution for no snowmelt and no snowpack but there was net snow that evaporated, check pptmix\_nopack = 0 to be sure net\_rain isn’t added twice.

stream\_temp.f90 –Added code to allow for replacement of stream temperature for selected segments using parameter **tempIN\_segment** and variable **stream\_temp** read in by obs.f90 with **nstream\_temp** number of values. Bug fix: when strmtemp\_humidity\_flag = 0, seg\_humid was set to humidity\_hru as percent instead of fraction; use seg\_length and seg\_slope from routing and set seg\_length\_km; if error checking nhumid stop with message;

temp\_map.f90 – removed declaring precip\_form and read\_cbh\_date as EXTERNAL as not used.

utils\_prms.f90 – code related to **cbh\_binary\_flag** removed from routines find\_current\_time and find\_header\_end. Added routine find\_cbh\_header\_end for exclusive use reading CBH Files.

water\_balance.f90 – add Dunnian Dprst WB and glacrb\_melt to header of water balance file; add move information to debug print statements; check It0\_pkwater\_equiv> 0.0 instead of > DNEARZERO; use It0 variables where available instead of \_ante that were saved in snowcomp, soilzone and srunoff; bug fix: basin\_dprst\_wb was not initialized for each time step; check It0\_pkwater\_equiv>0.0 instead of > DNEARZERO; bug when checking basin surface water runoff for srunoff module added subtraction of basin\_dunnian as this is calculated in soilzone instead of sroff; bug fix; net\_rain was added twice for very rare condition where there was an antecedent snowpack, that evaporates the current snowpack with no snowmelt; set TOOSMALL to 1.0E-05 instead of 3.1E-05; compute availh2o\_total for srunoff water balance then subtract outflows; add variables and organize BALUNT output for HRU srunoff and HRU water balances; fix add hru\_cfgi\_sroff as surface runoff when frozen soil;

Stream\_temp changes

IF ( declparam( MODNAME, 'lat\_temp\_adj', 'nsegment,nmonths', 'real', &

& '0.0', '-5.0', '5.0', &

& 'Correction factor to adjust the bias of the temperature of the lateral inflow', &

& 'Additive correction factor to adjust the bias of the temperature of the lateral inflow', &

& 'degrees Celsius')/=0 ) CALL read\_error(1, 'lat\_temp\_adj')

ALLOCATE ( Seg\_length(Nsegment) )

IF ( declparam( MODNAME, 'seg\_length', 'nsegment', 'real', &

& '1000.0', '1.0', '100000.0', &

& 'Length of each segment', &

& 'Length of each segment', &

& 'meters')/=0 ) CALL read\_error(1, 'seg\_length')

ALLOCATE ( Seg\_slope(Nsegment) )

IF ( declparam( MODNAME, 'seg\_slope', 'nsegment', 'real', &

& '0.0001', '0.0000001', '2.0', &

& 'Surface slope of each segment', &

& 'Surface slope of each segment as approximation for bed slope of stream', &

& 'decimal fraction')/=0 ) CALL read\_error(1, 'seg\_slope')

IF ( declparam(MODNAME, 'sro\_to\_dprst', 'nhru', 'real', &

& '0.2', '0.0', '1.0', &

& 'Fraction of pervious surface runoff that flows into surface-depression storage', &

& 'Fraction of pervious surface runoff that'// &

& ' flows into surface-depression storage; the remainder'// &

& ' flows to the stream network for each HRU', &

& 'decimal fraction')/=0 ) CALL read\_error(1, 'sro\_to\_dprst')

ENDIF

IF ( PRMS4\_flag==OFF .OR. Model==DOCUMENTATION ) THEN

IF ( declparam(MODNAME, 'sro\_to\_dprst\_perv', 'nhru', 'real', &

& '0.2', '0.0', '1.0', &

& 'Fraction of pervious surface runoff that flows into surface-depression storage', &

& 'Fraction of pervious surface runoff that'// &

& ' flows into surface-depression storage; the remainder'// &

& ' flows the a stream network for each HRU', &

& 'decimal fraction')/=0 ) CALL read\_error(1, 'sro\_to\_dprst\_perv')

IF ( declparam(MODNAME, 'sro\_to\_dprst\_imperv', 'nhru', 'real', &

& '0.2', '0.0', '1.0', &

& 'Fraction of impervious surface runoff that flows into surface-depression storage', &

& 'Fraction of impervious surface runoff that'// &

& ' flows into surface-depression storage; the remaind

IF ( declparam(MODNAME, 'lake\_evap\_adj', 'nmonths,nlake', 'real', &

& '1.0', '0.5', '1.5', &

& 'Monthly potet factor to adjust potet on lakes', &

& 'Monthly (January to December) multiplicative adjustment factor for potential ET for each lake', &

& 'decimal fraction')/=0 ) CALL read\_error(1, 'lake\_evap\_adj')

ENDIF

ALLOCATE ( Slowcoef\_lin(Nhru) )

IF ( declparam(MODNAME, 'slowcoef\_lin', 'nhru', 'real', &

& '0.015', '0.0', '1.0', &

& 'Linear gravity-flow reservoir routing coefficient', &

& 'Linear coefficient in equation to route gravity-reservoir storage for each HRU', &

& 'fraction/day')/=0 ) CALL read\_error(1, 'slowcoef\_lin')

ALLOCATE ( Slowcoef\_sq(Nhru) )

IF ( declparam(MODNAME, 'slowcoef\_sq', 'nhru', 'real', &

& '0.1', '0.0', '1.0', &

& 'Non-linear gravity-flow reservoir routing coefficient', &

& 'Non-linear coefficient in equation to route'// &

& ' gravity-reservoir storage for each HRU', &

IF ( declparam(MODNAME, 'pref\_flow\_infil\_frac', 'nhru', 'real', &

& '-1.0', '-1.0', '1.0', &

& 'Fraction of the soilwater infiltration partitioned to the preferential reservoir storage for each HRU', &

& 'Fraction of the soilwater infiltration partitioned to the preferential reservoir storage for each HRU', &

& 'decimal fraction')/=0 ) CALL read\_error(1,'pref\_flow\_infil\_frac')

ALLOCATE ( Soil2gw\_max(Nhru) )

IF ( declparam(MODNAME, 'soil2gw\_max', 'nhru', 'real', &

& '0.0', '0.0', '5.0', &

& 'Maximum value for capillary reservoir excess to groundwater storage', &

& 'Maximum amount of the capillary reservoir excess that'// &

& ' is routed directly to groundwater storage for each HRU', &

& '0.1', '0.0', '1.5', &

& 'Linear preferential-flow routing coefficient', &

& 'Linear coefficient in equation to route preferential-flow storage for each HRU', &

& 'fraction/day')/=0 ) CALL read\_error(1, 'fastcoef\_lin')

ALLOCATE ( Fastcoef\_sq(Nhru) )

IF ( declparam(MODNAME, 'fastcoef\_sq', 'nhru', 'real', &

& '0.8', '0.0', '1.0', &

& 'Non-linear preferential-flow routing coefficient', &

& 'Non-linear coefficient in equation used to route'// &

& ' preferential-flow storage for each HRU', &

& 'none')/=0 ) CALL read\_error(1, 'fastcoef\_sq')

ALLOCATE ( Ssr2gw\_rate(Nhru) )

IF ( declparam(MODNAME, 'ssr2gw\_rate', 'nssr', 'real', &

& '0.1', '0.0001', '999.0', &

& 'Coefficient to route water from gravity reservoir to groundwater storage', &

& 'Linear coefficient in equation used to route water from'// &

& ' the gravity reservoir to groundwater storage for each HRU', &

& 'inches/day')/=0 ) CALL read\_error(1, 'ssr2gw\_rate')

ALLOCATE ( Ssr2gw\_exp(Nhru) )

IF ( declparam(MODNAME, 'ssr2gw\_exp', 'nssr', 'real', &

& '1.0', '0.0', '3.0', &

& 'Coefficient to route water from subsurface to groundwater storage', &

& 'Non-linear coefficient in equation used to route water'// &

& ' from the gravity reservoir to groundwater storage for each HRU', &