

SUMMARY OF PRMS

Version 5.0.0 – May 30, 2019

This file describes changes to the Precipitation-Runoff Modeling System (PRMS) with each official release. Modules, file names, and user input are identified by using `Courier New` font. Input parameters and dimensions are identified by using **bold** font. State and flux variables are identified by using *italic* font.

Any use of trade, product, or firm names is for descriptive purposes only and does not imply endorsement by the U.S. Government. This software has been approved for release by the U.S. Geological Survey (USGS). Although the software has been subjected to rigorous review, the USGS reserves the right to update the software as needed pursuant to further analysis and review. No warranty, expressed or implied, is made by the USGS or the U.S. Government as to the functionality of the software and related material nor shall the fact of release constitute any such warranty. Furthermore, the software is released on condition that neither the USGS nor the U.S. Government shall be held liable for any damages resulting from its authorized or unauthorized use.

ABSTRACT

The need to assess the effects of variability in climate, biota, geology, and human activities on water availability and flow can be assessed with computer models that simulate the hydrologic cycle at a watershed scale. The Precipitation-Runoff Modeling System (PRMS) is a deterministic, distributed-parameter; physical process-based modeling system developed to evaluate the response of various combinations of climate and land use on streamflow and general watershed hydrology. The primary objectives of PRMS are: (1) simulate hydrologic processes including evaporation, transpiration, runoff, infiltration, and interflow as determined by the energy and water budgets of the plant canopy, snowpack, and soil zone on the basis of distributed climate information (temperature, precipitation, and solar radiation); (2) simulate hydrologic water budgets at the watershed scale for temporal scales ranging from days to centuries; (3) integrate PRMS with other models used for natural-resource management or with models from other scientific disciplines; and (4) provide a modular design that allows for selection of alternative hydrologic-process algorithms from the standard PRMS module library.

HISTORY

PRMS Version 5.0.0 (5/30/2019)
PRMS Version 4.0.3 (6/19/2017)
PRMS Version 4.0.2 (5/01/2016)
PRMS Version 4.0.1 (3/11/2015)
PRMS Version 4.0.0 (2/01/2015)
PRMS Version 3.0.5 (4/24/2012)
PRMS Version 3.0.4 (1/15/2012)
PRMS Version 3.0.3 (11/29/2012)
PRMS Version 3.0.2 (9/22/2012)
PRMS Version 3.0.1 (2/6/2012)
PRMS Version 3.0.0 (11/15/2011)
MMS Versions 1.0.0 through 1.2.1 (1991 through 2002)
PRMS Version 2.1 (1/17/1996)
PRMS Pre-version 2.1 (1983 - 1996)

SYSTEM REQUIREMENTS

PRMS is written in the Fortran 90 and C programming languages. The code has been used on personal computers running various forms of the Microsoft Windows operating system and Linux based computers. A typically small model, i.e., around 100 HRUs can be executed on almost all computers. Large models, i.e., greater than 100,000 HRUs may need at least 8 GB of RAM to run effectively. Executables provided in releases are built to run on 64-bit computers.

DOCUMENTATION AND ADDITIONAL RESOURCES

PDFs of the reports listed below are provided in the 'doc' subdirectory.

- Markstrom, S.L., Regan, R.S., Hay, L.E., Viger, R.J., Webb, R.M.T., Payn, R.A., and LaFontaine, J.H., 2015, PRMS-IV, the precipitation-runoff modeling system, version 4: U.S. Geological Survey Techniques and Methods, book 6, chap. B7, 158 p., <http://dx.doi.org/10.3133/tm6B7>. File tm6-b7_prms4.pdf.
- Regan, R.S., and LaFontaine, J.H., 2017, Documentation of the dynamic parameter, water-use, stream and lake flow routing, and two summary output modules and updates to surface-depression storage simulation and initial conditions specification options with the Precipitation-Runoff Modeling System (PRMS): U.S. Geological Survey Techniques and Methods, book 6, chap. B8, 60 p., <https://doi.org/10.3133/tm6B8>. File tm6-b8_enhancements.pdf.
- Regan, R.S., Markstrom, S.L., Hay, L.E., Viger, R.J., Norton, P.A., Driscoll, J.M., LaFontaine, J.H., 2018, Description of the National Hydrologic Model for use with the Precipitation-Runoff Modeling System (PRMS): U.S. Geological Survey Techniques and Methods, book 6, chap B9, 38 p., <https://doi.org/10.3133/tm6B9>. File tm6-b9_nhm_prms.pdf.

Additional PRMS Documentation:

- Markstrom, S.L., Niswonger, R.G., Regan, R.S., Prudic, D.E., and Barlow, P.M., 2008, GSFLOW—Coupled ground-water and surface-water flow model based on the integration of the precipitation-runoff modeling system (PRMS) and the modular ground-water flow model (MODFLOW–2005): U.S. Geological Survey Techniques and Methods, book 6, chap. D1, 240 p., <http://pubs.usgs.gov/tm/tm6d1/>.
- Regan, R.S., Niswonger, R.G., Markstrom, S.L., and Barlow, P.M., 2015, Documentation of a restart option for the U.S. Geological Survey coupled groundwater and surface-water flow (GSFLOW) model: U.S. Geological Survey Techniques and Methods, book 6, chap. D3, 19 p., <http://dx.doi.org/10.3133/tm6D3/>.
- Leavesley, G.H., Lichty, R.W, Troutman, B.M., and Saindon, L.G., 1983, Precipitation-runoff modeling system--User's manual: U.S. Geological Survey Water-Resources Investigations Report 83-4238, 207 p.
- Leavesley, G.H., Restrepo, P.J., Markstrom, S.L., Dixon, M., and Stannard, L.G., 1996, The Modular Modeling System (MMS): User's manual: U.S. Geological Survey Open-File Report 96-151, 142 p.
- Viger, R.J., Hay, L.E., Jones, J.W., and Buell, G.R., 2010, Effects of including surface depressions in the application of the Precipitation-Runoff Modeling System in the Upper Flint River Basin, Georgia: U.S. Geological Survey Scientific Investigations Report 2010-5062, 36 p.

Online Resources:

- <https://www.usgs.gov/software/precipitation-runoff-modeling-system-prms-0>

PRMS Version 5.0.0 – May 30, 2019

This major release adds new functionality for:

- input of dynamic parameters and water use data;
- Muskingum streamflow routing with simulation of lakes;
- output of subbasin, stream segment, and basin variables to Comma-Separated-Variable (CSV) files by activating modules `nsub_summary`, `nsegment_summary`, and/or `basin_summary`;
- an option to compute potential evapotranspiration by using the Penman-Monteith formulation on the basis of specified windspeed and humidity data in the Data File; and
- distribution of air temperature on the basis of specified minimum and maximum data in the Data File.

Also, corrected major bugs in the `potet_pm` and `potet_pt` modules and minor bugs in the `soilzone` and `gwflow` modules. See “Update of PRMS PET modules.pdf” file in the ‘doc’ subdirectory for corrections to PRMS-IV documentation report regarding the potential evapotranspiration modules.

Restart Files generated by previous versions are not compatible with the current version, thus they must be regenerated. The simulation time period for the antecedent simulation is written into the file and printed after it is read.

Several PRMS parameters are deprecated and replaced by new parameters that were added to facilitate automated model calibration. Most of the deprecated parameters were codependent with other parameters and were specified as a depth per unit area. The new parameters are specified as a fraction of, or offset to the related deprecated parameter. To use the new parameters, specify **model_mode** as `PRMS5`. To retain downward compatibility, to use the deprecated parameters, specify **model_mode** as `PRMS`. Parameter Files for example problems were converted to use the new `PRMS5` parameters instead of the equivalent `PRMS4` parameters. A new module was added to facilitate adding the new parameters to an existing model. To activate this module, specify control parameter **model_mode** as `CONVERT` to produce a file of the new parameters. Similarly, specifying **model_mode** as `CONVERT4` will produce a file of the old parameters based on a model using the new parameters. Once the new parameters are added to a Parameter File it is recommended the old parameters be removed.

See the file ‘`tm6b8_PRMS_enhancements.pdf`’ (Regan and LaFontaine, 2017) in the ‘doc’ subdirectory for documentation on the dynamic parameters; water-use, lake simulation, and HRU summary options. The file ‘`PRMS_tables_5.0.pdf`’ in the ‘doc’ subdirectory provides updated tables for the PRMS-IV documentation report (Markstrom and others, 2015; <http://pubs.usgs.gov/tm/6b7/>) that describe PRMS modules, dimensions for parameters and variables, parameters in the Control File, parameters in the Parameter File, and input and output variables. Changes from PRMS version 4.0.3 are highlighted in yellow and green in that document.

FUNCTIONALITY [yellow highlight designates new modules for this version]

The current version of PRMS includes the following modules and Utility Routines (listed in computation order):

basin	Basin Module
climateflow	Climate and flow parameters and variables Module
cascade	Cascading-Flow Module
obs	Observed-Data Module
dynamic_param_read	Dynamic Parameter Input Module
water_use_read	Water Use Input Module
prms_time	Time Variable Computation Utility Routine
soltab	Potential Solar-Radiation Module
temp_1sta	One-Station Air-Temperature-Distribution Module
temp_laps	Lapse-Station Air-Temperature-Distribution Module
temp_dist2	Inverse-Distance Air-Temperature-Distribution Module
temp_sta	Modified One-Station Air-Temperature-Distribution Module
precip_1sta	One-Station Precipitation-Distribution Module
precip_laps	Lapse-Station Precipitation-Distribution Module
precip_dist2	Inverse-Distance Precipitation-Distribution Module
xyz_dist	Multiple Linear Regression Precipitation and Temperature-Distribution Module
ide_dist	Inverse distance and elevation Precipitation and Temperature-Distribution Module
climate_hru	Pre-computed and Distributed Climate Module
ddsolrad	Degree-Day Solar-Radiation Distribution Module
ccsolrad	Cloud-Cover Solar-Radiation Distribution Module
potet_jh	Jensen-Haise Potential-Evapotranspiration Module
potet_hamon	Hamon Potential-Evapotranspiration Module
potet_pan	Pan-Evaporation Potential-Evapotranspiration Module
potet_hs	Hargreaves-Samani Potential-Evapotranspiration Module
potet_pt	Priestley–Taylor Potential-Evapotranspiration Module
potet_pm	Penman–Monteith Potential-Evapotranspiration Module; using specified windspeed and humidity in CBH Files
potet_pm_sta	Penman–Monteith Potential-Evapotranspiration Module using specified windspeed and humidity in the Data File
transp_frost	Frost-Based Active Transpiration Period Module
frost_date	Frost Date Preprocess Utility Routine
transp_tindex	Temperature Index Based Active Transpiration Period Module
intcp	Precipitation-Interception Module
snowcomp	Snow Module
srunoff_smidx	Non-linear Source Area Surface-Runoff and Infiltration Module
srunoff_carea	Linear Source Area Surface-Runoff and Infiltration Module
soilzone	Soil-Zone Module
gwflow	Groundwater Reservoir Module

subbasin	Subbasin Module
routing	Stream Network Computations Routing Utility Routine
strmflow	Streamflow Module
muskingum	Muskingum Streamflow Routing Module
strmflow_in_out	Streamflow Routing Module with inflow equals outflow for each segment
muskingum_lake	Muskingum Streamflow and Lake Routing Module
water_balance	Water Balance Debug Utility Routine
nhru_summary	Write User-Selected HRU-based Variables to CSV File Module
nsegment_summary	Write User-Selected Stream Segment Variables to CSV File Module
nsub_summary	Write User-Selected Subbasin Variables to CSV File Module
basin_summary	Write User-Selected Basin Variables to CSV File Module
prms_summary	Write Watershed Summary Basin-averaged Variables to CSV File Module
basin_sum	Watershed Flow-Summary Module
map_results	Map Based Output Module
write_climate_hru	Generate Climate-by-HRU Files Preprocess Module
convert_params	Generate PRMS-IV or PRMS-V Parameters Preprocess Module

SCREEN OUTPUT

Screen and Model Output File output updated for readability and output of additional information. There can be a large amount of information printed to the screen including: general information, warning messages, and error messages during initialization of a simulation. Sometimes this can make it difficult to see important error messages. There are several ways to reduce screen output. Set control parameter **parameter_check_flag** to 1 in the PRMS Control File to minimize warning messages about parameter values falling within the suggested range. Set control parameter **print_debug** to -2 (minimum output) or -1 (less output). However, setting **parameter_check_flag** = 1 and **print_debug** = 0 is good practice during initial model development as all warning and error messages are available. But, once warnings are deemed ok, **parameter_check_flag** should be set to 0 and **print_debug** to -1 or -2.

INITIAL CONDITIONS FILES

- Initial Condition Files made with previous PRMS versions are not compatible with this version; thus, they must be regenerated. The simulation time period for the antecedent simulation is written into the file and printed when used. The new files are about 75 percent smaller as only variables needed to maintain a water balance are stored. Modules with fewer values saved are: snowcomp, soilzone, srunoff_smidx, srunoff_carea, routing, gwflow, intcp, and muskingum.
- Restriction for changing to or from modules potet_pt, potet_pm, potet_pm_sta, ide_dist, and xyz_dist for restart simulations has been removed.
- Restriction for changing the content of Data File(s) for restart simulations has been removed.

- Options that cannot change for a restart simulation include: a) surface depression storage simulation option; b) cascading flow simulation option; c) dimensions **nhru**, **nssr**, **ngw**, **nsegment**, **nhru**, **nlake**; and d) the model mode (PRMS or PRMS5); and e) use of modules `temp_1sta`, `temp_laps`, `temp_dist2`, `potet_pan`, `transp_tindex`.
- Various states can be updated for a restart simulation by using new options as specified by control parameter `init_vars_from_file` (0 = read all initial value parameters; 1 = do not read initial value parameters; 2 = read **dprst_frac_init**, **snowpack_init**, **segment_flow_init**, **elevlake_init**, **gwstor_init**, (**soil_rechr_init**, **soil_moist_init**, **ssstor_init** for **model_mode** = PRMS) or (**soil_rechr_init_frac**, **soil_moist_init_frac**, **ssstor_init_frac** for **model_mode** = PRMS5); 3 = read **snowpack_init**; 4 = read **elevlake_init**; 5 = read (**soil_rechr_init_frac**, **soil_moist_init_frac**, **ssstor_init_frac** for **model_mode** = PRMS5); 6 = read **gwstor_init**; 7 = read **dprst_frac_init**). This option could be used, for example, to update the snowpack based on observed values of snowpack water equivalent by restarting a simulation and specifying values for **snowpack_init** to reflect an observed or model snow water equivalent data set.

BUG FIXES – by module:

snowcomp

- When a snowpack is melting and there is a new snowfall, the code saves the point on the depletion curve (variable `snowcov_areasv`) and then depletes the snowpack from **snarea_curve**(11,k) over the subsequent time steps until the snow cover is less than or equal to `snowcov_areasv` on the basis of linear interpolation. See Figure 1-4, page 97 of the PRMS-IV documentation report (TM6-B7, Markstrom and others, 2015) for a depiction of how to compute snow-water equivalent (SWE) using a depletion curve after a secondary snowfall. However, `snowcov_areasv` was set to **snarea_curve**(11,k), the maximum value of a snow depletion curve, instead of current value of `snowcov_area`. This issue was fixed. This change can affect results of the snowpack depletion for secondary snowfalls as depletion begins at the maximum value of the depletion curve instead of returning to the point of departure from the depletion curve from which it was melting prior to the snowfall. Thus, there will be no change for many simulations, but, noticeable changes could occur for some simulations.
- Variable `frac_swe` was not computed for all time steps, this was corrected and the variables were changed to single precision; this change did not affect other computations.
- Variable `tcal` was not reset to 0.0 for all time steps, which might have meant values were carried over after a snowpack melted, this was corrected; this change did not affect other computations.
- Initial snow cover area was computed incorrectly, this would only affect the first timestep.

temp_dist2

- Values for parameter **tmax_adj** and **tmin_adj** were not read from the Parameter File and were likely set to zero or very small values.

climate_hru

- Corrected check that looked for first simulation time step in CBH Files that failed if simulation start month and year were earlier than the first date in the CBH File.

soilzone

- Parameters **soil_moist_max**, **soil_rechr_max**, and **soil_rechr_max_frac** could have values specified equal to 0, which would cause divide by zero in several equations. To prevent divide by zeros and to allow for some minimum storage in capillary reservoirs to receive cascading flow, the code was modified to issue an error if values for these parameters are specified less than 0.00001. If control parameter **parameter_check_flag** is specified equal to 0, a warning message is issued and the value is set to 0.00001.
- Parameters **slowcoef_lin**, **slowcoef_sq**, **fastcoef_lin**, and **fastcoef_sq** can have values specified equal to 0. However, under rare conditions, doing so could cause a divide by zero. The code was modified to trap for possible divide by zero and ensure a valid value is computed for interflow. Also, for some combinations of these parameters and a large gravity storage value it was possible to compute very small negative values of interflow due to floating point precision round off error. The code was modified to set any computed negative values of interflow to 0.

prms_summary

- If values for optional parameter **poi_gage_segment** were greater than 999999 values in the header row were truncated at 7 digits. This has been corrected. Additionally, the format of the output values was changed from a fixed 10-digit exponential to values with 4 decimal places with as many values to the left of the decimal places as required.

routing

- Values for solar radiation and potential evapotranspiration could be computed incorrectly for segments that did not have associated HRUs, this has been fixed to assign values from closest upstream segment.

CHANGES THAT CAN AFFECT SIMULATION RESULTS

snowcomp

- For calculations of emissivity and energy related to convection or condensation, the code checks if there is precipitation on an HRU instead of anywhere in the model domain to adjust these values based on precipitation and cloud cover.
- Snow depletion curves must be specified, previous versions allowed snow depletion curves to not be specified when dimension **ndepl** was specified equal to 0.
- Several checks for small negative round-off errors were removed to reduce bias in round-off error. Thus, it is possible for some variables to have very small negative values. This should be a rare occurrence.

soilzone

- Values of **ssr2gw_rate** specified less than 0.000001 were treated as 0. Now, all **ssr2gw_rate** values > 0.0 are used to compute gravity drainage. In general, specifying parameter values < 0.000001 are going to produce results within or below the limits of floating-point precision, so they will be adding more noise and increased execution time than useful results.

intcp

- Previously, if there was canopy storage at the changeover of a transpiration period (winter to summer or summer to winter) and the canopy could not hold the antecedent water content,

this water was added to *net_rain*. This water is now kept track of in variables *intcp_changeover* and *basin_changeover* and added in the same manner as *net_rain* in the `srunoff` module. Overall results don't change, except that *net_rain* is consistent with the total amount of precipitation.

NEW FUNCTIONALITY

`dynamic_param_read`

- Read and makes available dynamic parameters by HRU from pre-processed files.
- Dynamic parameters include those that specify impervious surface fraction and storage capacity; storage capacity of the capillary and recharge reservoirs of the soil zone; total surface-depression storage and open surface-depression fractions, depth, pervious and impervious surface-runoff capture fraction, and storage threshold for open depressions to spill; canopy type, density, and storage capacity; plant transpiration period; and solar radiation transmission and potential evapotranspiration (ET) computation coefficients.

`water_use_read`

- Read and makes available water-use data (diversions and gains) from pre-processed files.
- Water can be withdrawn from five sources: (1) stream segment flow, (2) groundwater reservoir storage, (3) open surface-depression storage, (4) external locations, and (5) lake storage. Source water can be transferred to any of eight destinations: (1) stream segments, (2) groundwater reservoir storage, (3) open surface-depression storage, (4) external locations, (5) lake storage, (6) capillary reservoir storage, (7) internal consumptive-use locations, and (8) plant canopy storage. Water transfers can be any source/destination combination. Multiple transfers can originate from each source, and each destination can receive water from multiple sources.
- Modules `intcp`, `soilzone`, `srunoff_smidx`, `srunoff_carea`, `gwflow`, `strmflow_in_out`, `muskingum`, and `muskingum_lake` were modified to account for water-use transfers.

`potet_pm_sta`

- Computes the potential evapotranspiration by using the Penman-Monteith formulation (Penman, 1948; Monteith, 1965) using specified windspeed and humidity in the Data File.

`muskingum_lake`

- Routes water between segments in the system using Muskingum routing and on-channel water body storage and flow routing.

`temp_sta`

- Similar temperature distribution to `temp_1sta` except that no adjustment is made using lapse rates. The value of the associated temperature from a station by parameter `hru_tsta` is adjusted using parameters `tmax_adj` and `tmin_adj`.

`nhru_summary`

- Added output yearly total and mean yearly time series on the basis of control parameter `nhruout_freq` (1 = daily, 2 = monthly, 3 = both, 4 = mean monthly, 5 = mean yearly, 6 = yearly total); daily files have the suffix `.csv`; monthly files have the suffix `_monthly.csv`; mean monthly have the suffix `_meanmonthly.csv`; and mean yearly and yearly total have the suffix `_yearly.csv`.

- Added the option to output of NHM HRU identification number (parameter **nhm_id**) to CSV files when control parameter **nhruOutON_OFF** is specified equal to 2.
- Added the capability to output integer variables.
- Added the option to output using different formats using new control parameter **nhruOut_format** (1 = scientific notation with 4 significant digits (default); 2 = 2 decimal places; 3 = 3 decimal places; 4 = 4 decimal places; 5 = 5 decimal places).
- Parameter **prms_warmup** is now specified in the Control File.

`nsub_summary`

- Summary output module that operates similar to `nhru_summary` and with similar control parameters, except that it is used to write values of variables dimensioned with **nsub** and variables dimensioned with the value of **nhru** when parameter **hru_subbasin** is specified to separate CSV files at daily, monthly, mean monthly, mean yearly, and yearly total time steps when control parameter **nsubOutON_OFF** is specified equal to 1.
- Added the capability to output integer variables in addition to real and double variables.
- Added the new control parameter **nsubOut_format** for outputting different number formats (1 = scientific notation with 4 significant digits (default); 2 = 2 decimal places; 3 = 3 decimal places; 4 = 4 decimal places; 5 = 5 decimal places).

`nsegment_summary`

- New summary output module that operates similarly to `nhru_summary` and with similar control parameters except that it is used to write values of variables dimensioned with **nsegment** to separate CSV files at daily, monthly, mean monthly, mean yearly, and yearly total time steps when control parameter **nsegmentOutON_OFF** is specified equal to 1.
- Added the new control parameter **nsegmentOut_format** for outputting different number formats (1 = scientific notation with 4 significant digits (default); 2 = 2 decimal places; 3 = 3 decimal places; 4 = 4 decimal places; 5 = 5 decimal places).

`basin_summary`

- New summary output module that operates like `nhru_summary`, with similar control parameters, except that it is used to write values of variables dimensioned with **one** to separate CSV Files at daily, monthly, mean monthly, mean yearly, and yearly total time steps when control parameter **basinOutON_OFF** is specified equal to 1.

`map_results`

- Parameter **prms_warmup** is now specified in the Control File.

`potet_pt, potet_pm, and potet_pm_sta`

- Allows humidity data to be specified using new parameter **humidity_percent** that has maximum dimensions `nhru` by **nmonths** as an alternative to specifying humidity in a CBH file. Note, previous versions of the `potet_pt` module did not require humidity data.

`cascade`

- Parameter **hru_segment** can now be used to specify simple one-to-one HRU to stream segment cascade paths instead of specifying all cascade parameters with **cascade_flag** set equal to 2. This option assumes **ncascade = ncascdgw = nhru** and indicates that parameters **hru_up_id**,

hru_strmseg_down_id, hru_down_id, hru_pct_up, gw_up_id, gw_strmseg_down_id, gw_down_id, gw_pct_up, cascade_tol, cascade_flg, and circle_switch are not specified.

`convert_params`

- Produces an output file of replacement parameters added to facilitate calibration efforts. To activate this module, specify control parameter **model_mode** as **CONVERT** to produce a file of the new parameters (**tmax_allrain_offset, soil_rechr_max_frac, soil_rechr_init_frac, soil_moist_init_frac, and soil_moist_init_frac**). Similarly, specifying **model_mode** as **CONVERT4** will produce a file of the old parameters (**tmax_allrain, soil_rechr_max, soil_rechr_init, soil_moist_init, and soil_moist_init**) based on a model using the new parameters.

Temperature modules

- To make a consistent check for valid temperature ranges the constants **MAXTEMP** (value = 200.0) and **MINTEMP** (value = -150) were added to `basin.f90`. Distributed values are compared to these values to determine if they are “valid”. Values outside this range are treated as missing values. Previously, some modules used the range 150.0 to -99.0.

MODULE REMOVED

`strmflow_lake`

- This module (files `strmflow_lake.f90` and `lake_route.f90`) was replaced by the `muskingum_lake` module with all functionality retained in the new module, which includes routing of streamflow using the Muskingum method.

PARAMETER CHANGES

Notes: there are several ways to specify parameters. Traditionally, a single value is specified per line. However, multiple values can be specified per line if fewer than 12,000 characters are specified and there are no trailing blanks. This might be useful to specify a parameter as a grid of values as **ncol, nrow** with the upper left value specified for column 1 row 1 and the bottom right value specified for column **ncol**, row **nrow**. See the section of flexible dimension option described below under release version 4.0.1 for a simplified method of specifying parameters with constant values for any dimension of a parameter.

`print_debug`

- If specified equal to -2 screen output is very limited and the model output file is not generated and the `basin_sum` module is not used. Thus, all computed variables unique to `basin_sum` are not available.

`prms_warmup`

- This parameter is now a Control Parameter to allow it to be changed more easily. If specified in the Parameter File it is ignored. This parameter is used in `nhru_summary, map_results, nsub_summary, nsegment_summary, and basin_summary`.

`gw_seep_coef`

- The dimension is now **ngw** instead of **nlake**

`humidity_percent`

- This new parameter can be used as an alternative to specification of humidity in a CBH file for modules `potet_pt`, `potet_pm`, and `potet_pm_sta`. It has maximum dimensions **nhru** by **nmonths**.

obsout_segment

- This new parameter is used to specify a replacement flow for outflow of a segment. This is like **obsin_segment** that can be used to specify a replacement flow for the inflow to a segment. Both parameters can be used with modules `strmflow_in_out`, `muskingum`, and `muskingum_lake`, which are available in PRMS or PRMS5 simulation modes. Replacement flows are specified in the Data File and used to assign measured flows to segments instead of the simulated flow. Use of replacement flows breaks conservation of mass, but, might be useful to set flows below a managed water body.

segment_type

- Specifies type of segment (0 = normal; 1 = headwater; 2 = lake; 3 = replacement flow; 4 = inbound to nation; 5 = outbound from nation; 6 = inbound to region; 7 = outbound from region; 8 = drains to ocean; 9 = sink (terminus to soil); 10 = inbound from Great Lakes; 11 = outbound to Great Lakes; 12 = ephemeral; + 100 user updated; 1000 user virtual segment 100 = user normal; 101 - 108 = not used; 109 sink. This parameter can be used with modules `strmflow_in_out`, `muskingum`, and `muskingum_lake`, with the sole purpose of accumulating total flows for each segment type.

Parameters used when **model_mode** = PRMS5

tmax_allrain_offset

- Equals old parameter **tmax_allrain** – **tmax_allsnow**

soil_rechr_max_frac

- Equals old parameter **soil_rechr_max** / **soil_moist_max**

soil_rechr_init_frac

- Equals old parameter **soil_rechr_init** / **soil_rechr_max**

soil_moist_init_frac

- Equals old parameter **soil_moist_init** / **soil_moist_max**

ssstor_init_frac

- Equals old parameter **ssstor_init** / **sat_threshold**

sro_to_dprst_perv

- Equals old parameter **sro_to_dprst**

dprst_frac

- Equals old parameter **dprst_frac_hru**

Parameters used when **model_mode** = PRMS

tmax_allrain

- Replaced by **tmax_allrain_offset**

soil_rechr_max

- Replaced by `soil_rechr_max_frac`
- `soil_rechr_init`
- Replaced by `soil_rechr_init_frac`
- `soil_moist_init`
- Replaced by `soil_moist_init_frac`
- `ssstor_init_init`
- Replaced by `soil_moist_init_frac`
- `sro_to_dprst`
- Name change to `sro_to_dprst_perv`
- `dprst_area` or `dprst_frac_hru`
- Replaced by `dprst_frac`

NEW VARIABLES

`strmflow_in_out`, `muskingum`, `muskingum_lake`

- Added support for additional segment types using the parameter **segment_type** that provides output by category. These are *flow_to_lakes*, *flow_to_ocean*, *flow_to_great_lakes*, *flow_out_region*, *flow_out_NHM*, *flow_in_region*, *flow_terminus*, *flow_in_nation*, *flow_headwater*, *flow_in_great_lakes*, *flow_replacement*.

`intcp`

- *intcp_changeover* and *basin_changeover*

Removed variables `cascade_interflow`, `cascade_dunnianflow`, `interflow_max`, `cpr_stor_frac`, `pfr_stor_frac`, `gvr_stor_frac`, `soil_moist_frac`, `soil_rechr_ratio`, `snowevap_aet_frac`, `perv_avail_et`, `cap_upflow_max`

REMOVED VARIABLES

- *cascade_interflow*, *cascade_dunnianflow*, *interflow_max*, *cpr_stor_frac*, *pfr_stor_frac*, *gvr_stor_frac*, *soil_moist_frac*, *soil_rechr_ratio*, *snowevap_aet_frac*, *perv_avail_et*, and *cap_upflow_max*

SMALL CHANGES THAT DO NOT AFFECT ANY COMPUTATIONS

Determination of Muskingum routing variables moved from source file `muskingum.f90` to `routing.f90` so that the `muskingum` and `muskingum_lake` modules could share the same code.

PRMS UTILITIES

- If an error is encountered reading a Parameter File an error message is issued and the file is no longer read instead of trying to continue reading the file.

- The delimiter between specification of multiple values on a line in a Parameter File was changed from a comma to a space. Trailing blanks cannot be specified on a line as this produces a read error.
- The end of a line for parameter values is determined by either a new line or null character instead of just a new line.
- Possible security issues addressed related mostly to buffer overflow and underflow.

PRMS Version 4.0.3 – May 05, 2017

This release corrected major bugs in the `potet_pm` and `potet_pt` modules and minor bugs in the `soilzone` and `gwflow` modules. Screen and Model Output File output updated for readability and output of additional information. See the updated tables for updates to parameters, variables, modules, and dimensions at:

ftp://brrftp.cr.usgs.gov/pub/mows/software/prms/4.0.3/PRMS_tableUpdates_4.0.3.pdf

BUG FIXES – by module:

`potet_pt`

- The computation of the net long wave radiation was corrected.

`potet_pm`

- The computation of the net long wave and net radiation coefficients used to compute potential ET were corrected.

`srunoff`

- The computation of pervious and impervious capture of pervious and impervious surface runoff into depressions did not account for the fraction of pervious and impervious to compute actual inflow, which lead to the possibility of excess water to the stream network.

`muskingum`

- If `K_coef` was set < 1.0, it was possible to get a divide by 0 under rare conditions, this was corrected.

`soilzone`

- Changed declared dimensions from `nhru` to `one` for variables `basin_cpr_stor_frac`, `basin_gvr_stor_frac`, `basin_pfr_stor_frac`, `basin_soil_lower_stor_frac`, `basin_soil_rechr_stor_frac`, and `basin_sz_stor_frac`. This does not affect any computations, but, could cause unpredictable results for output of these variables in `statvar` and animation files and runtime graphs.
- Computation of `basin_pfr_stor_frac` was computed incorrectly using `pref_flow_stor` instead of `pfr_stor_frac`. This does not affect any other computations.

`gwflow`

- The determination on whether to add storage on the basis of the value of `gwstor_minarea` is done after adding recharge instead of before.

`basin`

- If an HRU had open and closed surface-depression storage portions, computations were incorrect, this was corrected.

SMALL CHANGES THAT DO NOT AFFECT ANY COMPUTATIONS

call_modules

- Updated output to screen and to the model output file for improved readability and added output of the Parameter File name.

cascade

- Removed checks for specification of farfield flow as farfield computations are not supported.

subbasin

- Changed units in variable descriptions for *subinc_wb* and *subinc_deltastor* from cfs to inches.

map_results

- Messages for incomplete or excess accounting for mapping between HRU map and target map includes more information.

basin

- Summary statistics for areal portions and date and times reformatted.

NEW FUNCTIONALITY

nhru_summary

- Added output yearly total and mean yearly time series on the basis of control parameter **nhruout_freq** (1 = daily, 2 = monthly, 3 = both, 4 = mean monthly, 5 = mean yearly, 6 = yearly total); daily files have the suffix .csv; monthly files have the suffix _monthly.csv; mean monthly have the suffix _meanmonthly.csv; and mean yearly and yearly total have the suffix _yearly.csv.

NEW VARIABLES

- Added variable *basin_swrad*, which is set equal to *basin_potsw*. This change impacts modules *ccsolrad*, *ddsolrad*, *climate_hru*, and *climateflow*.
- Added *subbasin* variables: *subinc_rain*, *subinc_snow*, *subinc_stor*, *subinc_recharge*, *subinc_szstor_frac*, and *subinc_capstor_frac*.
- Added *basin_sum* variables: *basin_swrad_yr*, *basin_swrad_tot*, and *basin_swrad_mo*, which are added to summaries in the **model_output_file** on the basis of selected print options.

PRMS Version 4.0.2 – July 29, 2016

This release added new parameters and variables, fixed bugs, added more checks for valid input values, general code clean up, mostly to reduce mixed floating-point computations by changing some variables to double precision, some to single precision, and using FORTRAN intrinsic functions to convert variables prior to mixed-precision computations. Additional information about input and output files and simulation period are printed to the screen. See the updated tables for updates to parameters, variables, modules, and dimensions at:

ftp://brrftp.cr.usgs.gov/pub/mows/software/prms/4.0.2/PRMS_tableUpdates_4.0.2.pdf

BUG FIXES – by module:

potet_pt

- The code incorrectly used the value of the average HRU air temperature in degrees Fahrenheit when it was supposed to be in degrees Celsius in one equation. The equation is correct in the PRMS-IV documentation report.

potet_pm

- The equation is different from the PRMS-IV documentation report.

potet_hs

- The kt coefficient was first being computed each day using an equation based on average temperature as described in http://www.zohrabsamani.com/research_material/files/Hargreaves-samani.pdf. Then this value was multiplied by the hs_krs parameter. Now, only the hs_krs parameter is used in the Hargreaves-Samani equation, which matches the PRMS-IV documentation. The kt coefficient equation can be used to estimate the hs_krs parameter: $\text{coef_kt} = 0.00185 * (\text{temp_dif} ** 2) - 0.0433 * \text{temp_dif} + 0.4023$, where temp_dif is the difference between the HRU maximum and minimum temperature in units of degrees Celsius.

precip_laps

- Arrays for parameters **rain_adj_lapse** and **snow_adj_lapse** were not allocated causing application to fail.

routing

- Variables *seginc_potet*, *seginc_gwflow*, *seginc_ssflow*, *seginc_sroff*, *seginc_swrad*, *seg_gwflow*, *seg_ssflow*, and *seg_sroff* were computed if the cascading flow option was active, which was an error if parameter **hru_segment** was not specified. These variables are not computed if the cascading flow option is active.
- The value of *seg_lateral_inflow* was not set correctly when the cascading flow option was active; it is now set to *strm_seg_in*.

subbasin

- Subbasin storage did not include surface-depression or lake storage.

climate_hru

- Variable *basin_obs_ppt* was set to 0.0 if precipitation values are not input in a CBH file; thus, an error occurred if parameter **precip_module** was not specified equal to `climate_hru`.

map_results

- Yearly output was not computed when parameter **mapvars_freq** = 5.
- Corrected IF blocks that determine whether or not parameters **gvr_cell_pct** and **gvr_hru_id** were needed; **gvr_cell_pct** needed when **nhru** not equal **ngwcell** and **gvr_hru_id** needed when **nhru** not equal **nhru**.

gwflow

Variable *basin_lake_seepage* was computed incorrectly by using incorrect values of **hru_area**.

basin_sum

- The value of the yearly observed streamflow or monthly value of computed basin streamflow and monthly observed streamflow was not printed for **print_type** = 2.
- Detailed output did not have the right line length, thus missing a value.
- Yearly detailed output now includes evaporation for lakes and interception storage.

ide_dist

- Computations of inverse distance and elevation allowed to be less than 0 for precipitation computations so that they are consistent with computations for temperature computations.

srunoff_smidx and srunoff_carea

- Variable *contrib_fraction* was declared as dimension **one** and data type double when it should be dimension **nhru** and data type single. This error could have caused memory problems if the

variable was output in a Statistics Variable (statvar) or Animation File, or using the PRMS GUI runtime plots, but would not affect any computations.

soilzone

- Computation of variable *soil_lower_ratio* was set incorrectly to *soil_moist/soil_moist_max* instead of *soil_lower/soil_lower_stor_max* and values of *soil_lower_ratio* for lake and inactive HRUs were not initialized to 0. No other computations are affected.
- In the compute_soilmoist routine, the check for infiltration water in excess of **soil_moist_max** minus *soil_to_gw* (local variable *excs*) was $> Infil$ should have been a check for $excs > Infil * perv_frac$ as *excs* is computed for the whole HRU area and *infil* for only the pervious area on an HRU; this condition was not likely to have occurred.
- Make sure any flow from gravity to preferential-flow reservoirs is set to 0 or current value (previous versions could have used value from previous HRU if current HRU in loop does not have preferential flow reservoir).

basin

- Corrected code to ensure a divide by zero is not permitted for the condition that there are closed depressions but not open depressions; that is, if any values for parameter **dprst_frac_open** are specified = 0.
- If the first value of parameter **dprst_area** was specified < 0 or not included in the Parameter File, the values of **dprst_frac_hru** (default value = 0) were ignored if specified. Thus, if **dprst_frac_hru** was not included in the Parameter File no surface-depression storage would be simulated; this check should have been reversed. Use of parameter **dprst_area** is now deprecated, so users should now use **dprst_frac_hru** and specify values ≥ 0 . The default value of **dprst_frac_hru** is now -1.0.

snowcomp

- Added check to be sure *snowcov_area* is not equal to zero when *pkwater_equiv* is greater than 0. This would be very rare and only possible when the second value of a snow-depletion curve was specified equal to zero.
- Checks were added to be sure *pkwater_equiv* is set to 0.0 if it is computed as a very small negative value, which would be very rare.
- Added check for when $pkwater_equiv < 1.0E-10$ and $snowcov_area > 0$; if true, set *snowcov_area* to 0. This is possible when the energy is enough to melt the snowpack on the previous time step. This bug affected computations for impervious and surface-depression storage evaporation for days without snow and snow depletion curves that specified snow-covered area exists when the snow-water equivalent is equal 0.
- In calin function, *snowcov_area* was set to 0 when it should not be because it is the snow cover after adding precipitation and before any melt or sublimation; this condition affects computations for impervious and surface-depression storage evaporation for days with snow.
- If snowpack exists and the snowpack density (*pk_den*) ≤ 0 , be sure *pk_den* and the snowpack depth (*pk_depth*) have a value based on parameter **den_max** in run function and calin function; this condition would be rare.
- In routine calin, added check for the value of $freeh2o - pwcap > pkwater_equiv$; if so, use the value of *pkwater_equiv*. This condition would be rare.
- In routine calin, the local variable *apk_ice* did not have a value if $snowcov_area \leq 0$; if true, it is set to 0. This condition would be rare.

water_balance module:

- Full array for *hru_sroffp* and *hru_sroffi* instead of HRU value printed for water-budget issue for **srnoff** with cascades active.

- Variable *soil_to_ssr* was not included in HRU **soilzone** water-budget computation. This variable is needed because the variable *cap_waterin*, actual water into capillary reservoir, replaced *cap_water_maxin*, the maximum potential water into the capillary reservoir.

UPDATES THAT MIGHT PRODUCE SLIGHT CHANGES IN ASSOCIATED COMPUTATIONS –

general:

- Some single-precision variables were changed to double-precision variables and vice versa; also, most modules now use FORTRAN intrinsic functions to explicitly designate mixed-precision computations. These changes are intended to limit the possibilities of different results on different computers and compilers, to provide more consistent floating-point comparisons, and to have more consistent round-off issues. These updates could change memory requirements and execution time very slightly. Modules affected: *obs*, *cascade*, *ccsolrad*, *ddsolrad*, *climate_hru*, *temp_dist2*, *precip_dist2*, *ide_dist*, *xyz_dist*, *potet_hamon*, *intcp*, *snowcomp*, *srunoff_smidx*, *srunoff_carea*, *soilzone*, *gwflow*, *routing*, *water_balance*, *nhru_summary*, *map_results*, *prms_summary*, *subbasin*, *climateflow*, and *basin*.
- Small values (>0.0 and $< 1.0E-05$) of precipitation are used in computations; previously these values were assumed to be below round-off tolerance and set to 0. This affects modules *obs*, *precip_1sta*, *climate_hru*, *ide_dist*, *xyz_dist2*, and *precip_laps*.
- Small values (>0.0 and $< 1.0E-06$) of computed potential evapotranspiration were considered to be round-off error and set to 0; now those values are used in computations. This change affects modules *potet_hamon*, *potet_hs*, *potet_jh*, *potet_pan*, *potet_pm*, and *potet_pt*.
- *climateflow* module: Small values (>0.0 and $< 1.0E-06$) of computed mixed precipitation were considered to be round-off error and the event was set to all snow or all rain depending on the precipitation form, now those values are used in computations.
- Computation of saturation vapor pressure for module *potet_pm* now uses an equation by Irmak and others (2012; Journal of Hydrology, v. 420-421, p. 228), to be consistent with module *potet_pt*. This can reduce execution time.
- Small values (>0.0 and $< 1.0E-04$) specified for **hru_percent_imperv** and **dprst_area** are used in computations; previously these values were assumed to be below round-off tolerance and set to zero.
- Values of canopy interception computed between 0.0 and $1.0E-05$ were considered to be round-off error and set to 0.0; these values are now left in the canopy. Module affected: *intcp*.
- Double precision values $< 1.0D-15$ are treated as zero instead of $< 1.0D-10$.
- Module *cascade*: changed check for excess GWR cascade fraction so that any values > 1 set to 1 instead of only > 1.00001 . This change makes the check consistent with the HRU cascade fraction check.
- Module *snowcomp*:
 - instead of using $< -1.0E-10$ to check for round-off issues in some computations, < 0.0 is now used. Instead of using $> 1.0E-06$ to check for round-off issues in some places, $> 1.0E-09$ is now used.

- values of snowpack water equivalent (*pkwater_equiv*) computed between 0.0 and 1.0E-09 were considered to be round-off error and set to 0.0; these values are now used.
- Values < 0.0 are set to 0.0 with a warning message printed when control parameter **print_debug** > -1. This condition accounts for negative round-off error due to mixed precision computations and may occur under rare conditions.

NEW FUNCTIONALITY

The values of all parameters specified in the Parameter File are now checked to determine if they fall within the suggested minimum and maximum range. If values are specified outside the range, a warning message is issued. If the user wants to specify values out of the range, set control parameter **parameter_check_flag** = 0 to deactivate these checks. Previously, select parameters were checked within individual modules.

nhru_summary

- Added functionality to output monthly and mean monthly time series based on the value of new control parameter **nhruout_freq** (1 = daily, 2 = monthly, 3 = both, 4 = mean monthly), daily files have the suffix .csv; monthly files have the suffix _monthly.csv, and mean monthly have the suffix _meanmonthly.csv.

gwflow and cascade

- Added option to allow GWRs to be swales with new control parameter **gwr_swale_flag** (0 = not allowed; 1 = groundwater flow routed to groundwater sink; 2 = groundwater flow routed to stream network).

cascade

- Allow groundwater-reservoir cascades to be equal to HRU cascades with **cascadegw_flag** = 2.

call_modules

- Specification of control parameter **print_debug** = -1 reduces amount of screen output.

ccsolrad

- Removed restart subroutine so **ccsolrad** can be used in a restart simulation if **ddsolrad** was used for the antecedent simulation. The switch from **ddsolrad** to **ccsolrad** is still allowed.

snowcomp

- Code added to initialize snow states based on **snowpack_init** and related parameters in the "init" procedure.

water_balance

- Code related to GSFLOW removed.

basin_sum

- Labeled first value of reports as initial storage.
- Added print of water balance values to reports when **print_type** = 1
- Added print of *basin_intcp_stor* and *basin_lakeevap_yr* to yearly report when **print_type** = 2

map_results

- Stop if negative mapping fraction specified.
- Allow values > 0.0 and 1.0E-06 for parameter **gvr_cell_pct** instead of treat them as 0.0.

MODULES REMOVED

`check_nhru_params` – code for module moved to function in file `utils_prms.f90` that is called within `call_modules`.
`cloud_cover` – code returned to module `ccsolrad`.

INPUT SPECIFICATION CHANGES:

`obs` and `climate_hru`

Values in Data File or CBH File are not checked for being specified as not a number (NaN), thus user must ensure such values are not specified.

`climate_hru`

Precipitation values specified in CBH Files less than 0.0 are flagged as an ERROR.

`precip_dist2`

Values specified for parameter `psta_mon` less than or equal to 0 are flagged as an ERROR.

OUTPUT FILE REMOVED

The `prms.log` file is not generated as it is redundant with information written to the model output file. The log was present to produce a similar file as output by GSFLOW and included description and version of PRMS and modules.

PRMS Version 4.0.1 – March 11, 2015

This release added new functionality, parameters, and variables, fixed bugs, added more checks for valid input values, general code clean up. The primary change is the addition of the flexible dimension option described below that allows flexibility for specification the spatial and temporal distribution of parameter values.

NEW MODULES (code taken mostly from existing modules)

<code>potet_pm</code>	Penman-Monteith potential evapotranspiration computation and distribution module, activated using control parameter <code>et_module</code> .
<code>check_nhru_params</code>	select <code>nhru</code> -dimensioned parameter values are checked to see if specified values are within the suggested range.
<code>prms_time</code>	module for time related computations, such as current time step, Julian day, and whether in summer or winter.
<code>routing</code>	computes variables used by stream flow modules, such as code related to fluxes associated with each segment.
<code>lake_route</code>	reads most lake related dimensions, parameters and computes some of variables used with lake computation modules.

<code>water_balance</code>	debug module that checks that water balances are maintained for the overall water budget, soil-zone computations, canopy interception computations, snow computations, surface runoff and impervious computations, and groundwater computations. Active when print_debug is specified equal to 1.
<code>nhru_summary</code>	produces output files of daily computed results for user-selected nhru-dimensional variables in a CSV format. Results for each selected variable are written to a separate file.

GENERAL

1. Initialize procedure is completed prior to stopping for input parameter range errors. Previously, the simulation would stop after finding the first invalid parameter value. Thus, more invalid values can be caught in a single execution. This is activated using new control parameter **parameter_check_flag**, which is described below.
2. Updated many parameter and variable descriptions to be consistent with PRMS documentation manual.
3. Initialize procedure is completed prior to stopping for input parameter range errors. Previously, the simulation would stop after finding the first invalid parameter value. Thus, more invalid values can be caught in a single execution. This is activated using new control parameter **parameter_check_flag**, which is described below.
4. Updated many parameter and variable descriptions to be consistent with PRMS users' manual.
5. Use of initial conditions options (activated using control parameters **init_vars_from_file** and **save_vars_to_file**) does not write parameter values to the Initial Condition File. Parameter values specified in the Parameter File are read and used in a restart simulation. Storage initialization parameter values, parameters **soil_moist_init**, **soil_rechr_init**, **snowpack_init**, **ssstor_init**, and **gwstor_init**, are ignored in a restart simulation. Writing to and reading from Initial Conditions Files are removed from modules `potet_hamon`, `transp_frost`, `potet_jh`, `smbal_prms`, `ssflow_prms`, `soltab`, `precip_dist2`, `hru_sum_prms`, `ccsolrad`, `ddsolrad`, `precip_1sta_laps`, `prms_summary`, `subbasin`, `basin`, `cascade`, and `map_results`. Thus, these modules do not have to be active in a restart simulation if active in a spin-up (or antecedent) simulation, except for `basin`, which is always active.
6. New variable (**IGNOREPPT=1.0E-5**) added. It is used to check for values of measured or computed precipitation values that are considered too small to be actual precipitation. If a precipitation value is less than **IGNOREPPT** the value is set to 0.0.
7. New variable (**SMALLPARAM=1.0E-4**) added. It is used to check a few parameter values for being too small to be used and thus set to 0 a warning message is printed if **parameter_check_flag** specified equal to 0. Parameters checked include **pmn_mo**, **psta_mon**, and **lake_coef**. An error message is issued and execution stops if control parameter **parameter_check_flag** is specified equal to 1 for this check.

8. Code related to computations of far field flows (flow leaving the model domain boundary not through the stream network) was removed. Adding an extra stream segment to send flow outside the domain is the recommended replacement.

NEW FUNCTIONALITY ADDED TO MODULES

Flexible dimension option: Previously, parameters had only one option for the number of values (dimension(s)) specified in the Parameter File. Now, many parameters can be specified using the original dimension(s) or using compatible dimensions up to a maximum number of values based on the specified dimension(s). For example, some parameters having dimension of **nmonths** now have a maximum dimension of **nhru,nmonths**. Flexible dimensions for a parameter with a maximum dimension of **nhru,nmonths** are **one**, **nmonths**, **nsub**, **nsub,nmonths**, **nhru**, and **nhru,nmonths**. Flexible dimensions for a parameter with a maximum dimension of **nhru** are **one**, **nsub**, and **nhru**. Flexible dimensions for a parameter with a maximum dimension of **nssr** are **one**, **nsub**, and **nssr**. Flexible dimensions for a parameter with a maximum dimension of **ngw** are **one**, **nsub**, and **ngw**. PRMS will read the dimension, number of values, and values from the Parameter File. If the parameter is not specified at the maximum dimension, the parameter values will be automatically expanded to the maximum dimension by the code. Thus, the user has several options to specify the number of parameter values based on the spatial and temporal variability, available data, or for some other purpose. The maximum number of values for most parameters has not changed. Maximum parameter dimensions are identified in PRMS-IV updated tables 1-1 and 1-3

(ftp://brrftp.cr.usgs.gov/pub/mows/software/prms/4.0.1/PRMS_tableUpdates_4.0.1.pdf). The flexible dimension option was added to accommodate simulation of large model domains that required increased spatial and/or temporal distribution of parameter values. Additionally, the number of lines in Parameter Files can be significantly reduced by specifying a single (dimension **one**) or **nsub** values for parameters that have a constant value for all HRUs or subbasins. This capability may change results when dimensions are specified greater than the original dimension(s). If the parameter dimensions are not changed, results should be the same. However, some computations in `ddsolrad` and `ccsolrad` are based on variables for each HRU rather than basin-wide variables, so the potential solar radiation (variable `swrad`) can be significantly different than previous versions for large model domains.

Parameters affected with maximum dimension indicated are: **hamon_coef(nhru,12)** in module `potet_hamon`, **jh_coef(nhru,12)** in module `potet_jh`, **radadj_intcp(nhru,12)**, **radadj_slope(nhru,12)**, **dday_intcp(nhru,12)**, **dday_slope(nhru,12)**, **tmax_index(nhru,12)**, and **radmax(nhru,12)** in module `ddsolrad`, **radadj_sppt(nhru)**, **radaj_wppt(nhru)**, **ppt_rad_adj(nhru,12)**, **crad_coef(nhru,12)**, **crad_exp(nhru,12)**, and **radmax(nhru,12)** in module `ddsolrad`, **epan_coef(nhru,12)** in modules `potet_pan` and `intcp`, **potet_sublim(nhru)** in modules `snowcomp` and `intcp`, **frost_temp(nhru)** in module `frost_date`, **tmax_lapse(nhru,12)** and **tmin_lapse(nhru,12)** in module `temp_laps`, **tmax_adj(nhru,12)** and **tmin_adj(nhru,12)** in modules `temp_1sta`, `temp_dist2`, `ide_dist`, `xyz_dist2` and `temp_laps`, **tmax_allrain(nhru,12)**, **tmax_allsnow(nhru,12)**, and **adjmix_rain(nhru,12)** in modules `precip_1sta`, `precip_dist2`, `climate_hru`, `ide_dist`, `xyz_dist` and `precip_laps`, **adjust_snow(nhru,12)** and **adjust_rain(nhru,12)** in modules `ide_dist` and `xyz_dist` **melt_look(nhru)**, **melt_force(nhru)**,

tstorm_mo(nhru,12), **cecn_coef**(nhru,12) , **emis_noppt**(nhru,12), and **freeh2o_cap**(nhru,12) in module `snowcomp`

Parameters **adj_by_hru**, **hru_subbasin**, **rain_sub_adj**, and **snow_sub_adj** that were specified for module `climate_hru` were removed as the flexible dimension option makes these unnecessary. **rain_adj** and **snow_adj** can have the dimensions (**nsub,nmonths**), so are used instead of **rain_sub_adj**, and **snow_sub_adj**. Parameter **hru_subbasin** is required if module `subbasin` is active.

An example of parameter with a maximum number of values equal to **nhru** that might have the same value for all HRUs is **carea_max**. In this case, values for each HRU can be specified to the default value as:

```
####  
carea_max  
1  
one  
2  
0.6
```

<code>map_results</code>	added option to output daily mapped results (parameter mapvars_freq specified equal to 7). Notes: the units of output variables are equal to the units of the variable for mapvars_units specified equal to 0. Units for variables could be, for example, inches, inches/day, degrees Fahrenheit, and degrees Celsius. However, when the mapvars_units value is specified greater than 0, the units of all selected map_results output variables as specified by control parameter mapOutVar_names must have units of either inches or inches/day as the code only converts inches to a metric unit.
<code>climate_hru</code>	CBH files can be input as binary files with the identical information when new control parameter cbh_binary_flag specified equal to 1. Humidity and wind speed data can be input as CBH files. Potential ET can be adjusted from values input in CBH files using new parameter potet_cbh_adj .
<code>strmflow_lake</code>	the order of stream segments and lakes was specified by parameters input to the <code>cascade</code> module with wtrbdy prefix-named parameters. The module now uses the same parameters as other stream network routing modules, that is, tosegment , hru_segment , and obsin_segment . This change requires that each lake includes a stream segment as well as being specified as a lake HRU. A segment within a lake is specified using segment_type .
<code>obs and call_modules</code>	store dimension values in Initial Conditions Files for a spin-up simulation and are checked to be sure they are the equal to the dimensions for a restart simulations.
<code>prms_summary</code>	added output of the first value of the variable <code>streamflow_cfs</code> array to the CSV file (control parameter csvON_OFF specified equal to 1); if the

value of dimension **nobs** is specified equal to 0, then the value 0.0 is output for *streamflow_cfs*.

NEW PARAMETERS

Parameters specified in the Control File:

parameter_check_flag	0 means treat some parameter range checks as WARNINGS as done for most parameters in previous versions, if specified as 1 these checks are treated as ERRORS, if specified as 2 the parameters are checked and then the simulation stops even if no ERRORS, default = 1.
cbh_check_flag	0 means do not check values in CBH file; 1 means to check for invalid values, such as NaN, character strings, < lower bound, and > upper bound, plus end of file during a simulation and non-sequential time series, such as not having the correct number of days in each year, default = 1. Specifying cbh_check_flag equal to 0 should only be done after the CBH file(s) are verified using cbh_check_flag specified equal to 1. Setting cbh_check_flag equal to 0 can reduce execution time.
cbh_binary_flag	0 means CBH files are text files as in the past; 1 means the CBH files are in binary format with same order of values as text file version.
nhruOutVar_names	array of variable names to write to CSV files each day.
nhruOutBaseFileName	string to define the prefix for each output file for the <code>nhru_summary</code> module, one for each variable; this can be a full or relative path.
nhruOutON_OFF	flag to activate <code>nhru_summary</code> module (0 means do not use module; 1 activates module, default = 0).
nhruOutVars	number of output variables, i.e., the number of variable names specified using nhruOutVars_names .
csv_output_file	pathname of output file written by the <code>prms_summary</code> module.
csvON_OFF	flag to activate the <code>prms_summary</code> module (0 means do not use module; 1 activate module, default = 0).
humidity_cbh_flag	flag to specify to read a CBH file with humidity values (0=no; 1=yes, default = 0).
humidity_day	file name of the humidity CBH file; this can be a full or relative path.
windspeed_cbh_flag	flag to specify to read a CBH file with wind speed values (0=no; 1=yes, default = 0).
windspeed_day	file name of the wind speed CBH file; this can be a full or relative path.
print_debug	new option (<code>print_debug</code> specified equal to -1) added to this existing parameter that minimizes warning messages and other messages printed to the screen during a simulation, such as when values of variables <i>slow_stor</i> are computed greater than <i>pref_flow_thrsh</i> . This can increase execution efficiency.

Parameters specified in the Parameter File(s):

dprst_et_coef(nhru)	surface-depression evaporation adjustment factor, range 0.0 to 1.0, default = 1.0.
sro_to_dprst_imperv(nhru)	impervious surface runoff fraction captured by surface depressions within each HRU. Related parameter sro_to_dprst(nhru) is pervious surface runoff capture fraction captured by surface depressions within each HRU, range 0.0 to 1.0, default = 0.2.
segment_flow_init(nsegment)	initial flow rate in each segment, in cfs, range 0.0 to 1.0E07, default = 0.0.
segment_type(nsegment)	Segment type—0=segment; 1=diversion; 2=lake; 3=replace inflow, default = 0.
hs_krs(nhru,nmonths)	calibration coefficient for module <code>potet_hs</code> ; replaces removed parameter potet_coef_hru_mo(nhru,nmonths) , range 0.005 to 0.06, default = 0.0135.
pt_alpha(nhru,nmonths)	calibration coefficient for module <code>potet_pt</code> ; replaces removed parameter potet_coef_hru_mo(nhru,nmonths) , range 1.0 to 2.0, default = 1.26.
potet_cbh_adj(nhru,nmonths)	calibration coefficient for values specified in a potet_day CBH file, range 0.5 to 1.5, default = 1.0.
snowpack_init(nhru)	initial snowpack-water equivalent, in inches, range 0.0 to 500.0, default = 0.0.

NEW COMPUTED VARIABLES (available depending on which computation options are active):

<i>canopy_covden(nhru)</i>	current value of cover density fraction for each HRU based on whether HRU is in transpiration mode.
<i>hru_intcpevap(nhru)</i>	canopy evaporation for each HRU, in inches.
<i>hru_storage(nhru)</i>	sum of all storage for all water-holding reservoir for each HRU, in inches.
<i>lake_seepage_gwr(nhru)</i>	the seepage from lakes to each GWR in units of inches.
<i>gw_seep_lakein(nhru)</i>	dimensioned changed from nhru to nlake , in inches.
<i>dprst_stor_hru(nhru)</i>	average surface depression storage for each HRU, in inches.
<i>basin_contrib_fraction</i>	basin-area weighted average contributing area fraction of pervious area.
<i>contrib_fraction(nhru)</i>	contributing area fraction of the pervious area of each HRU.
<i>basin_lake_stor</i>	basin-area weighted average precipitation into all lakes, in inches.
<i>basin_lakeevap</i>	basin-area weighted average evaporation from all lakes, in inches.
<i>unused_potet(nhru)</i>	unsatisfied potential evapotranspiration after all ET computations, in inches.
<i>basin_segment_storage</i>	basin-area weighted average total storage in each segment, in inches.
<i>segment_delta_flow(nsegment)</i>	cumulative difference of flow into minus outflow for each stream

<i>seginc_potet</i> (nsegment)	segment, in cfs.
<i>lake_inflow</i> (nlakes)	potential ET associated with each segment, in inches.
<i>lake_outflow</i> (nlakes)	total inflow into each lake, in cfs.
<i>lake_lateral_inflow</i> (nlakes)	total evaporation and seepage from each lake, in cfs.
<i>hru_streamflow_out</i> (nhru)	total lateral flow into each lake, in cfs.
<i>hru_hortn_cascflow</i> (nhru)	total flow to stream network for each HRU, in cfs.
<i>snowdepth</i> (nsnow)	name change for variable <i>hru_hortonian_cascade</i> flow, in inches.
<i>lake_outcfs</i> (nsegment)	name change of variable <i>snow</i> , in inches.
	name change of variable <i>lake_outq</i> , in cfs.

BUG FIXES – by module:

<code>snowcomp</code>	Computation of snowpack density and snowpack depth are computed based on a finite difference approximation, which produced slightly incorrect results for days when new snow falls when a snow pack does not exist. This bug could produce significant error when the new snow fall is large compared to the existing snow pack. There was the possibility that the snowpack water equivalent was computed as a value < 0 when the amount of free water was less than the computed amount of free water that the snowpack could hold. Though this condition is likely very rare, it could have resulted in very slight differences in results in the value of variables <i>pkwater_equiv</i> , <i>pk_depth</i> , and <i>snowmelt</i> .
<code>ide_dist</code> and <code>xyz_dist</code>	measured precipitation values were modified from those in the Data File, thus output values in runtime graphs or files could be different than those in the Data File; a local array is used instead.
<code>basin_sum</code>	water balance computations did not include water in stream segments when using Muskingum flow routing or surface depression storage, if any. Water balance computations are only computed when control parameter <code>print_debug = 1</code> . Used variable <i>basin_stflow_in</i> instead of <i>basin_stflow_out</i> to set the basin outflow variable <i>basin_cfs</i> , this bug only occurred when the <code>muskingum</code> module was active.
<code>basin</code>	The contributing area to each stream segment was computed incorrectly for some cases. Error check added for specification of any tosegment value equal to that segment, if this were done the code would go into an infinite loop. Checks added for valid values of tosegment; error if > nsegment, <0, or equal to the segment id. Setting of local variable <i>Hru_elev_feet</i> was not set when parameter elev_units was specified as 1; this only affected computations when module <code>potet_pt</code> is active. Print of fraction of impervious, pervious, and depression storage were labeled as area instead of as fractions of the active basin area.

gwflow

Values of parameter **gwflow_coef** specified less than 1.0 are allowed and if specified > 1.0 a warning is issued. Any water added due to specified values of parameter **gwstor_min** was added twice. Error message for the condition that an HRU is specified as a lake and the associated value of parameter **lake_hru_id** is specified equal to 0, printed the incorrect GWR Identification number. If groundwater discharge to a lake was computed greater than the available groundwater storage, the value was not limited by the available storage; now the discharge is set to the available groundwater storage and a warning message is issued. Computation of lake seepage was removed from the initialize procedure so the initial lake elevation and storage is based only on input parameters instead of after lake elevation and groundwater discharge computations as computed in the run procedure. Warning checks of parameters used incorrect array index. If any HRUs were specified as inactive then the last number of inactive HRUs values were not checked. Seepage from lakes used the incorrect array index, thus, was set incorrectly. *Basin_lake_seep* was divided by the total active area of the model domain twice, thus the value was incorrect.

soilzone

Error check added for interflow computation of the equation $\text{SQRT}(\text{coef_lin}^2 + 4.0 * \text{coef_sq} * \text{ssres_in}) = 0.0$, if true a divide by 0 would have occurred, this would be a very rare condition. Corrected setting of variable *soil_zone_max* to account for parameter **soil_moist_max** only being applicable to the pervious area of each HRU, this does not affect other soilzone computations, just the values of variables *soil_zone_max* and *soil_moist_frac*, which are computed results that are not used in other computations. Water balance check when control parameter **print_debug** = 1 was incorrect when swale HRUs are present.

ccsolrad, ddsolrad, ccsolrad_prms, and ddsolrad_prms assumed it was always winter for Southern Hemisphere applications.

write_climate_hru

if a transpiration CBH file was specified and the value for control parameter **transp_module** was specified as *climate_hru_mo*, the CBH file was not correctly produced.

frost_date

write of parameter names was incomplete as the output string was not allotted enough characters. Fix was made in functions *write_integer_param* and *write_real_param* that are found in the file *utils_prms.f90*. If a spring or fall frost was not found in a year a divide by zero error was possible. The values for output parameters **spring_frost** and **fall_frost** were output as solar days instead of calendar day, i.e., 10

transp_frost	days are now added to the computed solar day. Function <code>julian_day</code> , found in the file <code>utils_prms.f90</code> , incorrectly computed solar day.
lake_route and strmflow_lake	Function <code>julian_day</code> , found in the file <code>utils_prms.f90</code> , incorrectly computed solar day
srunoff	if an HRU is a lake, the value for the local variable <code>gwr_type</code> for that HRU was incorrectly set to 2; it is now set to 1. The amount of water used to compute Hortonian runoff from cascading surface runoff was computed as a value on pervious areas only instead of the amount of water from the entire contributing HRUs; thus could produce significant error when impervious fraction of HRUs is large. The values of variables <code>hru_sroffi</code> and <code>basin_sroffi</code> were not set correctly when parameter <code>imperv_stor_max</code> was specified equal to 0.0 for impervious portions of HRUs.

CHANGES THAT MIGHT PRODUCE SLIGHT CHANGES IN ASSOCIATED COMPUTATIONS –

general:

- Most double precision variables changed to single precision to provide more consistent floating-point comparisons and have more consistent round-off issues. Change reduces memory requirements up to 10% and slight decrease in execution time.
 - Changed variables `seginc_ssflow`, `seginc_sroff`, `seginc_gwflow`, `seginc_swrads`, `seg_outflow`, `seg_inflow`, `lake_area`, `dprst_sroff_hru`, `dprst_seep_hru`, and `hortonian_lakes` changed from single to double precision as they are sums for sets of HRUs. Only `seg_outflow` and `seg_inflow` are used in computations in multiple modules, while; the others only are computed for output purposes.
1. Changed check for very small values from 5.0E-6 to 1.0E-6 to better represent round-off error.

CHANGES THAT MIGHT PRODUCE SLIGHT CHANGES IN ASSOCIATED COMPUTATIONS – by

module:

cascade	tolerance used to check for the fraction leaving an HRU or GWR adding up to exactly 1.0 changed from 1.00001 to 1.001. If this check finds an issue the cascade links are adjusted.
potet_pan, potet_jh, potet_hamon, potet_hs, potet_pt, and potet_pm	now sets potential evapotranspiration to 0.0 for values computed less than 1.0E-6 instead of only when a computed value was negative. Thus, very small computed potential evapotranspiration values are ignored.
potet_pt	constant used to compute saturation vapor pressure changed from 17.269 to 17.26939. Equation to compute the latent heat of vaporization were computed in units of Calories/gram and then converted to kilojoules/kilogram. Now they are computed in units of kilojoules/kilogram.

ccsolrad	now sets computed canopy density to 0.0 for values < 1.0E-6 instead of only if computed value was negative. Thus, very small computed canopy density values are ignored.
intcp	values of parameters covden_win or covden_sum specified < 1.0E-4 and some computations that result in values very < 1.0E-5 are not reset to 0.0 as previously done.
snowcomp	set some and compares other computations to being less than 1.0E-6 instead of less than 0.0, which can allow for very small values of some variables. Canopy density on each HRU for each time step is used in computations instead of only values of parameter covden_win . If parameters den_init and/or den_max values are specified less than 1.0E-06, they are set to 1.0E-06 instead of 0.1 and 0.6, respectively.
srunoff	value of variable <i>hru_hortn_cascflow</i> was not reset to 0.0 for each time step; computed value is not used in other computations and is only available when cascading flow is active. Removed check for computed values of <i>imperv_stor</i> less than 0.0 and then reset to 0.0 to allow for round-off error bias to be reduced.
gwflow	computation of surface-depression storage fractions for round-off issues would sometimes produce very small closed depression storage fractions, which is now fixed.
soilzone	Added checks for computed infiltration less than 0.0 and computed interflow coefficient equal to 0.0, these would not likely occur.
ddsolrad	if computed radiation adjustment factor based on parameters dday_slope and dday_intcp is greater than the value of parameter radmax , it is set to radmax rather than as computed.
frost_date	the sum of local variables <i>fallFrostSum</i> and <i>springFrostSum</i> computed as floating point values instead of integer prior to rounding to nearest integer. These values were set to the truncated integer values, they are now rounded, that is, set to the nearest integer. Maximum value of the fall frost changed from 366 to 365 and minimum value of spring frost changed from 0 to 1. Computation of the <i>basin_fall</i> and <i>basin_spring</i> uses rounding of the integer value instead of truncation.
transp_tindex	all computations use degrees Fahrenheit instead of checking temp_units and converting computations to Fahrenheit, this should not produce a noticeable change in results, but, does increase execution efficiency. When the value of parameter transp_end equals the current month and the current day is the first day of the month transpiration is turned off (variable <i>transp_on</i> is set to 0) and the related local variables, <i>transp_check</i> and <i>tmax_sum</i> , are set to 0. After this check, if the current month equals the value of parameter transp_beg , <i>transp_check</i> and <i>tmax_sum</i> are checked to determine if transpiration needs to be turned on if the value of <i>tmax_sum</i> is greater than the value of parameter

transp_tmax. If true variable *transp_on* is set to 1, *transp_check* is set to 1 and, *tmax_sum* is set to 0. Previously, the checks were reversed, which could keep transpiration on for a few days in a month, depending on the value of *tmax_sum* when the value of **transp_beg** and **transp_end** were specified equal to each other.

muskingum allow specification of parameter **K_coef** to be less than 1.0, if found a warning message is issued instead of an error message when control parameter **parameter_check_flag** is specified greater than 0. If computed Muskingum flow coefficient (*c2*), based on parameters **K_coef** and **x_coef**, for a segment is < 0.0, this means the travel time through segment is small, thus outflow is primarily equal to inflow for the segment. If this condition is true, the computed Muskingum coefficient *c1* is decreased by the computed value of coefficient *c2* and *c2* is set to 0.0. If the computed value of Muskingum coefficient *c0* is < 0.0 then outflow from a segment would be greater than a day, the maximum travel time, which is mainly dependent on yesterday's flows. If this condition is true, the value of *c1* is decreased by *c0* and *c0* is set to 0.0. Previously, those Muskingum equation coefficients were not adjusted based on these conditions.

temp_1sta and temp_laps initial value used to replace a missing value changed from 50.0 **tmax_allrain(start_month)**, this only affects the first time step.

basin reordered code to determine variables *hru_perv*, *hru_imperv*, *dprst_frac_hru*, *dprst_area_max*, *dprst_frac_open*, *dprst_frac_clos*, *dprst_area_clos_max*, *dprst_area_open_max* to maintain restriction that the fraction of the HRU that is pervious is at least 0.0001. This change might produce slight changes in results for rare combinations of input parameters.

map_results **gvr_cell_pct** values specified less than 1.0E-6 are set to 0.0, previously the lower limit was 1.0E-10.

obs measured precipitation values specified less than 0.0001 are set to 0.0.

INPUT SPECIFICATION CHANGES – by module:

ccsolrad and ddsolrad check added to be sure at least one value of parameter **hru_solsta** is specified greater than 0 when dimension **nsol** is specified greater than 0. If all values of **hru_solsta** were specified equal to 0 an array would be referenced beyond its memory limit.

basin_sum the default value for parameter **print_freq** changed from 1 to 3.

basin error check added to be sure dimensions **nhru = nssr = ngw** when using the surface-depression storage computation option. Error check added for values of parameter **op_flow_thres** specified greater than 1.0. Added check for values of parameter **hru_percent_imperv** specified less than 0.0 or greater than 0.999. Previously, if either of these conditions were true, the values were set to 0.0 and 0.999, respectively; now an error message is printed and execution stops. Added check for HRUs not connected to a stream segment as specified using

parameter **hru_segment**. This check is applicable only when stream-segment routing and/or cascading flow routing are inactive; a warning message is issued to the screen and written in the Model Output File identifying any HRUs not connected. Added check for values of parameter **dprst_area** specified greater than the corresponding value of **hru_area** for any HRU; an error message is printed and execution stops. Added check for isolated stream segments, that is a segment that does not receive inflow or flow to another segment as specified by parameter **tosegment**); error unless control parameter **parameter_check_flag** is specified equal to 0. Added checks to be sure valid values of **elev_units** and **cov_type** are specified; if an invalid value an error message is printed and execution stops. Values of parameters **soil_moist_max** and **soil_rechr_max** are set to 0.0001 if less than 0.0001 instead of set to 0.001 if less than 0.001. Values of parameters **hru_percent_imperv**, **dprst_area**, **op_flow_thres** and computed values of **dprst_area/hru_area** are set to 0.0 if less than 0.0001 with warning message issued.

gwflow

checks added for values of parameter **gwflow_coef** specified greater than 1.0 or GWRs specified as being a swale; if these conditions are found a warning message issued.

srunoff_smidx and srunoff_carea added check for values of parameter **carea_max**

specified greater than 1.0; if true an error message is printed and execution stops. Added check for the computation of local variable *carea_max* for **srunoff_smidx** based on parameters **smidx_coef**, **smidx_exp**, and **soil_moist_max** results in a value greater than 2.0; warning message is printed if control parameter **parameter_check_flag** specified equal to 1. Added checks for values of parameters **dprst_flow_coef**, **dprst_seep_rate_open** and **dprst_seep_rate_clos** specified greater than 1.0; if true an error message is printed and execution stops.

soilzone

don't reset parameters **ssr2gw_rate**, **soil_moist_init**, **soil_rechr_init**, **ssstor_init**, **sat_threshold**, **slowcoef_lin**, **slowcoef_sq**, **fastcoef_lin**, and **fastcoef_sq** if they are specified as invalid, instead if true an error message is printed and execution stops. Specification of **soil_rechr_max>soil_moist_max**, **soil_moist_init>soil_moist_max**, **soil_rechr_init>soil_moist_max**, **ssres_stor>sat_threshold**, **soil_rechr_init>soil_rechr_max**, **soil_moist_max<0.0001**, and **soil_rechr_max<0.0001** are treated as warnings and reset to valid values instead of errors when control parameter **parameter_check_flag** specified equal to 0, previously these always were treated as errors. Added checks for **slowcoef_sq**, **slowcoef_lin**, **fastcoef_sq**, and **fastcoef_lin** specified equal to 0 as this could cause a divide by zero; if true an error message is printed and execution stops unless **parameter_check_flag** is specified equal to 0, for which a warning message is printed and any values less than 0.0001 are set to 0.0001. Check added for specifying valid values of

soil_type, which can be 1, 2, or 3; if any invalid values are specified an error message is printed and execution stops.

`climateflow` make sure parameters **basin_solsta** and **basin_tsta** are always assigned a valid value, which is needed as they are written to the restart file. Added check for computed values of *tmaxf* greater than *tminf*; if true a warning message is printed and execution continues.

`potet_pan` added check to be sure pan evaporation time series is included in Data File, if values of parameter **hru_pansta** are specified greater than dimension **nevap** or equal to 0 an error message is printed and execution stops.

`intcp` if an values of parameter **epan_coef** are specified less than 0 an error message is printed and execution stops. This check is made only when pan evaporation data are included in the Data File. If any values of parameters **covden_win** or **covden_sum** are specified less than 0 or greater than 1 an error message is printed and execution stops.

`xyz_dist` and `ide_dist` added checks to be sure data for at least 2 temperature and precipitation measurement stations are specified to be used in computations.

`xyz_dist` added check for specifying values of parameters **x_div**, **z_div**, **y_div**, **ppt_div**, **tmax_div**, and **tmin_div** equal to 0.0, as 0.0 would cause a divide by 0.0. If true an error message is printed and execution stops.

`potet_jh` added check for values of parameter **jh_coef_hru** specified greater than 150 or less than -50 and values of **jh_coef** specified greater than 10 or less than -1; if true a warning message is printed when control parameter **parameter_check_flag** is specified equal to 0; else an error message is printed and execution stops.

`map_results` added check to be sure the type of selected output variables is either real or double. Added check for selected output variables dimensioned by a value not equal to dimension **nhru**. Added checks for specification of **gvr_cell_id** values greater than dimension **ngwcell** and **gvr_hru_id** values specified greater than dimension **nhru**. If any of these checks are true an error message is printed and execution stops. If the value of parameter **mapvars_freq** is specified equal to 0 then the `map_results` module is not active even if control parameter **mapOutON_OFF** is specified equal to 1. If dimensions **nhru** and **nhruCell** are specified with equal values, then parameters **gvr_hru_id** and **gvr_cell_pct** are not required and ignored if specified in the Parameter File(s).

`temp_1sta` and `temp_laps` if too many missing values (greater than parameter **max_missing**) were found an error message is printed and the execution stops instead of continuing.

`call_modules` added checks for deprecated module names, if found a warning message is printed, but allow to maintain downward compatibility for older applications the module name is set to the current name.

OUTPUT CHANGES

- Print of list of valid modules with longer and more consistent descriptions.
- When module `climate_hru` is active for more than one type of CBH file, the module description is printed for each climate type. For example, if **climate_hru** is used to input temperature and precipitation values, a line is printed with the `climate_hru` version identification for the active Temperature Distribution and Precipitation Distribution modules. Previously, the identification was printed once no matter how many climate types were input using `climate_hru`.
- HRU identification numbers are printed allowing for seven digits instead of six throughout the code.
- Module: `temp_1sta_laps` added print of module version identification as it was missing.
- File `srunoff.f90` print of module version identification changed to indicate whether using `smidx` or `carea` method.
- Module: `basin_sum` – when control parameter **print_debug** value is specified equal to 4, some debug values were summed instead of printed individually. Write variable `basin_potsw` instead of `orad` in summary tables. Printing of water balance values changed from F9.4 to F9.3.
- New function `print_date` added in file `utils_prms.f90`, which is used to print the current time step in consistent formats for warning and error messages: modules affected: `potet_pan`, `ddsolrad`, `precip_dist2`, `ccsolrad`, `temp_dist2`, `obs`, `ide_dist`, `snowcomp`, `gwflow`, `climateflow`, and `soilzone`.
- New functions `check_param_value`, `check_param_limits`, `checkint_param_limits`, `checkdim_param_limits`, `check_param_zero`, and `check_restart_dimen` added in file `utils_prms.f90` so that duplicate code is reduced for checking many parameters for valid input values and to make warning and error messages more consistent.
- Module: `soilzone` – message for values of variables `slow_stor > pref_flow_thrsh` can be turned off as well as many warning messages for other modules if control parameter `print_debug` is specified equal to -1; this message was printing for very small differences.
- Module: `call_modules` – allow the same file name to be used control parameters `save_vars_to_file` and `init_vars_from_file`, need to be careful as this means the values stored when the file was first generated are overwritten. The start and end clock date and time and the elapsed time at end of simulation are output to the screen in Model Output file.
- Module: `prms_summary` – changed output format of variable values from F12.6 to PE14.6.

PRMS Version 3.0.5 (4/24/2012)

This release mainly focused on general code clean up and bug fixes.

General

1. MMF declmodule replaced with Fortran `print_module`

2. MMF write and open of model_output_file replaced with Fortran routines (opstr replaced with write_outfile). Affected modules: hru_sum_prms.f90, basin_sum.f90, basin.f90, call_modules.f90
3. MMF functions isleap and julian replaced with Fortran functions leap_day and julian_day
4. MMF function getstep replaced with an internal counter the current time stop (variable Timestep) is set to 0 in basin and incremented in obs.
5. Parameter checks in initialize procedure only performed for active HRUs
6. Restart capability added to all except: potet_hs.f90, potet_pt.f90, potet_hamon_prms.f, temp_2sta_prms.f, ccsolrad_prms.f, ddsolrad_prms.f, hru_sum_prms.f90, frost_date.f90, write_climate_hru.f90
7. precip_1sta.f90 and precip_laps.f90 combined into single file: precip_1sta_laps.f90
8. temp_1sta.f90 and temp_laps.f90 combined into single module: temp_1sta_laps.f90
9. srunoff_smidx.f and srunoff_carea.f combined into single module: srunoff.f90
10. transp_frost.f90: parameters spring_frost and fall_frost only checked for active HRUs to set initial values of basin_transp_on and transp_on
11. write_climate_hru.f90: changed output format from E10.2 to E12.4.
12. ddsolrad.f90: determination if the simulation time is in the summer based on equinox for the Northern and Southern Hemisphere is determined in the obs module instead of this module so that this code is in on location.
13. temp_dist2.f, precip_dist2.f, strmflo_w_lake.f, soilzone.f, climateflow.f, call_modules.f converted to F90: temp_dist2.f90, precip_dist2.f90, strmflo_w_lake.f90, soilzone.f90, climateflow.f90, call_modules.f90
14. obs.f90: make sure runoff_units has a value when nob=0; set summer_flag, increment Timestep
15. utils.f90: added check to be sure when reading orad that the variable name is swrad, which is the variable name used by write_climate_hru. This could be a problem for people who generated their own swrad CBH file and used a different variable name.
16. basin.f90: changed NEARZERO from 1.0E-7 to 5.0E-6; DNEARZERO from 1.0D-12 to 1.0D-10 to better represent round-off error. Removed parameter basin_area. Segment routing order computation moved to here and removed from streamflow routing modules to reduce duplicate code.
17. cascade.f90: code related to water-body cascades removed. Instead the tosegment is used and a stream segment must be associated with each lake if present.
18. New module: strmflo_in_out, which routes flow through the stream network without assuming a routing travel time; thus, outflow from a stream segment equals the sum of the inflow from any upstream segments and any lateral flows from associated HRUs.
19. potet_jh.f90 is a merge of files potet_jh.f90 and potet_jh_hru.f90.
20. srunoff.f90: merge of srunoff_smidx.f and srunoff_carea.f. New parameter dprst_et_coef(nhru) was added to adjust evaporation potential from surface depressions. Evaporation from open depressions is computed prior to evaporation from closed. Both are computed at the potential rate, limited by unsatisfied ET. Unsatisfied ET is reduced after computing open evaporation to then limit closed evaporation. This was being done in a complicated way, which is now streamlined.
21. strmflo_w_lake.f90: routing strategy using cascade water body code replaced with tosegment order, which requires a stream segment to be associated with each lake. Code to compute fluxes by segment, such as seginc_gwflow, seginc_ssflow, seginc_sroff, seginc_swrad, seg_lateral_inflow, hru_outflow. Module was reorganized and has had little testing

BUG FIXES

1. potet_pan.f90: check dimension nevap>0 in declare procedure instead of initialize to avoid memory error. CRITICAL BUG FIX: module would not run as return code set to 1 instead of 0. If nevap<1, simulation stops with error message. Valid values for hru_pansta checked for active HRUs only.
2. transp_index.f90: changed maximum value from 12 to 13 so module works with paramtool fix option. Set transp_end_12 for each HRU so that there is a value so as to write the variable to the restart file.
3. basin_sum.f90: print_freq = 6 did not produce monthly and yearly tables, now does. Variables basin_runoff_ratio and basin_runoff_ratio_mo were in units of inches/cfs, changed to inches/inches (decimal fraction).
4. snowcomp.f90: allow cov_type to be > 2 to designate trees, was limited to cov_type=3
5. gwflow.f90: minimum elevlake_init changed from 0.0 to -10000.0 feet. BUG FIX: with strmflow_lake active with weirs or gates, seepage_gwr was in inches when it needed to be in acre-inches.
6. muskingum.f90: default for variable TS was set to 0.0 instead of 1.0, which for K_coef values < 1.0 would cause x_max to be set to 0.0 (not sure if this was a bug, Michael identified this as a problem). Checks were added for K_coef > 24.0. If parameter_check_flag = 1, this is an error, else a warning message is issued and K_coef is set to 24. Check added for C2<NEARZERO, if so, C1 set to C1 + C2 and C2 set to 0.0 (short travel time). Check added for C0<NEARZERO, if so, C1 set to C1 + C0 and C0 set to 0.0 (long travel time). BUG FIX: the routing loop was modified to properly include streamflow_cfs(obsin_segment(iorder)) in the current inflow of a segment and to set the seg_upstream_inflow to 0.0 for each internal time step. This module needs more verification. BUG FIX: obsin_segment was not working when K_coef was not equal to 1.0, the value was divided by the timestep instead of using the input value.
7. soilzone.f90: all preferential flow computations only performed when pref_flow_den > 0.0 for an HRU, previously some were always computed when at least one HRU had a preferential flow reservoir. BUG FIX: the Dunnian flow from the gravity reservoir was not set correctly if a preferential flow reservoir was not present, this was only an issue when computing the soil zone water balance when print_debug = 1. This did not affect the results of the soil zone computations.

PRMS Version 3.0.4 (1/15/2012)

This release mainly focused on bug fixes and changes to warning and error messages.

BUG FIXES

1. Module muskingum: Value of NaN was produced for streamflow if value of K_coef < 1.
 2. Module musroute: seginc_swrad for first segment when no HRU contributes to that segment is set to second segment instead of leaving it set to 0.0.
 3. Module xclimate_hru: CBH files are checked to be sure first time step is valid and if end-of-file or missing day found during a simulation.
 4. Modules srunoff_carea and srunoff_smidx: amount of pervious and impervious surface runoff from open and closed depressions computed incorrectly resulting in too small of surface depression storage surface runoff.
 5. basin_sum: water balance and basin_stflow_tot computed and outflow from basin printed using basin_stflow_in instead of basin_stflow_out; added variable basin_runoff_ratio = basin_ppt/basin_cfs.
 6. Input error checking changed to get through reading Parameter File (mostly) before stopping on error condition instead of stopping at first input error
-

PRMS Version 3.0.3 (11/29/2012)

This version was prepared but never released because it failed the test cases.

PRMS Version 3.0.2 (9/22/2012)

This release mainly focused on bug fixes with a few minor changes to some input/output files, and parameters.

BUG FIXES

1. Module soilzone: A) limit flows produced in each reservoir to maximum capacities instead of antecedent storage + all inflow to a reservoir; this fix may result in substantial changes in results and is consistent with the PRMS conceptualization of the soil zone.
2. Module soilzone: B) set available potet for the average depth over the pervious portion instead of the whole HRU, this bug may have generated increased ET used by the capillary reservoir when impervious surfaces and surface depression storage are present in an HRU.
3. Module soilzone: C) basin_recharge was computed incorrectly, it was only set to the recharge value of the last HRU. D)if using strmflow_lake and the parameter lake_hru_id was not specified an invalid memory access could take place which would cause evaporation from lakes to be computed incorrectly.
4. Module strmflow_lake: flow out of lakes based on gate openings were computed using values specific to a single lake model, this was changed to be applicable to any model.
5. Module cascade: when greater than 2 cascades from a source to a destination, one or more of these cascades could have been ignored and fraction of cascade computed incorrectly. This would be a rare occurrence and was true for HRU and GWR cascade specifications.
6. Module intcp – A) if pan evaporation data are in Data File (nevap>0) and the ET module is not potet_pan then they are ignored rather than used as potet; B) allow winter cover density to be greater than summer density; previously if winter density was specified > summer, winter density was set to summer.
7. Module map_results: the number of output rows was computed incorrectly, if the dimension ngwcell was not equal to ngwcell/ncol and if nhru does not equal nhru/ncol.
8. Module obs – negative input values of observed runoff were set to -11; they are not changed from the values specified in the Data File, this could have caused problems with automated calibration programs.

OUTPUT AND INPUT ISSUES

1. Some variables changed from single precision to double precision, this might cause a very slight change in results that may have been due to round-off error - in modules: precip_dist2, temp_dist2, xyz_dist, strmflow_lake, basin_sum, and map_results.
2. A list of available and active modules is output to the user's screen.
3. Module climate_hru – new functionality: can input whether an HRU is in a transpiration day in a CBH file (0=no or 1=yes), previously the transpiration period was fixed for the entire simulation time period.

PARAMETER NAME CHANGES

1. temperature adjustment parameters were changed from **tmax_adj** and **tmin_adj** to **tmax_cbh_adj** and **tmin_cbh_adj** to avoid possible duplicate use of the parameter values. Similarly, precipitation adjustment parameters were changed from **rain_adj** and **snow_adj** to **rain_cbh_adj** and **snow_cbh_adj**. Check the messages about unused **tmax_adj**, **tmin_adj**, **rain_adj**, and/or **snow_adj** parameters and **tmax_cbh_adj**, **tmin_cbh_adj**, **rain_cbh_adj** and/or **snow_cbh_adj** are required but not specified. The value of **solrad_tmax** and **solrad_tmin** is now always set to **basin_tmax** and **basin_tmin** instead of using **tmax(basin_tsta)** and **tmin(basin_tsta)** when **ntemp>0**; this can produce differences in results.
2. Data File: A) missing temperature value indicator changed from less than -89 to less than -99 for modules: **temp_laps**, **temp_1sta**, **temp_dist2**, and **xyz_dist**.
3. Data File: B) added in variables humidity and wind_speed and dimensions **nwind** and **nhumid**
4. Data File and Climate-by-HRU (CBH) files: Values are checked to see if any are specified as NaN (not a number), which results in an error message and stop of the execution. If any values in a CBH files are missing, the number of missing values and data type are output to the user's screen and the execution stops.
5. Module **strmflow_lake**: Parameter and variable names using "sfres" or "res" as part of their name, changed to use "lake" instead. The dimension **nsfres** changed to **nlake**. If the dimension **nsfres** is specified, a list of all parameter names that need to change is output to the user's screen and execution halts.
6. Module **temp_dist2** - if no valid temperature values found, include in a warning message to indicate that the last valid temperature values is used for that time step.
7. Module **hru_sum_prms** has been deprecated and is only active if one or more of the following deprecated module are active: **potet_hamon_prms**, **ddsolrad_prms**, **ccsolrad_prms**, **soltab_prms**. If module **hru_sum_prms** is not active, parameters **pmo** and **moyrsum** are not needed.
8. Variable **basin_stflow** is replaced with **basin_stflow_in**, the total amount of lateral flow into the stream network. Variable **basin_stflow_out** was added, which is the total amount streamflow out of the model domain.
9. Module **map_results** –Added check to be sure the mapping specification is complete (**gvr_cell_id(i)>0**, **gvr_hru_id(i)>0**, and **gvr_cell_pct>1.0E-10**) from the HRU map to the target map; a warning message is output to the user's screen if this condition is detected. This check produces a warning message, as in some cases it might be desirable to map only a portion of an HRU to a target map spatial unit, such as, at a model boundary where the HRU map and target map have different spatial extents.

PRMS Version 3.0.1 (2/6/2012)

This release mainly focused on improving the messages that come to the command line window. Many users of PRMS were ignoring the error and warning messages because that they could not be understood. Also, users were uncertain about which modules were active in the runs they were making. The new messages will help with both of these cases.

1. There were some changes made to increase the precision of the snow and soil moisture module algorithms. Some of the FORTRAN variables were changed from single precision to double precision floating points. The consequence of this is some changes in the results in the simulation of snowpack storage and melt, and soil-moisture storage and recharge. These changes are evident in (sometimes) the 4th and (usually) the 5th significant digit.

2. No changes were made in this version which impact the format or content of the input and output files in any way.

PRMS Version 3.0.0 (11/15/2011)

This version of PRMS has been designed as a stand-alone program that can be executed on a Linux or Microsoft Windows platform. In some ways, this version may appear to return to the concepts and design of the earliest versions of PRMS. Much of the support functionality provided by MMS has been stripped away in favor of a “batch execution” mode for maximum application flexibility and computational efficiency. This approach also supports maximum portability between computers running the Windows and Linux operating systems. Focus has been placed on ease of deployment, installation, and reliability over the MMS concepts of “model building.” However, the module and function library developed for the MMS version of PRMS have been shown to be useful and have been retained.

MMS Version MMS-windows-07-01-26 (1/26/2007)

The MMS version of PRMS was ported to PCs running the Windows operating system using Cygwin.

MMS Versions 1.0.0 through 1.2.1 (1991 through 2002)

- These were UNIX only releases. Although computationally efficient, the procedure required to add hydrologic-process algorithms to the original code was less than adequate. As a result, the architecture and modular structure of PRMS were redesigned to allow better integration and hydrologic-process algorithm-development capabilities. This new structure was the USGS Modular Modeling System (MMS), an integrated system of computer software developed for simulating a variety of water, energy, and biogeochemical processes that included PRMS.
- The basic hydrologic-process algorithms in PRMS were maintained in the MMS version; however, the use of MMS enabled the addition of new process algorithms and the enhancement of many of the features and capabilities in the original PRMS. These additions included graphical systems and networked data systems that took advantage of increased computational power.

PRMS Version 2.1 (1/17/1996)

This was the pre-MMS version of PRMS. It was precompiled for computers running the DOS and DG UNIX operating systems.

PRMS Pre-version 2.0 (1983 - 1996)

PRMS originally was developed as a single FORTRAN 77 program composed of algorithms encoded in subroutines, each representing an individual physical process of the hydrologic cycle.

For the processes related to temperature distribution, solar-radiation distribution, evaporation, transpiration, and surface runoff, two or more different algorithms were encoded, each representing a different conceptual approach. A specific algorithm was selected at run time by setting values in the input file. This modular-design concept enabled the creation and application of a model that was most appropriate for a given study and supported the long-term goal to expand the available process-simulation capabilities of PRMS.

This version of PRMS uses “punch card” formats for input files, line-printer-generated output plots, use of the USGS’s National Water Data Storage and Retrieval system, and the job-control language specifications necessary to execute PRMS on the Amdahl and Prime computer systems.