Raven Cheat Sheet

Parameter Commands (*.rvp/*.rvt)

rvp

```
:GlobalParameter [PARAM_NAME] [value]
:VegetationParameterList
  :Parameters,  [PARAM_NAME1],[PARAM_NAME2],...
  :Units,            [UNITS1],      [UNITS2],...
  [DEFAULT],     [pvalue_def], [pvalue_def],...
  VCLASS_a,        [pvalue1a][1],  [pvalue2a],...
  VCLASS_b,        [pvalue1b],    [pvalue2b],...
  ...
:EndVegetationParameterList
```

Also for:

:SoilParameterList-:EndSoilParameterList

:LandUseParameterList-:EndLandUseParameterList

[1]can take value of `_DEFAULT` (uses default class value) or `_AUTO` (parameter estimated by Raven)

```
:VegetationChange [HRU grp] [new class] YYYY-mm-dd
:LandUseChange    [HRU grp] [new class] YYYY-mm-dd
:HRUTypeChange    [HRU grp] [new type]  YYYY-mm-dd
```

```
:TransientParameter [NAME] [class] [classname] *
```

*Use TS formats

①②③

rvt

```
:EndTransientParameter
e.g.
:TransientParameter POROSITY SOIL LOAM
:TransientParameter MAX_LAI VEGETATION CORN
```

classes: `SOIL VEGETATION LANDUSE TERRAIN GLOBALS`

Note: Raven ignores units and will not do units conversion

Command line options and mode controls

`Raven.exe [modelname] -flag [flagoption]`

`-o ./output/` - specify output folder

`-t my_rvt.rvt` – specify rvt file to use

`-c solution.rvc` – specify rvc file to use (e.g., warm start)

`-i, -p, -h, -e, -l` – specify rvi/rvp/rvh/rve/rvl file to use

`-r` – specify run name

`-m A` – specify run mode as mode 'A'

`-s` – silent mode

`-n` - noisy mode

rvi

```
:RunMode
:IfModeEquals A C
  #... any commands
:EndIfModeEquals
```

rvi/rvh/rvp/rvt/rvc/rve

-runs the commands within the if statement if the current mode matches any of the modes in the list

Soil Profile Commands (*.rvp)

rvp

```
:SoilProfiles
 [name, #Horizons,class1[1],thick1[2],class2,thick2]
```

e.g.,

```
 HORIZ_A, 2, SILTY_SAND, 0.8, CLAY, 1.2
 HORIZ_B, 3, SILTY_SAND, 0.4, SILT, 0.6, CLAY, 0.5
:EndSoilProfiles
```

[1]Class name from `:SoilClasses` command [2]Thicknesses in metres

Special 'soil' profiles with zero soil horizons:

`GLACIER, 0`	Rain/melt accumulates as `PONDED_WATER`
`LAKE, 0`	Rain/melt added to `LAKE` storage
`ROCK, 0`	Rain/melt shed as runoff

Temperature Simulation Commands (*.rvi, *.rvh)

rvi

```
:Transport TEMPERATURE

:FixedTemperature TEMPERATURE [SV] [value] {grp}
e.g.,
:FixedTemperature TEMPERATURE ATMOS_PRECIP -9999
:FixedTemperature TEMPERATURE SNOW         0.0
:FixedTemperature TEMPERATURE SOIL[3]   3.0 midHrus
```

rvh

All non-headwater subbasins require water HRU to represent reach, indicated using subbasin parameter HRU_REACH_ID

Key in-reach model parameters :

- subbasin properties CONVECT_COEFF (~2.0), HYPORHEIC_FLUX, RIVERBED_CONDUCTIVITY, and RIVERBED_THICKNESS;
- for lakes: LAKE_CONVECT_COEFF LAKEBED_CONDUCTIVITY, LAKEBED_THICKNESS;
- channel cross section and lake/reservoir depth.

State variables (SV): `SOIL[0] SOIL[n] SNOW SNOW_LIQ COLD_CONTENT CANOPY_SNOW CANOPY DEPRESSION WETLAND LAKE_STORAGE PONDED_WATER ATMOS_PRECIP ATMOSPHERE SNOW_DEPTH ICE_THICKNESS THAW_DEPTH SNOW_COVER GLACIER GLACIER_ICE SNOW_DEFICIT SNOW_ALBEDO CONVOLUTION`

For Reporting: `AET, RUNOFF, STREAMFLOW, RESERVOIR_STAGE`

For transport variables: `![constname]|[store],`
`e.g., !NITRATE|SOIL[0] or !TEMPERATURE|CANOPY`

Raven Cheat Sheet

Output Commands (*.rvi)

rvi

`:RunName [name]` – adds prefix to output filenames

`:OutputDirectory [dir]` – sends output to here

`:CreateRVPTemplate` – generates .rvp template file

`:WriteMassBalanceFile` – writes all internal fluxes to file

`:WriteForcingFunctions` – writes all forcings to file

`:WriteEnsimFormat` - output written to .tb0

`:WriteNetCDFFormat` –output written to .nc

`:SilentMode` – no console output `:NoisyMode` – lots of output

`:CustomOutput [time] [stat] [var] [space]`

time: DAILY MONTHLY YEARLY WATER_YEARLY CONTINUOUS
stat: AVERAGE MAXIMUM MINIMUM RANGE MEDIAN QUARTILES
var: SOIL[0] SOIL[1] SNOW PET From:SNOW To:SNOW Between:SOIL[0].And.ATMOSPHERE ...
(all state variables or forcings)
space: BY_BASIN BY_HRU BY_HRU_GROUP BY_WATERSHED

Calibration/Model Evaluation (*.rvi/.rvt)

rvi

```
:EvaluationMetrics [diag1] [diag2] ...
:EvaluationPeriod [name] yyyy-mm-dd yyyy-mm-dd
```

diag: NASH_SUTCLIFFE PCT_BIAS LOG_NASH KLING_GUPTA RMSE ABSERR ABSMAX RCOEF R2 NSC RSR PDIFF TMVOL MBF PERSINDEX DAILY_KGE

rvt

```
:ObservationData [datatype] [ID] {units} *
e.g.,:ObservationData HYDROGRAPH [basinID] m3/s
e.g.,:ObservationData SNOW [HRU_ID] mm
:ObservationWeights [datatype] [ID] *
```

*Use TS formats ①②③

```
:IrregularObservations [data type] [ID] [N]
  [yyyy-mm-dd] [hh:mm:ss.0] value1
  ...
  [yyyy-mm-dd] [hh:mm:ss.0] valueN
:EndIrregularObservations
```

④

```
:IrregularWeights [data type] [ID] [N] **
:OverrideReservoirFlow [basinID] *
:OverrideStreamflow [basinID] *
```

**Use TS format ④

HRU Groups (*.rvi/*.rvh)

<u>HRU groups</u> – used for conditional processes, custom output, HRU disabling, transport boundary conditions, etc. [define in .rvi file]. Groups are named (and used) in the .rvi, but populated in the .rvh.

rvi

```
:DefineHRUGroups [grp1] {grp2} ... {grpN}
```

rvh

```
:HRUGroup [name]
  18,20, 32-49
:EndHRUGroup
```

– populates HRU groups by HRU ID, ID range

rvh

```
:PopulateHRUGroup [grp] With [condition]
```

e.g.,
```
CropHRUs With LANDUSE EQUALS CROPLAND
NonCropHRUs With LANDUSE NOTEQUALS CROPLAND
BroadHRUs With VEGETATION EQUALS BROADLEAF
NotRock With HRUS NOTWITHIN RockHRUGroup
LowBand With ELEVATION BETWEEN 0 500
```

Time Series Commands (*.rvt)

rvt

```
:Gauge {name}
  :Latitude  [lat]
  :Longitude [long]
  :Elevation [elev]
  :RedirectToFile [gaugedata.rvt]
:EndGauge
:Data [forcing] {unit}
  [yyyy-mm-dd] [hh:mm:ss.0] [interval (d)] [N]
  value1
  ...
  valueN
:EndData
```

①

*These time series specification formats also work for time series commands below

Blank data=-1.2345

```
:Data [forcing] {unit}
  :ReadFromNetCDF
    :FileNameNC     [.nc file]
    :VarNameNC      [variable in .nc file]
    :DimNamesNC     [station] [time_name]
    :StationIdx     [ID]
  :EndReadFromNetCDF
:EndData
```

②

```
:Data [forcing] {unit}
  :AnnualCycle J F M A M J J A S O N D
:EndData
```

③

forcings: PRECIP TEMP_MIN TEMP_MAX TEMP_AVE SNOWFALL RAINFALL PET OW_PET WIND_VEL AIR_DENS AIR_PRES REL_HUMIDITY SW_RADIA SW_RADIA_NET LW_INCOMING LW_RADIA_NET POTENTIAL_MELT RECHARGE

Reservoir Commands (*.rvh/*.rvt)

Reservoirs – used to represent lakes or reservoirs at *outlet* of subbasin.

rvh

```
:Reservoir {name}
  :SubBasinID [SBID]
  :HRUID [HRUID]
  :StageRelations
    [N]
    [stage, flow, volume, area, {underflow}]x[N]
  :EndStageRelations
:EndReservoir
```

Type 1: (managed reservoirs) uses stage-discharge-volume curves

rvh

```
:Reservoir {name}
  :SubBasinID [SBID]
  :HRUID [HRUID]
  :WeirCoefficient [C]
  :CrestWidth [width [m]
  :MaxDepth [depth [m]]
  :LakeArea [area [m2]]
 {:AbsoluteCrestHeight [elevation [masl]]}
:EndReservoir
```

Type 2: (natural lakes) uses overflow weir relation; calibrate crest width

rvt

```
:ReservoirExtraction [basinID] *
:VariableWeirHeight [basinID] *
:ReservoirMaxStage [basinID] *
:ReservoirMinStage [basinID] *
:ReservoirMinStageFlow [basinID] *
:BasinInflowHydrograph [basinID] *
```

*Use TS formats ①②③