

SWAT

Generated by Doxygen 1.8.16



<b>1 SWAT</b>	<b>1</b>
<b>2 Modules Index</b>	<b>7</b>
2.1 Modules List . . . . .	7
<b>3 Data Type Index</b>	<b>9</b>
3.1 Data Types List . . . . .	9
<b>4 File Index</b>	<b>11</b>
4.1 File List . . . . .	11
<b>5 Module Documentation</b>	<b>13</b>
5.1 parm Module Reference . . . . .	13
5.1.1 Detailed Description . . . . .	80
5.1.2 Variable Documentation . . . . .	80
5.1.2.1 igropt . . . . .	81
<b>6 Data Type Documentation</b>	<b>83</b>
6.1 parm::ascrv Interface Reference . . . . .	83
6.2 parm::atri Interface Reference . . . . .	83
6.3 parm::aunif Interface Reference . . . . .	83
6.4 parm::dstn1 Interface Reference . . . . .	84
6.5 parm::ee Interface Reference . . . . .	84
6.6 parm::expo Interface Reference . . . . .	84
6.7 parm::fcgd Interface Reference . . . . .	84
6.8 parm::HQDAV Interface Reference . . . . .	85
6.9 parm::layersplit Interface Reference . . . . .	85
6.10 parm::ndenit Interface Reference . . . . .	85
6.11 parm::qman Interface Reference . . . . .	85
6.12 parm::regres Interface Reference . . . . .	86
6.13 parm::rsedaa Interface Reference . . . . .	86
6.14 parm::tair Interface Reference . . . . .	86
6.15 parm::theta Interface Reference . . . . .	86
6.16 parm::vbl Interface Reference . . . . .	86
<b>7 File Documentation</b>	<b>87</b>
7.1 allocate_parms.f90 File Reference . . . . .	87
7.1.1 Detailed Description . . . . .	87
7.2 ascrv.f90 File Reference . . . . .	87
7.2.1 Detailed Description . . . . .	87
7.2.2 Function/Subroutine Documentation . . . . .	88
7.2.2.1 ascrv() . . . . .	88
7.3 aunif.f90 File Reference . . . . .	88
7.3.1 Detailed Description . . . . .	88

7.3.2 Function/Subroutine Documentation	89
7.3.2.1 aunif()	89
7.4 caps.f90 File Reference	89
7.4.1 Detailed Description	89
7.4.2 Function/Subroutine Documentation	89
7.4.2.1 caps()	89
7.5 estimate_ksat.f90 File Reference	90
7.5.1 Detailed Description	90
7.5.2 Function/Subroutine Documentation	90
7.5.2.1 estimate_ksat()	90
7.6 gcycl.f90 File Reference	90
7.6.1 Detailed Description	91
7.7 getallo.f90 File Reference	91
7.7.1 Detailed Description	91
7.8 h2omgt_init.f90 File Reference	91
7.8.1 Detailed Description	91
7.9 headout.f90 File Reference	92
7.9.1 Detailed Description	92
7.10 hruallo.f90 File Reference	92
7.10.1 Detailed Description	92
7.11 hydroinit.f90 File Reference	92
7.11.1 Detailed Description	92
7.12 impnd_init.f90 File Reference	93
7.12.1 Detailed Description	93
7.13 jdt.f90 File Reference	93
7.13.1 Detailed Description	93
7.13.2 Function/Subroutine Documentation	93
7.13.2.1 jdt()	93
7.14 lwqdef.f90 File Reference	94
7.14.1 Detailed Description	94
7.15 main.f90 File Reference	94
7.15.1 Detailed Description	94
7.16 modparm.f90 File Reference	95
7.16.1 Detailed Description	162
7.17 openwth.f90 File Reference	162
7.17.1 Detailed Description	162
7.18 qman.f90 File Reference	162
7.18.1 Detailed Description	163
7.18.2 Function/Subroutine Documentation	163
7.18.2.1 qman()	163
7.19 readatmodep.f90 File Reference	163
7.19.1 Detailed Description	163

7.20 readbsn.f90 File Reference . . . . .	164
7.20.1 Detailed Description . . . . .	164
7.21 readchm.f90 File Reference . . . . .	164
7.21.1 Detailed Description . . . . .	164
7.22 readcnst.f90 File Reference . . . . .	164
7.22.1 Detailed Description . . . . .	164
7.23 readfcst.f90 File Reference . . . . .	165
7.23.1 Detailed Description . . . . .	165
7.24 readfert.f90 File Reference . . . . .	165
7.24.1 Detailed Description . . . . .	165
7.25 readfig.f90 File Reference . . . . .	165
7.25.1 Detailed Description . . . . .	165
7.26 readfile.f90 File Reference . . . . .	166
7.26.1 Detailed Description . . . . .	166
7.27 readgw.f90 File Reference . . . . .	166
7.27.1 Detailed Description . . . . .	166
7.28 readhru.f90 File Reference . . . . .	166
7.28.1 Detailed Description . . . . .	166
7.29 readinpt.f90 File Reference . . . . .	167
7.29.1 Detailed Description . . . . .	167
7.30 readlup.f90 File Reference . . . . .	167
7.30.1 Detailed Description . . . . .	167
7.31 readlwq.f90 File Reference . . . . .	167
7.31.1 Detailed Description . . . . .	167
7.32 readmgt.f90 File Reference . . . . .	168
7.32.1 Detailed Description . . . . .	168
7.33 readmon.f90 File Reference . . . . .	168
7.33.1 Detailed Description . . . . .	168
7.34 readops.f90 File Reference . . . . .	168
7.34.1 Detailed Description . . . . .	168
7.35 readpest.f90 File Reference . . . . .	169
7.35.1 Detailed Description . . . . .	169
7.36 readplant.f90 File Reference . . . . .	169
7.36.1 Detailed Description . . . . .	169
7.37 readpnd.f90 File Reference . . . . .	169
7.37.1 Detailed Description . . . . .	169
7.38 readres.f90 File Reference . . . . .	170
7.38.1 Detailed Description . . . . .	170
7.39 readrte.f90 File Reference . . . . .	170
7.39.1 Detailed Description . . . . .	170
7.40 readru.f90 File Reference . . . . .	170
7.40.1 Detailed Description . . . . .	170

7.41 readsdr.f90 File Reference . . . . .	171
7.41.1 Detailed Description . . . . .	171
7.42 readsepticbz.f90 File Reference . . . . .	171
7.42.1 Detailed Description . . . . .	171
7.43 readseptwq.f90 File Reference . . . . .	171
7.43.1 Detailed Description . . . . .	171
7.43.2 Function/Subroutine Documentation . . . . .	172
7.43.2.1 readseptwq() . . . . .	172
7.44 readsno.f90 File Reference . . . . .	172
7.44.1 Detailed Description . . . . .	172
7.45 readsol.f90 File Reference . . . . .	172
7.45.1 Detailed Description . . . . .	172
7.46 readsub.f90 File Reference . . . . .	173
7.46.1 Detailed Description . . . . .	173
7.47 readswq.f90 File Reference . . . . .	173
7.47.1 Detailed Description . . . . .	173
7.48 readtill.f90 File Reference . . . . .	173
7.48.1 Detailed Description . . . . .	173
7.49 readurban.f90 File Reference . . . . .	174
7.49.1 Detailed Description . . . . .	174
7.50 readwgn.f90 File Reference . . . . .	174
7.50.1 Detailed Description . . . . .	174
7.51 readwus.f90 File Reference . . . . .	174
7.51.1 Detailed Description . . . . .	174
7.52 readwwq.f90 File Reference . . . . .	175
7.52.1 Detailed Description . . . . .	175
7.53 readyr.f90 File Reference . . . . .	175
7.53.1 Detailed Description . . . . .	175
7.54 rteinit.f90 File Reference . . . . .	175
7.54.1 Detailed Description . . . . .	175
7.55 sim_inityr.f90 File Reference . . . . .	176
7.55.1 Detailed Description . . . . .	176
7.56 simulate.f90 File Reference . . . . .	176
7.56.1 Detailed Description . . . . .	176
7.57 soil_chem.f90 File Reference . . . . .	176
7.57.1 Detailed Description . . . . .	176
7.57.2 Function/Subroutine Documentation . . . . .	176
7.57.2.1 soil_chem() . . . . .	176
7.58 soil_phys.f90 File Reference . . . . .	177
7.58.1 Detailed Description . . . . .	177
7.58.2 Function/Subroutine Documentation . . . . .	177
7.58.2.1 soil_phys() . . . . .	177

7.59 std1.f90 File Reference . . . . .	177
7.59.1 Detailed Description . . . . .	178
7.60 std2.f90 File Reference . . . . .	178
7.60.1 Detailed Description . . . . .	178
7.61 std3.f90 File Reference . . . . .	178
7.61.1 Detailed Description . . . . .	178
7.62 storeinitial.f90 File Reference . . . . .	178
7.62.1 Detailed Description . . . . .	179
7.63 ttcoef.f90 File Reference . . . . .	179
7.63.1 Detailed Description . . . . .	179
7.63.2 Function/Subroutine Documentation . . . . .	179
7.63.2.1 ttcoef() . . . . .	179
7.64 xmon.f90 File Reference . . . . .	179
7.64.1 Detailed Description . . . . .	180
7.65 zero0.f90 File Reference . . . . .	180
7.65.1 Detailed Description . . . . .	180
7.66 zero1.f90 File Reference . . . . .	180
7.66.1 Detailed Description . . . . .	180
7.67 zero2.f90 File Reference . . . . .	180
7.67.1 Detailed Description . . . . .	181
7.68 zero_urban.f90 File Reference . . . . .	181
7.68.1 Detailed Description . . . . .	181
7.69 zeroini.f90 File Reference . . . . .	181
7.69.1 Detailed Description . . . . .	181
<b>Bibliography</b>	<b>183</b>
<b>Index</b>	<b>185</b>





# Chapter 1

## SWAT

An updated SWAT 2012 revision 670 code

### Objectives

- Standard indentation and translation to Fortran 90 by using `findent`. See the `translate-fortran90.pl` perl script file (:heavy\_check\_mark:)
- Exhaustive use of the "implicit none" directive to detect bad variable usage (:heavy\_check\_mark:)
- Generate a GNU `Make` makefile and compile with GNU `GFortran`. See the `generate-makefile.pl` perl script file (:heavy\_check\_mark:)
- Remove non-used variables and format labels (:heavy\_check\_mark:)
- Detect and solve all uninitialized variables (:heavy\_check\_mark: :construction:, some proposed solutions could be incorrect)
- Remove unneeded variable initializations (:heavy\_check\_mark:) as:  

```
j=0 ! this line is not necessary  
j=ihru
```
- Remove redundant code (:heavy\_check\_mark:)
- Exhaustive use of the "parameter" directive on constants (:heavy\_check\_mark:)
- Remove global counters (as `i` or `idum` in module `parm`). Using local counters or passing values as argument are preferred (:construction:)
- Generate a detailed list of issues detected in the original code (:heavy\_check\_mark:, see at the end of this README)
- Remove obsolete commented code (:x:)
- Update variable descriptions in comments (:construction:, a lot of work)
- Standardize comments by using Doxygen style in order to generate documentation. See at `latex/refman.pdf` (:construction:, a lot of work)

## Required tools

- `GFortran` (to compile the source code)
- `Make` (to build the executable file)
- `Perl` (optional: to execute the perl scripts to update the makefile or to translate original files to Fortran 90)
- `Findent` (optional: to translate original files to Fortran 90 with a standard indentation)
- `Doxygen` (optional: to generate a reference programming manual from source code)
- `TeX Live` or `MiKTeX` (optional: to generate a reference programming manual from source code)
- On Microsoft Windows systems you have to install `MSYS2` and the required utilities ( `GFortran` and `Make`). You can follow detailed instructions in `install-unix`

## Instructions to generate Fortran 90 style code from original code

In order to generate Fortran 90 style code with standard indentation from original code you have to type on a UNIX type terminal (you need `Perl` and `Findent`):

```
$ perl translate-fortran90.pl
```

## Instructions to generate an initial GNU make Makefile

Type on the UNIX type terminal, when translated the original code to Fortran 90 style (you need `Perl`):

```
$ perl generate-makefile.pl
```

## Instructions to generate an executable to test

Type on the UNIX type terminal (you need `GFortran` and `Make`)

- In UNIX type operative systems:

```
$ make
```

- In a `MSYS2` terminal in Microsoft Windows:

```
$ EXE=".exe" LDFLAGS="-static" make
```

- Cross-compiling a 32 bits Microsoft Windows executable in a UNIX type operative system:

```
$ prefix="i686-w64-mingw32-" EXE=".exe" LDFLAGS="-static" make
```

- Cross-compiling a 64 bits Microsoft Windows executable in a UNIX type operative system:

```
$ prefix="x86_64-w64-mingw32-" EXE=".exe" LDFLAGS="-static" make
```

## Instructions to generate an optimized executable file

Type on the UNIX type terminal (you need **GFortran** and **Make**)

- In UNIX type operative systems:

```
$ CFLAGS="-march=native -flto" LDFLAGS="-flto" make strip
```

- In a **MSYS2** terminal in Microsoft Windows:

```
$ EXE=".exe" CFLAGS="-flto" LDFLAGS="-flto -static" make strip
```

- Cross-compiling a 32 bits Microsoft Windows executable in a UNIX type operative system:

```
$ prefix="i686-w64-mingw32-" EXE=".exe" CFLAGS="-flto" LDFLAGS="-flto -static" make strip
```

- Cross-compiling a 64 bits Microsoft Windows executable in a UNIX type operative system:

```
$ prefix="x86_64-w64-mingw32-" EXE=".exe" CFLAGS="-flto" LDFLAGS="-flto -static" make strip
```

## Instructions to generate a reference programming manual from source code

Type on the UNIX type terminal (you need **Doxygen** and **TeX Live** or **MiKTeX**):

```
$ make latex/refman.pdf
```

The reference programming manual file latex/refman.pdf is generated from source code in PDF format

## Issues in the original source code

This is a list of possible issues detected in the original source code. These issues have been mostly detected by the **GFortran** compiler warnings. Some of them could not arise because the logic of the variables is not possible.

- In biofilm.f:

- `dcoef` is used but not initialized. `dcoef=3` as in `watqual.f`? Then, I propose at beginning: `real*8, parameter :: dcoef = 3.`

- In `bmp_ri_pond.f`:

- `qseep` and `qet` could be used not initialized at lines 133 and 134. However the problem only arises for `nstep<1`

- In `bmp_sand_filter.f`:

- `sed_removed` at line 342 could be used not initialized if `sfsedstdev<=0`

- In `bpm_sed_pond.f`:

- `bmp_sed_pond` seems to be `bmp_sed_pond` at line 186

- In `bmp_wet_pond.f`:

- `hvol` could be used not initialized in `ext_dpth` subroutine at line 267 in first bucle iteration

- In `clicon.f`:
  - `tmxbsb`, `tmnbsb`, `rbsb`, `rstpbsb`, `rhdbbsb`, `rabsb`, `rmxbsb`, `daylbsb`, `fradbsb` and `u10bsb` could be used not initialized at 186-207 lines
- In `conapply.f`:
  - `k` and `kk` could be used not initialized at 121-122 lines if `iday_pest(j) /= ipst_freq(j)` and `curyr > nyskip`
- In `confert.f`:
  - `ifrt` seems to be it at line 214
- In `curno.f`:
  - `smxold` could be used not initialized if `cn1(h) <= 1.e-6` and `curyr /= 0` at line 96
- In `drains.f`:
  - `nlayer` could be used not initialized at line 23. However, the problem only arises if it is not set in the previous bucle (`mlyr <= 1` or `sol_z(j1, j) <= 0`)
- In `etact.f`:
  - `sev` could be used not initialized at line 286 if `dep >= esd` and `ly == 2`
- In `filter.f`:
  - `remove21` seems to be `remove2` at line 316
- In `grass_wway.f`:
  - `sf_depth` and `sf_sed` could be used not initialized at lines 133 and 137 if `sf_area > 0` and `sf_area <= 1.e-6`
- In `hhnoqual.f`:
  - `algon` seems to be `algcon` at line 190
- In `hhwatqual.f`
  - `orgnpin` seems to be `orgpin` at line 278
  - `thour=1.0` at line 377 overwrites previous `thour` calculation. It is wrong
- In `hmeas.f`:
  - `rhdbbsb` could be used not initialized at line 84
- In `killop.f`:
  - `ff1` and `ff2` are used but not initialized at lines 167 and 267. They are set in `harvkillop.f` file (lines 257-258). They have to be included in `modparm.f` to share `harvkillop.f` values? or they have to be redefined as in `harvkillop.f`?
- In `NCsed_leach.f90`:
  - `perc_clyr` could be used not initialized at line 221 if `sol_nly(j) < 2`
- In `nrain.f`:
  - `no2pcp` seems to be `no3pcp` at line 72
- In `pmeas.f`:
  - `rbsb` could be used not initialized at line 143
  - `flag` could be used not initialized if `'a=='` at line 210 `-rainsb` could be used not initialized, however only if `fnstep <= 0`

- In `pminrl2.f`:
  - at line 95 a comma is necessary between `base` and `vara`
  - `ssp` could be used not initialized at line 196 if `xx<=1.e-6`
- In `pothole.f`:
  - `solp_tileo` could be used not initialized at line 593 if `pot_vol(j)<=1.e-6` or `potvol_↵tile<=1.e-6`
- In `potholehr.f`:
  - `potflow` seems to be `potflwo` at line 447
- In `readatmodep.f`:
  - `momax=12*nbyr` is defined at line 65 but not used. It has to be `mo_max?` but then, it overwrites the file read
- In `readops.f`:
  - `year = 0.` seems to be `iyyear = 0` at line 98
  - `mg13` seems to be `mgt13` at line 206
- In `readpnd.f`:
  - `vselsetlpnd` seems to be `velsetlpnd` at line 279
- In `readru.f`:
  - `tck` is used but not initialized at line 79
- In `readsepticbz.f`:
  - at line 135 `4.e-8` seems to be `4.e-8`
- In `rewind_init.f`:
  - `orig_tnylda` is used but not initialized at line 174
- In `routes.f`:
  - `dstor` is used but not initialized at line 134. It has to be calculated as in `watbal.f?` or as in the commented line 109?
  - `latqout` and `gwqout` could be used not initialized at lines 142-143
- In `rtbact.f`:
  - `netwtr` could be used not initialized at line 124, however only if `nstep<1`
- In `rthpest.f`:
  - `thour=1.0` at line 183 overwrites previous `thour` calculation. It is wrong
  - `frsol` and `frsrb` could be used not initialized at lines 289-290 if `hrtwtr(ii)>0.001` and `hrtwtr(ii)/(idt*60)<=0.01`
- In `rtpest.f`:
  - `tday=1.0` at line 180 overwrites previous `tday` calculation. It is wrong
- In `sched_mgt.f`:
  - `< =` seems to be `<=` at 202 line
  - `husc` and `igrow` at lines 264-265 are used but not initialized. `husc` has to be `phu_op(iop, ihru)` has in `readmgt.f?` `igrow` has to be `igro(ihru)` has in `readmgt.f?`
- In `smeas.f`:

- `rabsb` could be used not initialized at line 86
- In `sweep.f`:
  - `fr_curb` is used but not initialized at line 56. It has to be added to `modparm.f` to share result with `sched_mgt.f`? or it has to be `mgt5op(nop(ihru), ihru)` as in `sched_mgt.f`?
- In `tmeas.f`:
  - `tmxbsb` and `tmnbsb` could be used not initialized at lines 109-110
- In `transfer.f`:
  - `ratio, xx` and `ratio1` could be used not initialized at lines 236, 239 and 241 if `ihout==2`
- In `wmeas.f`:
  - `u10bsb` could be used not initialized at line 85
- In `zero0.f`:
  - `sol_sumn03` seems to be `sol_sumno3` at line 508
- In `zero_urban.f`:
  - `stp_stagdis` seems to be `stp_stagdis` at line 84
  - `subdr_kg` seems to be `subdr_km` at line 149
  - `spl_eros` is not defined at line 21, it could be `eros_spl`?

## Chapter 2

# Modules Index

### 2.1 Modules List

Here is a list of all documented modules with brief descriptions:

<a href="#">parm</a>	Main module containing the global variables . . . . .	<a href="#">13</a>
----------------------	---	--------------------





## Chapter 3

# Data Type Index

### 3.1 Data Types List

Here are the data types with brief descriptions:

<a href="#">parm::ascrv</a>	83
<a href="#">parm::atri</a>	83
<a href="#">parm::aunif</a>	83
<a href="#">parm::dstn1</a>	84
<a href="#">parm::ee</a>	84
<a href="#">parm::expo</a>	84
<a href="#">parm::fcgd</a>	84
<a href="#">parm::HQDAV</a>	85
<a href="#">parm::layersplit</a>	85
<a href="#">parm::ndenit</a>	85
<a href="#">parm::qman</a>	85
<a href="#">parm::regres</a>	86
<a href="#">parm::rsedaa</a>	86
<a href="#">parm::tair</a>	86
<a href="#">parm::theta</a>	86
<a href="#">parm::vbl</a>	86



## Chapter 4

# File Index

### 4.1 File List

Here is a list of all documented files with brief descriptions:

<a href="#">allocate_parms.f90</a>	87
<a href="#">ascrv.f90</a>	87
<a href="#">aunif.f90</a>	88
<a href="#">caps.f90</a>	89
<a href="#">estimate_ksat.f90</a>	90
<a href="#">gcycl.f90</a>	90
<a href="#">getallo.f90</a>	91
<a href="#">h2omgt_init.f90</a>	91
<a href="#">headout.f90</a>	92
<a href="#">hruallo.f90</a>	92
<a href="#">hydroinit.f90</a>	92
<a href="#">impnd_init.f90</a>	93
<a href="#">jdt.f90</a>	93
<a href="#">lwqdef.f90</a>	94
<a href="#">main.f90</a>	94
<a href="#">modparm.f90</a>	95
<a href="#">openwth.f90</a>	162
<a href="#">qman.f90</a>	162
<a href="#">readatmodep.f90</a>	163
<a href="#">readbsn.f90</a>	164
<a href="#">readchm.f90</a>	164
<a href="#">readcnst.f90</a>	164
<a href="#">readfcst.f90</a>	165
<a href="#">readfert.f90</a>	165
<a href="#">readfig.f90</a>	165
<a href="#">readfile.f90</a>	166
<a href="#">readgw.f90</a>	166
<a href="#">readhru.f90</a>	166
<a href="#">readinpt.f90</a>	167
<a href="#">readlup.f90</a>	167
<a href="#">readlwq.f90</a>	167
<a href="#">readmgt.f90</a>	168
<a href="#">readmon.f90</a>	168
<a href="#">readops.f90</a>	168
<a href="#">readpest.f90</a>	169

readplant.f90	169
readpnd.f90	169
readres.f90	170
readrte.f90	170
readru.f90	170
readsdr.f90	171
readsepticbz.f90	171
readseptwq.f90	171
readsno.f90	172
readsol.f90	172
readsub.f90	173
readswq.f90	173
readtill.f90	173
readurban.f90	174
readwgn.f90	174
readwus.f90	174
readwwq.f90	175
readyr.f90	175
rteinit.f90	175
sim_inityr.f90	176
simulate.f90	176
soil_chem.f90	176
soil_phys.f90	177
std1.f90	177
std2.f90	178
std3.f90	178
storeinitial.f90	178
ttcoef.f90	179
xmon.f90	179
zero0.f90	180
zero1.f90	180
zero2.f90	180
zero_urban.f90	181
zeroini.f90	181

## Chapter 5

# Module Documentation

### 5.1 parm Module Reference

main module containing the global variables

#### Data Types

- interface [ascrv](#)
- interface [atri](#)
- interface [aunif](#)
- interface [dstn1](#)
- interface [ee](#)
- interface [expo](#)
- interface [fcgd](#)
- interface [HQDAV](#)
- interface [layersplit](#)
- interface [ndenit](#)
- interface [qman](#)
- interface [regres](#)
- interface [rsedaa](#)
- interface [tair](#)
- interface [theta](#)
- interface [vbl](#)

#### Variables

- integer, parameter [mvaro](#) = 33  
*max number of variables routed through the reach*
- integer, parameter [mhruo](#) = 79  
*maximum number of variables written to HRU output file (output.hru) (none)*
- integer, parameter [mrcho](#) = 62  
*maximum number of variables written to reach output file (.rch) (none)*
- integer, parameter [msub](#) = 24  
*maximum number of variables written to subbasin output file (output.sub) (none)*
- integer, parameter [mstdo](#) = 113

*max number of variables summarized in output.std*

- integer, parameter **motot** = 600
- character(len=80), parameter **prog** = "SWAT Sep 7 VER 2018/Rev 670"

*SWAT program header string (name and version)*

- character(len=13), dimension(**mhruo**), parameter **heds** = (/ "PRECIPmm", "SNOFALLmm", "SNOMELTmm", "IRRmm", "PETmm", "ETmm", "SW\_INITmm", "SW\_ENDmm", "PERCmm", "GW\_RCHGmm", "DA\_RCHGmm", "REVPmm", "SA\_IRRmm", "DA\_IRRmm", "SA\_STmm", "DA\_STmm", "SURQ\_GENmm", "SURQ\_CNTmm", "TLOSSmm", "LATQGENmm", "GW\_Qmm", "WYLDmm", "DAILYCN", "TMP\_AVdgC", "TMP\_MXdgC", "TMP\_MNdgc", "SOL\_TMPdgC", "SOLARMJ/m2", "SYLDt/ha", "USLEt/ha", "N\_APPkg/ha", "P\_APPkg/ha", "NAUTOk/ha", "PAUTOk/ha", "NGRZkg/ha", "PGRZkg/ha", "NCFRTkg/ha", "PCFRTkg/ha", "NRAINkg/ha", "NFIKkg/ha", "F-MNkg/ha", "A-MNkg/ha", "A-SNkg/ha", "F-MPkg/ha", "AO-LPkg/ha", "L-APkg/ha", "A-SPkg/ha", "DNITkg/ha", "NUPkg/ha", "PUPkg/ha", "ORGNkg/ha", "ORGPkg/ha", "SEDPkg/ha", "NSURQkg/ha", "NLATQkg/ha", "NO3Lkg/ha", "NO3GWkg/ha", "SOLPkg/ha", "P\_GWkg/ha", "W\_STRS", "TMP\_STRS", "N\_STRS", "P\_STRS", "BIOMt/ha", "LAI", "YLDt/ha", "BACTPct", "BACTLPct", "WTAB CLIm", "WTAB SOLm", "SNOmm", "CMUPkg/ha", "CMTOTkg/ha", "QTILEmm", "TNO3kg/ha", "LNO3kg/ha", "GW\_QDmm", "LATQCNTmm", "TVAPkg/ha"/)

*column headers for HRU output file*

- integer, dimension(**mhruo**), parameter **icols** = (/43,53,63,73,83,93,103,113,123,133,143,153,163,173,183,193,203,213,223,233,243,253,263,273,283,293,303,313,323,333,343,353,363,373,383,393,403,413,423,433,443,453,463,473,483,493,503,513,523,533,543,553,563,573,583,593,603,613,623,633,643,653,663,673,683,693,703,713,723,733,743,753,763,773,783,793,803,813,823,833,843,853,863,873,883,893,903,913,923,933,943,953,963,973,983,993,1003,1013,1023,1033,1043,1053,1063,1073,1083,1093,1103,1113,1123,1133,1143,1153,1163,1173,1183,1193,1203,1213,1223,1233,1243,1253,1263,1273,1283,1293,1303,1313,1323,1333,1343,1353,1363,1373,1383,1393,1403,1413,1423,1433,1443,1453,1463,1473,1483,1493,1503,1513,1523,1533,1543,1553,1563,1573,1583,1593,1603,1613,1623,1633,1643,1653,1663,1673,1683,1693,1703,1713,1723,1733,1743,1753,1763,1773,1783,1793,1803,1813,1823,1833,1843,1853,1863,1873,1883,1893,1903,1913,1923,1933,1943,1953,1963,1973,1983,1993,2003,2013,2023,2033,2043,2053,2063,2073,2083,2093,2103,2113,2123,2133,2143,2153,2163,2173,2183,2193,2203,2213,2223,2233,2243,2253,2263,2273,2283,2293,2303,2313,2323,2333,2343,2353,2363,2373,2383,2393,2403,2413,2423,2433,2443,2453,2463,2473,2483,2493,2503,2513,2523,2533,2543,2553,2563,2573,2583,2593,2603,2613,2623,2633,2643,2653,2663,2673,2683,2693,2703,2713,2723,2733,2743,2753,2763,2773,2783,2793,2803,2813,2823,2833,2843,2853,2863,2873,2883,2893,2903,2913,2923,2933,2943,2953,2963,2973,2983,2993,3003,3013,3023,3033,3043,3053,3063,3073,3083,3093,3103,3113,3123,3133,3143,3153,3163,3173,3183,3193,3203,3213,3223,3233,3243,3253,3263,3273,3283,3293,3303,3313,3323,3333,3343,3353,3363,3373,3383,3393,3403,3413,3423,3433,3443,3453,3463,3473,3483,3493,3503,3513,3523,3533,3543,3553,3563,3573,3583,3593,3603,3613,3623,3633,3643,3653,3663,3673,3683,3693,3703,3713,3723,3733,3743,3753,3763,3773,3783,3793,3803,3813,3823,3833,3843,3853,3863,3873,3883,3893,3903,3913,3923,3933,3943,3953,3963,3973,3983,3993,4003,4013,4023,4033,4043,4053,4063,4073,4083,4093,4103,4113,4123,4133,4143,4153,4163,4173,4183,4193,4203,4213,4223,4233,4243,4253,4263,4273,4283,4293,4303,4313,4323,4333,4343,4353,4363,4373,4383,4393,4403,4413,4423,4433,4443,4453,4463,4473,4483,4493,4503,4513,4523,4533,4543,4553,4563,4573,4583,4593,4603,4613,4623,4633,4643,4653,4663,4673,4683,4693,4703,4713,4723,4733,4743,4753,4763,4773,4783,4793,4803,4813,4823,4833,4843,4853,4863,4873,4883,4893,4903,4913,4923,4933,4943,4953,4963,4973,4983,4993,5003,5013,5023,5033,5043,5053,5063,5073,5083,5093,5103,5113,5123,5133,5143,5153,5163,5173,5183,5193,5203,5213,5223,5233,5243,5253,5263,5273,5283,5293,5303,5313,5323,5333,5343,5353,5363,5373,5383,5393,5403,5413,5423,5433,5443,5453,5463,5473,5483,5493,5503,5513,5523,5533,5543,5553,5563,5573,5583,5593,5603,5613,5623,5633,5643,5653,5663,5673,5683,5693,5703,5713,5723,5733,5743,5753,5763,5773,5783,5793,5803,5813,5823,5833,5843,5853,5863,5873,5883,5893,5903,5913,5923,5933,5943,5953,5963,5973,5983,5993,6003,6013,6023,6033,6043,6053,6063,6073,6083,6093,6103,6113,6123,6133,6143,6153,6163,6173,6183,6193,6203,6213,6223,6233,6243,6253,6263,6273,6283,6293,6303,6313,6323,6333,6343,6353,6363,6373,6383,6393,6403,6413,6423,6433,6443,6453,6463,6473,6483,6493,6503,6513,6523,6533,6543,6553,6563,6573,6583,6593,6603,6613,6623,6633,6643,6653,6663,6673,6683,6693,6703,6713,6723,6733,6743,6753,6763,6773,6783,6793,6803,6813,6823,6833,6843,6853,6863,6873,6883,6893,6903,6913,6923,6933,6943,6953,6963,6973,6983,6993,7003,7013,7023,7033,7043,7053,7063,7073,7083,7093,7103,7113,7123,7133,7143,7153,7163,7173,7183,7193,7203,7213,7223,7233,7243,7253,7263,7273,7283,7293,7303,7313,7323,7333,7343,7353,7363,7373,7383,7393,7403,7413,7423,7433,7443,7453,7463,7473,7483,7493,7503,7513,7523,7533,7543,7553,7563,7573,7583,7593,7603,7613,7623,7633,7643,7653,7663,7673,7683,7693,7703,7713,7723,7733,7743,7753,7763,7773,7783,7793,7803,7813,7823,7833,7843,7853,7863,7873,7883,7893,7903,7913,7923,7933,7943,7953,7963,7973,7983,7993,8003,8013,8023,8033,8043,8053,8063,8073,8083,8093,8103,8113,8123,8133,8143,8153,8163,8173,8183,8193,8203,8213,8223,8233,8243,8253,8263,8273,8283,8293,8303,8313,8323,8333,8343,8353,8363,8373,8383,8393,8403,8413,8423,8433,8443,8453,8463,8473,8483,8493,8503,8513,8523,8533,8543,8553,8563,8573,8583,8593,8603,8613,8623,8633,8643,8653,8663,8673,8683,8693,8703,8713,8723,8733,8743,8753,8763,8773,8783,8793,8803,8813,8823,8833,8843,8853,8863,8873,8883,8893,8903,8913,8923,8933,8943,8953,8963,8973,8983,8993,9003,9013,9023,9033,9043,9053,9063,9073,9083,9093,9103,9113,9123,9133,9143,9153,9163,9173,9183,9193,9203,9213,9223,9233,9243,9253,9263,9273,9283,9293,9303,9313,9323,9333,9343,9353,9363,9373,9383,9393,9403,9413,9423,9433,9443,9453,9463,9473,9483,9493,9503,9513,9523,9533,9543,9553,9563,9573,9583,9593,9603,9613,9623,9633,9643,9653,9663,9673,9683,9693,9703,9713,9723,9733,9743,9753,9763,9773,9783,9793,9803,9813,9823,9833,9843,9853,9863,9873,9883,9893,9903,9913,9923,9933,9943,9953,9963,9973,9983,9993,1000,1001,1002,1003,1004,1005,1006,1007,1008,1009,1010,1011,1012,1013,1014,1015,1016,1017,1018,1019,1020,1021,1022,1023,1024,1025,1026,1027,1028,1029,1030,1031,1032,1033,1034,1035,1036,1037,1038,1039,1040,1041,1042,1043,1044,1045,1046,1047,1048,1049,1050,1051,1052,1053,1054,1055,1056,1057,1058,1059,1060,1061,1062,1063,1064,1065,1066,1067,1068,1069,1070,1071,1072,1073,1074,1075,1076,1077,1078,1079,1080,1081,1082,1083,1084,1085,1086,1087,1088,1089,1090,1091,1092,1093,1094,1095,1096,1097,1098,1099,1100,1101,1102,1103,1104,1105,1106,1107,1108,1109,1110,1111,1112,1113,1114,1115,1116,1117,1118,1119,1120,1121,1122,1123,1124,1125,1126,1127,1128,1129,1130,1131,1132,1133,1134,1135,1136,1137,1138,1139,1140,1141,1142,1143,1144,1145,1146,1147,1148,1149,1150,1151,1152,1153,1154,1155,1156,1157,1158,1159,1160,1161,1162,1163,1164,1165,1166,1167,1168,1169,1170,1171,1172,1173,1174,1175,1176,1177,1178,1179,1180,1181,1182,1183,1184,1185,1186,1187,1188,1189,1190,1191,1192,1193,1194,1195,1196,1197,1198,1199,1200,1201,1202,1203,1204,1205,1206,1207,1208,1209,1210,1211,1212,1213,1214,1215,1216,1217,1218,1219,1220,1221,1222,1223,1224,1225,1226,1227,1228,1229,1230,1231,1232,1233,1234,1235,1236,1237,1238,1239,1240,1241,1242,1243,1244,1245,1246,1247,1248,1249,1250,1251,1252,1253,1254,1255,1256,1257,1258,1259,1260,1261,1262,1263,1264,1265,1266,1267,1268,1269,1270,1271,1272,1273,1274,1275,1276,1277,1278,1279,1280,1281,1282,1283,1284,1285,1286,1287,1288,1289,1290,1291,1292,1293,1294,1295,1296,1297,1298,1299,1300,1301,1302,1303,1304,1305,1306,1307,1308,1309,1310,1311,1312,1313,1314,1315,1316,1317,1318,1319,1320,1321,1322,1323,1324,1325,1326,1327,1328,1329,1330,1331,1332,1333,1334,1335,1336,1337,1338,1339,1340,1341,1342,1343,1344,1345,1346,1347,1348,1349,1350,1351,1352,1353,1354,1355,1356,1357,1358,1359,1360,1361,1362,1363,1364,1365,1366,1367,1368,1369,1370,1371,1372,1373,1374,1375,1376,1377,1378,1379,1380,1381,1382,1383,1384,1385,1386,1387,1388,1389,1390,1391,1392,1393,1394,1395,1396,1397,1398,1399,1400,1401,1402,1403,1404,1405,1406,1407,1408,1409,1410,1411,1412,1413,1414,1415,1416,1417,1418,1419,1420,1421,1422,1423,1424,1425,1426,1427,1428,1429,1430,1431,1432,1433,1434,1435,1436,1437,1438,1439,1440,1441,1442,1443,1444,1445,1446,1447,1448,1449,1450,1451,1452,1453,1454,1455,1456,1457,1458,1459,1460,1461,1462,1463,1464,1465,1466,1467,1468,1469,1470,1471,1472,1473,1474,1475,1476,1477,1478,1479,1480,1481,1482,1483,1484,1485,1486,1487,1488,1489,1490,1491,1492,1493,1494,1495,1496,1497,1498,1499,1500,1501,1502,1503,1504,1505,1506,1507,1508,1509,1510,1511,1512,1513,1514,1515,1516,1517,1518,1519,1520,1521,1522,1523,1524,1525,1526,1527,1528,1529,1530,1531,1532,1533,1534,1535,1536,1537,1538,1539,1540,1541,1542,1543,1544,1545,1546,1547,1548,1549,1550,1551,1552,1553,1554,1555,1556,1557,1558,1559,1560,1561,1562,1563,1564,1565,1566,1567,1568,1569,1570,1571,1572,1573,1574,1575,1576,1577,1578,1579,1580,1581,1582,1583,1584,1585,1586,1587,1588,1589,1590,1591,1592,1593,1594,1595,1596,1597,1598,1599,1600,1601,1602,1603,1604,1605,1606,1607,1608,1609,1610,1611,1612,1613,1614,1615,1616,1617,1618,1619,1620,1621,1622,1623,1624,1625,1626,1627,1628,1629,1630,1631,1632,1633,1634,1635,1636,1637,1638,1639,1640,1641,1642,1643,1644,1645,1646,1647,1648,1649,1650,1651,1652,1653,1654,1655,1656,1657,1658,1659,1660,1661,1662,1663,1664,1665,1666,1667,1668,1669,1670,1671,1672,1673,1674,1675,1676,1677,1678,1679,1680,1681,1682,1683,1684,1685,1686,1687,1688,1689,1690,1691,1692,1693,1694,1695,1696,1697,1698,1699,1700,1701,1702,1703,1704,1705,1706,1707,1708,1709,1710,1711,1712,1713,1714,1715,1716,1717,1718,1719,1720,1721,1722,1723,1724,1725,1726,1727,1728,1729,1730,1731,1732,1733,1734,1735,1736,1737,1738,1739,1740,1741,1742,1743,1744,1745,1746,1747,1748,1749,1750,1751,1752,1753,1754,1755,1756,1757,1758,1759,1760,1761,1762,1763,1764,1765,1766,1767,1768,1769,1770,1771,1772,1773,1774,1775,1776,1777,1778,1779,1780,1781,1782,1783,1784,1785,1786,1787,1788,1789,1790,1791,1792,1793,1794,1795,1796,1797,1798,1799,1800,1801,1802,1803,1804,1805,1806,1807,1808,1809,1810,1811,1812,1813,1814,1815,1816,1817,1818,1819,1820,1821,1822,1823,1824,1825,1826,1827,1828,1829,1830,1831,1832,1833,1834,1835,1836,1837,1838,1839,1840,1841,1842,1843,1844,1845,1846,1847,1848,1849,1850,1851,1852,1853,1854,1855,1856,1857,1858,1859,1860,1861,1862,1863,1864,1865,1866,1867,1868,1869,1870,1871,1872,1873,1874,1875,1876,1877,1878,1879,1880,1881,1882,1883,1884,1885,1886,1887,1888,1889,1890,1891,1892,1893,1894,1895,1896,1897,1898,1899,1900,1901,1902,1903,1904,1905,1906,1907,1908,1909,1910,1911,1912,1913,1914,1915,1916,1917,1918,1919,1920,1921,1922,1923,1924,1925,1926,1927,1928,1929,1930,1931,1932,1933,1934,1935,1936,1937,1938,1939,1940,1941,1942,1943,1944,1945,1946,1947,1948,1949,1950,1951,1952,1953,1954,1955,1956,1957,1958,1959,1960,1961,1962,1963,1964,1965,1966,1967,1968,1969,1970,1971,1972,1973,1974,1975,1976,1977,1978,1979,1980,1981,1982,1983,1984,1985,1986,1987,1988,1989,1990,1991,1992,1993,1994,1995,1996,1997,1998,1999,2000,2001,2002,2003,2004,2005,2006,2007,2008,2009,2010,2011,2012,2013,2014,2015,2016,2017,2018,2019,2020,2021,2022,2023,2024,2025,2026,2027,2028,2029,2030,2031,2032,2033,2034,2035,2036,2037,2038,2039,2040,2041,2042,2043,2044,2045,2046,2047,2048,2049,2050,2051,2052,2053,2054,2055,2056,2057,2058,2059,2060,2061,2062,2063,2064,2065,2066,2067,2068,2069,2070,2071,2072,2073,2074,2075,2076,2077,2078,2079,2080,2081,2082,2083,2084,2085,2086,2087,2088,2089,2090,2091,2092,2093,2094,2095,2096,2097,2098,2099,2100,2101,2102,2103,2104,2105,2106,2107,2108,2109,2110,2111,2112,2113,2114,2115,2116,2117,2118,2119,2120,2121,2122,2123,2124,2125,2126,2127,2128,2129,2130,2131,2132,2133,2134,2135,2136,2137,2138,2139,2140,2141,2142,2143,2144,2145,2146,2147,2148,2149,2150,2151,2152,2153,2154,2155,2156,2157,2158,2159,2160,2161,2162,2163,2164,2165,2166,2167,2168,2169,2170,2171,2172,2173,2174,2175,2176,2177,2178,2179,2180,2181,2182,2183,2184,2185,2186,2187,2188,2189,2190,2191,2192,2193,2194,2195,2196,2197,2198,2199,2200,2201,2202,2203,2204,2205,2206,2207,2208,2209,2210,2211,2212,2213,2214,2215,2216,2217,2218,2219,2220,2221,2222,2223,2224,2225,2226,2227,2228,2229,2230,2231,2232,2233,2234,2235,2236,2237,2238,2239,2240,2241,2242,2243,2244,2245,2246,2247,2248,2249,2250,2251,2252,2253,2254,2255,2256,2257,2258,2259,2260,2261,2262,2263,2264,2265,2266,2267,2268,2269,2270,2271,2272,2273,2274,2275,2276,2277,2278,2279,2280,2281,2282,2283,2284,2285,2286,2287,2288,2289,2290,2291,2292,2293,2294,2295,2296,2297,2298,2299,2300,2301,2302,2303,2304,2305,2306,2307,2308,2309

- column headers for HRU impoundment output file*
- integer **i**
  - forecast region, subbasin, HRU, reach, reservoir or file number (none)*
- integer **icalen**
- real \*8 **prf\_bsn**
  - Basinwide peak rate adjustment factor for sediment routing in the channel. Allows impact of peak flow rate on sediment routing and channel reshaping to be taken into account.*
- real \*8 **co2\_x2**
- real \*8 **co2\_x**
- real \*8, dimension(:), allocatable **alph\_e**
- real \*8, dimension(:), allocatable **cdn**
  - denitrification exponential rate coefficient*
- real \*8, dimension(:), allocatable **nperco**
  - nitrate percolation coefficient (0-1)*
  - 0:concentration of nitrate in surface runoff is zero*
  - 1:percolate has same concentration of nitrate as surface runoff*
- real \*8, dimension(:), allocatable **surlag**
  - Surface runoff lag time. This parameter is needed in subbasins where the time of concentration is greater than 1 day. SURLAG is used to create a "storage" for surface runoff to allow the runoff to take longer than 1 day to reach the subbasin outlet (days)*
- real \*8, dimension(:), allocatable **co\_p**
- real \*8, dimension(:), allocatable **cmn**
  - rate factor for humus mineralization on active organic N*
- real \*8, dimension(:), allocatable **phoskd**
  - Phosphorus soil partitioning coefficient. Ratio of soluble phosphorus in surface layer to soluble phosphorus in runoff.*
- real \*8, dimension(:), allocatable **psp**
  - Phosphorus availability index. The fraction of fertilizer P remaining in labile pool after initial rapid phase of P sorption (none)*
- real \*8, dimension(:), allocatable **sdnco**
  - denitrification threshold: fraction of field capacity triggering denitrification*
- real \*8 **r2adj\_bsn**
  - basinwide retention parameter adjustment factor (greater than 1)*
- real \*8 **pst\_kg**
  - amount of pesticide applied to HRU (kg/ha)*
- real \*8 **yield**
- real \*8 **burn\_frlb**
- real \*8 **yieldgrn**
- real \*8 **yieldbms**
- real \*8 **yieldtbr**
- real \*8 **yieldn**
- real \*8 **yieldp**
- real \*8 **hi\_bms**
- real \*8 **hi\_rsd**
- real \*8 **yieldrsd**
- real \*8, dimension(:), allocatable **I\_k1**
- real \*8, dimension(:), allocatable **I\_k2**
- real \*8, dimension(:), allocatable **I\_lambda**
- real \*8, dimension(:), allocatable **I\_beta**
- real \*8, dimension(:), allocatable **I\_gama**
- real \*8, dimension(:), allocatable **I\_harea**
- real \*8, dimension(:), allocatable **I\_vleng**
- real \*8, dimension(:), allocatable **I\_vslope**
- real \*8, dimension(:), allocatable **I\_ktc**
- real \*8, dimension(:), allocatable **biofilm\_mumax**

- real \*8, dimension(:), allocatable **biofilm\_kinv**
- real \*8, dimension(:), allocatable **biofilm\_klw**
- real \*8, dimension(:), allocatable **biofilm\_kla**
- real \*8, dimension(:), allocatable **biofilm\_cdet**
- real \*8, dimension(:), allocatable **biofilm\_bm**
- real \*8, dimension(:,:), allocatable **hru\_rufr**
- real \*8, dimension(:,:), allocatable **daru\_km**
- real \*8, dimension(:,:), allocatable **ru\_k**
- real \*8, dimension(:,:), allocatable **ru\_c**
- real \*8, dimension(:,:), allocatable **ru\_eiq**
- real \*8, dimension(:,:), allocatable **ru\_ovsl**
- real \*8, dimension(:,:), allocatable **ru\_a**
- real \*8, dimension(:,:), allocatable **ru\_ovs**
- real \*8, dimension(:,:), allocatable **ru\_ktc**
- real \*8, dimension(:), allocatable **gwq\_ru**
- real \*8, dimension(:), allocatable **qdayout**
- integer, dimension(:), allocatable **ils2**
- integer, dimension(:), allocatable **ils2flag**
- integer **ipest**  
*pesticide identification number from pest.dat (none)*
- integer **iru**
- integer **mru**
- integer **irch**
- integer **isub**
- integer **mhyd\_bsn**
- integer **ils\_nofig**
- integer **mhru1**
- integer, dimension(:), allocatable **mhyd1**
- integer, dimension(:), allocatable **irtun**
- real \*8 **wshd\_sepno3**
- real \*8 **wshd\_sepnh3**
- real \*8 **wshd\_seporgn**
- real \*8 **wshd\_sepfon**
- real \*8 **wshd\_seporgp**
- real \*8 **wshd\_sepfop**
- real \*8 **wshd\_sepsolp**
- real \*8 **wshd\_sepbod**
- real \*8 **wshd\_sepmm**
- integer, dimension(:), allocatable **isep\_hru**
- real \*8 **fixco**  
*nitrogen fixation coefficient*
- real \*8 **nfixmx**  
*maximum daily n-fixation (kg/ha)*
- real \*8 **res\_stlr\_co**  
*reservoir sediment settling coefficient*
- real \*8 **rsd\_covco**  
*residue cover factor for computing frac of cover*
- real \*8 **vcrit**  
*critical velocity*
- real \*8 **wshd\_snob**  
*average amount of water stored in snow at the beginning of the simulation for the entire watershed (mm H2O)*
- real \*8 **wshd\_sw**  
*average amount of water stored in soil for the entire watershed (mm H2O)*



- real \*8 **wshd\_pndfr**  
*fraction of watershed area which drains into ponds (none)*
- real \*8 **wshd\_pndsed**  
*total amount of suspended sediment in ponds in the watershed (metric tons)*
- real \*8 **wshd\_pndv**  
*total volume of water in ponds in the watershed ( $m^3$ )*
- real \*8 **percop**  
*pesticide percolation coefficient (0-1)*  
*0: concentration of pesticide in surface runoff is zero*  
*1: percolate has same concentration of pesticide as surface runoff*
- real \*8 **wshd\_resfr**  
*fraction of watershed area that drains into reservoirs (none)*
- real \*8 **wshd\_pndha**  
*watershed area in hectares which drains into ponds (ha)*
- real \*8 **wshd\_resha**  
*watershed area in hectares which drains into reservoirs (ha)*
- real \*8 **wshd\_wetfr**  
*fraction of watershed area which drains into wetlands (none)*
- real \*8 **wshd\_fminp**
- real \*8 **wshd\_ftotn**
- real \*8 **wshd\_fnh3**
- real \*8 **wshd\_fno3**
- real \*8 **wshd\_forgn**
- real \*8 **wshd\_forgp**
- real \*8 **wshd\_ftotp**
- real \*8 **wshd\_yldn**
- real \*8 **wshd\_yldp**
- real \*8 **wshd\_fixn**
- real \*8 **wshd\_pup**
- real \*8 **wshd\_wstrs**
- real \*8 **wshd\_nstrs**
- real \*8 **wshd\_pstrs**
- real \*8 **wshd\_tstrs**
- real \*8 **wshd\_astrs**
- real \*8 **ffcb**  
*initial soil water content expressed as a fraction of field capacity*
- real \*8 **wshd\_hmn**
- real \*8 **wshd\_rwn**
- real \*8 **wshd\_hmp**
- real \*8 **wshd\_rmn**
- real \*8 **wshd\_dnit**
- real \*8 **wdpq**  
*die-off factor for persistent bacteria in soil solution (1/day)*
- real \*8 **wshd\_rmp**
- real \*8 **wshd\_voln**
- real \*8 **wshd\_nitn**
- real \*8 **wshd\_pas**
- real \*8 **wshd\_pal**
- real \*8 **wof\_p**  
*wash off fraction for persistent bacteria on foliage during a rainfall event*
- real \*8 **wshd\_plch**
- real \*8 **wshd\_raino3**
- real \*8 **ressedc**

- real \*8 **basno3f**
- real \*8 **basorgnf**
- real \*8 **wshd\_pinlet**
- real \*8 **wshd\_ptile**
- real \*8 **sftmp**  
Snowfall temperature (deg C)
- real \*8 **smfmn**  
Minimum melt rate for snow during year (Dec. 21) where deg C refers to the air temperature. (mm/deg C/day)
- real \*8 **smfmx**  
Maximum melt rate for snow during year (June 21) where deg C refers to the air temperature. SMFMX and SM<sub>FMN</sub> allow the rate of snow melt to vary through the year. These parameters are accounting for the impact of soil temperature on snow melt. (mm/deg C/day)
- real \*8 **smtmp**  
Snow melt base temperature. Mean air temperature at which snow melt will occur. (deg C)
- real \*8 **wgppq**  
growth factor for persistent bacteria in soil solution (1/day)
- real \*8 **basminpf**
- real \*8 **basorgpf**
- real \*8 **wdlpq**  
die-off factor for less persistent bacteria in soil solution (1/day)
- real \*8 **wshd\_ressed**  
total amount of suspended sediment in reservoirs in the watershed (metric tons)
- real \*8 **wshd\_resv**  
total volume of water in all reservoirs in the watershed (m<sup>3</sup>)
- real \*8 **basminpi**  
average amount of phosphorus initially in the mineral P pool in watershed soil (kg P/ha)
- real \*8 **basno3i**  
average amount of nitrogen initially in the nitrate pool in watershed soil (kg N/ha)
- real \*8 **basorgni**  
average amount of nitrogen initially in the organic N pool in watershed soil (kg N/ha)
- real \*8 **wdps**  
die-off factor for persistent bacteria adsorbed to soil particles (1/day)
- real \*8 **wglpq**  
growth factor for less persistent bacteria in soil solution (1/day)
- real \*8 **basorgpi**  
average amount of phosphorus initially in the organic P pool in watershed soil (kg P/ha)
- real \*8 **peakr**
- real \*8 **pndsedin**
- real \*8 **sw\_excess**
- real \*8 **albday**
- real \*8 **timp**  
Snow pack temperature lag factor (0-1)  
1 = no lag (snow pack temp=current day air temp) as the lag factor goes to zero, the snow pack's temperature will be less influenced by the current day's air temperature.
- real \*8 **wtabelo**
- real \*8 **tilep**
- real \*8 **wt\_shall**
- real \*8 **sq\_rto**
- real \*8 **tloss**
- real \*8 **inflpcp**
- real \*8 **snomlt**
- real \*8 **snofall**
- real \*8 **fixn**

- real \*8 **qtile**
- real \*8 **crk**
- real \*8 **latlyr**
- real \*8 **pndloss**
- real \*8 **wetloss**
- real \*8 **potloss**
- real \*8 **lpndloss**
- real \*8 **lwetloss**
- real \*8 **sedrch**
- real \*8 **fertn**
- real \*8 **sol\_rd**
- real \*8 **cfertn**
- real \*8 **cfertp**
- real \*8 **sepday**
- real \*8 **bioday**
- real \*8 **sepcrk**
- real \*8 **sepcrktot**
- real \*8 **fertno3**
- real \*8 **fertnh3**
- real \*8 **fertorgn**
- real \*8 **fertsolp**
- real \*8 **fertorgp**
- real \*8 **wgps**  
*growth factor for persistent bacteria adsorbed to soil particles (1/day)*
- real \*8 **fertp**
- real \*8 **grazn**
- real \*8 **grazp**
- real \*8 **soxy**
- real \*8 **qdfr**
- real \*8 **sdti**
- real \*8 **rtwtr**
- real \*8 **ressa**
- real \*8 **wdlps**  
*die-off factor for less persistent bacteria absorbed to soil particles (1/day)*
- real \*8 **wglps**  
*growth factor for less persistent bacteria adsorbed to soil particles (1/day)*
- real \*8 **da\_km**  
*area of the watershed in square kilometers (km<sup>2</sup>)*
- real \*8 **rttime**
- real \*8 **rchdep**
- real \*8 **rtevp**
- real \*8 **rttlc**
- real \*8 **resflwi**
- real \*8 **wdprch**  
*die-off factor for persistent bacteria in streams (1/day)*
- real \*8 **resflwo**
- real \*8 **respcp**
- real \*8 **resev**
- real \*8 **ressep**
- real \*8 **ressedi**
- real \*8 **ressedo**
- real \*8 **dtot**
- real \*8 **pperco\_bsn**  
*phosphorus percolation coefficient. Ratio of soluble phosphorus in surface to soluble phosphorus in percolate*

- real \*8 **nperco\_bsn**  
*basin nitrate percolation coefficient (0-1)*  
*0:concentration of nitrate in surface runoff is zero*  
*1:percolate has same concentration of nitrate as surface runoff*
- real \*8 **rsdco**  
*residue decomposition coefficient. The fraction of residue which will decompose in a day assuming optimal moisture, temperature, C:N ratio, and C:P ratio*
- real \*8 **phoskd\_bsn**
- real \*8 **voltot**
- real \*8 **msk\_x**  
*weighting factor controlling relative importance of inflow rate and outflow rate in determining storage on reach*
- real \*8 **volcrmin**
- real \*8 **bactkdq**  
*bacteria soil partitioning coefficient. Ratio of solution bacteria in surface layer to solution bacteria in runoff soluble and sorbed phase in surface runoff.*
- real \*8 **wdpf**  
*die-off factor for persistent bacteria on foliage (1/day)*
- real \*8 **uno3d**
- real \*8 **canev**
- real \*8 **usle**
- real \*8 **rcn**
- real \*8 **surlag\_bsn**
- real \*8 **precipday**
- real \*8 **thbact**  
*temperature adjustment factor for bacteria die-off/growth*
- real \*8 **wlpq20**  
*overall rate change for less persistent bacteria in soil solution (1/day)*
- real \*8 **wlps20**  
*overall rate change for less persistent bacteria adsorbed to soil particles (1/day)*
- real \*8 **wpq20**  
*overall rate change for persistent bacteria in soil solution (1/day)*
- real \*8 **wps20**  
*overall rate change for persistent bacteria adsorbed to soil particles (1/day)*
- real \*8 **bactrop**
- real \*8 **bactsedp**
- real \*8 **wgpf**  
*growth factor for persistent bacteria on foliage (1/day)*
- real \*8 **bactlchp**
- real \*8 **bactlchlp**
- real \*8 **enratio**
- real \*8 **wetpcp**
- real \*8 **pndpcp**
- real \*8 **wetsep**
- real \*8 **pndsep**
- real \*8 **wetev**
- real \*8 **pndev**
- real \*8 **pndsedo**
- real \*8 **wetsedo**
- real \*8 **pndfliwi**
- real \*8 **wetfliwi**
- real \*8 **da\_ha**  
*drainage area of watershed in hectares (ha)*
- real \*8 **pndfliwo**

- real \*8 **wetflwo**
- real \*8 **wetsedi**
- real \*8 **vpd**
- real \*8 **evlai**  
*leaf area index at which no evaporation occurs. This variable is used in ponded HRUs where evaporation from the water surface is restricted by the plant canopy cover. Evaporation from the water surface equals potential ET when LAI = 0 and decreased linearly to 0 when LAI = EVLAI*
- real \*8 **evrch**  
*Reach evaporation adjustment factor. Evaporation from the reach is multiplied by EVRCH. This variable was created to limit the evaporation predicted in arid regions.*
- real \*8 **wdlpf**  
*die-off factor for less persistent bacteria on foliage (1/day)*
- real \*8 **bactrolp**
- real \*8 **bactsedlp**
- real \*8 **pet\_day**
- real \*8 **ep\_day**
- real \*8 **adj\_pkr**  
*peak rate adjustment factor in the subbasin. Used in the MUSLE equation to account for impact of peak flow on erosion (none)*
- real \*8 **n\_updis**  
*nitrogen uptake distribution parameter. This parameter controls the amount of nitrogen removed from the different soil layer layers by the plant. In particular, this parameter allows the amount of nitrogen removed from the surface layer via plant uptake to be controlled. While the relationship between UBN and N removed from the surface layer is affected by the depth of the soil profile, in general, as UBN increases the amount of N removed from the surface layer relative to the amount removed from the entire profile increases*
- real \*8 **nactfr**  
*nitrogen active pool fraction. The fraction of organic nitrogen in the active pool (none)*
- real \*8 **p\_updis**  
*phosphorus uptake distribution parameter This parameter controls the amount of phosphorus removed from the different soil layers by the plant. In particular, this parameter allows the amount of phosphorus removed from the surface layer via plant uptake to be controlled. While the relationship between UBP and P uptake from the surface layer is affected by the depth of the soil profile, in general, as UBP increases the amount of P removed from the surface layer relative to the amount removed from the entire profile increases*
- real \*8 **snoev**
- real \*8 **sno3up**
- real \*8 **reactw**
- real \*8 **sdiégropq**
- real \*8 **sdiégrolpq**
- real \*8 **sdiégrops**
- real \*8 **sdiégrolps**
- real \*8 **es\_day**
- real \*8 **wof\_lp**  
*wash off fraction for less persistent bacteria on foliage during a rainfall event*
- real \*8 **sbactrop**
- real \*8 **sbactrolp**
- real \*8 **sbactsedp**
- real \*8 **sbactsedlp**
- real \*8 **ep\_max**
- real \*8 **sbactlchp**
- real \*8 **sbactlchlp**
- real \*8 **psp\_bsn**
- real \*8 **rchwtr**
- real \*8 **resuspst**
- real \*8 **setlpst**
- real \*8 **bsprev**

- real \*8 **bssprev**
- real \*8 **spadyo**
- real \*8 **spadyev**
- real \*8 **spadysp**
- real \*8 **spadyrfv**
- real \*8 **spadyosp**
- real \*8 **qday**
- real \*8 **usle\_ei**
- real \*8 **al5**
- real \*8 **pndsedc**
- real \*8 **no3pcp**
- real \*8 **rcharea**
- real \*8 **volatpst**
- real \*8 **ubw**

*water uptake distribution parameter. This parameter controls the amount of water removed from the different soil layers by the plant. In particular, this parameter allows the amount of water removed from the surface layer via plant uptake to be controlled. While the relationship between UBW and H2O removed from the surface layer is affected by the depth of the soil profile, in general, as UBW increases the amount of water removed from the surface layer relative to the amount removed from the entire profile increases*

- real \*8 **uobn**

*nitrogen uptake normalization parameter. This variable normalizes the nitrogen uptake so that the model can easily verify that uptake from the different soil layers sums to 1.0*

- real \*8 **uobp**

*phosphorus uptake normalization parameter. This variable normalizes the phosphorus uptake so that the model can easily verify that uptake from the different soil layers sums to 1.0*

- real \*8 **uobw**

*water uptake normalization parameter. This variable normalizes the water uptake so that the model can easily verify that uptake from the different soil layers sums to 1.0*

- real \*8 **wglpf**

*growth factor for less persistent bacteria on foliage (1/day)*

- real \*8 **wetsedc**
- real \*8 **respesti**
- real \*8 **rcor**

*correction coefficient for generated rainfall to ensure that the annual means for generated and observed values are comparable (needed only if IDIST=1)*

- real \*8 **rexp**

*value of exponent for mixed exponential rainfall distribution (needed only if IDIST=1)*

- real \*8 **snocov1**

*1st shape parameter for snow cover equation. This parameter is determined by solving the equation for 50% snow cover*

- real \*8 **snocov2**

*2nd shape parameter for snow cover equation. This parameter is determined by solving the equation for 95% snow cover*

- real \*8 **snocovmx**

*Minimum snow water content that corresponds to 100% snow cover. If the snow water content is less than SNOC<sub>←</sub>OVMX, then a certain percentage of the ground will be bare (mm H2O)*

- real \*8 **lyrtile**
- real \*8 **lyrtilex**
- real \*8 **sno50cov**

*Fraction of SNOCOV<sub>←</sub>MX that corresponds to 50% snow cover. SWAT assumes a nonlinear relationship between snow water and snow cover.*

- real \*8 **ai0**

*ratio of chlorophyll-a to algal biomass (ug chla/mg alg)*

- real \*8 **ai1**

- fraction of algal biomass that is nitrogen (mg N/mg alg)*
- real \*8 **ai2**
- fraction of algal biomass that is phosphorus (mg P/mg alg)*
- real \*8 **ai3**
- the rate of oxygen production per unit of algal photosynthesis (mg O2/mg alg)*
- real \*8 **ai4**
- the rate of oxygen uptake per unit of algae respiration (mg O2/mg alg)*
- real \*8 **ai5**
- the rate of oxygen uptake per unit of NH3 nitrogen oxidation (mg O2/mg N)*
- real \*8 **ai6**
- the rate of oxygen uptake per unit of NO2 nitrogen oxidation (mg O2/mg N)*
- real \*8 **rhoq**
- algal respiration rate (1/day or 1/hr)*
- real \*8 **tfact**
- fraction of solar radiation computed in the temperature heat balance that is photosynthetically active*
- real \*8 **k\_l**
- half-saturation coefficient for light (MJ/(m2\*hr))*
- real \*8 **k\_n**
- Michaelis-menton half-saturation constant for nitrogen (mg N/L)*
- real \*8 **k\_p**
- Michaelis-menton half saturation constant for phosphorus (mg P/L)*
- real \*8 **lambda0**
- non-algal portion of the light extinction coefficient (1/m)*
- real \*8 **lambda1**
- linear algal self-shading coefficient (1/(m\*ug chla/L))*
- real \*8 **lambda2**
- nonlinear algal self-shading coefficient ((1/m)(ug chla/L)\*\*(-2/3))*
- real \*8 **mumax**
- maximum specific algal growth rate (1/day or 1/hr)*
- real \*8 **p\_n**
- algal preference factor for ammonia*
- real \*8 **rnum1**
- real \*8 **autop**
- real \*8 **auton**
- real \*8 **etday**
- real \*8 **hmntl**
- real \*8 **rwntl**
- real \*8 **hmptl**
- real \*8 **rmn2tl**
- real \*8 **rmptl**
- real \*8 **wdntl**
- real \*8 **cmn\_bsn**
- real \*8 **rmp1tl**
- real \*8 **roctl**
- real \*8 **gwseep**
- real \*8 **revapday**
- real \*8 **reswtr**
- real \*8 **wdlprch**
- die-off factor for less persistent bacteria in streams (1/day)*
- real \*8 **wdpres**
- die-off factor for persistent bacteria in reservoirs (1/day)*
- real \*8 **bury**

- real \*8 **difus**
- real \*8 **reactb**
- real \*8 **solpesto**
- real \*8 **petmeas**
- real \*8 **wdlpres**  
*die-off factor for less persistent bacteria in reservoirs (1/day)*
- real \*8 **sorpesto**
- real \*8 **spcon\_bsn**
- real \*8 **spexp\_bsn**
- real \*8 **solpesti**
- real \*8 **sorpesti**
- real \*8 **msk\_co1**  
*calibration coefficient to control impact of the storage time constant for the reach at bankfull depth (phi(10,:)) upon the storage time constant for the reach used in the Muskingum flow method*
- real \*8 **msk\_co2**  
*calibration coefficient to control impact of the storage time constant for the reach at 0.1 bankfull depth (phi(13,:)) upon the storage time constant for the reach used in the Muskingum flow method*
- real \*8 **snoprev**
- real \*8 **swprev**
- real \*8 **shallstp**
- real \*8 **deepstp**
- real \*8 **ressolpo**
- real \*8 **resorgno**
- real \*8 **resorgpo**
- real \*8 **resno3o**
- real \*8 **reschlao**
- real \*8 **resno2o**
- real \*8 **resnh3o**
- real \*8 **qdbank**
- real \*8 **potpcpmm**
- real \*8 **potevmm**
- real \*8 **potsepmm**
- real \*8 **potflwo**
- real \*8 **bactminlp**  
*Threshold detection level for less persistent bacteria. When bacteria levels drop to this amount the model considers bacteria in the soil to be insignificant and sets the levels to zero (cfu/m<sup>2</sup>)*
- real \*8 **bactminp**  
*Threshold detection level for persistent bacteria. When bacteria levels drop to this amount the model considers bacteria in the soil to be insignificant and sets the levels to zero (cfu/m<sup>2</sup>)*
- real \*8 **trnsrch**  
*fraction of transmission losses from main channel that enter deep aquifer*
- real \*8 **wp20p\_plt**  
*overall rate change for persistent bacteria on foliage (1/day)*
- real \*8 **potsedo**
- real \*8 **pest\_sol**
- real \*8 **bact\_swf**  
*fraction of manure containing active colony forming units (cfu)*
- real \*8 **bactmx**  
*bacteria percolation coefficient. Ratio of solution bacteria in surface layer to solution bacteria in percolate*
- real \*8 **cncoef**  
*plant ET curve number coefficient*
- real \*8 **wp20lp\_plt**  
*overall rate change for less persistent bacteria on foliage (1/day)*



- real \*8 **cdn\_bsn**
- real \*8 **sdnco\_bsn**
- real \*8 **bactmin**
- real \*8 **cn\_froz**  
*drainge coefficient (mm day -1)*
- real \*8 **dorm\_hr**  
*time threshold used to define dormant (hours)*
- real \*8 **smxco**  
*adjustment factor for max curve number s factor (0-1)*
- real \*8 **tb\_adj**  
*adjustment factor for subdaily unit hydrograph basetime*
- real \*8 **chla\_subco**  
*regional adjustment on sub chla\_a loading (fraction)*
- real \*8 **depimp\_bsn**  
*depth to impervious layer. Used to model perched water tables in all HRUs in watershed (mm)*
- real \*8 **ddrain\_bsn**  
*depth to the sub-surface drain (mm)*
- real \*8 **tdrain\_bsn**  
*time to drain soil to field capacity (hours)*
- real \*8 **gdrain\_bsn**
- real \*8 **rch\_san**
- real \*8 **rch\_sil**
- real \*8 **rch\_cla**
- real \*8 **rch\_sag**
- real \*8 **rch\_lag**
- real \*8 **rch\_gra**
- real \*8 **hlife\_ngw\_bsn**  
*Half-life of nitrogen in groundwater? (days)*
- real \*8 **ch\_opco\_bsn**
- real \*8 **ch\_onco\_bsn**
- real \*8 **decr\_min**  
*Minimum daily residue decay.*
- real \*8 **rcn\_sub\_bsn**  
*Concentration of nitrogen in the rainfall (mg/kg)*
- real \*8 **bc1\_bsn**
- real \*8 **bc2\_bsn**
- real \*8 **bc3\_bsn**
- real \*8 **bc4\_bsn**
- real \*8 **anion\_excl\_bsn**
- real \*8, dimension(:), allocatable **wat\_tbl**
- real \*8, dimension(:), allocatable **sol\_swpwt**
- real \*8, dimension(:,:), allocatable **vwt**
- real \*8 **re\_bsn**  
*Effective radius of drains (range 3.0 - 40.0) (mm)*
- real \*8 **sdrain\_bsn**  
*Distance bewtween two drain or tile tubes (range 7600.0 - 30000.0) (mm)*
- real \*8 **sstmaxd\_bsn**
- real \*8 **drain\_co\_bsn**  
*Drainage coeffcient (range 10.0 - 51.0) (mm-day-1)*
- real \*8 **latksatf\_bsn**  
*Multiplication factor to determine lateral ksat from SWAT ksat input value for HRU (range 0.01 - 4.0)*
- real \*8 **pc\_bsn**

- Pump capacity (def val = 1.042 mm h-1 or 25 mm day-1) (mm h-1)*
- integer **i\_subhw**
- integer **imgt**
- integer **idlast**
- integer **iwtr**
- integer **ifrttyp**
- integer **mo\_atmo**
- integer **mo\_atmo1**
- integer **ifirstatmo**
- integer **iy\_r\_atmo**
- integer **iy\_r\_atmo1**
- integer **matmo**
- integer **mch**
  - maximum number of channels*
- integer **mcr**
  - maximum number of crops grown per year*
- integer **mcrdb**
  - maximum number of crops/landcover in database file (crop.dat)*
- integer **mfctst**
  - maximum number of forecast stations*
- integer **mfdb**
  - max number of fertilizers in fert.dat*
- integer **mhru**
  - maximum number of HRUs in watershed*
- integer **mhyd**
  - maximum number of hydrograph nodes*
- integer **mpdb**
  - max number of pesticides in pest.dat*
- integer **mrg**
  - max number of rainfall/temp gages*
- integer **mcut**
  - maximum number of cuttings per year*
- integer **mgr**
  - maximum number of grazings per year*
- integer **mnr**
  - max number of years of rotation*
- integer **myr**
  - max number of years of simulation*
- integer **isubwq**
  - subbasin water quality code*
  - 0 do not calculate algae/CBOD 1 calculate algae/CBOD drainmod tile equations*
- integer **fcst**
- integer **isproj**
  - special project code (none):*
  - 1 test rewind (run simulation twice)*
- integer **nbyr**
  - number of calendar years simulated (none)*
- integer **irte**
  - water routing method (none):*
  - 0 variable storage method*
  - 1 Muskingum method*
- integer **nrch**

- number of reaches in watershed (none)*
- integer **nres**
  - number of reservoirs in watershed (none)*
- integer **nhru**
  - number of last HRU in previous subbasin (none)*
- integer **i\_mo**
  - current month being simulated (none)*
- integer **mo**
- integer **immo**
- integer **wndsim**
  - wind speed input code*
  - 1 measured data read for each subbasin*
  - 2 data simulated for each subbasin*
- integer **ihru**
  - HRU number (none)*
- integer **icode**
- integer **ihout**
- integer **inum1**
- integer **inum2**
- integer **inum3**
- integer **inum4**
- integer **icfac**
  - icfac = 0 for C-factor calculation using Cmin (as described in manual)*
  - = 1 for new C-factor calculation from RUSLE (no minimum needed)*
- integer **inum5**
- integer **inum6**
- integer **inum7**
- integer **inum8**
- integer **mrech**
  - maximum number of rechour files*
- integer **nrgage**
  - number of raingage files (none)*
- integer **nrgfil**
  - number of rain gages per file (none)*
- integer **nrtot**
  - total number of rain gages (none)*
- integer **ntgage**
  - number of temperature gage files (none)*
- integer **ntgfil**
  - number of temperature gages per file (none)*
- integer **nttot**
  - total number of temperature gages (none)*
- integer **tmpsim**
  - temperature input code (none)*
  - 1 measured data read for each subbasin*
  - 2 data simulated for each subbasin*
- integer **icrk**
  - crack flow code*
  - 1: compute flow in cracks*
- integer **irtpest**
  - number of pesticide to be routed through the watershed. Redefined to the sequence number of pesticide in NPNO(:) which is to be routed through the watershed (none)*
- integer **igropt**

- Qual2E option for calculating the local specific growth rate of algae*
  - 1: multiplicative.*
- integer **lao**
  - Qual2E light averaging option. Qual2E defines four light averaging options. The only option currently available in SWAT is #2.*
- integer **npmx**
  - number of different pesticides used in the simulation (none)*
- integer **curyr**
  - current year in simulation (sequence) (none)*
- integer **ihru**
- integer **itdrn**
  - tile drainage equations flag/code*
  - 1 simulate tile flow using subroutine drains(wt\_shall)*
  - 0 simulate tile flow using subroutine origtile(wt\_shall,d)*
- integer **iwtndn**
  - water table depth algorithms flag/code*
  - 1 simulate wt\_shall using subroutine new water table depth routine*
  - 0 simulate wt\_shall using subroutine original water table depth routine*
- integer **ismax**
  - maximum depressional storage selection flag/code*
  - 0 = static depressional storage*
  - 1 = dynamic storage based on tillage and cumulative rainfall*
- integer **iroutunit**
  - not being implemented in this version drainmod tile equations*
- integer **ires\_nut**
- integer **iclb**
  - auto-calibration flag*
- integer **mrecc**
  - maximum number of reccnst files*
- integer **mrecd**
  - maximum number of recday files*
- integer **mrecm**
  - maximum number of recmon files*
- integer **mtil**
  - max number of tillage types in till.dat*
- integer **mudb**
  - maximum number of urban land types in urban.dat*
- integer **idist**
  - rainfall distribution code*
  - 0 for skewed normal dist*
  - 1 for mixed exponential distribution*
- integer **mrecy**
  - maximum number of recyear files*
- integer **nyskip**
  - number of years to not print output*
- integer **slrsim**
  - solar radiation input code*
  - 1 measured data read for each subbasin*
  - 2 data simulated for each subbasin*
- integer **ideg**
  - channel degradation code*
  - 1: compute channel degradation (downcutting and widening)*
- integer **ievent**

- rainfall/runoff code*
  - 0 daily rainfall/curve number technique 1 sub-daily rainfall/Green&Ampt/hourly routing 3 sub-daily rainfall/↔ Green&Ampt/hourly routing*
- integer **ipet**
  - code for potential ET method*
  - 0 Priestley-Taylor method*
  - 1 Penman/Monteith method*
  - 2 Hargreaves method*
  - 3 read in daily potential ET data*
- integer **iopera**
- integer **idaf**
  - beginning day of simulation (julian date)*
- integer **idal**
  - ending day of simulation (julian date)*
- integer **rhsim**
  - relative humidity input code*
  - 1 measured data read for each subbasin*
  - 2 data simulated for each subbasin*
- integer **leapyr**
  - leap year flag (none)*
  - 0 leap year*
  - 1 regular year*
- integer **id1**
- integer **mo\_chk**
- integer **nhtot**
  - number of relative humidity records in file*
- integer **nstot**
  - number of solar radiation records in file*
- integer **nwtot**
  - number of wind speed records in file*
- integer **ifirsts**
- integer **ifirsth**
- integer **ifirstw**
- integer **icst**
- integer **ilog**
  - streamflow print code*
- integer **itotr**
  - number of output variables printed (output.rch)*
- integer **iy**
  - beginning year of simulation (year)*
- integer **iwq**
  - stream water quality code*
  - 0 do not model stream water quality*
  - 1 model stream water quality (QUAL2E & pesticide transformations)*
- integer **iskip**
  - flag for calculations performed only for the first year of simulation (none)*
- integer **ifirstpet**
- integer **iprm**
  - print code for output.pst file*
  - 0 do not print pesticide output*
  - 1 print pesticide output*
- integer **itotb**
  - number of output variables printed (output.sub)*
- integer **itots**

- number of output variables printed (output.hru)*
- integer **itoth**
  - number of HRUs printed (output.hru/output.wtr)*
- integer **pcpsim**
  - rainfall input code*
  - 1 measured data read for each subbasin*
  - 2 data simulated for each subbasin*
- integer **nd\_30**
- integer **iops**
- integer **iphr**
- integer **isto**
- integer **isol**
- integer **fcstcycles**
  - number of times forecast period is simulated (using different weather generator seeds each time)*
- integer **fcstday**
  - beginning date of forecast period (julian date)*
- integer **fcstyr**
  - beginning year of forecast period*
- integer **iscen**
  - scenarios counter*
- integer **subtot**
  - number of subbasins in watershed (none)*
- integer **ogen**
- integer **mapp**
  - maximum number of applications*
- integer **mlyr**
  - maximum number of soil layers*
- integer **mpst**
  - max number of pesticides used in wshed*
- integer **mres**
  - maximum number of reservoirs*
- integer **msub**
  - maximum number of subbasins*
- integer **igen**
  - random number generator seed code (none):*
  - 0: use default numbers*
  - 1: generate new numbers in every simulation*
- integer **iprint**
  - print code: 0=monthly, 1=daily, 2=annual*
- integer **iida**
  - day being simulated (current julian day) (julian date)*
- integer **icn**
  - CN method flag (for testing alternative method):*
  - 0 use traditional SWAT method which bases CN on soil moisture*
  - 1 use alternative method which bases CN on plant ET.*
- integer **ised\_det**
  - max half-hour rainfall fraction calc option:*
  - 0 generate max half-hour rainfall fraction from triangular distribution*
  - 1 use monthly mean max half-hour rainfall fraction*
- integer **fcstcnt**
- integer **mtran**
- integer **idtill**
- integer, dimension(100) **ida\_lup**

- integer, dimension(100) **iyр\_lup**
- integer **no\_lup**
- integer **no\_up**
- integer **nostep**
- character(len=8) **date**  
*date simulation is performed where leftmost eight characters are set to a value of yyyymmdd, where yyyy is the year, mm is the month and dd is the day*
- character(len=10) **time**  
*time simulation is performed where leftmost ten characters are set to a value of hhmmss.sss, where hh is the hour, mm is the minutes and ss.sss is the seconds and milliseconds*
- character(len=5) **zone**  
*time difference with respect to Coordinated Universal Time (ie Greenwich Mean Time)*
- character(len=13) **calfile**  
*name of file containing calibration parameters*
- character(len=13) **rhfile**  
*relative humidity file name (.hmd)*
- character(len=13) **slrfile**  
*solar radiation file name (.slr)*
- character(len=13) **wndfile**  
*wind speed file name (.wnd)*
- character(len=13) **petfile**  
*potential ET file name (.pet)*
- character(len=13) **atmofile**
- character(len=13) **lucfile**
- character(len=13) **septdb**  
*name of septic tank database file (septwq1.dat)*
- character(len=13) **dpd\_file**
- character(len=13) **wpd\_file**
- character(len=13) **rib\_file**
- character(len=13) **sfb\_file**
- character(len=13) **lid\_file**
- integer, dimension(9) **idg**  
*array location of random number seed used for a given process*
- integer, dimension(:), allocatable **ifirstr**
- integer, dimension(:), allocatable **ifirsthr**
- integer, dimension(8) **values**  
*values(1): year simulation is performed  
values(2): month simulation is performed  
values(3): day in month simulation is performed  
values(4): time difference with respect to Coordinated Universal Time (ie Greenwich Mean Time)  
values(5): hour simulation is performed  
values(6): minute simulation is performed  
values(7): second simulation is performed  
values(8): millisecond simulation is performed*
- integer, dimension(13) **ndays**  
*julian date for last day of preceding month (where the array location is the number of the month). The dates are for leap years (julian date)*
- integer, dimension(13) **ndays\_noleap**
- integer, dimension(13) **ndays\_leap**
- integer **mapex**
- real \*8, dimension(:), allocatable **flodaya**
- real \*8, dimension(:), allocatable **seddaya**
- real \*8, dimension(:), allocatable **orgndaya**
- real \*8, dimension(:), allocatable **orgpdaya**

- real \*8, dimension(:), allocatable **no3daya**
- real \*8, dimension(:), allocatable **minpdaya**
- real \*8, dimension(:), allocatable **hi\_targ**  
*harvest index target of cover defined at planting ((kg/ha)/(kg/ha))*
- real \*8, dimension(:), allocatable **bio\_targ**  
*biomass target (kg/ha)*
- real \*8, dimension(:), allocatable **tnyld**
- integer, dimension(:), allocatable **idapa**
- integer, dimension(:), allocatable **itya**
- integer, dimension(:), allocatable **ifirsta**
- integer, dimension(100) **mo\_transb**
- integer, dimension(100) **mo\_transe**
- integer, dimension(100) **ih\_tran**
- integer **msdb**  
*maximum number of sept wq data database (none)*
- integer **iseptic**
- real \*8, dimension(:), allocatable **sptqs**  
*flow rate of the septic tank effluent per capita (m3/d)*
- real \*8, dimension(:), allocatable **percp**
- real \*8, dimension(:), allocatable **sptbodconcs**  
*Biological Oxygen Demand of the septic tank effluent (mg/l)*
- real \*8, dimension(:), allocatable **sptssconcs**  
*concentration of total suspended solid in the septic tank effluent (mg/l)*
- real \*8, dimension(:), allocatable **sptnconcs**  
*concentration of total nitrogen in the septic tank effluent (mg/l)*
- real \*8, dimension(:), allocatable **sptnh4concs**  
*concentration of total phosphorus of the septic tank effluent (mg/l)*
- real \*8, dimension(:), allocatable **sptno3concs**  
*concentration of nitrate in the septic tank effluent (mg/l)*
- real \*8, dimension(:), allocatable **sptno2concs**  
*concentration of nitrite in the septic tank effluent (mg/l)*
- real \*8, dimension(:), allocatable **sptorgnconcs**  
*concentration of organic nitrogen in the septic tank effluent (mg/l)*
- real \*8, dimension(:), allocatable **sptpconcs**  
*concentration of total phosphorus in the septic tank effluent (mg/l)*
- real \*8, dimension(:), allocatable **sptminps**  
*concentration of mineral phosphorus in the septic tank effluent (mg/l)*
- real \*8, dimension(:), allocatable **sptorgps**  
*concentration of organic phosphorus in the septic tank effluent (mg/l)*
- real \*8, dimension(:), allocatable **sptfcolis**  
*concentration of the facel caliform in the septic tank effluent (cfu/100ml)*
- real \*8, dimension(:), allocatable **failyr**
- real \*8, dimension(:), allocatable **qstemm**
- real \*8, dimension(:), allocatable **bio\_amn**
- real \*8, dimension(:), allocatable **bio\_bod**
- real \*8, dimension(:), allocatable **biom**
- real \*8, dimension(:), allocatable **rbiom**
- real \*8, dimension(:), allocatable **fcoli**
- real \*8, dimension(:), allocatable **bio\_ntr**
- real \*8, dimension(:), allocatable **bz\_perc**
- real \*8, dimension(:), allocatable **sep\_cap**  
*number of permanent residents in the hourse (none)*



- real \*8, dimension(:), allocatable **plqm**
- real \*8, dimension(:), allocatable **bz\_area**
- real \*8, dimension(:), allocatable **bz\_z**  
*Depth of biozone layer(mm)*
- real \*8, dimension(:), allocatable **bz\_thk**  
*thickness of biozone (mm)*
- real \*8, dimension(:), allocatable **bio\_bd**  
*density of biomass (kg/m<sup>3</sup>) carbon outputs for .hru file*
- real \*8, dimension(:), allocatable **cmup\_kgh**
- real \*8, dimension(:), allocatable **cmtot\_kgh**
- real \*8, dimension(:), allocatable **coeff\_denitr**  
*denitrification rate coefficient (none)*
- real \*8, dimension(:), allocatable **coeff\_bod\_dc**  
*BOD decay rate coefficient (m<sup>3</sup>/day)*
- real \*8, dimension(:), allocatable **coeff\_bod\_conv**  
*BOD to live bacteria biomass conversion factor (none)*
- real \*8, dimension(:), allocatable **coeff\_fc1**  
*field capacity calibration parameter 1 (none)*
- real \*8, dimension(:), allocatable **coeff\_fc2**  
*field capacity calibration parameter 2 (none)*
- real \*8, dimension(:), allocatable **coeff\_fecal**  
*fecal coliform bacteria decay rate coefficient (m<sup>3</sup>/day)*
- real \*8, dimension(:), allocatable **coeff\_mrt**  
*mortality rate coefficient (none)*
- real \*8, dimension(:), allocatable **coeff\_nitr**  
*nitrification rate coefficient (none)*
- real \*8, dimension(:), allocatable **coeff\_plq**  
*conversion factor for plaque from TDS (none)*
- real \*8, dimension(:), allocatable **coeff\_rsp**  
*respiration rate coefficient (none)*
- real \*8, dimension(:), allocatable **coeff\_slg1**  
*slough-off calibration parameter (none)*
- real \*8, dimension(:), allocatable **coeff\_slg2**  
*slough-off calibration parameter (none)*
- real \*8, dimension(:), allocatable **coeff\_pdistrib**
- real \*8, dimension(:), allocatable **coeff\_solpslp**
- real \*8, dimension(:), allocatable **coeff\_solpintc**
- real \*8, dimension(:), allocatable **coeff\_psorpmx**
- integer, dimension(:), allocatable **isep\_typ**  
*septic system type (none)*
- integer, dimension(:), allocatable **i\_sep**
- integer, dimension(:), allocatable **isep\_opt**  
*septic system operation flag (1=active, 2=failing, 3=not operated) (none)*
- integer, dimension(:), allocatable **sep\_tsincefail**
- integer, dimension(:), allocatable **isep\_tfail**
- integer, dimension(:), allocatable **isep\_iyr**
- integer, dimension(:), allocatable **sep\_strm\_dist**
- integer, dimension(:), allocatable **sep\_den**
- real \*8, dimension(:), allocatable **sol\_sumno3**
- real \*8, dimension(:), allocatable **sol\_sumsolp**
- real \*8, dimension(:), allocatable **strsw\_sum**
- real \*8, dimension(:), allocatable **strstmp\_sum**

- real \*8, dimension(:), allocatable **strsn\_sum**
- real \*8, dimension(:), allocatable **strsp\_sum**
- real \*8, dimension(:), allocatable **strsa\_sum**
- real \*8, dimension(:), allocatable **spill\_hru**
- real \*8, dimension(:), allocatable **tile\_out**
- real \*8, dimension(:), allocatable **hru\_in**
- real \*8, dimension(:), allocatable **spill\_precip**
- real \*8, dimension(:), allocatable **pot\_seep**
- real \*8, dimension(:), allocatable **pot\_evap**
- real \*8, dimension(:), allocatable **pot\_sedin**
- real \*8, dimension(:), allocatable **pot\_solp**  
*soluble P loss rate in the pothole (.01 - 0.5) (1/d)*
- real \*8, dimension(:), allocatable **pot\_solpi**
- real \*8, dimension(:), allocatable **pot\_orgp**
- real \*8, dimension(:), allocatable **pot\_orgpi**
- real \*8, dimension(:), allocatable **pot\_orgn**
- real \*8, dimension(:), allocatable **pot\_orgni**
- real \*8, dimension(:), allocatable **pot\_mps**
- real \*8, dimension(:), allocatable **pot\_mpsi**
- real \*8, dimension(:), allocatable **pot\_mpa**
- real \*8, dimension(:), allocatable **pot\_mpai**
- real \*8, dimension(:), allocatable **pot\_no3i**
- real \*8, dimension(:), allocatable **precip\_in**
- real \*8, dimension(:), allocatable **tile\_sedo**
- real \*8, dimension(:), allocatable **tile\_no3o**
- real \*8, dimension(:), allocatable **tile\_solpo**
- real \*8, dimension(:), allocatable **tile\_orgno**
- real \*8, dimension(:), allocatable **tile\_orgpo**
- real \*8, dimension(:), allocatable **tile\_minpso**
- real \*8, dimension(:), allocatable **tile\_minpao**
- integer **ia\_b**
- integer **ihumus**
- integer **itemp**
- integer **isnow**
- integer, dimension(46) **ipdvar**  
*output variable codes for output.rch file (none)*
- integer, dimension(**mhruo**) **ipdvas**  
*output variable codes for output.hru file (none)*
- integer, dimension(**msubo**) **ipdvab**  
*output variable codes for output.sub file (none)*
- integer, dimension(:), allocatable **ipdhru**  
*HRUs whose output information will be printed to the output.hru and output.wtr files.*
- real \*8, dimension(**mstdo**) **wshddayo**
- real \*8, dimension(**mstdo**) **wshdmono**
- real \*8, dimension(**mstdo**) **wshdyro**
- real \*8, dimension(16) **fcstaao**
- real \*8, dimension(**mstdo**) **wshdaao**
- real \*8, dimension(:,:), allocatable **wpstdayo**
- real \*8, dimension(:,:), allocatable **wpstmono**
- real \*8, dimension(:,:), allocatable **wpstyro**
- real \*8, dimension(:,:), allocatable **yldkg**
- real \*8, dimension(:,:), allocatable **bio\_hv**
- real \*8, dimension(:,:), allocatable **rchmono**  
*reach monthly output array (varies)*

- real \*8, dimension(:,:), allocatable **wpstaao**
- real \*8, dimension(:,:), allocatable **rchyro**
- real \*8, dimension(:,:), allocatable **hrumono**  
*HRU monthly output data array (varies)*
- real \*8, dimension(:,:), allocatable **rchaao**
- real \*8, dimension(:,:), allocatable **rchdy**
- real \*8, dimension(:,:), allocatable **hruyro**
- real \*8, dimension(:,:), allocatable **submono**  
*subbasin monthly output array (varies)*
- real \*8, dimension(:,:), allocatable **hruaao**
- real \*8, dimension(:,:), allocatable **subyro**
- real \*8, dimension(:,:), allocatable **subaao**
- real \*8, dimension(:,:), allocatable **resoutm**  
*reservoir monthly output array (varies)*
- real \*8, dimension(:,:), allocatable **resouty**
- real \*8, dimension(:,:), allocatable **resouta**
- real \*8, dimension(12, 8) **wshd\_aamon**
- real \*8, dimension(:,:), allocatable **wtrmon**  
*HRU monthly output data array for impoundments (varies)*
- real \*8, dimension(:,:), allocatable **wtryr**
- real \*8, dimension(:,:), allocatable **wtraa**
- real \*8, dimension(:,:), allocatable **sub\_smfmx**  
*max melt rate for snow during year (June 21) for subbasin(:) where deg C refers to the air temperature. SUB\_SMFMX and SMFMN allow the rate of snow melt to vary through the year. These parameters are accounting for the impact of soil temperature on snow melt (range: -5.0/5.0) (mm/deg C/day)*
- real \*8, dimension(:,:), allocatable **sub\_smfmn**  
*min melt rate for snow during year (Dec 21) for subbasin(:) (range: -5.0/5.0) where deg C refers to the air temperature (mm/deg C/day)*
- real \*8, dimension(:,:), allocatable **hrupstd**
- real \*8, dimension(:,:), allocatable **hrupsta**
- real \*8, dimension(:,:), allocatable **hrupstm**
- real \*8, dimension(:,:), allocatable **hrupsty**
- integer, dimension(:), allocatable **ifirstt**
- integer, dimension(:), allocatable **ifirstpcp**
- integer, dimension(:), allocatable **elevp**  
*elevation of precipitation gage station (m)*
- integer, dimension(:), allocatable **elevt**  
*elevation of temperature gage station (m)*
- real \*8, dimension(:,:), allocatable **ftmpmn**  
*avg monthly minimum air temperature (deg C)*
- real \*8, dimension(:,:), allocatable **ftmpmx**  
*avg monthly maximum air temperature (deg C)*
- real \*8, dimension(:,:), allocatable **ftmpstdmn**  
*standard deviation for avg monthly minimum air temperature (deg C)*
- real \*8, dimension(:,:), allocatable **ftmpstdmx**  
*standard deviation for avg monthly maximum air temperature (deg C)*
- real \*8, dimension(:,:), allocatable **fpcp\_stat**  
*fpcp\_stat(:,1,:): average amount of precipitation falling in one day for the month (mm/day)  
fpcp\_stat(:,2,:): standard deviation for the average daily precipitation (mm/day)  
fpcp\_stat(:,3,:): skew coefficient for the average daily precipitation (none)*
- real \*8, dimension(:,:), allocatable **fpr\_w1**  
*probability of wet day after dry day in month (none)*
- real \*8, dimension(:,:), allocatable **fpr\_w2**

- probability of wet day after wet day in month (none)*

  - real \*8, dimension(:,:), allocatable **fpr\_w3**
- proportion of wet days in the month (none)*

  - real \*8, dimension(:), allocatable **ch\_d**
- average depth of main channel (m)*

  - real \*8, dimension(:), allocatable **flwin**
  - real \*8, dimension(:), allocatable **flwout**
  - real \*8, dimension(:), allocatable **bankst**
  - real \*8, dimension(:), allocatable **ch\_wi**
  - real \*8, dimension(:), allocatable **ch\_onco**
- channel organic n concentration (ppm)*

  - real \*8, dimension(:), allocatable **ch\_opco**
- channel organic p concentration (ppm)*

  - real \*8, dimension(:), allocatable **ch\_orgn**
  - real \*8, dimension(:), allocatable **ch\_orgp**
  - real \*8, dimension(:), allocatable **drift**
  - real \*8, dimension(:), allocatable **rch\_dox**
  - real \*8, dimension(:), allocatable **rch\_bactp**
  - real \*8, dimension(:), allocatable **alpha\_bnk**
- alpha factor for bank storage recession curve (days)*

  - real \*8, dimension(:), allocatable **alpha\_bnke**
- $\exp(-\alpha_{bnk})$  (none)*

  - real \*8, dimension(:), allocatable **disolvp**
  - real \*8, dimension(:), allocatable **algae**
  - real \*8, dimension(:), allocatable **sedst**
  - real \*8, dimension(:), allocatable **rchstor**
  - real \*8, dimension(:), allocatable **organicon**
  - real \*8, dimension(:), allocatable **organicp**
  - real \*8, dimension(:), allocatable **chlora**
  - real \*8, dimension(:), allocatable **ch\_li**
- initial length of main channel (km)*

  - real \*8, dimension(:), allocatable **ch\_si**
- initial slope of main channel (m/m)*

  - real \*8, dimension(:), allocatable **nitraten**
  - real \*8, dimension(:), allocatable **nitriten**
  - real \*8, dimension(:), allocatable **ch\_bnk\_san**
  - real \*8, dimension(:), allocatable **ch\_bnk\_sil**
  - real \*8, dimension(:), allocatable **ch\_bnk\_cla**
  - real \*8, dimension(:), allocatable **ch\_bnk\_gra**
  - real \*8, dimension(:), allocatable **ch\_bed\_san**
  - real \*8, dimension(:), allocatable **ch\_bed\_sil**
  - real \*8, dimension(:), allocatable **ch\_bed\_cla**
  - real \*8, dimension(:), allocatable **ch\_bed\_gra**
  - real \*8, dimension(:), allocatable **depfp**
  - real \*8, dimension(:), allocatable **depsanfp**
  - real \*8, dimension(:), allocatable **depsilfp**
  - real \*8, dimension(:), allocatable **depclafp**
  - real \*8, dimension(:), allocatable **depsagfp**
  - real \*8, dimension(:), allocatable **deplagfp**
  - real \*8, dimension(:), allocatable **depch**
  - real \*8, dimension(:), allocatable **depsanch**
  - real \*8, dimension(:), allocatable **depsilch**
  - real \*8, dimension(:), allocatable **depclach**

- real \*8, dimension(:), allocatable **depsagch**
- real \*8, dimension(:), allocatable **deplagch**
- real \*8, dimension(:), allocatable **depgrach**
- real \*8, dimension(:), allocatable **depgrafp**
- real \*8, dimension(:), allocatable **grast**
- real \*8, dimension(:), allocatable **r2adj**  
*curve number retention parameter adjustment factor to adjust surface runoff for flat slopes (0.5 - 3.0) (dimensionless)*
- real \*8, dimension(:), allocatable **prf**  
*Reach peak rate adjustment factor for sediment routing in the channel. Allows impact of peak flow rate on sediment routing and channel reshaping to be taken into account (none)*
- real \*8, dimension(:), allocatable **depprch**
- real \*8, dimension(:), allocatable **depprfp**
- real \*8, dimension(:), allocatable **spcon**  
*linear parameter for calculating sediment reentrained in channel sediment routing*
- real \*8, dimension(:), allocatable **spexp**  
*exponent parameter for calculating sediment reentrained in channel sediment routing*
- real \*8, dimension(:), allocatable **sanst**
- real \*8, dimension(:), allocatable **silst**
- real \*8, dimension(:), allocatable **clast**
- real \*8, dimension(:), allocatable **sagst**
- real \*8, dimension(:), allocatable **lagst**
- real \*8, dimension(:), allocatable **pot\_san**
- real \*8, dimension(:), allocatable **pot\_sil**
- real \*8, dimension(:), allocatable **pot\_cla**
- real \*8, dimension(:), allocatable **pot\_sag**
- real \*8, dimension(:), allocatable **pot\_lag**
- real \*8, dimension(:), allocatable **potsani**
- real \*8, dimension(:), allocatable **potsili**
- real \*8, dimension(:), allocatable **potclai**
- real \*8, dimension(:), allocatable **potsagi**
- real \*8, dimension(:), allocatable **potlagi**
- real \*8, dimension(:), allocatable **sanyld**
- real \*8, dimension(:), allocatable **silyld**
- real \*8, dimension(:), allocatable **clayld**
- real \*8, dimension(:), allocatable **sagyld**
- real \*8, dimension(:), allocatable **lagyld**
- real \*8, dimension(:), allocatable **grayld**
- real \*8, dimension(:), allocatable **res\_san**
- real \*8, dimension(:), allocatable **res\_sil**
- real \*8, dimension(:), allocatable **res\_cla**
- real \*8, dimension(:), allocatable **res\_sag**
- real \*8, dimension(:), allocatable **res\_lag**
- real \*8, dimension(:), allocatable **res\_gra**
- real \*8, dimension(:), allocatable **pnd\_san**
- real \*8, dimension(:), allocatable **pnd\_sil**
- real \*8, dimension(:), allocatable **pnd\_cla**
- real \*8, dimension(:), allocatable **pnd\_sag**
- real \*8, dimension(:), allocatable **pnd\_lag**
- real \*8, dimension(:), allocatable **wet\_san**
- real \*8, dimension(:), allocatable **wet\_sil**
- real \*8, dimension(:), allocatable **wet\_cla**
- real \*8, dimension(:), allocatable **wet\_lag**
- real \*8, dimension(:), allocatable **wet\_sag**
- real \*8 **ressano**

- real \*8 **ressilo**
- real \*8 **resclao**
- real \*8 **ressago**
- real \*8 **reslago**
- real \*8 **resgrao**
- real \*8 **ressani**
- real \*8 **ressili**
- real \*8 **resclai**
- real \*8 **ressagi**
- real \*8 **reslagi**
- real \*8 **resgrai**
- real \*8 **potsano**
- real \*8 **potsilo**
- real \*8 **potclao**
- real \*8 **potsago**
- real \*8 **potlago**
- real \*8 **pndsanin**
- real \*8 **pndsilin**
- real \*8 **pndclain**
- real \*8 **pndsagin**
- real \*8 **pndlagin**
- real \*8 **pndsano**
- real \*8 **pndsilo**
- real \*8 **pndclao**
- real \*8 **pndsago**
- real \*8 **pndlago**
- real \*8, dimension(:), allocatable [ch\\_di](#)  
*initial depth of main channel (m)*
- real \*8, dimension(:), allocatable [ch\\_erod](#)  
*channel erodibility factor (0.0-1.0) (none)*  
*0 non-erosive channel*  
*1 no resistance to erosion*
- real \*8, dimension(:), allocatable [ch\\_l2](#)  
*length of main channel (km)*
- real \*8, dimension(:), allocatable **ch\_cov**
- real \*8, dimension(:), allocatable [ch\\_bnk\\_bd](#)  
*bulk density of channel bank sediment (1.1-1.9) (g/cc)*
- real \*8, dimension(:), allocatable [ch\\_bed\\_bd](#)  
*bulk density of channel bed sediment (1.1-1.9) (g/cc)*
- real \*8, dimension(:), allocatable [ch\\_bnk\\_kd](#)  
*erodibility of channel bank sediment by jet test (Peter Allen needs to give more info on this)*
- real \*8, dimension(:), allocatable [ch\\_bed\\_kd](#)  
*erodibility of channel bed sediment by jet test (Peter Allen needs to give more info on this)*
- real \*8, dimension(:), allocatable [ch\\_bnk\\_d50](#)  
*D50(median) particle size diameter of channel bank sediment (0.001 - 20)*
- real \*8, dimension(:), allocatable [ch\\_bed\\_d50](#)  
*D50(median) particle size diameter of channel bed sediment (micrometers) (0.001 - 20)*
- real \*8, dimension(:), allocatable [ch\\_cov1](#)  
*channel erodibility factor (0.0-1.0) (none)*  
*0 non-erosive channel*  
*1 no resistance to erosion*
- real \*8, dimension(:), allocatable [ch\\_cov2](#)

- channel cover factor (0.0-1.0) (none)*  
*0 channel is completely protected from erosion by cover*  
*1 no vegetative cover on channel*
- real \*8, dimension(:), allocatable **tc\_bed**  
*critical shear stress of channel bed (N/m<sup>2</sup>)*
  - real \*8, dimension(:), allocatable **tc\_bnk**  
*critical shear stress of channel bank (N/m<sup>2</sup>)*
  - integer, dimension(:), allocatable **ch\_eqn**  
*sediment routine methods (DAILY):*  
*0 = original SWAT method*  
*1 = Bagnold's*  
*2 = Kodatie*  
*3 = Molinas WU*  
*4 = Yang*
  - real \*8, dimension(:), allocatable **chpst\_rea**  
*pesticide reaction coefficient in reach (1/day)*
  - real \*8, dimension(:), allocatable **chpst\_vol**  
*pesticide volatilization coefficient in reach (m/day)*
  - real \*8, dimension(:), allocatable **chpst\_conc**
  - real \*8, dimension(:), allocatable **chpst\_koc**  
*pesticide partition coefficient between water and sediment in reach (m<sup>3</sup>/g)*
  - real \*8, dimension(:), allocatable **chpst\_rsp**  
*resuspension velocity in reach for pesticide sorbed to sediment (m/day)*
  - real \*8, dimension(:), allocatable **chpst\_stl**  
*settling velocity in reach for pesticide sorbed to sediment (m/day)*
  - real \*8, dimension(:), allocatable **ch\_wdr**  
*channel width to depth ratio (m/m)*
  - real \*8, dimension(:), allocatable **chpst\_mix**  
*mixing velocity (diffusion/dispersion) for pesticide in reach (m/day)*
  - real \*8, dimension(:), allocatable **sedpst\_conc**  
*inital pesticide concentration in river bed sediment (mg/m<sup>3</sup>)*
  - real \*8, dimension(:), allocatable **sedpst\_bry**  
*pesticide burial velocity in river bed sediment (m/day)*
  - real \*8, dimension(:), allocatable **sedpst\_rea**  
*pesticide reaction coefficient in river bed sediment (1/day)*
  - real \*8, dimension(:), allocatable **sedpst\_act**  
*depth of active sediment layer in reach for pesticide (m)*
  - real \*8, dimension(:), allocatable **rch\_cbod**
  - real \*8, dimension(:), allocatable **rch\_bactlp**
  - real \*8, dimension(:), allocatable **chside**  
*change in horizontal distance per unit vertical distance (0.0 - 5)*  
*0 = for vertical channel bank*  
*5 = for channel bank with gentl side slope*
  - real \*8, dimension(:), allocatable **rs1**  
*local algal settling rate in reach at 20 deg C (m/day or m/hour)*
  - real \*8, dimension(:), allocatable **rs2**  
*benthos source rate for dissolved phosphorus in reach at 20 deg C ((mg disP-P)/(m<sup>2</sup>\*day) or (mg disP-P)/(m<sup>2</sup>\*hour))*
  - real \*8, dimension(:), allocatable **rs3**  
*benthos source rate for ammonia nitrogen in reach at 20 deg C ((mg NH4-N)/(m<sup>2</sup>\*day) or (mg NH4-N)/(m<sup>2</sup>\*hour))*
  - real \*8, dimension(:), allocatable **rs4**  
*rate coefficient for organic nitrogen settling in reach at 20 deg C (1/day or 1/hour)*
  - real \*8, dimension(:), allocatable **rs5**

- organic phosphorus settling rate in reach at 20 deg C (1/day or 1/hour)*
- real \*8, dimension(:), allocatable [rk1](#)
- CBOD deoxygenation rate coefficient in reach at 20 deg C (1/day or 1/hour)*
- real \*8, dimension(:), allocatable [rk2](#)
- reaeration rate in accordance with Fickian diffusion in reach at 20 deg C (1/day or 1/hour)*
- real \*8, dimension(:), allocatable [rk3](#)
- rate of loss of CBOD due to settling in reach at 20 deg C (1/day or 1/hour)*
- real \*8, dimension(:), allocatable [rk4](#)
- sediment oxygen demand rate in reach at 20 deg C (mg O2/(m<sup>2</sup>\*day) or mg O2/(m<sup>2</sup>\*hour))*
- real \*8, dimension(:), allocatable [rk5](#)
- coliform die-off rate in reach (1/day)*
- real \*8, dimension(:), allocatable [rs6](#)
- rate coefficient for settling of arbitrary non-conservative constituent in reach (1/day)*
- real \*8, dimension(:), allocatable [rs7](#)
- benthic source rate for arbitrary non-conservative constituent in reach ((mg ANC)/(m<sup>2</sup>\*day))*
- real \*8, dimension(:), allocatable [bc1](#)
- rate constant for biological oxidation of NH3 to NO2 in reach at 20 deg C (1/day or 1/hour)*
- real \*8, dimension(:), allocatable [bc2](#)
- rate constant for biological oxidation of NO2 to NO3 in reach at 20 deg C (1/day or 1/hour)*
- real \*8, dimension(:), allocatable [bc3](#)
- rate constant for hydrolysis of organic N to ammonia in reach at 20 deg C (1/day or 1/hour)*
- real \*8, dimension(:), allocatable [bc4](#)
- rate constant for the decay of organic P to dissolved P in reach at 20 deg C (1/day or 1/hour)*
- real \*8, dimension(:), allocatable [rk6](#)
- decay rate for arbitrary non-conservative constituent in reach (1/day)*
- real \*8, dimension(:), allocatable **ammonian**
- real \*8, dimension(:), allocatable **orig\_sedpstconc**
- real \*8, dimension(:,:), allocatable [wurch](#)
- average daily water removal from the reach for the month (10<sup>4</sup> m<sup>3</sup>/day)*
- integer, dimension(:), allocatable **icanal**
- integer, dimension(:), allocatable **itb**
- real \*8, dimension(:), allocatable [ch\\_revap](#)
- revap coeff: this variable controls the amount of water moving from bank storage to the root zone as a result of soil moisture depletion(none)*
- real \*8, dimension(:), allocatable **dep\_chan**
- real \*8, dimension(:), allocatable [harg\\_petco](#)
- coefficient related to radiation used in hargreaves eq (range: 0.0019 - 0.0032)*
- real \*8, dimension(:), allocatable **subfr\_nowtr**
- real \*8, dimension(:), allocatable [cncoef\\_sub](#)
- soil water depletion coefficient used in the new (modified curve number method) same as soil index coeff used in APEX range: 0.5 - 2.0*
- real \*8, dimension(:), allocatable **dr\_sub**
- real \*8, dimension(:), allocatable [sub\\_fr](#)
- fraction of total watershed area contained in subbasin (km2/km2)*
- real \*8, dimension(:), allocatable **wcklsp**
- real \*8, dimension(:), allocatable **sub\_minp**
- real \*8, dimension(:), allocatable **sub\_sw**
- real \*8, dimension(:), allocatable **sub\_sumfc**
- real \*8, dimension(:), allocatable **sub\_gwno3**
- real \*8, dimension(:), allocatable **sub\_gwsolp**
- real \*8, dimension(:), allocatable [co2](#)
- CO2 concentration (ppmv)*



- real \*8, dimension(:), allocatable [sub\\_km](#)  
*area of subbasin in square kilometers (km<sup>2</sup>)*
- real \*8, dimension(:), allocatable [wlat](#)  
*latitude of weather station used to compile data (degrees)*
- real \*8, dimension(:), allocatable [sub\\_tc](#)  
*time of concentration for subbasin (hour)*
- real \*8, dimension(:), allocatable [sub\\_pet](#)
- real \*8, dimension(:), allocatable [welev](#)  
*elevation of weather station used to compile data (m)*
- real \*8, dimension(:), allocatable [sub\\_orgn](#)
- real \*8, dimension(:), allocatable [sub\\_orgp](#)
- real \*8, dimension(:), allocatable [sub\\_bd](#)
- real \*8, dimension(:), allocatable [sub\\_wtmp](#)
- real \*8, dimension(:), allocatable [sub\\_sedpa](#)
- real \*8, dimension(:), allocatable [sub\\_sedps](#)
- real \*8, dimension(:), allocatable [daylmn](#)  
*shortest daylength occurring during the year (hour)*
- real \*8, dimension(:), allocatable [sub\\_minpa](#)
- real \*8, dimension(:), allocatable [sub\\_minps](#)
- real \*8, dimension(:), allocatable [latcos](#)  
*cos(latitude) (none)*
- real \*8, dimension(:), allocatable [latsin](#)  
*sin(latitude) (none)*
- real \*8, dimension(:), allocatable [phutot](#)  
*total potential heat units for year (used when no crop is growing) (heat unit)*
- real \*8, dimension(:), allocatable [plaps](#)  
*precipitation lapse rate: precipitation change due to change in elevation (mm H<sub>2</sub>O/km)*
- real \*8, dimension(:), allocatable [tlaps](#)  
*temperature lapse rate: temperature change due to change in elevation (deg C/km)*
- real \*8, dimension(:), allocatable [tmp\\_an](#)  
*average annual air temperature (deg C)*
- real \*8, dimension(:), allocatable [sub\\_precip](#)
- real \*8, dimension(:), allocatable [rammo\\_sub](#)  
*atmospheric deposition of ammonium values for entire watershed (mg/l)*
- real \*8, dimension(:), allocatable [rcn\\_sub](#)  
*atmospheric deposition of nitrate for entire watershed (mg/l)*
- real \*8, dimension(:), allocatable [pcpdays](#)
- real \*8, dimension(:), allocatable [atmo\\_day](#)
- real \*8, dimension(:), allocatable [sub\\_snom](#)
- real \*8, dimension(:), allocatable [sub\\_qd](#)
- real \*8, dimension(:), allocatable [sub\\_sedy](#)
- real \*8, dimension(:), allocatable [sub\\_tran](#)
- real \*8, dimension(:), allocatable [sub\\_no3](#)
- real \*8, dimension(:), allocatable [sub\\_latno3](#)
- real \*8, dimension(:,:), allocatable [sub\\_sftmp](#)  
*snowfall temperature for subbasin(:). Mean air temperature at which precip is equally likely to be rain as snow/freezing rain (range: -5.0/5.0) (deg C)*
- real \*8, dimension(:,:), allocatable [sub\\_smtmp](#)  
*snow melt base temperature for subbasin(:) mean air temperature at which snow melt will occur (range: -5.0/5.0) (deg C)*
- real \*8, dimension(:,:), allocatable [sub\\_timp](#)  
*snow pack temperature lag factor (0-1) (none)*

- real \*8, dimension(:), allocatable **sub\_tilen3**
- real \*8, dimension(:), allocatable **sub\_solp**
- real \*8, dimension(:), allocatable **sub\_subp**
- real \*8, dimension(:), allocatable **sub\_etday**
- real \*8, dimension(:), allocatable **sub\_elev**  
*average elevation of subbasin (m)*
- real \*8, dimension(:), allocatable **sub\_wyld**
- real \*8, dimension(:), allocatable **sub\_surfq**
- real \*8, dimension(:), allocatable **qird**
- real \*8, dimension(:), allocatable **sub\_gwq**
- real \*8, dimension(:), allocatable **sub\_sep**
- real \*8, dimension(:), allocatable **sub\_chl**
- real \*8, dimension(:), allocatable **sub\_cbod**
- real \*8, dimension(:), allocatable **sub\_dox**
- real \*8, dimension(:), allocatable **sub\_solpst**
- real \*8, dimension(:), allocatable **sub\_sorpst**
- real \*8, dimension(:), allocatable **sub\_yorgn**
- real \*8, dimension(:), allocatable **sub\_yorgp**
- real \*8, dimension(:), allocatable **sub\_lat**  
*latitude of HRU/subbasin (degrees)*
- real \*8, dimension(:), allocatable **sub\_bactp**
- real \*8, dimension(:), allocatable **sub\_bactlp**
- real \*8, dimension(:), allocatable **sub\_latq**
- real \*8, dimension(:), allocatable **sub\_gwq\_d**
- real \*8, dimension(:), allocatable **sub\_tileq**
- real \*8, dimension(:), allocatable **sub\_vaptile**
- real \*8, dimension(:), allocatable **sub\_dsan**
- real \*8, dimension(:), allocatable **sub\_dsil**
- real \*8, dimension(:), allocatable **sub\_dcla**
- real \*8, dimension(:), allocatable **sub\_dsag**
- real \*8, dimension(:), allocatable **sub\_dlag**
- real \*8 **vap\_tile**
- real \*8, dimension(:), allocatable **wnan**
- real \*8, dimension(:,:), allocatable **sol\_stpwt**
- real \*8, dimension(:,:), allocatable **sub\_pst**
- real \*8, dimension(:,:), allocatable **sub\_hhq**
- real \*8, dimension(:,:), allocatable **sub\_hhwtmp**
- real \*8, dimension(:,:), allocatable **huminc**  
*monthly humidity adjustment. Daily values for relative humidity within the month are rasied or lowered by the specified amount (used in climate change studies) (none)*
- real \*8, dimension(:,:), allocatable **radinc**  
*monthly solar radiation adjustment. Daily radiation within the month is raised or lowered by the specified amount. (used in climate change studies) (MJ/m<sup>2</sup>)*
- real \*8, dimension(:,:), allocatable **rfinc**  
*monthly rainfall adjustment. Daily rainfall within the month is adjusted to the specified percentage of the original value (used in climate change studies)(%)*
- real \*8, dimension(:,:), allocatable **tmpinc**  
*monthly temperature adjustment. Daily maximum and minimum temperatures within the month are raised or lowered by the specified amount (used in climate change studies) (deg C)*
- real \*8, dimension(:), allocatable **ch\_k1**  
*effective hydraulic conductivity of tributary channel alluvium (mm/hr)*
- real \*8, dimension(:), allocatable **ch\_k2**  
*effective hydraulic conductivity of main channel alluvium (mm/hr)*
- real \*8, dimension(:,:), allocatable **elevb**

- elevation at the center of the band (m)*
- real \*8, dimension(:,:), allocatable [elevb\\_fr](#)
  - fraction of subbasin area within elevation band (the same fractions should be listed for all HRUs within the subbasin)*
  - (none)*
- real \*8, dimension(:,:), allocatable [wndav](#)
  - average wind speed for the month (m/s)*
- real \*8, dimension(:), allocatable [ch\\_n1](#)
  - Manning's "n" value for the tributary channels (none)*
- real \*8, dimension(:), allocatable [ch\\_n2](#)
  - Manning's "n" value for the main channel (none)*
- real \*8, dimension(:), allocatable [ch\\_s1](#)
  - average slope of tributary channels (m/m)*
- real \*8, dimension(:), allocatable [ch\\_s2](#)
  - average slope of main channel (m/m)*
- real \*8, dimension(:), allocatable [ch\\_w1](#)
  - average width of tributary channels (m)*
- real \*8, dimension(:), allocatable [ch\\_w2](#)
  - average width of main channel (m)*
- real \*8, dimension(:,:), allocatable [dewpt](#)
  - average dew point temperature for the month (deg C)*
- real \*8, dimension(:,:), allocatable [amp\\_r](#)
  - average fraction of total daily rainfall occurring in maximum half-hour period for month (none)*
- real \*8, dimension(:,:), allocatable [solarav](#)
  - average daily solar radiation for the month (MJ/m<sup>2</sup>/day)*
- real \*8, dimension(:,:), allocatable **tmpstdmx**
- real \*8, dimension(:,:), allocatable [pcf](#)
  - normalization coefficient for precipitation generator (none)*
- real \*8, dimension(:,:), allocatable [tmpmn](#)
  - avg monthly minimum air temperature (deg C)*
- real \*8, dimension(:,:), allocatable [tmpmx](#)
  - avg monthly maximum air temperature (deg C)*
- real \*8, dimension(:,:), allocatable **tmpstdmn**
- real \*8, dimension(:,:), allocatable **otmpstdmn**
- real \*8, dimension(:,:), allocatable **otmpmn**
- real \*8, dimension(:,:), allocatable **otmpmx**
- real \*8, dimension(:,:), allocatable **otmpstdmx**
- real \*8, dimension(:,:), allocatable **ch\_erodmo**
- real \*8, dimension(:,:), allocatable **uh**
- real \*8, dimension(:,:), allocatable **hqdsave**
- real \*8, dimension(:,:), allocatable **hsdsave**
- real \*8, dimension(:,:), allocatable [pr\\_w1](#)
  - probability of wet day after dry day in month (none)*
- real \*8, dimension(:,:), allocatable [pr\\_w2](#)
  - probability of wet day after wet day in month (none)*
- real \*8, dimension(:,:), allocatable [pr\\_w3](#)
  - proportion of wet days in the month (none)*
- real \*8, dimension(:,:), allocatable **pcp\_stat**
- real \*8, dimension(:,:), allocatable **opr\_w1**
- real \*8, dimension(:,:), allocatable **opr\_w2**
- real \*8, dimension(:,:), allocatable **opr\_w3**
- real \*8, dimension(:,:), allocatable **opcp\_stat**
- integer, dimension(:), allocatable [ireg](#)

- precipitation category (none):*
  - 1 *precipitation* ≤ 508 mm/yr
  - 2 *precipitation* > 508 and ≤ 1016 mm/yr
  - 3 *precipitation* > 1016 mm/yr
- integer, dimension(:), allocatable **hrutot**  
*number of HRUs in subbasin (none)*
- integer, dimension(:), allocatable **hru1**
- integer, dimension(:), allocatable **ihgage**  
*subbasin relative humidity data code (none)*
- integer, dimension(:), allocatable **isgage**  
*subbasin radiation gage data code (none)*
- integer, dimension(:), allocatable **iwgage**  
*subbasin wind speed gage data code (none)*
- integer, dimension(:), allocatable **subgis**  
*GIS code printed to output files (output.sub) (none)*
- integer, dimension(:), allocatable **irgage**  
*subbasin rain gage data code (none)*
- integer, dimension(:), allocatable **itgage**  
*subbasin temp gage data code (none)*
- integer, dimension(:), allocatable **irelh**  
*(none) irelh = 0 (dewpoint)*  
*irelh = 1 (relative humidity)*  
*note: inputs > 1.0 (dewpoint)*  
*inputs < 1.0 (relative hum)*
- integer, dimension(:), allocatable **fcst\_reg**
- real \*8, dimension(:,:), allocatable **sol\_aorgn**  
*amount of nitrogen stored in the active organic (humic) nitrogen pool (kg N/ha)*
- real \*8, dimension(:,:), allocatable **sol\_fon**  
*amount of nitrogen stored in the fresh organic (residue) pool (kg N/ha)*
- real \*8, dimension(:,:), allocatable **sol\_tmp**
- real \*8, dimension(:,:), allocatable **sol\_awc**  
*available water capacity of soil layer (mm H2O/mm soil)*
- real \*8, dimension(:,:), allocatable **volcr**  
*crack volume for soil layer (mm)*
- real \*8, dimension(:,:), allocatable **sol\_prk**
- real \*8, dimension(:,:), allocatable **pperco\_sub**  
*subbasin phosphorus percolation coefficient. Ratio of soluble phosphorus in surface to soluble phosphorus in percolate*
- real \*8, dimension(:,:), allocatable **sol\_stap**  
*amount of phosphorus in the soil layer stored in the stable mineral phosphorus pool(kg P/ha)*
- real \*8, dimension(:,:), allocatable **conv\_wt**  
*factor which converts kg/kg soil to kg/ha (none)*
- real \*8, dimension(:,:), allocatable **sol\_actp**  
*amount of phosphorus stored in the active mineral phosphorus pool (kg P/ha)*
- real \*8, dimension(:,:), allocatable **sol\_solp**  
*soluble P concentration in top soil layer (mg P/kg soil) or*  
*amount of phosphorus stored in solution. NOTE UNIT CHANGE! (kg P/ha)*
- real \*8, dimension(:,:), allocatable **crdep**  
*maximum or potential crack volume (mm)*
- real \*8, dimension(:,:), allocatable **sol\_fc**  
*amount of water available to plants in soil layer at field capacity (fc - wp) (mm H2O)*
- real \*8, dimension(:,:), allocatable **sol\_ul**  
*amount of water held in the soil layer at saturation (sat - wp water) (mm H2O)*

- real \*8, dimension(:,:), allocatable **sol\_bd**  
*bulk density of the soil ( $\text{Mg/m}^3$ )*
- real \*8, dimension(:,:), allocatable **sol\_z**  
*depth to bottom of soil layer (mm)*
- real \*8, dimension(:,:), allocatable **sol\_st**  
*amount of water stored in the soil layer on any given day (less wp water) (mm H<sub>2</sub>O)*
- real \*8, dimension(:,:), allocatable **sol\_up**  
*water content of soil at -0.033 MPa (field capacity) (mm H<sub>2</sub>O/mm soil)*
- real \*8, dimension(:,:), allocatable **sol\_clay**  
*percent clay content in soil material (UNIT CHANGE!) (% or none)*
- real \*8, dimension(:,:), allocatable **sol\_hk**  
*beta coefficient to calculate hydraulic conductivity (none)*
- real \*8, dimension(:,:), allocatable **flat**
- real \*8, dimension(:,:), allocatable **sol\_nh3**
- real \*8, dimension(:,:), allocatable **sol\_ec**  
*electrical conductivity of soil layer (dS/m)*
- real \*8, dimension(:,:), allocatable **sol\_orgn**  
*amount of nitrogen stored in the stable organic N pool. NOTE UNIT CHANGE! (mg N/kg soil or kg N/ha)*
- real \*8, dimension(:,:), allocatable **sol\_por**  
*total porosity of soil layer expressed as a fraction of the total volume (none)*
- real \*8, dimension(:,:), allocatable **sol\_wp**  
*water content of soil at -1.5 MPa (wilting point) (mm H<sub>2</sub>O/mm soil)*
- real \*8, dimension(:,:), allocatable **sol\_orgp**  
*amount of phosphorus stored in the organic P pool. NOTE UNIT CHANGE! (mg P/kg soil or kg P/ha)*
- real \*8, dimension(:,:), allocatable **sol\_hum**  
*amount of organic matter in the soil layer classified as humic substances (kg humus/ha)*
- real \*8, dimension(:,:), allocatable **sol\_wpmmm**  
*water content of soil at -1.5 MPa (wilting point) (mm H<sub>2</sub>O)*
- real \*8, dimension(:,:), allocatable **sol\_no3**  
*amount of nitrogen stored in the nitrate pool. This variable is read in as a concentration and converted to kg/ha (this value is read from the .sol file in units of mg/kg) (kg N/ha)*
- real \*8, dimension(:,:), allocatable **sol\_cbn**  
*percent organic carbon in soil layer (%)*
- real \*8, dimension(:,:), allocatable **sol\_k**  
*saturated hydraulic conductivity of soil layer (mm/hour)*
- real \*8, dimension(:,:), allocatable **sol\_rsd**  
*amount of organic matter in the soil layer classified as residue (kg/ha)*
- real \*8, dimension(:,:), allocatable **sol\_fop**  
*amount of phosphorus stored in the fresh organic (residue) pool (kg P/ha)*
- real \*8, dimension(:,:), allocatable **sol\_rock**  
*percent of rock fragments in soil layer (%)*
- real \*8, dimension(:,:), allocatable **sol\_silt**  
*percent silt content in soil material (UNIT CHANGE!) (% or none)*
- real \*8, dimension(:,:), allocatable **sol\_sand**  
*percent sand content of soil material (%)*
- real \*8, dimension(:,:), allocatable **orig\_solno3**
- real \*8, dimension(:,:), allocatable **orig\_solorgn**
- real \*8, dimension(:,:), allocatable **orig\_solsolp**
- real \*8, dimension(:,:), allocatable **orig\_solorgp**
- real \*8, dimension(:,:), allocatable **orig\_soltmp**
- real \*8, dimension(:,:), allocatable **orig\_solrsd**

- real \*8, dimension(:,:), allocatable **orig\_solfop**
- real \*8, dimension(:,:), allocatable **orig\_solfon**
- real \*8, dimension(:,:), allocatable **orig\_solaorgn**
- real \*8, dimension(:,:), allocatable **orig\_solst**
- real \*8, dimension(:,:), allocatable **orig\_solactp**
- real \*8, dimension(:,:), allocatable **orig\_solstap**
- real \*8, dimension(:,:), allocatable **orig\_volcr**
- real \*8, dimension(:,:), allocatable **conk**
- real \*8, dimension(:,:), allocatable **sol\_pst**  
*sol\_pst(:,1) initial amount of pesticide in first layer read in from .chm file (mg/kg)*  
*sol\_pst(:,,:) amount of pesticide in layer. NOTE UNIT CHANGE! (kg/ha)*
- real \*8, dimension(:,:), allocatable **sol\_kp**  
*pesticide sorption coefficient, Kp; the ratio of the concentration in the solid phase to the concentration in solution ((mg/kg)/(mg/L))*
- real \*8, dimension(:,:), allocatable **orig\_solpst**
- real \*8, dimension(:), allocatable **velsetlr**
- real \*8, dimension(:), allocatable **velsetlp**
- real \*8, dimension(:), allocatable **br1**  
*1st shape parameter for reservoir surface area equation (none)*
- real \*8, dimension(:), allocatable **evrsv**  
*lake evaporation coefficient (none)*
- real \*8, dimension(:), allocatable **res\_k**  
*hydraulic conductivity of the reservoir bottom (mm/hr)*
- real \*8, dimension(:), allocatable **lkpst\_conc**  
*pesticide concentration in lake water (mg/m<sup>3</sup>)*
- real \*8, dimension(:), allocatable **res\_evol**  
*volume of water needed to fill the reservoir to the emergency spillway (read in as 10<sup>4</sup> m<sup>3</sup> and converted to m<sup>3</sup>) (m<sup>3</sup>)*
- real \*8, dimension(:), allocatable **res\_pvol**  
*volume of water needed to fill the reservoir to the principal spillway (read in as 10<sup>4</sup> m<sup>3</sup> and converted to m<sup>3</sup>) (m<sup>3</sup>)*
- real \*8, dimension(:), allocatable **res\_vol**  
*reservoir volume (read in as 10<sup>4</sup> m<sup>3</sup> and converted to m<sup>3</sup>) (m<sup>3</sup>)*
- real \*8, dimension(:), allocatable **res\_psa**  
*reservoir surface area when reservoir is filled to principal spillway (ha)*
- real \*8, dimension(:), allocatable **lkpst\_rea**  
*pesticide reaction coefficient in lake water (1/day)*
- real \*8, dimension(:), allocatable **lkpst\_vol**  
*pesticide volatilization coefficient in lake water (m/day)*
- real \*8, dimension(:), allocatable **br2**  
*2nd shape parameter for reservoir surface area equation (none)*
- real \*8, dimension(:), allocatable **res\_rr**  
*average daily principal spillway release volume (read in as a release rate in m<sup>3</sup>/s and converted to m<sup>3</sup>/day) (m<sup>3</sup>/day)*
- real \*8, dimension(:), allocatable **res\_sed**  
*amount of sediment in reservoir (read in as mg/L and converted to kg/L) (kg/L)*
- real \*8, dimension(:), allocatable **lkpst\_koc**  
*pesticide partition coefficient between water and sediment in lake water (m<sup>3</sup>/g)*
- real \*8, dimension(:), allocatable **lkpst\_mix**  
*mixing velocity (diffusion/dispersion) in lake water for pesticide (m/day)*
- real \*8, dimension(:), allocatable **lkpst\_rsp**  
*resuspension velocity in lake water for pesticide sorbed to sediment (m/day)*

- real \*8, dimension(:), allocatable **lkpst\_stl**  
*settling velocity in lake water for pesticide sorbed to sediment (m/day)*
- real \*8, dimension(:), allocatable **lkspst\_conc**  
*pesticide concentration in lake bed sediment (mg/m<sup>3</sup>)*
- real \*8, dimension(:), allocatable **lkspst\_rea**  
*pesticide reaction coefficient in lake bed sediment (1/day)*
- real \*8, dimension(:), allocatable **theta\_n**
- real \*8, dimension(:), allocatable **theta\_p**
- real \*8, dimension(:), allocatable **con\_nirr**
- real \*8, dimension(:), allocatable **con\_pirr**
- real \*8, dimension(:), allocatable **lkspst\_act**  
*depth of active sediment layer in lake for for pesticide (m)*
- real \*8, dimension(:), allocatable **lkspst\_bry**  
*pesticide burial velocity in lake bed sediment (m/day)*
- real \*8, dimension(:), allocatable **sed\_stlr**
- real \*8, dimension(7) **resdata**
- real \*8, dimension(:), allocatable **res\_nsed**  
*normal amount of sediment in reservoir (read in as mg/L and convert to kg/L) (kg/L)*
- real \*8, dimension(:), allocatable **wurtmf**  
*fraction of water removed from the reservoir via WURESN which is returned and becomes flow from the reservoir outlet (none)*
- real \*8, dimension(:), allocatable **chlar**  
*chlorophyll-a production coefficient for reservoir (none)*
- real \*8, dimension(:), allocatable **res\_no3**  
*amount of nitrate in reservoir (kg N)*
- real \*8, dimension(:), allocatable **res\_orgn**  
*amount of organic N in reservoir (kg N)*
- real \*8, dimension(:), allocatable **res\_orgp**  
*amount of organic P in reservoir (kg P)*
- real \*8, dimension(:), allocatable **res\_solp**  
*amount of soluble P in reservoir (kg P)*
- real \*8, dimension(:), allocatable **res\_chla**
- real \*8, dimension(:), allocatable **res\_seci**
- real \*8, dimension(:), allocatable **res\_esa**  
*reservoir surface area when reservoir is filled to emergency spillway (ha)*
- real \*8, dimension(:), allocatable **res\_nh3**  
*amount of ammonia in reservoir (kg N)*
- real \*8, dimension(:), allocatable **res\_no2**  
*amount of nitrite in reservoir (kg N)*
- real \*8, dimension(:), allocatable **seccir**  
*water clarity coefficient for reservoir (none)*
- real \*8, dimension(:), allocatable **res\_bactp**
- real \*8, dimension(:), allocatable **res\_bactlp**
- real \*8, dimension(:), allocatable **oflowmn\_fps**  
*minimum reservoir outflow as a fraction of the principal spillway volume (0-1) (fraction)*
- real \*8, dimension(:), allocatable **starg\_fps**  
*target volume as a fraction of the principal spillway volume (.1-5) (fraction)*
- real \*8, dimension(:), allocatable **weirc**
- real \*8, dimension(:), allocatable **weirk**
- real \*8, dimension(:), allocatable **weirw**
- real \*8, dimension(:), allocatable **acoef**
- real \*8, dimension(:), allocatable **bcoef**



- real \*8, dimension(:), allocatable **ccoef**
- real \*8, dimension(:), allocatable **orig\_resvol**
- real \*8, dimension(:), allocatable **orig\_ressed**
- real \*8, dimension(:), allocatable **orig\_lkpstconc**
- real \*8, dimension(:), allocatable **orig\_lkspstconc**
- real \*8, dimension(:), allocatable **orig\_ressolp**
- real \*8, dimension(:), allocatable **orig\_resorgp**
- real \*8, dimension(:), allocatable **orig\_resno3**
- real \*8, dimension(:), allocatable **orig\_resno2**
- real \*8, dimension(:), allocatable **orig\_resnh3**
- real \*8, dimension(:), allocatable **orig\_resorgn**
- real \*8, dimension(:,:), allocatable **oflowmn**  
*minimum daily outflow for the month (read in as  $m^3/s$  and converted to  $m^3/day$ ) ( $m^3/day$ )*
- real \*8, dimension(:,:), allocatable **oflowmx**  
*maximum daily outflow for the month (read in as  $m^3/s$  and converted to  $m^3/day$ ) ( $m^3/day$ )*
- real \*8, dimension(:,:), allocatable **starg**  
*monthly target reservoir storage (needed if IRESKO=2) (read in as  $10^4 m^3$  and converted to  $m^3$ ) ( $m^3$ )*
- real \*8, dimension(:), allocatable **psetlr1**  
*phosphorus settling rate for mid-year period (read in as  $m/year$  and converted to  $m/day$ ) ( $m/day$ )*
- real \*8, dimension(:), allocatable **psetlr2**  
*phosphorus settling rate for remainder of year (read in as  $m/year$  and converted to  $m/day$ ) ( $m/day$ )*
- real \*8, dimension(:), allocatable **nsetlr1**  
*nitrogen settling rate for mid-year period (read in as  $m/year$  and converted to  $m/day$ ) ( $m/day$ )*
- real \*8, dimension(:), allocatable **nsetlr2**  
*nitrogen settling rate for remainder of year (read in as  $m/year$  and converted to  $m/day$ ) ( $m/day$ )*
- real \*8, dimension(:,:), allocatable **wuresn**  
*average amount of water withdrawn from reservoir each month for consumptive water use (read in as  $10^4 m^3$  and converted to  $m^3$ ) ( $m^3$ )*
- real \*8, dimension(:,:), allocatable **res\_out**  
*measured average daily outflow from the reservoir for the month (needed if IRESKO=1) (read in as  $m^3/s$  and converted to  $m^3/day$ ) ( $m^3/day$ )*
- integer, dimension(:), allocatable **res\_sub**  
*number of subbasin reservoir is in (weather for the subbasin is used for the reservoir) (none)*
- integer, dimension(:), allocatable **ires1**  
*beginning of mid-year nutrient settling "season" (none)*
- integer, dimension(:), allocatable **ires2**  
*end of mid-year nutrient settling "season" (none)*
- integer, dimension(:), allocatable **iresco**  
*outflow simulation code (none):*  
*0 compute outflow for uncontrolled reservoir with average annual release rate*  
*1 measured monthly outflow*  
*2 simulated controlled outflow-target release*  
*3 measured daily outflow*  
*4 stage/volume/outflow relationship*
- integer, dimension(:), allocatable **iyres**  
*year of the simulation that the reservoir becomes operational (none)*
- integer, dimension(:), allocatable **mores**  
*month the reservoir becomes operational (none)*
- integer, dimension(:), allocatable **iflod1r**  
*beginning month of non-flood season (needed if IRESKO=2) (none)*
- integer, dimension(:), allocatable **iflod2r**  
*ending month of non-flood season (needed if IRESKO=2) (none)*
- integer, dimension(:), allocatable **ndtargr**



- number of days to reach target storage from current reservoir storage (needed if IRESKO=2) (days)*
- real \*8, dimension(:), allocatable **ap\_ef**
  - application efficiency (0-1) (none)*
- real \*8, dimension(:), allocatable **decay\_f**
  - exponential of the rate constant for degradation of the pesticide on foliage (none)*
- real \*8, dimension(:), allocatable **skoc**
  - soil adsorption coefficient normalized for soil organic carbon content ((mg/kg)/(mg/L))*
- real \*8, dimension(:), allocatable **decay\_s**
  - exponential of the rate constant for degradation of the pesticide in soil (none)*
- real \*8, dimension(:), allocatable **hlife\_f**
  - half-life of pesticide on foliage (days)*
- real \*8, dimension(:), allocatable **hlife\_s**
  - half-life of pesticide in soil (days)*
- real \*8, dimension(:), allocatable **pst\_wof**
  - fraction of pesticide on foliage which is washed-off by a rainfall event (none)*
- real \*8, dimension(:), allocatable **pst\_wsol**
  - solubility of chemical in water (mg/L (ppm))*
- real \*8, dimension(:), allocatable **irramt**
- real \*8, dimension(:), allocatable **phusw**
- real \*8, dimension(:), allocatable **phusw\_nocrop**
- integer, dimension(:), allocatable **pstflg**
  - flag for types of pesticide used in watershed. Array location is pesticide ID number*
  - 0: pesticide not used*
  - 1: pesticide used*
- integer, dimension(:), allocatable **nope**
  - sequence number of pesticide in NPNO(:) (none)*
- integer, dimension(:), allocatable **nop**
- integer, dimension(:), allocatable **yr\_skip**
- integer, dimension(:), allocatable **isweep**
- integer, dimension(:), allocatable **icrmx**
- integer, dimension(:), allocatable **nopmx**
- integer, dimension(:,:), allocatable **mgtop**
- integer, dimension(:,:), allocatable **idop**
- integer, dimension(:,:), allocatable **mgt1iop**
- integer, dimension(:,:), allocatable **mgt2iop**
- integer, dimension(:,:), allocatable **mgt3iop**
- real \*8, dimension(:,:), allocatable **mgt4op**
- real \*8, dimension(:,:), allocatable **mgt5op**
- real \*8, dimension(:,:), allocatable **mgt6op**
- real \*8, dimension(:,:), allocatable **mgt7op**
- real \*8, dimension(:,:), allocatable **mgt8op**
- real \*8, dimension(:,:), allocatable **mgt9op**
- real \*8, dimension(:,:), allocatable **mgt10iop**
- real \*8, dimension(:,:), allocatable **phu\_op**
- real \*8, dimension(:), allocatable **cnyld**
  - fraction of nitrogen in yield (kg N/kg yield)*
- real \*8, dimension(:), allocatable **rsdco\_pl**
  - plant residue decomposition coefficient. The fraction of residue which will decompose in a day assuming optimal moisture, temperature, C:N ratio, and C:P ratio (none)*
- real \*8, dimension(:), allocatable **wac21**
  - 1st shape parameter for radiation use efficiency equation (none)*
- real \*8, dimension(:), allocatable **wac22**
  - 2nd shape parameter for radiation use efficiency equation (none)*

- real \*8, dimension(:), allocatable [alai\\_min](#)  
*minimum LAI during winter dormant period ( $m^2/m^2$ )*
- real \*8, dimension(:), allocatable [leaf1](#)  
*1st shape parameter for leaf area development equation (none)*
- real \*8, dimension(:), allocatable [leaf2](#)  
*2nd shape parameter for leaf area development equation (none)*
- real \*8, dimension(:), allocatable [wsyf](#)  
*Value of harvest index between 0 and HVSTI which represents the lowest value expected due to water stress ((kg/ha)/(kg/ha))*
- real \*8, dimension(:), allocatable [bio\\_e](#)  
*biomass-energy ratio. The potential (unstressed) growth rate per unit of intercepted photosynthetically active radiation. ((kg/ha)/(MJ/m\*\*2))*
- real \*8, dimension(:), allocatable [hvsti](#)  
*harvest index: crop yield/aboveground biomass ((kg/ha)/(kg/ha))*
- real \*8, dimension(:), allocatable [t\\_base](#)  
*minimum temperature for plant growth (deg C)*
- real \*8, dimension(:), allocatable [t\\_opt](#)  
*optimal temperature for plant growth (deg C)*
- real \*8, dimension(:), allocatable [chtmx](#)  
*maximum canopy height (m)*
- real \*8, dimension(:), allocatable [cvm](#)  
*natural log of USLE\_C (none)*
- real \*8, dimension(:), allocatable [gsi](#)  
*maximum stomatal conductance (m/s)*
- real \*8, dimension(:), allocatable [vpd2](#)  
*rate of decline in stomatal conductance per unit increase in vapor pressure deficit ((m/s)\*(1/kPa))*
- real \*8, dimension(:), allocatable [wavp](#)  
*rate of decline in radiation use efficiency as a function of vapor pressure deficit (none)*
- real \*8, dimension(:), allocatable [bio\\_leaf](#)  
*fraction of leaf/needle biomass that drops during dormancy (for trees only) (none)*
- real \*8, dimension(:), allocatable [blai](#)  
*maximum (potential) leaf area index (none)*
- real \*8, dimension(:), allocatable [cpyld](#)  
*fraction of phosphorus in yield (kg P/kg yield)*
- real \*8, dimension(:), allocatable [dlai](#)  
*fraction of growing season when leaf area declines (none)*
- real \*8, dimension(:), allocatable [rdmx](#)  
*maximum root depth of plant (m)*
- real \*8, dimension(:), allocatable [bio\\_n1](#)  
*1st shape parameter for plant N uptake equation (none)*
- real \*8, dimension(:), allocatable [bio\\_n2](#)  
*2nd shape parameter for plant N uptake equation (none)*
- real \*8, dimension(:), allocatable [bio\\_p1](#)  
*1st shape parameter for plant P uptake equation (none)*
- real \*8, dimension(:), allocatable [bio\\_p2](#)  
*2st shape parameter for plant P uptake equation (none)*
- real \*8, dimension(:), allocatable [bm\\_dieoff](#)  
*fraction above ground biomass that dies off at dormancy (fraction)*
- real \*8, dimension(:), allocatable [bm\\_x\\_trees](#)
- real \*8, dimension(:), allocatable [ext\\_coef](#)
- real \*8, dimension(:), allocatable [rsr1](#)

- initial root to shoot ratio at the beg of growing season*
  - real \*8, dimension(:), allocatable **rsr2**
  - root to shoot ratio at the end of the growing season*
    - real \*8, dimension(:), allocatable **pltnfr1**
    - nitrogen uptake parameter #1: normal fraction of N in crop biomass at emergence (kg N/kg biomass)*
      - real \*8, dimension(:), allocatable **pltnfr2**
      - nitrogen uptake parameter #2: normal fraction of N in crop biomass at 0.5 maturity (kg N/kg biomass)*
        - real \*8, dimension(:), allocatable **pltnfr3**
        - nitrogen uptake parameter #3: normal fraction of N in crop biomass at maturity (kg N/kg biomass)*
          - real \*8, dimension(:), allocatable **pltpfr1**
          - phosphorus uptake parameter #1: normal fraction of P in crop biomass at emergence (kg P/kg biomass)*
            - real \*8, dimension(:), allocatable **pltpfr2**
            - phosphorus uptake parameter #2: normal fraction of P in crop biomass at 0.5 maturity (kg P/kg biomass)*
              - real \*8, dimension(:), allocatable **pltpfr3**
              - phosphorus uptake parameter #3: normal fraction of P in crop biomass at maturity (kg P/kg biomass)*
      - integer, dimension(:), allocatable **idc**
        - crop/landcover category:*
        - 1 warm season annual legume*
        - 2 cold season annual legume*
        - 3 perennial legume*
        - 4 warm season annual*
        - 5 cold season annual*
        - 6 perennial*
        - 7 trees*
      - integer, dimension(:), allocatable **mat\_yrs**
      - real \*8, dimension(:), allocatable **bactpdb**
        - concentration of persistent bacteria in manure (fertilizer) (cfu/g manure)*
      - real \*8, dimension(:), allocatable **fminn**
        - fraction of mineral N (NO<sub>3</sub> + NH<sub>3</sub>) (kg minN/kg fert)*
      - real \*8, dimension(:), allocatable **forgn**
        - fraction of organic N (kg orgN/kg fert)*
      - real \*8, dimension(:), allocatable **forgp**
        - fraction of organic P (kg orgP/kg fert)*
      - real \*8, dimension(:), allocatable **bactkddb**
        - bacteria partition coefficient (none):*
        - 1: all bacteria in solution*
        - 0: all bacteria sorbed to soil particles*
      - real \*8, dimension(:), allocatable **bactlpdb**
        - concentration of less persistent bacteria in manure (fertilizer) (cfu/g manure)*
      - real \*8, dimension(:), allocatable **fminp**
        - fraction of mineral P (kg minP/kg fert)*
      - real \*8, dimension(:), allocatable **fnh3n**
        - fraction of NH<sub>3</sub>-N in mineral N (kg NH<sub>3</sub>-N/kg minN)*
      - character(len=8), dimension(200) **fertnm**
        - name of fertilizer*
      - real \*8, dimension(:), allocatable **curbden**
        - curb length density in HRU (km/ha)*
      - real \*8, dimension(:), allocatable **dirtmx**
        - maximum amount of solids allowed to build up on impervious surfaces (kg/curb km)*
      - real \*8, dimension(:), allocatable **fimp**
        - fraction of HRU area that is impervious (both directly and indirectly connected)(fraction)*
      - real \*8, dimension(:), allocatable **urbcoef**

- wash-off coefficient for removal of constituents from an impervious surface (1/mm)*
- real \*8, dimension(:), allocatable **thalf**
  - time for the amount of solids on impervious areas to build up to 1/2 the maximum level (days)*
- real \*8, dimension(:), allocatable **tnconc**
  - concentration of total nitrogen in suspended solid load from impervious areas (mg N/kg sed)*
- real \*8, dimension(:), allocatable **tno3conc**
  - concentration of NO3-N in suspended solid load from impervious areas (mg NO3-N/kg sed)*
- real \*8, dimension(:), allocatable **tpconc**
  - concentration of total phosphorus in suspended solid load from impervious areas (mg P/kg sed)*
- real \*8, dimension(:), allocatable **fcimp**
  - fraction of HRU area that is classified as directly connected impervious (fraction)*
- real \*8, dimension(:), allocatable **urbcn2**
  - SCS curve number for moisture condition II in impervious areas (none)*
- real \*8 **fr\_curb**
  - availability factor, the fraction of the curb length that is sweepable (none)*
- real \*8 **frt\_kg**
  - amount of fertilizer applied to HRU (kg/ha)*
- real \*8 **pst\_dep**
  - depth of pesticide in the soil (mm)*
- real \*8 **swepeff**
- real \*8, dimension(:), allocatable **ranrns\_hru**
- integer, dimension(:), allocatable **itill**
- real \*8, dimension(:), allocatable **deptil**
  - depth of mixing caused by operation (mm)*
- real \*8, dimension(:), allocatable **effmix**
  - mixing efficiency of operation (none)*
- real \*8, dimension(:), allocatable **ranrns**
  - random roughness of a given tillage operation (mm)*
- character(len=8), dimension(550) **tillnm**
  - 8-character name for the tillage operation*
- real \*8, dimension(:), allocatable **rnum1s**
  - For ICODES equal to (none)*
  - 0,1,3,5,9: not used*
  - 2: Fraction of flow in channel*
  - 4: amount of water transferred (as defined by INUM4S)*
  - 7,8,10,11: drainage area in square kilometers associated with the record file.*
- real \*8, dimension(:), allocatable **hyd\_dakm**
  - total drainage area of hydrograph in square kilometers (km<sup>2</sup>)*
- real \*8, dimension(:,:), allocatable **varoute**
- real \*8, dimension(:,:), allocatable **shyd**
- real \*8, dimension(:,:), allocatable **vartran**
- real \*8, dimension(:,:), allocatable **hhvaroute**
- integer, dimension(:), allocatable **icodes**
  - routing command code (none):*
  - 0 = finish*
  - 1 = subbasin*
  - 2 = route*
  - 3 = routres*
  - 4 = transfer*
  - 5 = add*
  - 6 = rechour*
  - 7 = recmon*
  - 8 = recyear*
  - 9 = save*

10 = recday  
 11 = reccnst  
 12 = structure  
 13 = apex  
 14 = saveconc  
 15 =

- integer, dimension(:), allocatable **ihouts**  
 For ICODES equal to (none)  
 0: not used  
 1,2,3,5,7,8,10,11: hydrograph storage location number  
 4: departure type (1=reach, 2=reservoir)  
 9: hydrograph storage location of data to be printed to event file  
 14: hydrograph storage location of data to be printed to saveconc file.
- integer, dimension(:), allocatable **inum1s**  
 For ICODES equal to (none)  
 0: not used  
 1: subbasin number  
 2: reach number  
 3: reservoir number  
 4: reach or res # flow is diverted from  
 5: hydrograph storage location of 1st dataset to be added  
 7,8,9,10,11,14: file number.
- integer, dimension(:), allocatable **inum2s**  
 For ICODES equal to (none)  
 0,1,7,8,10,11: not used  
 2,3: inflow hydrograph storage location  
 4: destination type (1=reach, 2=reservoir)  
 5: hydrograph storage location of 2nd dataset to be added  
 9,14: print frequency (0=daily, 1=hourly)
- integer, dimension(:), allocatable **inum3s**  
 For ICODES equal to (none)  
 0,1,2,3,5,7,8,10,11: not used  
 4: destination number. Reach or reservoir receiving water  
 9: print format (0=normal, fixed format; 1=txt format for AV interface, recday)
- integer, dimension(:), allocatable **inum4s**  
 For ICODES equal to (none)  
 0,2,3,5,7,8,9,10,11: not used  
 1: GIS code printed to output file (optional)  
 4: rule code governing transfer of water (1=fraction transferred out, 2=min volume or flow left, 3=exact amount transferred)
- integer, dimension(:), allocatable **inum5s**
- integer, dimension(:), allocatable **inum6s**
- integer, dimension(:), allocatable **inum7s**
- integer, dimension(:), allocatable **inum8s**
- integer, dimension(:), allocatable **subed**
- character(len=10), dimension(:), allocatable **recmonps**
- character(len=10), dimension(:), allocatable **reccnstps**
- character(len=5), dimension(:), allocatable **subnum**
- character(len=4), dimension(:), allocatable **hruno**
- real \*8, dimension(:), allocatable **grwat\_n**  
 Mannings's n for grassed waterway (none)
- real \*8, dimension(:), allocatable **grwat\_i**  
 flag for the simulation of grass waterways (none)  
 = 0 inactive  
 = 1 active
- real \*8, dimension(:), allocatable **grwat\_l**  
 length of grass waterway (km)
- real \*8, dimension(:), allocatable **grwat\_w**

- *average width of grassed waterway (m)*  
real \*8, dimension(:), allocatable [grwat\\_d](#)
- *depth of grassed waterway from top of bank to bottom (m)*  
real \*8, dimension(:), allocatable [grwat\\_s](#)
- *average slope of grassed waterway channel (m)*  
real \*8, dimension(:), allocatable [grwat\\_spcon](#)
- *linear parameter for calculating sediment in grassed waterways (none)*  
real \*8, dimension(:), allocatable **tc\_gwat**
- *real \*8, dimension(:), allocatable **pot\_volmm***
- *real \*8, dimension(:), allocatable **pot\_tilemm***
- *real \*8, dimension(:), allocatable **pot\_volxmm***
- *real \*8, dimension(:), allocatable **pot\_fr***
- *fraction of HRU area that drains into pothole ( $\text{km}^2/\text{km}^2$ )*  
real \*8, dimension(:), allocatable [pot\\_tile](#)
- *average daily outflow to main channel from tile flow if drainage tiles are installed in pothole (needed only if current HRU is IPOT) ( $\text{m}^3/\text{s}$ )*  
real \*8, dimension(:), allocatable [pot\\_vol](#)
- *initial volume of water stored in the depression/impounded area (read in as mm and converted to  $\text{m}^3$ ) (needed only if current HRU is IPOT) (mm)*  
real \*8, dimension(:), allocatable **potsa**
- *real \*8, dimension(:), allocatable **pot\_volx***
- *maximum volume of water stored in the depression/impounded area (read in as mm and converted to  $\text{m}^3$ ) (needed only if current HRU is IPOT) (mm)*  
real \*8, dimension(:), allocatable [wfsh](#)
- *wetting front matric potential (mm)*  
real \*8, dimension(:), allocatable **potfliwi**
- *real \*8, dimension(:), allocatable **potsedi***
- *real \*8, dimension(:), allocatable **pot\_no3l***
- *nitrate decay rate in impounded area (1/day)*  
real \*8, dimension(:), allocatable [pot\\_nsed](#)
- *normal sediment concentration in impounded water (needed only if current HRU is IPOT)(mg/L)*  
real \*8, dimension(:), allocatable [gwno3](#)
- *nitrate-N concentration in groundwater loading to reach (mg N/L)*  
real \*8, dimension(:), allocatable **newrti**
- *real \*8, dimension(:), allocatable **fsred***
- *reduction in bacteria loading from filter strip (none)*  
real \*8, dimension(:), allocatable **pot\_sed**
- *real \*8, dimension(:), allocatable **pot\_no3***
- *real \*8, dimension(:), allocatable **tmpavp***
- *real \*8, dimension(:), allocatable **dis\_stream***
- *average distance to stream (m)*  
real \*8, dimension(:), allocatable [evpot](#)
- *pothole evaporation coefficient (none)*  
real \*8, dimension(:), allocatable **pot\_solpl**
- *real \*8, dimension(:), allocatable **sed\_con***
- *real \*8, dimension(:), allocatable **orgn\_con***
- *real \*8, dimension(:), allocatable **orgp\_con***
- *real \*8, dimension(:), allocatable **pot\_k***
- *hydraulic conductivity of soil surface of pothole defaults to conductivity of upper soil (0.↵ 01-10.) layer*  
real \*8, dimension(:), allocatable **soln\_con**
- *real \*8, dimension(:), allocatable **solp\_con***
- *real \*8, dimension(:), allocatable **n\_reduc***

- nitrogen uptake reduction factor (not currently used; defaulted 300.)*
- real \*8, dimension(:), allocatable **n\_lag**  
*lag coefficient for calculating nitrate concentration in subsurface drains (0.001 - 1.0) (dimensionless)*
- real \*8, dimension(:), allocatable **n\_ln**  
*power function exponent for calculating nitrate concentration in subsurface drains (1.0 - 3.0) (dimensionless)*
- real \*8, dimension(:), allocatable **n\_lnc**  
*coefficient for power function for calculating nitrate concentration in subsurface drains (0.5 - 4.0) (dimensionless)*
- integer, dimension(:), allocatable **ioper**
- integer, dimension(:), allocatable **ngrwat**
- real \*8, dimension(:), allocatable **usle\_ls**  
*USLE equation length slope (LS) factor (none)*
- real \*8, dimension(:), allocatable **filterw**  
*filter strip width for bacteria transport (m)*
- real \*8, dimension(:), allocatable **phuacc**  
*fraction of plant heat units accumulated continuous fertilization is initialized(none)*
- real \*8, dimension(:), allocatable **sumix**  
*sum of all tillage mixing efficiencies for HRU operation (none)*
- real \*8, dimension(:), allocatable **epco**  
*plant water uptake compensation factor (0-1) (none)*
- real \*8, dimension(:), allocatable **esco**  
*soil evaporation compensation factor (0-1) (none)*
- real \*8, dimension(:), allocatable **hru\_slp**  
*average slope steepness (m/m)*
- real \*8, dimension(:), allocatable **slsubbsn**  
*average slope length for subbasin (m)*
- real \*8, dimension(:), allocatable **erorgn**  
*organic N enrichment ratio, if left blank the model will calculate for every event (none)*
- real \*8, dimension(:), allocatable **erorgp**  
*organic P enrichment ratio, if left blank the model will calculate for every event (none)*
- real \*8, dimension(:), allocatable **biomix**  
*biological mixing efficiency. Mixing of soil due to activity of earthworms and other soil biota. Mixing is performed at the end of every calendar year (none)*
- real \*8, dimension(:), allocatable **pnd\_seci**
- real \*8, dimension(:), allocatable **canmx**  
*maximum canopy storage (mm H2O)*
- real \*8, dimension(:), allocatable **divmax**  
*maximum daily irrigation diversion from the reach (when IRRSC=1 or IRR=3): when value is positive the units are mm H2O; when the value is negative, the units are (10<sup>4</sup> m<sup>3</sup> H2O) (mm H2O or 10<sup>4</sup> m<sup>3</sup> H2O)*
- real \*8, dimension(:), allocatable **flowmin**  
*minimum instream flow for irrigation diversions when IRRSC=1, irrigation water will be diverted only when streamflow is at or above FLOWMIN (m<sup>3</sup>/s)*
- real \*8, dimension(:), allocatable **usle\_p**  
*USLE equation support practice (P) factor (none)*
- real \*8, dimension(:), allocatable **lat\_sed**  
*sediment concentration in lateral flow (g/L)*
- real \*8, dimension(:), allocatable **rch\_dakm**  
*total drainage area contributing to flow at the outlet (pour point) of the reach in square kilometers (km<sup>2</sup>)*
- real \*8, dimension(:), allocatable **pnd\_no3s**
- real \*8, dimension(:), allocatable **cn1**
- real \*8, dimension(:), allocatable **lat\_ttime**  
*lateral flow travel time or exponential of the lateral flow travel time (days or none)*
- real \*8, dimension(:), allocatable **cn2**

- SCS runoff curve number for moisture condition II (none)*

  - real \*8, dimension(:), allocatable [flowfr](#)
- fraction of available flow in reach that is allowed to be applied to the HRU (none)*

  - real \*8, dimension(:), allocatable [sol\\_zmx](#)
- maximum rooting depth (mm)*

  - real \*8, dimension(:), allocatable [tile\\_ttime](#)
- exponential of the tile flow travel time (none)*

  - real \*8, dimension(:), allocatable [slsoil](#)
- slope length for lateral subsurface flow (m)*

  - real \*8, dimension(:), allocatable [gwminp](#)
- soluble P concentration in groundwater loading to reach (mg P/L)*

  - real \*8, dimension(:), allocatable [sol\\_cov](#)
- amount of residue on soil surface (kg/ha)*

  - real \*8, dimension(:), allocatable [sed\\_stl](#)
- fraction of sediment remaining suspended in impoundment after settling for one day (kg/kg)*

  - real \*8, dimension(:), allocatable [ov\\_n](#)
- Manning's "n" value for overland flow (none)*

  - real \*8, dimension(:), allocatable [pnd\\_no3](#)
- amount of nitrate in pond (kg N)*

  - real \*8, dimension(:), allocatable [pnd\\_solp](#)
- amount of soluble P in pond (kg P)*

  - real \*8, dimension(:), allocatable [yldanu](#)
- annual yield (dry weight) in the HRU (metric tons/ha)*

  - real \*8, dimension(:), allocatable [driftco](#)
- coefficient for pesticide drift directly onto stream (none)*

  - real \*8, dimension(:), allocatable [pnd\\_orgn](#)
- amount of organic N in pond (kg N)*

  - real \*8, dimension(:), allocatable [pnd\\_orgp](#)
- amount of organic P in pond (kg P)*

  - real \*8, dimension(:), allocatable [cn3](#)
- amount of water held in soil profile at saturation (mm H<sub>2</sub>O)*

  - real \*8, dimension(:), allocatable [twlpnd](#)
- area of HRU in square kilometers (km<sup>2</sup>)*

  - real \*8, dimension(:), allocatable [twlwet](#)
- fraction of subbasin area contained in HRU (km<sup>2</sup>/km<sup>2</sup>)*

  - real \*8, dimension(:), allocatable [hru\\_fr](#)
- amount of water held in soil profile at saturation (mm H<sub>2</sub>O)*

  - real \*8, dimension(:), allocatable [sol\\_sumul](#)
- area of HRU in square kilometers (km<sup>2</sup>)*

  - real \*8, dimension(:), allocatable [pnd\\_chla](#)
- cover/crop biomass (kg/ha)*

  - real \*8, dimension(:), allocatable [hru\\_km](#)
- albedo when soil is moist (none)*

  - real \*8, dimension(:), allocatable [bio\\_ms](#)
- fraction of HRU/subbasin area that drains into ponds (none)*

  - real \*8, dimension(:), allocatable [sol\\_alb](#)
- hydraulic conductivity through bottom of ponds (mm/hr)*

  - real \*8, dimension(:), allocatable [strsw](#)
- surface area of ponds when filled to principal spillway (ha)*

  - real \*8, dimension(:), allocatable [pnd\\_fr](#)
- fraction of subbasin area contained in HRU (km<sup>2</sup>/km<sup>2</sup>)*

  - real \*8, dimension(:), allocatable [pnd\\_k](#)
- surface area of ponds when filled to principal spillway (ha)*

  - real \*8, dimension(:), allocatable [pnd\\_psa](#)
- fraction of subbasin area contained in HRU (km<sup>2</sup>/km<sup>2</sup>)*

  - real \*8, dimension(:), allocatable [pnd\\_pvol](#)



- runoff volume from catchment area needed to fill the ponds to the principal spillway (UNIT CHANGE!) ( $10^4 \text{ m}^3 \text{ H}_2\text{O}$  or  $\text{m}^3 \text{ H}_2\text{O}$ )*
- real \*8, dimension(:), allocatable [pnd\\_esa](#)  
*surface area of ponds when filled to emergency spillway (ha)*
- real \*8, dimension(:), allocatable [pnd\\_evol](#)  
*runoff volume from catchment area needed to fill the ponds to the emergency spillway (UNIT CHANGE!) ( $10^4 \text{ m}^3 \text{ H}_2\text{O}$  or  $\text{m}^3 \text{ H}_2\text{O}$ )*
- real \*8, dimension(:), allocatable [pnd\\_vol](#)  
*volume of water in ponds (UNIT CHANGE!) ( $10^4 \text{ m}^3 \text{ H}_2\text{O}$  or  $\text{m}^3 \text{ H}_2\text{O}$ )*
- real \*8, dimension(:), allocatable [yldaa](#)  
*average annual yield in the HRU (metric tons)*
- real \*8, dimension(:), allocatable [pnd\\_nsed](#)  
*normal sediment concentration in pond water (UNIT CHANGE!) (mg/kg or kg/kg)*
- real \*8, dimension(:), allocatable [pnd\\_sed](#)  
*sediment concentration in pond water (UNIT CHANGE!) (mg/kg or kg/kg)*
- real \*8, dimension(:), allocatable **strsa**
- real \*8, dimension(:), allocatable **dep\_imp**
- real \*8, dimension(:), allocatable **evpnd**
- real \*8, dimension(:), allocatable **evwet**
- real \*8, dimension(:), allocatable [wet\\_fr](#)  
*fraction of HRU/subbasin area that drains into wetlands (none)*
- real \*8, dimension(:), allocatable [wet\\_k](#)  
*hydraulic conductivity of bottom of wetlands (mm/hr)*
- real \*8, dimension(:), allocatable [wet\\_nsa](#)  
*surface area of wetlands in subbasin at normal water level (ha)*
- real \*8, dimension(:), allocatable [wet\\_nvol](#)  
*runoff volume from catchment area needed to fill wetlands to normal water level (UNIT CHANGE!) ( $10^4 \text{ m}^3 \text{ H}_2\text{O}$  or  $\text{m}^3 \text{ H}_2\text{O}$ )*
- integer, dimension(:), allocatable **iwetgw**
- integer, dimension(:), allocatable **iwetile**
- real \*8, dimension(:), allocatable [wet\\_mxsa](#)  
*surface area of wetlands at maximum water level (ha)*
- real \*8, dimension(:), allocatable [wet\\_mxvol](#)  
*runoff volume from catchment area needed to fill wetlands to maximum water level (UNIT CHANGE!) ( $10^4 \text{ m}^3 \text{ H}_2\text{O}$  or  $\text{m}^3 \text{ H}_2\text{O}$ )*
- real \*8, dimension(:), allocatable [wet\\_vol](#)  
*volume of water in wetlands (UNIT CHANGE!) ( $10^4 \text{ m}^3 \text{ H}_2\text{O}$  or  $\text{m}^3 \text{ H}_2\text{O}$ )*
- real \*8, dimension(:), allocatable [wet\\_nsed](#)  
*normal sediment concentration in wetland water (UNIT CHANGE!) (mg/kg or kg/kg)*
- real \*8, dimension(:), allocatable [wet\\_sed](#)  
*sediment concentration in wetland water (UNIT CHANGE!) (mg/L or kg/L)*
- real \*8, dimension(:), allocatable [bp1](#)  
*1st shape parameter for pond surface area equation (none)*
- real \*8, dimension(:), allocatable [bp2](#)  
*2nd shape parameter for the pond surface area equation (none)*
- real \*8, dimension(:), allocatable **smx**
- real \*8, dimension(:), allocatable **sci**
- real \*8, dimension(:), allocatable [bw1](#)  
*1st shape parameter for the wetland surface area equation (none)*
- real \*8, dimension(:), allocatable [bw2](#)  
*2nd shape parameter for the wetland surface area equation (none)*
- real \*8, dimension(:), allocatable **bactpq**

- real \*8, dimension(:), allocatable **bactp\_plt**
- real \*8, dimension(:), allocatable **bactlp\_plt**
- real \*8, dimension(:), allocatable **cnday**
- real \*8, dimension(:), allocatable **auto\_eff**  
*fertilizer application efficiency calculated as the amount of N applied divided by the amount of N removed at harvest (none)*
- real \*8, dimension(:), allocatable **secciw**  
*water clarity coefficient for wetland (none)*
- real \*8, dimension(:), allocatable **sol\_sw**  
*amount of water stored in soil profile on any given day (mm H2O)*
- real \*8, dimension(:), allocatable **bactlpq**
- real \*8, dimension(:), allocatable **chlaw**  
*chlorophyll-a production coefficient for wetland (none)*
- real \*8, dimension(:), allocatable **bactps**
- real \*8, dimension(:), allocatable **bactlps**
- real \*8, dimension(:), allocatable **tmpav**
- real \*8, dimension(:), allocatable **sno\_hru**  
*amount of water stored as snow (mm H2O)*
- real \*8, dimension(:), allocatable **wet\_orgn**  
*amount of organic N in wetland (kg N)*
- real \*8, dimension(:), allocatable **subp**
- real \*8, dimension(:), allocatable **hru\_ra**
- real \*8, dimension(:), allocatable **rsdin**  
*initial residue cover (kg/ha)*
- real \*8, dimension(:), allocatable **tmx**
- real \*8, dimension(:), allocatable **tmn**
- real \*8, dimension(:), allocatable **tmp\_hi**
- real \*8, dimension(:), allocatable **tmp\_lo**
- real \*8, dimension(:), allocatable **usle\_k**  
*USLE equation soil erodibility (K) factor (none)*
- real \*8, dimension(:), allocatable **tconc**  
*time of concentration for HRU (hour)*
- real \*8, dimension(:), allocatable **rwt**
- real \*8, dimension(:), allocatable **olai**
- real \*8, dimension(:), allocatable **hru\_rmx**
- real \*8, dimension(:), allocatable **usle\_cfac**
- real \*8, dimension(:), allocatable **usle\_eifac**
- real \*8, dimension(:), allocatable **sol\_sumfc**  
*amount of water held in soil profile at field capacity (mm H2O)*
- real \*8, dimension(:), allocatable **t\_ov**  
*time for flow from farthest point in subbasin to enter a channel (hour)*
- real \*8, dimension(:), allocatable **anano3**  
*total amount of NO3 applied during the year in auto-fertilization (kg N/ha)*
- real \*8, dimension(:), allocatable **aird**
- real \*8, dimension(:), allocatable **wet\_orgp**  
*amount of organic P in wetland (kg P)*
- real \*8, dimension(:), allocatable **sol\_avpor**  
*average porosity for entire soil profile (none)*
- real \*8, dimension(:), allocatable **usle\_mult**  
*product of USLE K,P,LS,exp(rock) (none)*
- real \*8, dimension(:), allocatable **aairr**
- real \*8, dimension(:), allocatable **cht**

- real \*8, dimension(:), allocatable **u10**
- real \*8, dimension(:), allocatable **rh**
- real \*8, dimension(:), allocatable **lai\_aamx**  
*maximum leaf area index for the entire period of simulation in the HRU (none)*
- real \*8, dimension(:), allocatable **shallirr**
- real \*8, dimension(:), allocatable **deepirr**
- real \*8, dimension(:), allocatable **ch\_l1**  
*longest tributary channel length in subbasin (km)*
- real \*8, dimension(:), allocatable **wet\_no3**  
*amount of nitrate in wetland (kg N)*
- real \*8, dimension(:), allocatable **canstor**
- real \*8, dimension(:), allocatable **ovrlnd**
- real \*8, dimension(:), allocatable **irr\_mx**  
*maximum irrigation amount per auto application (mm)*
- real \*8, dimension(:), allocatable **auto\_wstr**  
*water stress factor which triggers auto irrigation (none or mm)*
- real \*8, dimension(:), allocatable **cfrt\_id**  
*fertilizer/manure id number from database (none)*
- real \*8, dimension(:), allocatable **cfrt\_kg**  
*amount of fertilizer applied to HRU on a given day (kg/ha)*
- real \*8, dimension(:), allocatable **cpst\_id**
- real \*8, dimension(:), allocatable **cpst\_kg**
- real \*8, dimension(:), allocatable **irr\_asq**  
*surface runoff ratio*
- real \*8, dimension(:), allocatable **irr\_eff**
- real \*8, dimension(:), allocatable **irrsq**  
*surface runoff ratio (0-1) .1 is 10% surface runoff (frac)*
- real \*8, dimension(:), allocatable **irrefm**
- real \*8, dimension(:), allocatable **irrsalt**
- real \*8, dimension(:), allocatable **bio\_eat**  
*dry weight of biomass removed by grazing daily ((kg/ha)/day)*
- real \*8, dimension(:), allocatable **bio\_trmp**  
*dry weight of biomass removed by trampling daily ((kg/ha)/day)*
- integer, dimension(:), allocatable **ifrt\_freq**
- integer, dimension(:), allocatable **ipst\_freq**
- integer, dimension(:), allocatable **irr\_noa**
- integer, dimension(:), allocatable **irr\_sc**
- integer, dimension(:), allocatable **irr\_no**
- integer, dimension(:), allocatable **imp\_trig**  
*release/impound action code (none):*  
*0 begin impounding water*  
*1 release impounded water*
- integer, dimension(:), allocatable **fert\_days**
- integer, dimension(:), allocatable **irr\_sca**
- integer, dimension(:), allocatable **idplt**  
*land cover/crop identification code for first crop grown in HRU (the only crop if there is no rotation) (none)*
- integer, dimension(:), allocatable **pest\_days**
- integer, dimension(:), allocatable **wstrs\_id**
- real \*8, dimension(:,:), allocatable **bio\_aahv**
- real \*8, dimension(:), allocatable **cumei**
- real \*8, dimension(:), allocatable **cumeira**
- real \*8, dimension(:), allocatable **cumrt**
- real \*8, dimension(:), allocatable **cumrai**

- real \*8, dimension(:), allocatable [wet\\_solp](#)  
*amount of soluble P in wetland (kg P)*
- real \*8, dimension(:), allocatable **wet\_no3s**
- real \*8, dimension(:), allocatable **wet\_chla**
- real \*8, dimension(:), allocatable **wet\_seci**
- real \*8, dimension(:), allocatable **pnd\_no3g**
- real \*8, dimension(:), allocatable **pstsol**
- real \*8, dimension(:), allocatable [delay](#)  
*groundwater delay: time required for water leaving the bottom of the root zone to reach the shallow aquifer (days)*
- real \*8, dimension(:), allocatable [gwht](#)  
*groundwater height (m)*
- real \*8, dimension(:), allocatable **gw\_q**
- real \*8, dimension(:), allocatable **pnd\_solpg**
- real \*8, dimension(:), allocatable [alpha\\_bf](#)  
*alpha factor for groundwater recession curve (1/days)*
- real \*8, dimension(:), allocatable [alpha\\_bfe](#)  
 $\exp(-\alpha_{bf})$  (none)
- real \*8, dimension(:), allocatable [gw\\_spyld](#)  
*specific yield for shallow aquifer ( $m^3/m^3$ )*
- real \*8, dimension(:), allocatable [alpha\\_bf\\_d](#)  
*alpha factor for groundwater recession curve of the deep aquifer (1/days)*
- real \*8, dimension(:), allocatable [alpha\\_bfe\\_d](#)  
 $\exp(-\alpha_{bf_d})$  for deep aquifer (none)
- real \*8, dimension(:), allocatable **gw\_qdeep**
- real \*8, dimension(:), allocatable [gw\\_delaye](#)  
 $\exp(-1/delay)$  (none)
- real \*8, dimension(:), allocatable [gw\\_revap](#)  
*revap coeff: this variable controls the amount of water moving from the shallow aquifer to the root zone as a result of soil moisture depletion (none)*
- real \*8, dimension(:), allocatable [rchrg\\_dp](#)  
*recharge to deep aquifer: the fraction of root zone percolation that reaches the deep aquifer (none)*
- real \*8, dimension(:), allocatable [anion\\_excl](#)  
*fraction of porosity from which anions are excluded*
- real \*8, dimension(:), allocatable [revapmn](#)  
*threshold depth of water in shallow aquifer required to allow revap to occur (mm H2O)*
- real \*8, dimension(:), allocatable **rchrg**
- real \*8, dimension(:), allocatable [bio\\_min](#)  
*minimum plant biomass for grazing (kg/ha)*
- real \*8, dimension(:), allocatable [ffc](#)  
*initial HRU soil water content expressed as fraction of field capacity (none)*
- real \*8, dimension(:), allocatable **surqsolp**
- real \*8, dimension(:), allocatable [deepst](#)  
*depth of water in deep aquifer (mm H2O)*
- real \*8, dimension(:), allocatable [shallst](#)  
*depth of water in shallow aquifer (mm H2O)*
- real \*8, dimension(:), allocatable **cklsp**
- real \*8, dimension(:), allocatable **wet\_solpg**
- real \*8, dimension(:), allocatable **rchrg\_src**
- real \*8, dimension(:), allocatable [trapeff](#)  
*filter strip trapping efficiency (used for everything but bacteria) (none)*
- real \*8, dimension(:), allocatable [sol\\_avbd](#)  
*average bulk density for soil profile ( $Mg/m^3$ )*

- real \*8, dimension(:), allocatable **wet\_no3g**
- real \*8, dimension(:), allocatable **tdrain**  
*time to drain soil to field capacity yield used in autofertilization (hours)*
- real \*8, dimension(:), allocatable **gwqmn**  
*threshold depth of water in shallow aquifer required before groundwater flow will occur (mm H2O)*
- real \*8, dimension(:), allocatable **pplnt**
- real \*8, dimension(:), allocatable **snotmp**
- real \*8, dimension(:), allocatable **gdrain**  
*drain tile lag time: the amount of time between the transfer of water from the soil to the drain tile and the release of the water from the drain tile to the reach (hours)*
- real \*8, dimension(:), allocatable **ddrain**  
*depth to the sub-surface drain (mm)*
- real \*8, dimension(:), allocatable **sol\_crk**  
*crack volume potential of soil (none)*
- real \*8, dimension(:), allocatable **brt**  
*fraction of surface runoff within the subbasin which takes 1 day or less to reach the subbasin outlet (none)*
- real \*8, dimension(:), allocatable **dayl**
- real \*8, dimension(:), allocatable **sstmaxd**  
*static maximum depressional storage; read from .sdr (mm)*
- real \*8, dimension(:), allocatable **re**  
*effective radius of drains (mm)*
- real \*8, dimension(:), allocatable **sdrain**  
*distance between two drain tubes or tiles (mm)*
- real \*8, dimension(:), allocatable **ddrain\_hru**
- real \*8, dimension(:), allocatable **drain\_co**  
*drainage coefficient (mm/day)*
- real \*8, dimension(:), allocatable **latksatf**  
*multiplication factor to determine  $conk(j1,j)$  from  $sol\_k(j1,j)$  for HRU (none)*
- real \*8, dimension(:), allocatable **pc**  
*pump capacity (default pump capacity = 1.042mm/hr or 25mm/day) (mm/hr)*
- real \*8, dimension(:), allocatable **stmaxd**
- real \*8, dimension(:), allocatable **twash**
- real \*8, dimension(:), allocatable **rnd2**
- real \*8, dimension(:), allocatable **rnd3**
- real \*8, dimension(:), allocatable **sol\_cnsu**
- real \*8, dimension(:), allocatable **doxq**
- real \*8, dimension(:), allocatable **rnd8**
- real \*8, dimension(:), allocatable **rnd9**
- real \*8, dimension(:), allocatable **percn**
- real \*8, dimension(:), allocatable **sol\_sumwp**
- real \*8, dimension(:), allocatable **tauton**
- real \*8, dimension(:), allocatable **tautop**
- real \*8, dimension(:), allocatable **cbodu**
- real \*8, dimension(:), allocatable **chl\_a**
- real \*8, dimension(:), allocatable **qdr**
- real \*8, dimension(:), allocatable **tfertn**
- real \*8, dimension(:), allocatable **tfertp**
- real \*8, dimension(:), allocatable **tgrazn**
- real \*8, dimension(:), allocatable **tgrazp**
- real \*8, dimension(:), allocatable **latno3**
- real \*8, dimension(:), allocatable **latq**
- real \*8, dimension(:), allocatable **minpgw**
- real \*8, dimension(:), allocatable **no3gw**

- real \*8, dimension(:), allocatable **npInt**
- real \*8, dimension(:), allocatable **tileq**
- real \*8, dimension(:), allocatable **tileno3**
- real \*8, dimension(:), allocatable **sedminpa**
- real \*8, dimension(:), allocatable **sedminps**
- real \*8, dimension(:), allocatable **sedorgn**
- real \*8, dimension(:), allocatable **sedorgp**
- real \*8, dimension(:), allocatable **sedyld**
- real \*8, dimension(:), allocatable **sepbtm**
- real \*8, dimension(:), allocatable **strsn**
- real \*8, dimension(:), allocatable **strsp**
- real \*8, dimension(:), allocatable **strstmp**
- real \*8, dimension(:), allocatable **surfq**
- real \*8, dimension(:), allocatable **surqno3**
- real \*8, dimension(:), allocatable **hru\_ha**  
*area of HRU in hectares (ha)*
- real \*8, dimension(:), allocatable **hru\_dafr**  
*fraction of total watershed area contained in HRU (km2/km2)*
- real \*8, dimension(:), allocatable **tcfrtn**
- real \*8, dimension(:), allocatable **tcfrtp**
- real \*8, dimension(:), allocatable **drydep\_no3**  
*atmospheric dry deposition of nitrates (kg/ha/yr)*
- real \*8, dimension(:), allocatable **drydep\_nh4**  
*atmospheric dry deposition of ammonia (kg/ha/yr)*
- real \*8, dimension(:), allocatable **bio\_yrms**  
*annual biomass (dry weight) in the HRU (metric tons/ha)*
- real \*8, dimension(:), allocatable **phubase**  
*base zero total heat units (used when no land cover is growing) (heat units)*
- real \*8, dimension(:), allocatable **hvstiadj**
- real \*8, dimension(:), allocatable **laiday**  
*leaf area index ( $m^2/m^2$ )*
- real \*8, dimension(:), allocatable **chlap**  
*chlorophyll-a production coefficient for pond (none)*
- real \*8, dimension(:), allocatable **laimxfr**
- real \*8, dimension(:), allocatable **pnd\_psed**
- real \*8, dimension(:), allocatable **seccip**  
*water clarity coefficient for pond (none)*
- real \*8, dimension(:), allocatable **wet\_psed**
- real \*8, dimension(:), allocatable **plantn**
- real \*8, dimension(:), allocatable **plt\_et**
- real \*8, dimension(:), allocatable **bio\_aams**  
*average annual biomass in the HRU (metric tons)*
- real \*8, dimension(:), allocatable **plt\_pet**
- real \*8, dimension(:), allocatable **plantp**
- real \*8, dimension(:), allocatable **dormhr**  
*time threshold used to define dormant period for plant (when daylength is within the time specified by dl from the minimum daylength for the area, the plant will go dormant) (hour)*
- real \*8, dimension(:), allocatable **lai\_yrmx**  
*maximum leaf area index for the year in the HRU (none)*
- real \*8, dimension(:), allocatable **bio\_aamx**
- real \*8, dimension(:), allocatable **lat\_pst**
- real \*8, dimension(:), allocatable **fld\_fr**  
*fraction of HRU area that drains into floodplain ( $km^2/km^2$ )*

- real \*8, dimension(:), allocatable **orig\_snohru**
- real \*8, dimension(:), allocatable **orig\_potvol**
- real \*8, dimension(:), allocatable **orig\_alai**
- real \*8, dimension(:), allocatable **orig\_bioms**
- real \*8, dimension(:), allocatable **pltfr\_n**
- real \*8, dimension(:), allocatable **orig\_phuacc**
- real \*8, dimension(:), allocatable **orig\_sumix**
- real \*8, dimension(:), allocatable **pltfr\_p**
- real \*8, dimension(:), allocatable **phu\_plt**  
*total number of heat units to bring plant to maturity (heat units)*
- real \*8, dimension(:), allocatable **orig\_phu**
- real \*8, dimension(:), allocatable **orig\_shallst**
- real \*8, dimension(:), allocatable **orig\_deepst**
- real \*8, dimension(:), allocatable **rip\_fr**  
*fraction of HRU area that drains into riparian zone ( $\text{km}^2/\text{km}^2$ )*
- real \*8, dimension(:), allocatable **orig\_pndvol**
- real \*8, dimension(:), allocatable **orig\_pndsed**
- real \*8, dimension(:), allocatable **orig\_pndno3**
- real \*8, dimension(:), allocatable **orig\_pndsolp**
- real \*8, dimension(:), allocatable **orig\_pndorgn**
- real \*8, dimension(:), allocatable **orig\_pndorgp**
- real \*8, dimension(:), allocatable **orig\_wetvol**
- real \*8, dimension(:), allocatable **orig\_wetsed**
- real \*8, dimension(:), allocatable **orig\_wetno3**
- real \*8, dimension(:), allocatable **orig\_wetsolp**
- real \*8, dimension(:), allocatable **orig\_wetorgn**
- real \*8, dimension(:), allocatable **orig\_wetorgp**
- real \*8, dimension(:), allocatable **orig\_solcov**
- real \*8, dimension(:), allocatable **orig\_solsw**
- real \*8, dimension(:), allocatable **orig\_potno3**
- real \*8, dimension(:), allocatable **orig\_potsed**
- real \*8, dimension(:), allocatable **wtab**
- real \*8, dimension(:), allocatable **wtab\_mn**
- real \*8, dimension(:), allocatable **wtab\_mx**
- real \*8, dimension(:), allocatable **shallst\_n**  
*nitrate concentration in shallow aquifer converted to kg/ha (ppm NO3-N)*
- real \*8, dimension(:), allocatable **gw\_nloss**
- real \*8, dimension(:), allocatable **rchrg\_n**
- real \*8, dimension(:), allocatable **det\_san**
- real \*8, dimension(:), allocatable **det\_sil**
- real \*8, dimension(:), allocatable **det\_cla**
- real \*8, dimension(:), allocatable **det\_sag**
- real \*8, dimension(:), allocatable **det\_lag**
- real \*8, dimension(:), allocatable **afrt\_surface**  
*fraction of fertilizer which is applied to top 10 mm of soil (the remaining fraction is applied to first soil layer) (none)*
- real \*8, dimension(:), allocatable **tnylida**
- real \*8 **ftr\_surface**  
*fraction of fertilizer which is applied to the top 10 mm of soil (the remaining fraction is applied to the first soil layer) (none)*
- real \*8, dimension(:), allocatable **auto\_nyr**  
*maximum NO3-N content allowed to be applied in one year (kg NO3-N/ha)*
- real \*8, dimension(:), allocatable **auto\_napp**  
*maximum NO3-N content allowed in one fertilizer application (kg NO3-N/ha)*
- real \*8, dimension(:), allocatable **auto\_nstrs**



*nitrogen stress factor which triggers auto fertilization (none)*

- real \*8, dimension(:), allocatable **manure\_kg**
- real \*8, dimension(:,:), allocatable **rcn\_mo**
- real \*8, dimension(:,:), allocatable **rammo\_mo**
- real \*8, dimension(:,:), allocatable **drydep\_no3\_mo**
- real \*8, dimension(:,:), allocatable **drydep\_nh4\_mo**
- real \*8, dimension(:), allocatable **rcn\_d**
- real \*8, dimension(:), allocatable **rammo\_d**
- real \*8, dimension(:), allocatable **drydep\_no3\_d**
- real \*8, dimension(:), allocatable **drydep\_nh4\_d**
- real \*8, dimension(:,:), allocatable **yldn**
- real \*8, dimension(:,:), allocatable **gwati**
- real \*8, dimension(:,:), allocatable **gwatn**
- real \*8, dimension(:,:), allocatable **gwatl**
- real \*8, dimension(:,:), allocatable **gwatw**
- real \*8, dimension(:,:), allocatable **gwatd**
- real \*8, dimension(:,:), allocatable **gwatveg**
- real \*8, dimension(:,:), allocatable **gwata**
- real \*8, dimension(:,:), allocatable **gwats**
- real \*8, dimension(:,:), allocatable **gwatspcon**
- real \*8, dimension(:,:), allocatable **rfqeo\_30d**
- real \*8, dimension(:,:), allocatable **eo\_30d**
- real \*8, dimension(:), allocatable **psetlp1**

*phosphorus settling rate for 1st season (m/day)*

- real \*8, dimension(:), allocatable **psetlp2**

*phosphorus settling rate for 2nd season (m/day)*

- real \*8, dimension(:,:), allocatable **wgncur**
- real \*8, dimension(:,:), allocatable **wgnold**
- real \*8, dimension(:,:), allocatable **wrt**
- real \*8, dimension(:,:), allocatable **pst\_enr**

*pesticide enrichment ratio (none)*

- real \*8, dimension(:,:), allocatable **zdb**
- real \*8, dimension(:,:), allocatable **pst\_surq**
- real \*8, dimension(:,:), allocatable **plt\_pst**

*pesticide on plant foliage (kg/ha)*

- real \*8, dimension(:), allocatable **psetlw1**

*phosphorus settling rate for 1st season (m/day)*

- real \*8, dimension(:), allocatable **psetlw2**

*phosphorus settling rate for 2nd season (m/day)*

- real \*8, dimension(:,:), allocatable **pst\_sed**
- real \*8, dimension(:,:), allocatable **wupnd**

*average daily water removal from the pond for the month ( $10^4$  m<sup>3</sup>/day)*

- real \*8, dimension(:,:), allocatable **phi**

*phi(1,:) cross-sectional area of flow at bankfull depth (m<sup>2</sup>) phi(2,:) (none) phi(3,:) (none) phi(4,:) (none) phi(5,:) (none) phi(6,:) bottom width of main channel (m) phi(7,:) depth of water when reach is at bankfull depth (m) phi(8,:) average velocity when reach is at bankfull depth (m/s) phi(9,:) wave celerity when reach is at bankfull depth (m/s) phi(10,:) storage time constant for reach at bankfull depth (ratio of storage to discharge) (hour) phi(11,:) average velocity when reach is at 0.1 bankfull depth (low flow) (m/s) phi(12,:) wave celerity when reach is at 0.1 bankfull depth (low flow) (m/s) phi(13,:) storage time constant for reach at 0.1 bankfull depth (low flow) (ratio of storage to discharge) (hour)*

- real \*8, dimension(:,:), allocatable **pcpband**
- real \*8, dimension(:,:), allocatable **tavband**
- real \*8, dimension(:,:), allocatable **wat\_phi**
- real \*8, dimension(:,:), allocatable **snoeb**



- initial snow water content in elevation band (mm H2O)*
- real \*8, dimension(:,:), allocatable [wudeep](#)
- average daily water removal from the deep aquifer for the month ( $10^4 \text{ m}^3/\text{day}$ )*
- real \*8, dimension(:,:), allocatable [wushal](#)
- average daily water removal from the shallow aquifer for the month ( $10^4 \text{ m}^3/\text{day}$ )*
- real \*8, dimension(:,:), allocatable **tmnband**
- real \*8, dimension(:), allocatable **bss1**
- real \*8, dimension(:), allocatable **bss2**
- real \*8, dimension(:), allocatable **bss3**
- real \*8, dimension(:), allocatable **bss4**
- real \*8, dimension(:), allocatable [nsetlw1](#)
- nitrogen settling rate for 1st season (m/day)*
- real \*8, dimension(:), allocatable [nsetlw2](#)
- nitrogen settling rate for 2nd season (m/day)*
- real \*8, dimension(:,:), allocatable **snotmpeb**
- real \*8, dimension(:,:), allocatable **surf\_bs**
- real \*8, dimension(:), allocatable [nsetlp1](#)
- nitrogen settling rate for 1st season (m/day)*
- real \*8, dimension(:), allocatable [nsetlp2](#)
- nitrogen settling rate for 2nd season (m/day)*
- real \*8, dimension(:,:), allocatable **tmxband**
- real \*8, dimension(:,:), allocatable **rainsub**
- real \*8, dimension(:,:), allocatable **frad**
- real \*8, dimension(:), allocatable **rstopbsb**
- real \*8, dimension(:,:), allocatable **orig\_snoeb**
- real \*8, dimension(:,:), allocatable **orig\_pltpst**
- real \*8, dimension(:,:), allocatable **terr\_p**
- real \*8, dimension(:,:), allocatable **terr\_cn**
- real \*8, dimension(:,:), allocatable **terr\_sl**
- real \*8, dimension(:,:), allocatable **drain\_d**
- real \*8, dimension(:,:), allocatable **drain\_t**
- real \*8, dimension(:,:), allocatable **drain\_g**
- real \*8, dimension(:,:), allocatable **drain\_idep**
- real \*8, dimension(:,:), allocatable **cont\_cn**
- real \*8, dimension(:,:), allocatable **cont\_p**
- real \*8, dimension(:,:), allocatable **filt\_w**
- real \*8, dimension(:,:), allocatable **strip\_n**
- real \*8, dimension(:,:), allocatable **strip\_cn**
- real \*8, dimension(:,:), allocatable **strip\_c**
- real \*8, dimension(:,:), allocatable **strip\_p**
- real \*8, dimension(:,:), allocatable **fire\_cn**
- real \*8, dimension(:,:), allocatable **cropno\_upd**
- real \*8, dimension(:,:), allocatable **hi\_upd**
- real \*8, dimension(:,:), allocatable **laimx\_upd**
- real \*8, dimension(:,:), allocatable [phug](#)
- fraction of plant heat units at which grazing begins (none)*
- real \*8, dimension(:,:), allocatable **pst\_lag**
- integer, dimension(:), allocatable [hrupest](#)
- pesticide use flag (none)*
- 0: no pesticides used in HRU*
- 1: pesticides used in HRU*
- integer, dimension(:), allocatable [nrelease](#)
- sequence number of impound/release operation within the year (none)*

- integer, dimension(:), allocatable **swtrg**
- integer, dimension(:), allocatable **nrot**  
*number of years of rotation (none)*
- integer, dimension(:), allocatable **nfert**  
*sequence number of fertilizer application within the year (none)*
- integer, dimension(:), allocatable **nro**
- integer, dimension(:), allocatable **igro**  
*land cover status code (none). This code informs the model whether or not a land cover is growing at the beginning of the simulation*  
*0 no land cover growing*  
*1 land cover growing*
- integer, dimension(:), allocatable **ipnd1**  
*beginning month of nutrient settling season (none)*
- integer, dimension(:), allocatable **ipnd2**  
*ending month of nutrient settling season (none)*
- integer, dimension(:), allocatable **nair**  
*sequence number of auto-irrigation application within the year (none)*
- integer, dimension(:), allocatable **iflod1**  
*beginning month of non-flood season (none)*
- integer, dimension(:), allocatable **iflod2**  
*ending month of non-flood season (none)*
- integer, dimension(:), allocatable **ndtarg**  
*number of days required to reach target storage from current pond storage (none)*
- integer, dimension(:), allocatable **nirr**  
*sequence number of irrigation application within the year (none)*
- integer, dimension(:), allocatable **iaftrtyp**
- integer, dimension(:), allocatable **nstress**
- integer, dimension(:), allocatable **igrotree**
- integer, dimension(:), allocatable **grz\_days**
- integer, dimension(:), allocatable **nmgt**  
*management code (for GIS output only) (none)*
- integer, dimension(:), allocatable **nafert**  
*sequence number of auto-fert application within the year (none)*
- integer, dimension(:), allocatable **nsweep**  
*sequence number of street sweeping operation within the year (none)*
- integer, dimension(:), allocatable **icr**
- integer, dimension(:), allocatable **ncut**
- integer, dimension(:), allocatable **irrno**  
*irrigation source location (none)*  
*if IRRSC=1, IRRNO is the number of the reach*  
*if IRRSC=2, IRRNO is the number of the reservoir*  
*if IRRSC=3, IRRNO is the number of the subbasin*  
*if IRRSC=4, IRRNO is the number of the subbasin*  
*if IRRSC=5, not used*
- integer, dimension(:), allocatable **sol\_nly**  
*number of soil in soil profile layers (none)*
- integer, dimension(:), allocatable **irn**
- integer, dimension(:), allocatable **npcp**
- integer, dimension(:), allocatable **ncf**  
*sequence number of continuous fertilization operation within the year (none)*
- integer, dimension(:), allocatable **ngr**  
*sequence number of grazing operation within the year (none)*
- integer, dimension(:), allocatable **igrz**

- integer, dimension(:), allocatable **ndeat**
- integer, dimension(:), allocatable **hru\_sub**  
*subbasin in which HRU is located (none)*
- integer, dimension(:), allocatable **urblu**  
*urban land type identification number from urban.dat (none)*
- integer, dimension(:), allocatable **ldrain**  
*soil layer where drainage tile is located (none)*
- integer, dimension(:), allocatable **idorm**
- integer, dimension(:), allocatable **hru\_seq**
- integer, dimension(:), allocatable **iurban**  
*urban simulation code (none):*  
0 no urban sections in HRU  
1 urban sections in HRU, simulate using USGS regression equations  
2 urban sections in HRU, simulate using build up/wash off algorithm
- integer, dimension(:), allocatable **iday\_fert**
- integer, dimension(:), allocatable **icfrt**
- integer, dimension(:), allocatable **ifld**  
*number of HRU (in subbasin) that is a floodplain (none)*
- integer, dimension(:), allocatable **irip**  
*number of HRU (in subbasin) that is a riparian zone (none)*
- integer, dimension(:), allocatable **ndcfrt**
- integer, dimension(:), allocatable **hrugis**
- integer, dimension(:), allocatable **irjsc**  
*irrigation source code (none):*  
1 divert water from reach  
2 divert water from reservoir  
3 divert water from shallow aquifer  
4 divert water from deep aquifer  
5 divert water from source outside watershed
- integer, dimension(:), allocatable **orig\_igro**
- integer, dimension(:), allocatable **ntil**
- integer, dimension(:), allocatable **iwatable**
- integer, dimension(:), allocatable **curyr\_mat**
- integer, dimension(:), allocatable **ncpest**
- integer, dimension(:), allocatable **icpst**
- integer, dimension(:), allocatable **ndcpst**
- integer, dimension(:), allocatable **iday\_pest**
- integer, dimension(:), allocatable **irr\_flag**
- integer, dimension(:), allocatable **irra\_flag**
- integer, dimension(:,:), allocatable **rndseed**  
*random number generator seed. The seeds in the array are used to generate random numbers for the following purposes (none):*  
(1) wet/dry day probability  
(2) solar radiation  
(3) precipitation  
(4) USLE rainfall erosion index  
(5) wind speed  
(6) 0.5 hr rainfall fraction  
(7) relative humidity  
(8) maximum temperature  
(9) minimum temperature  
(10) generate new random numbers
- integer, dimension(:,:), allocatable **iterr**
- integer, dimension(:,:), allocatable **iyterr**
- integer, dimension(:,:), allocatable **itdrain**
- integer, dimension(:,:), allocatable **iydrain**

- integer, dimension(:,:), allocatable **ncrops**
- integer, dimension(:), allocatable **manure\_id**  
*manure (fertilizer) identification number from fert.dat (none)*
- integer, dimension(:,:), allocatable **mgt\_sdr**
- integer, dimension(:,:), allocatable **idplot**
- integer, dimension(:,:), allocatable **icont**
- integer, dimension(:,:), allocatable **iycont**
- integer, dimension(:,:), allocatable **ifilt**
- integer, dimension(:,:), allocatable **iyfilt**
- integer, dimension(:,:), allocatable **istrip**
- integer, dimension(:,:), allocatable **ystrip**
- integer, dimension(:,:), allocatable **iopday**
- integer, dimension(:,:), allocatable **iopyr**
- integer, dimension(:,:), allocatable **mgt\_ops**
- real \*8, dimension(:), allocatable **wshd\_pstap**
- real \*8, dimension(:), allocatable **wshd\_pstdg**
- integer, dimension(12) **ndmo**
- integer, dimension(:), allocatable **npno**  
*array of unique pesticides used in watershed (none)*
- integer, dimension(:), allocatable **mcrhru**
- character(len=13), dimension(18) **rfile**  
*rainfall file names (.pcp)*
- character(len=13), dimension(18) **tfile**  
*temperature file names (.tmp)*
- character(len=4), dimension(1000) **urbname**  
*name of urban land use*
- character(len=1), dimension(:), allocatable **kirr**  
*irrigation in HRU*
- character(len=1), dimension(:), allocatable **hydgrp**
- character(len=16), dimension(:), allocatable **snam**  
*soil series name*
- character(len=17), dimension(300) **pname**  
*name of pesticide/toxin*
- character(len=4), dimension(60) **title**  
*description lines in file.cio (1st 3 lines)*
- character(len=4), dimension(5000) **cpnm**  
*four character code to represent crop name*
- character(len=17), dimension(50) **fname**
- real \*8, dimension(:,:,:), allocatable **flomon**  
*average daily water loading for month ( $m^3/day$ )*
- real \*8, dimension(:,:,:), allocatable **solpstmon**  
*average daily soluble pesticide loading for month (mg pst/day)*
- real \*8, dimension(:,:,:), allocatable **srbpstmon**  
*average daily sorbed pesticide loading for month (mg pst/day)*
- real \*8, dimension(:,:,:), allocatable **orgnmon**  
*average daily organic N loading for month (kg N/day)*
- real \*8, dimension(:,:,:), allocatable **orgpmon**  
*average daily organic P loading for month (kg P/day)*
- real \*8, dimension(:,:,:), allocatable **sedmon**  
*average daily sediment loading for month (metric tons/day)*
- real \*8, dimension(:,:,:), allocatable **minpmon**  
*average daily mineral P loading for month (kg P/day)*

- real \*8, dimension(:,:,:), allocatable [nh3mon](#)  
*average amount of NH3-N loaded to stream on a given day in the month (kg N/day)*
- real \*8, dimension(:,:,:), allocatable [no3mon](#)  
*average daily NO3-N loading for month (kg N/day)*
- real \*8, dimension(:,:,:), allocatable [bactlpmon](#)  
*average amount of less persistent bacteria loaded to stream on a given day in the month (# bact/day)*
- real \*8, dimension(:,:,:), allocatable [bactpmon](#)  
*average amount of persistent bacteria loaded to stream on a given day in the month (# bact/day)*
- real \*8, dimension(:,:,:), allocatable [no2mon](#)  
*average amount of NO2-N loaded to stream on a given day in the month (kg N/day)*
- real \*8, dimension(:,:,:), allocatable [cmtl1mon](#)  
*average amount of conservative metal #1 loaded to stream on a given day in the month (# bact/day)*
- real \*8, dimension(:,:,:), allocatable [cmtl2mon](#)  
*average amount of conservative metal #2 loaded to stream on a given day in the month (# bact/day)*
- real \*8, dimension(:,:,:), allocatable [cmtl3mon](#)  
*average amount of conservative metal #3 loaded to stream on a given day in the month (# bact/day)*
- real \*8, dimension(:,:,:), allocatable [cbodmon](#)  
*average daily loading of CBOD in month (kg/day)*
- real \*8, dimension(:,:,:), allocatable [chlamon](#)  
*average daily loading of chlorophyll-a in month (kg/day)*
- real \*8, dimension(:,:,:), allocatable [disoxmon](#)  
*average daily loading of dissolved O2 in month (kg/day)*
- real \*8, dimension(:,:), allocatable [floyr](#)  
*average daily water loading for year ( $m^3/day$ )*
- real \*8, dimension(:,:), allocatable [orgnyr](#)  
*average daily organic N loading for year (kg N/day)*
- real \*8, dimension(:,:), allocatable [orgpyr](#)  
*average daily organic P loading for year (kg P/day)*
- real \*8, dimension(:,:), allocatable [sedyr](#)  
*average daily sediment loading for year (metric tons/day)*
- real \*8, dimension(:,:), allocatable [minpyr](#)  
*average daily mineral P loading for year (kg P/day)*
- real \*8, dimension(:,:), allocatable [nh3yr](#)  
*average daily NH3-N loading for year (kg N/day)*
- real \*8, dimension(:,:), allocatable [no2yr](#)  
*average daily NO2-N loading for year (kg N/day)*
- real \*8, dimension(:,:), allocatable [no3yr](#)  
*average daily NO3-N loading for year (kg N/day)*
- real \*8, dimension(:,:), allocatable [bactlpyr](#)  
*average daily loading of less persistent bacteria for year (# bact/day)*
- real \*8, dimension(:,:), allocatable [bactpyr](#)  
*average daily loading of persistent bacteria for year (# bact/day)*
- real \*8, dimension(:,:), allocatable [cmtl1yr](#)  
*average daily loading of conservative metal #1 for year (kg/day)*
- real \*8, dimension(:,:), allocatable [chlayr](#)  
*average daily loading of chlorophyll-a in year (kg/day)*
- real \*8, dimension(:,:), allocatable [cmtl2yr](#)  
*average daily loading of conservative metal #2 for year (kg/day)*
- real \*8, dimension(:,:), allocatable [cmtl3yr](#)  
*average daily loading of conservative metal #3 for year (kg/day)*
- real \*8, dimension(:,:), allocatable [cbodyr](#)

- average daily loading of CBOD in year (kg/day)*
- real \*8, dimension(:,:), allocatable [disoxyr](#)
  - average daily loading of dissolved O2 in year (kg/day)*
- real \*8, dimension(:,:), allocatable [solpstyr](#)
  - average daily soluble pesticide loading for year (mg pst/day)*
- real \*8, dimension(:,:), allocatable [srbpstyr](#)
  - average daily sorbed pesticide loading for year (mg pst/day)*
- real \*8, dimension(:,:), allocatable **sol\_mc**
- real \*8, dimension(:,:), allocatable **sol\_mn**
- real \*8, dimension(:,:), allocatable **sol\_mp**
- real \*8, dimension(:), allocatable **flocnst**
- real \*8, dimension(:), allocatable [orgncnst](#)
  - average daily organic N loading to reach (kg N/day)*
- real \*8, dimension(:), allocatable [sedcnst](#)
  - average daily sediment loading for reach (metric tons/day)*
- real \*8, dimension(:), allocatable [minpcnst](#)
  - average daily soluble P loading to reach (kg P/day)*
- real \*8, dimension(:), allocatable [no3cnst](#)
  - average daily nitrate loading to reach (kg N/day)*
- real \*8, dimension(:), allocatable [orgpcnst](#)
  - average daily organic P loading to reach (kg P/day)*
- real \*8, dimension(:), allocatable [bactpcnst](#)
  - average daily persistent bacteria loading to reach (# bact/day)*
- real \*8, dimension(:), allocatable [nh3cnst](#)
  - average daily ammonia loading to reach (kg N/day)*
- real \*8, dimension(:), allocatable [no2cnst](#)
  - average daily nitrite loading to reach (kg N/day)*
- real \*8, dimension(:), allocatable [bactlpcnst](#)
  - average daily less persistent bacteria loading to reach (# bact/day)*
- real \*8, dimension(:), allocatable [cmtl1cnst](#)
  - average daily conservative metal #1 loading (kg/day)*
- real \*8, dimension(:), allocatable [cmtl2cnst](#)
  - average daily conservative metal #2 loading (kg/day)*
- real \*8, dimension(:), allocatable [chlacnst](#)
  - average daily loading of chlorophyll-a (kg/day)*
- real \*8, dimension(:), allocatable [cmtl3cnst](#)
  - average daily conservative metal #3 loading (kg/day)*
- real \*8, dimension(:), allocatable [disoxcnst](#)
  - average daily loading of dissolved O2 (kg/day)*
- real \*8, dimension(:), allocatable [cbodcnst](#)
  - average daily loading of CBOD to reach (kg/day)*
- real \*8, dimension(:), allocatable [solpstcnst](#)
  - average daily soluble pesticide loading (mg/day)*
- real \*8, dimension(:), allocatable [srbpstcnst](#)
  - average daily sorbed pesticide loading (mg/day)*
- integer [nstep](#)
  - max number of time steps per day or number of lines of rainfall data for each day*
- integer [idt](#)
  - length of time step used to report precipitation data for sub-daily modeling (minutes)*
- real \*8, dimension(:), allocatable **hrtwtr**
- real \*8, dimension(:), allocatable **hhstor**

- real \*8, dimension(:), allocatable **hdepth**
- real \*8, dimension(:), allocatable **hsdti**
- real \*8, dimension(:), allocatable **hrchwtr**
- real \*8, dimension(:), allocatable **halgae**
- real \*8, dimension(:), allocatable **horgn**
- real \*8, dimension(:), allocatable **hnh4**
- real \*8, dimension(:), allocatable **hno2**
- real \*8, dimension(:), allocatable **hno3**
- real \*8, dimension(:), allocatable **horgp**
- real \*8, dimension(:), allocatable **hsolp**
- real \*8, dimension(:), allocatable **hbod**
- real \*8, dimension(:), allocatable **hdisox**
- real \*8, dimension(:), allocatable **hchla**
- real \*8, dimension(:), allocatable **hsedyld**
- real \*8, dimension(:), allocatable **hsedst**
- real \*8, dimension(:), allocatable **hharea**
- real \*8, dimension(:), allocatable **hsolpst**
- real \*8, dimension(:), allocatable **hsorpst**
- real \*8, dimension(:), allocatable **hhqday**
- real \*8, dimension(:), allocatable **precipdt**
- real \*8, dimension(:), allocatable **hhtime**
- real \*8, dimension(:), allocatable **hbactp**
- real \*8, dimension(:), allocatable **hbactlp**
- integer, dimension(10) **ivar\_orig**
- real \*8, dimension(10) **rvar\_orig**
- integer **nsave**  
*number of save commands in .fig file*
- integer **nauto**
- integer **iatmodep**
- real \*8, dimension(:), allocatable **wattemp**
- real \*8, dimension(:), allocatable **lkpst\_mass**
- real \*8, dimension(:), allocatable **lkspst\_mass**
- real \*8, dimension(:), allocatable **vel\_chan**
- real \*8, dimension(:), allocatable **vfscn**  
*fraction of the total runoff from the entire field entering the most concentrated 10% of the VFS (none)*
- real \*8, dimension(:), allocatable **vfsratio**  
*field area/VFS area ratio (none)*
- real \*8, dimension(:), allocatable **vfsch**  
*fraction of flow entering the most concentrated 10% of the VFS which is fully channelized (none)*
- real \*8, dimension(:), allocatable **vfsi**
- real \*8, dimension(:, :), allocatable **filter\_i**
- real \*8, dimension(:, :), allocatable **filter\_ratio**
- real \*8, dimension(:, :), allocatable **filter\_con**
- real \*8, dimension(:, :), allocatable **filter\_ch**
- real \*8, dimension(:, :), allocatable **sol\_n**
- integer **cswat**  
*= 0 Static soil carbon (old mineralization routines)*  
*= 1 C-FARM one carbon pool model*  
*= 2 Century model*
- real \*8, dimension(:, :), allocatable **sol\_bdp**
- real \*8, dimension(:, :), allocatable **tillagef**
- real \*8, dimension(:), allocatable **rtfr**
- real \*8, dimension(:), allocatable **stsol\_rd**
- integer **urban\_flag**

- integer **dorm\_flag**
- real \*8 **bf\_flg**
- real \*8 **iabstr**
- real \*8, dimension(:), allocatable **ubnrunoff**
- real \*8, dimension(:), allocatable **ubntss**
- real \*8, dimension(:,:), allocatable **sub\_ubnrunoff**
- real \*8, dimension(:,:), allocatable **sub\_ubntss**
- real \*8, dimension(:,:), allocatable **ovrlnd\_dt**
- real \*8, dimension(:,:), allocatable **hhsurf\_bs**
- integer **iuh**  
*unit hydrograph method: 1=triangular UH; 2=gamma funtion UH;*
- integer **sed\_ch**  
*channel routing for HOURLY; 0=Bagnold; 2=Brownlie; 3=Yang;*
- real \*8 **eros\_expo**  
*an exponent in the overland flow erosion equation ranges 1.5-3.0*
- real \*8 **eros\_spl**  
*coefficient of splash erosion varing 0.9-3.1*
- real \*8 **rill\_mult**  
*Multiplier to USLE\_K for soil susceptible to rill erosion, range 0.5-2.0.*
- real \*8 **sedprev**
- real \*8 **c\_factor**
- real \*8 **ch\_d50**  
*median particle diameter of channel bed (mm)*
- real \*8 **sig\_g**  
*geometric standard deviation of particle sizes for the main channel. Mean air temperature at which precipitation is equally likely to be rain as snow/freezing rain.*
- real \*8 **uhalpha**  
*alpha coefficient for estimating unit hydrograph using a gamma function (\*.bsn)*
- real \*8 **abstinit**
- real \*8 **abstmax**
- real \*8, dimension(:,:), allocatable **hhsedy**
- real \*8, dimension(:,:), allocatable **sub\_subp\_dt**
- real \*8, dimension(:,:), allocatable **sub\_hhsedy**
- real \*8, dimension(:,:), allocatable **sub\_atmp**
- real \*8, dimension(:), allocatable **rhy**
- real \*8, dimension(:), allocatable **init\_abstrc**
- real \*8, dimension(:), allocatable **dratio**
- real \*8, dimension(:), allocatable **hrtevp**
- real \*8, dimension(:), allocatable **hrttic**
- real \*8, dimension(:,:), allocatable **rchhr**
- real \*8, dimension(:), allocatable **hhresflwi**
- real \*8, dimension(:), allocatable **hhresflwo**
- real \*8, dimension(:), allocatable **hhressedi**
- real \*8, dimension(:), allocatable **hhressedo**
- character(len=4), dimension(:), allocatable **lu\_nodrain**
- integer, dimension(:), allocatable **bmpdrain**
- real \*8, dimension(:), allocatable **sub\_cn2**
- real \*8, dimension(:), allocatable **sub\_ha\_urb**
- real \*8, dimension(:), allocatable **bmp\_recharge**
- real \*8, dimension(:), allocatable **sub\_ha\_imp**
- real \*8, dimension(:), allocatable **subdr\_km**
- real \*8, dimension(:), allocatable **subdr\_ickm**
- real \*8, dimension(:,:), allocatable **sf\_im**



- real \*8, dimension(:,:), allocatable **sf\_iy**
- real \*8, dimension(:,:), allocatable **sp\_sa**
- real \*8, dimension(:,:), allocatable **sp\_pvol**
- real \*8, dimension(:,:), allocatable **sp\_pd**
- real \*8, dimension(:,:), allocatable **sp\_sedi**
- real \*8, dimension(:,:), allocatable **sp\_sede**
- real \*8, dimension(:,:), allocatable **ft\_sa**
- real \*8, dimension(:,:), allocatable **ft\_fsa**
- real \*8, dimension(:,:), allocatable **ft\_dep**
- real \*8, dimension(:,:), allocatable **ft\_h**
- real \*8, dimension(:,:), allocatable **ft\_pd**
- real \*8, dimension(:,:), allocatable **ft\_k**
- real \*8, dimension(:,:), allocatable **ft\_dp**
- real \*8, dimension(:,:), allocatable **ft\_dc**
- real \*8, dimension(:,:), allocatable **ft\_por**
- real \*8, dimension(:,:), allocatable **tss\_den**
- real \*8, dimension(:,:), allocatable **ft\_alp**
- real \*8, dimension(:,:), allocatable **sf\_fr**
- real \*8, dimension(:,:), allocatable **sp\_qi**
- real \*8, dimension(:,:), allocatable **sp\_k**
- real \*8, dimension(:,:), allocatable **ft\_qpnd**
- real \*8, dimension(:,:), allocatable **sp\_dp**
- real \*8, dimension(:,:), allocatable **ft\_qsw**
- real \*8, dimension(:,:), allocatable **ft\_qin**
- real \*8, dimension(:,:), allocatable **ft\_qout**
- real \*8, dimension(:,:), allocatable **ft\_sedpnd**
- real \*8, dimension(:,:), allocatable **sp\_bpw**
- real \*8, dimension(:,:), allocatable **ft\_bpw**
- real \*8, dimension(:,:), allocatable **ft\_sed\_cumul**
- real \*8, dimension(:,:), allocatable **sp\_sed\_cumul**
- integer, dimension(:), allocatable **num\_sf**
- integer, dimension(:,:), allocatable **sf\_typ**
- integer, dimension(:,:), allocatable **sf\_dim**
- integer, dimension(:,:), allocatable **ft\_qfg**
- integer, dimension(:,:), allocatable **sp\_qfg**
- integer, dimension(:,:), allocatable **sf\_ptp**
- integer, dimension(:,:), allocatable **ft\_fc**
- real \*8 **sfsedmean**
- real \*8 **sfsedstdev**
- integer, dimension(:), allocatable **dtp\_imo**  
*month the reservoir becomes operational (none)*
- integer, dimension(:), allocatable **dtp\_iyr**  
*year of the simulation that the reservoir becomes operational (none)*
- integer, dimension(:), allocatable **dtp\_numstage**  
*total number of stages in the weir (none)*
- integer, dimension(:), allocatable **dtp\_numweir**  
*total number of weirs in the BMP (none)*
- integer, dimension(:), allocatable **dtp\_onoff**  
*sub-basin detention pond is associated with (none)*
- integer, dimension(:), allocatable **dtp\_reltype**

- equations for stage-discharge relationship (none):*
    - 1=exponential function,
    - 2=linear,
    - 3=logarithmic,
    - 4=cubic,
    - 5=power
- integer, dimension(:), allocatable [dtp\\_stagdis](#)
  - (none):
  - 0=use weir/orifice discharge equation to calculate outflow,
  - 1=use stage-dicharge relationship
- integer, dimension(:), allocatable **dtp\_subnum**
- real \*8, dimension(:), allocatable [cf](#)
  - this parameter controls the response of decomposition to the combined effect of soil temperature and moisture.*
- real \*8, dimension(:), allocatable [cfh](#)
  - maximum humification rate*
- real \*8, dimension(:), allocatable [cfdec](#)
  - the undisturbed soil turnover rate under optimum soil water and temperature. Increasing it will increase carbon and organic N decomp.*
- real \*8, dimension(:), allocatable **lat\_orgn**
- real \*8, dimension(:), allocatable **lat\_orgp**
- integer, dimension(:, :), allocatable [dtp\\_weirdim](#)
  - weir dimensions (none),*
  - 1=read user input,
  - 0=use model calculation
- integer, dimension(:, :), allocatable [dtp\\_weirtype](#)
  - type of weir (none):*
  - 1=rectangular and
  - 2=circular
- real \*8, dimension(:), allocatable [dtp\\_coef1](#)
  - coefficient of 3rd degree in the polynomial equation (none)*
- real \*8, dimension(:), allocatable [dtp\\_coef2](#)
  - coefficient of 2nd degree in the polynomial equation (none)*
- real \*8, dimension(:), allocatable [dtp\\_coef3](#)
  - coefficient of 1st degree in the polynomial equation (none)*
- real \*8, dimension(:), allocatable [dtp\\_evrsv](#)
  - detention pond evaporation coefficient (none)*
- real \*8, dimension(:), allocatable [dtp\\_expont](#)
  - exponent used in the exponential equation (none)*
- real \*8, dimension(:), allocatable [dtp\\_intcept](#)
  - intercept used in regression equations (none)*
- real \*8, dimension(:), allocatable [dtp\\_lwratio](#)
  - ratio of length to width of water back up (none)*
- real \*8, dimension(:), allocatable [dtp\\_totwrwid](#)
  - total constructed width of the detention wall across the creek (m)*
- real \*8, dimension(:), allocatable **dtp\_inflvol**
- real \*8, dimension(:), allocatable **dtp\_wdep**
- real \*8, dimension(:), allocatable **dtp\_totdep**
- real \*8, dimension(:), allocatable **dtp\_watdepact**
- real \*8, dimension(:), allocatable **dtp\_outflow**
- real \*8, dimension(:), allocatable **dtp\_totrel**
- real \*8, dimension(:), allocatable **dtp\_backoff**
- real \*8, dimension(:), allocatable **dtp\_seep\_sa**
- real \*8, dimension(:), allocatable **dtp\_evap\_sa**
- real \*8, dimension(:), allocatable **dtp\_pet\_day**

- real \*8, dimension(:), allocatable **ntp\_pcpvol**
- real \*8, dimension(:), allocatable **ntp\_seepvol**
- real \*8, dimension(:), allocatable **ntp\_evapvol**
- real \*8, dimension(:), allocatable **ntp\_flowin**
- real \*8, dimension(:), allocatable **ntp\_backup\_length**
- real \*8, dimension(:), allocatable **ntp\_ivol**
- real \*8, dimension(:), allocatable **ntp\_ised**
- integer, dimension(:,:), allocatable **so\_res\_flag**
- integer, dimension(:,:), allocatable **ro\_bmp\_flag**
- real \*8, dimension(:,:), allocatable **sol\_watp**
- real \*8, dimension(:,:), allocatable **sol\_solp\_pre**
- real \*8, dimension(:,:), allocatable **psp\_store**
- real \*8, dimension(:,:), allocatable **ssp\_store**
- real \*8, dimension(:,:), allocatable **so\_res**
- real \*8, dimension(:,:), allocatable **sol\_cal**
- real \*8, dimension(:,:), allocatable **sol\_ph**
- integer **sol\_p\_model**
- integer, dimension(:,:), allocatable **a\_days**
- integer, dimension(:,:), allocatable **b\_days**
- real \*8, dimension(:), allocatable **harv\_min**
- real \*8, dimension(:), allocatable **fstap**
- real \*8, dimension(:), allocatable **min\_res**
- real \*8, dimension(:,:), allocatable **ro\_bmp\_flo**
- real \*8, dimension(:,:), allocatable **ro\_bmp\_sed**
- real \*8, dimension(:,:), allocatable **ro\_bmp\_bac**
- real \*8, dimension(:,:), allocatable **ro\_bmp\_pp**
- real \*8, dimension(:,:), allocatable **ro\_bmp\_sp**
- real \*8, dimension(:,:), allocatable **ro\_bmp\_pn**
- real \*8, dimension(:,:), allocatable **ro\_bmp\_sn**
- real \*8, dimension(:,:), allocatable **ro\_bmp\_flos**
- real \*8, dimension(:,:), allocatable **ro\_bmp\_seds**
- real \*8, dimension(:,:), allocatable **ro\_bmp\_bacs**
- real \*8, dimension(:,:), allocatable **ro\_bmp\_pps**
- real \*8, dimension(:,:), allocatable **ro\_bmp\_sps**
- real \*8, dimension(:,:), allocatable **ro\_bmp\_pns**
- real \*8, dimension(:,:), allocatable **ro\_bmp\_sns**
- real \*8, dimension(:,:), allocatable **ro\_bmp\_flot**
- real \*8, dimension(:,:), allocatable **ro\_bmp\_sedt**
- real \*8, dimension(:,:), allocatable **ro\_bmp\_bact**
- real \*8, dimension(:,:), allocatable **ro\_bmp\_ppt**
- real \*8, dimension(:,:), allocatable **ro\_bmp\_spt**
- real \*8, dimension(:,:), allocatable **ro\_bmp\_pnt**
- real \*8, dimension(:,:), allocatable **ro\_bmp\_snt**
- real \*8, dimension(:), allocatable **bmp\_flo**
- real \*8, dimension(:), allocatable **bmp\_sed**
- real \*8, dimension(:), allocatable **bmp\_bac**
- real \*8, dimension(:), allocatable **bmp\_pp**
- real \*8, dimension(:), allocatable **bmp\_sp**
- real \*8, dimension(:), allocatable **bmp\_pn**
- real \*8, dimension(:), allocatable **bmp\_sn**
- real \*8, dimension(:), allocatable **bmp\_flag**
- real \*8, dimension(:), allocatable **bmp\_flos**
- real \*8, dimension(:), allocatable **bmp\_seds**
- real \*8, dimension(:), allocatable **bmp\_bacs**
- real \*8, dimension(:), allocatable **bmp\_pps**

- real \*8, dimension(:), allocatable **bmp\_sps**
- real \*8, dimension(:), allocatable **bmp\_pns**
- real \*8, dimension(:), allocatable **bmp\_sns**
- real \*8, dimension(:), allocatable **bmp\_flot**
- real \*8, dimension(:), allocatable **bmp\_sedt**
- real \*8, dimension(:), allocatable **bmp\_bact**
- real \*8, dimension(:), allocatable **bmp\_ppt**
- real \*8, dimension(:), allocatable **bmp\_spt**
- real \*8, dimension(:), allocatable **bmp\_pnt**
- real \*8, dimension(:), allocatable **bmp\_snt**
- real \*8, dimension(:, :), allocatable **ntp\_addon**  
*the distance between spillway levels (m)*
- real \*8, dimension(:, :), allocatable **ntp\_cdis**  
*discharge coefficient for weir/orifice flow (none)*
- real \*8, dimension(:, :), allocatable **ntp\_depweir**  
*depth of rectangular weir at different stages (m)*
- real \*8, dimension(:, :), allocatable **ntp\_diaweir**  
*diameter of orifice hole at different stages (m)*
- real \*8, dimension(:, :), allocatable **ntp\_flowrate**  
*maximum discharge from each stage of the weir/hole ( $m^3/s$ )*
- real \*8, dimension(:, :), allocatable **ntp\_pcpret**  
*precipitation for different return periods (not used) (mm)*
- real \*8, dimension(:, :), allocatable **ntp\_retperd**  
*return period at different stages (years)*
- real \*8, dimension(:, :), allocatable **ntp\_wdratio**  
*width depth ratio of rectangular weirs (none)*
- real \*8, dimension(:, :), allocatable **ntp\_wrwid**
- real \*8, dimension(:), allocatable **ri\_subkm**
- real \*8, dimension(:), allocatable **ri\_totpv**
- real \*8, dimension(:), allocatable **irmmdt**
- real \*8, dimension(:, :), allocatable **ri\_sed**
- real \*8, dimension(:, :), allocatable **ri\_fr**
- real \*8, dimension(:, :), allocatable **ri\_dim**
- real \*8, dimension(:, :), allocatable **ri\_im**
- real \*8, dimension(:, :), allocatable **ri\_iy**
- real \*8, dimension(:, :), allocatable **ri\_sa**
- real \*8, dimension(:, :), allocatable **ri\_vol**
- real \*8, dimension(:, :), allocatable **ri\_qi**
- real \*8, dimension(:, :), allocatable **ri\_k**
- real \*8, dimension(:, :), allocatable **ri\_dd**
- real \*8, dimension(:, :), allocatable **ri\_evrsv**
- real \*8, dimension(:, :), allocatable **ri\_dep**
- real \*8, dimension(:, :), allocatable **ri\_ndt**
- real \*8, dimension(:, :), allocatable **ri\_pmpvol**
- real \*8, dimension(:, :), allocatable **ri\_sed\_cumul**
- real \*8, dimension(:, :), allocatable **hrnopcp**
- real \*8, dimension(:, :), allocatable **ri\_qloss**
- real \*8, dimension(:, :), allocatable **ri\_pumpv**
- real \*8, dimension(:, :), allocatable **ri\_sedi**
- character(len=4), dimension(:, :), allocatable **ri\_nirr**
- integer, dimension(:), allocatable **num\_ri**
- integer, dimension(:), allocatable **ri\_luflg**
- integer, dimension(:), allocatable **num\_noirr**

- integer, dimension(:), allocatable **wtp\_subnum**
- integer, dimension(:), allocatable **wtp\_onoff**
- integer, dimension(:), allocatable **wtp\_imo**
- integer, dimension(:), allocatable **wtp\_iyr**
- integer, dimension(:), allocatable **wtp\_dim**
- integer, dimension(:), allocatable **wtp\_stagdis**
- integer, dimension(:), allocatable **wtp\_sdtype**
- real \*8, dimension(:), allocatable **wtp\_pvol**
- real \*8, dimension(:), allocatable **wtp\_pdepth**
- real \*8, dimension(:), allocatable **wtp\_sdslope**
- real \*8, dimension(:), allocatable **wtp\_lenwidth**
- real \*8, dimension(:), allocatable **wtp\_extdepth**
- real \*8, dimension(:), allocatable **wtp\_hydeff**
- real \*8, dimension(:), allocatable **wtp\_evsv**
- real \*8, dimension(:), allocatable **wtp\_sdintc**
- real \*8, dimension(:), allocatable **wtp\_sdexp**
- real \*8, dimension(:), allocatable **wtp\_sdc1**
- real \*8, dimension(:), allocatable **wtp\_sdc2**
- real \*8, dimension(:), allocatable **wtp\_sdc3**
- real \*8, dimension(:), allocatable **wtp\_pdia**
- real \*8, dimension(:), allocatable **wtp\_plen**
- real \*8, dimension(:), allocatable **wtp\_pmann**
- real \*8, dimension(:), allocatable **wtp\_ploss**
- real \*8, dimension(:), allocatable **wtp\_k**
- real \*8, dimension(:), allocatable **wtp\_dp**
- real \*8, dimension(:), allocatable **wtp\_sedi**
- real \*8, dimension(:), allocatable **wtp\_sede**
- real \*8, dimension(:), allocatable **wtp\_qi**
- real \*8 **lai\_init**  
*initial leaf area index of transplants*
- real \*8 **bio\_init**  
*initial biomass of transplants (kg/ha)*
- real \*8 **cnop**  
*SCS runoff curve number for moisture condition II (none)*
- real \*8 **harveff**  
*harvest efficiency: fraction of harvested yield that is removed from HRU; the remainder becomes residue on the soil surface(none)*
- real \*8 **hi\_ovr**  
*harvest index target specified at harvest ((kg/ha)/(kg/ha))*
- real \*8 **frac\_harvk**
- real \*8 **lid\_vgcl**
- real \*8 **lid\_vgcm**
- real \*8 **lid\_qsurf\_total**
- real \*8 **lid\_farea\_sum**
- real \*8, dimension(:,:), allocatable **lid\_cuminf\_last**
- real \*8, dimension(:,:), allocatable **lid\_sw\_last**
- real \*8, dimension(:,:), allocatable **interval\_last**
- real \*8, dimension(:,:), allocatable **lid\_f\_last**
- real \*8, dimension(:,:), allocatable **lid\_cumr\_last**
- real \*8, dimension(:,:), allocatable **lid\_str\_last**
- real \*8, dimension(:,:), allocatable **lid\_farea**
- real \*8, dimension(:,:), allocatable **lid\_qsurf**
- real \*8, dimension(:,:), allocatable **lid\_sw\_add**
- real \*8, dimension(:,:), allocatable **lid\_cumqperc\_last**

- real \*8, dimension(:,:), allocatable **lid\_cumirr\_last**
- real \*8, dimension(:,:), allocatable **lid\_excum\_last**
- integer, dimension(:,:), allocatable **gr\_onoff**
- integer, dimension(:,:), allocatable **gr\_imo**
- integer, dimension(:,:), allocatable **gr\_iyr**
- real \*8, dimension(:,:), allocatable **gr\_farea**
- real \*8, dimension(:,:), allocatable **gr\_solop**
- real \*8, dimension(:,:), allocatable **gr\_etcoef**
- real \*8, dimension(:,:), allocatable **gr\_fc**
- real \*8, dimension(:,:), allocatable **gr\_wp**
- real \*8, dimension(:,:), allocatable **gr\_ksat**
- real \*8, dimension(:,:), allocatable **gr\_por**
- real \*8, dimension(:,:), allocatable **gr\_hydeff**
- real \*8, dimension(:,:), allocatable **gr\_soldpt**
- integer, dimension(:,:), allocatable **rg\_onoff**
- integer, dimension(:,:), allocatable **rg\_imo**
- integer, dimension(:,:), allocatable **rg\_iyr**
- real \*8, dimension(:,:), allocatable **rg\_farea**
- real \*8, dimension(:,:), allocatable **rg\_solop**
- real \*8, dimension(:,:), allocatable **rg\_etcoef**
- real \*8, dimension(:,:), allocatable **rg\_fc**
- real \*8, dimension(:,:), allocatable **rg\_wp**
- real \*8, dimension(:,:), allocatable **rg\_ksat**
- real \*8, dimension(:,:), allocatable **rg\_por**
- real \*8, dimension(:,:), allocatable **rg\_hydeff**
- real \*8, dimension(:,:), allocatable **rg\_soldpt**
- real \*8, dimension(:,:), allocatable **rg\_dimop**
- real \*8, dimension(:,:), allocatable **rg\_sarea**
- real \*8, dimension(:,:), allocatable **rg\_vol**
- real \*8, dimension(:,:), allocatable **rg\_sth**
- real \*8, dimension(:,:), allocatable **rg\_sdia**
- real \*8, dimension(:,:), allocatable **rg\_bdia**
- real \*8, dimension(:,:), allocatable **rg\_sts**
- real \*8, dimension(:,:), allocatable **rg\_orifice**
- real \*8, dimension(:,:), allocatable **rg\_oheight**
- real \*8, dimension(:,:), allocatable **rg\_odia**
- integer, dimension(:,:), allocatable **cs\_onoff**
- integer, dimension(:,:), allocatable **cs\_imo**
- integer, dimension(:,:), allocatable **cs\_iyr**
- integer, dimension(:,:), allocatable **cs\_grcon**
- real \*8, dimension(:,:), allocatable **cs\_farea**
- real \*8, dimension(:,:), allocatable **cs\_vol**
- real \*8, dimension(:,:), allocatable **cs\_rdepth**
- integer, dimension(:,:), allocatable **pv\_onoff**
- integer, dimension(:,:), allocatable **pv\_imo**
- integer, dimension(:,:), allocatable **pv\_iyr**
- integer, dimension(:,:), allocatable **pv\_solop**
- real \*8, dimension(:,:), allocatable **pv\_grvdep**
- real \*8, dimension(:,:), allocatable **pv\_grvpor**
- real \*8, dimension(:,:), allocatable **pv\_farea**
- real \*8, dimension(:,:), allocatable **pv\_drcoef**
- real \*8, dimension(:,:), allocatable **pv\_fc**
- real \*8, dimension(:,:), allocatable **pv\_wp**
- real \*8, dimension(:,:), allocatable **pv\_ksat**
- real \*8, dimension(:,:), allocatable **pv\_por**

- real \*8, dimension(:,:), allocatable **pv\_hydeff**
- real \*8, dimension(:,:), allocatable **pv\_soldpt**
- integer, dimension(:,:), allocatable **lid\_onoff**
- real \*8, dimension(:,:), allocatable **sol\_bmc**
- real \*8, dimension(:,:), allocatable **sol\_bmn**
- real \*8, dimension(:,:), allocatable **sol\_hsc**
- real \*8, dimension(:,:), allocatable **sol\_hsn**
- real \*8, dimension(:,:), allocatable **sol\_hpc**
- real \*8, dimension(:,:), allocatable **sol\_hpn**
- real \*8, dimension(:,:), allocatable **sol\_lm**
- real \*8, dimension(:,:), allocatable **sol\_lmc**
- real \*8, dimension(:,:), allocatable **sol\_lmn**
- real \*8, dimension(:,:), allocatable **sol\_ls**
- real \*8, dimension(:,:), allocatable **sol\_lsl**
- real \*8, dimension(:,:), allocatable **sol\_lsc**
- real \*8, dimension(:,:), allocatable **sol\_lsn**
- real \*8, dimension(:,:), allocatable **sol\_rnmn**
- real \*8, dimension(:,:), allocatable **sol\_lslc**
- real \*8, dimension(:,:), allocatable **sol\_lslnc**
- real \*8, dimension(:,:), allocatable **sol\_rspc**
- real \*8, dimension(:,:), allocatable **sol\_woc**
- real \*8, dimension(:,:), allocatable **sol\_won**
- real \*8, dimension(:,:), allocatable **sol\_hp**
- real \*8, dimension(:,:), allocatable **sol\_hs**
- real \*8, dimension(:,:), allocatable **sol\_bm**
- real \*8, dimension(:,:), allocatable **sol\_cac**
- real \*8, dimension(:,:), allocatable **sol\_cec**
- real \*8, dimension(:,:), allocatable **sol\_percc**
- real \*8, dimension(:,:), allocatable **sol\_latc**
- real \*8, dimension(:), allocatable **sedc\_d**
- real \*8, dimension(:), allocatable **surfqc\_d**
- real \*8, dimension(:), allocatable **latc\_d**
- real \*8, dimension(:), allocatable **percc\_d**
- real \*8, dimension(:), allocatable **foc\_d**
- real \*8, dimension(:), allocatable **nppc\_d**
- real \*8, dimension(:), allocatable **rsdc\_d**
- real \*8, dimension(:), allocatable **grainc\_d**
- real \*8, dimension(:), allocatable **stoverc\_d**
- real \*8, dimension(:), allocatable **soc\_d**
- real \*8, dimension(:), allocatable **rspc\_d**
- real \*8, dimension(:), allocatable **emitc\_d**
- real \*8, dimension(:), allocatable **sub\_sedc\_d**
- real \*8, dimension(:), allocatable **sub\_surfqc\_d**
- real \*8, dimension(:), allocatable **sub\_latc\_d**
- real \*8, dimension(:), allocatable **sub\_percc\_d**
- real \*8, dimension(:), allocatable **sub\_foc\_d**
- real \*8, dimension(:), allocatable **sub\_nppc\_d**
- real \*8, dimension(:), allocatable **sub\_rsdc\_d**
- real \*8, dimension(:), allocatable **sub\_grainc\_d**
- real \*8, dimension(:), allocatable **sub\_stoverc\_d**
- real \*8, dimension(:), allocatable **sub\_emitc\_d**
- real \*8, dimension(:), allocatable **sub\_soc\_d**
- real \*8, dimension(:), allocatable **sub\_rspc\_d**
- real \*8, dimension(:), allocatable **sedc\_m**
- real \*8, dimension(:), allocatable **surfqc\_m**

- real \*8, dimension(:), allocatable **latc\_m**
- real \*8, dimension(:), allocatable **percc\_m**
- real \*8, dimension(:), allocatable **foc\_m**
- real \*8, dimension(:), allocatable **nppc\_m**
- real \*8, dimension(:), allocatable **rsdc\_m**
- real \*8, dimension(:), allocatable **grainc\_m**
- real \*8, dimension(:), allocatable **stoverc\_m**
- real \*8, dimension(:), allocatable **emitc\_m**
- real \*8, dimension(:), allocatable **soc\_m**
- real \*8, dimension(:), allocatable **rspc\_m**
- real \*8, dimension(:), allocatable **sedc\_a**
- real \*8, dimension(:), allocatable **surfqc\_a**
- real \*8, dimension(:), allocatable **latc\_a**
- real \*8, dimension(:), allocatable **percc\_a**
- real \*8, dimension(:), allocatable **foc\_a**
- real \*8, dimension(:), allocatable **nppc\_a**
- real \*8, dimension(:), allocatable **rsdc\_a**
- real \*8, dimension(:), allocatable **grainc\_a**
- real \*8, dimension(:), allocatable **stoverc\_a**
- real \*8, dimension(:), allocatable **emitc\_a**
- real \*8, dimension(:), allocatable **soc\_a**
- real \*8, dimension(:), allocatable **rspc\_a**
- integer, dimension(:), allocatable **tillage\_switch**
- real \*8, dimension(:), allocatable **tillage\_depth**
- integer, dimension(:), allocatable **tillage\_days**
- real \*8, dimension(:), allocatable **tillage\_factor**
- real \*8 **dthy**  
*time interval for subdaily routing*
- integer, dimension(4) **ihx**
- integer, dimension(:), allocatable **nhy**
- real \*8, dimension(:), allocatable **rchx**
- real \*8, dimension(:), allocatable **rcss**
- real \*8, dimension(:), allocatable **qcap**
- real \*8, dimension(:), allocatable **chxa**
- real \*8, dimension(:), allocatable **chxp**
- real \*8, dimension(:, :, :), allocatable **qhy**
- real \*8 **ff1**
- real \*8 **ff2**

### 5.1.1 Detailed Description

main module containing the global variables

### 5.1.2 Variable Documentation



## 5.1.2.1 igropt

integer parm::igropt

Qual2E option for calculating the local specific growth rate of algae

1: multiplicative.

$$u = mumax fll fnn fpp$$

2: limiting nutrient

$$u = mumax fll \min(fnn, fpp)$$

3: harmonic mean

$$u = mumax fll \frac{2}{\frac{1}{fnn} + \frac{1}{fpp}}$$



## Chapter 6

# Data Type Documentation

### 6.1 `parm::ascrv` Interface Reference

#### Public Member Functions

- subroutine **ascrv** (`x1`, `x2`, `x3`, `x4`, `x5`, `x6`)

The documentation for this interface was generated from the following file:

- [modparm.f90](#)

### 6.2 `parm::atri` Interface Reference

#### Public Member Functions

- real \*8 function **atri** (`at1`, `at2`, `at3`, `at4i`)

The documentation for this interface was generated from the following file:

- [modparm.f90](#)

### 6.3 `parm::aunif` Interface Reference

#### Public Member Functions

- real \*8 function **aunif** (`x1`)

The documentation for this interface was generated from the following file:

- [modparm.f90](#)

## 6.4 parm::dstn1 Interface Reference

### Public Member Functions

- real \*8 function **dstn1** (rn1, rn2)

The documentation for this interface was generated from the following file:

- [modparm.f90](#)

## 6.5 parm::ee Interface Reference

### Public Member Functions

- real \*8 function **ee** (tk)

The documentation for this interface was generated from the following file:

- [modparm.f90](#)

## 6.6 parm::expo Interface Reference

### Public Member Functions

- real \*8 function **expo** (xx)

The documentation for this interface was generated from the following file:

- [modparm.f90](#)

## 6.7 parm::fcgd Interface Reference

### Public Member Functions

- real \*8 function **fcgd** (xx)

The documentation for this interface was generated from the following file:

- [modparm.f90](#)

## 6.8 parm::HQDAV Interface Reference

### Public Member Functions

- subroutine **hqdav** (A, CBW, QQ, SSS, ZCH, ZX, CHW, FPW, jrch)

The documentation for this interface was generated from the following file:

- [modparm.f90](#)

## 6.9 parm::layersplit Interface Reference

### Public Member Functions

- subroutine **layersplit** (dep\_new)

The documentation for this interface was generated from the following file:

- [modparm.f90](#)

## 6.10 parm::ndenit Interface Reference

### Public Member Functions

- subroutine **ndenit** (k, j, cdg, wdn, void)

The documentation for this interface was generated from the following file:

- [modparm.f90](#)

## 6.11 parm::qman Interface Reference

### Public Member Functions

- real \*8 function **qman** (x1, x2, x3, x4)

The documentation for this interface was generated from the following file:

- [modparm.f90](#)

## 6.12 parm::regres Interface Reference

### Public Member Functions

- real \*8 function **regres** (k)

The documentation for this interface was generated from the following file:

- [modparm.f90](#)

## 6.13 parm::rsedaa Interface Reference

### Public Member Functions

- subroutine **rsedaa** (years)

The documentation for this interface was generated from the following file:

- [modparm.f90](#)

## 6.14 parm::tair Interface Reference

### Public Member Functions

- real \*8 function **tair** (hr, jj)

The documentation for this interface was generated from the following file:

- [modparm.f90](#)

## 6.15 parm::theta Interface Reference

### Public Member Functions

- real \*8 function **theta** (r20, thk, tmp)

The documentation for this interface was generated from the following file:

- [modparm.f90](#)

## 6.16 parm::vbl Interface Reference

### Public Member Functions

- subroutine **vbl** (evx, spx, pp, qin, ox, vx1, vy, yi, yo, ysx, vf, vyf, aha)

The documentation for this interface was generated from the following file:

- [modparm.f90](#)

## Chapter 7

# File Documentation

### 7.1 allocate\_parms.f90 File Reference

#### Functions/Subroutines

- subroutine [allocate\\_parms](#)  
*this subroutine allocates array sizes*

#### 7.1.1 Detailed Description

file containing the subroutine allocate\_parms

Author

modified by Javier Burguete

### 7.2 ascrv.f90 File Reference

#### Functions/Subroutines

- subroutine [ascrv](#) (x1, x2, x3, x4, x5, x6)  
*this subroutine computes shape parameters x5 and x6 for the S curve equation*

#### 7.2.1 Detailed Description

file containing the subroutine ascrv

Author

modified by Javier Burguete

## 7.2.2 Function/Subroutine Documentation

### 7.2.2.1 ascrv()

```
subroutine ascrv (
    real*8, intent(in) x1,
    real*8, intent(in) x2,
    real*8, intent(in) x3,
    real*8, intent(in) x4,
    real*8, intent(out) x5,
    real*8, intent(out) x6 )
```

this subroutine computes shape parameters x5 and x6 for the S curve equation

$$x = \frac{y}{y + \exp(x5 + x6 y)}$$

given 2 (x,y) points along the curve. x5 is determined by solving the equation with  $x$  and  $y$  values measured around the midpoint of the curve (approx. 50% of the maximum value for  $x$ ) and x6 is determined by solving the equation with  $x$  and  $y$  values measured close to one of the endpoints of the curve (100% of the maximum value for  $x$ ). This subroutine is called from [readbsn.f90](#) and [readplant.f90](#)

#### Parameters

in	x1	value for x in the above equation for first datapoint, x1 should be close to 0.5 (the midpoint of the curve)
in	x2	value for x in the above equation for second datapoint, x2 should be close to 0.0 or 1.0
in	x3	value for y in the above equation corresponding to x1
in	x4	value for y in the above equation corresponding to x2
out	x5	1st shape parameter for S curve equation characterizing the midpoint of the curve
out	x6	2nd shape parameter for S curve equation characterizing the regions close to the endpoints of the curve

## 7.3 aunif.f90 File Reference

### Functions/Subroutines

- real \*8 function [aunif](#) (x1)

*This function generates random numbers ranging from 0.0 to 1.0. In the process of calculating the random number, the seed (x1) is set to a new value. This function implements the prime-modulus generator.*

### 7.3.1 Detailed Description

file containing the function aunif

#### Author

modified by Javier Burguete



## 7.3.2 Function/Subroutine Documentation

### 7.3.2.1 aunif()

```
real*8 function aunif (
    integer, intent(inout) x1 )
```

This function generates random numbers ranging from 0.0 to 1.0. In the process of calculating the random number, the seed (x1) is set to a new value. This function implements the prime-modulus generator.

$$x_i = 16807 x_i \bmod (2^{31} - 1)$$

using code which ensures that no intermediate result uses more than 31 bits. The theory behind the code is summarized in [1]

#### Parameters

<i>x1</i>	random number generator seed (integer) where $0 < x1 < 2147483647$
-----------	--

#### Returns

random number ranging from 0.0 to 1.0

## 7.4 caps.f90 File Reference

### Functions/Subroutines

- subroutine [caps](#) (file\_name)  
*this subroutine reads the input and output names given in file.cio and converts all capital letters to lowercase letters.*

### 7.4.1 Detailed Description

file containing the subroutine caps

#### Author

modified by Javier Burguete

## 7.4.2 Function/Subroutine Documentation

### 7.4.2.1 caps()

```
subroutine caps (
    character (len=*) file_name )
```

this subroutine reads the input and output names given in file.cio and converts all capital letters to lowercase letters.

## Parameters

<code>file_name</code>	dummy argument, file name character string
------------------------	--

## 7.5 estimate\_ksat.f90 File Reference

### Functions/Subroutines

- subroutine `estimate_ksat` (`perc_clay`, `esti_ksat`)

*This subroutine calculates ksat value for a soil layer given the % of clay in the soil layer.*

#### 7.5.1 Detailed Description

file containing the subroutine `estimate_ksat`

## Author

modified by Javier Burguete

#### 7.5.2 Function/Subroutine Documentation

##### 7.5.2.1 estimate\_ksat()

```
subroutine estimate_ksat (
    real*8, intent(in) perc_clay,
    real*8, intent(out) esti_ksat )
```

This subroutine calculates ksat value for a soil layer given the % of clay in the soil layer.

Background: published work of Walter Rawls. Calculated ksat values based on soil texture (sand, silt and clay).  
Idea: there exists a relationship between % clay and Ksat. Equations used in this subroutine are based on the above idea (Jimmy Willimas)

## Parameters

in	<code>perc_clay</code>	clay percentage (%)
out	<code>esti_ksat</code>	estimated ksat

## 7.6 gcycl.f90 File Reference

### Functions/Subroutines

- subroutine `gcycl`

*This subroutine initializes the random number seeds. If the user desires a different set of random numbers for each simulation run, the random number generator is used to reset the values of the seeds.*

### 7.6.1 Detailed Description

file containing the subroutine gcycl

Author

modified by Javier Burguete

## 7.7 getallo.f90 File Reference

### Functions/Subroutines

- subroutine [getallo](#)

*This subroutine calculates the number of HRUs, subbasins, etc. in the simulation. These values are used to allocate array sizes.*

### 7.7.1 Detailed Description

file containing the subroutine getallo

Author

modified by Javier Burguete

## 7.8 h2omgt\_init.f90 File Reference

### Functions/Subroutines

- subroutine [h2omgt\\_init](#)

*This subroutine initializes variables related to water management (irrigation, consumptive water use, etc.)*

### 7.8.1 Detailed Description

file containing the subroutine h2omgt\_init

Author

modified by Javier Burguete

## 7.9 headout.f90 File Reference

### Functions/Subroutines

- subroutine [headout](#)  
*this subroutine writes the headings to the major output files*

#### 7.9.1 Detailed Description

file containing the subroutine headout

Author

modified by Javier Burguete

## 7.10 hruallo.f90 File Reference

### Functions/Subroutines

- subroutine [hruallo](#)  
*This subroutine calculates the number of management operation types, etc. used in the simulation. These values are used to allocate array sizes for processes occurring in the HRU.*

#### 7.10.1 Detailed Description

file containing the subroutine hruallo

Author

modified by Javier Burguete

## 7.11 hydroinit.f90 File Reference

### Functions/Subroutines

- subroutine [hydroinit](#)  
*This subroutine computes variables related to the watershed hydrology: the time of concentration for the subbasins, lagged surface runoff, the coefficient for the peak runoff rate equation, and lateral flow travel time.*

#### 7.11.1 Detailed Description

file containing the subroutine hydroinit

Author

modified by Javier Burguete

## 7.12 impnd\_init.f90 File Reference

### Functions/Subroutines

- subroutine [impnd\\_init](#)

*this subroutine initializes variables related to impoundments (ponds, wetlands, reservoirs and potholes)*

#### 7.12.1 Detailed Description

file containing the subroutine impnd\_init

Author

modified by Javier Burguete

## 7.13 jdt.f90 File Reference

### Functions/Subroutines

- integer function [jdt](#) (numdays, i, m)

*this function computes the julian date given the month and the day of the month*

#### 7.13.1 Detailed Description

file containing the function jdt

Author

modified by Javier Burguete

#### 7.13.2 Function/Subroutine Documentation

##### 7.13.2.1 jdt()

```
integer function jdt (  
    integer, dimension (13), intent(in) numdays,  
    integer, intent(in) i,  
    integer, intent(in) m )
```

this function computes the julian date given the month and the day of the month

**Parameters**

in	<i>numdays</i>	julian date for last day of preceding month (where the array location is the number of the month). The dates are for leap years (numdays=ndays) (julian date)
in	<i>i</i>	day
in	<i>m</i>	month

## 7.14 lwqdef.f90 File Reference

### Functions/Subroutines

- subroutine [lwqdef](#)

*this subroutine assigns default values for the lake water quality (.lwq) when the lake water quality file does not exists*

#### 7.14.1 Detailed Description

file containing the subroutine lwqdef

**Author**

modified by Javier Burguete

## 7.15 main.f90 File Reference

### Functions/Subroutines

- program [main](#)

*this is the main program that reads input, calls the main simulation model, and writes output*

#### 7.15.1 Detailed Description

file containing the main program that reads input, calls the main simulation model, and writes output.

**Author**

modified by Javier Burguete Tolosa

## 7.16 modparm.f90 File Reference

### Data Types

- interface [parm::atri](#)
- interface [parm::aunif](#)
- interface [parm::dstn1](#)
- interface [parm::ee](#)
- interface [parm::expo](#)
- interface [parm::fcgd](#)
- interface [parm::qman](#)
- interface [parm::regres](#)
- interface [parm::tair](#)
- interface [parm::theta](#)
- interface [parm::ascrv](#)
- interface [parm::HQDAV](#)
- interface [parm::layersplit](#)
- interface [parm::ndenit](#)
- interface [parm::rsedaa](#)
- interface [parm::vbl](#)

### Modules

- module [parm](#)  
*main module containing the global variables*

### Variables

- integer, parameter [parm::mvar](#) = 33  
*max number of variables routed through the reach*
- integer, parameter [parm::mhruo](#) = 79  
*maximum number of variables written to HRU output file (output.hru) (none)*
- integer, parameter [parm::mrcho](#) = 62  
*maximum number of variables written to reach output file (.rch) (none)*
- integer, parameter [parm::msubo](#) = 24  
*maximum number of variables written to subbasin output file (output.sub) (none)*
- integer, parameter [parm::mstdo](#) = 113  
*max number of variables summarized in output.std*
- integer, parameter [parm::motot](#) = 600
- character(len=80), parameter [parm::prog](#) = "SWAT Sep 7 VER 2018/Rev 670"  
*SWAT program header string (name and version)*
- character(len=13), dimension(mhruo), parameter [parm::heds](#) = (/ "PRECIPmm", "SNOFALLmm", "SNOM←  
ELTmm", "IRRmm", "PETmm", "ETmm", "SW\_INITmm", "SW\_ENDmm", "PERCmm", "GW\_RCHGmm", "  
DA\_RCHGmm", "REVAPmm", "SA\_IRRmm", "DA\_IRRmm", "SA\_STmm", "DA\_STmm", "SURQ\_GE←  
Nmm", "SURQ\_CNTmm", "TLOSSmm", "LATQGENmm", "GW\_Qmm", "WYLDmm", "DAILYCN", "TMP←  
\_AVdgC", "TMP\_MXdgC", "TMP\_MNdgc", "SOL\_TMPdgC", "SOLARMJ/m2", "SYLDt/ha", "USLEt/ha", "N←  
APPkg/ha", "P\_APPkg/ha", "NAUTOkg/ha", "PAUTOkg/ha", "NGRZkg/ha", "PGRZkg/ha", "NCFRTkg/ha", "P←  
CFRTkg/ha", "NRAINkg/ha", "NFIKkg/ha", "F-MNkg/ha", "A-MNkg/ha", "A-SNkg/ha", "F-MPkg/ha", "AO-L←  
Pkg/ha", "L-APkg/ha", "A-SPkg/ha", "DNITkg/ha", "NUPkg/ha", "PUPkg/ha", "ORGNkg/ha", "ORGPkg/ha", "  
SEDPkg/ha", "NSURQkg/ha", "NLATQkg/ha", "NO3Lkg/ha", "NO3GWkg/ha", "SOLPkg/ha", "P\_GWkg/ha", "  
W\_STRS", "TMP\_STRS", "N\_STRS", "P\_STRS", "BIOMt/ha", "LAI", "YLDt/ha", "BACTPct", "BACTL←  
Pct", "WTAB CLIm", "WTAB SOLm", "SNOmm", "CMUPkg/ha", "CMTOTkg/ha", "QTILEmm", "TNO3kg/ha", "  
LNO3kg/ha", "GW\_Q\_Dmm", "LATQCNTmm", "TVAPkg/ha"/)





- nitrate percolation coefficient (0-1)*  
*0:concentration of nitrate in surface runoff is zero*  
*1:percolate has same concentration of nitrate as surface runoff*
- real \*8, dimension(:), allocatable **parm::surlag**  
*Surface runoff lag time. This parameter is needed in subbasins where the time of concentration is greater than 1 day. SURLAG is used to create a "storage" for surface runoff to allow the runoff to take longer than 1 day to reach the subbasin outlet (days)*
  - real \*8, dimension(:), allocatable **parm::co\_p**
  - real \*8, dimension(:), allocatable **parm::cmn**  
*rate factor for humus mineralization on active organic N*
  - real \*8, dimension(:), allocatable **parm::phoskd**  
*Phosphorus soil partitioning coefficient. Ratio of soluble phosphorus in surface layer to soluble phosphorus in runoff.*
  - real \*8, dimension(:), allocatable **parm::psp**  
*Phosphorus availability index. The fraction of fertilizer P remaining in labile pool after initial rapid phase of P sorption (none)*
  - real \*8, dimension(:), allocatable **parm::sdnco**  
*denitrification threshold: fraction of field capacity triggering denitrification*
  - real \*8 **parm::r2adj\_bsn**  
*basinwide retention parameter adjustment factor (greater than 1)*
  - real \*8 **parm::pst\_kg**  
*amount of pesticide applied to HRU (kg/ha)*
  - real \*8 **parm::yield**
  - real \*8 **parm::burn\_frlb**
  - real \*8 **parm::yieldgrn**
  - real \*8 **parm::yieldbms**
  - real \*8 **parm::yieldtbr**
  - real \*8 **parm::yieldn**
  - real \*8 **parm::yieldp**
  - real \*8 **parm::hi\_bms**
  - real \*8 **parm::hi\_rsd**
  - real \*8 **parm::yieldrsd**
  - real \*8, dimension(:), allocatable **parm::l\_k1**
  - real \*8, dimension(:), allocatable **parm::l\_k2**
  - real \*8, dimension(:), allocatable **parm::l\_lambda**
  - real \*8, dimension(:), allocatable **parm::l\_beta**
  - real \*8, dimension(:), allocatable **parm::l\_gama**
  - real \*8, dimension(:), allocatable **parm::l\_harea**
  - real \*8, dimension(:), allocatable **parm::l\_vleng**
  - real \*8, dimension(:), allocatable **parm::l\_vslope**
  - real \*8, dimension(:), allocatable **parm::l\_ktc**
  - real \*8, dimension(:), allocatable **parm::biofilm\_mumax**
  - real \*8, dimension(:), allocatable **parm::biofilm\_kinv**
  - real \*8, dimension(:), allocatable **parm::biofilm\_klw**
  - real \*8, dimension(:), allocatable **parm::biofilm\_kla**
  - real \*8, dimension(:), allocatable **parm::biofilm\_cdet**
  - real \*8, dimension(:), allocatable **parm::biofilm\_bm**
  - real \*8, dimension(:,:), allocatable **parm::hru\_rufr**
  - real \*8, dimension(:,:), allocatable **parm::daru\_km**
  - real \*8, dimension(:,:), allocatable **parm::ru\_k**
  - real \*8, dimension(:,:), allocatable **parm::ru\_c**
  - real \*8, dimension(:,:), allocatable **parm::ru\_eiq**
  - real \*8, dimension(:,:), allocatable **parm::ru\_ovsl**
  - real \*8, dimension(:,:), allocatable **parm::ru\_a**
  - real \*8, dimension(:,:), allocatable **parm::ru\_ovs**

- real \*8, dimension(:,:), allocatable **parm::ru\_ktc**
- real \*8, dimension(:), allocatable **parm::gwq\_ru**
- real \*8, dimension(:), allocatable **parm::qdayout**
- integer, dimension(:), allocatable **parm::ils2**
- integer, dimension(:), allocatable **parm::ils2flag**
- integer **parm::ipest**  
*pesticide identification number from pest.dat (none)*
- integer **parm::iru**
- integer **parm::mru**
- integer **parm::irch**
- integer **parm::isub**
- integer **parm::mhyd\_bsn**
- integer **parm::ils\_nofig**
- integer **parm::mhru1**
- integer, dimension(:), allocatable **parm::mhyd1**
- integer, dimension(:), allocatable **parm::irtun**
- real \*8 **parm::wshd\_sepno3**
- real \*8 **parm::wshd\_sepnh3**
- real \*8 **parm::wshd\_seporgn**
- real \*8 **parm::wshd\_sepfon**
- real \*8 **parm::wshd\_seporgp**
- real \*8 **parm::wshd\_sepfop**
- real \*8 **parm::wshd\_sepsolp**
- real \*8 **parm::wshd\_sepbod**
- real \*8 **parm::wshd\_sepmm**
- integer, dimension(:), allocatable **parm::isep\_hru**
- real \*8 **parm::fixco**  
*nitrogen fixation coefficient*
- real \*8 **parm::nfixmx**  
*maximum daily n-fixation (kg/ha)*
- real \*8 **parm::res\_stlr\_co**  
*reservoir sediment settling coefficient*
- real \*8 **parm::rsd\_covco**  
*residue cover factor for computing frac of cover*
- real \*8 **parm::vcrit**  
*critical velocity*
- real \*8 **parm::wshd\_snob**  
*average amount of water stored in snow at the beginning of the simulation for the entire watershed (mm H2O)*
- real \*8 **parm::wshd\_sw**  
*average amount of water stored in soil for the entire watershed (mm H2O)*
- real \*8 **parm::wshd\_pndfr**  
*fraction of watershed area which drains into ponds (none)*
- real \*8 **parm::wshd\_pndsed**  
*total amount of suspended sediment in ponds in the watershed (metric tons)*
- real \*8 **parm::wshd\_pndv**  
*total volume of water in ponds in the watershed ( $m^3$ )*
- real \*8 **parm::percop**  
*pesticide percolation coefficient (0-1)*  
*0: concentration of pesticide in surface runoff is zero*  
*1: percolate has same concentration of pesticide as surface runoff*
- real \*8 **parm::wshd\_resfr**  
*fraction of watershed area that drains into reservoirs (none)*
- real \*8 **parm::wshd\_pndha**

- watershed area in hectares which drains into ponds (ha)*
  - real \*8 `parm::wshd_resha`
  - watershed area in hectares which drains into reservoirs (ha)*
    - real \*8 `parm::wshd_wetfr`
    - fraction of watershed area which drains into wetlands (none)*
      - real \*8 `parm::wshd_fminp`
      - real \*8 `parm::wshd_ftotn`
      - real \*8 `parm::wshd_fnh3`
      - real \*8 `parm::wshd_fno3`
      - real \*8 `parm::wshd_forgn`
      - real \*8 `parm::wshd_forgp`
      - real \*8 `parm::wshd_ftotp`
      - real \*8 `parm::wshd_yldn`
      - real \*8 `parm::wshd_yldp`
      - real \*8 `parm::wshd_fixn`
      - real \*8 `parm::wshd_pup`
      - real \*8 `parm::wshd_wstrs`
      - real \*8 `parm::wshd_nstrs`
      - real \*8 `parm::wshd_pstrs`
      - real \*8 `parm::wshd_tstrs`
      - real \*8 `parm::wshd_astrs`
      - real \*8 `parm::ffcb`
      - initial soil water content expressed as a fraction of field capacity*
        - real \*8 `parm::wshd_hmn`
        - real \*8 `parm::wshd_rwn`
        - real \*8 `parm::wshd_hmp`
        - real \*8 `parm::wshd_rmn`
        - real \*8 `parm::wshd_dnit`
        - real \*8 `parm::wdpq`
        - die-off factor for persistent bacteria in soil solution (1/day)*
          - real \*8 `parm::wshd_rmp`
          - real \*8 `parm::wshd_voln`
          - real \*8 `parm::wshd_nitn`
          - real \*8 `parm::wshd_pas`
          - real \*8 `parm::wshd_pal`
          - real \*8 `parm::wof_p`
          - wash off fraction for persistent bacteria on foliage during a rainfall event*
            - real \*8 `parm::wshd_plch`
            - real \*8 `parm::wshd_raino3`
            - real \*8 `parm::ressedc`
            - real \*8 `parm::basno3f`
            - real \*8 `parm::basorgnf`
            - real \*8 `parm::wshd_pinlet`
            - real \*8 `parm::wshd_ptile`
            - real \*8 `parm::sftmp`
            - Snowfall temperature (deg C)*
              - real \*8 `parm::smfmn`
              - Minimum melt rate for snow during year (Dec. 21) where deg C refers to the air temperature. (mm/deg C/day)*
                - real \*8 `parm::smfmx`
                - Maximum melt rate for snow during year (June 21) where deg C refers to the air temperature. SMFMX and SM<sub>↔</sub>FMN allow the rate of snow melt to vary through the year. These parameters are accounting for the impact of soil temperature on snow melt. (mm/deg C/day)*
                  - real \*8 `parm::smtmp`

- Snow melt base temperature. Mean air temperature at which snow melt will occur. (deg C)*
- real \*8 [parm::wgppq](#)
  - growth factor for persistent bacteria in soil solution (1/day)*
- real \*8 **parm::basminpf**
- real \*8 **parm::basorgpf**
- real \*8 [parm::wdlpq](#)
  - die-off factor for less persistent bacteria in soil solution (1/day)*
- real \*8 [parm::wshd\\_ressed](#)
  - total amount of suspended sediment in reservoirs in the watershed (metric tons)*
- real \*8 [parm::wshd\\_resv](#)
  - total volume of water in all reservoirs in the watershed (m<sup>3</sup>)*
- real \*8 [parm::basminpi](#)
  - average amount of phosphorus initially in the mineral P pool in watershed soil (kg P/ha)*
- real \*8 [parm::basno3i](#)
  - average amount of nitrogen initially in the nitrate pool in watershed soil (kg N/ha)*
- real \*8 [parm::basorgni](#)
  - average amount of nitrogen initially in the organic N pool in watershed soil (kg N/ha)*
- real \*8 [parm::wdps](#)
  - die-off factor for persistent bacteria adsorbed to soil particles (1/day)*
- real \*8 [parm::wglpq](#)
  - growth factor for less persistent bacteria in soil solution (1/day)*
- real \*8 [parm::basorgpi](#)
  - average amount of phosphorus initially in the organic P pool in watershed soil (kg P/ha)*
- real \*8 **parm::peakr**
- real \*8 **parm::pndsedin**
- real \*8 **parm::sw\_excess**
- real \*8 **parm::albday**
- real \*8 [parm::timp](#)
  - Snow pack temperature lag factor (0-1)*
  - 1 = no lag (snow pack temp=current day air temp) as the lag factor goes to zero, the snow pack's temperature will be less influenced by the current day's air temperature.*
- real \*8 **parm::wtabelo**
- real \*8 **parm::tilep**
- real \*8 **parm::wt\_shall**
- real \*8 **parm::sq\_rto**
- real \*8 **parm::tloss**
- real \*8 **parm::inflpcp**
- real \*8 **parm::snomlt**
- real \*8 **parm::snofall**
- real \*8 **parm::fixn**
- real \*8 **parm::qtile**
- real \*8 **parm::crk**
- real \*8 **parm::latlyr**
- real \*8 **parm::pndloss**
- real \*8 **parm::wetloss**
- real \*8 **parm::potloss**
- real \*8 **parm::lpndloss**
- real \*8 **parm::lwetloss**
- real \*8 **parm::sedrch**
- real \*8 **parm::ftrtn**
- real \*8 **parm::sol\_rd**
- real \*8 **parm::cftrtn**
- real \*8 **parm::cftrtp**

- real \*8 **parm::sepday**
- real \*8 **parm::bioday**
- real \*8 **parm::sepcrk**
- real \*8 **parm::sepcrktot**
- real \*8 **parm::fertno3**
- real \*8 **parm::fertnh3**
- real \*8 **parm::fertorgn**
- real \*8 **parm::fertsolp**
- real \*8 **parm::fertorgp**
- real \*8 **parm::wgps**  
*growth factor for persistent bacteria adsorbed to soil particles (1/day)*
- real \*8 **parm::fertp**
- real \*8 **parm::grazn**
- real \*8 **parm::grazp**
- real \*8 **parm::soxy**
- real \*8 **parm::qdfr**
- real \*8 **parm::sdti**
- real \*8 **parm::rtwtr**
- real \*8 **parm::ressa**
- real \*8 **parm::wdlps**  
*die-off factor for less persistent bacteria adsorbed to soil particles (1/day)*
- real \*8 **parm::wglps**  
*growth factor for less persistent bacteria adsorbed to soil particles (1/day)*
- real \*8 **parm::da\_km**  
*area of the watershed in square kilometers (km<sup>2</sup>)*
- real \*8 **parm::rttime**
- real \*8 **parm::rchdep**
- real \*8 **parm::rtevp**
- real \*8 **parm::rttlc**
- real \*8 **parm::resflwi**
- real \*8 **parm::wdprch**  
*die-off factor for persistent bacteria in streams (1/day)*
- real \*8 **parm::resflwo**
- real \*8 **parm::respcp**
- real \*8 **parm::resev**
- real \*8 **parm::ressep**
- real \*8 **parm::ressedi**
- real \*8 **parm::ressedo**
- real \*8 **parm::dtot**
- real \*8 **parm::pperco\_bsn**  
*phosphorus percolation coefficient. Ratio of soluble phosphorus in surface to soluble phosphorus in percolate*
- real \*8 **parm::nperco\_bsn**  
*basin nitrate percolation coefficient (0-1)*  
*0:concentration of nitrate in surface runoff is zero*  
*1:percolate has same concentration of nitrate as surface runoff*
- real \*8 **parm::rsdco**  
*residue decomposition coefficient. The fraction of residue which will decompose in a day assuming optimal moisture, temperature, C:N ratio, and C:P ratio*
- real \*8 **parm::phoskd\_bsn**
- real \*8 **parm::voltot**
- real \*8 **parm::msk\_x**  
*weighting factor controlling relative importance of inflow rate and outflow rate in determining storage on reach*
- real \*8 **parm::volcrmin**
- real \*8 **parm::bactkdq**

*bacteria soil partitioning coefficient. Ratio of solution bacteria in surface layer to solution bacteria in runoff soluble and sorbed phase in surface runoff.*

- real \*8 **parm::wdpf**  
*die-off factor for persistent bacteria on foliage (1/day)*
- real \*8 **parm::uno3d**
- real \*8 **parm::canev**
- real \*8 **parm::usle**
- real \*8 **parm::rcn**
- real \*8 **parm::surlag\_bsn**
- real \*8 **parm::precipday**
- real \*8 **parm::thbact**  
*temperature adjustment factor for bacteria die-off/growth*
- real \*8 **parm::wlpq20**  
*overall rate change for less persistent bacteria in soil solution (1/day)*
- real \*8 **parm::wlps20**  
*overall rate change for less persistent bacteria adsorbed to soil particles (1/day)*
- real \*8 **parm::wpq20**  
*overall rate change for persistent bacteria in soil solution (1/day)*
- real \*8 **parm::wps20**  
*overall rate change for persistent bacteria adsorbed to soil particles (1/day)*
- real \*8 **parm::bactrop**
- real \*8 **parm::bactsedp**
- real \*8 **parm::wgpfi**  
*growth factor for persistent bacteria on foliage (1/day)*
- real \*8 **parm::bactlchp**
- real \*8 **parm::bactlchlp**
- real \*8 **parm::enratio**
- real \*8 **parm::wetpcp**
- real \*8 **parm::pndpcp**
- real \*8 **parm::wetsep**
- real \*8 **parm::pndsep**
- real \*8 **parm::wetev**
- real \*8 **parm::pndev**
- real \*8 **parm::pndsedo**
- real \*8 **parm::wetsedo**
- real \*8 **parm::pndflwi**
- real \*8 **parm::wetflwi**
- real \*8 **parm::da\_ha**  
*drainage area of watershed in hectares (ha)*
- real \*8 **parm::pndflwo**
- real \*8 **parm::wetflwo**
- real \*8 **parm::wetsedi**
- real \*8 **parm::vpd**
- real \*8 **parm::evlai**  
*leaf area index at which no evaporation occurs. This variable is used in ponded HRUs where evaporation from the water surface is restricted by the plant canopy cover. Evaporation from the water surface equals potential ET when LAI = 0 and decreased linearly to 0 when LAI = EVLAI*
- real \*8 **parm::evrch**  
*Reach evaporation adjustment factor. Evaporation from the reach is multiplied by EVRCH. This variable was created to limit the evaporation predicted in arid regions.*
- real \*8 **parm::wdlpi**  
*die-off factor for less persistent bacteria on foliage (1/day)*
- real \*8 **parm::bactrolp**

- real \*8 **parm::bactsedlp**
- real \*8 **parm::pet\_day**
- real \*8 **parm::ep\_day**
- real \*8 **parm::adj\_pkr**  
*peak rate adjustment factor in the subbasin. Used in the MUSLE equation to account for impact of peak flow on erosion (none)*
- real \*8 **parm::n\_updis**  
*nitrogen uptake distribution parameter. This parameter controls the amount of nitrogen removed from the different soil layer layers by the plant. In particular, this parameter allows the amount of nitrogen removed from the surface layer via plant uptake to be controlled. While the relationship between UBN and N removed from the surface layer is affected by the depth of the soil profile, in general, as UBN increases the amount of N removed from the surface layer relative to the amount removed from the entire profile increases*
- real \*8 **parm::nactfr**  
*nitrogen active pool fraction. The fraction of organic nitrogen in the active pool (none)*
- real \*8 **parm::p\_updis**  
*phosphorus uptake distribution parameter This parameter controls the amount of phosphorus removed from the different soil layers by the plant. In particular, this parameter allows the amount of phosphorus removed from the surface layer via plant uptake to be controlled. While the relationship between UBP and P uptake from the surface layer is affected by the depth of the soil profile, in general, as UBP increases the amount of P removed from the surface layer relative to the amount removed from the entire profile increases*
- real \*8 **parm::snoev**
- real \*8 **parm::sno3up**
- real \*8 **parm::reactw**
- real \*8 **parm::sdielopq**
- real \*8 **parm::sdielopq**
- real \*8 **parm::sdielopps**
- real \*8 **parm::sdielopps**
- real \*8 **parm::es\_day**
- real \*8 **parm::wof\_lp**  
*wash off fraction for less persistent bacteria on foliage during a rainfall event*
- real \*8 **parm::sbactrop**
- real \*8 **parm::sbactrolp**
- real \*8 **parm::sbactsedp**
- real \*8 **parm::sbactsedlp**
- real \*8 **parm::ep\_max**
- real \*8 **parm::sbactlchp**
- real \*8 **parm::sbactlchlp**
- real \*8 **parm::psp\_bsn**
- real \*8 **parm::rchwtr**
- real \*8 **parm::resuspst**
- real \*8 **parm::setlpst**
- real \*8 **parm::bsprev**
- real \*8 **parm::bssprev**
- real \*8 **parm::spadyo**
- real \*8 **parm::spadyev**
- real \*8 **parm::spadysp**
- real \*8 **parm::spadyrfv**
- real \*8 **parm::spadyosp**
- real \*8 **parm::qday**
- real \*8 **parm::usle\_ei**
- real \*8 **parm::al5**
- real \*8 **parm::pndsdc**
- real \*8 **parm::no3pcp**
- real \*8 **parm::rcharea**
- real \*8 **parm::volatpst**

- real \*8 **parm::ubw**  
*water uptake distribution parameter. This parameter controls the amount of water removed from the different soil layers by the plant. In particular, this parameter allows the amount of water removed from the surface layer via plant uptake to be controlled. While the relationship between UBW and H2O removed from the surface layer is affected by the depth of the soil profile, in general, as UBW increases the amount of water removed from the surface layer relative to the amount removed from the entire profile increases*
- real \*8 **parm::uobn**  
*nitrogen uptake normalization parameter. This variable normalizes the nitrogen uptake so that the model can easily verify that upake from the different soil layers sums to 1.0*
- real \*8 **parm::uobp**  
*phosphorus uptake normalization parameter. This variable normalizes the phosphorus uptake so that the model can easily verify that uptake from the different soil layers sums to 1.0*
- real \*8 **parm::uobw**  
*water uptake normalization parameter. This variable normalizes the water uptake so that the model can easily verify that uptake from the different soil layers sums to 1.0*
- real \*8 **parm::wglpf**  
*growth factor for less persistent bacteria on foliage (1/day)*
- real \*8 **parm::wetsedc**
- real \*8 **parm::respesti**
- real \*8 **parm::rcor**  
*correction coefficient for generated rainfall to ensure that the annual means for generated and observed values are comparable (needed only if IDIST=1)*
- real \*8 **parm::rexp**  
*value of exponent for mixed exponential rainfall distribution (needed only if IDIST=1)*
- real \*8 **parm::snocov1**  
*1st shape parameter for snow cover equation. This parameter is determined by solving the equation for 50% snow cover*
- real \*8 **parm::snocov2**  
*2nd shape parameter for snow cover equation. This parameter is determined by solving the equation for 95% snow cover*
- real \*8 **parm::snocovmx**  
*Minimum snow water content that corresponds to 100% snow cover. If the snow water content is less than SNOC<sub>←</sub> OVMX, then a certain percentage of the ground will be bare (mm H2O)*
- real \*8 **parm::lyrtile**
- real \*8 **parm::lyrtilex**
- real \*8 **parm::sno50cov**  
*Fraction of SNOC OVMX that corresponds to 50% snow cover. SWAT assumes a nonlinear relationship between snow water and snow cover.*
- real \*8 **parm::ai0**  
*ratio of chlorophyll-a to algal biomass (ug chla/mg alg)*
- real \*8 **parm::ai1**  
*fraction of algal biomass that is nitrogen (mg N/mg alg)*
- real \*8 **parm::ai2**  
*fraction of algal biomass that is phosphorus (mg P/mg alg)*
- real \*8 **parm::ai3**  
*the rate of oxygen production per unit of algal photosynthesis (mg O2/mg alg)*
- real \*8 **parm::ai4**  
*the rate of oxygen uptake per unit of algae respiration (mg O2/mg alg)*
- real \*8 **parm::ai5**  
*the rate of oxygen uptake per unit of NH3 nitrogen oxidation (mg O2/mg N)*
- real \*8 **parm::ai6**  
*the rate of oxygen uptake per unit of NO2 nitrogen oxidation (mg O2/mg N)*
- real \*8 **parm::rhoq**



- algal respiration rate (1/day or 1/hr)*
- real \*8 **parm::tfact**
  - fraction of solar radiation computed in the temperature heat balance that is photosynthetically active*
- real \*8 **parm::k\_l**
  - half-saturation coefficient for light ( $\text{MJ}/(\text{m}^2 \cdot \text{hr})$ )*
- real \*8 **parm::k\_n**
  - michaelis-menton half-saturation constant for nitrogen ( $\text{mg N/L}$ )*
- real \*8 **parm::k\_p**
  - michaelis-menton half saturation constant for phosphorus ( $\text{mg P/L}$ )*
- real \*8 **parm::lambda0**
  - non-algal portion of the light extinction coefficient ( $1/\text{m}$ )*
- real \*8 **parm::lambda1**
  - linear algal self-shading coefficient ( $1/(\text{m} \cdot \text{ug chl a/L})$ )*
- real \*8 **parm::lambda2**
  - nonlinear algal self-shading coefficient ( $((1/\text{m})(\text{ug chl a/L}))^{(-2/3)}$ )*
- real \*8 **parm::mumax**
  - maximum specific algal growth rate (1/day or 1/hr)*
- real \*8 **parm::p\_n**
  - algal preference factor for ammonia*
- real \*8 **parm::rnum1**
- real \*8 **parm::autop**
- real \*8 **parm::auton**
- real \*8 **parm::etday**
- real \*8 **parm::hmntl**
- real \*8 **parm::rwntl**
- real \*8 **parm::hmptl**
- real \*8 **parm::rmn2tl**
- real \*8 **parm::rmptl**
- real \*8 **parm::wdntl**
- real \*8 **parm::cmn\_bsn**
- real \*8 **parm::rmp1tl**
- real \*8 **parm::roctl**
- real \*8 **parm::gwseep**
- real \*8 **parm::revapday**
- real \*8 **parm::reswtr**
- real \*8 **parm::wdlprch**
  - die-off factor for less persistent bacteria in streams (1/day)*
- real \*8 **parm::wdpres**
  - die-off factor for persistent bacteria in reservoirs (1/day)*
- real \*8 **parm::bury**
- real \*8 **parm::difus**
- real \*8 **parm::reactb**
- real \*8 **parm::solpesto**
- real \*8 **parm::petmeas**
- real \*8 **parm::wdlpres**
  - die-off factor for less persistent bacteria in reservoirs (1/day)*
- real \*8 **parm::sorpesto**
- real \*8 **parm::spcon\_bsn**
- real \*8 **parm::spexp\_bsn**
- real \*8 **parm::solpesti**
- real \*8 **parm::sorpesti**
- real \*8 **parm::msk\_co1**

- calibration coefficient to control impact of the storage time constant for the reach at bankfull depth ( $\phi(10,:)$  upon the storage time constant for the reach used in the Muskingum flow method*

  - real \*8 **parm::msk\_co2**
- calibration coefficient to control impact of the storage time constant for the reach at 0.1 bankfull depth ( $\phi(13,:)$  upon the storage time constant for the reach used in the Muskingum flow method*

  - real \*8 **parm::snoprev**
  - real \*8 **parm::swprev**
  - real \*8 **parm::shallstp**
  - real \*8 **parm::deepstp**
  - real \*8 **parm::ressolpo**
  - real \*8 **parm::resorgno**
  - real \*8 **parm::resorgpo**
  - real \*8 **parm::resno3o**
  - real \*8 **parm::reschlao**
  - real \*8 **parm::resno2o**
  - real \*8 **parm::resnh3o**
  - real \*8 **parm::qdbank**
  - real \*8 **parm::potpcpmm**
  - real \*8 **parm::potevmm**
  - real \*8 **parm::potsepmm**
  - real \*8 **parm::potflwo**
  - real \*8 **parm::bactminlp**
- Threshold detection level for less persistent bacteria. When bacteria levels drop to this amount the model considers bacteria in the soil to be insignificant and sets the levels to zero ( $\text{cfu}/\text{m}^2$ )*

  - real \*8 **parm::bactminp**
- Threshold detection level for persistent bacteria. When bacteria levels drop to this amount the model considers bacteria in the soil to be insignificant and sets the levels to zero ( $\text{cfu}/\text{m}^2$ )*

  - real \*8 **parm::trnsrch**
- fraction of transmission losses from main channel that enter deep aquifer*

  - real \*8 **parm::wp20p\_plt**
- overall rate change for persistent bacteria on foliage (1/day)*

  - real \*8 **parm::potsedo**
- overall rate change for persistent bacteria on foliage (1/day)*

  - real \*8 **parm::pest\_sol**
- fraction of manure containing active colony forming units (cfu)*

  - real \*8 **parm::bact\_swf**
- fraction of manure containing active colony forming units (cfu)*

  - real \*8 **parm::bactmx**
- bacteria percolation coefficient. Ratio of solution bacteria in surface layer to solution bacteria in percolate*

  - real \*8 **parm::cncoef**
- plant ET curve number coefficient*

  - real \*8 **parm::wp20lp\_plt**
- overall rate change for less persistent bacteria on foliage (1/day)*

  - real \*8 **parm::cdn\_bsn**
- overall rate change for less persistent bacteria on foliage (1/day)*

  - real \*8 **parm::sdnco\_bsn**
- overall rate change for less persistent bacteria on foliage (1/day)*

  - real \*8 **parm::bactmin**
- overall rate change for less persistent bacteria on foliage (1/day)*

  - real \*8 **parm::cn\_froz**
- drainage coefficient ( $\text{mm day}^{-1}$ )*

  - real \*8 **parm::dorm\_hr**
- time threshold used to define dormant (hours)*

  - real \*8 **parm::smxco**
- adjustment factor for max curve number s factor (0-1)*

  - real \*8 **parm::tb\_adj**
- adjustment factor for subdaily unit hydrograph basetime*

  - real \*8 **parm::chla\_subco**

- regional adjustment on sub chla\_a loading (fraction)*
- real \*8 **parm::depimp\_bsn**
  - depth to impervious layer. Used to model perched water tables in all HRUs in watershed (mm)*
- real \*8 **parm::ddrain\_bsn**
  - depth to the sub-surface drain (mm)*
- real \*8 **parm::tdrain\_bsn**
  - time to drain soil to field capacity (hours)*
- real \*8 **parm::gdrain\_bsn**
- real \*8 **parm::rch\_san**
- real \*8 **parm::rch\_sil**
- real \*8 **parm::rch\_cla**
- real \*8 **parm::rch\_sag**
- real \*8 **parm::rch\_lag**
- real \*8 **parm::rch\_gra**
- real \*8 **parm::hlife\_ngw\_bsn**
  - Half-life of nitrogen in groundwater? (days)*
- real \*8 **parm::ch\_opco\_bsn**
- real \*8 **parm::ch\_onco\_bsn**
- real \*8 **parm::decr\_min**
  - Minimum daily residue decay.*
- real \*8 **parm::rcn\_sub\_bsn**
  - Concentration of nitrogen in the rainfall (mg/kg)*
- real \*8 **parm::bc1\_bsn**
- real \*8 **parm::bc2\_bsn**
- real \*8 **parm::bc3\_bsn**
- real \*8 **parm::bc4\_bsn**
- real \*8 **parm::anion\_excl\_bsn**
- real \*8, dimension(:), allocatable **parm::wat\_tbl**
- real \*8, dimension(:), allocatable **parm::sol\_swpwt**
- real \*8, dimension(:,:), allocatable **parm::vwt**
- real \*8 **parm::re\_bsn**
  - Effective radius of drains (range 3.0 - 40.0) (mm)*
- real \*8 **parm::sdrain\_bsn**
  - Distance bewtween two drain or tile tubes (range 7600.0 - 30000.0) (mm)*
- real \*8 **parm::sstmaxd\_bsn**
- real \*8 **parm::drain\_co\_bsn**
  - Drainage coefficient (range 10.0 - 51.0) (mm-day-1)*
- real \*8 **parm::latksatf\_bsn**
  - Multiplication factor to determine lateral ksat from SWAT ksat input value for HRU (range 0.01 - 4.0)*
- real \*8 **parm::pc\_bsn**
  - Pump capacity (def val = 1.042 mm h-1 or 25 mm day-1) (mm h-1)*
- integer **parm::i\_subhw**
- integer **parm::imgt**
- integer **parm::idlast**
- integer **parm::iwtr**
- integer **parm::ifrttyp**
- integer **parm::mo\_atmo**
- integer **parm::mo\_atmo1**
- integer **parm::ifirstatmo**
- integer **parm::iyr\_atmo**
- integer **parm::iyr\_atmo1**
- integer **parm::matmo**
- integer **parm::mch**

- maximum number of channels*
- integer `parm::mcr`
  - maximum number of crops grown per year*
- integer `parm::mcldb`
  - maximum number of crops/landcover in database file (crop.dat)*
- integer `parm::mfctst`
  - maximum number of forecast stations*
- integer `parm::mfdb`
  - max number of fertilizers in fert.dat*
- integer `parm::mhru`
  - maximum number of HRUs in watershed*
- integer `parm::mhyd`
  - maximum number of hydrograph nodes*
- integer `parm::mpdb`
  - max number of pesticides in pest.dat*
- integer `parm::mrg`
  - max number of rainfall/temp gages*
- integer `parm::mcut`
  - maximum number of cuttings per year*
- integer `parm::mgr`
  - maximum number of grazings per year*
- integer `parm::mnr`
  - max number of years of rotation*
- integer `parm::myr`
  - max number of years of simulation*
- integer `parm::isubwq`
  - subbasin water quality code*
  - 0 do not calculate algae/CBOD 1 calculate algae/CBOD drainmod tile equations*
- integer `parm::ffctst`
- integer `parm::isproj`
  - special project code (none):*
  - 1 test rewind (run simulation twice)*
- integer `parm::nbyr`
  - number of calendar years simulated (none)*
- integer `parm::irte`
  - water routing method (none):*
  - 0 variable storage method*
  - 1 Muskingum method*
- integer `parm::nrch`
  - number of reaches in watershed (none)*
- integer `parm::nres`
  - number of reservoirs in watershed (none)*
- integer `parm::nhru`
  - number of last HRU in previous subbasin (none)*
- integer `parm::i_mo`
  - current month being simulated (none)*
- integer `parm::mo`
- integer `parm::immo`
- integer `parm::wndsim`
  - wind speed input code*
  - 1 measured data read for each subbasin*
  - 2 data simulated for each subbasin*

- integer **parm::ihru**  
*HRU number (none)*
- integer **parm::icode**
- integer **parm::ihout**
- integer **parm::inum1**
- integer **parm::inum2**
- integer **parm::inum3**
- integer **parm::inum4**
- integer **parm::icfac**  
*icfac = 0 for C-factor calculation using Cmin (as described in manual)  
= 1 for new C-factor calculation from RUSLE (no minimum needed)*
- integer **parm::inum5**
- integer **parm::inum6**
- integer **parm::inum7**
- integer **parm::inum8**
- integer **parm::mrech**  
*maximum number of rehour files*
- integer **parm::nrgage**  
*number of raingage files (none)*
- integer **parm::nrgfil**  
*number of rain gages per file (none)*
- integer **parm::nrtot**  
*total number of rain gages (none)*
- integer **parm::ntgage**  
*number of temperature gage files (none)*
- integer **parm::ntgfil**  
*number of temperature gages per file (none)*
- integer **parm::nttot**  
*total number of temperature gages (none)*
- integer **parm::tmpsim**  
*temperature input code (none)  
1 measured data read for each subbasin  
2 data simulated for each subbasin*
- integer **parm::icrk**  
*crack flow code  
1: compute flow in cracks*
- integer **parm::irtpest**  
*number of pesticide to be routed through the watershed. Redefined to the sequence number of pesticide in NPNO(:)  
which is to be routed through the watershed (none)*
- integer **parm::igropt**  
*Qual2E option for calculating the local specific growth rate of algae  
1: multiplicative.*
- integer **parm::lao**  
*Qual2E light averaging option. Qual2E defines four light averaging options. The only option currently available in SWAT is #2.*
- integer **parm::npmx**  
*number of different pesticides used in the simulation (none)*
- integer **parm::curyr**  
*current year in simulation (sequence) (none)*
- integer **parm::iihru**
- integer **parm::itdrn**  
*tile drainage equations flag/code  
1 simulate tile flow using subroutine drains(wt\_shall)  
0 simulate tile flow using subroutine origtile(wt\_shall,d)*

- integer **parm::iwtdn**  
*water table depth algorithms flag/code*  
*1 simulate wt\_shall using subroutine new water table depth routine*  
*0 simulate wt\_shall using subroutine original water table depth routine*
- integer **parm::ismax**  
*maximum depressional storage selection flag/code*  
*0 = static depressional storage*  
*1 = dynamic storage based on tillage and cumulative rainfall*
- integer **parm::iroutunit**  
*not being implemented in this version drainmod tile equations*
- integer **parm::ires\_nut**
- integer **parm::iclb**  
*auto-calibration flag*
- integer **parm::mrecc**  
*maximum number of reccnst files*
- integer **parm::mrecd**  
*maximum number of recday files*
- integer **parm::mrecm**  
*maximum number of recmon files*
- integer **parm::mtil**  
*max number of tillage types in till.dat*
- integer **parm::mudb**  
*maximum number of urban land types in urban.dat*
- integer **parm::idist**  
*rainfall distribution code*  
*0 for skewed normal dist*  
*1 for mixed exponential distribution*
- integer **parm::mrecy**  
*maximum number of recyear files*
- integer **parm::nyskip**  
*number of years to not print output*
- integer **parm::slrsim**  
*solar radiation input code*  
*1 measured data read for each subbasin*  
*2 data simulated for each subbasin*
- integer **parm::ideg**  
*channel degradation code*  
*1: compute channel degradation (downcutting and widening)*
- integer **parm::ievent**  
*rainfall/runoff code*  
*0 daily rainfall/curve number technique 1 sub-daily rainfall/Green&Ampt/hourly routing 3 sub-daily rainfall/↵*  
*Green&Ampt/hourly routing*
- integer **parm::ipet**  
*code for potential ET method*  
*0 Priestley-Taylor method*  
*1 Penman/Monteith method*  
*2 Hargreaves method*  
*3 read in daily potential ET data*
- integer **parm::iopera**
- integer **parm::idaf**  
*beginning day of simulation (julian date)*
- integer **parm::idal**  
*ending day of simulation (julian date)*
- integer **parm::rhsim**

- relative humidity input code*
  - 1 measured data read for each subbasin*
  - 2 data simulated for each subbasin*
- integer **parm::leapyr**
  - leap year flag (none)*
  - 0 leap year*
  - 1 regular year*
- integer **parm::id1**
- integer **parm::mo\_chk**
- integer **parm::nhtot**
  - number of relative humidity records in file*
- integer **parm::nstot**
  - number of solar radiation records in file*
- integer **parm::nwtot**
  - number of wind speed records in file*
- integer **parm::ifirsts**
- integer **parm::ifirsth**
- integer **parm::ifirstw**
- integer **parm::icst**
- integer **parm::ilog**
  - streamflow print code*
- integer **parm::itotr**
  - number of output variables printed (output.rch)*
- integer **parm::iyr**
  - beginning year of simulation (year)*
- integer **parm::iwq**
  - stream water quality code*
  - 0 do not model stream water quality*
  - 1 model stream water quality (QUAL2E & pesticide transformations)*
- integer **parm::iskip**
  - flag for calculations performed only for the first year of simulation (none)*
- integer **parm::ifirstpet**
- integer **parm::iprp**
  - print code for output.pst file*
  - 0 do not print pesticide output*
  - 1 print pesticide output*
- integer **parm::itotb**
  - number of output variables printed (output.sub)*
- integer **parm::itots**
  - number of output variables printed (output.hru)*
- integer **parm::itoth**
  - number of HRUs printed (output.hru/output.wtr)*
- integer **parm::pcpsim**
  - rainfall input code*
  - 1 measured data read for each subbasin*
  - 2 data simulated for each subbasin*
- integer **parm::nd\_30**
- integer **parm::iops**
- integer **parm::iphr**
- integer **parm::isto**
- integer **parm::isol**
- integer **parm::fcstcycles**
  - number of times forecast period is simulated (using different weather generator seeds each time)*
- integer **parm::fcstday**

- beginning date of forecast period (julian date)*
- integer **parm::fcstyr**
  - beginning year of forecast period*
- integer **parm::iscen**
  - scenarios counter*
- integer **parm::subtot**
  - number of subbasins in watershed (none)*
- integer **parm::ogen**
- integer **parm::mapp**
  - maximum number of applications*
- integer **parm::mlyr**
  - maximum number of soil layers*
- integer **parm::mpst**
  - max number of pesticides used in wshed*
- integer **parm::mres**
  - maximum number of reservoirs*
- integer **parm::msub**
  - maximum number of subbasins*
- integer **parm::igen**
  - random number generator seed code (none):*
  - 0: use default numbers*
  - 1: generate new numbers in every simulation*
- integer **parm::iprint**
  - print code: 0=monthly, 1=daily, 2=annual*
- integer **parm::iida**
  - day being simulated (current julian day) (julian date)*
- integer **parm::icn**
  - CN method flag (for testing alternative method):*
  - 0 use traditional SWAT method which bases CN on soil moisture*
  - 1 use alternative method which bases CN on plant ET.*
- integer **parm::ised\_det**
  - max half-hour rainfall fraction calc option:*
  - 0 generate max half-hour rainfall fraction from triangular distribution*
  - 1 use monthly mean max half-hour rainfall fraction*
- integer **parm::fcstcnt**
- integer **parm::mtran**
- integer **parm::idtil**
- integer, dimension(100) **parm::ida\_lup**
- integer, dimension(100) **parm::iyr\_lup**
- integer **parm::no\_lup**
- integer **parm::no\_up**
- integer **parm::nostep**
- character(len=8) **parm::date**
  - date simulation is performed where leftmost eight characters are set to a value of yyyyymmdd, where yyyy is the year, mm is the month and dd is the day*
- character(len=10) **parm::time**
  - time simulation is performed where leftmost ten characters are set to a value of hhmmss.sss, where hh is the hour, mm is the minutes and ss.sss is the seconds and milliseconds*
- character(len=5) **parm::zone**
  - time difference with respect to Coordinated Universal Time (ie Greenwich Mean Time)*
- character(len=13) **parm::calfile**
  - name of file containing calibration parameters*
- character(len=13) **parm::rhfile**



- relative humidity file name (.hmd)*
- character(len=13) **parm::slrfile**
  - solar radiation file name (.slr)*
- character(len=13) **parm::wndfile**
  - wind speed file name (.wnd)*
- character(len=13) **parm::petfile**
  - potential ET file name (.pet)*
- character(len=13) **parm::atmofile**
- character(len=13) **parm::lucfile**
- character(len=13) **parm::septdb**
  - name of septic tank database file (septwq1.dat)*
- character(len=13) **parm::dpd\_file**
- character(len=13) **parm::wpd\_file**
- character(len=13) **parm::rib\_file**
- character(len=13) **parm::sfb\_file**
- character(len=13) **parm::lid\_file**
- integer, dimension(9) **parm::idg**
  - array location of random number seed used for a given process*
- integer, dimension(:), allocatable **parm::ifistr**
- integer, dimension(:), allocatable **parm::ifirsthr**
- integer, dimension(8) **parm::values**
  - values(1): year simulation is performed*
  - values(2): month simulation is performed*
  - values(3): day in month simulation is performed*
  - values(4): time difference with respect to Coordinated Universal Time (ie Greenwich Mean Time)*
  - values(5): hour simulation is performed*
  - values(6): minute simulation is performed*
  - values(7): second simulation is performed*
  - values(8): millisecond simulation is performed*
- integer, dimension(13) **parm::ndays**
  - julian date for last day of preceding month (where the array location is the number of the month). The dates are for leap years (julian date)*
- integer, dimension(13) **parm::ndays\_noleap**
- integer, dimension(13) **parm::ndays\_leap**
- integer **parm::mapex**
- real \*8, dimension(:), allocatable **parm::flodaya**
- real \*8, dimension(:), allocatable **parm::seddaya**
- real \*8, dimension(:), allocatable **parm::orgndaya**
- real \*8, dimension(:), allocatable **parm::orgpdaya**
- real \*8, dimension(:), allocatable **parm::no3daya**
- real \*8, dimension(:), allocatable **parm::minpdaya**
- real \*8, dimension(:), allocatable **parm::hi\_targ**
  - harvest index target of cover defined at planting ((kg/ha)/(kg/ha))*
- real \*8, dimension(:), allocatable **parm::bio\_targ**
  - biomass target (kg/ha)*
- real \*8, dimension(:), allocatable **parm::tnyld**
- integer, dimension(:), allocatable **parm::idapa**
- integer, dimension(:), allocatable **parm::iypa**
- integer, dimension(:), allocatable **parm::ifirsta**
- integer, dimension(100) **parm::mo\_transb**
- integer, dimension(100) **parm::mo\_transe**
- integer, dimension(100) **parm::ih\_tran**
- integer **parm::msdb**
  - maximum number of sept wq data database (none)*

- integer **parm::iseptic**
- real \*8, dimension(:), allocatable **parm::sptqs**  
*flow rate of the septic tank effluent per capita (m3/d)*
- real \*8, dimension(:), allocatable **parm::percp**
- real \*8, dimension(:), allocatable **parm::sptbodconcs**  
*Biological Oxygen Demand of the septic tank effluent (mg/l)*
- real \*8, dimension(:), allocatable **parm::spttssconcs**  
*concentration of total suspended solid in the septic tank effluent (mg/l)*
- real \*8, dimension(:), allocatable **parm::spttnconcs**  
*concentration of total nitrogen in the septic tank effluent (mg/l)*
- real \*8, dimension(:), allocatable **parm::sptnh4concs**  
*concentration of total phosphorus of the septic tank effluent (mg/l)*
- real \*8, dimension(:), allocatable **parm::sptno3concs**  
*concentration of nitrate in the septic tank effluent (mg/l)*
- real \*8, dimension(:), allocatable **parm::sptno2concs**  
*concentration of nitrite in the septic tank effluent (mg/l)*
- real \*8, dimension(:), allocatable **parm::sptorgnconcs**  
*concentration of organic nitrogen in the septic tank effluent (mg/l)*
- real \*8, dimension(:), allocatable **parm::spttpconcs**  
*concentration of total phosphorus in the septic tank effluent (mg/l)*
- real \*8, dimension(:), allocatable **parm::sptminps**  
*concentration of mineral phosphorus in the septic tank effluent (mg/l)*
- real \*8, dimension(:), allocatable **parm::sptorgps**  
*concentration of organic phosphorus in the septic tank effluent (mg/l)*
- real \*8, dimension(:), allocatable **parm::sptfcolis**  
*concentration of the facel caliform in the septic tank effluent (cfu/100ml)*
- real \*8, dimension(:), allocatable **parm::failyr**
- real \*8, dimension(:), allocatable **parm::qstemm**
- real \*8, dimension(:), allocatable **parm::bio\_amn**
- real \*8, dimension(:), allocatable **parm::bio\_bod**
- real \*8, dimension(:), allocatable **parm::biom**
- real \*8, dimension(:), allocatable **parm::rbiom**
- real \*8, dimension(:), allocatable **parm::fcoli**
- real \*8, dimension(:), allocatable **parm::bio\_ntr**
- real \*8, dimension(:), allocatable **parm::bz\_perc**
- real \*8, dimension(:), allocatable **parm::sep\_cap**  
*number of permanent residents in the hourse (none)*
- real \*8, dimension(:), allocatable **parm::plqm**
- real \*8, dimension(:), allocatable **parm::bz\_area**
- real \*8, dimension(:), allocatable **parm::bz\_z**  
*Depth of biozone layer(mm)*
- real \*8, dimension(:), allocatable **parm::bz\_thk**  
*thickness of biozone (mm)*
- real \*8, dimension(:), allocatable **parm::bio\_bd**  
*density of biomass (kg/m<sup>3</sup>) carbon outputs for .hru file*
- real \*8, dimension(:), allocatable **parm::cmup\_kgh**
- real \*8, dimension(:), allocatable **parm::cmtot\_kgh**
- real \*8, dimension(:), allocatable **parm::coeff\_denitr**  
*denitrification rate coefficient (none)*
- real \*8, dimension(:), allocatable **parm::coeff\_bod\_dc**  
*BOD decay rate coefficient (m<sup>3</sup>/day)*
- real \*8, dimension(:), allocatable **parm::coeff\_bod\_conv**

- BOD to live bacteria biomass conversion factor (none)*
- real \*8, dimension(:), allocatable **parm::coeff\_fc1**  
*field capacity calibration parameter 1 (none)*
- real \*8, dimension(:), allocatable **parm::coeff\_fc2**  
*field capacity calibration parameter 2 (none)*
- real \*8, dimension(:), allocatable **parm::coeff\_fecal**  
*fecal coliform bacteria decay rate coefficient ( $m^3/day$ )*
- real \*8, dimension(:), allocatable **parm::coeff\_mrt**  
*mortality rate coefficient (none)*
- real \*8, dimension(:), allocatable **parm::coeff\_nitr**  
*nitrification rate coefficient (none)*
- real \*8, dimension(:), allocatable **parm::coeff\_plq**  
*conversion factor for plaque from TDS (none)*
- real \*8, dimension(:), allocatable **parm::coeff\_rsp**  
*respiration rate coefficient (none)*
- real \*8, dimension(:), allocatable **parm::coeff\_slg1**  
*slough-off calibration parameter (none)*
- real \*8, dimension(:), allocatable **parm::coeff\_slg2**  
*slough-off calibration parameter (none)*
- real \*8, dimension(:), allocatable **parm::coeff\_pdistrb**
- real \*8, dimension(:), allocatable **parm::coeff\_solpslp**
- real \*8, dimension(:), allocatable **parm::coeff\_solpintc**
- real \*8, dimension(:), allocatable **parm::coeff\_psorpmax**
- integer, dimension(:), allocatable **parm::isep\_typ**  
*septic system type (none)*
- integer, dimension(:), allocatable **parm::i\_sep**
- integer, dimension(:), allocatable **parm::isep\_opt**  
*septic system operation flag (1=active, 2=failing, 3=not operated) (none)*
- integer, dimension(:), allocatable **parm::sep\_tsincefail**
- integer, dimension(:), allocatable **parm::isep\_tfail**
- integer, dimension(:), allocatable **parm::isep\_iyr**
- integer, dimension(:), allocatable **parm::sep\_strm\_dist**
- integer, dimension(:), allocatable **parm::sep\_den**
- real \*8, dimension(:), allocatable **parm::sol\_sumno3**
- real \*8, dimension(:), allocatable **parm::sol\_sumsolp**
- real \*8, dimension(:), allocatable **parm::strsw\_sum**
- real \*8, dimension(:), allocatable **parm::strstmp\_sum**
- real \*8, dimension(:), allocatable **parm::strsn\_sum**
- real \*8, dimension(:), allocatable **parm::strsp\_sum**
- real \*8, dimension(:), allocatable **parm::strsa\_sum**
- real \*8, dimension(:), allocatable **parm::spill\_hru**
- real \*8, dimension(:), allocatable **parm::tile\_out**
- real \*8, dimension(:), allocatable **parm::hru\_in**
- real \*8, dimension(:), allocatable **parm::spill\_precip**
- real \*8, dimension(:), allocatable **parm::pot\_seep**
- real \*8, dimension(:), allocatable **parm::pot\_evap**
- real \*8, dimension(:), allocatable **parm::pot\_sedin**
- real \*8, dimension(:), allocatable **parm::pot\_solp**  
*soluble P loss rate in the pothole (.01 - 0.5) (1/d)*
- real \*8, dimension(:), allocatable **parm::pot\_solpi**
- real \*8, dimension(:), allocatable **parm::pot\_orgp**
- real \*8, dimension(:), allocatable **parm::pot\_orgpi**
- real \*8, dimension(:), allocatable **parm::pot\_orn**

- real \*8, dimension(:), allocatable **parm::pot\_orgni**
- real \*8, dimension(:), allocatable **parm::pot\_mps**
- real \*8, dimension(:), allocatable **parm::pot\_mpsi**
- real \*8, dimension(:), allocatable **parm::pot\_mpa**
- real \*8, dimension(:), allocatable **parm::pot\_mpai**
- real \*8, dimension(:), allocatable **parm::pot\_no3i**
- real \*8, dimension(:), allocatable **parm::precip\_in**
- real \*8, dimension(:), allocatable **parm::tile\_sedo**
- real \*8, dimension(:), allocatable **parm::tile\_no3o**
- real \*8, dimension(:), allocatable **parm::tile\_solpo**
- real \*8, dimension(:), allocatable **parm::tile\_orgno**
- real \*8, dimension(:), allocatable **parm::tile\_orgpo**
- real \*8, dimension(:), allocatable **parm::tile\_minpso**
- real \*8, dimension(:), allocatable **parm::tile\_minpao**
- integer **parm::ia\_b**
- integer **parm::ihumus**
- integer **parm::itemp**
- integer **parm::isnow**
- integer, dimension(46) **parm::ipdvar**  
*output variable codes for output.rch file (none)*
- integer, dimension(mhruo) **parm::ipdvas**  
*output variable codes for output.hru file (none)*
- integer, dimension(msubo) **parm::ipdvab**  
*output variable codes for output.sub file (none)*
- integer, dimension(:), allocatable **parm::ipdhru**  
*HRUs whose output information will be printed to the output.hru and output.wtr files.*
- real \*8, dimension(mstdo) **parm::wshddayo**
- real \*8, dimension(mstdo) **parm::wshdmono**
- real \*8, dimension(mstdo) **parm::wshdyro**
- real \*8, dimension(16) **parm::fcstaa**
- real \*8, dimension(mstdo) **parm::wshdaao**
- real \*8, dimension(:,:), allocatable **parm::wpstdayo**
- real \*8, dimension(:,:), allocatable **parm::wpstmono**
- real \*8, dimension(:,:), allocatable **parm::wpstyro**
- real \*8, dimension(:,:), allocatable **parm::yldkg**
- real \*8, dimension(:,:), allocatable **parm::bio\_hv**
- real \*8, dimension(:,:), allocatable **parm::rchmono**  
*reach monthly output array (varies)*
- real \*8, dimension(:,:), allocatable **parm::wpstaa**
- real \*8, dimension(:,:), allocatable **parm::rchyro**
- real \*8, dimension(:,:), allocatable **parm::hrumono**  
*HRU monthly output data array (varies)*
- real \*8, dimension(:,:), allocatable **parm::rchaao**
- real \*8, dimension(:,:), allocatable **parm::rchdy**
- real \*8, dimension(:,:), allocatable **parm::hruyro**
- real \*8, dimension(:,:), allocatable **parm::submono**  
*subbasin monthly output array (varies)*
- real \*8, dimension(:,:), allocatable **parm::hruaao**
- real \*8, dimension(:,:), allocatable **parm::subyro**
- real \*8, dimension(:,:), allocatable **parm::subaao**
- real \*8, dimension(:,:), allocatable **parm::resoutm**  
*reservoir monthly output array (varies)*
- real \*8, dimension(:,:), allocatable **parm::resouty**

- real \*8, dimension(:,:), allocatable **parm::resouta**
- real \*8, dimension(12, 8) **parm::wshd\_aamon**
- real \*8, dimension(:,:), allocatable **parm::wtrmon**  
*HRU monthly output data array for impoundments (varies)*
- real \*8, dimension(:,:), allocatable **parm::wtryr**
- real \*8, dimension(:,:), allocatable **parm::wtraa**
- real \*8, dimension(:,:), allocatable **parm::sub\_smfmx**  
*max melt rate for snow during year (June 21) for subbasin(:) where deg C refers to the air temperature. SUB\_SMFMX and SMFMN allow the rate of snow melt to vary through the year. These parameters are accounting for the impact of soil temperature on snow melt (range: -5.0/5.0) (mm/deg C/day)*
- real \*8, dimension(:,:), allocatable **parm::sub\_smfmn**  
*min melt rate for snow during year (Dec 21) for subbasin(:) (range: -5.0/5.0) where deg C refers to the air temperature (mm/deg C/day)*
- real \*8, dimension(:,:), allocatable **parm::hrupstd**
- real \*8, dimension(:,:), allocatable **parm::hrupsta**
- real \*8, dimension(:,:), allocatable **parm::hrupstm**
- real \*8, dimension(:,:), allocatable **parm::hrupsty**
- integer, dimension(:), allocatable **parm::ifirstt**
- integer, dimension(:), allocatable **parm::ifirstpcp**
- integer, dimension(:), allocatable **parm::elevp**  
*elevation of precipitation gage station (m)*
- integer, dimension(:), allocatable **parm::elevt**  
*elevation of temperature gage station (m)*
- real \*8, dimension(:,:), allocatable **parm::ftmpmn**  
*avg monthly minimum air temperature (deg C)*
- real \*8, dimension(:,:), allocatable **parm::ftmpmx**  
*avg monthly maximum air temperature (deg C)*
- real \*8, dimension(:,:), allocatable **parm::ftmpstdmn**  
*standard deviation for avg monthly minimum air temperature (deg C)*
- real \*8, dimension(:,:), allocatable **parm::ftmpstdmx**  
*standard deviation for avg monthly maximum air temperature (deg C)*
- real \*8, dimension(:,:), allocatable **parm::fpcp\_stat**  
*fpcp\_stat(:,1,:): average amount of precipitation falling in one day for the month (mm/day)*  
*fpcp\_stat(:,2,:): standard deviation for the average daily precipitation (mm/day)*  
*fpcp\_stat(:,3,:): skew coefficient for the average daily precipitation (none)*
- real \*8, dimension(:,:), allocatable **parm::fpr\_w1**  
*probability of wet day after dry day in month (none)*
- real \*8, dimension(:,:), allocatable **parm::fpr\_w2**  
*probability of wet day after wet day in month (none)*
- real \*8, dimension(:,:), allocatable **parm::fpr\_w3**  
*proportion of wet days in the month (none)*
- real \*8, dimension(:), allocatable **parm::ch\_d**  
*average depth of main channel (m)*
- real \*8, dimension(:), allocatable **parm::flwin**
- real \*8, dimension(:), allocatable **parm::flwout**
- real \*8, dimension(:), allocatable **parm::bankst**
- real \*8, dimension(:), allocatable **parm::ch\_wi**
- real \*8, dimension(:), allocatable **parm::ch\_onco**  
*channel organic n concentration (ppm)*
- real \*8, dimension(:), allocatable **parm::ch\_opco**  
*channel organic p concentration (ppm)*
- real \*8, dimension(:), allocatable **parm::ch\_orgn**
- real \*8, dimension(:), allocatable **parm::ch\_orgp**

- real \*8, dimension(:), allocatable **parm::drift**
- real \*8, dimension(:), allocatable **parm::rch\_dox**
- real \*8, dimension(:), allocatable **parm::rch\_bactp**
- real \*8, dimension(:), allocatable **parm::alpha\_bnk**  
*alpha factor for bank storage recession curve (days)*
- real \*8, dimension(:), allocatable **parm::alpha\_bnke**  
 $\exp(-\alpha_{bnk})$  (none)
- real \*8, dimension(:), allocatable **parm::disolvp**
- real \*8, dimension(:), allocatable **parm::algae**
- real \*8, dimension(:), allocatable **parm::sedst**
- real \*8, dimension(:), allocatable **parm::rchstor**
- real \*8, dimension(:), allocatable **parm::organicn**
- real \*8, dimension(:), allocatable **parm::organicp**
- real \*8, dimension(:), allocatable **parm::chlora**
- real \*8, dimension(:), allocatable **parm::ch\_li**  
*initial length of main channel (km)*
- real \*8, dimension(:), allocatable **parm::ch\_si**  
*initial slope of main channel (m/m)*
- real \*8, dimension(:), allocatable **parm::nitraten**
- real \*8, dimension(:), allocatable **parm::nitriten**
- real \*8, dimension(:), allocatable **parm::ch\_bnk\_san**
- real \*8, dimension(:), allocatable **parm::ch\_bnk\_sil**
- real \*8, dimension(:), allocatable **parm::ch\_bnk\_cla**
- real \*8, dimension(:), allocatable **parm::ch\_bnk\_gra**
- real \*8, dimension(:), allocatable **parm::ch\_bed\_san**
- real \*8, dimension(:), allocatable **parm::ch\_bed\_sil**
- real \*8, dimension(:), allocatable **parm::ch\_bed\_cla**
- real \*8, dimension(:), allocatable **parm::ch\_bed\_gra**
- real \*8, dimension(:), allocatable **parm::depfp**
- real \*8, dimension(:), allocatable **parm::depsanfp**
- real \*8, dimension(:), allocatable **parm::depsilfp**
- real \*8, dimension(:), allocatable **parm::depclafp**
- real \*8, dimension(:), allocatable **parm::depsagfp**
- real \*8, dimension(:), allocatable **parm::deplagfp**
- real \*8, dimension(:), allocatable **parm::depch**
- real \*8, dimension(:), allocatable **parm::depsanch**
- real \*8, dimension(:), allocatable **parm::depsilch**
- real \*8, dimension(:), allocatable **parm::depclach**
- real \*8, dimension(:), allocatable **parm::depsagch**
- real \*8, dimension(:), allocatable **parm::deplagch**
- real \*8, dimension(:), allocatable **parm::depgrach**
- real \*8, dimension(:), allocatable **parm::depgrafp**
- real \*8, dimension(:), allocatable **parm::grast**
- real \*8, dimension(:), allocatable **parm::r2adj**  
*curve number retention parameter adjustment factor to adjust surface runoff for flat slopes (0.5 - 3.0) (dimensionless)*
- real \*8, dimension(:), allocatable **parm::prf**  
*Reach peak rate adjustment factor for sediment routing in the channel. Allows impact of peak flow rate on sediment routing and channel reshaping to be taken into account (none)*
- real \*8, dimension(:), allocatable **parm::depprch**
- real \*8, dimension(:), allocatable **parm::depprfp**
- real \*8, dimension(:), allocatable **parm::spcon**  
*linear parameter for calculating sediment reentrained in channel sediment routing*
- real \*8, dimension(:), allocatable **parm::spexp**

*exponent parameter for calculating sediment reentrained in channel sediment routing*

- real \*8, dimension(:), allocatable **parm::sanst**
- real \*8, dimension(:), allocatable **parm::silst**
- real \*8, dimension(:), allocatable **parm::clast**
- real \*8, dimension(:), allocatable **parm::sagst**
- real \*8, dimension(:), allocatable **parm::lagst**
- real \*8, dimension(:), allocatable **parm::pot\_san**
- real \*8, dimension(:), allocatable **parm::pot\_sil**
- real \*8, dimension(:), allocatable **parm::pot\_cla**
- real \*8, dimension(:), allocatable **parm::pot\_sag**
- real \*8, dimension(:), allocatable **parm::pot\_lag**
- real \*8, dimension(:), allocatable **parm::potsani**
- real \*8, dimension(:), allocatable **parm::potsili**
- real \*8, dimension(:), allocatable **parm::potclai**
- real \*8, dimension(:), allocatable **parm::potsagi**
- real \*8, dimension(:), allocatable **parm::potlagi**
- real \*8, dimension(:), allocatable **parm::sanyld**
- real \*8, dimension(:), allocatable **parm::silyld**
- real \*8, dimension(:), allocatable **parm::clayld**
- real \*8, dimension(:), allocatable **parm::sagyld**
- real \*8, dimension(:), allocatable **parm::lagyld**
- real \*8, dimension(:), allocatable **parm::grayld**
- real \*8, dimension(:), allocatable **parm::res\_san**
- real \*8, dimension(:), allocatable **parm::res\_sil**
- real \*8, dimension(:), allocatable **parm::res\_cla**
- real \*8, dimension(:), allocatable **parm::res\_sag**
- real \*8, dimension(:), allocatable **parm::res\_lag**
- real \*8, dimension(:), allocatable **parm::res\_gra**
- real \*8, dimension(:), allocatable **parm::pnd\_san**
- real \*8, dimension(:), allocatable **parm::pnd\_sil**
- real \*8, dimension(:), allocatable **parm::pnd\_cla**
- real \*8, dimension(:), allocatable **parm::pnd\_sag**
- real \*8, dimension(:), allocatable **parm::pnd\_lag**
- real \*8, dimension(:), allocatable **parm::wet\_san**
- real \*8, dimension(:), allocatable **parm::wet\_sil**
- real \*8, dimension(:), allocatable **parm::wet\_cla**
- real \*8, dimension(:), allocatable **parm::wet\_lag**
- real \*8, dimension(:), allocatable **parm::wet\_sag**
- real \*8 **parm::ressano**
- real \*8 **parm::ressilo**
- real \*8 **parm::resclao**
- real \*8 **parm::ressago**
- real \*8 **parm::reslago**
- real \*8 **parm::resgrao**
- real \*8 **parm::ressani**
- real \*8 **parm::ressili**
- real \*8 **parm::resclai**
- real \*8 **parm::ressagi**
- real \*8 **parm::reslagi**
- real \*8 **parm::resgrai**
- real \*8 **parm::potsano**
- real \*8 **parm::potsilo**
- real \*8 **parm::potclao**
- real \*8 **parm::potsago**
- real \*8 **parm::potlago**

- real \*8 **parm::pndsarin**
- real \*8 **parm::pndsilin**
- real \*8 **parm::pndclain**
- real \*8 **parm::pndsagin**
- real \*8 **parm::pndlugin**
- real \*8 **parm::pndsano**
- real \*8 **parm::pndsilo**
- real \*8 **parm::pndclao**
- real \*8 **parm::pndsago**
- real \*8 **parm::pndlago**
- real \*8, dimension(:), allocatable **parm::ch\_di**  
*initial depth of main channel (m)*
- real \*8, dimension(:), allocatable **parm::ch\_erod**  
*channel erodibility factor (0.0-1.0) (none)*  
*0 non-erosive channel*  
*1 no resistance to erosion*
- real \*8, dimension(:), allocatable **parm::ch\_l2**  
*length of main channel (km)*
- real \*8, dimension(:), allocatable **parm::ch\_cov**
- real \*8, dimension(:), allocatable **parm::ch\_bnk\_bd**  
*bulk density of channel bank sediment (1.1-1.9) (g/cc)*
- real \*8, dimension(:), allocatable **parm::ch\_bed\_bd**  
*bulk density of channel bed sediment (1.1-1.9) (g/cc)*
- real \*8, dimension(:), allocatable **parm::ch\_bnk\_kd**  
*erodibility of channel bank sediment by jet test (Peter Allen needs to give more info on this)*
- real \*8, dimension(:), allocatable **parm::ch\_bed\_kd**  
*erodibility of channel bed sediment by jet test (Peter Allen needs to give more info on this)*
- real \*8, dimension(:), allocatable **parm::ch\_bnk\_d50**  
*D50(median) particle size diameter of channel bank sediment (0.001 - 20)*
- real \*8, dimension(:), allocatable **parm::ch\_bed\_d50**  
*D50(median) particle size diameter of channel bed sediment (micrometers) (0.001 - 20)*
- real \*8, dimension(:), allocatable **parm::ch\_cov1**  
*channel erodibility factor (0.0-1.0) (none)*  
*0 non-erosive channel*  
*1 no resistance to erosion*
- real \*8, dimension(:), allocatable **parm::ch\_cov2**  
*channel cover factor (0.0-1.0) (none)*  
*0 channel is completely protected from erosion by cover*  
*1 no vegetative cover on channel*
- real \*8, dimension(:), allocatable **parm::tc\_bed**  
*critical shear stress of channel bed (N/m2)*
- real \*8, dimension(:), allocatable **parm::tc\_bnk**  
*critical shear stress of channel bank (N/m2)*
- integer, dimension(:), allocatable **parm::ch\_eqn**  
*sediment routine methods (DAILY):*  
*0 = original SWAT method*  
*1 = Bagnold's*  
*2 = Kodatie*  
*3 = Molinas WU*  
*4 = Yang*
- real \*8, dimension(:), allocatable **parm::chpst\_rea**  
*pesticide reaction coefficient in reach (1/day)*
- real \*8, dimension(:), allocatable **parm::chpst\_vol**  
*pesticide volatilization coefficient in reach (m/day)*



- real \*8, dimension(:), allocatable **parm::chpst\_conc**
- real \*8, dimension(:), allocatable **parm::chpst\_koc**  
*pesticide partition coefficient between water and sediment in reach ( $m^3/g$ )*
- real \*8, dimension(:), allocatable **parm::chpst\_rsp**  
*resuspension velocity in reach for pesticide sorbed to sediment (m/day)*
- real \*8, dimension(:), allocatable **parm::chpst\_stl**  
*settling velocity in reach for pesticide sorbed to sediment (m/day)*
- real \*8, dimension(:), allocatable **parm::ch\_wdr**  
*channel width to depth ratio (m/m)*
- real \*8, dimension(:), allocatable **parm::chpst\_mix**  
*mixing velocity (diffusion/dispersion) for pesticide in reach (m/day)*
- real \*8, dimension(:), allocatable **parm::sedpst\_conc**  
*initil pesticide concentration in river bed sediment ( $mg/m^3$ )*
- real \*8, dimension(:), allocatable **parm::sedpst\_bry**  
*pesticide burial velocity in river bed sediment (m/day)*
- real \*8, dimension(:), allocatable **parm::sedpst\_rea**  
*pesticide reaction coefficient in river bed sediment (1/day)*
- real \*8, dimension(:), allocatable **parm::sedpst\_act**  
*depth of active sediment layer in reach for pesticide (m)*
- real \*8, dimension(:), allocatable **parm::rch\_cbod**
- real \*8, dimension(:), allocatable **parm::rch\_bactlp**
- real \*8, dimension(:), allocatable **parm::chside**  
*change in horizontal distance per unit vertical distance (0.0 - 5)  
0 = for vertical channel bank  
5 = for channel bank with gentl side slope*
- real \*8, dimension(:), allocatable **parm::rs1**  
*local algal settling rate in reach at 20 deg C (m/day or m/hour)*
- real \*8, dimension(:), allocatable **parm::rs2**  
*benthos source rate for dissolved phosphorus in reach at 20 deg C ((mg disP-P)/( $m^2 \cdot day$ ) or (mg disP-P)/( $m^2 \cdot hour$ ))*
- real \*8, dimension(:), allocatable **parm::rs3**  
*benthos source rate for ammonia nitrogen in reach at 20 deg C ((mg NH4-N)/( $m^2 \cdot day$ ) or (mg NH4-N)/( $m^2 \cdot hour$ ))*
- real \*8, dimension(:), allocatable **parm::rs4**  
*rate coefficient for organic nitrogen settling in reach at 20 deg C (1/day or 1/hour)*
- real \*8, dimension(:), allocatable **parm::rs5**  
*organic phosphorus settling rate in reach at 20 deg C (1/day or 1/hour)*
- real \*8, dimension(:), allocatable **parm::rk1**  
*CBOD deoxygenation rate coefficient in reach at 20 deg C (1/day or 1/hour)*
- real \*8, dimension(:), allocatable **parm::rk2**  
*reaeration rate in accordance with Fickian diffusion in reach at 20 deg C (1/day or 1/hour)*
- real \*8, dimension(:), allocatable **parm::rk3**  
*rate of loss of CBOD due to settling in reach at 20 deg C (1/day or 1/hour)*
- real \*8, dimension(:), allocatable **parm::rk4**  
*sediment oxygen demand rate in reach at 20 deg C ( $mg\ O_2/(m^2 \cdot day)$  or  $mg\ O_2/(m^2 \cdot hour)$ )*
- real \*8, dimension(:), allocatable **parm::rk5**  
*coliform die-off rate in reach (1/day)*
- real \*8, dimension(:), allocatable **parm::rs6**  
*rate coefficient for settling of arbitrary non-conservative constituent in reach (1/day)*
- real \*8, dimension(:), allocatable **parm::rs7**  
*benthos source rate for arbitrary non-conservative constituent in reach ((mg ANC)/( $m^2 \cdot day$ ))*
- real \*8, dimension(:), allocatable **parm::bc1**

- rate constant for biological oxidation of NH3 to NO2 in reach at 20 deg C (1/day or 1/hour)*

  - real \*8, dimension(:), allocatable [parm::bc2](#)
- rate constant for biological oxidation of NO2 to NO3 in reach at 20 deg C (1/day or 1/hour)*

  - real \*8, dimension(:), allocatable [parm::bc3](#)
- rate constant for hydrolysis of organic N to ammonia in reach at 20 deg C (1/day or 1/hour)*

  - real \*8, dimension(:), allocatable [parm::bc4](#)
- rate constant for the decay of organic P to dissolved P in reach at 20 deg C (1/day or 1/hour)*

  - real \*8, dimension(:), allocatable [parm::rk6](#)
- decay rate for arbitrary non-conservative constituent in reach (1/day)*

  - real \*8, dimension(:), allocatable **parm::ammonian**
  - real \*8, dimension(:), allocatable **parm::orig\_sedpstconc**
  - real \*8, dimension(:,:), allocatable [parm::wurch](#)
- average daily water removal from the reach for the month ( $10^4 \text{ m}^3/\text{day}$ )*

  - integer, dimension(:), allocatable **parm::icanal**
  - integer, dimension(:), allocatable **parm::itb**
  - real \*8, dimension(:), allocatable [parm::ch\\_revap](#)
- revap coeff: this variable controls the amount of water moving from bank storage to the root zone as a result of soil moisture depletion(none)*

  - real \*8, dimension(:), allocatable **parm::dep\_chan**
  - real \*8, dimension(:), allocatable [parm::harg\\_petco](#)
- coefficient related to radiation used in hargreaves eq (range: 0.0019 - 0.0032)*

  - real \*8, dimension(:), allocatable **parm::subfr\_nowtr**
  - real \*8, dimension(:), allocatable [parm::cncoef\\_sub](#)
- soil water depletion coefficient used in the new (modified curve number method) same as soil index coeff used in APEX range: 0.5 - 2.0*

  - real \*8, dimension(:), allocatable **parm::dr\_sub**
  - real \*8, dimension(:), allocatable [parm::sub\\_fr](#)
- fraction of total watershed area contained in subbasin (km2/km2)*

  - real \*8, dimension(:), allocatable **parm::wcklsp**
  - real \*8, dimension(:), allocatable **parm::sub\_minp**
  - real \*8, dimension(:), allocatable **parm::sub\_sw**
  - real \*8, dimension(:), allocatable **parm::sub\_sumfc**
  - real \*8, dimension(:), allocatable **parm::sub\_gwno3**
  - real \*8, dimension(:), allocatable **parm::sub\_gwsolp**
  - real \*8, dimension(:), allocatable [parm::co2](#)
- CO2 concentration (ppmv)*

  - real \*8, dimension(:), allocatable [parm::sub\\_km](#)
- area of subbasin in square kilometers ( $\text{km}^2$ )*

  - real \*8, dimension(:), allocatable [parm::wlat](#)
- latitude of weather station used to compile data (degrees)*

  - real \*8, dimension(:), allocatable [parm::sub\\_tc](#)
- time of concentration for subbasin (hour)*

  - real \*8, dimension(:), allocatable **parm::sub\_pet**
  - real \*8, dimension(:), allocatable [parm::welev](#)
- elevation of weather station used to compile data (m)*

  - real \*8, dimension(:), allocatable **parm::sub\_orgn**
  - real \*8, dimension(:), allocatable **parm::sub\_orgp**
  - real \*8, dimension(:), allocatable **parm::sub\_bd**
  - real \*8, dimension(:), allocatable **parm::sub\_wtmp**
  - real \*8, dimension(:), allocatable **parm::sub\_sedpa**
  - real \*8, dimension(:), allocatable **parm::sub\_sedps**
  - real \*8, dimension(:), allocatable [parm::daylmn](#)

- shortest daylength occurring during the year (hour)*
- real \*8, dimension(:), allocatable **parm::sub\_minpa**
- real \*8, dimension(:), allocatable **parm::sub\_minps**
- real \*8, dimension(:), allocatable **parm::latcos**
- cos(latitude) (none)*
- real \*8, dimension(:), allocatable **parm::latsin**
- sin(latitude) (none)*
- real \*8, dimension(:), allocatable **parm::phutot**
- total potential heat units for year (used when no crop is growing) (heat unit)*
- real \*8, dimension(:), allocatable **parm::plaps**
- precipitation lapse rate: precipitation change due to change in elevation (mm H2O/km)*
- real \*8, dimension(:), allocatable **parm::tlaps**
- temperature lapse rate: temperature change due to change in elevation (deg C/km)*
- real \*8, dimension(:), allocatable **parm::tmp\_an**
- average annual air temperature (deg C)*
- real \*8, dimension(:), allocatable **parm::sub\_precip**
- real \*8, dimension(:), allocatable **parm::rammo\_sub**
- atmospheric deposition of ammonium values for entire watershed (mg/l)*
- real \*8, dimension(:), allocatable **parm::rcn\_sub**
- atmospheric deposition of nitrate for entire watershed (mg/l)*
- real \*8, dimension(:), allocatable **parm::pcpdays**
- real \*8, dimension(:), allocatable **parm::atmo\_day**
- real \*8, dimension(:), allocatable **parm::sub\_snom**
- real \*8, dimension(:), allocatable **parm::sub\_qd**
- real \*8, dimension(:), allocatable **parm::sub\_sedy**
- real \*8, dimension(:), allocatable **parm::sub\_tran**
- real \*8, dimension(:), allocatable **parm::sub\_no3**
- real \*8, dimension(:), allocatable **parm::sub\_latno3**
- real \*8, dimension(:,:), allocatable **parm::sub\_sftmp**
- snowfall temperature for subbasin(:). Mean air temperature at which precip is equally likely to be rain as snow/freezing rain (range: -5.0/5.0) (deg C)*
- real \*8, dimension(:,:), allocatable **parm::sub\_smtmp**
- snow melt base temperature for subbasin(:) mean air temperature at which snow melt will occur (range: -5.0/5.0) (deg C)*
- real \*8, dimension(:,:), allocatable **parm::sub\_timp**
- snow pack temperature lag factor (0-1) (none)*
- real \*8, dimension(:), allocatable **parm::sub\_tileno3**
- real \*8, dimension(:), allocatable **parm::sub\_solp**
- real \*8, dimension(:), allocatable **parm::sub\_subp**
- real \*8, dimension(:), allocatable **parm::sub\_etday**
- real \*8, dimension(:), allocatable **parm::sub\_elev**
- average elevation of subbasin (m)*
- real \*8, dimension(:), allocatable **parm::sub\_wyld**
- real \*8, dimension(:), allocatable **parm::sub\_surfq**
- real \*8, dimension(:), allocatable **parm::qird**
- real \*8, dimension(:), allocatable **parm::sub\_gwq**
- real \*8, dimension(:), allocatable **parm::sub\_sep**
- real \*8, dimension(:), allocatable **parm::sub\_chl**
- real \*8, dimension(:), allocatable **parm::sub\_cbod**
- real \*8, dimension(:), allocatable **parm::sub\_dox**
- real \*8, dimension(:), allocatable **parm::sub\_solpst**
- real \*8, dimension(:), allocatable **parm::sub\_sorpst**
- real \*8, dimension(:), allocatable **parm::sub\_yorgn**

- real \*8, dimension(:), allocatable **parm::sub\_yorgp**
- real \*8, dimension(:), allocatable **parm::sub\_lat**  
*latitude of HRU/subbasin (degrees)*
- real \*8, dimension(:), allocatable **parm::sub\_bactp**
- real \*8, dimension(:), allocatable **parm::sub\_bactlp**
- real \*8, dimension(:), allocatable **parm::sub\_latq**
- real \*8, dimension(:), allocatable **parm::sub\_gwq\_d**
- real \*8, dimension(:), allocatable **parm::sub\_tileq**
- real \*8, dimension(:), allocatable **parm::sub\_vaptile**
- real \*8, dimension(:), allocatable **parm::sub\_dsan**
- real \*8, dimension(:), allocatable **parm::sub\_dsil**
- real \*8, dimension(:), allocatable **parm::sub\_dcla**
- real \*8, dimension(:), allocatable **parm::sub\_dsag**
- real \*8, dimension(:), allocatable **parm::sub\_dlag**
- real \*8 **parm::vap\_tile**
- real \*8, dimension(:), allocatable **parm::wnan**
- real \*8, dimension(:, :), allocatable **parm::sol\_stpwt**
- real \*8, dimension(:, :), allocatable **parm::sub\_pst**
- real \*8, dimension(:, :), allocatable **parm::sub\_hhq**
- real \*8, dimension(:, :), allocatable **parm::sub\_hhwtm**
- real \*8, dimension(:, :), allocatable **parm::huminc**  
*monthly humidity adjustment. Daily values for relative humidity within the month are raised or lowered by the specified amount (used in climate change studies) (none)*
- real \*8, dimension(:, :), allocatable **parm::radinc**  
*monthly solar radiation adjustment. Daily radiation within the month is raised or lowered by the specified amount. (used in climate change studies) (MJ/m<sup>2</sup>)*
- real \*8, dimension(:, :), allocatable **parm::rfinc**  
*monthly rainfall adjustment. Daily rainfall within the month is adjusted to the specified percentage of the original value (used in climate change studies)(%)*
- real \*8, dimension(:, :), allocatable **parm::tmpinc**  
*monthly temperature adjustment. Daily maximum and minimum temperatures within the month are raised or lowered by the specified amount (used in climate change studies) (deg C)*
- real \*8, dimension(:), allocatable **parm::ch\_k1**  
*effective hydraulic conductivity of tributary channel alluvium (mm/hr)*
- real \*8, dimension(:), allocatable **parm::ch\_k2**  
*effective hydraulic conductivity of main channel alluvium (mm/hr)*
- real \*8, dimension(:, :), allocatable **parm::elevb**  
*elevation at the center of the band (m)*
- real \*8, dimension(:, :), allocatable **parm::elevb\_fr**  
*fraction of subbasin area within elevation band (the same fractions should be listed for all HRUs within the subbasin) (none)*
- real \*8, dimension(:, :), allocatable **parm::wndav**  
*average wind speed for the month (m/s)*
- real \*8, dimension(:), allocatable **parm::ch\_n1**  
*Manning's "n" value for the tributary channels (none)*
- real \*8, dimension(:), allocatable **parm::ch\_n2**  
*Manning's "n" value for the main channel (none)*
- real \*8, dimension(:), allocatable **parm::ch\_s1**  
*average slope of tributary channels (m/m)*
- real \*8, dimension(:), allocatable **parm::ch\_s2**  
*average slope of main channel (m/m)*
- real \*8, dimension(:), allocatable **parm::ch\_w1**  
*average width of tributary channels (m)*

- real \*8, dimension(:), allocatable [parm::ch\\_w2](#)  
*average width of main channel (m)*
- real \*8, dimension(:,:), allocatable [parm::dewpt](#)  
*average dew point temperature for the month (deg C)*
- real \*8, dimension(:,:), allocatable [parm::amp\\_r](#)  
*average fraction of total daily rainfall occurring in maximum half-hour period for month (none)*
- real \*8, dimension(:,:), allocatable [parm::solarav](#)  
*average daily solar radiation for the month (MJ/m<sup>2</sup>/day)*
- real \*8, dimension(:,:), allocatable **parm::tmpstdmx**
- real \*8, dimension(:,:), allocatable [parm::pcf](#)  
*normalization coefficient for precipitation generator (none)*
- real \*8, dimension(:,:), allocatable [parm::tmpmn](#)  
*avg monthly minimum air temperature (deg C)*
- real \*8, dimension(:,:), allocatable [parm::tmpmx](#)  
*avg monthly maximum air temperature (deg C)*
- real \*8, dimension(:,:), allocatable **parm::tmpstdmn**
- real \*8, dimension(:,:), allocatable **parm::otmpstdmn**
- real \*8, dimension(:,:), allocatable **parm::otmpmn**
- real \*8, dimension(:,:), allocatable **parm::otmpmx**
- real \*8, dimension(:,:), allocatable **parm::otmpstdmx**
- real \*8, dimension(:,:), allocatable **parm::ch\_erodmo**
- real \*8, dimension(:,:), allocatable **parm::uh**
- real \*8, dimension(:,:), allocatable **parm::hqdsave**
- real \*8, dimension(:,:), allocatable **parm::hsdsave**
- real \*8, dimension(:,:), allocatable [parm::pr\\_w1](#)  
*probability of wet day after dry day in month (none)*
- real \*8, dimension(:,:), allocatable [parm::pr\\_w2](#)  
*probability of wet day after wet day in month (none)*
- real \*8, dimension(:,:), allocatable [parm::pr\\_w3](#)  
*proportion of wet days in the month (none)*
- real \*8, dimension(:,:), allocatable **parm::pcp\_stat**
- real \*8, dimension(:,:), allocatable **parm::opr\_w1**
- real \*8, dimension(:,:), allocatable **parm::opr\_w2**
- real \*8, dimension(:,:), allocatable **parm::opr\_w3**
- real \*8, dimension(:,:), allocatable **parm::opcp\_stat**
- integer, dimension(:), allocatable [parm::ireg](#)  
*precipitation category (none):*  
1 precipitation <= 508 mm/yr  
2 precipitation > 508 and <= 1016 mm/yr  
3 precipitation > 1016 mm/yr
- integer, dimension(:), allocatable [parm::hrutot](#)  
*number of HRUs in subbasin (none)*
- integer, dimension(:), allocatable **parm::hru1**
- integer, dimension(:), allocatable [parm::ihgage](#)  
*subbasin relative humidity data code (none)*
- integer, dimension(:), allocatable [parm::isgage](#)  
*subbasin radiation gage data code (none)*
- integer, dimension(:), allocatable [parm::iwgage](#)  
*subbasin wind speed gage data code (none)*
- integer, dimension(:), allocatable [parm::subgis](#)  
*GIS code printed to output files (output.sub) (none).*
- integer, dimension(:), allocatable [parm::irgage](#)

- subbasin rain gage data code (none)*
- integer, dimension(:), allocatable [parm::itgage](#)
  - subbasin temp gage data code (none)*
- integer, dimension(:), allocatable [parm::irelh](#)
  - (none) irelh = 0 (dewpoint)*
  - irelh = 1 (relative humidity)*
  - note: inputs > 1.0 (dewpoint)*
  - inputs < 1.0 (relative hum)*
- integer, dimension(:), allocatable [parm::fcst\\_reg](#)
- real \*8, dimension(:,:), allocatable [parm::sol\\_aorgn](#)
  - amount of nitrogen stored in the active organic (humic) nitrogen pool (kg N/ha)*
- real \*8, dimension(:,:), allocatable [parm::sol\\_fon](#)
  - amount of nitrogen stored in the fresh organic (residue) pool (kg N/ha)*
- real \*8, dimension(:,:), allocatable [parm::sol\\_tmp](#)
- real \*8, dimension(:,:), allocatable [parm::sol\\_awc](#)
  - available water capacity of soil layer (mm H2O/mm soil)*
- real \*8, dimension(:,:), allocatable [parm::volcr](#)
  - crack volume for soil layer (mm)*
- real \*8, dimension(:,:), allocatable [parm::sol\\_prk](#)
- real \*8, dimension(:,:), allocatable [parm::pperco\\_sub](#)
  - subbasin phosphorus percolation coefficient. Ratio of soluble phosphorus in surface to soluble phosphorus in percolate*
- real \*8, dimension(:,:), allocatable [parm::sol\\_stap](#)
  - amount of phosphorus in the soil layer stored in the stable mineral phosphorus pool(kg P/ha)*
- real \*8, dimension(:,:), allocatable [parm::conv\\_wt](#)
  - factor which converts kg/kg soil to kg/ha (none)*
- real \*8, dimension(:,:), allocatable [parm::sol\\_actp](#)
  - amount of phosphorus stored in the active mineral phosphorus pool (kg P/ha)*
- real \*8, dimension(:,:), allocatable [parm::sol\\_solp](#)
  - soluble P concentration in top soil layer (mg P/kg soil) or*
  - amount of phosphorus stored in solution. NOTE UNIT CHANGE! (kg P/ha)*
- real \*8, dimension(:,:), allocatable [parm::crdep](#)
  - maximum or potential crack volume (mm)*
- real \*8, dimension(:,:), allocatable [parm::sol\\_fc](#)
  - amount of water available to plants in soil layer at field capacity (fc - wp) (mm H2O)*
- real \*8, dimension(:,:), allocatable [parm::sol\\_ul](#)
  - amount of water held in the soil layer at saturation (sat - wp water) (mm H2O)*
- real \*8, dimension(:,:), allocatable [parm::sol\\_bd](#)
  - bulk density of the soil (Mg/m<sup>3</sup>)*
- real \*8, dimension(:,:), allocatable [parm::sol\\_z](#)
  - depth to bottom of soil layer (mm)*
- real \*8, dimension(:,:), allocatable [parm::sol\\_st](#)
  - amount of water stored in the soil layer on any given day (less wp water) (mm H2O)*
- real \*8, dimension(:,:), allocatable [parm::sol\\_up](#)
  - water content of soil at -0.033 MPa (field capacity) (mm H2O/mm soil)*
- real \*8, dimension(:,:), allocatable [parm::sol\\_clay](#)
  - percent clay content in soil material (UNIT CHANGE!) (% or none)*
- real \*8, dimension(:,:), allocatable [parm::sol\\_hk](#)
  - beta coefficient to calculate hydraulic conductivity (none)*
- real \*8, dimension(:,:), allocatable [parm::flat](#)
- real \*8, dimension(:,:), allocatable [parm::sol\\_nh3](#)
- real \*8, dimension(:,:), allocatable [parm::sol\\_ec](#)

- electrical conductivity of soil layer (dS/m)*
- real \*8, dimension(:,:), allocatable **parm::sol\_orn**  
*amount of nitrogen stored in the stable organic N pool. NOTE UNIT CHANGE! (mg N/kg soil or kg N/ha)*
- real \*8, dimension(:,:), allocatable **parm::sol\_por**  
*total porosity of soil layer expressed as a fraction of the total volume (none)*
- real \*8, dimension(:,:), allocatable **parm::sol\_wp**  
*water content of soil at -1.5 MPa (wilting point) (mm H2O/mm soil)*
- real \*8, dimension(:,:), allocatable **parm::sol\_organp**  
*amount of phosphorus stored in the organic P pool. NOTE UNIT CHANGE! (mg P/kg soil or kg P/ha)*
- real \*8, dimension(:,:), allocatable **parm::sol\_hum**  
*amount of organic matter in the soil layer classified as humic substances (kg humus/ha)*
- real \*8, dimension(:,:), allocatable **parm::sol\_wpm**  
*water content of soil at -1.5 MPa (wilting point) (mm H2O)*
- real \*8, dimension(:,:), allocatable **parm::sol\_no3**  
*amount of nitrogen stored in the nitrate pool. This variable is read in as a concentration and converted to kg/ha (this value is read from the .sol file in units of mg/kg) (kg N/ha)*
- real \*8, dimension(:,:), allocatable **parm::sol\_cbn**  
*percent organic carbon in soil layer (%)*
- real \*8, dimension(:,:), allocatable **parm::sol\_k**  
*saturated hydraulic conductivity of soil layer (mm/hour)*
- real \*8, dimension(:,:), allocatable **parm::sol\_rsd**  
*amount of organic matter in the soil layer classified as residue (kg/ha)*
- real \*8, dimension(:,:), allocatable **parm::sol\_fop**  
*amount of phosphorus stored in the fresh organic (residue) pool (kg P/ha)*
- real \*8, dimension(:,:), allocatable **parm::sol\_rock**  
*percent of rock fragments in soil layer (%)*
- real \*8, dimension(:,:), allocatable **parm::sol\_silt**  
*percent silt content in soil material (UNIT CHANGE!) (% or none)*
- real \*8, dimension(:,:), allocatable **parm::sol\_sand**  
*percent sand content of soil material (%)*
- real \*8, dimension(:,:), allocatable **parm::orig\_solno3**
- real \*8, dimension(:,:), allocatable **parm::orig\_solorn**
- real \*8, dimension(:,:), allocatable **parm::orig\_sololp**
- real \*8, dimension(:,:), allocatable **parm::orig\_solorgp**
- real \*8, dimension(:,:), allocatable **parm::orig\_soltmp**
- real \*8, dimension(:,:), allocatable **parm::orig\_solrsd**
- real \*8, dimension(:,:), allocatable **parm::orig\_solfop**
- real \*8, dimension(:,:), allocatable **parm::orig\_solfon**
- real \*8, dimension(:,:), allocatable **parm::orig\_solaorn**
- real \*8, dimension(:,:), allocatable **parm::orig\_solst**
- real \*8, dimension(:,:), allocatable **parm::orig\_solactp**
- real \*8, dimension(:,:), allocatable **parm::orig\_solstap**
- real \*8, dimension(:,:), allocatable **parm::orig\_volcr**
- real \*8, dimension(:,:), allocatable **parm::conk**
- real \*8, dimension(:,:), allocatable **parm::sol\_pst**  
*sol\_pst(:,1) initial amount of pesticide in first layer read in from .chm file (mg/kg)*  
*sol\_pst(:,2) amount of pesticide in layer. NOTE UNIT CHANGE! (kg/ha)*
- real \*8, dimension(:,:), allocatable **parm::sol\_kp**  
*pesticide sorption coefficient, Kp; the ratio of the concentration in the solid phase to the concentration in solution ((mg/kg)/(mg/L))*
- real \*8, dimension(:,:), allocatable **parm::orig\_solpst**
- real \*8, dimension(:,:), allocatable **parm::velsetlr**



- real \*8, dimension(:), allocatable **parm::velsetlp**
- real \*8, dimension(:), allocatable **parm::br1**  
*1st shape parameter for reservoir surface area equation (none)*
- real \*8, dimension(:), allocatable **parm::evrsv**  
*lake evaporation coefficient (none)*
- real \*8, dimension(:), allocatable **parm::res\_k**  
*hydraulic conductivity of the reservoir bottom (mm/hr)*
- real \*8, dimension(:), allocatable **parm::lkpst\_conc**  
*pesticide concentration in lake water (mg/m<sup>3</sup>)*
- real \*8, dimension(:), allocatable **parm::res\_evol**  
*volume of water needed to fill the reservoir to the emergency spillway (read in as 10<sup>4</sup> m<sup>3</sup> and converted to m<sup>3</sup>) (m<sup>3</sup>)*
- real \*8, dimension(:), allocatable **parm::res\_pvol**  
*volume of water needed to fill the reservoir to the principal spillway (read in as 10<sup>4</sup> m<sup>3</sup> and converted to m<sup>3</sup>) (m<sup>3</sup>)*
- real \*8, dimension(:), allocatable **parm::res\_vol**  
*reservoir volume (read in as 10<sup>4</sup> m<sup>3</sup> and converted to m<sup>3</sup>) (m<sup>3</sup>)*
- real \*8, dimension(:), allocatable **parm::res\_psa**  
*reservoir surface area when reservoir is filled to principal spillway (ha)*
- real \*8, dimension(:), allocatable **parm::lkpst\_rea**  
*pesticide reaction coefficient in lake water (1/day)*
- real \*8, dimension(:), allocatable **parm::lkpst\_vol**  
*pesticide volatilization coefficient in lake water (m/day)*
- real \*8, dimension(:), allocatable **parm::br2**  
*2nd shape parameter for reservoir surface area equation (none)*
- real \*8, dimension(:), allocatable **parm::res\_rr**  
*average daily principal spillway release volume (read in as a release rate in m<sup>3</sup>/s and converted to m<sup>3</sup>/day) (m<sup>3</sup>/day)*
- real \*8, dimension(:), allocatable **parm::res\_sed**  
*amount of sediment in reservoir (read in as mg/L and converted to kg/L) (kg/L)*
- real \*8, dimension(:), allocatable **parm::lkpst\_koc**  
*pesticide partition coefficient between water and sediment in lake water (m<sup>3</sup>/g)*
- real \*8, dimension(:), allocatable **parm::lkpst\_mix**  
*mixing velocity (diffusion/dispersion) in lake water for pesticide (m/day)*
- real \*8, dimension(:), allocatable **parm::lkpst\_rsp**  
*resuspension velocity in lake water for pesticide sorbed to sediment (m/day)*
- real \*8, dimension(:), allocatable **parm::lkpst\_stl**  
*settling velocity in lake water for pesticide sorbed to sediment (m/day)*
- real \*8, dimension(:), allocatable **parm::lkspst\_conc**  
*pesticide concentration in lake bed sediment (mg/m<sup>3</sup>)*
- real \*8, dimension(:), allocatable **parm::lkspst\_rea**  
*pesticide reaction coefficient in lake bed sediment (1/day)*
- real \*8, dimension(:), allocatable **parm::theta\_n**
- real \*8, dimension(:), allocatable **parm::theta\_p**
- real \*8, dimension(:), allocatable **parm::con\_nirr**
- real \*8, dimension(:), allocatable **parm::con\_pirr**
- real \*8, dimension(:), allocatable **parm::lkspst\_act**  
*depth of active sediment layer in lake for for pesticide (m)*
- real \*8, dimension(:), allocatable **parm::lkspst\_bry**  
*pesticide burial velocity in lake bed sediment (m/day)*
- real \*8, dimension(:), allocatable **parm::sed\_stlr**
- real \*8, dimension(7) **parm::resdata**



- real \*8, dimension(:), allocatable [parm::res\\_nsed](#)  
*normal amount of sediment in reservoir (read in as mg/L and convert to kg/L) (kg/L)*
- real \*8, dimension(:), allocatable [parm::wurnf](#)  
*fraction of water removed from the reservoir via WURESN which is returned and becomes flow from the reservoir outlet (none)*
- real \*8, dimension(:), allocatable [parm::chlar](#)  
*chlorophyll-a production coefficient for reservoir (none)*
- real \*8, dimension(:), allocatable [parm::res\\_no3](#)  
*amount of nitrate in reservoir (kg N)*
- real \*8, dimension(:), allocatable [parm::res\\_orgn](#)  
*amount of organic N in reservoir (kg N)*
- real \*8, dimension(:), allocatable [parm::res\\_orgp](#)  
*amount of organic P in reservoir (kg P)*
- real \*8, dimension(:), allocatable [parm::res\\_solp](#)  
*amount of soluble P in reservoir (kg P)*
- real \*8, dimension(:), allocatable **parm::res\_chla**
- real \*8, dimension(:), allocatable **parm::res\_seci**
- real \*8, dimension(:), allocatable [parm::res\\_esa](#)  
*reservoir surface area when reservoir is filled to emergency spillway (ha)*
- real \*8, dimension(:), allocatable [parm::res\\_nh3](#)  
*amount of ammonia in reservoir (kg N)*
- real \*8, dimension(:), allocatable [parm::res\\_no2](#)  
*amount of nitrite in reservoir (kg N)*
- real \*8, dimension(:), allocatable [parm::seccir](#)  
*water clarity coefficient for reservoir (none)*
- real \*8, dimension(:), allocatable **parm::res\_bactp**
- real \*8, dimension(:), allocatable **parm::res\_bactlp**
- real \*8, dimension(:), allocatable [parm::oflowmn\\_fps](#)  
*minimum reservoir outflow as a fraction of the principal spillway volume (0-1) (fraction)*
- real \*8, dimension(:), allocatable [parm::starg\\_fps](#)  
*target volume as a fraction of the principal spillway volume (.1-5) (fraction)*
- real \*8, dimension(:), allocatable **parm::weirc**
- real \*8, dimension(:), allocatable **parm::weirk**
- real \*8, dimension(:), allocatable **parm::weirw**
- real \*8, dimension(:), allocatable **parm::acoef**
- real \*8, dimension(:), allocatable **parm::bcoef**
- real \*8, dimension(:), allocatable **parm::ccoef**
- real \*8, dimension(:), allocatable **parm::orig\_resvol**
- real \*8, dimension(:), allocatable **parm::orig\_ressed**
- real \*8, dimension(:), allocatable **parm::orig\_lkpstconc**
- real \*8, dimension(:), allocatable **parm::orig\_lkspstconc**
- real \*8, dimension(:), allocatable **parm::orig\_ressolp**
- real \*8, dimension(:), allocatable **parm::orig\_resorgp**
- real \*8, dimension(:), allocatable **parm::orig\_resno3**
- real \*8, dimension(:), allocatable **parm::orig\_resno2**
- real \*8, dimension(:), allocatable **parm::orig\_resnh3**
- real \*8, dimension(:), allocatable **parm::orig\_resorgn**
- real \*8, dimension(:,:), allocatable [parm::oflowmn](#)  
*minimum daily outflow for the month (read in as  $m^3/s$  and converted to  $m^3/day$ ) ( $m^3/day$ )*
- real \*8, dimension(:,:), allocatable [parm::oflowmx](#)  
*maximum daily outflow for the month (read in as  $m^3/s$  and converted to  $m^3/day$ ) ( $m^3/day$ )*
- real \*8, dimension(:,:), allocatable [parm::starg](#)

- monthly target reservoir storage (needed if IRESKO=2) (read in as  $10^4 \text{ m}^3$  and converted to  $\text{m}^3$ ) ( $\text{m}^3$ )*

  - real \*8, dimension(:), allocatable [parm::psetlr1](#)  
*phosphorus settling rate for mid-year period (read in as m/year and converted to m/day) (m/day)*
  - real \*8, dimension(:), allocatable [parm::psetlr2](#)  
*phosphorus settling rate for remainder of year (read in as m/year and converted to m/day) (m/day)*
  - real \*8, dimension(:), allocatable [parm::nsetlr1](#)  
*nitrogen settling rate for mid-year period (read in as m/year and converted to m/day) (m/day)*
  - real \*8, dimension(:), allocatable [parm::nsetlr2](#)  
*nitrogen settling rate for remainder of year (read in as m/year and converted to m/day) (m/day)*
  - real \*8, dimension(:,:), allocatable [parm::wuresn](#)  
*average amount of water withdrawn from reservoir each month for consumptive water use (read in as  $10^4 \text{ m}^3$  and converted to  $\text{m}^3$ ) ( $\text{m}^3$ )*
  - real \*8, dimension(:,:), allocatable [parm::res\\_out](#)  
*measured average daily outflow from the reservoir for the month (needed if IRESKO=1) (read in as  $\text{m}^3/\text{s}$  and converted to  $\text{m}^3/\text{day}$ ) ( $\text{m}^3/\text{day}$ )*
  - integer, dimension(:), allocatable [parm::res\\_sub](#)  
*number of subbasin reservoir is in (weather for the subbasin is used for the reservoir) (none)*
  - integer, dimension(:), allocatable [parm::ires1](#)  
*beginning of mid-year nutrient settling "season" (none)*
  - integer, dimension(:), allocatable [parm::ires2](#)  
*end of mid-year nutrient settling "season" (none)*
  - integer, dimension(:), allocatable [parm::iresco](#)  
*outflow simulation code (none):*  
*0 compute outflow for uncontrolled reservoir with average annual release rate*  
*1 measured monthly outflow*  
*2 simulated controlled outflow-target release*  
*3 measured daily outflow*  
*4 stage/volume/outflow relationship*
  - integer, dimension(:), allocatable [parm::iyres](#)  
*year of the simulation that the reservoir becomes operational (none)*
  - integer, dimension(:), allocatable [parm::mores](#)  
*month the reservoir becomes operational (none)*
  - integer, dimension(:), allocatable [parm::iflod1r](#)  
*beginning month of non-flood season (needed if IRESKO=2) (none)*
  - integer, dimension(:), allocatable [parm::iflod2r](#)  
*ending month of non-flood season (needed if IRESKO=2) (none)*
  - integer, dimension(:), allocatable [parm::ndtargr](#)  
*number of days to reach target storage from current reservoir storage (needed if IRESKO=2) (days)*
  - real \*8, dimension(:), allocatable [parm::ap\\_ef](#)  
*application efficiency (0-1) (none)*
  - real \*8, dimension(:), allocatable [parm::decay\\_f](#)  
*exponential of the rate constant for degradation of the pesticide on foliage (none)*
  - real \*8, dimension(:), allocatable [parm::skoc](#)  
*soil adsorption coefficient normalized for soil organic carbon content ((mg/kg)/(mg/L))*
  - real \*8, dimension(:), allocatable [parm::decay\\_s](#)  
*exponential of the rate constant for degradation of the pesticide in soil (none)*
  - real \*8, dimension(:), allocatable [parm::hlife\\_f](#)  
*half-life of pesticide on foliage (days)*
  - real \*8, dimension(:), allocatable [parm::hlife\\_s](#)  
*half-life of pesticide in soil (days)*
  - real \*8, dimension(:), allocatable [parm::pst\\_wof](#)  
*fraction of pesticide on foliage which is washed-off by a rainfall event (none)*

- real \*8, dimension(:), allocatable [parm::pst\\_wsol](#)  
*solubility of chemical in water (mg/L (ppm))*
- real \*8, dimension(:), allocatable [parm::irramt](#)
- real \*8, dimension(:), allocatable [parm::phusw](#)
- real \*8, dimension(:), allocatable [parm::phusw\\_nocrop](#)
- integer, dimension(:), allocatable [parm::pstflg](#)  
*flag for types of pesticide used in watershed. Array location is pesticide ID number*  
*0: pesticide not used*  
*1: pesticide used*
- integer, dimension(:), allocatable [parm::nope](#)  
*sequence number of pesticide in NPNO(:) (none)*
- integer, dimension(:), allocatable [parm::nop](#)
- integer, dimension(:), allocatable [parm::yr\\_skip](#)
- integer, dimension(:), allocatable [parm::isweep](#)
- integer, dimension(:), allocatable [parm::icrmx](#)
- integer, dimension(:), allocatable [parm::nopmx](#)
- integer, dimension(:, :), allocatable [parm::mgtop](#)
- integer, dimension(:, :), allocatable [parm::idop](#)
- integer, dimension(:, :), allocatable [parm::mgt1iop](#)
- integer, dimension(:, :), allocatable [parm::mgt2iop](#)
- integer, dimension(:, :), allocatable [parm::mgt3iop](#)
- real \*8, dimension(:, :), allocatable [parm::mgt4op](#)
- real \*8, dimension(:, :), allocatable [parm::mgt5op](#)
- real \*8, dimension(:, :), allocatable [parm::mgt6op](#)
- real \*8, dimension(:, :), allocatable [parm::mgt7op](#)
- real \*8, dimension(:, :), allocatable [parm::mgt8op](#)
- real \*8, dimension(:, :), allocatable [parm::mgt9op](#)
- real \*8, dimension(:, :), allocatable [parm::mgt10iop](#)
- real \*8, dimension(:, :), allocatable [parm::phu\\_op](#)
- real \*8, dimension(:), allocatable [parm::cnyld](#)  
*fraction of nitrogen in yield (kg N/kg yield)*
- real \*8, dimension(:), allocatable [parm::rsdco\\_pl](#)  
*plant residue decomposition coefficient. The fraction of residue which will decompose in a day assuming optimal moisture, temperature, C:N ratio, and C:P ratio (none)*
- real \*8, dimension(:), allocatable [parm::wac21](#)  
*1st shape parameter for radiation use efficiency equation (none)*
- real \*8, dimension(:), allocatable [parm::wac22](#)  
*2nd shape parameter for radiation use efficiency equation (none)*
- real \*8, dimension(:), allocatable [parm::alai\\_min](#)  
*minimum LAI during winter dormant period ( $m^2/m^2$ )*
- real \*8, dimension(:), allocatable [parm::leaf1](#)  
*1st shape parameter for leaf area development equation (none)*
- real \*8, dimension(:), allocatable [parm::leaf2](#)  
*2nd shape parameter for leaf area development equation (none)*
- real \*8, dimension(:), allocatable [parm::wsyf](#)  
*Value of harvest index between 0 and HVSTI which represents the lowest value expected due to water stress ((kg/ha)/(kg/ha))*
- real \*8, dimension(:), allocatable [parm::bio\\_e](#)  
*biomass-energy ratio. The potential (unstressed) growth rate per unit of intercepted photosynthetically active radiation. ((kg/ha)/(MJ/m\*\*2))*
- real \*8, dimension(:), allocatable [parm::hvsti](#)  
*harvest index: crop yield/aboveground biomass ((kg/ha)/(kg/ha))*
- real \*8, dimension(:), allocatable [parm::t\\_base](#)

- minimum temperature for plant growth (deg C)*
- real \*8, dimension(:), allocatable [parm::t\\_opt](#)
  - optimal temperature for plant growth (deg C)*
- real \*8, dimension(:), allocatable [parm::chtmx](#)
  - maximum canopy height (m)*
- real \*8, dimension(:), allocatable [parm::cvm](#)
  - natural log of USLE\_C (none)*
- real \*8, dimension(:), allocatable [parm::gsi](#)
  - maximum stomatal conductance (m/s)*
- real \*8, dimension(:), allocatable [parm::vpd2](#)
  - rate of decline in stomatal conductance per unit increase in vapor pressure deficit ((m/s)\*(1/kPa))*
- real \*8, dimension(:), allocatable [parm::wavp](#)
  - rate of decline in radiation use efficiency as a function of vapor pressure deficit (none)*
- real \*8, dimension(:), allocatable [parm::bio\\_leaf](#)
  - fraction of leaf/needle biomass that drops during dormancy (for trees only) (none)*
- real \*8, dimension(:), allocatable [parm::blai](#)
  - maximum (potential) leaf area index (none)*
- real \*8, dimension(:), allocatable [parm::cpyld](#)
  - fraction of phosphorus in yield (kg P/kg yield)*
- real \*8, dimension(:), allocatable [parm::dlai](#)
  - fraction of growing season when leaf area declines (none)*
- real \*8, dimension(:), allocatable [parm::rdmx](#)
  - maximum root depth of plant (m)*
- real \*8, dimension(:), allocatable [parm::bio\\_n1](#)
  - 1st shape parameter for plant N uptake equation (none)*
- real \*8, dimension(:), allocatable [parm::bio\\_n2](#)
  - 2nd shape parameter for plant N uptake equation (none)*
- real \*8, dimension(:), allocatable [parm::bio\\_p1](#)
  - 1st shape parameter for plant P uptake equation (none)*
- real \*8, dimension(:), allocatable [parm::bio\\_p2](#)
  - 2st shape parameter for plant P uptake equation (none)*
- real \*8, dimension(:), allocatable [parm::bm\\_dieoff](#)
  - fraction above ground biomass that dies off at dormancy (fraction)*
- real \*8, dimension(:), allocatable [parm::bmx\\_trees](#)
- real \*8, dimension(:), allocatable [parm::ext\\_coef](#)
- real \*8, dimension(:), allocatable [parm::rsr1](#)
  - initial root to shoot ratio at the beg of growing season*
- real \*8, dimension(:), allocatable [parm::rsr2](#)
  - root to shoot ratio at the end of the growing season*
- real \*8, dimension(:), allocatable [parm::pltnfr1](#)
  - nitrogen uptake parameter #1: normal fraction of N in crop biomass at emergence (kg N/kg biomass)*
- real \*8, dimension(:), allocatable [parm::pltnfr2](#)
  - nitrogen uptake parameter #2: normal fraction of N in crop biomass at 0.5 maturity (kg N/kg biomass)*
- real \*8, dimension(:), allocatable [parm::pltnfr3](#)
  - nitrogen uptake parameter #3: normal fraction of N in crop biomass at maturity (kg N/kg biomass)*
- real \*8, dimension(:), allocatable [parm::pltpfr1](#)
  - phosphorus uptake parameter #1: normal fraction of P in crop biomass at emergence (kg P/kg biomass)*
- real \*8, dimension(:), allocatable [parm::pltpfr2](#)
  - phosphorus uptake parameter #2: normal fraction of P in crop biomass at 0.5 maturity (kg P/kg biomass)*
- real \*8, dimension(:), allocatable [parm::pltpfr3](#)

- phosphorus uptake parameter #3: normal fraction of P in crop biomass at maturity (kg P/kg biomass)*

  - integer, dimension(:), allocatable **parm::idc**
    - crop/landcover category:*
    - 1 warm season annual legume
    - 2 cold season annual legume
    - 3 perennial legume
    - 4 warm season annual
    - 5 cold season annual
    - 6 perennial
    - 7 trees
- integer, dimension(:), allocatable **parm::mat\_yrs**
- real \*8, dimension(:), allocatable **parm::bactpdb**
  - concentration of persistent bacteria in manure (fertilizer) (cfu/g manure)*
- real \*8, dimension(:), allocatable **parm::fminn**
  - fraction of mineral N (NO<sub>3</sub> + NH<sub>3</sub>) (kg minN/kg fert)*
- real \*8, dimension(:), allocatable **parm::forgn**
  - fraction of organic N (kg orgN/kg fert)*
- real \*8, dimension(:), allocatable **parm::forgp**
  - fraction of organic P (kg orgP/kg fert)*
- real \*8, dimension(:), allocatable **parm::bactkddb**
  - bacteria partition coefficient (none):*
  - 1: all bacteria in solution
  - 0: all bacteria sorbed to soil particles
- real \*8, dimension(:), allocatable **parm::bactlpdb**
  - concentration of less persistent bacteria in manure (fertilizer) (cfu/g manure)*
- real \*8, dimension(:), allocatable **parm::fminp**
  - fraction of mineral P (kg minP/kg fert)*
- real \*8, dimension(:), allocatable **parm::fnh3n**
  - fraction of NH<sub>3</sub>-N in mineral N (kg NH<sub>3</sub>-N/kg minN)*
- character(len=8), dimension(200) **parm::fertnm**
  - name of fertilizer*
- real \*8, dimension(:), allocatable **parm::curbden**
  - curb length density in HRU (km/ha)*
- real \*8, dimension(:), allocatable **parm::dirtmx**
  - maximum amount of solids allowed to build up on impervious surfaces (kg/curb km)*
- real \*8, dimension(:), allocatable **parm::fimp**
  - fraction of HRU area that is impervious (both directly and indirectly connected)(fraction)*
- real \*8, dimension(:), allocatable **parm::urbcoef**
  - wash-off coefficient for removal of constituents from an impervious surface (1/mm)*
- real \*8, dimension(:), allocatable **parm::thalf**
  - time for the amount of solids on impervious areas to build up to 1/2 the maximum level (days)*
- real \*8, dimension(:), allocatable **parm::tnconc**
  - concentration of total nitrogen in suspended solid load from impervious areas (mg N/kg sed)*
- real \*8, dimension(:), allocatable **parm::tno3conc**
  - concentration of NO<sub>3</sub>-N in suspended solid load from impervious areas (mg NO<sub>3</sub>-N/kg sed)*
- real \*8, dimension(:), allocatable **parm::tpconc**
  - concentration of total phosphorus in suspended solid load from impervious areas (mg P/kg sed)*
- real \*8, dimension(:), allocatable **parm::fcimp**
  - fraction of HRU area that is classified as directly connected impervious (fraction)*
- real \*8, dimension(:), allocatable **parm::urbcn2**
  - SCS curve number for moisture condition II in impervious areas (none)*
- real \*8 **parm::fr\_curb**
  - availability factor, the fraction of the curb length that is sweepable (none)*

- real \*8 **parm::ftr\_kg**  
*amount of fertilizer applied to HRU (kg/ha)*
- real \*8 **parm::pst\_dep**  
*depth of pesticide in the soil (mm)*
- real \*8 **parm::sweepeff**
- real \*8, dimension(:), allocatable **parm::ranrns\_hru**
- integer, dimension(:), allocatable **parm::itill**
- real \*8, dimension(:), allocatable **parm::deptil**  
*depth of mixing caused by operation (mm)*
- real \*8, dimension(:), allocatable **parm::effmix**  
*mixing efficiency of operation (none)*
- real \*8, dimension(:), allocatable **parm::ranrns**  
*random roughness of a given tillage operation (mm)*
- character(len=8), dimension(550) **parm::tillnm**  
*8-character name for the tillage operation*
- real \*8, dimension(:), allocatable **parm::rnum1s**  
*For ICODES equal to (none)*  
*0,1,3,5,9: not used*  
*2: Fraction of flow in channel*  
*4: amount of water transferred (as defined by INUM4S)*  
*7,8,10,11: drainage area in square kilometers associated with the record file.*
- real \*8, dimension(:), allocatable **parm::hyd\_dakm**  
*total drainage area of hydrograph in square kilometers (km<sup>2</sup>)*
- real \*8, dimension(:,:), allocatable **parm::varoute**
- real \*8, dimension(:,:), allocatable **parm::shyd**
- real \*8, dimension(:,:), allocatable **parm::vartran**
- real \*8, dimension(:,:,:), allocatable **parm::hhvaroute**
- integer, dimension(:), allocatable **parm::icodes**  
*routing command code (none):*  
*0 = finish*  
*1 = subbasin*  
*2 = route*  
*3 = routres*  
*4 = transfer*  
*5 = add*  
*6 = rechour*  
*7 = recmon*  
*8 = recyear*  
*9 = save*  
*10 = recday*  
*11 = recnst*  
*12 = structure*  
*13 = apex*  
*14 = saveconc*  
*15 =*
- integer, dimension(:), allocatable **parm::ihouts**  
*For ICODES equal to (none)*  
*0: not used*  
*1,2,3,5,7,8,10,11: hydrograph storage location number*  
*4: departure type (1=reach, 2=reservoir)*  
*9: hydrograph storage location of data to be printed to event file*  
*14:hydrograph storage location of data to be printed to saveconc file.*
- integer, dimension(:), allocatable **parm::inum1s**  
*For ICODES equal to (none)*  
*0: not used*  
*1: subbasin number*  
*2: reach number*

- 3: reservoir number
- 4: reach or res # flow is diverted from
- 5: hydrograph storage location of 1st dataset to be added
- 7,8,9,10,11,14: file number.
- integer, dimension(:), allocatable **parm::inum2s**
  - For ICODES equal to (none)
  - 0,1,7,8,10,11: not used
  - 2,3: inflow hydrograph storage location
  - 4: destination type (1=reach, 2=reservoir)
  - 5: hydrograph storage location of 2nd dataset to be added
  - 9,14: print frequency (0=daily, 1=hourly)
- integer, dimension(:), allocatable **parm::inum3s**
  - For ICODES equal to (none)
  - 0,1,2,3,5,7,8,10,11: not used
  - 4: destination number. Reach or reservoir receiving water
  - 9: print format (0=normal, fixed format; 1=txt format for AV interface, recday)
- integer, dimension(:), allocatable **parm::inum4s**
  - For ICODES equal to (none)
  - 0,2,3,5,7,8,9,10,11: not used
  - 1: GIS code printed to output file (optional)
  - 4: rule code governing transfer of water (1=fraction transferred out, 2=min volume or flow left, 3=exact amount transferred)
- integer, dimension(:), allocatable **parm::inum5s**
- integer, dimension(:), allocatable **parm::inum6s**
- integer, dimension(:), allocatable **parm::inum7s**
- integer, dimension(:), allocatable **parm::inum8s**
- integer, dimension(:), allocatable **parm::subed**
- character(len=10), dimension(:), allocatable **parm::recmonps**
- character(len=10), dimension(:), allocatable **parm::reccnstps**
- character(len=5), dimension(:), allocatable **parm::subnum**
- character(len=4), dimension(:), allocatable **parm::hruno**
- real \*8, dimension(:), allocatable **parm::grwat\_n**
  - Mannings's n for grassed waterway (none)
- real \*8, dimension(:), allocatable **parm::grwat\_i**
  - flag for the simulation of grass waterways (none)
  - = 0 inactive
  - = 1 active
- real \*8, dimension(:), allocatable **parm::grwat\_l**
  - length of grass waterway (km)
- real \*8, dimension(:), allocatable **parm::grwat\_w**
  - average width of grassed waterway (m)
- real \*8, dimension(:), allocatable **parm::grwat\_d**
  - depth of grassed waterway from top of bank to bottom (m)
- real \*8, dimension(:), allocatable **parm::grwat\_s**
  - average slope of grassed waterway channel (m)
- real \*8, dimension(:), allocatable **parm::grwat\_spcon**
  - linear parameter for calculating sediment in grassed waterways (none)
- real \*8, dimension(:), allocatable **parm::tc\_gwat**
- real \*8, dimension(:), allocatable **parm::pot\_volmm**
- real \*8, dimension(:), allocatable **parm::pot\_tilemm**
- real \*8, dimension(:), allocatable **parm::pot\_volxmm**
- real \*8, dimension(:), allocatable **parm::pot\_fr**
  - fraction of HRU area that drains into pothole ( $\text{km}^2/\text{km}^2$ )
- real \*8, dimension(:), allocatable **parm::pot\_tile**



- average daily outflow to main channel from tile flow if drainage tiles are installed in pothole (needed only if current HRU is IPOT) ( $m^3/s$ )*
- real \*8, dimension(:), allocatable **parm::pot\_vol**  
*initial volume of water stored in the depression/impounded area (read in as mm and converted to  $m^3$ ) (needed only if current HRU is IPOT) (mm)*
  - real \*8, dimension(:), allocatable **parm::potsa**
  - real \*8, dimension(:), allocatable **parm::pot\_volx**  
*maximum volume of water stored in the depression/impounded area (read in as mm and converted to  $m^3$ ) (needed only if current HRU is IPOT) (mm)*
  - real \*8, dimension(:), allocatable **parm::wfsh**  
*wetting front matric potential (mm)*
  - real \*8, dimension(:), allocatable **parm::potflwi**
  - real \*8, dimension(:), allocatable **parm::potsedi**
  - real \*8, dimension(:), allocatable **parm::pot\_no3l**  
*nitrate decay rate in impounded area (1/day)*
  - real \*8, dimension(:), allocatable **parm::pot\_nsed**  
*normal sediment concentration in impounded water (needed only if current HRU is IPOT)(mg/L)*
  - real \*8, dimension(:), allocatable **parm::gwno3**  
*nitrate-N concentration in groundwater loading to reach (mg N/L)*
  - real \*8, dimension(:), allocatable **parm::newrti**
  - real \*8, dimension(:), allocatable **parm::fsred**  
*reduction in bacteria loading from filter strip (none)*
  - real \*8, dimension(:), allocatable **parm::pot\_sed**
  - real \*8, dimension(:), allocatable **parm::pot\_no3**
  - real \*8, dimension(:), allocatable **parm::tmpavp**
  - real \*8, dimension(:), allocatable **parm::dis\_stream**  
*average distance to stream (m)*
  - real \*8, dimension(:), allocatable **parm::evpot**  
*pothole evaporation coefficient (none)*
  - real \*8, dimension(:), allocatable **parm::pot\_solpl**
  - real \*8, dimension(:), allocatable **parm::sed\_con**
  - real \*8, dimension(:), allocatable **parm::orgn\_con**
  - real \*8, dimension(:), allocatable **parm::orgp\_con**
  - real \*8, dimension(:), allocatable **parm::pot\_k**  
*hydraulic conductivity of soil surface of pothole defaults to conductivity of upper soil (0.4-10.0) layer*
  - real \*8, dimension(:), allocatable **parm::soln\_con**
  - real \*8, dimension(:), allocatable **parm::solp\_con**
  - real \*8, dimension(:), allocatable **parm::n\_reduc**  
*nitrogen uptake reduction factor (not currently used; defaulted 300.)*
  - real \*8, dimension(:), allocatable **parm::n\_lag**  
*lag coefficient for calculating nitrate concentration in subsurface drains (0.001 - 1.0) (dimensionless)*
  - real \*8, dimension(:), allocatable **parm::n\_ln**  
*power function exponent for calculating nitrate concentration in subsurface drains (1.0 - 3.0) (dimensionless)*
  - real \*8, dimension(:), allocatable **parm::n\_Inco**  
*coefficient for power function for calculating nitrate concentration in subsurface drains (0.5 - 4.0) (dimensionless)*
  - integer, dimension(:), allocatable **parm::ioper**
  - integer, dimension(:), allocatable **parm::ngrwat**
  - real \*8, dimension(:), allocatable **parm::usle\_ls**  
*USLE equation length slope (LS) factor (none)*
  - real \*8, dimension(:), allocatable **parm::filterw**  
*filter strip width for bacteria transport (m)*
  - real \*8, dimension(:), allocatable **parm::phuacc**



- fraction of plant heat units accumulated continuous fertilization is initialized(none)*
- real \*8, dimension(:), allocatable [parm::sumix](#)  
*sum of all tillage mixing efficiencies for HRU operation (none)*
- real \*8, dimension(:), allocatable [parm::epco](#)  
*plant water uptake compensation factor (0-1) (none)*
- real \*8, dimension(:), allocatable [parm::esco](#)  
*soil evaporation compensation factor (0-1) (none)*
- real \*8, dimension(:), allocatable [parm::hru\\_slp](#)  
*average slope steepness (m/m)*
- real \*8, dimension(:), allocatable [parm::slsubbsn](#)  
*average slope length for subbasin (m)*
- real \*8, dimension(:), allocatable [parm::erorgn](#)  
*organic N enrichment ratio, if left blank the model will calculate for every event (none)*
- real \*8, dimension(:), allocatable [parm::erorgp](#)  
*organic P enrichment ratio, if left blank the model will calculate for every event (none)*
- real \*8, dimension(:), allocatable [parm::biomix](#)  
*biological mixing efficiency. Mixing of soil due to activity of earthworms and other soil biota. Mixing is performed at the end of every calendar year (none)*
- real \*8, dimension(:), allocatable [parm::pnd\\_seci](#)
- real \*8, dimension(:), allocatable [parm::canmx](#)  
*maximum canopy storage (mm H2O)*
- real \*8, dimension(:), allocatable [parm::divmax](#)  
*maximum daily irrigation diversion from the reach (when IRRSC=1 or IRR=3): when value is positive the units are mm H2O; when the value is negative, the units are (10<sup>4</sup> m<sup>3</sup> H2O) (mm H2O or 10<sup>4</sup> m<sup>3</sup> H2O)*
- real \*8, dimension(:), allocatable [parm::flowmin](#)  
*minimum instream flow for irrigation diversions when IRRSC=1, irrigation water will be diverted only when streamflow is at or above FLOWMIN (m<sup>3</sup>/s)*
- real \*8, dimension(:), allocatable [parm::usle\\_p](#)  
*USLE equation support practice (P) factor (none)*
- real \*8, dimension(:), allocatable [parm::lat\\_sed](#)  
*sediment concentration in lateral flow (g/L)*
- real \*8, dimension(:), allocatable [parm::rch\\_dakm](#)  
*total drainage area contributing to flow at the outlet (pour point) of the reach in square kilometers (km<sup>2</sup>)*
- real \*8, dimension(:), allocatable [parm::pnd\\_no3s](#)
- real \*8, dimension(:), allocatable [parm::cn1](#)
- real \*8, dimension(:), allocatable [parm::lat\\_ttime](#)  
*lateral flow travel time or exponential of the lateral flow travel time (days or none)*
- real \*8, dimension(:), allocatable [parm::cn2](#)  
*SCS runoff curve number for moisture condition II (none)*
- real \*8, dimension(:), allocatable [parm::flowfr](#)  
*fraction of available flow in reach that is allowed to be applied to the HRU (none)*
- real \*8, dimension(:), allocatable [parm::sol\\_zmx](#)  
*maximum rooting depth (mm)*
- real \*8, dimension(:), allocatable [parm::tile\\_ttime](#)  
*exponential of the tile flow travel time (none)*
- real \*8, dimension(:), allocatable [parm::slsoil](#)  
*slope length for lateral subsurface flow (m)*
- real \*8, dimension(:), allocatable [parm::gwminp](#)  
*soluble P concentration in groundwater loading to reach (mg P/L)*
- real \*8, dimension(:), allocatable [parm::sol\\_cov](#)  
*amount of residue on soil surface (kg/ha)*
- real \*8, dimension(:), allocatable [parm::sed\\_stl](#)

- fraction of sediment remaining suspended in impoundment after settling for one day (kg/kg)*

  - real \*8, dimension(:), allocatable [parm::ov\\_n](#)
- Manning's "n" value for overland flow (none)*

  - real \*8, dimension(:), allocatable [parm::pnd\\_no3](#)
- amount of nitrate in pond (kg N)*

  - real \*8, dimension(:), allocatable [parm::pnd\\_solp](#)
- amount of soluble P in pond (kg P)*

  - real \*8, dimension(:), allocatable [parm::yldanu](#)
- annual yield (dry weight) in the HRU (metric tons/ha)*

  - real \*8, dimension(:), allocatable [parm::driftco](#)
- coefficient for pesticide drift directly onto stream (none)*

  - real \*8, dimension(:), allocatable [parm::pnd\\_orgn](#)
- amount of organic N in pond (kg N)*

  - real \*8, dimension(:), allocatable [parm::pnd\\_orgp](#)
- amount of organic P in pond (kg P)*

  - real \*8, dimension(:), allocatable [parm::cn3](#)
- real \*8, dimension(:), allocatable [parm::twlpnd](#)*
- real \*8, dimension(:), allocatable [parm::twlwet](#)*
- real \*8, dimension(:), allocatable [parm::hru\\_fr](#)*
- fraction of subbasin area contained in HRU (km<sup>2</sup>/km<sup>2</sup>)*

  - real \*8, dimension(:), allocatable [parm::sol\\_sumul](#)
- amount of water held in soil profile at saturation (mm H2O)*

  - real \*8, dimension(:), allocatable [parm::pnd\\_chla](#)
- real \*8, dimension(:), allocatable [parm::hru\\_km](#)*
- area of HRU in square kilometers (km<sup>2</sup>)*

  - real \*8, dimension(:), allocatable [parm::bio\\_ms](#)
- cover/crop biomass (kg/ha)*

  - real \*8, dimension(:), allocatable [parm::sol\\_alb](#)
- albedo when soil is moist (none)*

  - real \*8, dimension(:), allocatable [parm::strsw](#)
- real \*8, dimension(:), allocatable [parm::pnd\\_fr](#)*
- fraction of HRU/subbasin area that drains into ponds (none)*

  - real \*8, dimension(:), allocatable [parm::pnd\\_k](#)
- hydraulic conductivity through bottom of ponds (mm/hr)*

  - real \*8, dimension(:), allocatable [parm::pnd\\_psa](#)
- surface area of ponds when filled to principal spillway (ha)*

  - real \*8, dimension(:), allocatable [parm::pnd\\_pvol](#)
- runoff volume from catchment area needed to fill the ponds to the principal spillway (UNIT CHANGE!) (10<sup>4</sup> m<sup>3</sup> H2O or m<sup>3</sup> H2O)*

  - real \*8, dimension(:), allocatable [parm::pnd\\_esa](#)
- surface area of ponds when filled to emergency spillway (ha)*

  - real \*8, dimension(:), allocatable [parm::pnd\\_evol](#)
- runoff volume from catchment area needed to fill the ponds to the emergency spillway (UNIT CHANGE!) (10<sup>4</sup> m<sup>3</sup> H2O or m<sup>3</sup> H2O)*

  - real \*8, dimension(:), allocatable [parm::pnd\\_vol](#)
- volume of water in ponds (UNIT CHANGE!) (10<sup>4</sup> m<sup>3</sup> H2O or m<sup>3</sup> H2O)*

  - real \*8, dimension(:), allocatable [parm::yldaa](#)
- average annual yield in the HRU (metric tons)*

  - real \*8, dimension(:), allocatable [parm::pnd\\_nsed](#)
- normal sediment concentration in pond water (UNIT CHANGE!) (mg/kg or kg/kg)*

  - real \*8, dimension(:), allocatable [parm::pnd\\_sed](#)

- sediment concentration in pond water (UNIT CHANGE!) (mg/kg or kg/kg)*
- real \*8, dimension(:), allocatable **parm::strsa**
- real \*8, dimension(:), allocatable **parm::dep\_imp**
- real \*8, dimension(:), allocatable **parm::evpnd**
- real \*8, dimension(:), allocatable **parm::evwet**
- real \*8, dimension(:), allocatable **parm::wet\_fr**
- fraction of HRU/subbasin area that drains into wetlands (none)*
- real \*8, dimension(:), allocatable **parm::wet\_k**
- hydraulic conductivity of bottom of wetlands (mm/hr)*
- real \*8, dimension(:), allocatable **parm::wet\_nsa**
- surface area of wetlands in subbasin at normal water level (ha)*
- real \*8, dimension(:), allocatable **parm::wet\_nvol**
- runoff volume from catchment area needed to fill wetlands to normal water level (UNIT CHANGE!) ( $10^4 \text{ m}^3 \text{ H}_2\text{O}$  or  $\text{m}^3 \text{ H}_2\text{O}$ )*
- integer, dimension(:), allocatable **parm::iwetgw**
- integer, dimension(:), allocatable **parm::iwetile**
- real \*8, dimension(:), allocatable **parm::wet\_mxsa**
- surface area of wetlands at maximum water level (ha)*
- real \*8, dimension(:), allocatable **parm::wet\_mxvol**
- runoff volume from catchment area needed to fill wetlands to maximum water level (UNIT CHANGE!) ( $10^4 \text{ m}^3 \text{ H}_2\text{O}$  or  $\text{m}^3 \text{ H}_2\text{O}$ )*
- real \*8, dimension(:), allocatable **parm::wet\_vol**
- volume of water in wetlands (UNIT CHANGE!) ( $10^4 \text{ m}^3 \text{ H}_2\text{O}$  or  $\text{m}^3 \text{ H}_2\text{O}$ )*
- real \*8, dimension(:), allocatable **parm::wet\_nsed**
- normal sediment concentration in wetland water (UNIT CHANGE!) (mg/kg or kg/kg)*
- real \*8, dimension(:), allocatable **parm::wet\_sed**
- sediment concentration in wetland water (UNIT CHANGE!) (mg/L or kg/L)*
- real \*8, dimension(:), allocatable **parm::bp1**
- 1st shape parameter for pond surface area equation (none)*
- real \*8, dimension(:), allocatable **parm::bp2**
- 2nd shape parameter for the pond surface area equation (none)*
- real \*8, dimension(:), allocatable **parm::smx**
- real \*8, dimension(:), allocatable **parm::sci**
- real \*8, dimension(:), allocatable **parm::bw1**
- 1st shape parameter for the wetland surface area equation (none)*
- real \*8, dimension(:), allocatable **parm::bw2**
- 2nd shape parameter for the wetland surface area equation (none)*
- real \*8, dimension(:), allocatable **parm::bactpq**
- real \*8, dimension(:), allocatable **parm::bactp\_plt**
- real \*8, dimension(:), allocatable **parm::bactlp\_plt**
- real \*8, dimension(:), allocatable **parm::cnday**
- real \*8, dimension(:), allocatable **parm::auto\_eff**
- fertilizer application efficiency calculated as the amount of N applied divided by the amount of N removed at harvest (none)*
- real \*8, dimension(:), allocatable **parm::secciw**
- water clarity coefficient for wetland (none)*
- real \*8, dimension(:), allocatable **parm::sol\_sw**
- amount of water stored in soil profile on any given day (mm H<sub>2</sub>O)*
- real \*8, dimension(:), allocatable **parm::bactlpq**
- real \*8, dimension(:), allocatable **parm::chlaw**
- chlorophyll-a production coefficient for wetland (none)*
- real \*8, dimension(:), allocatable **parm::bactps**

- real \*8, dimension(:), allocatable **parm::bactlps**
- real \*8, dimension(:), allocatable **parm::tmpav**
- real \*8, dimension(:), allocatable **parm::sno\_hru**  
*amount of water stored as snow (mm H2O)*
- real \*8, dimension(:), allocatable **parm::wet\_orgn**  
*amount of organic N in wetland (kg N)*
- real \*8, dimension(:), allocatable **parm::subp**
- real \*8, dimension(:), allocatable **parm::hru\_ra**
- real \*8, dimension(:), allocatable **parm::rsdin**  
*initial residue cover (kg/ha)*
- real \*8, dimension(:), allocatable **parm::tmx**
- real \*8, dimension(:), allocatable **parm::tmn**
- real \*8, dimension(:), allocatable **parm::tmp\_hi**
- real \*8, dimension(:), allocatable **parm::tmp\_lo**
- real \*8, dimension(:), allocatable **parm::usle\_k**  
*USLE equation soil erodibility (K) factor (none)*
- real \*8, dimension(:), allocatable **parm::tconc**  
*time of concentration for HRU (hour)*
- real \*8, dimension(:), allocatable **parm::rwt**
- real \*8, dimension(:), allocatable **parm::olai**
- real \*8, dimension(:), allocatable **parm::hru\_rmx**
- real \*8, dimension(:), allocatable **parm::usle\_cfac**
- real \*8, dimension(:), allocatable **parm::usle\_eifac**
- real \*8, dimension(:), allocatable **parm::sol\_sumfc**  
*amount of water held in soil profile at field capacity (mm H2O)*
- real \*8, dimension(:), allocatable **parm::t\_ov**  
*time for flow from farthest point in subbasin to enter a channel (hour)*
- real \*8, dimension(:), allocatable **parm::anano3**  
*total amount of NO3 applied during the year in auto-fertilization (kg N/ha)*
- real \*8, dimension(:), allocatable **parm::aird**
- real \*8, dimension(:), allocatable **parm::wet\_orgp**  
*amount of organic P in wetland (kg P)*
- real \*8, dimension(:), allocatable **parm::sol\_avpor**  
*average porosity for entire soil profile (none)*
- real \*8, dimension(:), allocatable **parm::usle\_mult**  
*product of USLE K,P,LS,exp(rock) (none)*
- real \*8, dimension(:), allocatable **parm::aairr**
- real \*8, dimension(:), allocatable **parm::cht**
- real \*8, dimension(:), allocatable **parm::u10**
- real \*8, dimension(:), allocatable **parm::rhd**
- real \*8, dimension(:), allocatable **parm::lai\_aamx**  
*maximum leaf area index for the entire period of simulation in the HRU (none)*
- real \*8, dimension(:), allocatable **parm::shallirr**
- real \*8, dimension(:), allocatable **parm::deepirr**
- real \*8, dimension(:), allocatable **parm::ch\_l1**  
*longest tributary channel length in subbasin (km)*
- real \*8, dimension(:), allocatable **parm::wet\_no3**  
*amount of nitrate in wetland (kg N)*
- real \*8, dimension(:), allocatable **parm::canstor**
- real \*8, dimension(:), allocatable **parm::ovrlnd**
- real \*8, dimension(:), allocatable **parm::irr\_mx**  
*maximum irrigation amount per auto application (mm)*

- real \*8, dimension(:), allocatable [parm::auto\\_wstr](#)  
*water stress factor which triggers auto irrigation (none or mm)*
- real \*8, dimension(:), allocatable [parm::cfrt\\_id](#)  
*fertilizer/manure id number from database (none)*
- real \*8, dimension(:), allocatable [parm::cfrt\\_kg](#)  
*amount of fertilzier applied to HRU on a given day (kg/ha)*
- real \*8, dimension(:), allocatable [parm::cpst\\_id](#)
- real \*8, dimension(:), allocatable [parm::cpst\\_kg](#)
- real \*8, dimension(:), allocatable [parm::irr\\_asq](#)  
*surface runoff ratio*
- real \*8, dimension(:), allocatable [parm::irr\\_eff](#)
- real \*8, dimension(:), allocatable [parm::irrsq](#)  
*surface runoff ratio (0-1) .1 is 10% surface runoff (frac)*
- real \*8, dimension(:), allocatable [parm::irrefm](#)
- real \*8, dimension(:), allocatable [parm::irrsalt](#)
- real \*8, dimension(:), allocatable [parm::bio\\_eat](#)  
*dry weight of biomass removed by grazing daily ((kg/ha)/day)*
- real \*8, dimension(:), allocatable [parm::bio\\_trmp](#)  
*dry weight of biomass removed by trampling daily ((kg/ha)/day)*
- integer, dimension(:), allocatable [parm::ifrt\\_freq](#)
- integer, dimension(:), allocatable [parm::ipst\\_freq](#)
- integer, dimension(:), allocatable [parm::irr\\_noa](#)
- integer, dimension(:), allocatable [parm::irr\\_sc](#)
- integer, dimension(:), allocatable [parm::irr\\_no](#)
- integer, dimension(:), allocatable [parm::imp\\_trig](#)  
*release/impound action code (none):*  
*0 begin impounding water*  
*1 release impounded water*
- integer, dimension(:), allocatable [parm::fert\\_days](#)
- integer, dimension(:), allocatable [parm::irr\\_sca](#)
- integer, dimension(:), allocatable [parm::idplt](#)  
*land cover/crop identification code for first crop grown in HRU (the only crop if there is no rotation) (none)*
- integer, dimension(:), allocatable [parm::pest\\_days](#)
- integer, dimension(:), allocatable [parm::wstrs\\_id](#)
- real \*8, dimension(:,:), allocatable [parm::bio\\_aahv](#)
- real \*8, dimension(:), allocatable [parm::cumei](#)
- real \*8, dimension(:), allocatable [parm::cumeira](#)
- real \*8, dimension(:), allocatable [parm::cumrt](#)
- real \*8, dimension(:), allocatable [parm::cumrai](#)
- real \*8, dimension(:), allocatable [parm::wet\\_solp](#)  
*amount of soluble P in wetland (kg P)*
- real \*8, dimension(:), allocatable [parm::wet\\_no3s](#)
- real \*8, dimension(:), allocatable [parm::wet\\_chla](#)
- real \*8, dimension(:), allocatable [parm::wet\\_seci](#)
- real \*8, dimension(:), allocatable [parm::pnd\\_no3g](#)
- real \*8, dimension(:), allocatable [parm::pstsol](#)
- real \*8, dimension(:), allocatable [parm::delay](#)  
*groundwater delay: time required for water leaving the bottom of the root zone to reach the shallow aquifer (days)*
- real \*8, dimension(:), allocatable [parm::gwht](#)  
*groundwater height (m)*
- real \*8, dimension(:), allocatable [parm::gw\\_q](#)
- real \*8, dimension(:), allocatable [parm::pnd\\_solpg](#)
- real \*8, dimension(:), allocatable [parm::alpha\\_bf](#)

- alpha factor for groundwater recession curve (1/days)*

  - real \*8, dimension(:), allocatable [parm::alpha\\_bfe](#)  
 *$\exp(-\alpha_b f)$  (none)*
- specific yield for shallow aquifer ( $m^3/m^3$ )*

  - real \*8, dimension(:), allocatable [parm::gw\\_spyld](#)
- alpha factor for groundwater recession curve of the deep aquifer (1/days)*

  - real \*8, dimension(:), allocatable [parm::alpha\\_bfe\\_d](#)  
 *$\exp(-\alpha_b f_d)$  for deep aquifer (none)*
- real \*8, dimension(:), allocatable **parm::gw\_qdeep**
- $\exp(-1/\text{delay})$  (none)*

  - real \*8, dimension(:), allocatable [parm::gw\\_delaye](#)
- revap coeff: this variable controls the amount of water moving from the shallow aquifer to the root zone as a result of soil moisture depletion (none)*

  - real \*8, dimension(:), allocatable [parm::gw\\_revap](#)
- recharge to deep aquifer: the fraction of root zone percolation that reaches the deep aquifer (none)*

  - real \*8, dimension(:), allocatable [parm::rchrg\\_dp](#)
- fraction of porosity from which anions are excluded*

  - real \*8, dimension(:), allocatable [parm::anion\\_excl](#)
- threshold depth of water in shallow aquifer required to allow revap to occur (mm H2O)*

  - real \*8, dimension(:), allocatable [parm::revapmn](#)
- real \*8, dimension(:), allocatable **parm::rchrg**
- minimum plant biomass for grazing (kg/ha)*

  - real \*8, dimension(:), allocatable [parm::bio\\_min](#)
- initial HRU soil water content expressed as fraction of field capacity (none)*

  - real \*8, dimension(:), allocatable [parm::ffc](#)
- real \*8, dimension(:), allocatable **parm::surqsolp**
- depth of water in deep aquifer (mm H2O)*

  - real \*8, dimension(:), allocatable [parm::deepst](#)
- depth of water in shallow aquifer (mm H2O)*

  - real \*8, dimension(:), allocatable [parm::shallst](#)
- real \*8, dimension(:), allocatable **parm::cklsp**
- real \*8, dimension(:), allocatable **parm::wet\_solpg**
- real \*8, dimension(:), allocatable **parm::rchrg\_src**
- filter strip trapping efficiency (used for everything but bacteria) (none)*

  - real \*8, dimension(:), allocatable [parm::trapeff](#)
- average bulk density for soil profile ( $Mg/m^3$ )*

  - real \*8, dimension(:), allocatable [parm::sol\\_avbd](#)
- real \*8, dimension(:), allocatable **parm::wet\_no3g**
- time to drain soil to field capacity yield used in autofertilization (hours)*

  - real \*8, dimension(:), allocatable [parm::tdrain](#)
- threshold depth of water in shallow aquifer required before groundwater flow will occur (mm H2O)*

  - real \*8, dimension(:), allocatable [parm::gwqmn](#)
- real \*8, dimension(:), allocatable **parm::pplnt**
- real \*8, dimension(:), allocatable **parm::snotmp**
- drain tile lag time: the amount of time between the transfer of water from the soil to the drain tile and the release of the water from the drain tile to the reach (hours)*

  - real \*8, dimension(:), allocatable [parm::gdrain](#)
- depth to the sub-surface drain (mm)*

  - real \*8, dimension(:), allocatable [parm::ddrain](#)
- crack volume potential of soil (none)*

  - real \*8, dimension(:), allocatable [parm::sol\\_crk](#)

- real \*8, dimension(:), allocatable **parm::brt**  
*fraction of surface runoff within the subbasin which takes 1 day or less to reach the subbasin outlet (none)*
- real \*8, dimension(:), allocatable **parm::dayl**
- real \*8, dimension(:), allocatable **parm::sstmaxd**  
*static maximum depressional storage; read from .sdr (mm)*
- real \*8, dimension(:), allocatable **parm::re**  
*effective radius of drains (mm)*
- real \*8, dimension(:), allocatable **parm::sdrain**  
*distance between two drain tubes or tiles (mm)*
- real \*8, dimension(:), allocatable **parm::ddrain\_hru**
- real \*8, dimension(:), allocatable **parm::drain\_co**  
*drainage coefficient (mm/day)*
- real \*8, dimension(:), allocatable **parm::latksatf**  
*multiplication factor to determine  $conk(j1,j)$  from  $sol\_k(j1,j)$  for HRU (none)*
- real \*8, dimension(:), allocatable **parm::pc**  
*pump capacity (default pump capacity = 1.042mm/hr or 25mm/day) (mm/hr)*
- real \*8, dimension(:), allocatable **parm::stmaxd**
- real \*8, dimension(:), allocatable **parm::twash**
- real \*8, dimension(:), allocatable **parm::rnd2**
- real \*8, dimension(:), allocatable **parm::rnd3**
- real \*8, dimension(:), allocatable **parm::sol\_cns**
- real \*8, dimension(:), allocatable **parm::doxq**
- real \*8, dimension(:), allocatable **parm::rnd8**
- real \*8, dimension(:), allocatable **parm::rnd9**
- real \*8, dimension(:), allocatable **parm::perc**
- real \*8, dimension(:), allocatable **parm::sol\_sumwp**
- real \*8, dimension(:), allocatable **parm::tauton**
- real \*8, dimension(:), allocatable **parm::tautop**
- real \*8, dimension(:), allocatable **parm::cbodu**
- real \*8, dimension(:), allocatable **parm::chl\_a**
- real \*8, dimension(:), allocatable **parm::qdr**
- real \*8, dimension(:), allocatable **parm::tfertn**
- real \*8, dimension(:), allocatable **parm::tfertp**
- real \*8, dimension(:), allocatable **parm::tgrazn**
- real \*8, dimension(:), allocatable **parm::tgrazp**
- real \*8, dimension(:), allocatable **parm::latno3**
- real \*8, dimension(:), allocatable **parm::latq**
- real \*8, dimension(:), allocatable **parm::minpgw**
- real \*8, dimension(:), allocatable **parm::no3gw**
- real \*8, dimension(:), allocatable **parm::nplnt**
- real \*8, dimension(:), allocatable **parm::tileq**
- real \*8, dimension(:), allocatable **parm::tileno3**
- real \*8, dimension(:), allocatable **parm::sedminpa**
- real \*8, dimension(:), allocatable **parm::sedminps**
- real \*8, dimension(:), allocatable **parm::sedorgn**
- real \*8, dimension(:), allocatable **parm::sedorgp**
- real \*8, dimension(:), allocatable **parm::sedyld**
- real \*8, dimension(:), allocatable **parm::sepbtm**
- real \*8, dimension(:), allocatable **parm::strsn**
- real \*8, dimension(:), allocatable **parm::strsp**
- real \*8, dimension(:), allocatable **parm::strstmp**
- real \*8, dimension(:), allocatable **parm::surfq**
- real \*8, dimension(:), allocatable **parm::surqno3**



- real \*8, dimension(:), allocatable [parm::hru\\_ha](#)  
*area of HRU in hectares (ha)*
- real \*8, dimension(:), allocatable [parm::hru\\_dafr](#)  
*fraction of total watershed area contained in HRU (km<sup>2</sup>/km<sup>2</sup>)*
- real \*8, dimension(:), allocatable **parm::tcfrtn**
- real \*8, dimension(:), allocatable **parm::tcfrtp**
- real \*8, dimension(:), allocatable [parm::drydep\\_no3](#)  
*atmospheric dry deposition of nitrates (kg/ha/yr)*
- real \*8, dimension(:), allocatable [parm::drydep\\_nh4](#)  
*atmospheric dry deposition of ammonia (kg/ha/yr)*
- real \*8, dimension(:), allocatable [parm::bio\\_yrms](#)  
*annual biomass (dry weight) in the HRU (metric tons/ha)*
- real \*8, dimension(:), allocatable [parm::phubase](#)  
*base zero total heat units (used when no land cover is growing) (heat units)*
- real \*8, dimension(:), allocatable **parm::hvstiadj**
- real \*8, dimension(:), allocatable [parm::laiday](#)  
*leaf area index (m<sup>2</sup>/m<sup>2</sup>)*
- real \*8, dimension(:), allocatable [parm::chlap](#)  
*chlorophyll-a production coefficient for pond (none)*
- real \*8, dimension(:), allocatable **parm::laimxfr**
- real \*8, dimension(:), allocatable **parm::pnd\_psed**
- real \*8, dimension(:), allocatable [parm::seccip](#)  
*water clarity coefficient for pond (none)*
- real \*8, dimension(:), allocatable **parm::wet\_psed**
- real \*8, dimension(:), allocatable **parm::plantn**
- real \*8, dimension(:), allocatable **parm::plt\_et**
- real \*8, dimension(:), allocatable [parm::bio\\_aams](#)  
*average annual biomass in the HRU (metric tons)*
- real \*8, dimension(:), allocatable **parm::plt\_pet**
- real \*8, dimension(:), allocatable **parm::plantp**
- real \*8, dimension(:), allocatable [parm::dormhr](#)  
*time threshold used to define dormant period for plant (when daylength is within the time specified by dl from the minimum daylength for the area, the plant will go dormant) (hour)*
- real \*8, dimension(:), allocatable [parm::lai\\_yrmx](#)  
*maximum leaf area index for the year in the HRU (none)*
- real \*8, dimension(:), allocatable **parm::bio\_aamx**
- real \*8, dimension(:), allocatable **parm::lat\_pst**
- real \*8, dimension(:), allocatable [parm::fld\\_fr](#)  
*fraction of HRU area that drains into floodplain (km<sup>2</sup>/km<sup>2</sup>)*
- real \*8, dimension(:), allocatable **parm::orig\_snohru**
- real \*8, dimension(:), allocatable **parm::orig\_potvol**
- real \*8, dimension(:), allocatable **parm::orig\_alai**
- real \*8, dimension(:), allocatable **parm::orig\_bioms**
- real \*8, dimension(:), allocatable **parm::pltfr\_n**
- real \*8, dimension(:), allocatable **parm::orig\_phuacc**
- real \*8, dimension(:), allocatable **parm::orig\_sumix**
- real \*8, dimension(:), allocatable **parm::pltfr\_p**
- real \*8, dimension(:), allocatable [parm::phu\\_plt](#)  
*total number of heat units to bring plant to maturity (heat units)*
- real \*8, dimension(:), allocatable **parm::orig\_phu**
- real \*8, dimension(:), allocatable **parm::orig\_shallst**
- real \*8, dimension(:), allocatable **parm::orig\_deepst**



- real \*8, dimension(:), allocatable [parm::rip\\_fr](#)  
*fraction of HRU area that drains into riparian zone ( $\text{km}^2/\text{km}^2$ )*
- real \*8, dimension(:), allocatable **parm::orig\_pndvol**
- real \*8, dimension(:), allocatable **parm::orig\_pndsed**
- real \*8, dimension(:), allocatable **parm::orig\_pndno3**
- real \*8, dimension(:), allocatable **parm::orig\_pndsolv**
- real \*8, dimension(:), allocatable **parm::orig\_pndorgn**
- real \*8, dimension(:), allocatable **parm::orig\_pndorgp**
- real \*8, dimension(:), allocatable **parm::orig\_wetvol**
- real \*8, dimension(:), allocatable **parm::orig\_wetsed**
- real \*8, dimension(:), allocatable **parm::orig\_wetno3**
- real \*8, dimension(:), allocatable **parm::orig\_wetsolv**
- real \*8, dimension(:), allocatable **parm::orig\_wetorgn**
- real \*8, dimension(:), allocatable **parm::orig\_wetorgp**
- real \*8, dimension(:), allocatable **parm::orig\_solcov**
- real \*8, dimension(:), allocatable **parm::orig\_solsw**
- real \*8, dimension(:), allocatable **parm::orig\_potno3**
- real \*8, dimension(:), allocatable **parm::orig\_potsed**
- real \*8, dimension(:), allocatable **parm::wtab**
- real \*8, dimension(:), allocatable **parm::wtab\_mn**
- real \*8, dimension(:), allocatable **parm::wtab\_mx**
- real \*8, dimension(:), allocatable [parm::shallst\\_n](#)  
*nitrate concentration in shallow aquifer converted to kg/ha (ppm NO3-N)*
- real \*8, dimension(:), allocatable **parm::gw\_nloss**
- real \*8, dimension(:), allocatable **parm::rchrg\_n**
- real \*8, dimension(:), allocatable **parm::det\_san**
- real \*8, dimension(:), allocatable **parm::det\_sil**
- real \*8, dimension(:), allocatable **parm::det\_cla**
- real \*8, dimension(:), allocatable **parm::det\_sag**
- real \*8, dimension(:), allocatable **parm::det\_lag**
- real \*8, dimension(:), allocatable [parm::afrt\\_surface](#)  
*fraction of fertilizer which is applied to top 10 mm of soil (the remaining fraction is applied to first soil layer) (none)*
- real \*8, dimension(:), allocatable **parm::tnylda**
- real \*8 [parm::ftr\\_surface](#)  
*fraction of fertilizer which is applied to the top 10 mm of soil (the remaining fraction is applied to the first soil layer) (none)*
- real \*8, dimension(:), allocatable [parm::auto\\_nyr](#)  
*maximum NO3-N content allowed to be applied in one year (kg NO3-N/ha)*
- real \*8, dimension(:), allocatable [parm::auto\\_napp](#)  
*maximum NO3-N content allowed in one fertilizer application (kg NO3-N/ha)*
- real \*8, dimension(:), allocatable [parm::auto\\_nstrs](#)  
*nitrogen stress factor which triggers auto fertilization (none)*
- real \*8, dimension(:), allocatable **parm::manure\_kg**
- real \*8, dimension(:,:), allocatable **parm::rcn\_mo**
- real \*8, dimension(:,:), allocatable **parm::rammo\_mo**
- real \*8, dimension(:,:), allocatable **parm::drydep\_no3\_mo**
- real \*8, dimension(:,:), allocatable **parm::drydep\_nh4\_mo**
- real \*8, dimension(:), allocatable **parm::rcn\_d**
- real \*8, dimension(:), allocatable **parm::rammo\_d**
- real \*8, dimension(:), allocatable **parm::drydep\_no3\_d**
- real \*8, dimension(:), allocatable **parm::drydep\_nh4\_d**
- real \*8, dimension(:,:), allocatable **parm::yldn**
- real \*8, dimension(:,:), allocatable **parm::gwati**
- real \*8, dimension(:,:), allocatable **parm::gwatn**

- real \*8, dimension(:,:), allocatable **parm::gwatl**
- real \*8, dimension(:,:), allocatable **parm::gwatw**
- real \*8, dimension(:,:), allocatable **parm::gwatd**
- real \*8, dimension(:,:), allocatable **parm::gwatveg**
- real \*8, dimension(:,:), allocatable **parm::gwata**
- real \*8, dimension(:,:), allocatable **parm::gwats**
- real \*8, dimension(:,:), allocatable **parm::gwatspcon**
- real \*8, dimension(:,:), allocatable **parm::rfqeo\_30d**
- real \*8, dimension(:,:), allocatable **parm::eo\_30d**
- real \*8, dimension(:), allocatable **parm::psetlp1**  
*phosphorus settling rate for 1st season (m/day)*
- real \*8, dimension(:), allocatable **parm::psetlp2**  
*phosphorus settling rate for 2nd season (m/day)*
- real \*8, dimension(:,:), allocatable **parm::wgncur**
- real \*8, dimension(:,:), allocatable **parm::wgnold**
- real \*8, dimension(:,:), allocatable **parm::wrt**
- real \*8, dimension(:,:), allocatable **parm::pst\_enr**  
*pesticide enrichment ratio (none)*
- real \*8, dimension(:,:), allocatable **parm::zdb**
- real \*8, dimension(:,:), allocatable **parm::pst\_surq**
- real \*8, dimension(:,:), allocatable **parm::plt\_pst**  
*pesticide on plant foliage (kg/ha)*
- real \*8, dimension(:), allocatable **parm::psetlw1**  
*phosphorus settling rate for 1st season (m/day)*
- real \*8, dimension(:), allocatable **parm::psetlw2**  
*phosphorus settling rate for 2nd season (m/day)*
- real \*8, dimension(:,:), allocatable **parm::pst\_sed**
- real \*8, dimension(:,:), allocatable **parm::wupnd**  
*average daily water removal from the pond for the month ( $10^4 \text{ m}^3/\text{day}$ )*
- real \*8, dimension(:,:), allocatable **parm::phi**  
*phi(1,:) cross-sectional area of flow at bankfull depth ( $\text{m}^2$ ) phi(2,:) (none) phi(3,:) (none) phi(4,:) (none) phi(5,:) (none) phi(6,:) bottom width of main channel (m) phi(7,:) depth of water when reach is at bankfull depth (m) phi(8,:) average velocity when reach is at bankfull depth (m/s) phi(9,:) wave celerity when reach is at bankfull depth (m/s) phi(10,:) storage time constant for reach at bankfull depth (ratio of storage to discharge) (hour) phi(11,:) average velocity when reach is at 0.1 bankfull depth (low flow) (m/s) phi(12,:) wave celerity when reach is at 0.1 bankfull depth (low flow) (m/s) phi(13,:) storage time constant for reach at 0.1 bankfull depth (low flow) (ratio of storage to discharge) (hour)*
- real \*8, dimension(:,:), allocatable **parm::pcpband**
- real \*8, dimension(:,:), allocatable **parm::tavband**
- real \*8, dimension(:,:), allocatable **parm::wat\_phi**
- real \*8, dimension(:,:), allocatable **parm::snoeb**  
*initial snow water content in elevation band (mm H<sub>2</sub>O)*
- real \*8, dimension(:,:), allocatable **parm::wudeep**  
*average daily water removal from the deep aquifer for the month ( $10^4 \text{ m}^3/\text{day}$ )*
- real \*8, dimension(:,:), allocatable **parm::wushal**  
*average daily water removal from the shallow aquifer for the month ( $10^4 \text{ m}^3/\text{day}$ )*
- real \*8, dimension(:,:), allocatable **parm::tmnband**
- real \*8, dimension(:), allocatable **parm::bss1**
- real \*8, dimension(:), allocatable **parm::bss2**
- real \*8, dimension(:), allocatable **parm::bss3**
- real \*8, dimension(:), allocatable **parm::bss4**
- real \*8, dimension(:), allocatable **parm::nsetlw1**  
*nitrogen settling rate for 1st season (m/day)*

- real \*8, dimension(:), allocatable [parm::nsetlw2](#)  
*nitrogen settling rate for 2nd season (m/day)*
- real \*8, dimension(:,:), allocatable **parm::snotmpeb**
- real \*8, dimension(:,:), allocatable **parm::surf\_bs**
- real \*8, dimension(:), allocatable [parm::nsetlp1](#)  
*nitrogen settling rate for 1st season (m/day)*
- real \*8, dimension(:), allocatable [parm::nsetlp2](#)  
*nitrogen settling rate for 2nd season (m/day)*
- real \*8, dimension(:,:), allocatable **parm::tmxband**
- real \*8, dimension(:,:), allocatable **parm::rainsub**
- real \*8, dimension(:,:), allocatable **parm::frad**
- real \*8, dimension(:), allocatable **parm::rstpbsb**
- real \*8, dimension(:,:), allocatable **parm::orig\_snoeb**
- real \*8, dimension(:,:), allocatable **parm::orig\_pltpst**
- real \*8, dimension(:,:), allocatable **parm::terr\_p**
- real \*8, dimension(:,:), allocatable **parm::terr\_cn**
- real \*8, dimension(:,:), allocatable **parm::terr\_sl**
- real \*8, dimension(:,:), allocatable **parm::drain\_d**
- real \*8, dimension(:,:), allocatable **parm::drain\_t**
- real \*8, dimension(:,:), allocatable **parm::drain\_g**
- real \*8, dimension(:,:), allocatable **parm::drain\_idep**
- real \*8, dimension(:,:), allocatable **parm::cont\_cn**
- real \*8, dimension(:,:), allocatable **parm::cont\_p**
- real \*8, dimension(:,:), allocatable **parm::filt\_w**
- real \*8, dimension(:,:), allocatable **parm::strip\_n**
- real \*8, dimension(:,:), allocatable **parm::strip\_cn**
- real \*8, dimension(:,:), allocatable **parm::strip\_c**
- real \*8, dimension(:,:), allocatable **parm::strip\_p**
- real \*8, dimension(:,:), allocatable **parm::fire\_cn**
- real \*8, dimension(:,:), allocatable **parm::cropno\_upd**
- real \*8, dimension(:,:), allocatable **parm::hi\_upd**
- real \*8, dimension(:,:), allocatable **parm::laimx\_upd**
- real \*8, dimension(:,:), allocatable [parm::phug](#)  
*fraction of plant heat units at which grazing begins (none)*
- real \*8, dimension(:,:), allocatable **parm::pst\_lag**
- integer, dimension(:), allocatable [parm::hrupest](#)  
*pesticide use flag (none)*  
*0: no pesticides used in HRU*  
*1: pesticides used in HRU*
- integer, dimension(:), allocatable [parm::nrelease](#)  
*sequence number of impound/release operation within the year (none)*
- integer, dimension(:), allocatable **parm::swtrg**
- integer, dimension(:), allocatable [parm::nrot](#)  
*number of years of rotation (none)*
- integer, dimension(:), allocatable [parm::nfert](#)  
*sequence number of fertilizer application within the year (none)*
- integer, dimension(:), allocatable **parm::nro**
- integer, dimension(:), allocatable [parm::igro](#)  
*land cover status code (none). This code informs the model whether or not a land cover is growing at the beginning of the simulation*  
*0 no land cover growing*  
*1 land cover growing*
- integer, dimension(:), allocatable [parm::ipnd1](#)  
*beginning month of nutrient settling season (none)*

- integer, dimension(:), allocatable [parm::ipnd2](#)  
*ending month of nutrient settling season (none)*
- integer, dimension(:), allocatable [parm::nair](#)  
*sequence number of auto-irrigation application within the year (none)*
- integer, dimension(:), allocatable [parm::iflod1](#)  
*beginning month of non-flood season (none)*
- integer, dimension(:), allocatable [parm::iflod2](#)  
*ending month of non-flood season (none)*
- integer, dimension(:), allocatable [parm::ndtarg](#)  
*number of days required to reach target storage from current pond storage (none)*
- integer, dimension(:), allocatable [parm::nirr](#)  
*sequence number of irrigation application within the year (none)*
- integer, dimension(:), allocatable [parm::iafrttyp](#)
- integer, dimension(:), allocatable [parm::nstress](#)
- integer, dimension(:), allocatable [parm::igrotree](#)
- integer, dimension(:), allocatable [parm::grz\\_days](#)
- integer, dimension(:), allocatable [parm::nmgt](#)  
*management code (for GIS output only) (none)*
- integer, dimension(:), allocatable [parm::nafert](#)  
*sequence number of auto-fert application within the year (none)*
- integer, dimension(:), allocatable [parm::nsweep](#)  
*sequence number of street sweeping operation within the year (none)*
- integer, dimension(:), allocatable [parm::icr](#)
- integer, dimension(:), allocatable [parm::ncut](#)
- integer, dimension(:), allocatable [parm::irrno](#)  
*irrigation source location (none)*  
*if IRRSC=1, IRRNO is the number of the reach*  
*if IRRSC=2, IRRNO is the number of the reservoir*  
*if IRRSC=3, IRRNO is the number of the subbasin*  
*if IRRSC=4, IRRNO is the number of the subbasin*  
*if IRRSC=5, not used*
- integer, dimension(:), allocatable [parm::sol\\_nly](#)  
*number of soil in soil profile layers (none)*
- integer, dimension(:), allocatable [parm::irn](#)
- integer, dimension(:), allocatable [parm::npcp](#)
- integer, dimension(:), allocatable [parm::ncf](#)  
*sequence number of continuous fertilization operation within the year (none)*
- integer, dimension(:), allocatable [parm::ngr](#)  
*sequence number of grazing operation within the year (none)*
- integer, dimension(:), allocatable [parm::igrz](#)
- integer, dimension(:), allocatable [parm::ndeat](#)
- integer, dimension(:), allocatable [parm::hru\\_sub](#)  
*subbasin in which HRU is located (none)*
- integer, dimension(:), allocatable [parm::urblu](#)  
*urban land type identification number from urban.dat (none)*
- integer, dimension(:), allocatable [parm::ldrain](#)  
*soil layer where drainage tile is located (none)*
- integer, dimension(:), allocatable [parm::idorm](#)
- integer, dimension(:), allocatable [parm::hru\\_seq](#)
- integer, dimension(:), allocatable [parm::iurban](#)  
*urban simulation code (none):*  
*0 no urban sections in HRU*  
*1 urban sections in HRU, simulate using USGS regression equations*  
*2 urban sections in HRU, simulate using build up/wash off algorithm*

- integer, dimension(:), allocatable **parm::iday\_fert**
- integer, dimension(:), allocatable **parm::icfrt**
- integer, dimension(:), allocatable **parm::ifld**  
*number of HRU (in subbasin) that is a floodplain (none)*
- integer, dimension(:), allocatable **parm::irip**  
*number of HRU (in subbasin) that is a riparian zone (none)*
- integer, dimension(:), allocatable **parm::ndcfrt**
- integer, dimension(:), allocatable **parm::hrugis**
- integer, dimension(:), allocatable **parm::irrsc**  
*irrigation source code (none):*  
*1 divert water from reach*  
*2 divert water from reservoir*  
*3 divert water from shallow aquifer*  
*4 divert water from deep aquifer*  
*5 divert water from source outside watershed*
- integer, dimension(:), allocatable **parm::orig\_igro**
- integer, dimension(:), allocatable **parm::ntil**
- integer, dimension(:), allocatable **parm::iwatable**
- integer, dimension(:), allocatable **parm::curyr\_mat**
- integer, dimension(:), allocatable **parm::ncpest**
- integer, dimension(:), allocatable **parm::icpst**
- integer, dimension(:), allocatable **parm::ndcpst**
- integer, dimension(:), allocatable **parm::iday\_pest**
- integer, dimension(:), allocatable **parm::irr\_flag**
- integer, dimension(:), allocatable **parm::irra\_flag**
- integer, dimension(:,:), allocatable **parm::rndseed**  
*random number generator seed. The seeds in the array are used to generate random numbers for the following purposes (none):*  
*(1) wet/dry day probability*  
*(2) solar radiation*  
*(3) precipitation*  
*(4) USLE rainfall erosion index*  
*(5) wind speed*  
*(6) 0.5 hr rainfall fraction*  
*(7) relative humidity*  
*(8) maximum temperature*  
*(9) minimum temperature*  
*(10) generate new random numbers*
- integer, dimension(:,:), allocatable **parm::iterr**
- integer, dimension(:,:), allocatable **parm::iyterr**
- integer, dimension(:,:), allocatable **parm::itdrain**
- integer, dimension(:,:), allocatable **parm::iydrain**
- integer, dimension(:,:), allocatable **parm::ncrops**
- integer, dimension(:), allocatable **parm::manure\_id**  
*manure (fertilizer) identification number from fert.dat (none)*
- integer, dimension(:,:), allocatable **parm::mgt\_sdr**
- integer, dimension(:,:), allocatable **parm::idplot**
- integer, dimension(:,:), allocatable **parm::icont**
- integer, dimension(:,:), allocatable **parm::iycont**
- integer, dimension(:,:), allocatable **parm::ifilt**
- integer, dimension(:,:), allocatable **parm::iyfilt**
- integer, dimension(:,:), allocatable **parm::istrip**
- integer, dimension(:,:), allocatable **parm::iystrip**
- integer, dimension(:,:), allocatable **parm::iopday**
- integer, dimension(:,:), allocatable **parm::iopyr**
- integer, dimension(:,:), allocatable **parm::mgt\_ops**

- real \*8, dimension(:), allocatable **parm::wshd\_pstap**
- real \*8, dimension(:), allocatable **parm::wshd\_pstdg**
- integer, dimension(12) **parm::ndmo**
- integer, dimension(:), allocatable **parm::npno**  
*array of unique pesticides used in watershed (none)*
- integer, dimension(:), allocatable **parm::mcrhru**
- character(len=13), dimension(18) **parm::rfile**  
*rainfall file names (.pcp)*
- character(len=13), dimension(18) **parm::tfile**  
*temperature file names (.tmp)*
- character(len=4), dimension(1000) **parm::urbname**  
*name of urban land use*
- character(len=1), dimension(:), allocatable **parm::kirr**  
*irrigation in HRU*
- character(len=1), dimension(:), allocatable **parm::hydgrp**
- character(len=16), dimension(:), allocatable **parm::snam**  
*soil series name*
- character(len=17), dimension(300) **parm::pname**  
*name of pesticide/toxin*
- character(len=4), dimension(60) **parm::title**  
*description lines in file.cio (1st 3 lines)*
- character(len=4), dimension(5000) **parm::cpnm**  
*four character code to represent crop name*
- character(len=17), dimension(50) **parm::fname**
- real \*8, dimension(:, :, :), allocatable **parm::flomon**  
*average daily water loading for month ( $m^3/day$ )*
- real \*8, dimension(:, :, :), allocatable **parm::solpstmon**  
*average daily soluble pesticide loading for month (mg pst/day)*
- real \*8, dimension(:, :, :), allocatable **parm::srbspstmon**  
*average daily sorbed pesticide loading for month (mg pst/day)*
- real \*8, dimension(:, :, :), allocatable **parm::orgnmon**  
*average daily organic N loading for month (kg N/day)*
- real \*8, dimension(:, :, :), allocatable **parm::orgpmon**  
*average daily organic P loading for month (kg P/day)*
- real \*8, dimension(:, :, :), allocatable **parm::sedmon**  
*average daily sediment loading for month (metric tons/day)*
- real \*8, dimension(:, :, :), allocatable **parm::minpmon**  
*average daily mineral P loading for month (kg P/day)*
- real \*8, dimension(:, :, :), allocatable **parm::nh3mon**  
*average amount of NH3-N loaded to stream on a given day in the month (kg N/day)*
- real \*8, dimension(:, :, :), allocatable **parm::no3mon**  
*average daily NO3-N loading for month (kg N/day)*
- real \*8, dimension(:, :, :), allocatable **parm::bactlpmon**  
*average amount of less persistent bacteria loaded to stream on a given day in the month (# bact/day)*
- real \*8, dimension(:, :, :), allocatable **parm::bactpmon**  
*average amount of persistent bacteria loaded to stream on a given day in the month (# bact/day)*
- real \*8, dimension(:, :, :), allocatable **parm::no2mon**  
*average amount of NO2-N loaded to stream on a given day in the month (kg N/day)*
- real \*8, dimension(:, :, :), allocatable **parm::cmtl1mon**  
*average amount of conservative metal #1 loaded to stream on a given day in the month (# bact/day)*
- real \*8, dimension(:, :, :), allocatable **parm::cmtl2mon**

- average amount of conservative metal #2 loaded to stream on a given day in the month (# bact/day)*
- real \*8, dimension(:,:), allocatable [parm::cmtl3mon](#)
- average amount of conservative metal #3 loaded to stream on a given day in the month (# bact/day)*
- real \*8, dimension(:,:), allocatable [parm::cbodmon](#)
- average daily loading of CBOD in month (kg/day)*
- real \*8, dimension(:,:), allocatable [parm::chlamon](#)
- average daily loading of chlorophyll-a in month (kg/day)*
- real \*8, dimension(:,:), allocatable [parm::disoxmon](#)
- average daily loading of dissolved O2 in month (kg/day)*
- real \*8, dimension(:,:), allocatable [parm::floyr](#)
- average daily water loading for year ( $m^3/day$ )*
- real \*8, dimension(:,:), allocatable [parm::orgnyr](#)
- average daily organic N loading for year (kg N/day)*
- real \*8, dimension(:,:), allocatable [parm::orgpyr](#)
- average daily organic P loading for year (kg P/day)*
- real \*8, dimension(:,:), allocatable [parm::sedyr](#)
- average daily sediment loading for year (metric tons/day)*
- real \*8, dimension(:,:), allocatable [parm::minpyr](#)
- average daily mineral P loading for year (kg P/day)*
- real \*8, dimension(:,:), allocatable [parm::nh3yr](#)
- average daily NH3-N loading for year (kg N/day)*
- real \*8, dimension(:,:), allocatable [parm::no2yr](#)
- average daily NO2-N loading for year (kg N/day)*
- real \*8, dimension(:,:), allocatable [parm::no3yr](#)
- average daily NO3-N loading for year (kg N/day)*
- real \*8, dimension(:,:), allocatable [parm::bactlpyr](#)
- average daily loading of less persistent bacteria for year (# bact/day)*
- real \*8, dimension(:,:), allocatable [parm::bactpyr](#)
- average daily loading of persistent bacteria for year (# bact/day)*
- real \*8, dimension(:,:), allocatable [parm::cmtl1yr](#)
- average daily loading of conservative metal #1 for year (kg/day)*
- real \*8, dimension(:,:), allocatable [parm::chlayr](#)
- average daily loading of chlorophyll-a in year (kg/day)*
- real \*8, dimension(:,:), allocatable [parm::cmtl2yr](#)
- average daily loading of conservative metal #2 for year (kg/day)*
- real \*8, dimension(:,:), allocatable [parm::cmtl3yr](#)
- average daily loading of conservative metal #3 for year (kg/day)*
- real \*8, dimension(:,:), allocatable [parm::cbodyr](#)
- average daily loading of CBOD in year (kg/day)*
- real \*8, dimension(:,:), allocatable [parm::disoxyr](#)
- average daily loading of dissolved O2 in year (kg/day)*
- real \*8, dimension(:,:), allocatable [parm::solpstyr](#)
- average daily soluble pesticide loading for year (mg pst/day)*
- real \*8, dimension(:,:), allocatable [parm::srbpstyr](#)
- average daily sorbed pesticide loading for year (mg pst/day)*
- real \*8, dimension(:,:), allocatable [parm::sol\\_mc](#)
- real \*8, dimension(:,:), allocatable [parm::sol\\_mn](#)
- real \*8, dimension(:,:), allocatable [parm::sol\\_mp](#)
- real \*8, dimension(:), allocatable [parm::flocnst](#)
- real \*8, dimension(:), allocatable [parm::orgncnst](#)
- average daily organic N loading to reach (kg N/day)*



- real \*8, dimension(:), allocatable [parm::sedcnst](#)  
*average daily sediment loading for reach (metric tons/day)*
- real \*8, dimension(:), allocatable [parm::minpcnst](#)  
*average daily soluble P loading to reach (kg P/day)*
- real \*8, dimension(:), allocatable [parm::no3cnst](#)  
*average daily nitrate loading to reach (kg N/day)*
- real \*8, dimension(:), allocatable [parm::orgpcnst](#)  
*average daily organic P loading to reach (kg P/day)*
- real \*8, dimension(:), allocatable [parm::bactpcnst](#)  
*average daily persistent bacteria loading to reach (# bact/day)*
- real \*8, dimension(:), allocatable [parm::nh3cnst](#)  
*average daily ammonia loading to reach (kg N/day)*
- real \*8, dimension(:), allocatable [parm::no2cnst](#)  
*average daily nitrite loading to reach (kg N/day)*
- real \*8, dimension(:), allocatable [parm::bactlpcnst](#)  
*average daily less persistent bacteria loading to reach (# bact/day)*
- real \*8, dimension(:), allocatable [parm::cmtl1cnst](#)  
*average daily conservative metal #1 loading (kg/day)*
- real \*8, dimension(:), allocatable [parm::cmtl2cnst](#)  
*average daily conservative metal #2 loading (kg/day)*
- real \*8, dimension(:), allocatable [parm::chlacnst](#)  
*average daily loading of chlorophyll-a (kg/day)*
- real \*8, dimension(:), allocatable [parm::cmtl3cnst](#)  
*average daily conservative metal #3 loading (kg/day)*
- real \*8, dimension(:), allocatable [parm::disoxcnst](#)  
*average daily loading of dissolved O2 (kg/day)*
- real \*8, dimension(:), allocatable [parm::cbodcnst](#)  
*average daily loading of CBOD to reach (kg/day)*
- real \*8, dimension(:), allocatable [parm::solpstcnst](#)  
*average daily soluble pesticide loading (mg/day)*
- real \*8, dimension(:), allocatable [parm::srbpstcnst](#)  
*average daily sorbed pesticide loading (mg/day)*
- integer [parm::nstep](#)  
*max number of time steps per day or number of lines of rainfall data for each day*
- integer [parm::idt](#)  
*length of time step used to report precipitation data for sub-daily modeling (minutes)*
- real \*8, dimension(:), allocatable [parm::hrtwtr](#)
- real \*8, dimension(:), allocatable [parm::hhstor](#)
- real \*8, dimension(:), allocatable [parm::hdepth](#)
- real \*8, dimension(:), allocatable [parm::hsdti](#)
- real \*8, dimension(:), allocatable [parm::hrchwtr](#)
- real \*8, dimension(:), allocatable [parm::halgae](#)
- real \*8, dimension(:), allocatable [parm::horgn](#)
- real \*8, dimension(:), allocatable [parm::hnh4](#)
- real \*8, dimension(:), allocatable [parm::hno2](#)
- real \*8, dimension(:), allocatable [parm::hno3](#)
- real \*8, dimension(:), allocatable [parm::horgp](#)
- real \*8, dimension(:), allocatable [parm::hsolp](#)
- real \*8, dimension(:), allocatable [parm::hbod](#)
- real \*8, dimension(:), allocatable [parm::hdisox](#)
- real \*8, dimension(:), allocatable [parm::hchla](#)
- real \*8, dimension(:), allocatable [parm::hsedyld](#)



- real \*8, dimension(:), allocatable **parm::hsedst**
- real \*8, dimension(:), allocatable **parm::hharea**
- real \*8, dimension(:), allocatable **parm::hsolpst**
- real \*8, dimension(:), allocatable **parm::hsorpst**
- real \*8, dimension(:), allocatable **parm::hhqday**
- real \*8, dimension(:), allocatable **parm::precipdt**
- real \*8, dimension(:), allocatable **parm::hhtime**
- real \*8, dimension(:), allocatable **parm::hbactp**
- real \*8, dimension(:), allocatable **parm::hbactlp**
- integer, dimension(10) **parm::ivar\_orig**
- real \*8, dimension(10) **parm::rvar\_orig**
- integer **parm::nsave**  
*number of save commands in .fig file*
- integer **parm::nauto**
- integer **parm::iatmodep**
- real \*8, dimension(:), allocatable **parm::wattemp**
- real \*8, dimension(:), allocatable **parm::lkpst\_mass**
- real \*8, dimension(:), allocatable **parm::lkspst\_mass**
- real \*8, dimension(:), allocatable **parm::vel\_chan**
- real \*8, dimension(:), allocatable **parm::vfscon**  
*fraction of the total runoff from the entire field entering the most concentrated 10% of the VFS (none)*
- real \*8, dimension(:), allocatable **parm::vfsratio**  
*field area/VFS area ratio (none)*
- real \*8, dimension(:), allocatable **parm::vfsch**  
*fraction of flow entering the most concentrated 10% of the VFS which is fully channelized (none)*
- real \*8, dimension(:), allocatable **parm::vfsi**
- real \*8, dimension(:,:), allocatable **parm::filter\_i**
- real \*8, dimension(:,:), allocatable **parm::filter\_ratio**
- real \*8, dimension(:,:), allocatable **parm::filter\_con**
- real \*8, dimension(:,:), allocatable **parm::filter\_ch**
- real \*8, dimension(:,:), allocatable **parm::sol\_n**
- integer **parm::cswat**  
*= 0 Static soil carbon (old mineralization routines)*  
*= 1 C-FARM one carbon pool model*  
*= 2 Century model*
- real \*8, dimension(:,:), allocatable **parm::sol\_bdp**
- real \*8, dimension(:,:), allocatable **parm::tillagef**
- real \*8, dimension(:), allocatable **parm::rtfr**
- real \*8, dimension(:), allocatable **parm::stsol\_rd**
- integer **parm::urban\_flag**
- integer **parm::dorm\_flag**
- real \*8 **parm::bf\_flg**
- real \*8 **parm::iabstr**
- real \*8, dimension(:), allocatable **parm::ubnrunoff**
- real \*8, dimension(:), allocatable **parm::ubntss**
- real \*8, dimension(:,:), allocatable **parm::sub\_ubnrunoff**
- real \*8, dimension(:,:), allocatable **parm::sub\_ubntss**
- real \*8, dimension(:,:), allocatable **parm::ovrlnd\_dt**
- real \*8, dimension(:,:), allocatable **parm::hhsurf\_bs**
- integer **parm::iuh**  
*unit hydrograph method: 1=triangular UH; 2=gamma function UH;*
- integer **parm::sed\_ch**  
*channel routing for HOURLY; 0=Bagnold; 2=Brownlie; 3=Yang;*
- real \*8 **parm::eros\_expo**

- an exponent in the overland flow erosion equation ranges 1.5-3.0*
- real \*8 **parm::eros\_spl**
  - coefficient of splash erosion varing 0.9-3.1*
- real \*8 **parm::rill\_mult**
  - Multiplier to USLE\_K for soil susceptible to rill erosion, range 0.5-2.0.*
- real \*8 **parm::sedprev**
- real \*8 **parm::c\_factor**
- real \*8 **parm::ch\_d50**
  - median particle diameter of channel bed (mm)*
- real \*8 **parm::sig\_g**
  - geometric standard deviation of particle sizes for the main channel. Mean air temperature at which precipitation is equally likely to be rain as snow/freezing rain.*
- real \*8 **parm::uhalpha**
  - alpha coefficient for estimating unit hydrograph using a gamma function (\*.bsn)*
- real \*8 **parm::abstinit**
- real \*8 **parm::abstmax**
- real \*8, dimension(:,:), allocatable **parm::hhsedy**
- real \*8, dimension(:,:), allocatable **parm::sub\_subp\_dt**
- real \*8, dimension(:,:), allocatable **parm::sub\_hhsedy**
- real \*8, dimension(:,:), allocatable **parm::sub\_atmp**
- real \*8, dimension(:), allocatable **parm::rhy**
- real \*8, dimension(:), allocatable **parm::init\_abstrc**
- real \*8, dimension(:), allocatable **parm::dratio**
- real \*8, dimension(:), allocatable **parm::hrtevp**
- real \*8, dimension(:), allocatable **parm::hrttlc**
- real \*8, dimension(:,:), allocatable **parm::rchhr**
- real \*8, dimension(:), allocatable **parm::hhresflwi**
- real \*8, dimension(:), allocatable **parm::hhresflwo**
- real \*8, dimension(:), allocatable **parm::hhressedi**
- real \*8, dimension(:), allocatable **parm::hhressedo**
- character(len=4), dimension(:), allocatable **parm::lu\_nodrain**
- integer, dimension(:), allocatable **parm::bmpdrain**
- real \*8, dimension(:), allocatable **parm::sub\_cn2**
- real \*8, dimension(:), allocatable **parm::sub\_ha\_urb**
- real \*8, dimension(:), allocatable **parm::bmp\_recharge**
- real \*8, dimension(:), allocatable **parm::sub\_ha\_imp**
- real \*8, dimension(:), allocatable **parm::subdr\_km**
- real \*8, dimension(:), allocatable **parm::subdr\_ickm**
- real \*8, dimension(:,:), allocatable **parm::sf\_im**
- real \*8, dimension(:,:), allocatable **parm::sf\_iy**
- real \*8, dimension(:,:), allocatable **parm::sp\_sa**
- real \*8, dimension(:,:), allocatable **parm::sp\_pvol**
- real \*8, dimension(:,:), allocatable **parm::sp\_pd**
- real \*8, dimension(:,:), allocatable **parm::sp\_sedi**
- real \*8, dimension(:,:), allocatable **parm::sp\_sede**
- real \*8, dimension(:,:), allocatable **parm::ft\_sa**
- real \*8, dimension(:,:), allocatable **parm::ft\_fsa**
- real \*8, dimension(:,:), allocatable **parm::ft\_dep**
- real \*8, dimension(:,:), allocatable **parm::ft\_h**
- real \*8, dimension(:,:), allocatable **parm::ft\_pd**
- real \*8, dimension(:,:), allocatable **parm::ft\_k**
- real \*8, dimension(:,:), allocatable **parm::ft\_dp**
- real \*8, dimension(:,:), allocatable **parm::ft\_dc**
- real \*8, dimension(:,:), allocatable **parm::ft\_por**

- real \*8, dimension(:,:), allocatable **parm::tss\_den**
- real \*8, dimension(:,:), allocatable **parm::ft\_alp**
- real \*8, dimension(:,:), allocatable **parm::sf\_fr**
- real \*8, dimension(:,:), allocatable **parm::sp\_qi**
- real \*8, dimension(:,:), allocatable **parm::sp\_k**
- real \*8, dimension(:,:), allocatable **parm::ft\_qpnd**
- real \*8, dimension(:,:), allocatable **parm::sp\_dp**
- real \*8, dimension(:,:), allocatable **parm::ft\_qsw**
- real \*8, dimension(:,:), allocatable **parm::ft\_qin**
- real \*8, dimension(:,:), allocatable **parm::ft\_qout**
- real \*8, dimension(:,:), allocatable **parm::ft\_sedpnd**
- real \*8, dimension(:,:), allocatable **parm::sp\_bpw**
- real \*8, dimension(:,:), allocatable **parm::ft\_bpw**
- real \*8, dimension(:,:), allocatable **parm::ft\_sed\_cumul**
- real \*8, dimension(:,:), allocatable **parm::sp\_sed\_cumul**
- integer, dimension(:), allocatable **parm::num\_sf**
- integer, dimension(:,:), allocatable **parm::sf\_typ**
- integer, dimension(:,:), allocatable **parm::sf\_dim**
- integer, dimension(:,:), allocatable **parm::ft\_qfg**
- integer, dimension(:,:), allocatable **parm::sp\_qfg**
- integer, dimension(:,:), allocatable **parm::sf\_ptp**
- integer, dimension(:,:), allocatable **parm::ft\_fc**
- real \*8 **parm::sfsedmean**
- real \*8 **parm::sfsedstdev**
- integer, dimension(:), allocatable **parm::dtp\_imo**  
*month the reservoir becomes operational (none)*
- integer, dimension(:), allocatable **parm::dtp\_iyr**  
*year of the simulation that the reservoir becomes operational (none)*
- integer, dimension(:), allocatable **parm::dtp\_numstage**  
*total number of stages in the weir (none)*
- integer, dimension(:), allocatable **parm::dtp\_numweir**  
*total number of weirs in the BMP (none)*
- integer, dimension(:), allocatable **parm::dtp\_onoff**  
*sub-basin detention pond is associated with (none)*
- integer, dimension(:), allocatable **parm::dtp\_relytype**  
*equations for stage-discharge relationship (none):*  
*1=exponential function,*  
*2=linear,*  
*3=logarithmic,*  
*4=cubic,*  
*5=power*
- integer, dimension(:), allocatable **parm::dtp\_stagdis**  
*(none):*  
*0=use weir/orifice discharge equation to calculate outflow,*  
*1=use stage-dicharge relationship*
- integer, dimension(:), allocatable **parm::dtp\_subnum**
- real \*8, dimension(:), allocatable **parm::cf**  
*this parameter controls the response of decomposition to the combined effect of soil temperature and moisture.*
- real \*8, dimension(:), allocatable **parm::cfh**  
*maximum humification rate*
- real \*8, dimension(:), allocatable **parm::cfdec**  
*the undisturbed soil turnover rate under optimum soil water and temperature. Increasing it will increase carbon and organic N decomp.*
- real \*8, dimension(:), allocatable **parm::lat\_orgn**

- real \*8, dimension(:), allocatable **parm::lat\_orgp**
- integer, dimension(:,:), allocatable **parm::dtp\_weirdim**  
*weir dimensions (none),*  
*1=read user input,*  
*0=use model calculation*
- integer, dimension(:,:), allocatable **parm::dtp\_weirtype**  
*type of weir (none):*  
*1=rectangular and*  
*2=circular*
- real \*8, dimension(:), allocatable **parm::dtp\_coef1**  
*coefficient of 3rd degree in the polynomial equation (none)*
- real \*8, dimension(:), allocatable **parm::dtp\_coef2**  
*coefficient of 2nd degree in the polynomial equation (none)*
- real \*8, dimension(:), allocatable **parm::dtp\_coef3**  
*coefficient of 1st degree in the polynomial equation (none)*
- real \*8, dimension(:), allocatable **parm::dtp\_evrsv**  
*detention pond evaporation coefficient (none)*
- real \*8, dimension(:), allocatable **parm::dtp\_expont**  
*exponent used in the exponential equation (none)*
- real \*8, dimension(:), allocatable **parm::dtp\_intcept**  
*intercept used in regression equations (none)*
- real \*8, dimension(:), allocatable **parm::dtp\_lwratio**  
*ratio of length to width of water back up (none)*
- real \*8, dimension(:), allocatable **parm::dtp\_totwrwid**  
*total constructed width of the detention wall across the creek (m)*
- real \*8, dimension(:), allocatable **parm::dtp\_inflvol**
- real \*8, dimension(:), allocatable **parm::dtp\_wdep**
- real \*8, dimension(:), allocatable **parm::dtp\_totdep**
- real \*8, dimension(:), allocatable **parm::dtp\_watdepact**
- real \*8, dimension(:), allocatable **parm::dtp\_outflow**
- real \*8, dimension(:), allocatable **parm::dtp\_totrel**
- real \*8, dimension(:), allocatable **parm::dtp\_backoff**
- real \*8, dimension(:), allocatable **parm::dtp\_seep\_sa**
- real \*8, dimension(:), allocatable **parm::dtp\_evap\_sa**
- real \*8, dimension(:), allocatable **parm::dtp\_pet\_day**
- real \*8, dimension(:), allocatable **parm::dtp\_pcpvol**
- real \*8, dimension(:), allocatable **parm::dtp\_seepvol**
- real \*8, dimension(:), allocatable **parm::dtp\_evapvol**
- real \*8, dimension(:), allocatable **parm::dtp\_flowin**
- real \*8, dimension(:), allocatable **parm::dtp\_backup\_length**
- real \*8, dimension(:), allocatable **parm::dtp\_ivol**
- real \*8, dimension(:), allocatable **parm::dtp\_ised**
- integer, dimension(:,:), allocatable **parm::so\_res\_flag**
- integer, dimension(:,:), allocatable **parm::ro\_bmp\_flag**
- real \*8, dimension(:,:), allocatable **parm::sol\_watp**
- real \*8, dimension(:,:), allocatable **parm::sol\_solp\_pre**
- real \*8, dimension(:,:), allocatable **parm::psp\_store**
- real \*8, dimension(:,:), allocatable **parm::ssp\_store**
- real \*8, dimension(:,:), allocatable **parm::so\_res**
- real \*8, dimension(:,:), allocatable **parm::sol\_cal**
- real \*8, dimension(:,:), allocatable **parm::sol\_ph**
- integer **parm::sol\_p\_model**
- integer, dimension(:,:), allocatable **parm::a\_days**
- integer, dimension(:,:), allocatable **parm::b\_days**

- real \*8, dimension(:), allocatable **parm::harv\_min**
- real \*8, dimension(:), allocatable **parm::fstap**
- real \*8, dimension(:), allocatable **parm::min\_res**
- real \*8, dimension(:,), allocatable **parm::ro\_bmp\_flo**
- real \*8, dimension(:,), allocatable **parm::ro\_bmp\_sed**
- real \*8, dimension(:,), allocatable **parm::ro\_bmp\_bac**
- real \*8, dimension(:,), allocatable **parm::ro\_bmp\_pp**
- real \*8, dimension(:,), allocatable **parm::ro\_bmp\_sp**
- real \*8, dimension(:,), allocatable **parm::ro\_bmp\_pn**
- real \*8, dimension(:,), allocatable **parm::ro\_bmp\_sn**
- real \*8, dimension(:,), allocatable **parm::ro\_bmp\_flos**
- real \*8, dimension(:,), allocatable **parm::ro\_bmp\_seds**
- real \*8, dimension(:,), allocatable **parm::ro\_bmp\_bacs**
- real \*8, dimension(:,), allocatable **parm::ro\_bmp\_pps**
- real \*8, dimension(:,), allocatable **parm::ro\_bmp\_sps**
- real \*8, dimension(:,), allocatable **parm::ro\_bmp\_pns**
- real \*8, dimension(:,), allocatable **parm::ro\_bmp\_sns**
- real \*8, dimension(:,), allocatable **parm::ro\_bmp\_flot**
- real \*8, dimension(:,), allocatable **parm::ro\_bmp\_sedt**
- real \*8, dimension(:,), allocatable **parm::ro\_bmp\_bact**
- real \*8, dimension(:,), allocatable **parm::ro\_bmp\_ppt**
- real \*8, dimension(:,), allocatable **parm::ro\_bmp\_spt**
- real \*8, dimension(:,), allocatable **parm::ro\_bmp\_pnt**
- real \*8, dimension(:,), allocatable **parm::ro\_bmp\_snt**
- real \*8, dimension(:), allocatable **parm::bmp\_flo**
- real \*8, dimension(:), allocatable **parm::bmp\_sed**
- real \*8, dimension(:), allocatable **parm::bmp\_bac**
- real \*8, dimension(:), allocatable **parm::bmp\_pp**
- real \*8, dimension(:), allocatable **parm::bmp\_sp**
- real \*8, dimension(:), allocatable **parm::bmp\_pn**
- real \*8, dimension(:), allocatable **parm::bmp\_sn**
- real \*8, dimension(:), allocatable **parm::bmp\_flag**
- real \*8, dimension(:), allocatable **parm::bmp\_flos**
- real \*8, dimension(:), allocatable **parm::bmp\_seds**
- real \*8, dimension(:), allocatable **parm::bmp\_bacs**
- real \*8, dimension(:), allocatable **parm::bmp\_pps**
- real \*8, dimension(:), allocatable **parm::bmp\_sps**
- real \*8, dimension(:), allocatable **parm::bmp\_pns**
- real \*8, dimension(:), allocatable **parm::bmp\_sns**
- real \*8, dimension(:), allocatable **parm::bmp\_flot**
- real \*8, dimension(:), allocatable **parm::bmp\_sedt**
- real \*8, dimension(:), allocatable **parm::bmp\_bact**
- real \*8, dimension(:), allocatable **parm::bmp\_ppt**
- real \*8, dimension(:), allocatable **parm::bmp\_spt**
- real \*8, dimension(:), allocatable **parm::bmp\_pnt**
- real \*8, dimension(:), allocatable **parm::bmp\_snt**
- real \*8, dimension(:,), allocatable **parm::dtp\_addon**  
*the distance between spillway levels (m)*
- real \*8, dimension(:,), allocatable **parm::dtp\_cdis**  
*discharge coefficient for weir/orifice flow (none)*
- real \*8, dimension(:,), allocatable **parm::dtp\_depweir**  
*depth of rectangular weir at different stages (m)*
- real \*8, dimension(:,), allocatable **parm::dtp\_diaweir**  
*diameter of orifice hole at different stages (m)*

- real \*8, dimension(:,:), allocatable **parm::dtp\_flowrate**  
*maximum discharge from each stage of the weir/hole ( $m^3/s$ )*
- real \*8, dimension(:,:), allocatable **parm::dtp\_pcpret**  
*precipitation for different return periods (not used) (mm)*
- real \*8, dimension(:,:), allocatable **parm::dtp\_retper**  
*return period at different stages (years)*
- real \*8, dimension(:,:), allocatable **parm::dtp\_wdratio**  
*width depth ratio of rectangular weirs (none)*
- real \*8, dimension(:,:), allocatable **parm::dtp\_wrwid**
- real \*8, dimension(:), allocatable **parm::ri\_subkm**
- real \*8, dimension(:), allocatable **parm::ri\_totpvol**
- real \*8, dimension(:), allocatable **parm::irmmdt**
- real \*8, dimension(:,:), allocatable **parm::ri\_sed**
- real \*8, dimension(:,:), allocatable **parm::ri\_fr**
- real \*8, dimension(:,:), allocatable **parm::ri\_dim**
- real \*8, dimension(:,:), allocatable **parm::ri\_im**
- real \*8, dimension(:,:), allocatable **parm::ri\_iy**
- real \*8, dimension(:,:), allocatable **parm::ri\_sa**
- real \*8, dimension(:,:), allocatable **parm::ri\_vol**
- real \*8, dimension(:,:), allocatable **parm::ri\_qi**
- real \*8, dimension(:,:), allocatable **parm::ri\_k**
- real \*8, dimension(:,:), allocatable **parm::ri\_dd**
- real \*8, dimension(:,:), allocatable **parm::ri\_evrsv**
- real \*8, dimension(:,:), allocatable **parm::ri\_dep**
- real \*8, dimension(:,:), allocatable **parm::ri\_ndt**
- real \*8, dimension(:,:), allocatable **parm::ri\_pmpvol**
- real \*8, dimension(:,:), allocatable **parm::ri\_sed\_cumul**
- real \*8, dimension(:,:), allocatable **parm::hrnopcp**
- real \*8, dimension(:,:), allocatable **parm::ri\_qloss**
- real \*8, dimension(:,:), allocatable **parm::ri\_pumpv**
- real \*8, dimension(:,:), allocatable **parm::ri\_sedi**
- character(len=4), dimension(:,:), allocatable **parm::ri\_nirr**
- integer, dimension(:), allocatable **parm::num\_ri**
- integer, dimension(:), allocatable **parm::ri\_luflg**
- integer, dimension(:), allocatable **parm::num\_noirr**
- integer, dimension(:), allocatable **parm::wtp\_subnum**
- integer, dimension(:), allocatable **parm::wtp\_onoff**
- integer, dimension(:), allocatable **parm::wtp\_imo**
- integer, dimension(:), allocatable **parm::wtp\_iyr**
- integer, dimension(:), allocatable **parm::wtp\_dim**
- integer, dimension(:), allocatable **parm::wtp\_stagdis**
- integer, dimension(:), allocatable **parm::wtp\_sdtype**
- real \*8, dimension(:), allocatable **parm::wtp\_pvol**
- real \*8, dimension(:), allocatable **parm::wtp\_pdepth**
- real \*8, dimension(:), allocatable **parm::wtp\_sdslope**
- real \*8, dimension(:), allocatable **parm::wtp\_lenwidth**
- real \*8, dimension(:), allocatable **parm::wtp\_extdepth**
- real \*8, dimension(:), allocatable **parm::wtp\_hydeff**
- real \*8, dimension(:), allocatable **parm::wtp\_evrsv**
- real \*8, dimension(:), allocatable **parm::wtp\_sdintc**
- real \*8, dimension(:), allocatable **parm::wtp\_sdexp**
- real \*8, dimension(:), allocatable **parm::wtp\_sdc1**
- real \*8, dimension(:), allocatable **parm::wtp\_sdc2**
- real \*8, dimension(:), allocatable **parm::wtp\_sdc3**

- real \*8, dimension(:), allocatable **parm::wtp\_pdia**
- real \*8, dimension(:), allocatable **parm::wtp\_plen**
- real \*8, dimension(:), allocatable **parm::wtp\_pmann**
- real \*8, dimension(:), allocatable **parm::wtp\_ploss**
- real \*8, dimension(:), allocatable **parm::wtp\_k**
- real \*8, dimension(:), allocatable **parm::wtp\_dp**
- real \*8, dimension(:), allocatable **parm::wtp\_sedi**
- real \*8, dimension(:), allocatable **parm::wtp\_sede**
- real \*8, dimension(:), allocatable **parm::wtp\_qi**
- real \*8 **parm::lai\_init**  
*initial leaf area index of transplants*
- real \*8 **parm::bio\_init**  
*initial biomass of transplants (kg/ha)*
- real \*8 **parm::cnop**  
*SCS runoff curve number for moisture condition II (none)*
- real \*8 **parm::harveff**  
*harvest efficiency: fraction of harvested yield that is removed from HRU; the remainder becomes residue on the soil surface(none)*
- real \*8 **parm::hi\_ovr**  
*harvest index target specified at harvest ((kg/ha)/(kg/ha))*
- real \*8 **parm::frac\_harvk**
- real \*8 **parm::lid\_vgcl**
- real \*8 **parm::lid\_vgcm**
- real \*8 **parm::lid\_qsurf\_total**
- real \*8 **parm::lid\_farea\_sum**
- real \*8, dimension(:,:), allocatable **parm::lid\_cuminf\_last**
- real \*8, dimension(:,:), allocatable **parm::lid\_sw\_last**
- real \*8, dimension(:,:), allocatable **parm::interval\_last**
- real \*8, dimension(:,:), allocatable **parm::lid\_f\_last**
- real \*8, dimension(:,:), allocatable **parm::lid\_cumr\_last**
- real \*8, dimension(:,:), allocatable **parm::lid\_str\_last**
- real \*8, dimension(:,:), allocatable **parm::lid\_farea**
- real \*8, dimension(:,:), allocatable **parm::lid\_qsurf**
- real \*8, dimension(:,:), allocatable **parm::lid\_sw\_add**
- real \*8, dimension(:,:), allocatable **parm::lid\_cumqperc\_last**
- real \*8, dimension(:,:), allocatable **parm::lid\_cumirr\_last**
- real \*8, dimension(:,:), allocatable **parm::lid\_excum\_last**
- integer, dimension(:,:), allocatable **parm::gr\_onoff**
- integer, dimension(:,:), allocatable **parm::gr\_imo**
- integer, dimension(:,:), allocatable **parm::gr\_iyr**
- real \*8, dimension(:,:), allocatable **parm::gr\_farea**
- real \*8, dimension(:,:), allocatable **parm::gr\_solop**
- real \*8, dimension(:,:), allocatable **parm::gr\_etcoef**
- real \*8, dimension(:,:), allocatable **parm::gr\_fc**
- real \*8, dimension(:,:), allocatable **parm::gr\_wp**
- real \*8, dimension(:,:), allocatable **parm::gr\_ksat**
- real \*8, dimension(:,:), allocatable **parm::gr\_por**
- real \*8, dimension(:,:), allocatable **parm::gr\_hydeff**
- real \*8, dimension(:,:), allocatable **parm::gr\_soldpt**
- integer, dimension(:,:), allocatable **parm::rg\_onoff**
- integer, dimension(:,:), allocatable **parm::rg\_imo**
- integer, dimension(:,:), allocatable **parm::rg\_iyr**
- real \*8, dimension(:,:), allocatable **parm::rg\_farea**
- real \*8, dimension(:,:), allocatable **parm::rg\_solop**

- real \*8, dimension(:,:), allocatable **parm::rg\_etcoef**
- real \*8, dimension(:,:), allocatable **parm::rg\_fc**
- real \*8, dimension(:,:), allocatable **parm::rg\_wp**
- real \*8, dimension(:,:), allocatable **parm::rg\_ksat**
- real \*8, dimension(:,:), allocatable **parm::rg\_por**
- real \*8, dimension(:,:), allocatable **parm::rg\_hydeff**
- real \*8, dimension(:,:), allocatable **parm::rg\_soldpt**
- real \*8, dimension(:,:), allocatable **parm::rg\_dimop**
- real \*8, dimension(:,:), allocatable **parm::rg\_sarea**
- real \*8, dimension(:,:), allocatable **parm::rg\_vol**
- real \*8, dimension(:,:), allocatable **parm::rg\_sth**
- real \*8, dimension(:,:), allocatable **parm::rg\_sdia**
- real \*8, dimension(:,:), allocatable **parm::rg\_bdia**
- real \*8, dimension(:,:), allocatable **parm::rg\_sts**
- real \*8, dimension(:,:), allocatable **parm::rg\_orifice**
- real \*8, dimension(:,:), allocatable **parm::rg\_oheight**
- real \*8, dimension(:,:), allocatable **parm::rg\_odia**
- integer, dimension(:,:), allocatable **parm::cs\_onoff**
- integer, dimension(:,:), allocatable **parm::cs\_imo**
- integer, dimension(:,:), allocatable **parm::cs\_iyr**
- integer, dimension(:,:), allocatable **parm::cs\_grcon**
- real \*8, dimension(:,:), allocatable **parm::cs\_farea**
- real \*8, dimension(:,:), allocatable **parm::cs\_vol**
- real \*8, dimension(:,:), allocatable **parm::cs\_rdepth**
- integer, dimension(:,:), allocatable **parm::pv\_onoff**
- integer, dimension(:,:), allocatable **parm::pv\_imo**
- integer, dimension(:,:), allocatable **parm::pv\_iyr**
- integer, dimension(:,:), allocatable **parm::pv\_solop**
- real \*8, dimension(:,:), allocatable **parm::pv\_grvdep**
- real \*8, dimension(:,:), allocatable **parm::pv\_grvpor**
- real \*8, dimension(:,:), allocatable **parm::pv\_farea**
- real \*8, dimension(:,:), allocatable **parm::pv\_drcoef**
- real \*8, dimension(:,:), allocatable **parm::pv\_fc**
- real \*8, dimension(:,:), allocatable **parm::pv\_wp**
- real \*8, dimension(:,:), allocatable **parm::pv\_ksat**
- real \*8, dimension(:,:), allocatable **parm::pv\_por**
- real \*8, dimension(:,:), allocatable **parm::pv\_hydeff**
- real \*8, dimension(:,:), allocatable **parm::pv\_soldpt**
- integer, dimension(:,:), allocatable **parm::lid\_onoff**
- real \*8, dimension(:,:), allocatable **parm::sol\_bmc**
- real \*8, dimension(:,:), allocatable **parm::sol\_bmn**
- real \*8, dimension(:,:), allocatable **parm::sol\_hsc**
- real \*8, dimension(:,:), allocatable **parm::sol\_hsn**
- real \*8, dimension(:,:), allocatable **parm::sol\_hpc**
- real \*8, dimension(:,:), allocatable **parm::sol\_hpn**
- real \*8, dimension(:,:), allocatable **parm::sol\_lm**
- real \*8, dimension(:,:), allocatable **parm::sol\_lmc**
- real \*8, dimension(:,:), allocatable **parm::sol\_lmn**
- real \*8, dimension(:,:), allocatable **parm::sol\_ls**
- real \*8, dimension(:,:), allocatable **parm::sol\_lsl**
- real \*8, dimension(:,:), allocatable **parm::sol\_lsc**
- real \*8, dimension(:,:), allocatable **parm::sol\_lsn**
- real \*8, dimension(:,:), allocatable **parm::sol\_rnmn**
- real \*8, dimension(:,:), allocatable **parm::sol\_lslc**
- real \*8, dimension(:,:), allocatable **parm::sol\_lslnc**



- real \*8, dimension(:,:), allocatable **parm::sol\_rspc**
- real \*8, dimension(:,:), allocatable **parm::sol\_woc**
- real \*8, dimension(:,:), allocatable **parm::sol\_won**
- real \*8, dimension(:,:), allocatable **parm::sol\_hp**
- real \*8, dimension(:,:), allocatable **parm::sol\_hs**
- real \*8, dimension(:,:), allocatable **parm::sol\_bm**
- real \*8, dimension(:,:), allocatable **parm::sol\_cac**
- real \*8, dimension(:,:), allocatable **parm::sol\_cec**
- real \*8, dimension(:,:), allocatable **parm::sol\_percc**
- real \*8, dimension(:,:), allocatable **parm::sol\_latc**
- real \*8, dimension(:), allocatable **parm::sedc\_d**
- real \*8, dimension(:), allocatable **parm::surfqc\_d**
- real \*8, dimension(:), allocatable **parm::latc\_d**
- real \*8, dimension(:), allocatable **parm::percc\_d**
- real \*8, dimension(:), allocatable **parm::foc\_d**
- real \*8, dimension(:), allocatable **parm::nppc\_d**
- real \*8, dimension(:), allocatable **parm::rsdc\_d**
- real \*8, dimension(:), allocatable **parm::grainc\_d**
- real \*8, dimension(:), allocatable **parm::stoverc\_d**
- real \*8, dimension(:), allocatable **parm::soc\_d**
- real \*8, dimension(:), allocatable **parm::rspc\_d**
- real \*8, dimension(:), allocatable **parm::emitc\_d**
- real \*8, dimension(:), allocatable **parm::sub\_sedc\_d**
- real \*8, dimension(:), allocatable **parm::sub\_surfqc\_d**
- real \*8, dimension(:), allocatable **parm::sub\_latc\_d**
- real \*8, dimension(:), allocatable **parm::sub\_percc\_d**
- real \*8, dimension(:), allocatable **parm::sub\_foc\_d**
- real \*8, dimension(:), allocatable **parm::sub\_nppc\_d**
- real \*8, dimension(:), allocatable **parm::sub\_rsdc\_d**
- real \*8, dimension(:), allocatable **parm::sub\_grainc\_d**
- real \*8, dimension(:), allocatable **parm::sub\_stoverc\_d**
- real \*8, dimension(:), allocatable **parm::sub\_emitc\_d**
- real \*8, dimension(:), allocatable **parm::sub\_soc\_d**
- real \*8, dimension(:), allocatable **parm::sub\_rspc\_d**
- real \*8, dimension(:), allocatable **parm::sedc\_m**
- real \*8, dimension(:), allocatable **parm::surfqc\_m**
- real \*8, dimension(:), allocatable **parm::latc\_m**
- real \*8, dimension(:), allocatable **parm::percc\_m**
- real \*8, dimension(:), allocatable **parm::foc\_m**
- real \*8, dimension(:), allocatable **parm::nppc\_m**
- real \*8, dimension(:), allocatable **parm::rsdc\_m**
- real \*8, dimension(:), allocatable **parm::grainc\_m**
- real \*8, dimension(:), allocatable **parm::stoverc\_m**
- real \*8, dimension(:), allocatable **parm::emitc\_m**
- real \*8, dimension(:), allocatable **parm::soc\_m**
- real \*8, dimension(:), allocatable **parm::rspc\_m**
- real \*8, dimension(:), allocatable **parm::sedc\_a**
- real \*8, dimension(:), allocatable **parm::surfqc\_a**
- real \*8, dimension(:), allocatable **parm::latc\_a**
- real \*8, dimension(:), allocatable **parm::percc\_a**
- real \*8, dimension(:), allocatable **parm::foc\_a**
- real \*8, dimension(:), allocatable **parm::nppc\_a**
- real \*8, dimension(:), allocatable **parm::rsdc\_a**
- real \*8, dimension(:), allocatable **parm::grainc\_a**
- real \*8, dimension(:), allocatable **parm::stoverc\_a**

- real \*8, dimension(:), allocatable **parm::emitc\_a**
- real \*8, dimension(:), allocatable **parm::soc\_a**
- real \*8, dimension(:), allocatable **parm::rspc\_a**
- integer, dimension(:), allocatable **parm::tillage\_switch**
- real \*8, dimension(:), allocatable **parm::tillage\_depth**
- integer, dimension(:), allocatable **parm::tillage\_days**
- real \*8, dimension(:), allocatable **parm::tillage\_factor**
- real \*8 **parm::dthy**  
*time interval for subdaily routing*
- integer, dimension(4) **parm::ihx**
- integer, dimension(:), allocatable **parm::nhy**
- real \*8, dimension(:), allocatable **parm::rchx**
- real \*8, dimension(:), allocatable **parm::rcss**
- real \*8, dimension(:), allocatable **parm::qcap**
- real \*8, dimension(:), allocatable **parm::chxa**
- real \*8, dimension(:), allocatable **parm::chxp**
- real \*8, dimension(:, :, :), allocatable **parm::qhy**
- real \*8 **parm::ff1**
- real \*8 **parm::ff2**

### 7.16.1 Detailed Description

file containing the module parm

Author

modified by Javier Burguete Tolosa

## 7.17 openwth.f90 File Reference

### Functions/Subroutines

- subroutine **openwth**  
*this subroutine opens the precipitation, temperature, solar radiation, relative humidity and wind speed files for simulations using measured weather data*

### 7.17.1 Detailed Description

file containing the subroutine openwth

Author

modified by Javier Burguete

## 7.18 qman.f90 File Reference

### Functions/Subroutines

- real \*8 function **qman** (x1, x2, x3, x4)  
*this subroutine calculates flow rate or flow velocity using Manning's equation. If x1 is set to 1, the velocity is calculated. If x1 is set to cross-sectional area of flow, the flow rate is calculated.*

### 7.18.1 Detailed Description

file containing the function qman

Author

modified by Javier Burguete

### 7.18.2 Function/Subroutine Documentation

#### 7.18.2.1 qman()

```
real*8 function qman (
    real*8, intent(in) x1,
    real*8, intent(in) x2,
    real*8, intent(in) x3,
    real*8, intent(in) x4 )
```

this subroutine calculates flow rate or flow velocity using Manning's equation. If x1 is set to 1, the velocity is calculated. If x1 is set to cross-sectional area of flow, the flow rate is calculated.

Parameters

|    |    |   |
|----|----|---|
| in | x1 | cross-sectional flow area or 1 (m <sup>2</sup> or none) |
| in | x2 | hydraulic radius (m)                                    |
| in | x3 | Manning's "n" value for channel (none)                  |
| in | x4 | average slope of channel (m/m)                          |

Returns

flow rate or flow velocity (m<sup>3</sup>/s or m/s)

## 7.19 readatmodep.f90 File Reference

### Functions/Subroutines

- subroutine [readatmodep](#)  
*this subroutine reads the atmospheric deposition values*

#### 7.19.1 Detailed Description

file containing the subroutine readatmodep

Author

modified by Javier Burguete

## 7.20 readbsn.f90 File Reference

### Functions/Subroutines

- subroutine [readbsn](#)

*this subroutine reads data from the basin input file (.bsn). This file contains information related to processes modeled or defined at the watershed level*

#### 7.20.1 Detailed Description

file containing the suborutine readbsn

Author

modified by Javier Burguete

## 7.21 readchm.f90 File Reference

### Functions/Subroutines

- subroutine [readchm](#)

*This subroutine reads data from the HRU/subbasin soil chemical input file (.chm). This file contains initial amounts of pesticides/nutrients in the first soil layer. (Specifics about the first soil layer are given in the .sol file.) All data in the .chm file is optional input.*

#### 7.21.1 Detailed Description

file containing the subroutine readchm

Author

modified by Javier Burguete

## 7.22 readcnst.f90 File Reference

### Functions/Subroutines

- subroutine [readcnst](#)

*reads in the loading information for the reccnst command*

#### 7.22.1 Detailed Description

file containing the subroutine [readcnst.f90](#)

Author

modified by Javier Burguete

## 7.23 readfcst.f90 File Reference

### Functions/Subroutines

- subroutine [readfcst](#)

*this subroutine reads the HRU forecast weather generator parameters from the .cst file*

#### 7.23.1 Detailed Description

file containing the subroutine readfcst

Author

modified by Javier Burguete

## 7.24 readfert.f90 File Reference

### Functions/Subroutines

- subroutine [readfert](#)

*this subroutine reads input parameters from the fertilizer/manure (i.e. nutrient) database (fert.dat)*

#### 7.24.1 Detailed Description

file containing the subroutine readfert

Author

modified by Javier Burguete

## 7.25 readfig.f90 File Reference

### Functions/Subroutines

- subroutine [readfig](#)

*reads in the routing information from the watershed configuration input file (.fig) and calculates the number of sub-basins, reaches, and reservoirs*

#### 7.25.1 Detailed Description

file containing the subroutine readfig

Author

modified by Javier Burguete

## 7.26 readfile.f90 File Reference

### Functions/Subroutines

- subroutine [readfile](#)

*this subroutine opens the main input and output files and reads watershed information from the file.cio*

#### 7.26.1 Detailed Description

file containing the subroutine readfile

Author

modified by Javier Burguete

## 7.27 readgw.f90 File Reference

### Functions/Subroutines

- subroutine [readgw](#)

*this subroutine reads the parameters from the HRU/subbasin groundwater input file (.gw)*

#### 7.27.1 Detailed Description

file containing the subroutine readgw

Author

modified by Javier Burguete

## 7.28 readhru.f90 File Reference

### Functions/Subroutines

- subroutine [readhru](#)

*this subroutine reads data from the HRU general input file (.hru). This file contains data related to general processes modeled at the HRU level.*

#### 7.28.1 Detailed Description

file containing the subroutine readhru

Author

modified by Javier Burguete

## 7.29 readinpt.f90 File Reference

### Functions/Subroutines

- subroutine [readinpt](#)  
*this subroutine calls subroutines which read input data for the databases and the HRUs*

#### 7.29.1 Detailed Description

file containing the subroutine readinpt

Author

modified by Javier Burguete

## 7.30 readlup.f90 File Reference

### Functions/Subroutines

- subroutine [readlup](#)  
*this subroutine reads data from the HRU/subbasin management input file (.mgt). This file contains data related to management practices used in the HRU/subbasin.*

#### 7.30.1 Detailed Description

file containing the subroutine readlup

Author

modified by Javier Burguete

## 7.31 readlwq.f90 File Reference

### Functions/Subroutines

- subroutine [readlwq](#)  
*this subroutine reads data from the lake water quality input file (.lwq). This file contains data related to initial pesticide and nutrient levels in the lake/reservoir and transformation processes occurring within the lake/reservoir. Data in the lake water quality input file is assumed to apply to all reservoirs in the watershed.*

#### 7.31.1 Detailed Description

file containing the subroutine readlwq

Author

modified by Javier Burguete

## 7.32 readmgt.f90 File Reference

### Functions/Subroutines

- subroutine [readmgt](#)  
*this subroutine reads data from the HRU/subbasin management input file (.mgt). This file contains data related to management practices used in the HRU/subbasin.*

#### 7.32.1 Detailed Description

file containing the subroutine readmgt

Author

modified by Javier Burguete

## 7.33 readmon.f90 File Reference

### Functions/Subroutines

- subroutine [readmon](#)  
*reads in the input data for the recmon command*

#### 7.33.1 Detailed Description

file containing the subroutine readmon

Author

modified by Javier Burguete

## 7.34 readops.f90 File Reference

### Functions/Subroutines

- subroutine [readops](#)  
*this subroutine reads data from the HRU/subbasin management input file (.mgt). This file contains data related to management practices used in the HRU/subbasin.*

#### 7.34.1 Detailed Description

file containing the subroutine readops

Author

modified by Javier Burguete



## 7.35 readpest.f90 File Reference

### Functions/Subroutines

- subroutine [readpest](#)  
*this subroutine reads parameters from the toxin/pesticide database (pest.dat)*

#### 7.35.1 Detailed Description

file containing the subroutine readpest

Author

modified by Javier Burguete

## 7.36 readplant.f90 File Reference

### Functions/Subroutines

- subroutine [readplant](#)  
*this subroutine reads input parameters from the landuse/landcover database (plant.dat)*

#### 7.36.1 Detailed Description

file containing the subroutine readplant

Author

modified by Javier Burguete

## 7.37 readpnd.f90 File Reference

### Functions/Subroutines

- subroutine [readpnd](#)  
*This subroutine reads data from the HRU/subbasin pond input file (.pnd). This file contains data related to ponds and wetlands in the HRUs/subbasins.*

#### 7.37.1 Detailed Description

file containing the subroutine readpnd

Author

modified by Javier Burguete

## 7.38 readres.f90 File Reference

### Functions/Subroutines

- subroutine [readres](#)

*the purpose of this subroutine is to read in data from the reservoir input file (.res)*

#### 7.38.1 Detailed Description

file containing the subroutine readres

Author

modified by Javier Burguete

## 7.39 readrte.f90 File Reference

### Functions/Subroutines

- subroutine [readrte](#)

*this subroutine reads data from the reach (main channel) input file (.rte). This file contains data related to channel attributes. Only one reach file should be made for each subbasin. If multiple HRUs are modeled within a subbasin, the same .rte file should be listed for all HRUs in file.cio*

#### 7.39.1 Detailed Description

file containing the subroutine readrte

Author

modified by Javier Burguete

## 7.40 readru.f90 File Reference

### Functions/Subroutines

- subroutine [readru](#)

*this subroutine reads data from the sub input file (.sub). This file contains data related to routing*

#### 7.40.1 Detailed Description

file containing the subroutine readru

Author

modified by Javier Burguete

## 7.41 readsdr.f90 File Reference

### Functions/Subroutines

- subroutine [readsdr](#)

*this subroutine reads data from the HRU/subbasin management input file (.mgt). This file contains data related to management practices used in the HRU/subbasin.*

#### 7.41.1 Detailed Description

file containing the subroutine readsdr

Author

modified by Javier Burguete

## 7.42 readsepticbz.f90 File Reference

### Functions/Subroutines

- subroutine [readsepticbz](#)

*this subroutine reads data from the septic input file (.sep). This file contains information related to septic tanks modeled or defined at the watershed level*

#### 7.42.1 Detailed Description

file containing the subroutine readsepticbz

Author

modified by Javier Burguete

## 7.43 readseptwq.f90 File Reference

### Functions/Subroutines

- subroutine [readseptwq](#)

*this subroutine reads input parameters from the sept wq database (septwq.dat). Information is used when a hru has septic tank.*

#### 7.43.1 Detailed Description

file containing the subroutine readseptwq

Author

C. Santhi, modified by Javier Burguete

## 7.43.2 Function/Subroutine Documentation

### 7.43.2.1 readseptwq()

```
subroutine readseptwq ( )
```

this subroutine reads input parameters from the sept wq database (septwq.dat). Information is used when a hru has septic tank.

This routine was developed by C. Santhi. Inputs for this routine are provided in septwq.dat of septic documentation. Data were compiled from [3] and [2].

## 7.44 readsno.f90 File Reference

### Functions/Subroutines

- subroutine [readsno](#)  
*this subroutine reads snow data from the HRU/subbasin soil chemical input*

### 7.44.1 Detailed Description

file containing the subroutine readsno

Author

modified by Javier Burguete

## 7.45 readsol.f90 File Reference

### Functions/Subroutines

- subroutine [readsol](#)  
*this subroutine reads data from the HRU/subbasin soil properties file (.sol). This file contains data related to soil physical properties and general chemical properties.*

### 7.45.1 Detailed Description

file containing the subroutine readsol

Author

modified by Javier Burguete

## 7.46 readsub.f90 File Reference

### Functions/Subroutines

- subroutine [readsub](#)

*this subroutine reads data from the HRU/subbasin general input file (.sub). This file contains data related to general processes modeled at the HRU/subbasin level.*

#### 7.46.1 Detailed Description

file containing the subroutine readsub

Author

modified by Javier Burguete

## 7.47 readswq.f90 File Reference

### Functions/Subroutines

- subroutine [readswq](#)

*this subroutine reads parameters from the subbasin instream water quality file (.swq) and initializes the QUAL2E variables which apply to the individual subbasins*

#### 7.47.1 Detailed Description

file containing the subroutine readswq

Author

modified by Javier Burguete

## 7.48 readtill.f90 File Reference

### Functions/Subroutines

- subroutine [readtill](#)

*this subroutine reads input data from tillage database (till.dat)*

#### 7.48.1 Detailed Description

file containing the subroutine readtill

Author

modified by Javier Burguete

## 7.49 readurban.f90 File Reference

### Functions/Subroutines

- subroutine [readurban](#)

*this subroutine reads input parameters from the urban database (urban.dat). Information from this database is used only if the urban buildup/washoff routines are selected for the modeling of urban areas*

### 7.49.1 Detailed Description

file containing the subroutine readurban

Author

modified by Javier Burguete

## 7.50 readwgn.f90 File Reference

### Functions/Subroutines

- subroutine [readwgn](#)

*this subroutine reads the HRU weather generator parameters from the .wgn file*

### 7.50.1 Detailed Description

file containing the subroutine readwgn

Author

modified by Javier Burguete

## 7.51 readwus.f90 File Reference

### Functions/Subroutines

- subroutine [readwus](#)

*This subroutine reads data from the HRU/subbasin water use input file (.wus). The water use file extracts water from the subbasin and it is considered to be lost from the watershed. These variables should be used to remove water transported outside the watershed.*

### 7.51.1 Detailed Description

file containing the subroutine readwus

Author

modified by Javier Burguete

## 7.52 readwwq.f90 File Reference

### Functions/Subroutines

- subroutine [readwwq](#)  
*this subroutine reads the watershed stream water quality input data (.wwq file) and initializes the QUAL2E variables which apply to the entire watershed*

#### 7.52.1 Detailed Description

file containing the subroutine readwwq

Author

modified by Javier Burguete

## 7.53 readyr.f90 File Reference

### Functions/Subroutines

- subroutine [readyr](#)  
*reads in the input data for the recyear command*

#### 7.53.1 Detailed Description

file containing the subroutine readyr

Author

modified by Javier Burguete

## 7.54 rteinit.f90 File Reference

### Functions/Subroutines

- subroutine [rteinit](#)  
*This subroutine reads in the areas associated with files processed with the recday, recepic, recmon and recyear commands, calculates subbasin areas, calculates reach and hydrograph node drainage areas.*

#### 7.54.1 Detailed Description

file containing the subroutine rteinit

Author

modified by Javier Burguete

## 7.55 `sim_initr.f90` File Reference

### Functions/Subroutines

- subroutine `sim_initr`  
*this subroutine initializes variables at the beginning of the year*

#### 7.55.1 Detailed Description

file containing the subroutine `sim_initr`

Author

modified by Javier Burguete

## 7.56 `simulate.f90` File Reference

### Functions/Subroutines

- subroutine `simulate`  
*this subroutine contains the loops governing the modeling of processes in the watershed*

#### 7.56.1 Detailed Description

file containing the subroutine `simulate`

Author

modified by Javier Burguete

## 7.57 `soil_chem.f90` File Reference

### Functions/Subroutines

- subroutine `soil_chem` (ii)  
*this subroutine initializes soil chemical properties*

#### 7.57.1 Detailed Description

file containing the subroutine `soil_chem`

Author

modified by Javier Burguete

#### 7.57.2 Function/Subroutine Documentation

##### 7.57.2.1 `soil_chem()`

```
subroutine soil_chem (  
    integer, intent(in) ii )
```

this subroutine initializes soil chemical properties



## Parameters

|                 |                 |            |
|-----------------|-----------------|------------|
| <code>in</code> | <code>ii</code> | HRU number |
|-----------------|-----------------|------------|

## 7.58 soil\_phys.f90 File Reference

### Functions/Subroutines

- subroutine `soil_phys` (ii)  
*this subroutine initializes soil physical properties*

#### 7.58.1 Detailed Description

file containing the subroutine `soil_phys`

## Author

modified by Javier Burguete

#### 7.58.2 Function/Subroutine Documentation

##### 7.58.2.1 soil\_phys()

```
subroutine soil_phys (
    integer, intent(in) ii )
```

this subroutine initializes soil physical properties

## Parameters

|                 |                 |            |
|-----------------|-----------------|------------|
| <code>in</code> | <code>ii</code> | HRU number |
|-----------------|-----------------|------------|

## 7.59 std1.f90 File Reference

### Functions/Subroutines

- subroutine `std1`  
*this subroutine writes general information to the standard output file and header lines to miscellaneous output files*

### 7.59.1 Detailed Description

file containing the subroutine std1

Author

modified by Javier Burguete

## 7.60 std2.f90 File Reference

### Functions/Subroutines

- subroutine [std2](#)  
*this subroutine writes general information to the standard output file and to miscellaneous output files*

### 7.60.1 Detailed Description

file containing the subroutine std2

Author

modified by Javier Burguete

## 7.61 std3.f90 File Reference

### Functions/Subroutines

- subroutine [std3](#)  
*this subroutine writes the annual table header to the standard output file*

### 7.61.1 Detailed Description

file containing the subroutine std3

Author

modified by Javier Burguete

## 7.62 storeinitial.f90 File Reference

### Functions/Subroutines

- subroutine [storeinitial](#)  
*this subroutine saves initial values for variables that must be reset to rerun the simulation for different real time weather scenarios*

### 7.62.1 Detailed Description

file containing the subroutine storeinitial

Author

modified by Javier Burguete

## 7.63 ttcoef.f90 File Reference

### Functions/Subroutines

- subroutine [ttcoef](#) (k)  
*this subroutine computes travel time coefficients for routing along the main channel*

### 7.63.1 Detailed Description

file containing the subroutine ttcoef

Author

modified by Javier Burguete

### 7.63.2 Function/Subroutine Documentation

#### 7.63.2.1 ttcoef()

```
subroutine ttcoef (
    integer, intent(in) k )
```

this subroutine computes travel time coefficients for routing along the main channel

Parameters

|    |   |            |
|----|---|------------|
| in | k | HRU number |
|----|---|------------|

## 7.64 xmon.f90 File Reference

### Functions/Subroutines

- subroutine [xmon](#)  
*this subroutine determines the month, given the julian date and leap year flag*

### 7.64.1 Detailed Description

file containing the subroutine xmon

Author

modified by Javier Burguete

## 7.65 zero0.f90 File Reference

### Functions/Subroutines

- subroutine [zero0](#)  
*this subroutine initializes the values for some of the arrays*

### 7.65.1 Detailed Description

file containing the subroutine zero0

Author

modified by Javier Burguete

## 7.66 zero1.f90 File Reference

### Functions/Subroutines

- subroutine [zero1](#)  
*this subroutine initializes the values for some of the arrays*

### 7.66.1 Detailed Description

file containing the subroutine zero1

Author

modified by Javier Burguete

## 7.67 zero2.f90 File Reference

### Functions/Subroutines

- subroutine [zero2](#)  
*this subroutine zeros all array values*

### 7.67.1 Detailed Description

file containing the subroutine zero2

Author

modified by Javier Burguete

## 7.68 zero\_urban.f90 File Reference

### Functions/Subroutines

- subroutine [zero\\_urban](#)  
*this subroutine zeros all array values used in urban modeling*

### 7.68.1 Detailed Description

file containing the subroutine zero\_urban

Author

modified by Javier Burguete

## 7.69 zeroini.f90 File Reference

### Functions/Subroutines

- subroutine [zeroini](#)  
*this subroutine zeros values for single array variables*

### 7.69.1 Detailed Description

file containing the subroutine zeroini

Author

modified by Javier Burguete



# Bibliography

- [1] P Bratley, B L Fox, and L E Schrage. *A Guide to Simulation*. Springer-Verlag, New York, USA, 1983. [89](#)
- [2] J. E. McCray, S. L. Kirkland, R. L. Siegrist, and G. D. Thyne. Model parameters for simulating fate and transport of on-site wastewater nutrients. *Ground Water*, 43(4):628–639, 2005. [172](#)
- [3] R. L. Siegrist, J. McCray, L. Weintraub, C. Chen, J. Bagdol, P. Lemonds, S. Van Cuyk, K. Lowe, R. Goldstein, and J. Rada. Quantifying site-scale processes and watershed-scale cumulative effects of decentralized wastewater systems, project no. wu-ht-00-27. Prepared for the National Decentralized Water Resources Capacity Development Project, Washington University, St. Louis, MO, by the Colorado School of Mines, 2005. [172](#)





# Index

allocate\_parms.f90, [87](#)

ascrv

ascrv.f90, [88](#)

ascrv.f90, [87](#)

ascrv, [88](#)

aunif

aunif.f90, [89](#)

aunif.f90, [88](#)

aunif, [89](#)

caps

caps.f90, [89](#)

caps.f90, [89](#)

caps, [89](#)

estimate\_ksat

estimate\_ksat.f90, [90](#)

estimate\_ksat.f90, [90](#)

estimate\_ksat, [90](#)

gcycl.f90, [90](#)

getallo.f90, [91](#)

h2omgt\_init.f90, [91](#)

headout.f90, [92](#)

hruallo.f90, [92](#)

hydroinit.f90, [92](#)

igropt

parm, [80](#)

impnd\_init.f90, [93](#)

jdt

jdt.f90, [93](#)

jdt.f90, [93](#)

jdt, [93](#)

lwqdef.f90, [94](#)

main.f90, [94](#)

modparm.f90, [95](#)

openwth.f90, [162](#)

parm, [13](#)

igropt, [80](#)

parm::ascrv, [83](#)

parm::atri, [83](#)

parm::aunif, [83](#)

parm::dstn1, [84](#)

parm::ee, [84](#)

parm::expo, [84](#)

parm::fcgd, [84](#)

parm::HQDAV, [85](#)

parm::layersplit, [85](#)

parm::ndenit, [85](#)

parm::qman, [85](#)

parm::regres, [86](#)

parm::rsedaa, [86](#)

parm::tair, [86](#)

parm::theta, [86](#)

parm::vbl, [86](#)

qman

qman.f90, [163](#)

qman.f90, [162](#)

qman, [163](#)

readatmodep.f90, [163](#)

readbsn.f90, [164](#)

readchm.f90, [164](#)

readcnst.f90, [164](#)

readfcst.f90, [165](#)

readfert.f90, [165](#)

readfig.f90, [165](#)

readfile.f90, [166](#)

readgw.f90, [166](#)

readhru.f90, [166](#)

readinpt.f90, [167](#)

readlup.f90, [167](#)

readlwq.f90, [167](#)

readmgt.f90, [168](#)

readmon.f90, [168](#)

readops.f90, [168](#)

readpest.f90, [169](#)

readplant.f90, [169](#)

readpnd.f90, [169](#)

readres.f90, [170](#)

readrte.f90, [170](#)

readru.f90, [170](#)

readsdr.f90, [171](#)

readsepticbz.f90, [171](#)

readseptwq

readseptwq.f90, [172](#)

readseptwq.f90, [171](#)

readseptwq, [172](#)

readsno.f90, [172](#)

readsol.f90, [172](#)

readsub.f90, [173](#)

readswq.f90, [173](#)

readtill.f90, [173](#)

readurban.f90, [174](#)  
readwgn.f90, [174](#)  
readwus.f90, [174](#)  
readwwq.f90, [175](#)  
readyr.f90, [175](#)  
rteinit.f90, [175](#)  
  
sim\_inityr.f90, [176](#)  
simulate.f90, [176](#)  
soil\_chem  
    soil\_chem.f90, [176](#)  
soil\_chem.f90, [176](#)  
    soil\_chem, [176](#)  
soil\_phys  
    soil\_phys.f90, [177](#)  
soil\_phys.f90, [177](#)  
    soil\_phys, [177](#)  
std1.f90, [177](#)  
std2.f90, [178](#)  
std3.f90, [178](#)  
storeinitial.f90, [178](#)  
  
ttcoef  
    ttcoef.f90, [179](#)  
ttcoef.f90, [179](#)  
    ttcoef, [179](#)  
  
xmon.f90, [179](#)  
  
zero0.f90, [180](#)  
zero1.f90, [180](#)  
zero2.f90, [180](#)  
zero\_urbn.f90, [181](#)  
zeroini.f90, [181](#)