

MAIC-2

Generated by Doxygen 1.6.3

Wed Apr 20 14:07:47 2011



# Contents

|          |   |          |
|----------|---|----------|
| <b>1</b> | <b>Main Page</b>                            | <b>1</b> |
| 1.1      | Description . . . . .                       | 1        |
| 1.2      | Copyright . . . . .                         | 2        |
| 1.3      | License . . . . .                           | 2        |
| <b>2</b> | <b>Modules Index</b>                        | <b>3</b> |
| 2.1      | Modules List . . . . .                      | 3        |
| <b>3</b> | <b>Data Type Index</b>                      | <b>5</b> |
| 3.1      | Data Types List . . . . .                   | 5        |
| <b>4</b> | <b>File Index</b>                           | <b>7</b> |
| 4.1      | File List . . . . .                         | 7        |
| <b>5</b> | <b>Module Documentation</b>                 | <b>9</b> |
| 5.1      | condensation Module Reference . . . . .     | 9        |
| 5.1.1    | Detailed Description . . . . .              | 9        |
| 5.1.2    | Function/Subroutine Documentation . . . . . | 10       |
| 5.1.2.1  | getcond_1 . . . . .                         | 10       |
| 5.1.2.2  | getcond_2 . . . . .                         | 10       |
| 5.1.2.3  | setcondpar . . . . .                        | 10       |
| 5.1.3    | Variable Documentation . . . . .            | 10       |
| 5.1.3.1  | g . . . . .                                 | 10       |
| 5.1.3.2  | tau . . . . .                               | 10       |
| 5.2      | evaporation Module Reference . . . . .      | 11       |
| 5.2.1    | Detailed Description . . . . .              | 11       |
| 5.2.2    | Function/Subroutine Documentation . . . . . | 11       |

|         |   |    |
|---------|---|----|
| 5.2.2.1 | getevap_bd . . . . .                        | 11 |
| 5.2.2.2 | setevappar_bd . . . . .                     | 11 |
| 5.2.3   | Variable Documentation . . . . .            | 12 |
| 5.2.3.1 | diff_w_c . . . . .                          | 12 |
| 5.2.3.2 | evap_fact . . . . .                         | 12 |
| 5.2.3.3 | g . . . . .                                 | 12 |
| 5.2.3.4 | gamma_reg . . . . .                         | 12 |
| 5.2.3.5 | mol_c . . . . .                             | 12 |
| 5.2.3.6 | mol_w . . . . .                             | 12 |
| 5.2.3.7 | R_univ . . . . .                            | 12 |
| 5.2.3.8 | visc_c . . . . .                            | 12 |
| 5.3     | instemp Module Reference . . . . .          | 13 |
| 5.3.1   | Detailed Description . . . . .              | 13 |
| 5.3.2   | Function/Subroutine Documentation . . . . . | 13 |
| 5.3.2.1 | inst . . . . .                              | 13 |
| 5.3.2.2 | inst1 . . . . .                             | 14 |
| 5.3.2.3 | instam . . . . .                            | 14 |
| 5.3.2.4 | instmax . . . . .                           | 14 |
| 5.3.2.5 | setinstemp . . . . .                        | 14 |
| 5.4     | maic2_types Module Reference . . . . .      | 15 |
| 5.4.1   | Detailed Description . . . . .              | 15 |
| 5.4.2   | Variable Documentation . . . . .            | 15 |
| 5.4.2.1 | dp . . . . .                                | 15 |
| 5.4.2.2 | i2b . . . . .                               | 15 |
| 5.4.2.3 | i4b . . . . .                               | 15 |
| 5.4.2.4 | sp . . . . .                                | 16 |
| 5.5     | maic2_variables Module Reference . . . . .  | 17 |
| 5.5.1   | Detailed Description . . . . .              | 20 |
| 5.5.2   | Variable Documentation . . . . .            | 20 |
| 5.5.2.1 | a_net . . . . .                             | 20 |
| 5.5.2.2 | ave_data . . . . .                          | 20 |
| 5.5.2.3 | cond . . . . .                              | 20 |
| 5.5.2.4 | cos_phi_cb1 . . . . .                       | 20 |
| 5.5.2.5 | cos_phi_cb2 . . . . .                       | 20 |

|          |                          |    |
|----------|--------------------------|----|
| 5.5.2.6  | cp_data . . . . .        | 21 |
| 5.5.2.7  | diff_aux . . . . .       | 21 |
| 5.5.2.8  | dphi . . . . .           | 21 |
| 5.5.2.9  | dphi_inv . . . . .       | 21 |
| 5.5.2.10 | dtime_out . . . . .      | 21 |
| 5.5.2.11 | ecc_data . . . . .       | 21 |
| 5.5.2.12 | eps . . . . .            | 21 |
| 5.5.2.13 | evap . . . . .           | 21 |
| 5.5.2.14 | G . . . . .              | 22 |
| 5.5.2.15 | H . . . . .              | 22 |
| 5.5.2.16 | H_new . . . . .          | 22 |
| 5.5.2.17 | insol_ma_90 . . . . .    | 22 |
| 5.5.2.18 | insol_time_max . . . . . | 22 |
| 5.5.2.19 | insol_time_min . . . . . | 22 |
| 5.5.2.20 | insol_time_stp . . . . . | 22 |
| 5.5.2.21 | iter_out . . . . .       | 22 |
| 5.5.2.22 | iter_output . . . . .    | 23 |
| 5.5.2.23 | itercount . . . . .      | 23 |
| 5.5.2.24 | n_output . . . . .       | 23 |
| 5.5.2.25 | obl_data . . . . .       | 23 |
| 5.5.2.26 | p_surf . . . . .         | 23 |
| 5.5.2.27 | phi_cb1 . . . . .        | 23 |
| 5.5.2.28 | phi_cb2 . . . . .        | 23 |
| 5.5.2.29 | phi_node . . . . .       | 23 |
| 5.5.2.30 | pi . . . . .             | 24 |
| 5.5.2.31 | pi_180 . . . . .         | 24 |
| 5.5.2.32 | pi_180_inv . . . . .     | 24 |
| 5.5.2.33 | pi_inv . . . . .         | 24 |
| 5.5.2.34 | psi_tab . . . . .        | 24 |
| 5.5.2.35 | R . . . . .              | 24 |
| 5.5.2.36 | RHO . . . . .            | 24 |
| 5.5.2.37 | RHO_I . . . . .          | 24 |
| 5.5.2.38 | rho_inv . . . . .        | 25 |
| 5.5.2.39 | RHO_W . . . . .          | 25 |

|          |   |           |
|----------|---|-----------|
| 5.5.2.40 | sin_phi_cb1 . . . . .                                   | 25        |
| 5.5.2.41 | sin_phi_cb2 . . . . .                                   | 25        |
| 5.5.2.42 | temp_co2 . . . . .                                      | 25        |
| 5.5.2.43 | temp_surf . . . . .                                     | 25        |
| 5.5.2.44 | temp_surf_amp . . . . .                                 | 25        |
| 5.5.2.45 | time_output . . . . .                                   | 25        |
| 5.5.2.46 | water . . . . .   | 26        |
| 5.5.2.47 | water_new . . . . .                                     | 26        |
| <b>6</b> | <b>Data Type Documentation</b>                          | <b>27</b> |
| 6.1      | instemp::ins Type Reference . . . . .                   | 27        |
| 6.1.1    | Detailed Description . . . . .                          | 27        |
| 6.1.2    | Member Data Documentation . . . . .                     | 27        |
| 6.1.2.1  | t . . . . .   | 27        |
| 6.1.2.2  | tam . . . . .   | 27        |
| 6.1.2.3  | tmax . . . . .  | 27        |
| <b>7</b> | <b>File Documentation</b>                               | <b>29</b> |
| 7.1      | maic2.F90 File Reference . . . . .                      | 29        |
| 7.1.1    | Detailed Description . . . . .                          | 29        |
| 7.1.2    | Copyright . . . . .                                     | 29        |
| 7.1.3    | License . . . . .                                       | 29        |
| 7.1.4    | Function Documentation . . . . .                        | 30        |
| 7.1.4.1  | maic2 . . . . .   | 30        |
| 7.2      | subroutines/boundary_maic2.F90 File Reference . . . . . | 31        |
| 7.2.1    | Detailed Description . . . . .                          | 31        |
| 7.2.2    | Copyright . . . . .                                     | 31        |
| 7.2.3    | License . . . . .                                       | 31        |
| 7.2.4    | Function Documentation . . . . .                        | 31        |
| 7.2.4.1  | boundary_maic2 . . . . .                                | 31        |
| 7.3      | subroutines/calc_top_maic2.F90 File Reference . . . . . | 32        |
| 7.3.1    | Detailed Description . . . . .                          | 32        |
| 7.3.2    | Copyright . . . . .                                     | 32        |
| 7.3.3    | License . . . . .                                       | 32        |
| 7.3.4    | Function Documentation . . . . .                        | 32        |

---

|         |   |    |
|---------|---|----|
| 7.3.4.1 | calc_top_maic2 . . . . .                              | 32 |
| 7.4     | subroutines/condensation.f90 File Reference . . . . . | 33 |
| 7.4.1   | Detailed Description . . . . .                        | 33 |
| 7.4.2   | Copyright . . . . .                                   | 33 |
| 7.4.3   | License . . . . .                                     | 33 |
| 7.5     | subroutines/diff_trans.F90 File Reference . . . . .   | 35 |
| 7.5.1   | Detailed Description . . . . .                        | 35 |
| 7.5.2   | Copyright . . . . .                                   | 35 |
| 7.5.3   | License . . . . .                                     | 35 |
| 7.5.4   | Function Documentation . . . . .                      | 35 |
| 7.5.4.1 | diff_trans . . . . .                                  | 35 |
| 7.6     | subroutines/evaporation.f90 File Reference . . . . .  | 36 |
| 7.6.1   | Detailed Description . . . . .                        | 36 |
| 7.6.2   | Copyright . . . . .                                   | 36 |
| 7.6.3   | License . . . . .                                     | 37 |
| 7.7     | subroutines/get_orb_par.F90 File Reference . . . . .  | 38 |
| 7.7.1   | Detailed Description . . . . .                        | 38 |
| 7.7.2   | Copyright . . . . .                                   | 38 |
| 7.7.3   | License . . . . .                                     | 38 |
| 7.7.4   | Function Documentation . . . . .                      | 38 |
| 7.7.4.1 | get_orb_par . . . . .                                 | 38 |
| 7.8     | subroutines/get_psi_tab.F90 File Reference . . . . .  | 40 |
| 7.8.1   | Detailed Description . . . . .                        | 40 |
| 7.8.2   | Copyright . . . . .                                   | 40 |
| 7.8.3   | License . . . . .                                     | 40 |
| 7.8.4   | Function Documentation . . . . .                      | 40 |
| 7.8.4.1 | get_psi_tab . . . . .                                 | 40 |
| 7.9     | subroutines/instemp.f90 File Reference . . . . .      | 41 |
| 7.9.1   | Detailed Description . . . . .                        | 41 |
| 7.9.2   | Copyright . . . . .                                   | 41 |
| 7.9.3   | License . . . . .                                     | 42 |
| 7.10    | subroutines/maic2_types.F90 File Reference . . . . .  | 43 |
| 7.10.1  | Detailed Description . . . . .                        | 43 |
| 7.10.2  | Copyright . . . . .                                   | 43 |

|   |    |
|---|----|
| 7.10.3 License . . . . .                                      | 43 |
| 7.11 subroutines/maic2_variables.F90 File Reference . . . . . | 45 |
| 7.11.1 Detailed Description . . . . .                         | 48 |
| 7.11.2 Copyright . . . . .                                    | 48 |
| 7.11.3 License . . . . .                                      | 48 |
| 7.12 subroutines/output.F90 File Reference . . . . .          | 49 |
| 7.12.1 Detailed Description . . . . .                         | 49 |
| 7.12.2 Copyright . . . . .                                    | 49 |
| 7.12.3 License . . . . .                                      | 49 |
| 7.12.4 Function Documentation . . . . .                       | 49 |
| 7.12.4.1 output . . . . .                                     | 49 |
| 7.13 subroutines/p_sat.f90 File Reference . . . . .           | 50 |
| 7.13.1 Detailed Description . . . . .                         | 50 |
| 7.13.2 Copyright . . . . .                                    | 50 |
| 7.13.3 License . . . . .                                      | 50 |
| 7.13.4 Function Documentation . . . . .                       | 50 |
| 7.13.4.1 p_sat . . . . .                                      | 50 |
| 7.14 subroutines/tri_sle.F90 File Reference . . . . .         | 51 |
| 7.14.1 Detailed Description . . . . .                         | 51 |
| 7.14.2 Copyright . . . . .                                    | 51 |
| 7.14.3 License . . . . .                                      | 51 |
| 7.14.4 Function Documentation . . . . .                       | 51 |
| 7.14.4.1 tri_sle . . . . .                                    | 51 |

# Chapter 1

## Main Page

### 1.1 Description

The Mars Atmosphere-Ice Coupler MAIC-2 is a simple, latitudinal model, which consists of a set of parameterisations for the surface temperature, the atmospheric water transport and the surface mass balance (condensation minus evaporation) of water ice. It is driven directly by orbital parameters. A detailed description of the model is given in the publication by Greve et al. (2010).

The model equations of MAIC-2 are discretised by a finite-difference/finite-volume scheme. Coding is done in the programming language Fortran 90.

Required model forcing (as functions of time):

- Obliquity (axial tilt).
- Orbital eccentricity.
- Solar longitude ( $L_s$ ) of perihelion.

Output (as functions of latitude and time):

- Surface temperature.
- Evaporation rate of water ice.
- Condensation rate of water ice.
- Atmospheric water content.
- Surface mass balance of water ice.
- Ice thickness.

References:

- Greve, R., B. Grieger and O. J. Stenzel. 2010.  
MAIC-2, a latitudinal model for the Martian surface temperature, atmospheric water transport and surface glaciation.  
Planetary and Space Science 58 (6), 931-940.
- MAIC-2 website: <http://maic2.greveweb.net/>

## 1.2 Copyright

Copyright 2010, 2011 Ralf Greve, Bjoern Grieger, Oliver J. Stenzel

## 1.3 License

MAIC-2 is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

MAIC-2 is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with MAIC-2. If not, see <http://www.gnu.org/licenses/>.

## Chapter 2

# Modules Index

### 2.1 Modules List

Here is a list of all modules with brief descriptions:

|   |    |
|---|----|
| <a href="#">condensation</a> (Computation of the condensation rate ) . . . . .  | 9  |
| <a href="#">evaporation</a> (Computation of the evaporation rate (buoyant-diffusion approach by Ingersoll) ) . . . . .  | 11 |
| <a href="#">instemp</a> (Computation of the daily mean surface temperature of Mars based on obliquity, eccentricity and the anomaly of vernal equinox (local insolation temperature = LIS scheme) ) . . . . . | 13 |
| <a href="#">maic2_types</a> (Declarations of kind types for MAIC-2 ) . . . . .  | 15 |
| <a href="#">maic2_variables</a> (Declarations of global variables for MAIC-2 ) . . . . .  | 17 |



## Chapter 3

# Data Type Index

### 3.1 Data Types List

Here are the data types with brief descriptions:

[instemp::ins](#) (Surface temperatures ) . . . . . 27



## Chapter 4

# File Index

### 4.1 File List

Here is a list of all files with brief descriptions:

|  |    |
|--|----|
| <a href="#">maic2.F90</a> (Main program file of MAIC-2 ) . . . . .   | 29 |
| subroutines/ <a href="#">boundary_maic2.F90</a> (Determination of the surface temperature<br>and of the net mass balance (accumulation-ablation rate) for the<br>polar caps of Mars ) . . . . .                                      | 31 |
| subroutines/ <a href="#">calc_top_maic2.F90</a> (Computation of the ice-cap topography ) . .   | 32 |
| subroutines/ <a href="#">condensation.f90</a> (Computation of the condensation rate ) . . . .  | 33 |
| subroutines/ <a href="#">diff_trans.F90</a> (Diffusive transport of water in the Martian atmo-<br>sphere ) . . . . .   | 35 |
| subroutines/ <a href="#">evaporation.f90</a> (Computation of the evaporation rate (buoyant-<br>diffusion approach by Ingersoll) ) . . . . .  | 36 |
| subroutines/ <a href="#">get_orb_par.F90</a> (Determination of the orbital parameters (ec-<br>centricity, obliquity, climate parameter CP, anomaly of vernal<br>equinox, mean-annual north- or south-polar insolation) ) . . . . .   | 38 |
| subroutines/ <a href="#">get_psi_tab.F90</a> (Computation of the table of true anomalies (or-<br>bital positions with respect to perihelion) over a Martian year ) . . .   | 40 |
| subroutines/ <a href="#">instemp.f90</a> (Computation of the daily mean surface temperature<br>of Mars based on obliquity, eccentricity and the anomaly of vernal<br>equinox (local insolation temperature = LIS scheme) ) . . . . . | 41 |
| subroutines/ <a href="#">maic2_types.F90</a> (Declarations of kind types for MAIC-2 ) . . .  | 43 |
| subroutines/ <a href="#">maic2_variables.F90</a> (Declarations of global variables for<br>MAIC-2 ) . . . . .   | 45 |
| subroutines/ <a href="#">output.F90</a> (Data output ) . . . . .   | 49 |
| subroutines/ <a href="#">p_sat.f90</a> (Computation of the water-vapour saturation pressure )  | 50 |
| subroutines/ <a href="#">tri_sle.F90</a> (Solution of a system of linear equations $Ax=b$ with<br>tridiagonal matrix A ) . . . . .   | 51 |



# Chapter 5

## Module Documentation

### 5.1 condensation Module Reference

Computation of the condensation rate.

#### Functions/Subroutines

- subroutine [setcondpar](#) (gravity, timescale)  
*Setting of parameters.*
- subroutine [getcond\\_1](#) (temp, water, cond, dtime)  
*Computation of condensation (removal of water exceeding the saturation pressure at the surface).*
- subroutine [getcond\\_2](#) (temp, water, cond)  
*Computation of condensation (continuous, quadratic dependence on humidity).*

#### Variables

- real(dp) [g](#)
- real(dp) [tau](#)

#### 5.1.1 Detailed Description

Computation of the condensation rate.

## 5.1.2 Function/Subroutine Documentation

### 5.1.2.1 subroutine condensation::getcond\_1 (real(dp),dimension(:) *temp*, real(dp),dimension(:) *water*, real(dp),dimension(:) *cond*, real(dp) *dtime*)

Computation of condensation (removal of water exceeding the saturation pressure at the surface).

Definition at line 72 of file condensation.f90.

### 5.1.2.2 subroutine condensation::getcond\_2 (real(dp),dimension(:) *temp*, real(dp),dimension(:) *water*, real(dp),dimension(:) *cond*)

Computation of condensation (continuous, quadratic dependence on humidity).

Definition at line 102 of file condensation.f90.

### 5.1.2.3 subroutine condensation::setcondpar (real(dp),optional *gravity*, real(dp),optional *timescale*)

Setting of parameters.

Definition at line 48 of file condensation.f90.

## 5.1.3 Variable Documentation

### 5.1.3.1 real(dp) condensation::g

Definition at line 40 of file condensation.f90.

### 5.1.3.2 real(dp) condensation::tau

Definition at line 41 of file condensation.f90.

## 5.2 evaporation Module Reference

Computation of the evaporation rate (buoyant-diffusion approach by Ingersoll).

### Functions/Subroutines

- subroutine [setevappar\\_bd](#) (evap\_fact\_para, gamma\_reg\_para, R\_univ\_para, mol\_w\_para, mol\_c\_para, diff\_w\_c\_para, visc\_c\_para, g\_para)  
*Setting of parameters.*
- subroutine [getevap\\_bd](#) (temp, temp\_amp, p, H, evap)  
*Computation of evaporation.*

### Variables

- real(dp) [evap\\_fact](#)
- real(dp) [gamma\\_reg](#)
- real(dp) [R\\_univ](#)
- real(dp) [mol\\_w](#)
- real(dp) [mol\\_c](#)
- real(dp) [diff\\_w\\_c](#)
- real(dp) [visc\\_c](#)
- real(dp) [g](#)

#### 5.2.1 Detailed Description

Computation of the evaporation rate (buoyant-diffusion approach by Ingersoll).

#### 5.2.2 Function/Subroutine Documentation

- 5.2.2.1 subroutine evaporation::getevap\_bd** (real(dp),dimension(:) *temp*, real(dp),dimension(:) *temp\_amp*, real(dp),dimension(:) *p*, real(dp),dimension(:) *H*, real(dp),dimension(:) *evap*)

Computation of evaporation.

Definition at line 113 of file evaporation.f90.

- 5.2.2.2 subroutine evaporation::setevappar\_bd** (real(dp),optional *evap\_fact\_para*, real(dp),optional *gamma\_reg\_para*, real(dp),optional *R\_univ\_para*, real(dp),optional *mol\_w\_para*, real(dp),optional *mol\_c\_para*, real(dp),optional *diff\_w\_c\_para*, real(dp),optional *visc\_c\_para*, real(dp),optional *g\_para*)

Setting of parameters.

Definition at line 50 of file evaporation.f90.

### **5.2.3 Variable Documentation**

#### **5.2.3.1 real(dp) evaporation::diff\_w\_c**

Definition at line 42 of file evaporation.f90.

#### **5.2.3.2 real(dp) evaporation::evap\_fact**

Definition at line 42 of file evaporation.f90.

#### **5.2.3.3 real(dp) evaporation::g**

Definition at line 42 of file evaporation.f90.

#### **5.2.3.4 real(dp) evaporation::gamma\_reg**

Definition at line 42 of file evaporation.f90.

#### **5.2.3.5 real(dp) evaporation::mol\_c**

Definition at line 42 of file evaporation.f90.

#### **5.2.3.6 real(dp) evaporation::mol\_w**

Definition at line 42 of file evaporation.f90.

#### **5.2.3.7 real(dp) evaporation::R\_univ**

Definition at line 42 of file evaporation.f90.

#### **5.2.3.8 real(dp) evaporation::visc\_c**

Definition at line 42 of file evaporation.f90.

## 5.3 instemp Module Reference

Computation of the daily mean surface temperature of Mars based on obliquity, eccentricity and the anomaly of vernal equinox (local insolation temperature = LIS scheme).

### Data Types

- type [ins](#)

*Surface temperatures.*

### Functions/Subroutines

- subroutine [setinstemp](#) (o, ecc, ave, obl, sma, sa, sac, op, ct)

*Main subroutine of module instemp.*

- real(dp) [instam](#) (o, phi)

*Annual mean temperature at latitude phi.*

- real(dp) [instmax](#) (o, phi)

*Annual maximum temperature at latitude phi.*

- real(dp) [inst](#) (o, psi, phi)

*Temperature at orbit position psi and latitude phi.*

- real(dp) [inst1](#) (o, ipsi, phi)

*Temperature at orbit position ipsi (integer value) and latitude phi.*

#### 5.3.1 Detailed Description

Computation of the daily mean surface temperature of Mars based on obliquity, eccentricity and the anomaly of vernal equinox (local insolation temperature = LIS scheme).

#### 5.3.2 Function/Subroutine Documentation

##### 5.3.2.1 real(dp) instemp::inst (type(ins) o, real(dp) psi, real(dp) phi)

Temperature at orbit position psi and latitude phi.

Definition at line 272 of file instemp.f90.

**5.3.2.2 real(dp) instemp::inst1 (type(ins) *o*, integer(i4b) *ipsi*, real(dp) *phi*)**

Temperature at orbit position *ipsi* (integer value) and latitude *phi*.

Definition at line 294 of file instemp.f90.

**5.3.2.3 real(dp) instemp::instam (type(ins) *o*, real(dp) *phi*)**

Annual mean temperature at latitude *phi*.

Definition at line 228 of file instemp.f90.

**5.3.2.4 real(dp) instemp::instmax (type(ins) *o*, real(dp) *phi*)**

Annual maximum temperature at latitude *phi*.

Definition at line 250 of file instemp.f90.

**5.3.2.5 subroutine instemp::setinstemp (type(ins) *o*, real(dp),optional *ecc*,  
real(dp),optional *ave*, real(dp),optional *obl*, real(dp),optional *sma*,  
real(dp),optional *sa*, real(dp),optional *sac*, real(dp),optional *op*,  
real(dp),optional *ct*)**

Main subroutine of module instemp.

Definition at line 57 of file instemp.f90.

## 5.4 maic2\_types Module Reference

Declarations of kind types for MAIC-2.

### Variables

- integer, parameter **i2b** = selected\_int\_kind(4)  
*2-byte integers*
- integer, parameter **i4b** = selected\_int\_kind(9)  
*4-byte integers*
- integer, parameter **sp** = kind(1.0)  
*single-precision reals*
- integer, parameter **dp** = kind(1.0d0)  
*double-precision reals*

### 5.4.1 Detailed Description

Declarations of kind types for MAIC-2.

### 5.4.2 Variable Documentation

#### 5.4.2.1 integer,parameter maic2\_types::dp = kind(1.0d0)

double-precision reals

Definition at line 40 of file maic2\_types.F90.

#### 5.4.2.2 integer,parameter maic2\_types::i2b = selected\_int\_kind(4)

2-byte integers

Definition at line 37 of file maic2\_types.F90.

#### 5.4.2.3 integer,parameter maic2\_types::i4b = selected\_int\_kind(9)

4-byte integers

Definition at line 38 of file maic2\_types.F90.

#### 5.4.2.4 integer,parameter maic2\_types::sp = kind(1.0)

single-precision reals

Definition at line 39 of file maic2\_types.F90.

## 5.5 maic2\_variables Module Reference

Declarations of global variables for MAIC-2.

### Variables

- real(dp), dimension(0:lmax) [phi\\_node](#)  
*phi\_node(l): Latitude of grid point*
- real(dp), dimension(0:lmax) [phi\\_cb1](#)  
*phi\_cb1(l): Latitude of lower cell boundary*
- real(dp), dimension(0:lmax) [phi\\_cb2](#)  
*phi\_cb2(l): Latitude of upper cell boundary*
- real(dp), dimension(0:lmax) [cos\\_phi\\_cb1](#)  
*cos\_phi\_cb1(l): Cosine of phi\_cb1(l)*
- real(dp), dimension(0:lmax) [cos\\_phi\\_cb2](#)  
*cos\_phi\_cb2(l): Cosine of phi\_cb2(l)*
- real(dp), dimension(0:lmax) [sin\\_phi\\_cb1](#)  
*sin\_phi\_cb1(l): Sine of phi\_cb1(l)*
- real(dp), dimension(0:lmax) [sin\\_phi\\_cb2](#)  
*sin\_phi\_cb2(l): Sine of phi\_cb2(l)*
- real(dp), dimension(lmax) [dphi](#)  
*dphi(l): Grid spacing*
- real(dp), dimension(lmax) [dphi\\_inv](#)  
*dphi\_inv(l): Inverse of the grid spacing*
- real(dp), dimension(0:lmax) [temp\\_surf](#)  
*temp\_surf(l): Daily mean surface temperature*
- real(dp), dimension(0:lmax) [temp\\_surf\\_amp](#)  
*temp\_surf\_amp(l): Amplitude of the daily cycle of the surface temperature*
- real(dp), dimension(0:lmax) [temp\\_co2](#)  
*temp\_co2(l): CO2 condensation temperature*
- real(dp), dimension(0:lmax) [p\\_surf](#)  
*p\_surf(l): Surface pressure*
- real(dp), dimension(0:lmax) [water](#)

*water(l): Water content in the atmosphere*

- real(dp), dimension(0:lmax) [water\\_new](#)  
*(.)\_new: New value of quantity (.) computed during an integration step*
- real(dp), dimension(0:lmax) [cond](#)  
*cond(l): Condensation rate*
- real(dp), dimension(0:lmax) [evap](#)  
*evap(l): Evaporation rate*
- real(dp), dimension(0:lmax) [a\\_net](#)  
*a\_net(l): Net surface mass balance of water ice*
- real(dp), dimension(0:lmax) [H](#)  
*H(l): Ice thickness.*
- real(dp), dimension(0:lmax) [H\\_new](#)  
*(.)\_new: New value of quantity (.) computed during an integration step*
- real(dp) [RHO](#)  
*RHO: Density of ice-dust mixture.*
- real(dp) [RHO\\_I](#)  
*RHO\_I: Density of ice.*
- real(dp) [RHO\\_W](#)  
*RHO\_W: Density of pure water.*
- real(dp) [G](#)  
*G: Acceleration due to gravity.*
- real(dp) [R](#)  
*R: Radius of Mars.*
- real(dp) [rho\\_inv](#)  
*rho\_inv: Inverse of the density of ice-dust mixture*
- integer(i4b) [insol\\_time\\_min](#)  
*insol\_time\_min: Minimum time of the data values for the insolation etc.*
- integer(i4b) [insol\\_time\\_stp](#)  
*insol\_time\_stp: Time step of the data values for the insolation etc.*
- integer(i4b) [insol\\_time\\_max](#)  
*insol\_time\_max: Maximum time of the data values for the insolation etc.*

- integer(i4b) **itercount**  
*itercount*: Counter for the number of time integration steps
- real(dp), dimension(0:100000) **insol\_ma\_90**  
*insol\_ma\_90(n)*: Data for the mean-annual north- or south-polar insolation
- real(dp), dimension(0:100000) **obl\_data**  
*obl\_data(n)*: Data for the obliquity
- real(dp), dimension(0:100000) **ecc\_data**  
*ecc\_data(n)*: Data for the eccentricity
- real(dp), dimension(0:100000) **ave\_data**  
*ave\_data(n)*: Data for the anomaly of vernal equinox (= 360 deg - Ls of perihelion )
- real(dp), dimension(0:100000) **cp\_data**  
*cp\_data(n)*: Data for Laskar's climate parameter = eccentricity \*sin(Laskar's longitude of perihelion from moving equinox), ( where Laskar's longitude of perihelion from moving equinox = Ls of perihelion - 180 deg )
- real(dp), dimension(0:ntime) **psi\_tab**  
*psi\_tab(n)*: True anomalies (orbital positions with respect to perihelion) over a Martian year
- real(dp), dimension(0:lmax) **diff\_aux**  
*diff\_aux(l)*: Auxiliary quantity needed for the diffusional transport
- integer(i4b) **n\_output**  
*n\_output*: Number of specified times for data output
- real(dp) **dtime\_out**  
*dtime\_out*: Time step for data output
- real(dp), dimension(100) **time\_output**  
*time\_output(n)*: Specified times for data output
- integer(i4b) **iter\_out**  
*iter\_out*: Intervall of integration steps for data output
- integer(i4b), dimension(100) **iter\_output**  
*iter\_output(n)*: Specified integration steps for data output
- real(dp), parameter **pi** = 3.141592653589793\_dp  
*pi*: Mathematical constant
- real(dp), parameter **pi\_inv** = 1.0\_dp/pi

*pi\_inv: Inverse of pi*

- real(dp), parameter `pi_180` = `pi/180.0_dp`  
*pi\_180: pi divided by 180 (-> deg to rad)*
- real(dp), parameter `pi_180_inv` = `180.0_dp/pi`  
*pi\_180\_inv: 180 divided by pi (-> rad to deg)*
- real(dp), parameter `eps` = `1.0e-05_dp`  
*eps: Small number*

### 5.5.1 Detailed Description

Declarations of global variables for MAIC-2.

### 5.5.2 Variable Documentation

#### 5.5.2.1 `real(dp),dimension(0:lmax) maic2_variables::a_net`

`a_net(l)`: Net surface mass balance of water ice

Definition at line 76 of file `maic2_variables.F90`.

#### 5.5.2.2 `real(dp),dimension(0:100000) maic2_variables::ave_data`

`ave_data(n)`: Data for the anomaly of vernal equinox (= 360 deg - Ls of perihelion )

Definition at line 116 of file `maic2_variables.F90`.

#### 5.5.2.3 `real(dp),dimension(0:lmax) maic2_variables::cond`

`cond(l)`: Condensation rate

Definition at line 72 of file `maic2_variables.F90`.

#### 5.5.2.4 `real(dp),dimension(0:lmax) maic2_variables::cos_phi_cb1`

`cos_phi_cb1(l)`: Cosine of `phi_cb1(l)`

Definition at line 48 of file `maic2_variables.F90`.

#### 5.5.2.5 `real(dp),dimension(0:lmax) maic2_variables::cos_phi_cb2`

`cos_phi_cb2(l)`: Cosine of `phi_cb2(l)`

Definition at line 50 of file `maic2_variables.F90`.

**5.5.2.6 real(dp),dimension(0:100000) maic2\_variables::cp\_data**

cp\_data(n): Data for Laskar's climate parameter = eccentricity \*sin(Laskar's longitude of perihelion from moving equinox), ( where Laskar's longitude of perihelion from moving equinox = Ls of perihelion - 180 deg )

Definition at line 122 of file maic2\_variables.F90.

**5.5.2.7 real(dp),dimension(0:lmax) maic2\_variables::diff\_aux**

diff\_aux(l): Auxiliary quantity needed for the diffusional transport

Definition at line 129 of file maic2\_variables.F90.

**5.5.2.8 real(dp),dimension(lmax) maic2\_variables::dphi**

dphi(l): Grid spacing

Definition at line 56 of file maic2\_variables.F90.

**5.5.2.9 real(dp),dimension(lmax) maic2\_variables::dphi\_inv**

dphi\_inv(l): Inverse of the grid spacing

Definition at line 58 of file maic2\_variables.F90.

**5.5.2.10 real(dp) maic2\_variables::dtime\_out**

dtime\_out: Time step for data output

Definition at line 134 of file maic2\_variables.F90.

**5.5.2.11 real(dp),dimension(0:100000) maic2\_variables::ecc\_data**

ecc\_data(n): Data for the eccentricity

Definition at line 113 of file maic2\_variables.F90.

**5.5.2.12 real(dp),parameter maic2\_variables::eps = 1.0e-05\_dp**

eps: Small number

Definition at line 151 of file maic2\_variables.F90.

**5.5.2.13 real(dp),dimension(0:lmax) maic2\_variables::evap**

evap(l): Evaporation rate

Definition at line 74 of file maic2\_variables.F90.

**5.5.2.14 real(dp) maic2\_variables::G**

G: Acceleration due to gravity.

Definition at line 91 of file maic2\_variables.F90.

**5.5.2.15 real(dp),dimension(0:lmax) maic2\_variables::H**

H(l): Ice thickness.

Definition at line 78 of file maic2\_variables.F90.

**5.5.2.16 real(dp),dimension(0:lmax) maic2\_variables::H\_new**

(.)\_new: New value of quantity (.) computed during an integration step

Definition at line 80 of file maic2\_variables.F90.

**5.5.2.17 real(dp),dimension(0:100000) maic2\_variables::insol\_ma\_90**

insol\_ma\_90(n): Data for the mean-annual north- or south-polar insolation

Definition at line 109 of file maic2\_variables.F90.

**5.5.2.18 integer(i4b) maic2\_variables::insol\_time\_max**

insol\_time\_max: Maximum time of the data values for the insolation etc.

Definition at line 104 of file maic2\_variables.F90.

**5.5.2.19 integer(i4b) maic2\_variables::insol\_time\_min**

insol\_time\_min: Minimum time of the data values for the insolation etc.

Definition at line 100 of file maic2\_variables.F90.

**5.5.2.20 integer(i4b) maic2\_variables::insol\_time\_stp**

insol\_time\_stp: Time step of the data values for the insolation etc.

Definition at line 102 of file maic2\_variables.F90.

**5.5.2.21 integer(i4b) maic2\_variables::iter\_out**

iter\_out: Intervall of integration steps for data output

Definition at line 138 of file maic2\_variables.F90.

**5.5.2.22 integer(i4b),dimension(100) maic2\_variables::iter\_output**

iter\_output(n): Specified integration steps for data output

Definition at line 140 of file maic2\_variables.F90.

**5.5.2.23 integer(i4b) maic2\_variables::itercount**

itercount: Counter for the number of time integration steps

Definition at line 106 of file maic2\_variables.F90.

**5.5.2.24 integer(i4b) maic2\_variables::n\_output**

n\_output: Number of specified times for data output

Definition at line 132 of file maic2\_variables.F90.

**5.5.2.25 real(dp),dimension(0:100000) maic2\_variables::obl\_data**

obl\_data(n): Data for the obliquity

Definition at line 111 of file maic2\_variables.F90.

**5.5.2.26 real(dp),dimension(0:lmax) maic2\_variables::p\_surf**

p\_surf(l): Surface pressure

Definition at line 66 of file maic2\_variables.F90.

**5.5.2.27 real(dp),dimension(0:lmax) maic2\_variables::phi\_cb1**

phi\_cb1(l): Latitude of lower cell boundary

Definition at line 44 of file maic2\_variables.F90.

**5.5.2.28 real(dp),dimension(0:lmax) maic2\_variables::phi\_cb2**

phi\_cb2(l): Latitude of upper cell boundary

Definition at line 46 of file maic2\_variables.F90.

**5.5.2.29 real(dp),dimension(0:lmax) maic2\_variables::phi\_node**

phi\_node(l): Latitude of grid point

Definition at line 42 of file maic2\_variables.F90.

**5.5.2.30 real(dp),parameter maic2\_variables::pi = 3.141592653589793\_dp**

pi: Mathematical constant

Definition at line 143 of file maic2\_variables.F90.

**5.5.2.31 real(dp),parameter maic2\_variables::pi\_180 = pi/180.0\_dp**

pi\_180: pi divided by 180 (-> deg to rad)

Definition at line 147 of file maic2\_variables.F90.

**5.5.2.32 real(dp),parameter maic2\_variables::pi\_180\_inv = 180.0\_dp/pi**

pi\_180\_inv: 180 divided by pi (-> rad to deg)

Definition at line 149 of file maic2\_variables.F90.

**5.5.2.33 real(dp),parameter maic2\_variables::pi\_inv = 1.0\_dp/pi**

pi\_inv: Inverse of pi

Definition at line 145 of file maic2\_variables.F90.

**5.5.2.34 real(dp),dimension(0:ntime) maic2\_variables::psi\_tab**

psi\_tab(n): True anomalies (orbital positions with respect to perihelion) over a Martian year

Definition at line 126 of file maic2\_variables.F90.

**5.5.2.35 real(dp) maic2\_variables::R**

R: Radius of Mars.

Definition at line 93 of file maic2\_variables.F90.

**5.5.2.36 real(dp) maic2\_variables::RHO**

RHO: Density of ice-dust mixture.

Definition at line 85 of file maic2\_variables.F90.

**5.5.2.37 real(dp) maic2\_variables::RHO\_I**

RHO\_I: Density of ice.

Definition at line 87 of file maic2\_variables.F90.

**5.5.2.38 real(dp) maic2\_variables::rho\_inv**

rho\_inv: Inverse of the density of ice-dust mixture

Definition at line 95 of file maic2\_variables.F90.

**5.5.2.39 real(dp) maic2\_variables::RHO\_W**

RHO\_W: Density of pure water.

Definition at line 89 of file maic2\_variables.F90.

**5.5.2.40 real(dp),dimension(0:lmax) maic2\_variables::sin\_phi\_cb1**

sin\_phi\_cb1(l): Sine of phi\_cb1(l)

Definition at line 52 of file maic2\_variables.F90.

**5.5.2.41 real(dp),dimension(0:lmax) maic2\_variables::sin\_phi\_cb2**

sin\_phi\_cb2(l): Sine of phi\_cb2(l)

Definition at line 54 of file maic2\_variables.F90.

**5.5.2.42 real(dp),dimension(0:lmax) maic2\_variables::temp\_co2**

temp\_co2(l): CO2 condensation temperature

Definition at line 64 of file maic2\_variables.F90.

**5.5.2.43 real(dp),dimension(0:lmax) maic2\_variables::temp\_surf**

temp\_surf(l): Daily mean surface temperature

Definition at line 60 of file maic2\_variables.F90.

**5.5.2.44 real(dp),dimension(0:lmax) maic2\_variables::temp\_surf\_amp**

temp\_surf\_amp(l): Amplitude of the daily cycle of the surface temperature

Definition at line 62 of file maic2\_variables.F90.

**5.5.2.45 real(dp),dimension(100) maic2\_variables::time\_output**

time\_output(n): Specified times for data output

Definition at line 136 of file maic2\_variables.F90.

**5.5.2.46    `real(dp),dimension(0:lmax) maic2_variables::water`**

`water(1)`: Water content in the atmosphere

Definition at line 68 of file `maic2_variables.F90`.

**5.5.2.47    `real(dp),dimension(0:lmax) maic2_variables::water_new`**

`(.)_new`: New value of quantity `(.)` computed during an integration step

Definition at line 70 of file `maic2_variables.F90`.

## Chapter 6

# Data Type Documentation

### 6.1 instemp::ins Type Reference

Surface temperatures.

#### Public Attributes

- `real(dp), dimension(0:360,-90:90) t`
- `real(dp), dimension(-90:90) tam`
- `real(dp), dimension(-90:90) tmax`

#### 6.1.1 Detailed Description

Surface temperatures.

Definition at line 46 of file instemp.f90.

#### 6.1.2 Member Data Documentation

##### 6.1.2.1 `real(dp),dimension(0:360,-90:90) instemp::ins::t`

Definition at line 47 of file instemp.f90.

##### 6.1.2.2 `real(dp),dimension(-90:90) instemp::ins::tam`

Definition at line 48 of file instemp.f90.

##### 6.1.2.3 `real(dp),dimension(-90:90) instemp::ins::tmax`

Definition at line 49 of file instemp.f90.

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

- subroutines/[instemp.f90](#)

# Chapter 7

## File Documentation

### 7.1 maic2.F90 File Reference

Main program file of MAIC-2.

```
#include "maic2_specs.h"  
#include "subroutines/maic2_types.F90"
```

#### Functions/Subroutines

- program [maic2](#)  
*Main program of MAIC-2.*

#### 7.1.1 Detailed Description

Main program file of MAIC-2.

#### 7.1.2 Copyright

Copyright 2010, 2011 Ralf Greve, Bjoern Grieger, Oliver J. Stenzel

#### 7.1.3 License

This file is part of MAIC-2.

MAIC-2 is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

MAIC-2 is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with MAIC-2. If not, see <<http://www.gnu.org/licenses/>>.

Definition in file [maic2.F90](#).

## 7.1.4 Function Documentation

### 7.1.4.1 program maic2 ()

Main program of MAIC-2.

Definition at line 104 of file [maic2.F90](#).

## 7.2 subroutines/boundary\_maic2.F90 File Reference

Determination of the surface temperature and of the net mass balance (accumulation-ablation rate) for the polar caps of Mars.

### Functions/Subroutines

- subroutine [boundary\\_maic2](#) (time, dtime)  
*Determination of the surface temperature and of the net mass balance (accumulation-ablation rate) for the polar caps of Mars.*

#### 7.2.1 Detailed Description

Determination of the surface temperature and of the net mass balance (accumulation-ablation rate) for the polar caps of Mars.

#### 7.2.2 Copyright

Copyright 2010, 2011 Ralf Greve, Bjoern Grieger, Oliver J. Stenzel

#### 7.2.3 License

This file is part of MAIC-2.

MAIC-2 is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

MAIC-2 is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with MAIC-2. If not, see <<http://www.gnu.org/licenses/>>.

Definition in file [boundary\\_maic2.F90](#).

#### 7.2.4 Function Documentation

##### 7.2.4.1 subroutine [boundary\\_maic2](#) (real(dp),intent(in) *time*, real(dp),intent(in) *dtime*)

Determination of the surface temperature and of the net mass balance (accumulation-ablation rate) for the polar caps of Mars.

Definition at line 37 of file [boundary\\_maic2.F90](#).

## 7.3 subroutines/calc\_top\_maic2.F90 File Reference

Computation of the ice-cap topography.

### Functions/Subroutines

- subroutine `calc_top_maic2` (time, dtype)  
*Computation of the ice-cap topography.*

#### 7.3.1 Detailed Description

Computation of the ice-cap topography.

#### 7.3.2 Copyright

Copyright 2010, 2011 Ralf Greve, Bjoern Grieger, Oliver J. Stenzel

#### 7.3.3 License

This file is part of MAIC-2.

MAIC-2 is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

MAIC-2 is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with MAIC-2. If not, see <<http://www.gnu.org/licenses/>>.

Definition in file `calc_top_maic2.F90`.

#### 7.3.4 Function Documentation

##### 7.3.4.1 subroutine `calc_top_maic2` (real(dp),intent(in) *time*, real(dp),intent(in) *dtype*)

Computation of the ice-cap topography.

Definition at line 35 of file `calc_top_maic2.F90`.

## 7.4 subroutines/condensation.f90 File Reference

Computation of the condensation rate.

### Modules

- module [condensation](#)  
*Computation of the condensation rate.*

### Functions/Subroutines

- subroutine [condensation::setcondpar](#) (gravity, timescale)  
*Setting of parameters.*
- subroutine [condensation::getcond\\_1](#) (temp, water, cond, dtime)  
*Computation of condensation (removal of water exceeding the saturation pressure at the surface).*
- subroutine [condensation::getcond\\_2](#) (temp, water, cond)  
*Computation of condensation (continuous, quadratic dependence on humidity).*

### Variables

- real(dp) [condensation::g](#)
- real(dp) [condensation::tau](#)

#### 7.4.1 Detailed Description

Computation of the condensation rate.

#### 7.4.2 Copyright

Copyright 2010, 2011 Ralf Greve, Bjoern Grieger, Oliver J. Stenzel

#### 7.4.3 License

This file is part of MAIC-2.

MAIC-2 is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

MAIC-2 is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with MAIC-2. If not, see <<http://www.gnu.org/licenses/>>.

Definition in file [condensation.f90](#).

## 7.5 subroutines/diff\_trans.F90 File Reference

Diffusive transport of water in the Martian atmosphere.

### Functions/Subroutines

- subroutine [diff\\_trans](#) (dtime)  
*Diffusive transport of water in the Martian atmosphere.*

#### 7.5.1 Detailed Description

Diffusive transport of water in the Martian atmosphere.

#### 7.5.2 Copyright

Copyright 2010, 2011 Ralf Greve, Bjoern Grieger, Oliver J. Stenzel

#### 7.5.3 License

This file is part of MAIC-2.

MAIC-2 is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

MAIC-2 is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with MAIC-2. If not, see <<http://www.gnu.org/licenses/>>.

Definition in file [diff\\_trans.F90](#).

#### 7.5.4 Function Documentation

##### 7.5.4.1 subroutine [diff\\_trans](#) (real(dp),intent(in) dtime)

Diffusive transport of water in the Martian atmosphere.

Definition at line 35 of file [diff\\_trans.F90](#).

## 7.6 subroutines/evaporation.f90 File Reference

Computation of the evaporation rate (buoyant-diffusion approach by Ingersoll).

### Modules

- module [evaporation](#)

*Computation of the evaporation rate (buoyant-diffusion approach by Ingersoll).*

### Functions/Subroutines

- subroutine [evaporation::setevappar\\_bd](#) (evap\_fact\_para, gamma\_reg\_para, R\_univ\_para, mol\_w\_para, mol\_c\_para, diff\_w\_c\_para, visc\_c\_para, g\_para)

*Setting of parameters.*

- subroutine [evaporation::getevap\\_bd](#) (temp, temp\_amp, p, H, evap)

*Computation of evaporation.*

### Variables

- real(dp) [evaporation::evap\\_fact](#)
- real(dp) [evaporation::gamma\\_reg](#)
- real(dp) [evaporation::R\\_univ](#)
- real(dp) [evaporation::mol\\_w](#)
- real(dp) [evaporation::mol\\_c](#)
- real(dp) [evaporation::diff\\_w\\_c](#)
- real(dp) [evaporation::visc\\_c](#)
- real(dp) [evaporation::g](#)

#### 7.6.1 Detailed Description

Computation of the evaporation rate (buoyant-diffusion approach by Ingersoll).

#### 7.6.2 Copyright

Copyright 2010, 2011 Ralf Greve, Bjoern Grieger, Oliver J. Stenzel

### 7.6.3 License

This file is part of MAIC-2.

MAIC-2 is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

MAIC-2 is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with MAIC-2. If not, see <<http://www.gnu.org/licenses/>>.

Definition in file [evaporation.f90](#).

## 7.7 subroutines/get\_orb\_par.F90 File Reference

Determination of the orbital parameters (eccentricity, obliquity, climate parameter CP, anomaly of vernal equinox, mean-annual north- or south-polar insolation).

### Functions/Subroutines

- subroutine `get_orb_par` (time, ecc, obl, cp, ave, insol\_ma\_90NS)  
*Determination of the orbital parameters (eccentricity, obliquity, climate parameter CP, anomaly of vernal equinox, mean-annual north- or south-polar insolation).*

#### 7.7.1 Detailed Description

Determination of the orbital parameters (eccentricity, obliquity, climate parameter CP, anomaly of vernal equinox, mean-annual north- or south-polar insolation).

#### 7.7.2 Copyright

Copyright 2010, 2011 Ralf Greve, Bjoern Grieger, Oliver J. Stenzel

#### 7.7.3 License

This file is part of MAIC-2.

MAIC-2 is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

MAIC-2 is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with MAIC-2. If not, see <<http://www.gnu.org/licenses/>>.

Definition in file `get_orb_par.F90`.

#### 7.7.4 Function Documentation

**7.7.4.1** subroutine `get_orb_par` (real(dp),intent(in) *time*, real(dp),intent(out) *ecc*, real(dp),intent(out) *obl*, real(dp),intent(out) *cp*, real(dp),intent(out) *ave*, real(dp),intent(out) *insol\_ma\_90NS*)

Determination of the orbital parameters (eccentricity, obliquity, climate parameter CP, anomaly of vernal equinox, mean-annual north- or south-polar insolation).

Definition at line 39 of file get\_orb\_par.F90.

## 7.8 subroutines/get\_psi\_tab.F90 File Reference

Computation of the table of true anomalies (orbital positions with respect to perihelion) over a Martian year.

### Functions/Subroutines

- subroutine `get_psi_tab` (ecc, ave)

*Computation of the table of true anomalies (orbital positions with respect to perihelion) over a Martian year.*

#### 7.8.1 Detailed Description

Computation of the table of true anomalies (orbital positions with respect to perihelion) over a Martian year.

#### 7.8.2 Copyright

Copyright 2010, 2011 Ralf Greve, Bjoern Grieger, Oliver J. Stenzel

#### 7.8.3 License

This file is part of MAIC-2.

MAIC-2 is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

MAIC-2 is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with MAIC-2. If not, see <<http://www.gnu.org/licenses/>>.

Definition in file `get_psi_tab.F90`.

#### 7.8.4 Function Documentation

##### 7.8.4.1 subroutine `get_psi_tab` (real(dp),intent(in) ecc, real(dp),intent(in) ave)

Computation of the table of true anomalies (orbital positions with respect to perihelion) over a Martian year.

Definition at line 37 of file `get_psi_tab.F90`.

## 7.9 subroutines/instemp.f90 File Reference

Computation of the daily mean surface temperature of Mars based on obliquity, eccentricity and the anomaly of vernal equinox (local insolation temperature = LIS scheme).

### Data Types

- type `instemp::ins`  
*Surface temperatures.*

### Modules

- module `instemp`  
*Computation of the daily mean surface temperature of Mars based on obliquity, eccentricity and the anomaly of vernal equinox (local insolation temperature = LIS scheme).*

### Functions/Subroutines

- subroutine `instemp::setinstemp` (o, ecc, ave, obl, sma, sa, sac, op, ct)  
*Main subroutine of module instemp.*
- real(dp) `instemp::instam` (o, phi)  
*Annual mean temperature at latitude phi.*
- real(dp) `instemp::instmax` (o, phi)  
*Annual maximum temperature at latitude phi.*
- real(dp) `instemp::inst` (o, psi, phi)  
*Temperature at orbit position psi and latitude phi.*
- real(dp) `instemp::inst1` (o, ipsi, phi)  
*Temperature at orbit position ipsi (integer value) and latitude phi.*

#### 7.9.1 Detailed Description

Computation of the daily mean surface temperature of Mars based on obliquity, eccentricity and the anomaly of vernal equinox (local insolation temperature = LIS scheme).

#### 7.9.2 Copyright

Copyright 2010, 2011 Ralf Greve, Bjoern Grieger, Oliver J. Stenzel

### 7.9.3 License

This file is part of MAIC-2.

MAIC-2 is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

MAIC-2 is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with MAIC-2. If not, see <<http://www.gnu.org/licenses/>>.

Definition in file [instemp.f90](#).

## 7.10 subroutines/maic2\_types.F90 File Reference

Declarations of kind types for MAIC-2.

### Modules

- module [maic2\\_types](#)  
*Declarations of kind types for MAIC-2.*

### Variables

- integer, parameter [maic2\\_types::i2b](#) = selected\_int\_kind(4)  
*2-byte integers*
- integer, parameter [maic2\\_types::i4b](#) = selected\_int\_kind(9)  
*4-byte integers*
- integer, parameter [maic2\\_types::sp](#) = kind(1.0)  
*single-precision reals*
- integer, parameter [maic2\\_types::dp](#) = kind(1.0d0)  
*double-precision reals*

### 7.10.1 Detailed Description

Declarations of kind types for MAIC-2.

### 7.10.2 Copyright

Copyright 2010, 2011 Ralf Greve, Bjoern Grieger, Oliver J. Stenzel

### 7.10.3 License

This file is part of MAIC-2.

MAIC-2 is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

MAIC-2 is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with MAIC-2. If not, see <<http://www.gnu.org/licenses/>>.

Definition in file [maic2\\_types.F90](#).

## 7.11 subroutines/maic2\_variables.F90 File Reference

Declarations of global variables for MAIC-2.

### Modules

- module [maic2\\_variables](#)  
*Declarations of global variables for MAIC-2.*

### Variables

- real(dp), dimension(0:lmax) [maic2\\_variables::phi\\_node](#)  
*phi\_node(l): Latitude of grid point*
- real(dp), dimension(0:lmax) [maic2\\_variables::phi\\_cb1](#)  
*phi\_cb1(l): Latitude of lower cell boundary*
- real(dp), dimension(0:lmax) [maic2\\_variables::phi\\_cb2](#)  
*phi\_cb2(l): Latitude of upper cell boundary*
- real(dp), dimension(0:lmax) [maic2\\_variables::cos\\_phi\\_cb1](#)  
*cos\_phi\_cb1(l): Cosine of phi\_cb1(l)*
- real(dp), dimension(0:lmax) [maic2\\_variables::cos\\_phi\\_cb2](#)  
*cos\_phi\_cb2(l): Cosine of phi\_cb2(l)*
- real(dp), dimension(0:lmax) [maic2\\_variables::sin\\_phi\\_cb1](#)  
*sin\_phi\_cb1(l): Sine of phi\_cb1(l)*
- real(dp), dimension(0:lmax) [maic2\\_variables::sin\\_phi\\_cb2](#)  
*sin\_phi\_cb2(l): Sine of phi\_cb2(l)*
- real(dp), dimension(lmax) [maic2\\_variables::dphi](#)  
*dphi(l): Grid spacing*
- real(dp), dimension(lmax) [maic2\\_variables::dphi\\_inv](#)  
*dphi\_inv(l): Inverse of the grid spacing*
- real(dp), dimension(0:lmax) [maic2\\_variables::temp\\_surf](#)  
*temp\_surf(l): Daily mean surface temperature*
- real(dp), dimension(0:lmax) [maic2\\_variables::temp\\_surf\\_amp](#)  
*temp\_surf\_amp(l): Amplitude of the daily cycle of the surface temperature*

- `real(dp), dimension(0:lmax) maic2_variables::temp_co2`  
*temp\_co2(l): CO2 condensation temperature*
- `real(dp), dimension(0:lmax) maic2_variables::p_surf`  
*p\_surf(l): Surface pressure*
- `real(dp), dimension(0:lmax) maic2_variables::water`  
*water(l): Water content in the atmosphere*
- `real(dp), dimension(0:lmax) maic2_variables::water_new`  
*(.)\_new: New value of quantity (.) computed during an integration step*
- `real(dp), dimension(0:lmax) maic2_variables::cond`  
*cond(l): Condensation rate*
- `real(dp), dimension(0:lmax) maic2_variables::evap`  
*evap(l): Evaporation rate*
- `real(dp), dimension(0:lmax) maic2_variables::a_net`  
*a\_net(l): Net surface mass balance of water ice*
- `real(dp), dimension(0:lmax) maic2_variables::H`  
*H(l): Ice thickness.*
- `real(dp), dimension(0:lmax) maic2_variables::H_new`  
*(.)\_new: New value of quantity (.) computed during an integration step*
- `real(dp) maic2_variables::RHO`  
*RHO: Density of ice-dust mixture.*
- `real(dp) maic2_variables::RHO_I`  
*RHO\_I: Density of ice.*
- `real(dp) maic2_variables::RHO_W`  
*RHO\_W: Density of pure water.*
- `real(dp) maic2_variables::G`  
*G: Acceleration due to gravity.*
- `real(dp) maic2_variables::R`  
*R: Radius of Mars.*
- `real(dp) maic2_variables::rho_inv`  
*rho\_inv: Inverse of the density of ice-dust mixture*
- `integer(i4b) maic2_variables::insol_time_min`

*insol\_time\_min*: Minimum time of the data values for the insolation etc.

- integer(i4b) `maic2_variables::insol_time_stp`  
*insol\_time\_stp*: Time step of the data values for the insolation etc.
- integer(i4b) `maic2_variables::insol_time_max`  
*insol\_time\_max*: Maximum time of the data values for the insolation etc.
- integer(i4b) `maic2_variables::itercount`  
*itercount*: Counter for the number of time integration steps
- real(dp), dimension(0:100000) `maic2_variables::insol_ma_90`  
*insol\_ma\_90(n)*: Data for the mean-annual north- or south-polar insolation
- real(dp), dimension(0:100000) `maic2_variables::obl_data`  
*obl\_data(n)*: Data for the obliquity
- real(dp), dimension(0:100000) `maic2_variables::ecc_data`  
*ecc\_data(n)*: Data for the eccentricity
- real(dp), dimension(0:100000) `maic2_variables::ave_data`  
*ave\_data(n)*: Data for the anomaly of vernal equinox (= 360 deg - Ls of perihelion )
- real(dp), dimension(0:100000) `maic2_variables::cp_data`  
*cp\_data(n)*: Data for Laskar's climate parameter = eccentricity \*sin(Laskar's longitude of perihelion from moving equinox), ( where Laskar's longitude of perihelion from moving equinox = Ls of perihelion - 180 deg )
- real(dp), dimension(0:ntime) `maic2_variables::psi_tab`  
*psi\_tab(n)*: True anomalies (orbital positions with respect to perihelion) over a Martian year
- real(dp), dimension(0:lmax) `maic2_variables::diff_aux`  
*diff\_aux(l)*: Auxiliary quantity needed for the diffusional transport
- integer(i4b) `maic2_variables::n_output`  
*n\_output*: Number of specified times for data output
- real(dp) `maic2_variables::dtime_out`  
*dtime\_out*: Time step for data output
- real(dp), dimension(100) `maic2_variables::time_output`  
*time\_output(n)*: Specified times for data output
- integer(i4b) `maic2_variables::iter_out`  
*iter\_out*: Intervall of integration steps for data output

- integer(i4b), dimension(100) `maic2_variables::iter_output`  
*iter\_output(n): Specified integration steps for data output*
- real(dp), parameter `maic2_variables::pi` = 3.141592653589793\_dp  
*pi: Mathematical constant*
- real(dp), parameter `maic2_variables::pi_inv` = 1.0\_dp/pi  
*pi\_inv: Inverse of pi*
- real(dp), parameter `maic2_variables::pi_180` = pi/180.0\_dp  
*pi\_180: pi divided by 180 (-> deg to rad)*
- real(dp), parameter `maic2_variables::pi_180_inv` = 180.0\_dp/pi  
*pi\_180\_inv: 180 divided by pi (-> rad to deg)*
- real(dp), parameter `maic2_variables::eps` = 1.0e-05\_dp  
*eps: Small number*

### 7.11.1 Detailed Description

Declarations of global variables for MAIC-2.

### 7.11.2 Copyright

Copyright 2010, 2011 Ralf Greve, Bjoern Grieger, Oliver J. Stenzel

### 7.11.3 License

This file is part of MAIC-2.

MAIC-2 is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

MAIC-2 is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with MAIC-2. If not, see <<http://www.gnu.org/licenses/>>.

Definition in file `maic2_variables.F90`.

## 7.12 subroutines/output.F90 File Reference

Data output.

### Functions/Subroutines

- subroutine [output](#) (time)

*Data output.*

### 7.12.1 Detailed Description

Data output.

### 7.12.2 Copyright

Copyright 2010, 2011 Ralf Greve, Bjoern Grieger, Oliver J. Stenzel

### 7.12.3 License

This file is part of MAIC-2.

MAIC-2 is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

MAIC-2 is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with MAIC-2. If not, see <<http://www.gnu.org/licenses/>>.

Definition in file [output.F90](#).

### 7.12.4 Function Documentation

#### 7.12.4.1 subroutine [output](#) (real(dp),intent(in) *time*)

Data output.

Definition at line 35 of file output.F90.

## 7.13 subroutines/p\_sat.f90 File Reference

Computation of the water-vapour saturation pressure.

### Functions/Subroutines

- real(dp) `p_sat` (temp\_surf)

*Computation of the water-vapour saturation pressure.*

#### 7.13.1 Detailed Description

Computation of the water-vapour saturation pressure.

#### 7.13.2 Copyright

Copyright 2010, 2011 Ralf Greve, Bjoern Grieger, Oliver J. Stenzel

#### 7.13.3 License

This file is part of MAIC-2.

MAIC-2 is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

MAIC-2 is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with MAIC-2. If not, see <<http://www.gnu.org/licenses/>>.

Definition in file `p_sat.f90`.

#### 7.13.4 Function Documentation

##### 7.13.4.1 real(dp) `p_sat` (real(dp),intent(in) temp\_surf)

Computation of the water-vapour saturation pressure.

Definition at line 35 of file `p_sat.f90`.

## 7.14 subroutines/tri\_sle.F90 File Reference

Solution of a system of linear equations  $Ax=b$  with tridiagonal matrix  $A$ .

### Functions/Subroutines

- subroutine [tri\\_sle](#) ( $a0$ ,  $a1$ ,  $a2$ ,  $x$ ,  $b$ ,  $n\_rows$ )

*Solution of a system of linear equations  $Ax=b$  with tridiagonal matrix  $A$ .*

### 7.14.1 Detailed Description

Solution of a system of linear equations  $Ax=b$  with tridiagonal matrix  $A$ .

### 7.14.2 Copyright

Copyright 2010, 2011 Ralf Greve, Bjoern Grieger, Oliver J. Stenzel

### 7.14.3 License

This file is part of MAIC-2.

MAIC-2 is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

MAIC-2 is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with MAIC-2. If not, see <<http://www.gnu.org/licenses/>>.

Definition in file [tri\\_sle.F90](#).

### 7.14.4 Function Documentation

**7.14.4.1** subroutine `tri_sle` (`real(dp),dimension(0:*)`,`intent(inout)`  
`a0`, `real(dp),dimension(0:*)`,`intent(inout)`  
`a1`, `real(dp),dimension(0:*)`,`intent(inout)`  
`a2`, `real(dp),dimension(0:*)`,`intent(out)` `x`,  
`real(dp),dimension(0:*)`,`intent(inout)` `b`, `integer(i4b)`,`intent(in)` `n_rows`)

Solution of a system of linear equations  $Ax=b$  with tridiagonal matrix  $A$ .

**Parameters**

- ← ***a0***  $a_0(j)$  is element  $A_{(j,j-1)}$  of matrix  $A$
- ← ***a1***  $a_1(j)$  is element  $A_{(j,j)}$  of matrix  $A$
- ← ***a2***  $a_2(j)$  is element  $A_{(j,j+1)}$  of matrix  $A$
- ← ***b*** inhomogeneity vector
- ← ***n\_rows*** size of matrix  $A$  (indices run from 0 (!!!) to  $n\_rows$ )
- ***x*** solution vector.

Definition at line 41 of file tri\_sle.F90.

# Index

a\_net  
    maic2\_variables, 20

ave\_data  
    maic2\_variables, 20

boundary\_maic2  
    boundary\_maic2.F90, 31

boundary\_maic2.F90  
    boundary\_maic2, 31

calc\_top\_maic2  
    calc\_top\_maic2.F90, 32

calc\_top\_maic2.F90  
    calc\_top\_maic2, 32

cond  
    maic2\_variables, 20

condensation, 9  
    g, 10  
    getcond\_1, 10  
    getcond\_2, 10  
    setcondpar, 10  
    tau, 10

cos\_phi\_cb1  
    maic2\_variables, 20

cos\_phi\_cb2  
    maic2\_variables, 20

cp\_data  
    maic2\_variables, 20

diff\_aux  
    maic2\_variables, 21

diff\_trans  
    diff\_trans.F90, 35

diff\_trans.F90  
    diff\_trans, 35

diff\_w\_c  
    evaporation, 12

dp  
    maic2\_types, 15

dphi  
    maic2\_variables, 21

dphi\_inv  
    maic2\_variables, 21

dtime\_out  
    maic2\_variables, 21

ecc\_data  
    maic2\_variables, 21

eps  
    maic2\_variables, 21

evap  
    maic2\_variables, 21

evap\_fact  
    evaporation, 12

evaporation, 11  
    diff\_w\_c, 12  
    evap\_fact, 12  
    g, 12  
    gamma\_reg, 12  
    getevap\_bd, 11  
    mol\_c, 12  
    mol\_w, 12  
    R\_univ, 12  
    setevappar\_bd, 11  
    visc\_c, 12

G  
    maic2\_variables, 21

g  
    condensation, 10  
    evaporation, 12

gamma\_reg  
    evaporation, 12

get\_orb\_par  
    get\_orb\_par.F90, 38

get\_orb\_par.F90  
    get\_orb\_par, 38

get\_psi\_tab  
    get\_psi\_tab.F90, 40

get\_psi\_tab.F90  
    get\_psi\_tab, 40

getcond\_1

- condensation, [10](#)
- getcond\_2
  - condensation, [10](#)
- getevap\_bd
  - evaporation, [11](#)
- H
  - maic2\_variables, [22](#)
- H\_new
  - maic2\_variables, [22](#)
- i2b
  - maic2\_types, [15](#)
- i4b
  - maic2\_types, [15](#)
- insol\_ma\_90
  - maic2\_variables, [22](#)
- insol\_time\_max
  - maic2\_variables, [22](#)
- insol\_time\_min
  - maic2\_variables, [22](#)
- insol\_time\_stp
  - maic2\_variables, [22](#)
- inst
  - instemp, [13](#)
- inst1
  - instemp, [13](#)
- instam
  - instemp, [14](#)
- instemp, [13](#)
  - inst, [13](#)
  - inst1, [13](#)
  - instam, [14](#)
  - instmax, [14](#)
  - setinstemp, [14](#)
- instemp::ins, [27](#)
  - t, [27](#)
  - tam, [27](#)
  - tmax, [27](#)
- instmax
  - instemp, [14](#)
- iter\_out
  - maic2\_variables, [22](#)
- iter\_output
  - maic2\_variables, [22](#)
- itercount
  - maic2\_variables, [23](#)
- maic2
  - maic2.F90, [30](#)
- maic2.F90, [29](#)
  - maic2, [30](#)
- maic2\_types, [15](#)
  - dp, [15](#)
  - i2b, [15](#)
  - i4b, [15](#)
  - sp, [15](#)
- maic2\_variables, [17](#)
  - a\_net, [20](#)
  - ave\_data, [20](#)
  - cond, [20](#)
  - cos\_phi\_cb1, [20](#)
  - cos\_phi\_cb2, [20](#)
  - cp\_data, [20](#)
  - diff\_aux, [21](#)
  - dphi, [21](#)
  - dphi\_inv, [21](#)
  - dtime\_out, [21](#)
  - ecc\_data, [21](#)
  - eps, [21](#)
  - evap, [21](#)
  - G, [21](#)
  - H, [22](#)
  - H\_new, [22](#)
  - insol\_ma\_90, [22](#)
  - insol\_time\_max, [22](#)
  - insol\_time\_min, [22](#)
  - insol\_time\_stp, [22](#)
  - iter\_out, [22](#)
  - iter\_output, [22](#)
  - itercount, [23](#)
  - n\_output, [23](#)
  - obl\_data, [23](#)
  - p\_surf, [23](#)
  - phi\_cb1, [23](#)
  - phi\_cb2, [23](#)
  - phi\_node, [23](#)
  - pi, [23](#)
  - pi\_180, [24](#)
  - pi\_180\_inv, [24](#)
  - pi\_inv, [24](#)
  - psi\_tab, [24](#)
  - R, [24](#)
  - RHO, [24](#)
  - RHO\_I, [24](#)
  - rho\_inv, [24](#)
  - RHO\_W, [25](#)
  - sin\_phi\_cb1, [25](#)
  - sin\_phi\_cb2, [25](#)
  - temp\_co2, [25](#)

- temp\_surf, 25
- temp\_surf\_amp, 25
- time\_output, 25
- water, 25
- water\_new, 26
- mol\_c
  - evaporation, 12
- mol\_w
  - evaporation, 12
- n\_output
  - maic2\_variables, 23
- obl\_data
  - maic2\_variables, 23
- output
  - output.F90, 49
- output.F90
  - output, 49
- p\_sat
  - p\_sat.f90, 50
- p\_sat.f90
  - p\_sat, 50
- p\_surf
  - maic2\_variables, 23
- phi\_cb1
  - maic2\_variables, 23
- phi\_cb2
  - maic2\_variables, 23
- phi\_node
  - maic2\_variables, 23
- pi
  - maic2\_variables, 23
- pi\_180
  - maic2\_variables, 24
- pi\_180\_inv
  - maic2\_variables, 24
- pi\_inv
  - maic2\_variables, 24
- psi\_tab
  - maic2\_variables, 24
- R
  - maic2\_variables, 24
- R\_univ
  - evaporation, 12
- RHO
  - maic2\_variables, 24
- RHO\_I
  - maic2\_variables, 24
- rho\_inv
  - maic2\_variables, 24
- RHO\_W
  - maic2\_variables, 25
- setcondpar
  - condensation, 10
- setevappar\_bd
  - evaporation, 11
- setinstemp
  - instemp, 14
- sin\_phi\_cb1
  - maic2\_variables, 25
- sin\_phi\_cb2
  - maic2\_variables, 25
- sp
  - maic2\_types, 15
- subroutines/boundary\_maic2.F90, 31
- subroutines/calc\_top\_maic2.F90, 32
- subroutines/condensation.f90, 33
- subroutines/diff\_trans.F90, 35
- subroutines/evaporation.f90, 36
- subroutines/get\_orb\_par.F90, 38
- subroutines/get\_psi\_tab.F90, 40
- subroutines/instemp.f90, 41
- subroutines/maic2\_types.F90, 43
- subroutines/maic2\_variables.F90, 45
- subroutines/output.F90, 49
- subroutines/p\_sat.f90, 50
- subroutines/tri\_sle.F90, 51
- t
  - instemp::ins, 27
- tam
  - instemp::ins, 27
- tau
  - condensation, 10
- temp\_co2
  - maic2\_variables, 25
- temp\_surf
  - maic2\_variables, 25
- temp\_surf\_amp
  - maic2\_variables, 25
- time\_output
  - maic2\_variables, 25
- tmax
  - instemp::ins, 27
- tri\_sle
  - tri\_sle.F90, 51

tri\_sle.F90  
    tri\_sle, [51](#)

visc\_c  
    evaporation, [12](#)

water  
    maic2\_variables, [25](#)

water\_new  
    maic2\_variables, [26](#)