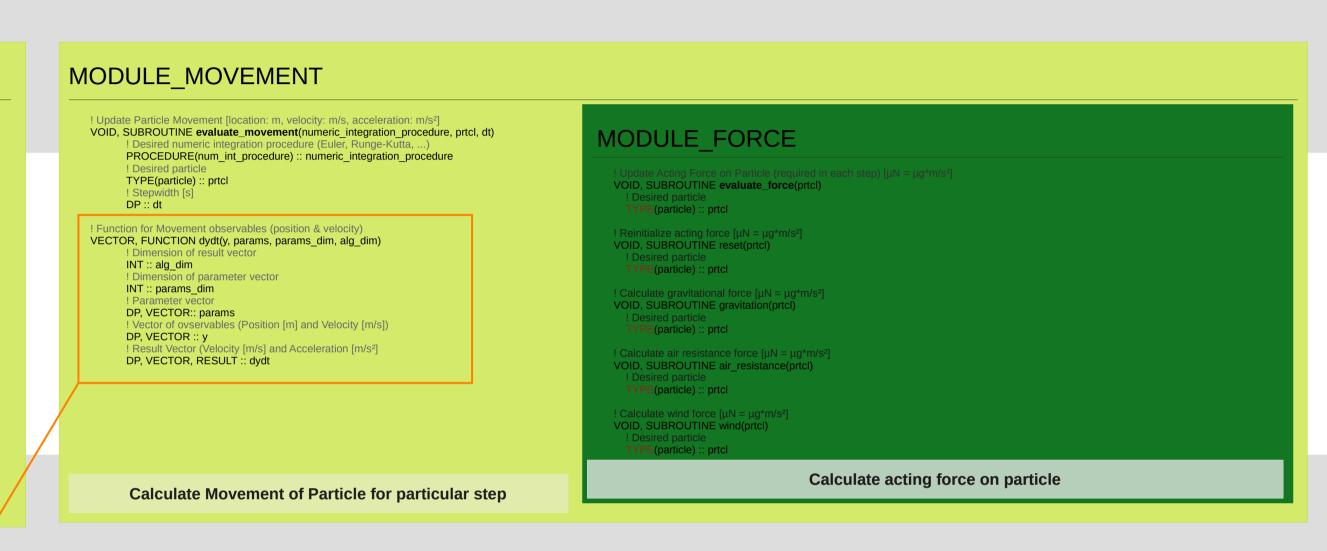
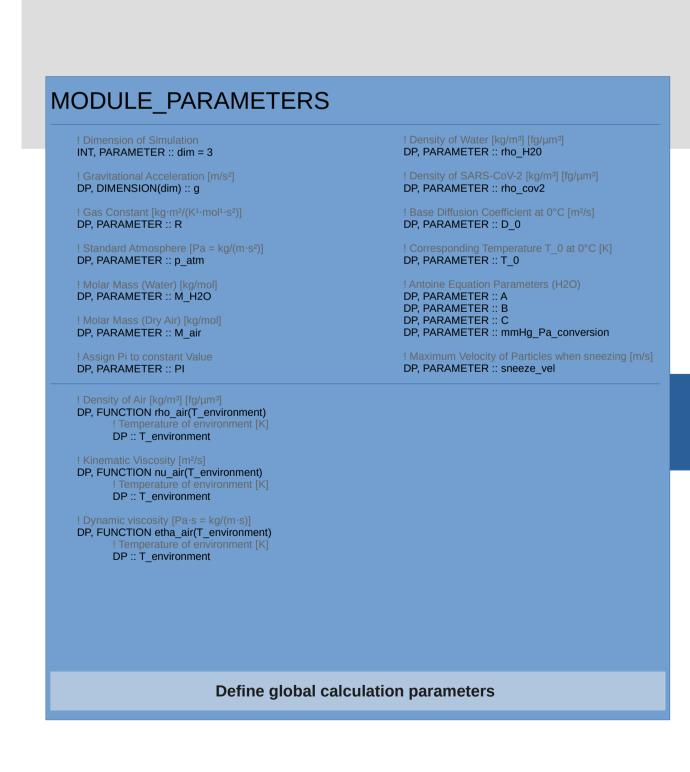
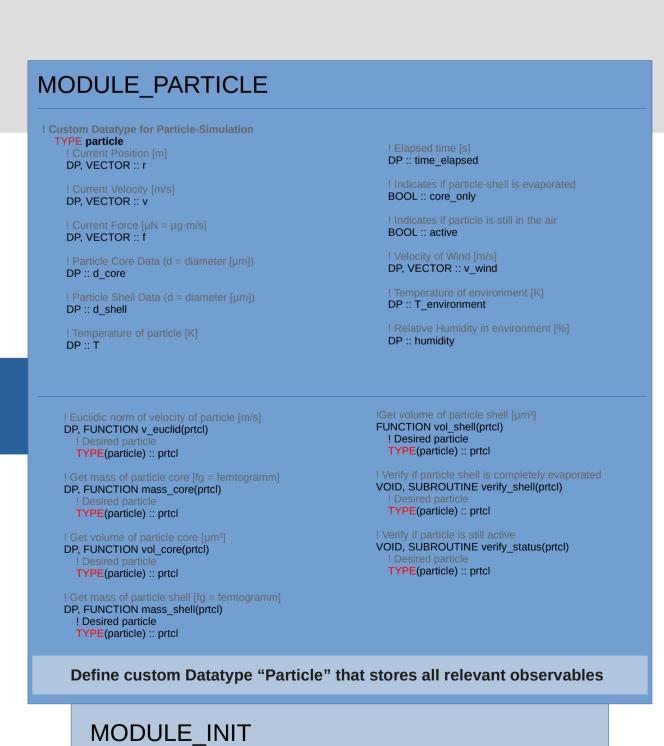
PARTICLE_SIMULATION List of the Control of Simulation Void DISCOME investigance control of Entrol of Control of Simulation Mill control of Simulation Mill control of Simulation Void DISCOME investigance control of Simulation Proceeding the Control of Simulation Void DISCOME investigance control of Simulation Void DISCOME investigance control of Simulation Void DISCOME investigance control of Simulation Mill control of Simulation Mill control of Simulation Manage Execution of Simulation

MODULE_EVAPORATION Last Order Derivative of particle diameter [um/s] Calculate Sherwood number [dimensionless] DP, FUNCTION Sh(velocity, diameter, T_environment, T_particle) VECTOR, FUNCTION dddt(y, params, params_dim, alg_dim) VOID, SUBROUTINE **evaluate_evaporation**(integration_procedure, prtcl, dt) DP :: T_environment, T_particle INT :: alg_dim PROCEDURE(num_int_procedure) :: integration_procedure INT :: params_dim TYPE(particle) :: prtcl ! Diameter of particle [µm] DP, VECTOR:: params DP, VECTOR :: y ! Diffusion Coefficient [$m^2/s = \mu m^2/s * 10^{12}$] ! Calculate Reynolds number [dimensionless] DP, FUNCTION D_Coeff(T_particle, T_environment) DP, FUNCTION Re(velocity, diameter, T_environment) DP, VECTOR :: dddt DP :: T_particle, T_environment DP :: velocity ! Calculate partial pressure of H2O at certain temperature [Pa = $kg \cdot m^{-1} \cdot s^{-2}$] DP :: diameter DP, FUNCTION p0_H2O(T) Temperature of environment and particle [K] DP :: T_environment Calculate Schmidt number [dimensionless] ! Calculate partial pressure of H2O in particle (fluid water) [Pa = $kg \cdot m^{-1} \cdot s^{-2}$] DP, FUNCTION Sc(T_environment, T_particle) DP, FUNCTION pw_H2O(T_particle) DP :: T_environment, T_particle DP :: T_particle ! Calculate mass transfer coefficient [µm/s] ! Calculate partial pressure of H2O in environment (steam) [Pa = $kg \cdot m^{-1} \cdot s^{-2}$] DP, FUNCTION h_m(diameter, T_environment, T_particle, velocity) DP, FUNCTION pinf_H2O(T_environment, humidity) DP :: T_environment, T_particle DP :: velocity DP :: diameter Calculate Evaporation of Particle-Shell for particular step







! Initialize circumstances of particle

! Temperature of environment [K]

! Relative Humidity in environment [%]

TYPE(particle) :: prtcl

! Velocity of Wind [m/s]

DP :: T environment

DP :: humidity

VOID, SUBROUTINE initialize_circumstances (prtcl, T_environment, humidity, v_wind)

! Initialize position of particles [m]

! Initialize velocity of particles [m/s]

! Initialize structure of particle

TYPE(particle) :: prtcl

! Velocity intensity coefficient [0...1]

VOID, SUBROUTINE initialize_structure(prtcl)

TYPE(particle) :: prtcl

! Desired particle TYPE(particle) :: prtcl

DP :: intensity

VOID, SUBROUTINE initialize_position(prtcl)

VOID, SUBROUTINE initialize_velocity(prtcl, intensity) DP, VECTOR :: v_wind

Semi-Automatic Initialization of Particle

