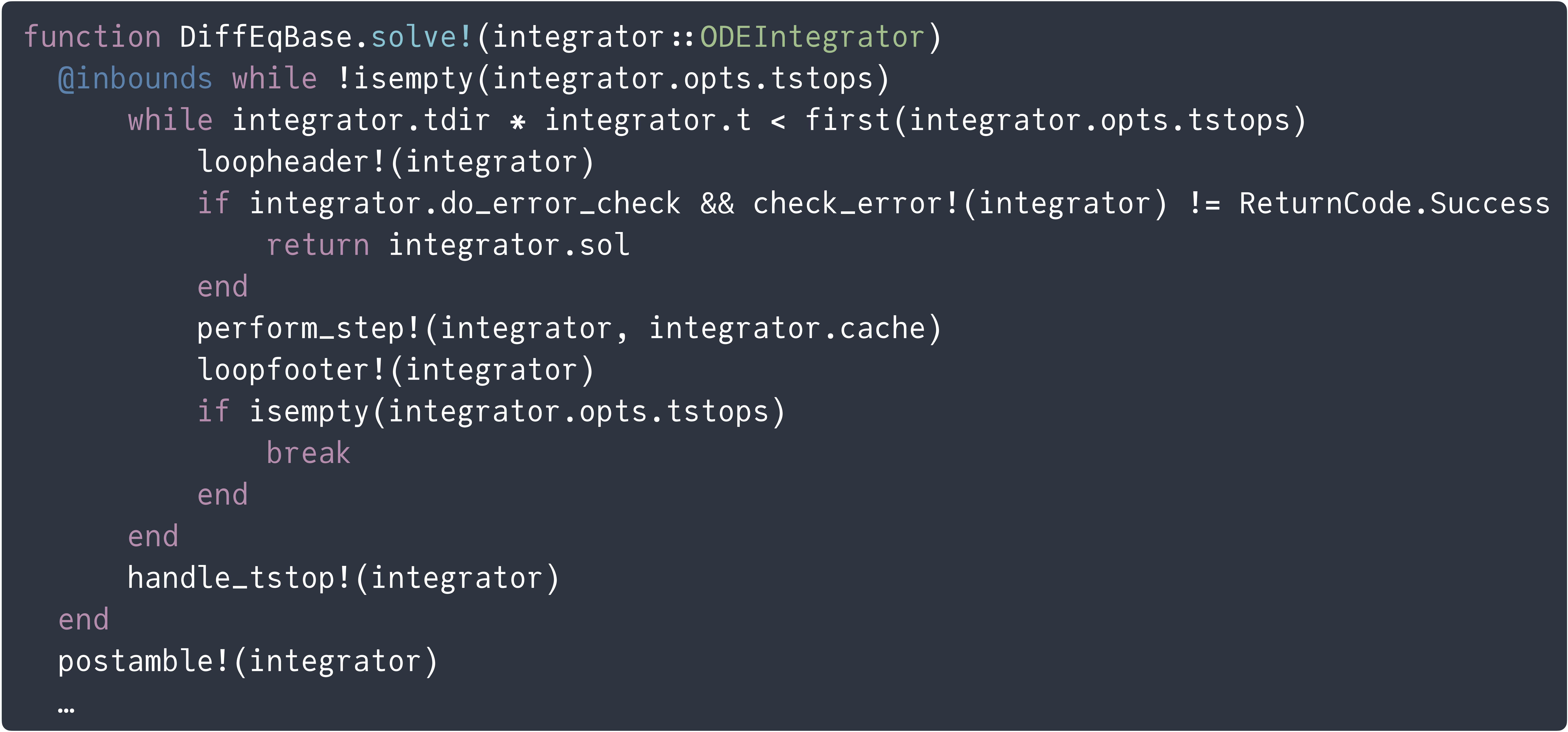

```
function DiffEqBase.solve!(integrator::ODEIntegrator)
  @inbounds while !isempty(integrator.opts.tstops)
      while integrator.tdir * integrator.t < first(integrator.opts.tstops)
          loopheader!(integrator)
          if integrator.do_error_check && check_error!(integrator) != ReturnCode.Success
              return integrator.sol
          end
          perform_step!(integrator, integrator.cache)
          loopfooter!(integrator)
          if isempty(integrator.opts.tstops)
              break
          end
      end
      handle_tstop!(integrator)
  end
  postamble!(integrator)
  …
```

```
OrdinaryDiffEq/src/solve.jl:514
```

OrdinaryDiffEq

74

NaN

https://github.com/SciML/OrdinaryDiffEq.jl/issues/1939

OrdinaryDiffEq

```julia
function DiffEqBase.solve!(integrator::ODEIntegrator)
  @inbounds while !isempty(integrator.opts.tstops)
      while integrator.tdir * integrator.t < first(integrator.opts.tstops)
          loopheader!(integrator)
          if integrator.do_error_check && check_error!(integrator) != ReturnCode.Success
              return integrator.sol
          end
          perform_step!(integrator, integrator.cache)
          loopfooter!(integrator)
          if isempty(integrator.opts.tstops)
              break
          end
      end
      handle_tstop!(integrator)
  end
  postamble!(integrator)
  …
```

https://github.com/SciML/OrdinaryDiffEq.jl/issues/1939

OrdinaryDiffEq/src/solve.jl:514

FlowFPX Internals