

Uovie Library

in development

Interface

Here list some class interfaces in Piano:

```
using namespace uovie::thermostat::ld;
ld_side(const std::string& fn_no_ex, const Global::basic_simu_para& bsp,
        const Global::system& sys, const double gamma);
ld_middle(const std::string& _fn_no_ex, const Global::basic_simu_para& _bsp,
          const Global::system& sys, const double gamma);
```

```
using namespace uovie::thermostat::at;
at_side(const std::string& fn_no_ex, const Global::basic_simu_para& bsp,
        const Global::system& sys, const double nu);
at_middle(const std::string& fn_no_ex, const Global::basic_simu_para& bsp,
          const Global::system& sys, const double nu);
```

```
using namespace uovie::thermostat::nhc;
nhc_global_side(const std::string& fn_no_ex, const Global::basic_simu_para& bsp,
                const Global::system& sys, const thermo_factor_scheme& tfs, const int
nchain);
nhc_global_middle(const std::string& fn_no_ex, const Global::basic_simu_para&
bsp,
                 const Global::system& sys, const thermo_factor_scheme& tfs, const int
nchain);
nhc_local_side(const std::string& fn_no_ex, const Global::basic_simu_para& bsp,
               const Global::system& sys, const thermo_factor_scheme& tfs, const int
nchain);
nhc_local_middle(const std::string& fn_no_ex, const Global::basic_simu_para&
bsp,
                 const Global::system& sys, const thermo_factor_scheme& tfs, const int
nchain);
```

```
using namespace uovie::pimd;
using namespace uovie::thermostat::nhc;
pimd_via_ld_side(const std::string& fn_no_ex, const Global::basic_simu_para&
bsp,
                 const Global::system& sys, const int nbead, const double gamma);
pimd_via_ld_middle(const std::string& fn_no_ex, const Global::basic_simu_para&
bsp,
                  const Global::system& sys, const int nbead, const double gamma);
pimd_via_at_side(const std::string& fn_no_ex, const Global::basic_simu_para&
bsp,
                 const Global::system& sys, const int nbead, const double nu);
pimd_via_at_middle(const std::string& fn_no_ex, const Global::basic_simu_para&
bsp,
                   const Global::system& sys, const int nbead, const double nu);
```

```
pimd_via_nhc_side(const std::string& fn_no_ex, const Global::basic_simu_para&
bsp,
    const Global::system& sys, const int nbead, const thermo_factor_scheme& tfs,
    const int nchain);
pimd_via_nhc_middle(const std::string& fn_no_ex, const Global::basic_simu_para&
bsp,
    const Global::system& sys, const int nbead, const thermo_factor_scheme& tfs,
    const int nchain);
```

All of them contain a member function `implement()`. To carry out certain kind of simulations, you just have to create a corresponding object and then invoke member function `implement()`.