Running EPS in DEODE

EPS Configuration by design

- Every DEODE run is now an ensemble - the deterministic case is just a special case with only one member
- Ensemble of configurations, where each member "perturb" the default configuration.
- One can e.g. think of running an ensemble of
  - a set of different model configurations (e.g. different physics options, different initial conditions, different boundary conditions, etc.)
  - a set of different CSCs
  - a set of different domains
  - a set of different domain resolutions and extends
  - a set of different time settings
  - etc.

Include EPS file to run ensemble

```
--config-file
deode/data/config_files/config.toml
deode/data/config_files/include/domains/500m_Helsinki_20241126.toml
deode/data/config_files/include/eps/eps_members.toml
deode/data/config_files/modifications/harmonie_arome.toml
deode/data/config_files/modifications/CY46h1.toml
deode/data/config_files/modifications/submission/@HOST@_CY46h1.toml
deode/data/config_files/modifications/submission/@HOST@_CY46h1_larg
```

`poetry run deode case ?/path/to/configuration --start-suite`

Example eps config files

In `./deode/data/config_files/include/eps/` there are a bunch of example eps toml files to try out:

```
deode/data/config_files/include/eps/
├── eps_3members_IFSENS_common_mars_prep_and_spp.toml
├── eps_3members_IFSENS_common_mars_prep.toml
├──
eps_3members_IFSENS_member_specific_mars_prep_and_spp.toml
├── eps_3members_IFSENS_member_specific_static_data.toml
├── eps_3members_ilv7.toml
```

```
├── eps_demo_cases1.toml
└── eps_demo_cases2.toml
```

E.g. try

```
poetry run deode case --config-file
./deode/data/config_files/config.toml
    ./deode/data/config_files/include/eps/\
      eps_3members_IFSENS_common_mars_prep_and_spp.toml --
start-suite
```

REMEMBER: Update the starttime for IFSENS experiment to be less than 2 weeks ago, since old eps data is removed from mars.

Example eps config files

Content of

```
./deode/data/config_files/include/eps/eps_3members_IFSENS_common_ma
```

:

```
[boundaries.ifs]
  bdmembers = [0, 1, 2]
  selection = "IFSENS"

[eps.general]
  members = "0:3"

[eps.member_settings.boundaries.ifs]
  bdmember = [0, 1, 2]

[eps.member_settings.namelist_update.master.forecast.namspp]
  iezdiag_pos = {0 = 1, "1:" = -1}
  lspp = {0 = false, "1:" = true}
```

Resulting config file

- Only deviating settings are saved

```
...

[eps.members]

[eps.members.0]

[eps.members.0.boundaries.ifs]
```

```
  bdmember = 0

[eps.members.0.namelist_update.master.forecast.namspp]
  iezdiag_pos = 1
  lspp = false

[eps.members.1]

[eps.members.1.boundaries.ifs]
  bdmember = 1

[eps.members.1.namelist_update.master.forecast.namspp]
  iezdiag_pos = -1
  lspp = true

...
```

How to specify member specific settings?

1. Single value -> all members get the same setting

```
[eps.member_settings]
parameter = value
```

2. List of values -> first member get first item, second member get second item, etc. (with "circular boundary condition")

```
[eps.member_settings]
parameter = [value1, value2, value3, ...]
```

3. Dict of mbr/value pairs -> a member get the value of the mbr key

```
[eps.member_settings]
parameter = {0: value1, 1: value2, "2:10:2": value3, ...}
```

How to specify member specific settings?

3. Dict of mbr/value pairs -> a member get the value of the mbr key

```
[eps.member_settings]
parameter = {0: value1, 1: value2, "2:10:2": value3, ...}
```

- "m:n" keys are interpreted as slices, that is `{"2:10:2": value3}` assigns `value3` to members 2, 4, 6, 8 and 10.
- The `[eps.general.members]` setting limits parameter slices. E.g. if `[eps.general.members] = "0:10"`,

```
[eps.member_settings]
parameter = {"6:16": value1}
```

will set `parameter = value1` for members 6, 7, 8, 9 and 10.

- For members with no mbr/value pair, the default is used. I.e. in the above example, members 0-5 will get the default value.

How to specify member specific settings?

4. Python subclass of `deode.eps.custom_generators:BaseGenerator`. Generates member settings based on list of realizations.

```
[eps.member_settings]
parameter = "deode.eps.custom_generators.BoolGenerator"
```

```
@pydantic_dataclass
class BoolGenerator(BaseGenerator[bool]):
    """Example generator class to generate random boolean
values."""

    def __iter__(self):
        for _ in self.members:
            yield random.choice([True, False])
```

How to specify member specific settings?

5. Using modification files. Handy in cases with many member specific adjustments, e.g. in CSC ensembles

```
[eps.member_settings.modifications]
mod1 = {1: "/path/to/mbr001/mod1.toml", 2:
"/path/to/mbr002/mod1.toml", ...}
```

NOTE:

- It's not important what the keys in the modification section are called. They are just used to label the different modification files.
- The settings in the modification files will overwrite any existing value for that setting.

How to specify member specific settings?

E.g. for ensemble of the 3 CSCs:

```
[eps.general]
  members = "0:3"
  run_continously = false
```

```
[eps.member_settings.include]
  csc_include = ["modifications/arome.toml",
"modifications/harmonie_arome.toml", "modifications/alaro.toml"]
  cycle_include = ["modifications/CY48t3.toml",
"modifications/CY46h1.toml", "modifications/CY48t3.toml"]
  submission_include =
["modifications/submission/@HOST@_CY48t3.toml",
"modifications/submission/@HOST@_CY46h1.toml",
"modifications/submission/@HOST@_CY48t3_alaro.toml"]
  vertical_levels_include = {2 =
"include/vertical_levels/MF_87.toml"}

[eps.member_settings.system]
  wrk = "@CASEDIR@/@YYYY@@MM@@DD@_@HH@@mm@/@MEMBER_STR@"

[suite_control]
  member_specific_mars_prep = false
  member_specific_static_data = true
```

Location of data

- mbr??? in paths, e.g.

```
ls
/scratch/dnk3604/deode/EPS_DEMO_CASES1_nwp_DEMO_60x80_2500m_2025020
config.toml                GRIBDEOD+0005h00m00s.sfx
GRIBPFDEOD+0004h00m00s     ICMSHDEOD+0001h00m00s.sfx
ICMSHDEOD+0004h00m00s.sfx
GRIBDEOD+0000h00m00s.sfx   GRIBDEOD+0006h00m00s.sfx
GRIBPFDEOD+0005h00m00s     ICMSHDEOD+0002h00m00s
ICMSHDEOD+0005h00m00s
GRIBDEOD+0001h00m00s.sfx   GRIBPFDEOD+0000h00m00s
GRIBPFDEOD+0006h00m00s     ICMSHDEOD+0002h00m00s.sfx
ICMSHDEOD+0005h00m00s.sfx
GRIBDEOD+0002h00m00s.sfx   GRIBPFDEOD+0001h00m00s
ICMSHDEOD+0000h00m00s      ICMSHDEOD+0003h00m00s
ICMSHDEOD+0006h00m00s
GRIBDEOD+0003h00m00s.sfx   GRIBPFDEOD+0002h00m00s
ICMSHDEOD+0000h00m00s.sfx  ICMSHDEOD+0003h00m00s.sfx
ICMSHDEOD+0006h00m00s.sfx
GRIBDEOD+0004h00m00s.sfx   GRIBPFDEOD+0003h00m00s
ICMSHDEOD+0001h00m00s      ICMSHDEOD+0004h00m00s
ICMSHDEODINIT.sfx

els
ec:/dnk3604/deode/EPS_DEMO_CASES1_nwp_DEMO_60x80_2500m_20250209/sql
FCTABLE_CCtot_202502_00.sqlite
FCTABLE_CCtot_202502_06.sqlite
FCTABLE_D10m_202502_00.sqlite
FCTABLE_D10m_202502_06.sqlite
```

```
FCTABLE_D_202502_00.sqlite
FCTABLE_D_202502_06.sqlite
FCTABLE_Gmax_202502_00.sqlite
FCTABLE_Gmax_202502_06.sqlite
FCTABLE_Pcp_202502_00.sqlite
...
```

Documentation

The EPS setup is documented at https://destination-earth-digital-twins.github.io/deode-workflow-docs/misc_section_in_doc_page.html#eps-configuration-by-design