聚变模拟与建模联合平台开发环境介绍（II）

（*Fu*Yun开发应用“数据绑定/插件机制”）

于治

2024.7.30

线上

关于 *FUSMAP*

- **FU**sion **S**imulation and **M**odelling **A**lliance **P**latform，(FUSMAP)
- 聚变模拟与建模联合平台

FUSMAP

- **https://fusmap.github.io/**

关于 *Fu*Yun

- 为FUSMAP提供集成交互环境
- **集成建模和数据分析工具集，**
- 面向科学工程的知识管理和计算环境
- 授权情况：
  - 框架主体开源，
  - 插件授权由独立开发者各自决定
- 开发状态： ***Sp*DM** 0.5.2/ ***Fy*Tok** 0.5.2
  - API 可能会由微小变动
  - 遇到 bug 请在github上提起issue
- **https://fusion-yun.github.io/**

关于*这个报告*

- **目标**：
  1. 向**用户**展示平台使用场景，持续收集需求反馈，改进、完善平台功能。
  2. 向**开发者**介绍平台开发环境和集成接口，增进交流，促进广泛合作。
- 相关文档和数据：

https://github.com/FusMap/fytok_tutorial

https://github.com/fusion-yun/fytok_tutorial

（关于*Fu*Yun的问题可在这个项目下发起issue）

主要内容：

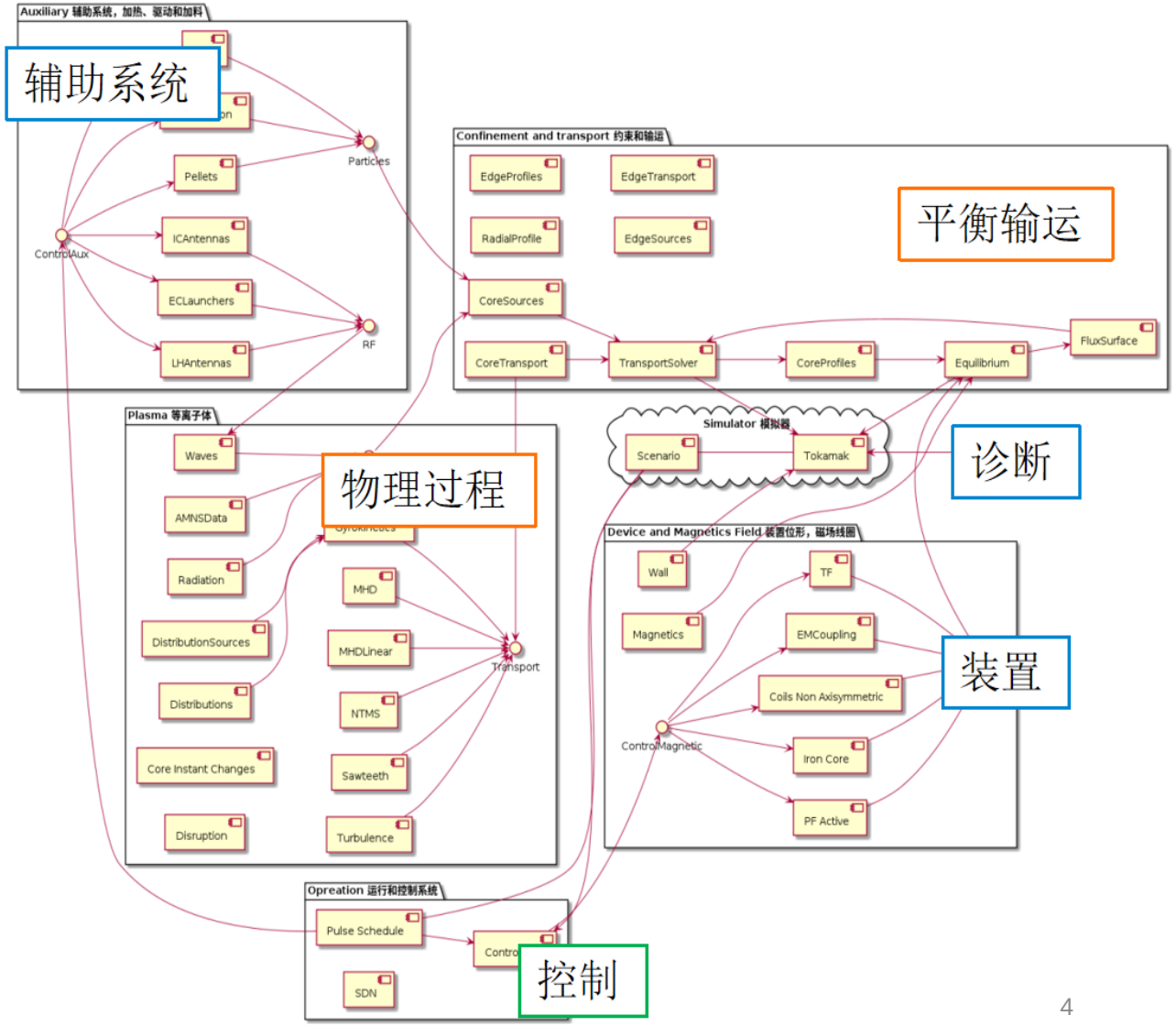
1. **运行环境和安装**
2. **数据集成：可视化，平衡分析**
3. **程序集成**
   - **数据结构和功能绑定**
   - **模块插件机制**
4. **物理集成**
5. **小结**

IMAS 数据字典（DD）

IMAS DD 的数据组织是一个“树”状结构，当增加“物理”约束后才成为一个“图”状结构，即“本体”。

托卡马克描述分解为多个**树状数据结构**（IDS）：

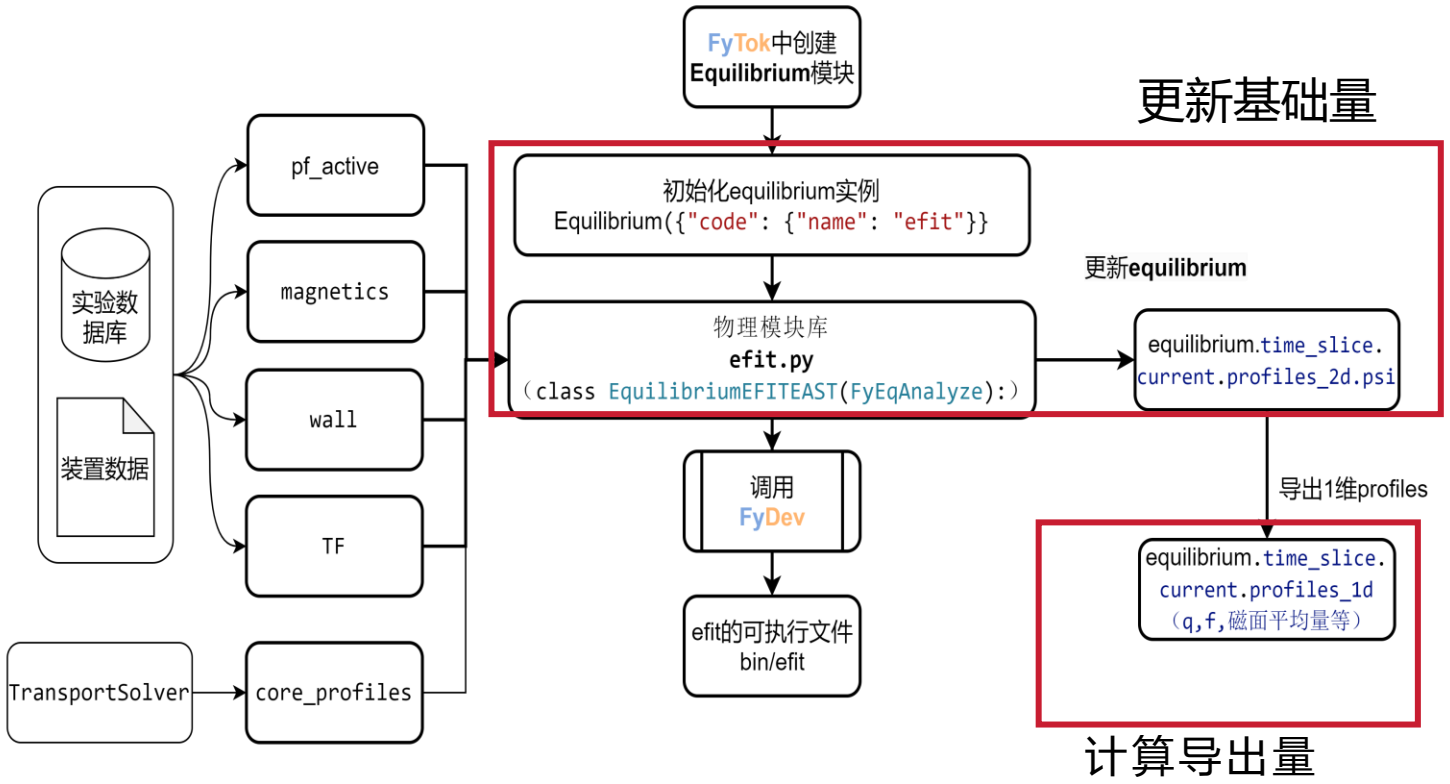
- **子系统**：描述**装置**几何参数，实验**诊断**或**控制**信号等。作为数据源，默认只读属性。如wall, pf_active, magnetics 等
- **物理概念**：描述抽象物理概念，通常为同一**物理概念**的物理量的集合。
  - **状态描述**：用于数据传递，无需调用模块程序，如equilibrium，core_profiles 等
  - **计算过程**：用于管理计算过程，需调用模块程序，如 core_transport, core_sources, transport_solver 等
- 划分并不绝对，具体实现会有变通



执行过程的组织

为了保证执行过程中，子程序中物理量的的统一，定义基础量和导出量。

- 将同一物理概念的相关物理量组织成树状结构，组成**状态树。**
- 每个**状态树**中包含两类物理量：
  - 基础量，包含物理概念的全部信息，如equilibrium.profiles_2d.psi
  - 导出量，由基础物理量计算得出，如equilibrium.profiles_1d.q
- 同一**状态树**的所有物理量必须保证描述的是同一**时刻**的同一物理对象。
- 架构保证了执行过程中数据的统一。



数据结构

- DD 数据四个要素：
  - 树状组织
  - 物理语义
  - 类型描述
  - 坐标关系
- 同一个子树下的元素常具有相同的坐标
- 值+坐标=插值函数
- 属性应以函数形式返回

ITER Physics Data Model Documentation for core_profiles

IDS 名

Core plasma profiles

Notation of array of structure indices: itime indicates a time index; i1, i2, i3, ... indicate other indices with their depth in the IDS. This notation clarifies the path of a given node, but should not be used to compare indices of different nodes (they may have different meanings).

Lifecycle status: active since version 3.1.0

Last change occured on version: 3.40.0

描述信息，版本号

Back to top IDS list

Flat display | Show/Hide errorbar nodes By convention, only the upper error node should be filled in case of symmetrical error bars. The upper and lower errors are absolute and defined positive and represent one standard deviation of the data. The effective values of the data (within one standard deviation) will be within the interval [data-data_error_lower, data+data_error_upper]. Thus whatever the sign of data, data_error_lower relates to the lower bound and data_error_upper to the upper bound of the error bar interval.

Full path name	Description	Data Type	Coordinates
▸ ids_properties	Interface Data Structure properties. This element identifies the node above as an IDS	structure	
▾ profiles_1d(itime)	Core plasma radial profiles for various time slices {dynamic}	array of structures	1- profiles_1d(itime)/time
▸ grid	Radial grid	structure	
▾ electrons	Quantities related to the electrons	structure	
temperature(:)	Temperature {dynamic} [eV]	FLT_1D	1- any of profiles_1d(itime)/grid/rho_tor_norm; profiles_1d(itime)/grid/rho_tor; profiles_1d(itime)/grid/psi; profiles_1d(itime)/grid/volume; profiles_1d(itime)/grid/area; profiles_1d(profiles_1d(
temperature_validity	Indicator of the validity of the temperature profile. 0: valid from automated processing, 1: valid and certified by the RO; - 1 means problem identified in the data processing (request verification by the RO), -2: invalid data, should not be used {dynamic}	INT_0D	
▸ temperature_fit	Information on the fit used to obtain the temperature profile [eV]	structure	
density(:)	Density (thermal+non-thermal) {dynamic} [m^-3]	FLT_1D	1- any of profiles_1d(itime)/grid/rho_tor_norm; profiles_1d(itime)/grid/rho_tor; profiles_1d(itime)/grid/psi; profiles_1d(itime)/grid/volume; profiles_1d(itime)/grid/area; profiles_1d(itime)/grid/surface; profiles_1d(itime)/grid/rho_pol_norm
	Indicator of the validity of the density profile. 0: valid from automated processing, 1: valid and certified by the RO; - 1		

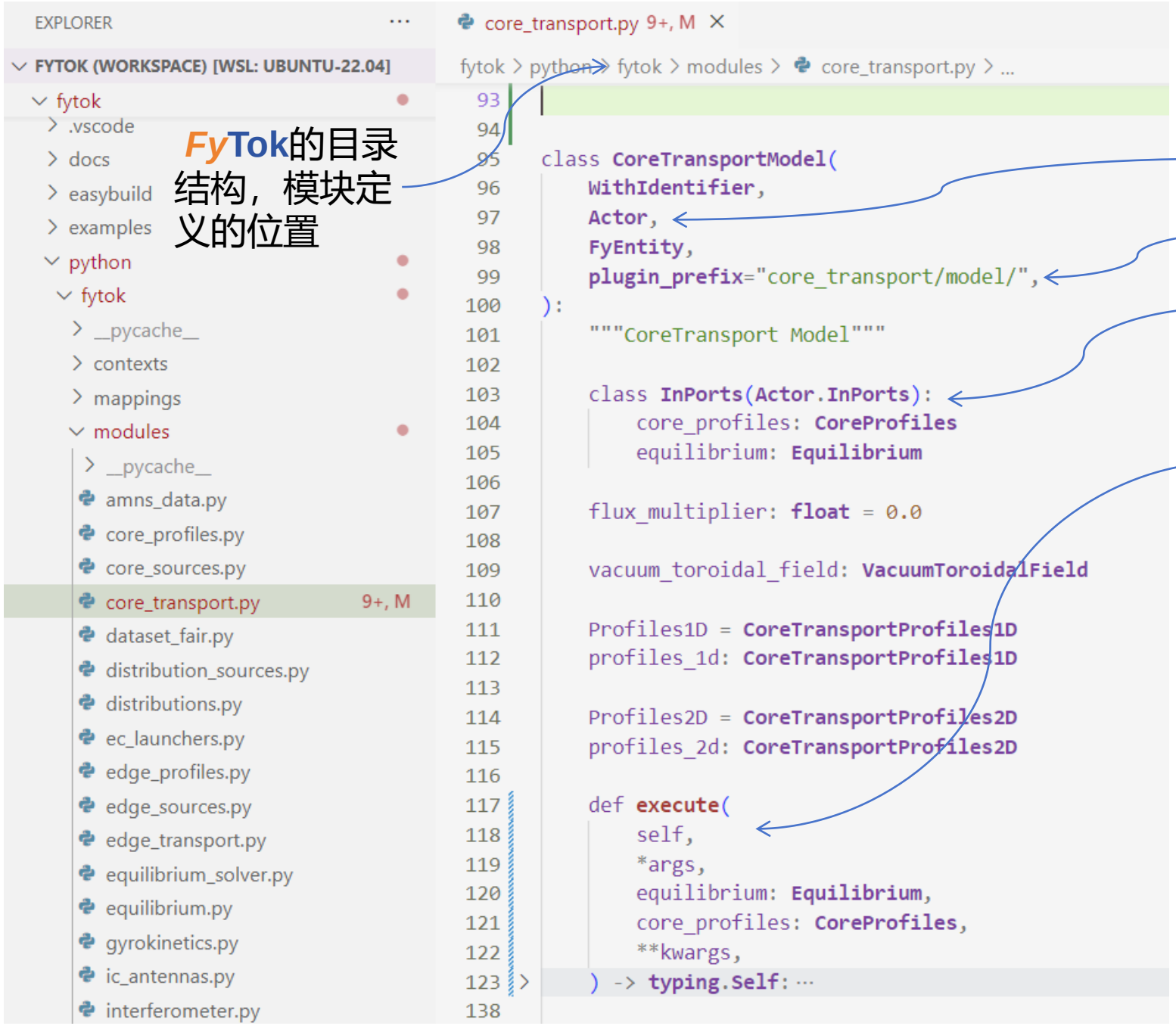
树状结构命名　语义描述　类型描述　坐标

状态描述数据绑定

- 声明树状结构 CoreProfiles1D
- 声明成员共同（默认）**坐标系**
- 声明子类型
- 声明成员类型
  Expression/Function/Field 。。。
- 定义成员的**元数据**
- 定义成员的**默认值** default_value
- 定义成员的**计算过程**

```python
class CoreProfiles1D(WithDomain, SpTree, domain="grid/rho_tor_norm"):
    """Core Profiles 1D"""

    grid: CoreRadialGrid = {"primary_coordinate": "rho_tor_norm"}

    Electrons = CoreProfilesElectrons
    electrons: CoreProfilesElectrons

    Ion = CoreProfilesIon
    ion: Set[CoreProfilesIon]

    Neutral = CoreProfilesNeutral
    neutral: Set[CoreProfilesNeutral]

    rho_tor_norm: Expression = annotation(label=r"\bar{\rho}_{tor}", units="-")
    rho_tor: Expression = annotation(label=r"\rho_{tor}", units="m")
    psi_norm: Expression = annotation(label=r"\bar{\psi}", units="-")
    psi: Expression = annotation(label=r"\psi", units="Wb")

    @sp_property
    def zeff(self) -> Expression:
        return sum(((ion.z_ion_1d**2) * ion.density) for ion in self.ion) / self.n_i_total

    @sp_property
    def pressure(self) -> Expression:
        return sum([ion.pressure for ion in self.ion], self.electrons.pressure)

    @sp_property
    def pprime(self) -> Expression:
        return self.pressure.d

    @sp_property
    def pressure_thermal(self) -> Expression:
        return sum(ion.pressure_thermal for ion in self.ion) + self.electrons.pressure_thermal

    @sp_property
    def t_i_average(self) -> Expression:
        return sum(ion.z_ion_1d * ion.temperature * ion.density for ion in self.ion) / self.n_i_total
```

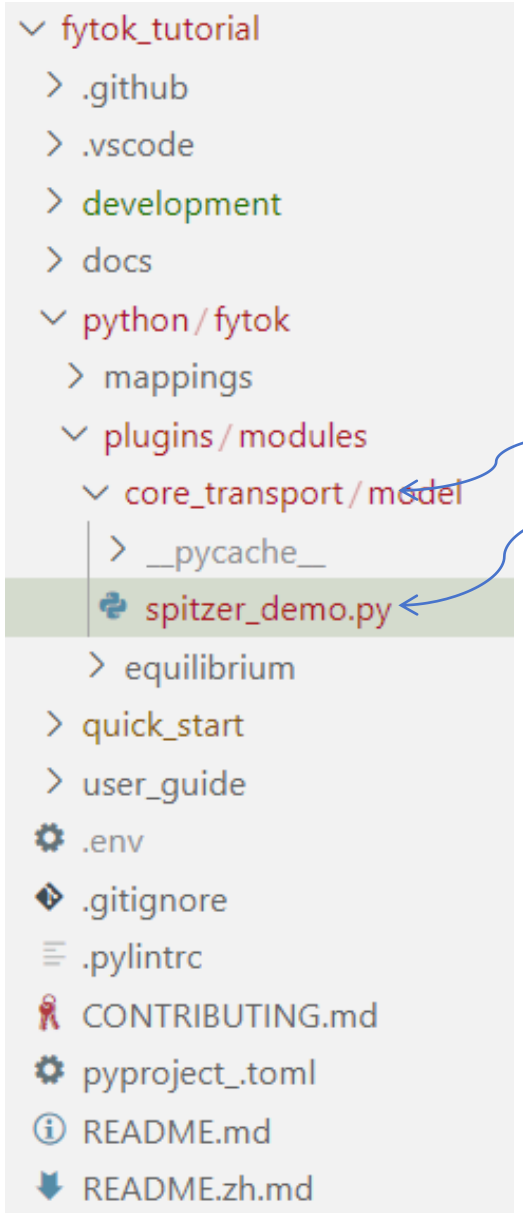
计算过程接口定义



*FyTok*的目录结构，模块定义的位置

- 声明为**执行体**Actor
- 声明**插件**查找路径
- 声明**输入**（依赖）
- 声明**输出**（Actor的属性）
- 定义执行**执行函数**
  - **Actor.execute** 根据输入执行计算返回结果，但不更换新Actor的状态
  - **Actor.refresh** 执行Actor.execute 并用结果更新Actor的状态

自定义插件



- 继承**接口类**
- 声明为模块的**分类**
- 插件以Pyton Package的路径 fytok/modules/**<plugin_prefix>/<plugin name>**
- 定义执行**执行函数** **Actor.execute** 根据输入执行计算返回结果，
- 采用默认 Actor.refresh 维护状态

```python
import typing
import numpy as np
import scipy.constants

from fytok.modules.equilibrium import Equilibrium
from fytok.modules.core_profiles import CoreProfiles
from fytok.modules.core_transport import CoreTransport
from fytok.modules.utilities import CoreRadialGrid


class SpitzerDemo(
    CoreTransport.Model,
    identifier="neoclassical",
    code={"name": "spitzer_demo"},
):
    """Spitzer resistivity model"""

    def execute(
        self, *args, equilibrium: Equilibrium, core_profiles: CorePr
    ) -> typing.Self:

        res: typing.Self = super().execute(
            *args,
            equilibrium=equilibrium,
            core_profiles=core_profiles,
            **kwargs,
        )

        eq1d: Equilibrium.Profiles1D = equilibrium.profiles_1d

        prof1d: CoreProfiles.Profiles1D = core_profiles.profiles_1d

        radial_grid: CoreRadialGrid = res.profiles_1d.grid
```

小结

- **数据集成**统一数据访问形式，通过（type hint/annotation）**自动绑定** raw data 实现**类型信息**和**元数据**（metadata）的结合;
- 通过绑定Function到属性sp_property，实现**成员**数据间的**物理语义约束**;
- 通过插件机制，实现**物理程序**接口统一到**计算模块**，实现了**程序集成**
- 当前版本 ***Sp*DM** 0.5.2/ ***Fy*Tok** 0.5.2
- 本报告示例 fytok_tutorial/quick_start/ch3_modules.ipynb
- 反馈 bug/issue
  https://github.com/FusMap/fytok_tutorial

  https://github.com/fusion-yun/fytok_tutorial

谢谢！