The Pineline:

Industrialization of High-Energy theory predictions

Andrea Barontini, A. Candido, J. Cruz Martinez, F. Hekhorn, C. Schwan

(Re)interpretation of LHC results for new physics

30/08/2023

Outline

Introduction and motivation

The Pineline

Applications and outlook

Introduction and motivation

The Pineline

Applications and outlook

Problems and Goals

High runtime and development time for new observables

No straightforward way to reproduce results

Proliferation of short standing codes

Including new computations

- Reduce **runtime** and **development** time
- Provide a **common I/O interface**

Ensure reproducibility

- **Storing** intermediate steps
- Produce and track **logs** and **metadata**

Make it last

- **Open Source** from the beginning
- **Fully documented**

The Pineline

https://github.com/NNPDF/pineline

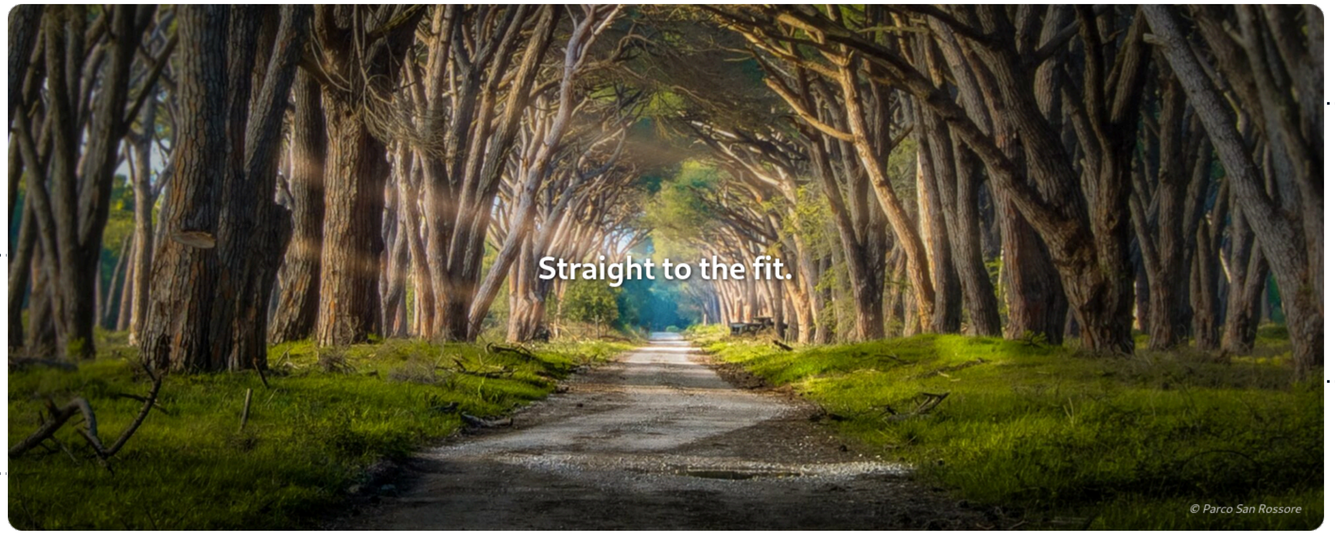
https://nnpdf.github.io/pineline

Single I/O format

Provides *translation* layers

Open Source

and fully documented



Industrialization

Assembly line of generators

Reproducibility

Easy inspection of metadata

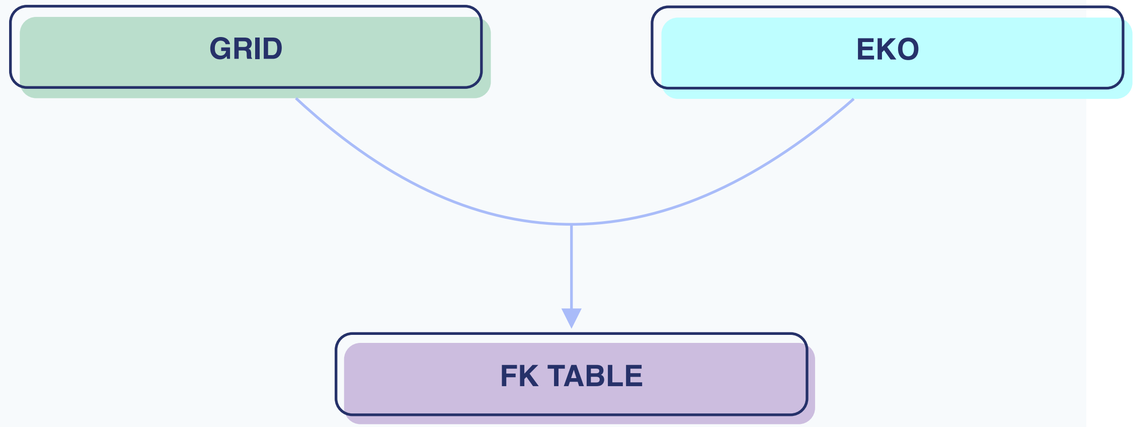
Introduction and motivation

The Pineline

Applications and outlook

What we deliver

Fast Kernel (**FK**) tables [NuclPhysB838.136]

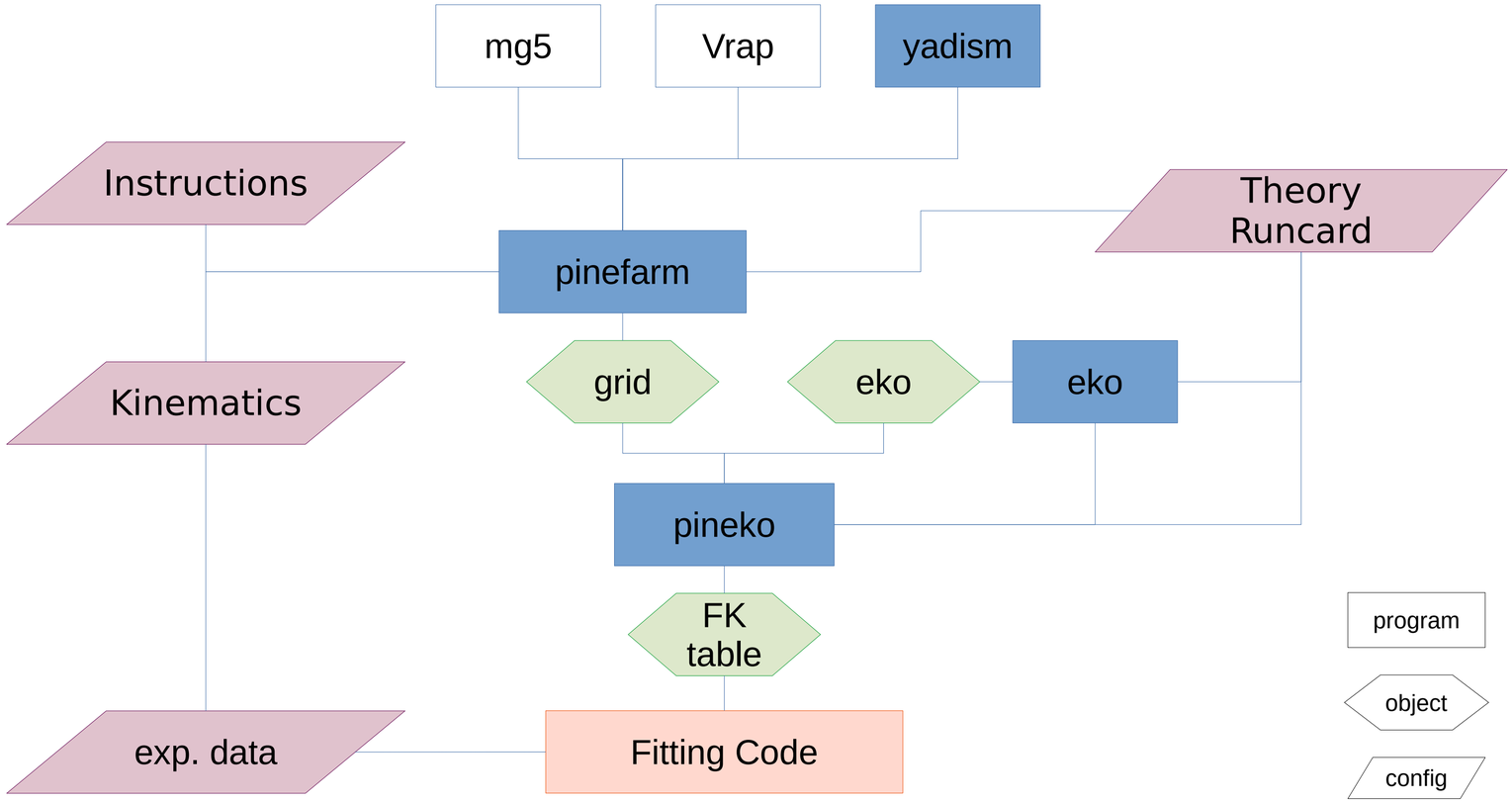


$$F(Q) = \hat{\sigma}(Q) \otimes f(Q)$$

$$= \hat{\sigma}(Q) \otimes E(Q \leftarrow Q_0) \otimes f(Q_0)$$

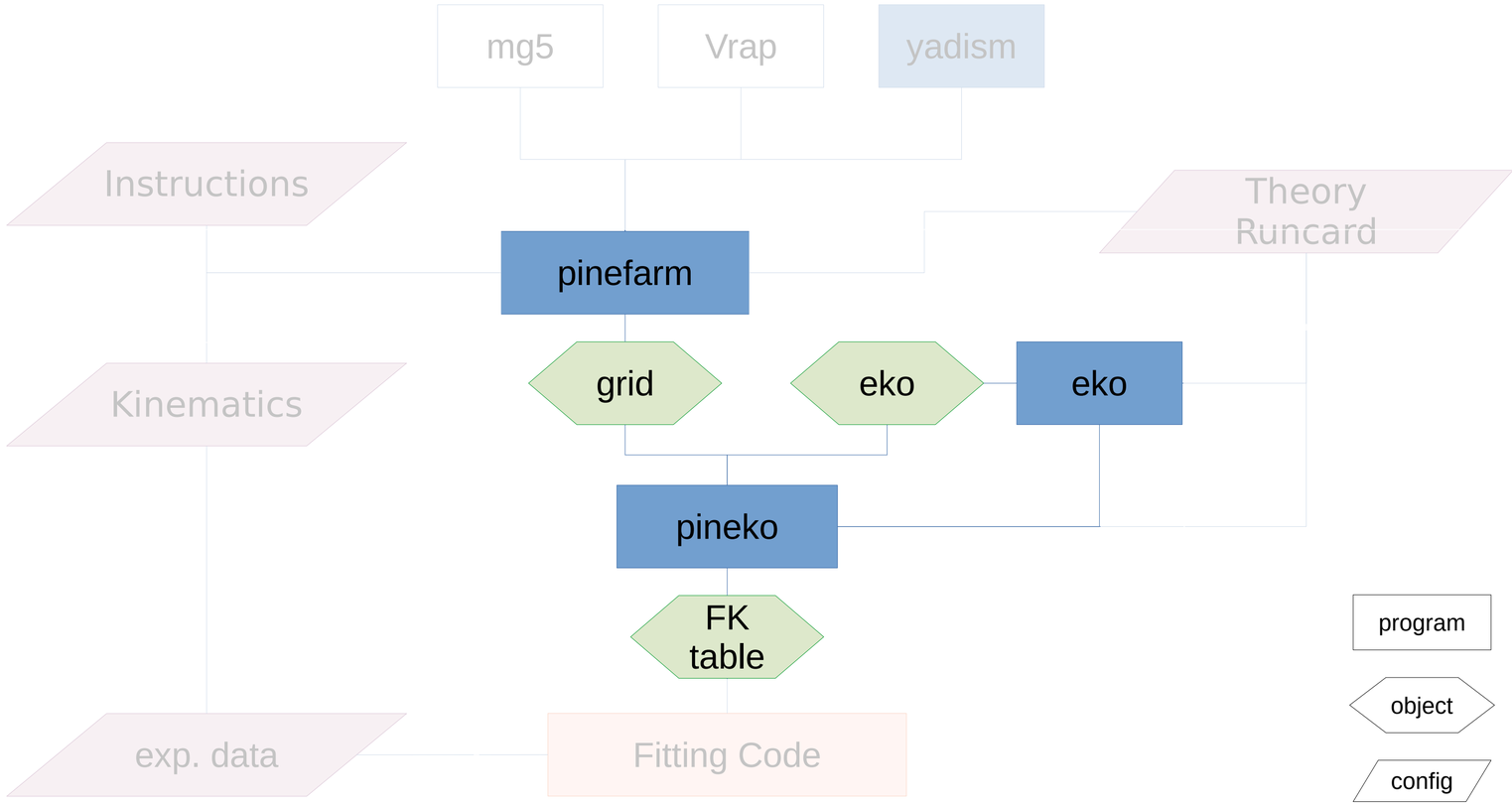
$$F(Q) = FK(Q \leftarrow Q_0) \otimes f(Q_0)$$

The pineline flow



The workhorse in the background → PineAPPL

The pineline flow

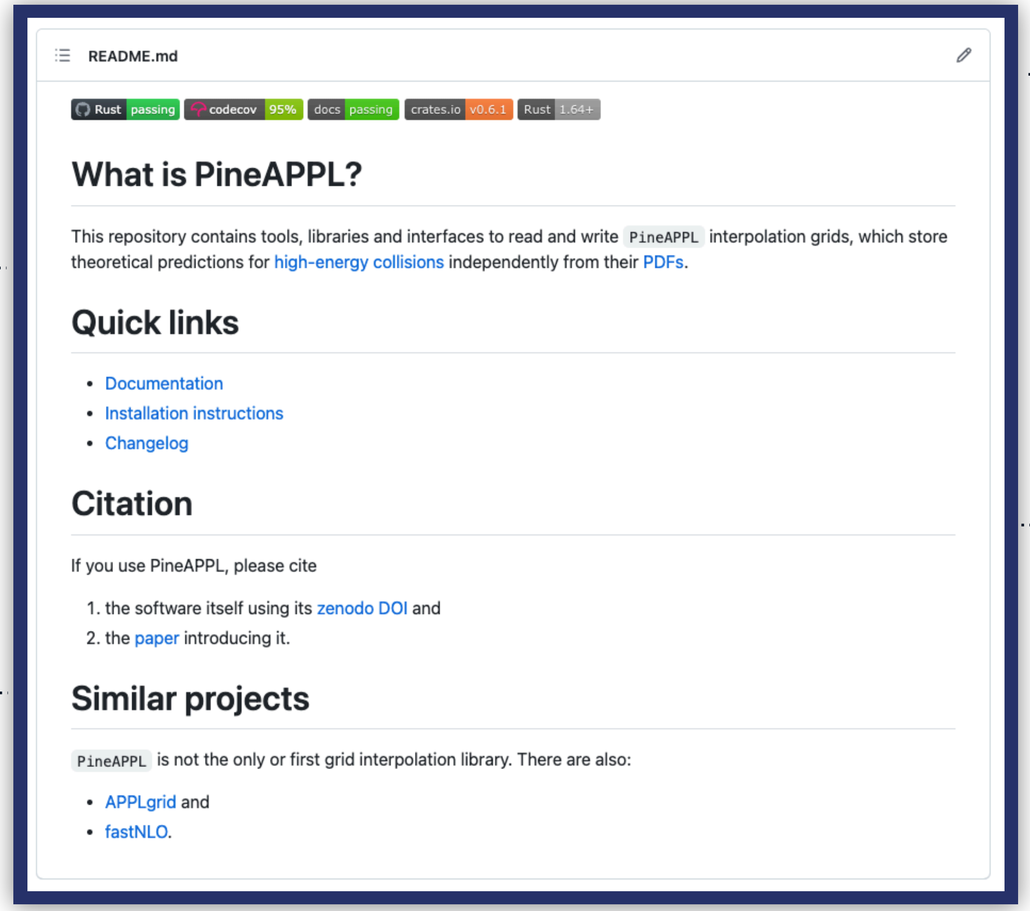


The workhorse in the background → PineAPPL

PineAPPL [JHEP12.108]

https://github.com/NNPDF/pineappl

https://nnpdf.github.io/pineappl



Very flexible

Extends to arbitrary orders in QCD and EW

Command line interface

for everyday tasks

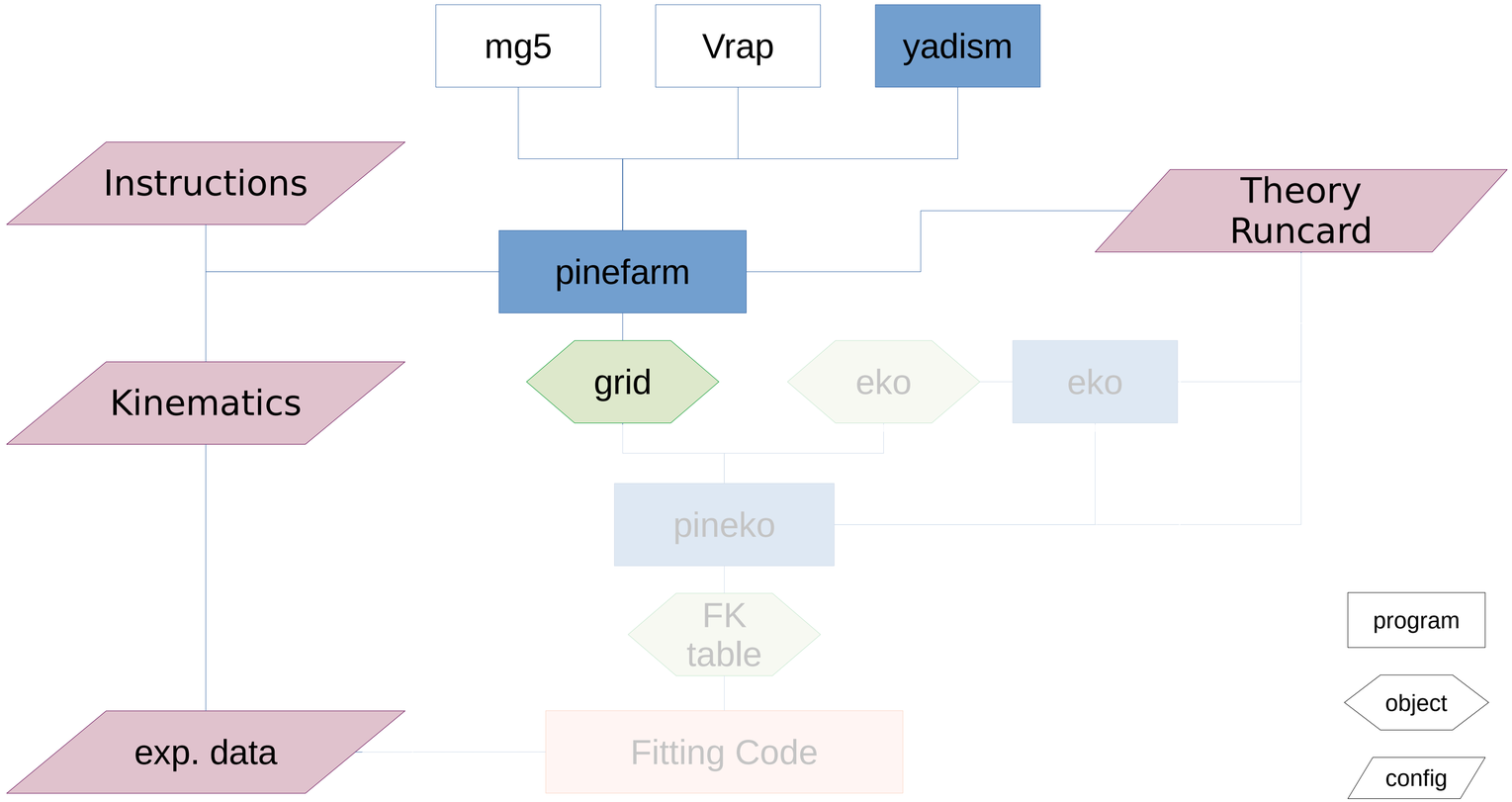
Fast interpolation grid library

Can convert **APPLgrid** and **FastNLO**

Several interfaces

C, C++, Fortran, Rust, Python

The pineline flow



The workhorse in the background → PineAPPL

Pinefarm [HEP-PH2302.12124]

https://github.com/NNPDF/pinefarm

https://pinefarm.readthedocs.io/en/latest

Pinefarm

tests passing docs passing

Generate PineAPPL grids from pinecards.

Installation

pinefarm is available via

- PyPI: pypi v0.3.0

```
pip install pinefarm
```

Dev

For development you need the following tools:

- `poetry`, follow installation instructions
- `poetry-dynamic-versioning`, used to manage the version (see repo)
- `pre-commit`, to run maintenance hooks before commits (see instructions)

See below for a few more dependencies (already available on most systems).

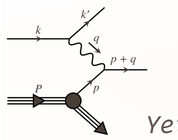
Documentation

- The documentation is available here: docs passing
- To build the documentation from source run these commands

```
poetry shell
cd docs
make html
make view
```

Different providers

MadGraph5, Vrap,Yadism, (Matrix)



Yadism

Yet Another DIS Module

```
*****************************************************
*                                                   *
*  W E L C O M E  to  M A D G R A P H 5 _ a M C @ N L O  *
*                                                   *
*                                                   *
*                *                  *               *
*                  *      * *       *               *
*                    * * * * 5 * * * *               *
*                  *      * *       *               *
*                *                  *               *
*                                                   *
*  The MadGraph5_aMC@NLO Development Team - Find us at  *
*  https://server06.fynu.ucl.ac.be/projects/madgraph  *
*                        and                        *
*              http://amcatnlo.cern.ch              *
*                                                   *
*                Code download from:                *
*           https://launchpad.net/madgraph5         *
*                                                   *
*  Please refer to: MadGraph5_aMC@NLO paper         *
*             J. Alwall et al.                      *
*             arXiv:1405.0301, JHEP 1407 (2014) 079 *
*                                                   *
*****************************************************
```

Produces the grids

Calls a providers according to configs

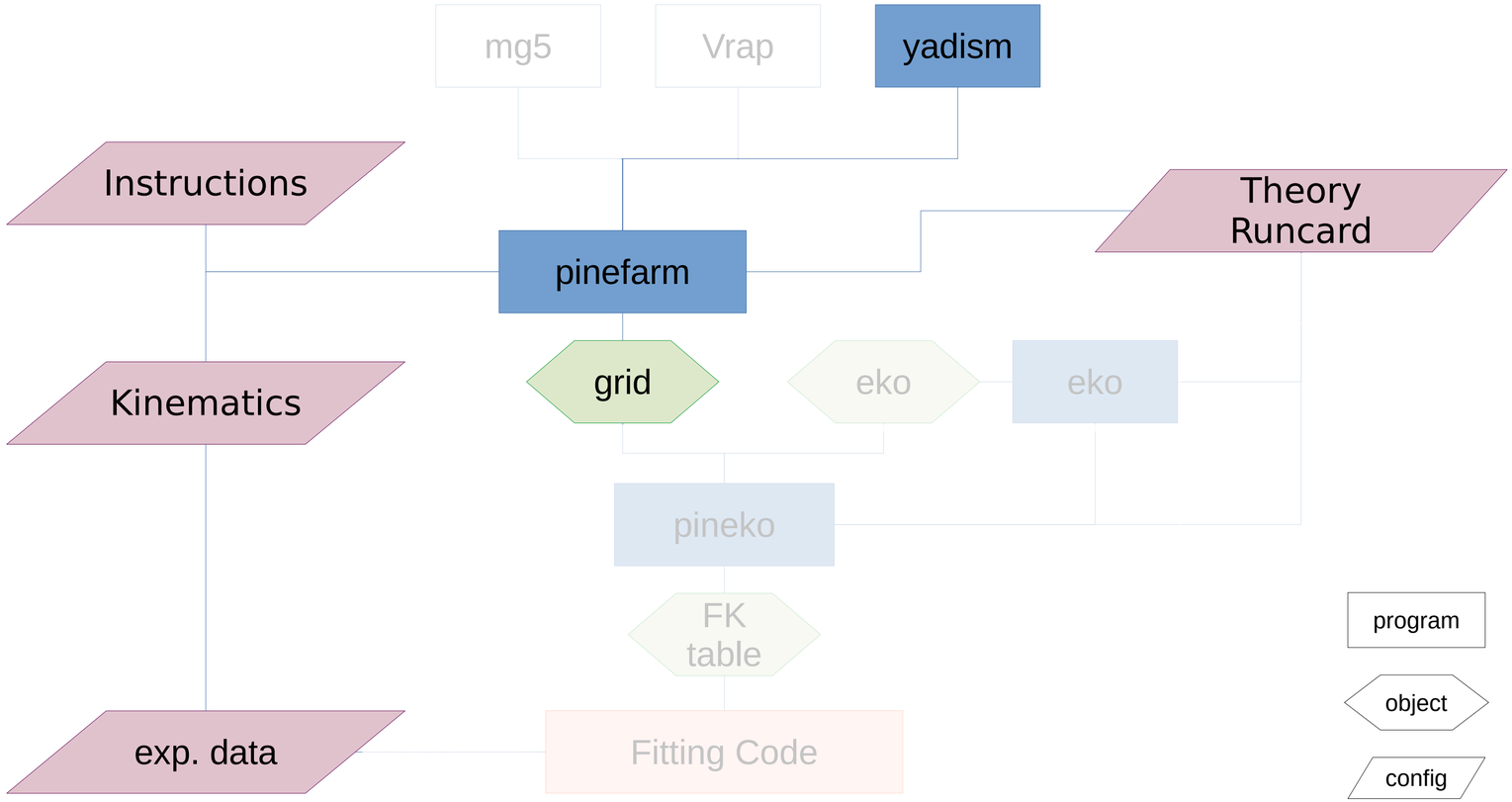
Standard input

Pinecards format

https://github.com/NNPDF/pinecards

ATLAS_TTB_8TEV_LJ_TTRAP	Fix ordering of model loading and model-specific settings	2 weeks ago
ATLAS_TTB_8TEV_TOT	Fix ordering of model loading and model-specific settings	2 weeks ago
ATLAS_WM_7TEV	Fix ordering of model loading and model-specific settings	2 weeks ago
ATLAS_WP_7TEV	Fix ordering of model loading and model-specific settings	2 weeks ago
BCDMS_NC_EM_D_F2	Export pinefarm to its own repo	3 months ago
BCDMS_NC_EM_P_F2	Export pinefarm to its own repo	3 months ago
CHORUS_CC_NB_PB_SIGMARED	Export pinefarm to its own repo	3 months ago
CHORUS_CC_NU_PB_SIGMARED	Export pinefarm to its own repo	3 months ago
CMS_2JET_7TEV_0005	Fix ordering of model loading and model-specific settings	2 weeks ago
CMS_2JET_7TEV_0510	Fix ordering of model loading and model-specific settings	2 weeks ago
CMS_2JET_7TEV_1015	Fix ordering of model loading and model-specific settings	2 weeks ago

The pineline flow



The workhorse in the background → PineAPPL

Yadism [in preparation]

https://github.com/NNPDF/yadism

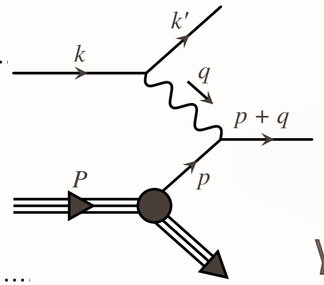
https://yadism.readthedocs.io/en/latest/

Flavor number schemes

FFNS, ZM-VFNS, FONLL

Coefficient function database

up to NNLO



Yadism

Yet Another DIS Module

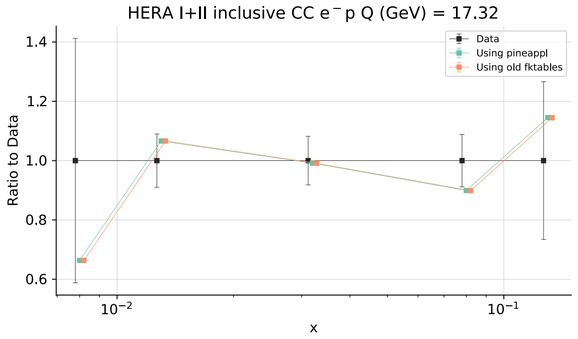
DIS provider

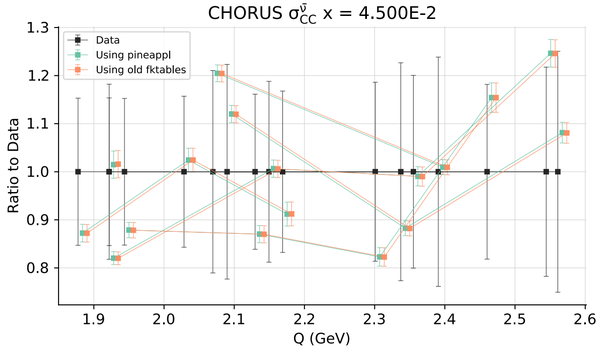
Independent of boundary condition

Benchmarked

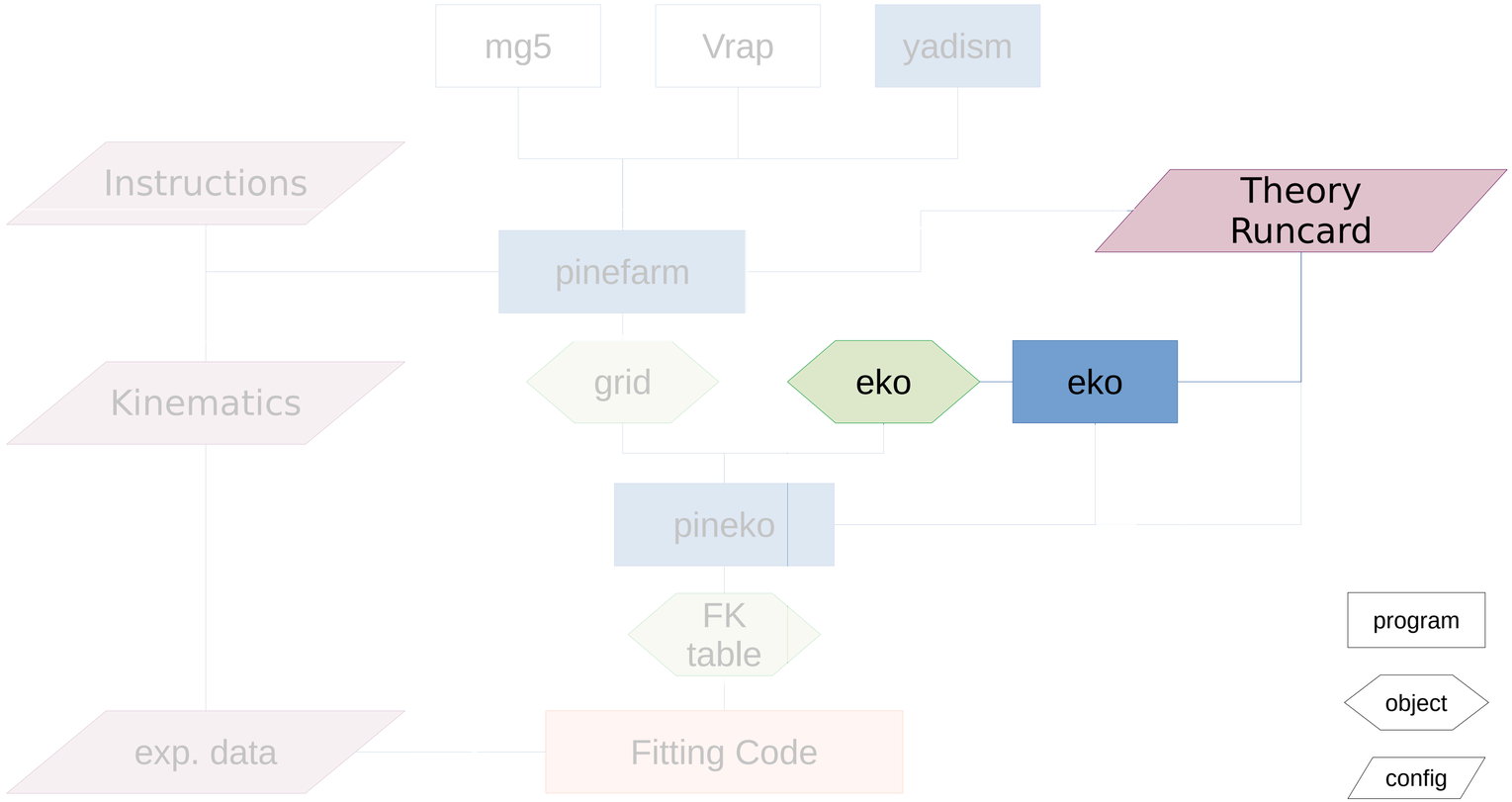
for example, with APFEL

	Light	Heavy	Intrinsic
NC	$\mathcal{O}(\alpha_s^2)$	$\mathcal{O}(\alpha_s^2)$	$\mathcal{O}(\alpha_s)$
CC	$\mathcal{O}(\alpha_s^2)$	$\mathcal{O}(\alpha_s)$	$\mathcal{O}(\alpha_s)$





The pineline flow



The workhorse in the background → PineAPPL

EKO [EPJC82.976]

https://github.com/NNPDF/eko

https://eko.readthedocs.io/en/latest/

Mellin space solution

but delivery in momentum space

Backward VFNS evolution

across thresholds and with intrinsic

EKO

Evolution Kernel Operators

Delivers DGLAP solution

in terms of an evolution kernel operator (**EKO**)

$$f(Q) = E(Q \leftarrow Q_0) \otimes f(Q_0)$$

Independent of boundary condition

Introduction and motivation

The Pineline

Applications and outlook

Applications and (future) improvements

We are here

Past

- Parton distributions and new physics searches: the Drell–Yan forward–backward asymmetry as a case study **[EPJC82.1160]**
- Evidence for intrinsic Charm in the proton **[Nature608.483]**

Coming soon

- QED corrections **(NNPDF40QED)**
- Theory errors **(NNPDF40MHOU)**
- Approximate N3LO **(NNPDF40N3LO)**

Future

- Interfaces to new providers
- Extension to fragmentation functions
- Addition of polarized setup

Conclusions

- The Pineline is a framework to produce High-Energy theory predictions in a fast and reproducible way
- It is completely Open Source and also provides interfaces to external providers
- It has been already used and it is being used for projects of PDF fitting but also for other kind of applications

Industrialization of High-Energy theory predictions

Andrea Barontini, A. Candido, J. Cruz Martinez, F. Hekhorn, C. Schwan

(Re)interpretation of LHC results for new physics

30/08/2023