Yoran Yeh
Department of Physics & Astronomy
University College London

Rivet, YODA and Contur: recent updates and ongoing developments

25th MCnet meeting 2023 at CERN

Yoran Yeh on behalf of the CEDAR team

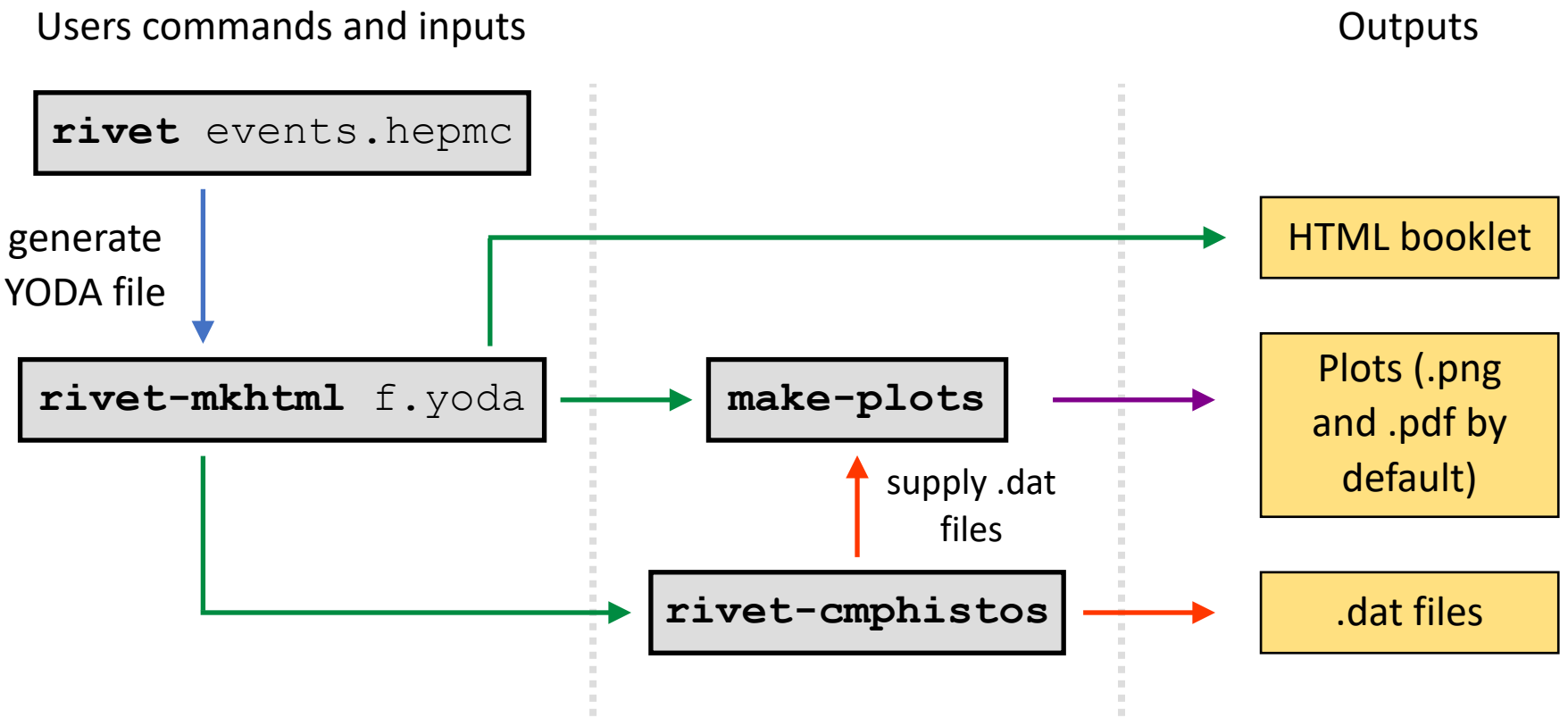
April 26

Rivet

- *Robust Independent Validation of Experiment and Theory*

- Upcoming 3.1.8 release
  - Many new analyses including O(200) new hadron decay analyses. Thanks to Peter Richardson for these!
  - First functional version of Matplotlib-based plotting

rivet-mkhtml



rivet-mkhtml

Users commands and inputs

Outputs

Getting started Rivet README

Installation

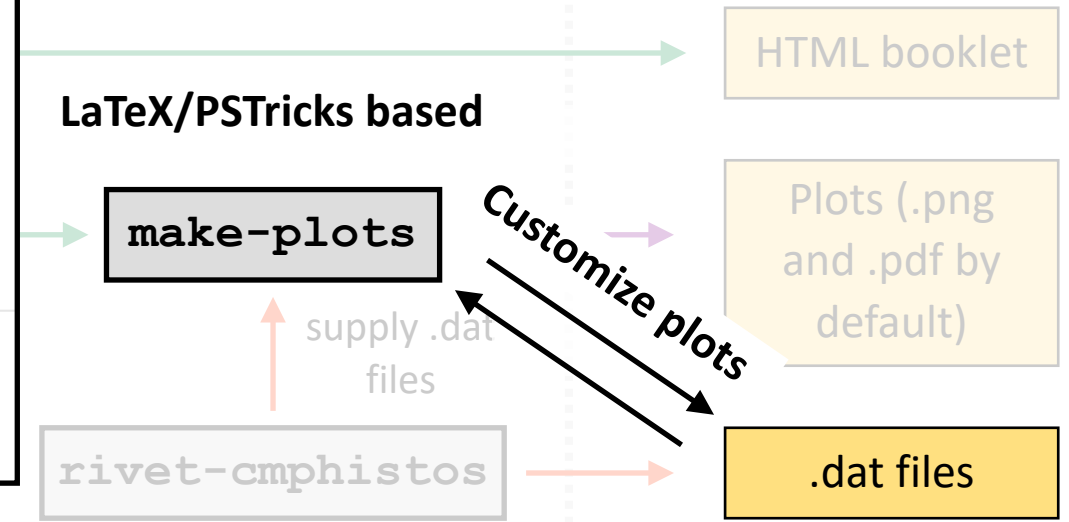
Rivet via Docker

First rivet run

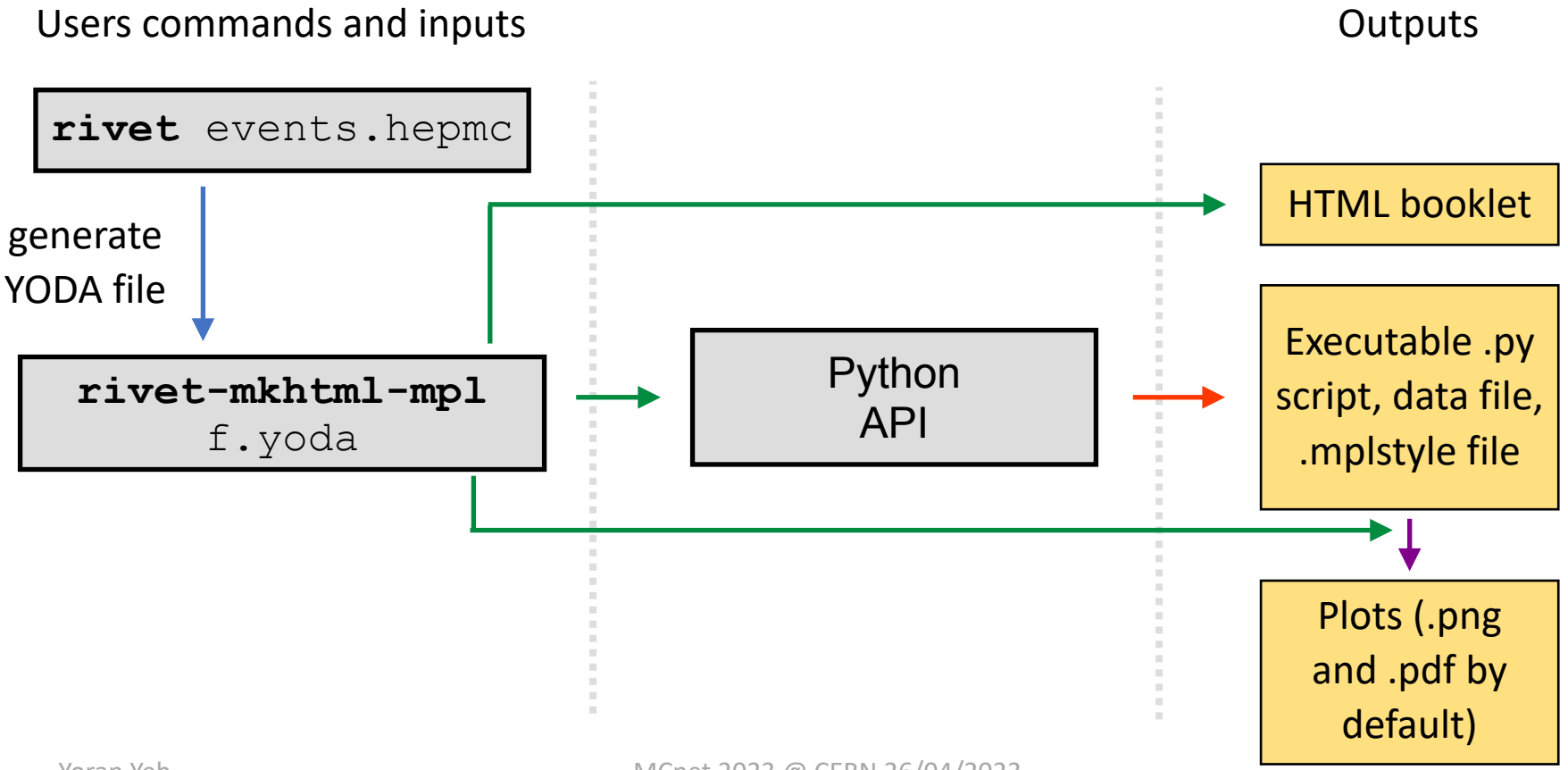
Plotting and run merging

Plotting with `rivet-mkhtml`

Customize plots with `make-plots`

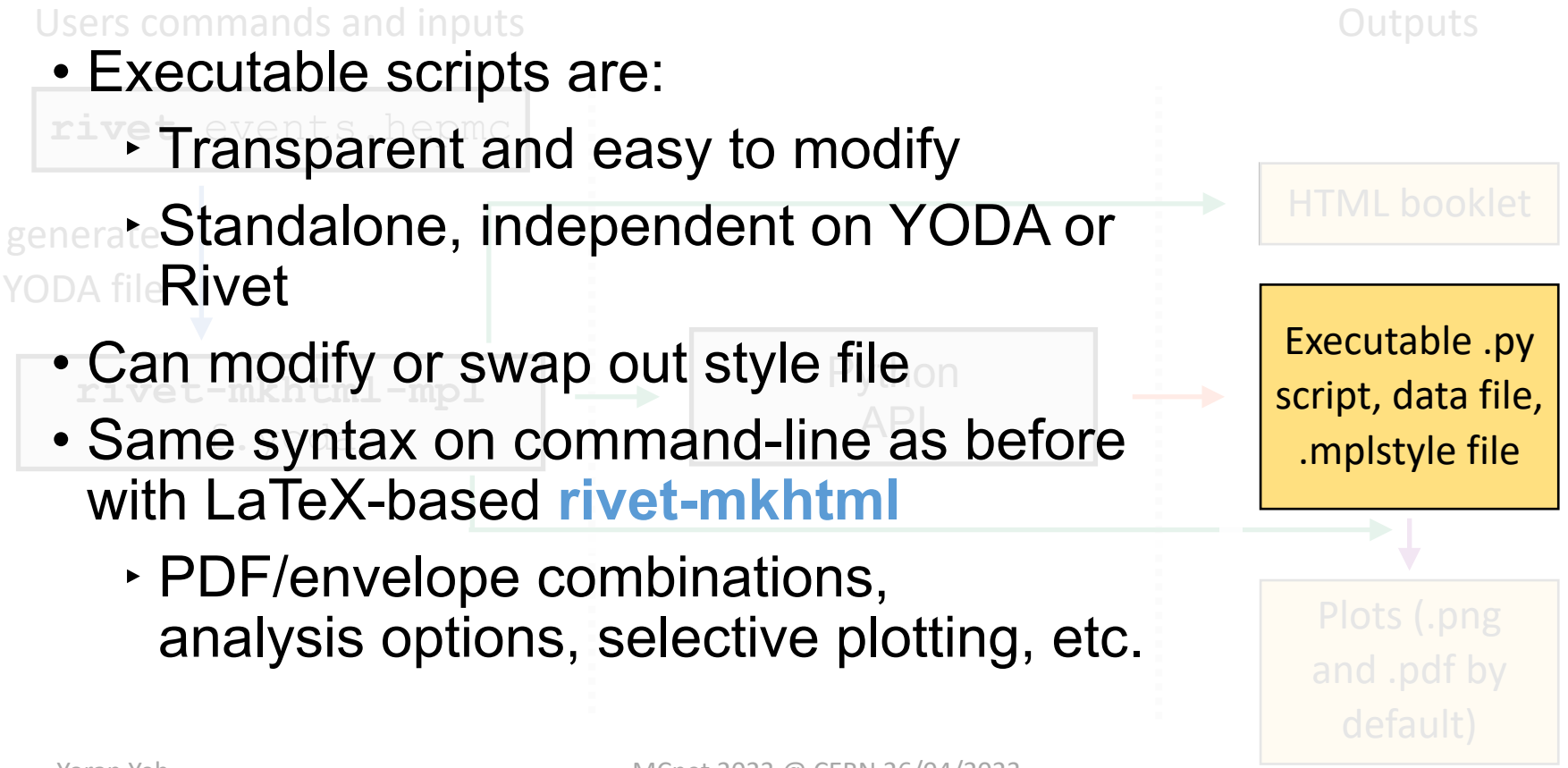


rivet-mkhtml-mpl



rivet-mkhtml-mpl

- Executable scripts are:
  - Transparent and easy to modify
  - Standalone, independent on YODA or Rivet
- Can modify or swap out style file
- Same syntax on command-line as before with LaTeX-based **rivet-mkhtml**
  - PDF/envelope combinations, analysis options, selective plotting, etc.



Example output directory

- rivet-plots
  - index.html
  - **default.mplstyle**
  - ATLAS_2010_S8918562
    - d03-x01-y01.py
    - d03-x01-y01__data.py
    - d03-x01-y01.png
    - d03-x01-y01.pdf
    - index.html

Style

- Default: make plots look like the Rivet plots we all love!

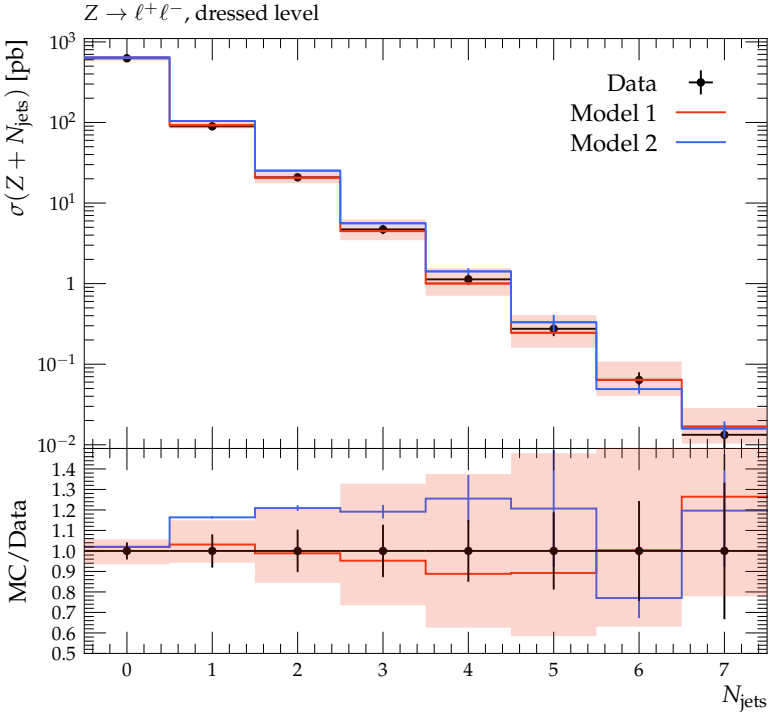
```
# font and font sizes
font.family: serif
font.serif: Palatino
mathtext.fontset: custom
mathtext.rm: Palatino
mathtext.bf: Palatino:bold
mathtext.it: Palatino:italic
mathtext.default: it
text.usetex: True
font.size: 10
axes.titlesize: 9
axes.labelsize: 10
xtick.labelsize: 8
ytick.labelsize: 8
legend.fontsize: 10
figure.titlesize:  10
lines.linewidth: 0.8
```

```
# figure layout
figure.figsize: 4.67, 4.21
figure.subplot.bottom: 0.092
figure.subplot.top: 0.934
figure.subplot.left: 0.125
figure.subplot.right: 0.968
figure.subplot.hspace: 0

# axes
axes.axisbelow: False
axes.titlepad: 7.5
axes.labelpad: 2
axes.linewidth: 0.3
xaxis.labellocation: right
lines.markersize: 2.5
axes.formatter.min_exponent: 1  # 10^0 should not be
axes.prop_cycle : cycler(color=['EE3311', '3366FF',
```

Style

- Default: dress the plots like the Rivet plots we all love!



Example output directory

- rivet-plots
  - index.html
  - default.mplstyle
  - ATLAS_2010_S8918562
    - **d03-x01-y01.py**
    - d03-x01-y01__data.py
    - d03-x01-y01.png
    - d03-x01-y01.pdf
    - index.html

Executable Python script

```python
#! /usr/bin/env python

# This Python script was auto-generated using YODA v1.9.7.
# Analysis object: /ATLAS_2010_S8918562/d03-x01-y01
# Timestamp: 21-04-2023 (17:17:08)

import matplotlib as mpl
import matplotlib.pyplot as plt
mpl.use('Agg') # comment out for interactive use
import sys, os
import numpy as np

plotDir = os.path.split(os.path.realpath(__file__))[0]

#plot style
plt.style.use(os.path.join(plotDir, '../default.mplstyle'))
# plot metadata
figW, figH = plt.rcParams['figure.figsize']
ax_xLabel = r'$\eta$'
ax_yLabel = r'$1/N_\mathrm{ev} \, \mathrm{d}N_\mathrm{ch}/\mathrm{d}\eta$'
ax_title  = r'Charged particle $\eta$ at 7$\,$TeV, track $p_\perp > 500\,\mathrm{MeV}$, for $N_\mathrm{ch} \geq 1$'
ax_xScale = 'linear'
ax_yScale = 'linear'
xLims = (-2.5, 2.5)
yLims = (0, 2.98)
# TeX-friendly labels for the legend
labels = [ r"Data", r"mc1" ]
```

dependencies

style file

plot options

Executable Python script

```
29  # Create figure and axis objects
30  fig, (ax, ratio0_ax) = plt.subplots(2, 1, sharex=True,
31                                      figsize=(figW,figH), gridspec_kw={'height_ratios': (2, 1)})
32  ratio0_ax.yaxis.set_major_locator(mpl.ticker.MultipleLocator(0.1))
33  ratio0_ax.set_ylabel('MC/Data')
34  ratio0_ax.set_ylim(0.5, 1.4999)
35  colors = {0: '#EE3311', 1: '#3366FF', 2: '#109618', 3: '#FF9900', 4: '#990099'}
36  # the numerical data is stored in a separate file
37  dataf = dict()
38  prefix = os.path.split(__file__)[0]
39  if prefix:    prefix = prefix + '/'
40  exec(open(prefix+'d03-x01-y01__data.py').read(), dataf)
41
42
43  legend_handles = [] # keep track of handles for the legend
44
45  # reference data in main panel
46  ref_curve = ax.errorbar(dataf['xpoints'], dataf['ref_yvals'],
47                          xerr=dataf['xerrs'], yerr=dataf['ref_errs'],
48                          fmt='o', ecolor = 'black', color='black', zorder=2)
49  legend_handles += [ ref_curve ]
```

hide lists with data in main .py script

Example output directory

- rivet-plots
  - index.html
  - default.mplstyle
  - ATLAS_2010_S8918562
    - d03-x01-y01.py
    - **d03-x01-y01__data.py**
    - d03-x01-y01.png
    - d03-x01-y01.pdf
    - index.html

__data.py

```python
from numpy import nan

xpoints  = [-2.45, -2.35, -2.25, -2.15, -2.05, -1.95, -1.85, -1.75, -1.65, -1.55, -1.45, -1.35, -1.25, -1.15, -1.05, -0.95, -0.85, -0.75, -0.65, -0.55, -0.45, -0.35, -0.25, -0.15, -0.05, 0.05, 0.15, 0.25, 0.35, 0.4
xedges   = [-2.5, -2.4, -2.3, -2.1999999999999997, -2.0999999999999996, -2.0, -1.9000000000000001, -1.8, -1.7, -1.6, -1.5, -1.4000000000000001, -1.3, -1.2, -1.1, -1.0, -0.9, -0.8, -0.7000000000000001, -0.6000000000
xmins    = [-2.5, -2.4, -2.3, -2.1999999999999997, -2.0999999999999996, -2.0, -1.9000000000000001, -1.8, -1.7, -1.6, -1.5, -1.4000000000000001, -1.3, -1.2, -1.1, -1.0, -0.9, -0.8, -0.7000000000000001, -0.6000000000
xmaxs    = [-2.4000000000000004, -2.3000000000000003, -2.2, -2.1, -1.9999999999999998, -1.9, -1.8, -1.7, -1.5999999999999999, -1.5, -1.4, -1.3, -1.2, -1.0999999999999999, -1.0, -0.8999999999999999, -0.7999999999999
xerrs = [
  [abs(xpoints[i] - xmins[i])   for i in range(len(xpoints))],
  [abs(xmaxs[i]   - xpoints[i]) for i in range(len(xpoints))]
]

ref_yvals = [2.196, 2.265, 2.295, 2.286, 2.334, 2.312, 2.355, 2.413, 2.431, 2.417, 2.408, 2.445, 2.471, 2.469, 2.451, 2.446, 2.451, 2.451, 2.448, 2.442, 2.438, 2.433, 2.428, 2.425, 2.422, 2.42, 2.426, 2.429, 2.43,
ref_errminus = [0.1510132, 0.1560128, 0.09102198, 0.09002222, 0.09202174, 0.09102198, 0.07102816, 0.07202777, 0.07302739, 0.07202777, 0.07202777, 0.07202777, 0.0510392, 0.05203845, 0.0510392, 0.05003998, 0.0510392,
ref_errplus = [0.1510132, 0.1560128, 0.09102198, 0.09002222, 0.09202174, 0.09102198, 0.07102816, 0.07202777, 0.07302739, 0.07202777, 0.07202777, 0.07202777, 0.0510392, 0.05203845, 0.0510392, 0.05003998, 0.0510392,
ref_errs = [ref_errminus, ref_errplus]

yvals = {
  'curve1' : [2.5357289999999977, 2.5469199999999974, 2.5536450000000009, 2.5877219999999976, 2.6051659999999974, 2.6370009999999975, 2.6304840000000036, 2.6577189999999975, 2.6571000000000033, 2.6760329999999977, 2
  'curve2' : [3.883289999999996, 3.9229909999999966, 3.952873000000014, 4.0023319999999964, 4.015687999999996, 4.053471999999997, 4.072342000000005, 4.108577999999996, 4.1172170000000055, 4.138725999999996, 4.15784
}
yedges = {
  'curve1' : [2.5357289999999977, 2.5357289999999977, 2.5469199999999974, 2.553645000000009, 2.5877219999999976, 2.6051659999999974, 2.6370009999999975, 2.6304840000000036, 2.6577189999999975, 2.6571000000000033, 2
  'curve2' : [3.883289999999996, 3.883289999999996, 3.9229909999999966, 3.952873000000014, 4.0023319999999964, 4.015687999999996, 4.053471999999997, 4.072342000000005, 4.108577999999996, 4.1172170000000055, 4.13872
}
yups = {
  'curve1' : [0.006114806620000336, 0.006128286057292033, 0.006136370262622707, 0.006177178158350294, 0.006197963375174133, 0.006235718883978074, 0.006228008509949235, 0.006260166930681636, 0.006259436875630274, 0
  'curve2' : [0.007566216359581577, 0.007604794540288376, 0.007633702902261813, 0.007681312387867056, 0.007694118533009477, 0.007730230915050332, 0.007748203017474449, 0.007782598537635845, 0.007790775956219006, 0
}
ydowns = {
  'curve1' : [0.006114806620000336, 0.006128286057292033, 0.006136370262622707, 0.006177178158350294, 0.006197963375174133, 0.006235718883978074, 0.006228008509949235, 0.006260166930681636, 0.006259436875630274, 0
  'curve2' : [0.007566216359581577, 0.007604794540288376, 0.007633702902261813, 0.007681312387867056, 0.007694118533009477, 0.007730230915050332, 0.007748203017474449, 0.007782598537635845, 0.007790775956219006, 0
}
variation_yvals = {
}


# lists for ratio plot
ratio0_ref_errminus = [0.06876739526411656, 0.0688798233995585, 0.03966099346405229, 0.0393797987751531, 0.03942662382176521, 0.03936936851211073, 0.030160577494692146, 0.029849883961873193, 0.030040061703002878, 0
ratio0_ref_errplus  = [0.06876739526411656, 0.0688798233995585, 0.03966099346405229, 0.0393797987751531, 0.03942662382176521, 0.03936936851211073, 0.030160577494692146, 0.029849883961873193, 0.030040061703002878, 0
ratio0_ref_errs     = [ratio0_ref_errminus, ratio0_ref_errplus]

ratio0_yvals = {
  'curve1' : [1.1547035519125672, 1.1547035519125672, 1.1244679911699766, 1.1126993464052326, 1.1319868766404189, 1.1161808054841462, 1.1405713667820059, 1.1169783439490462, 1.1014169084127632, 1.0930069930069943,
  'curve2' : [1.7683469945355172, 1.7683469945355172, 1.7320048565121398, 1.722384749455344, 1.7508013998250203, 1.7205175664095957, 1.7532318339100332, 1.7292322717622104, 1.7026846249481957, 1.693631016042783, 1
}
```

Summarising Rivet plot developments

- Fully functional, alternative plot machinery **rivet-mkhtml-mpl** available as of 3.1.8
  - Make default as of 3.2.x
  - We encourage everyone to start using this!

- **Your feedback, comments, suggestions, frustrations, compliments… all very welcome!**
  - Send an e-mail to rivet-support@cern.ch
  - Open an issue on Rivet Gitlab

Outlook: YODA → YODA2

- *Yet more Objects for Data Analysis*

- Complete rewrite using C++17 compile-time features
- Introduce BinnedStorage class that can support custom bin-content types
  - Generalised to arbitrary dimensionality
  - Coherent treatment of overflow and infinity-binning
  - Fillable object types (histograms, profiles) to allow tracing of fill index for on-the-fly estimates of bin-to-bin correlations
  - Introduces the “Estimate” as a new inert binned-content type that is to represent (HEPData) measurements with associated uncertainty breakdowns (i.e. MC predictions and reference data now *both* binned objects, unlike YODA1)
- Introduces concept of discrete axes, solves many issues with HEPData compatibility
- Improved support for (de-)serializing analysis objects in HPC applications relying on MPI communication

Contur

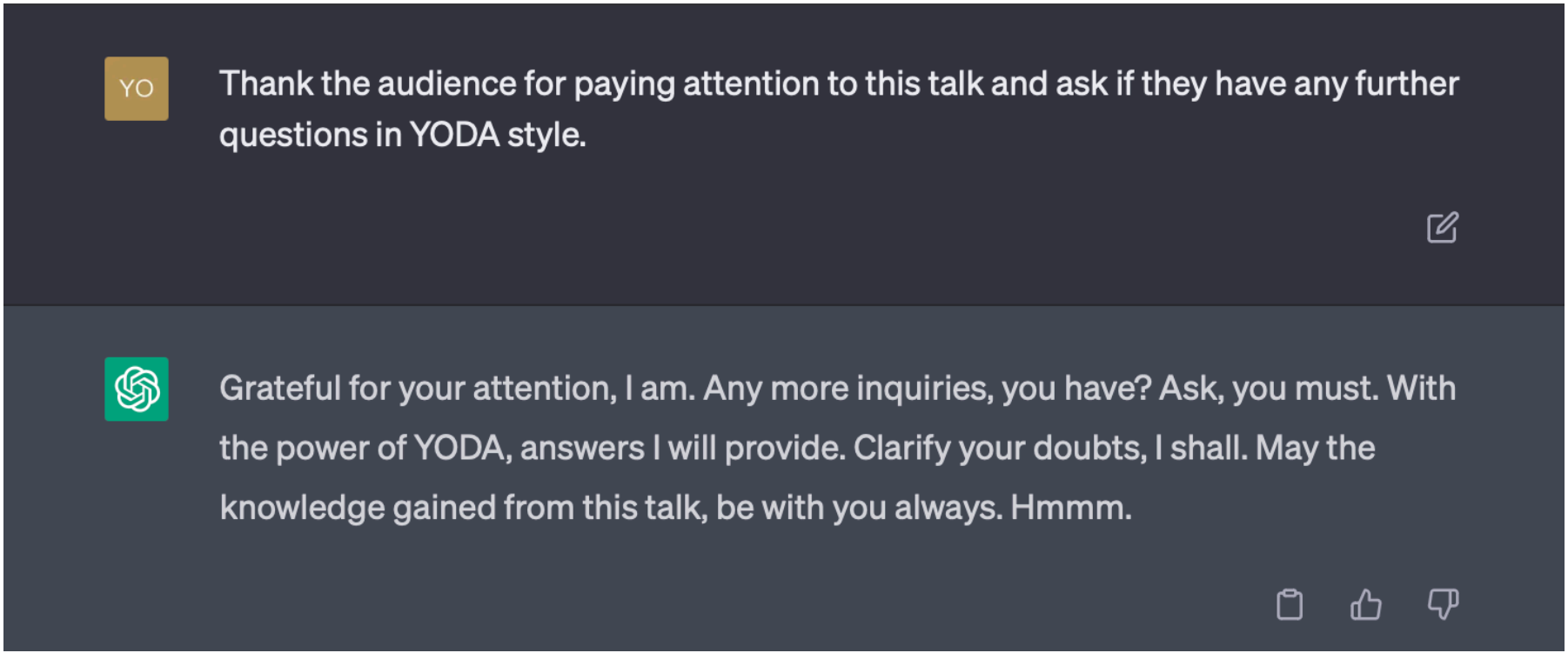
- *Constraints on New Theories Using Rivet*

- Upcoming release 2.4.2 and current 2.4.1:
  - Added Rivet routines
  - Updated SM theory treatment
  - Signal plus background mode
  - Internal code tidying, updated docs
- Features for a future release:
  - mpl-based backend for **contur-mkhtml**
  - Signal strength scans

[(Re)interpretation of the LHC results for New Physics] @ Durham, 29 Aug - 1 Sep

- Some recent results:
  - Include Contur in global GAMBIT fit constraints on electroweakinos in presence of light gravitino [2303.09082]
  - Testing the scalar triplet solution to CDF's heavy W problem at the LHC [Phys.Rev.D 107 (2023) 7, 075020]

Thanks for listening! Questions?



Backup

New analyses in Rivet 3.1.8

- Hundreds of hadron decay analyses
  - ALICE
  - ATLAS
  - BABAR
  - BELLE
  - BESII