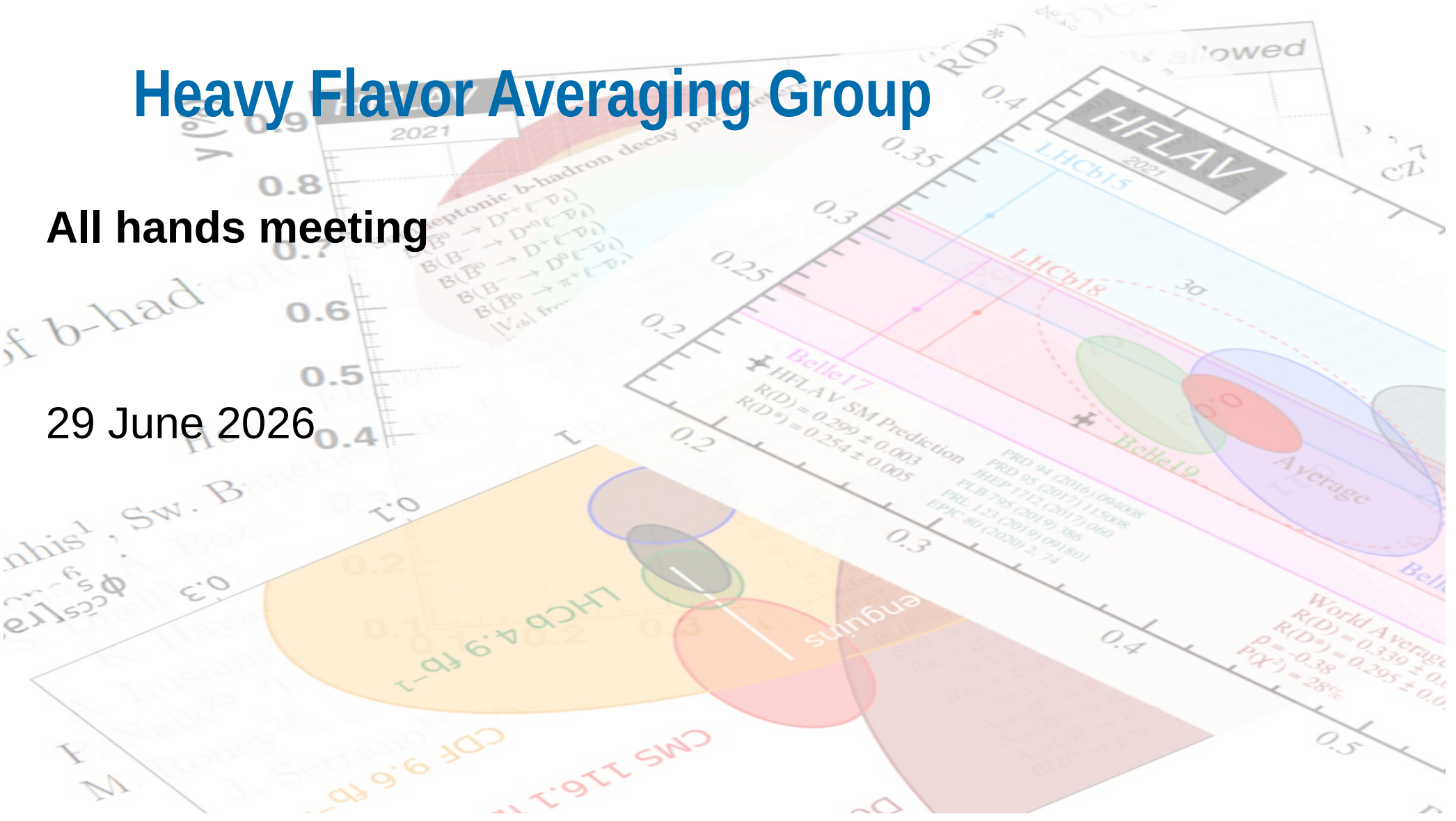
Heavy Flavor Averaging Group

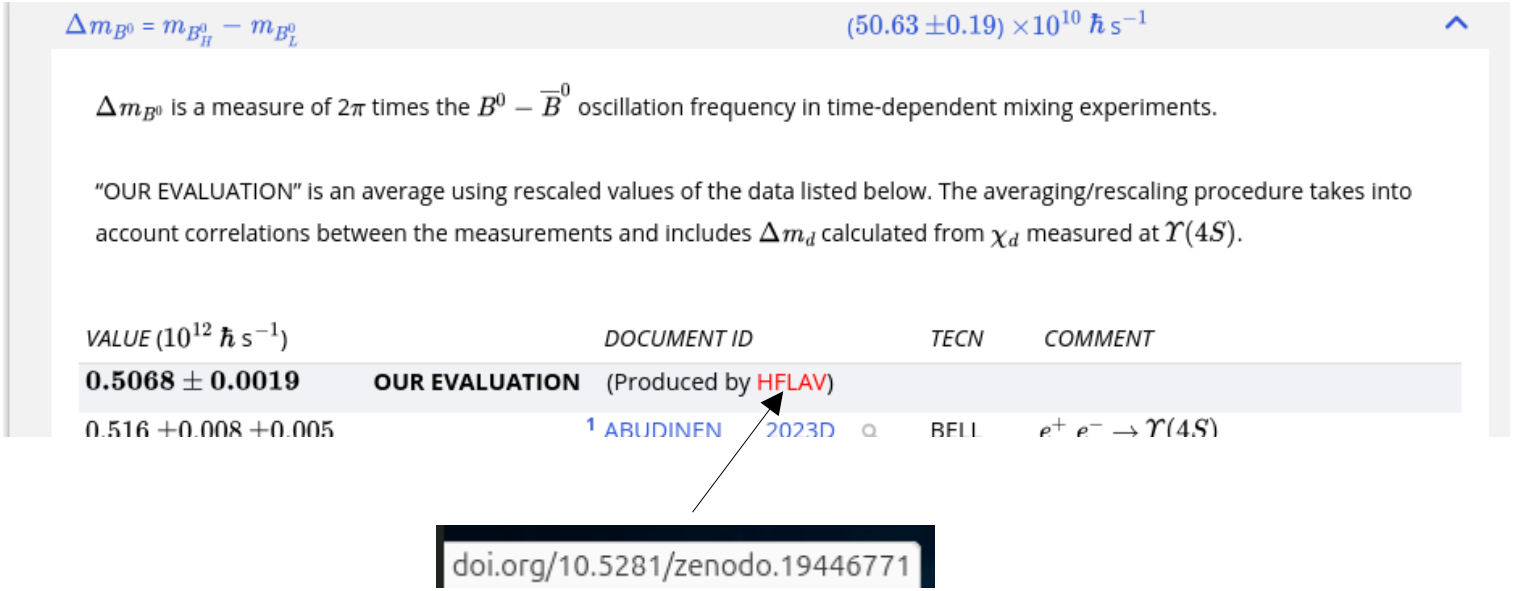
All hands meeting

29 June 2026



PDG 2026

- We now have specific references to the Zenodo records



- No longer ambiguous how results were obtained.

Zenodo records

- Records for PDG results were made available by all groups except “charm oscillations and CPV”.
  - For October we need to get ALL groups and ALL results.
- Records are of a good quality and are already cited

arXiv:2606.20797v1 [hep-ex]

Measurement of mixing-induced CP violation in the decay $B^0 \to \pi^0\pi^0$

The decay $B^0 \to J/\psi K^0_S$ provides a high-yield, low-background control channel, with time-dependent CP-violating coefficients $S_{J/\psi K^0_S}$ and $C_{J/\psi K^0_S}$ known with high precision [29], and therefore offers a stringent validation of the $t_{\rm tag}$-based approach.

[29] Amhis, Y.S., *et al.*: Averages of b-hadron, c-hadron, and τ-lepton properties as of 2021. Phys. Rev. **D107**, 052008 (2023). With specific results from doi:10.5281/zenodo.19371140

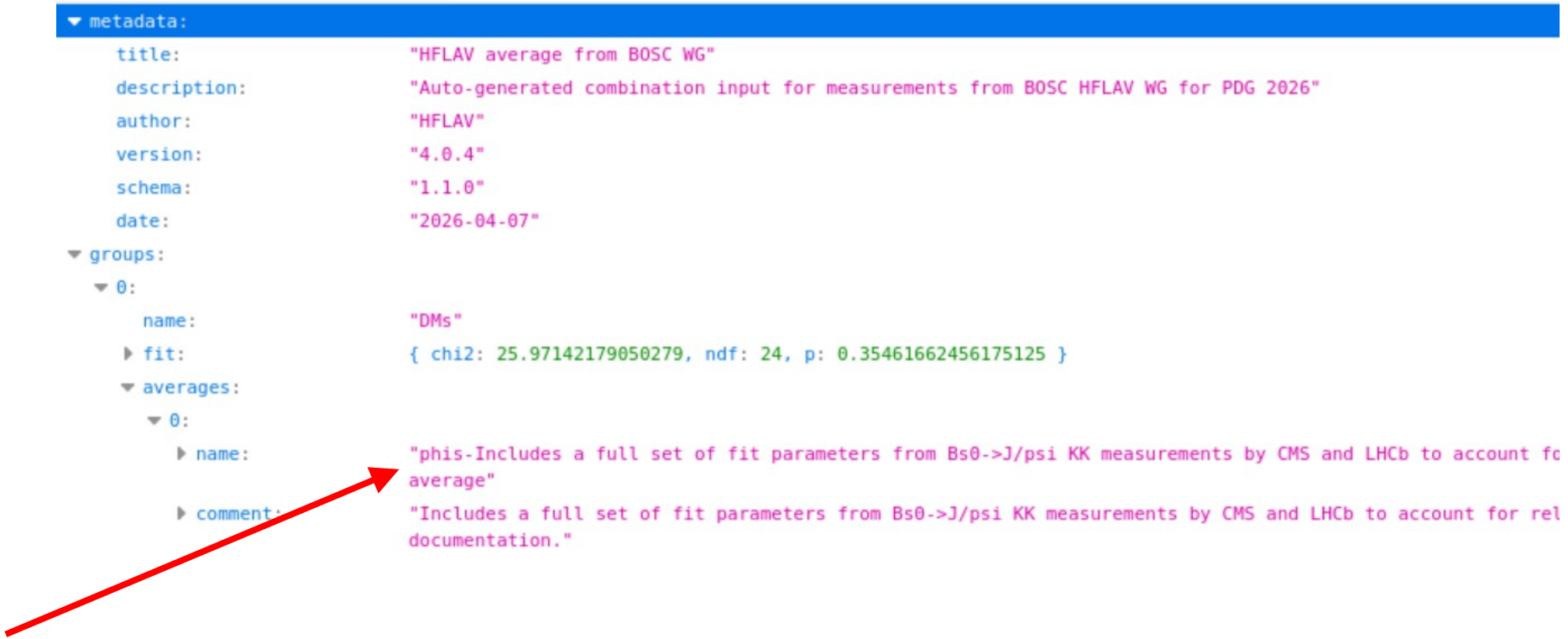
Issues with Zenodo records

- Still incomplete so not all HFLAV results are there
- Naming convention is unclear in places
- No API advertised
- Not browser friendly and no 1-to-1 link to website
- Link to PDG can be unclear

MONASH University

Conventions

- The name for measurements should be short



MONASH University

Clarity of parameter naming

- Merge request from Olivier to extend naming convention.
- For charm oscillations and CPV, some further expansion required.
- Most likely other things missing as well.

The possible parameter types are

- `BR` : branching fraction
- `ACP` : CP asymmetry
- `Azero` , `Aperp` , `Apara` : polarization amplitudes
- `Azero2` , `Aperp2` , `Apara2` : modulus square of the polarization amplitudes
- `Deltaperp` , `Deltapara` : relative polarization amplitude phases
- `fL` , `fperp` , `fpara` : polarization fractions
- `AA` : generic type for angular analyses (the exact name of the parameter can be added using a custom string, see the end of this page)
- `IA` : isospin asymmetry
- `M` : mass
- `LT` : average lifetime (or effective lifetime if daughters are specified)
- `G` : total decay width Γ
- `DG` : decay-width difference $\Delta\Gamma = \Gamma_{\rm light} - \Gamma_{\rm heavy}$ between the mass eigenstates of a neutral heavy-flavour meson system
- `DM` : mass difference $\Delta m = m_{\rm heavy} - m_{\rm light}$ between the mass eigenstates of a neutral heavy-flavour meson system
- `y` : mixing parameter $y = \frac{\Delta\Gamma}{2\Gamma}$
- `x` : mixing parameter $x = \frac{\Delta m}{\Gamma}$
- `chi` : time-integrated mixing probability $\chi = \frac{x^2+y^2}{2(x^2+1)}$
- `|q..p|` : parameter $|q/p|$ describing CP violation in B mixing, where $p = \langle B^0 | B_{\rm light}\rangle$ and $q = \langle \bar{B}^0 | B_{\rm light}\rangle$; note that $|q/p| = \sqrt[4]{\frac{1-A_{\rm SL}}{1+A_{\rm SL}}}$
- `ASL` : parameter $A_{\rm SL} = \frac{|p/q|^2-|q/p|^2}{|p/q|^2+|q/p|^2} = \frac{1-|q/p|^4}{1+|q/p|^4}$ describing CP violation in B mixing (so-called "semileptonic asymmetry")
- `ReepsB..` : parameter $\mathrm{Re}(\epsilon_B)/(1+|\epsilon_B|^2)$ describing CP violation in B mixing, where $\epsilon_B = \frac{p-q}{p+q}$; note that $q/p = \frac{1-\epsilon_B}{1+\epsilon_B}$ and $\mathrm{Re}(\epsilon_B)/(1+|\epsilon_B|^2) \simeq A_{\rm SL}/4$
- `phi` : CP-violating phase ϕ arising from the interference between B mixing and decay
- `beta` : CP-violating phase $-\phi/2$ arising from the interference between B mixing and decay

Particles (and not anti-particles) are used for mother particles, following the PDG convention.

Daughter particles are ordered by descending mass and then within an isospin multiplet by descending (absolute) charge (`pi+` before `pi-` before `pi0`) or particle before anti-particle.

The name of anti-particles that are written with a bar in latex is given by the name of the particle with `bar` added, e.g. `pbar` , `K0bar` , `Lambdac-bar` .

MONASH University

Updates to schema

- Add “comment” field for a fit (not just for parameters of fit)
- Make “latexname” mandatory
- Add an internal HFLAV identifier for all averages. Formatting to be agreed.
- Make “PDGcode” mandatory – so an explicit None is required if not possible.

MONASH University

HFLAV API

- There is now an API that can be installed from PyPi
  - `python -m pip install hflav-fair-client`

Example 1: Load a specific record by ID and navigate the information

```
from hflav_fair_client.services.service import Service

service = Service()

# Load a specific file from a known Zenodo record
data = service.load_data_file(
  record_id=19446771,
)

# Use attribute-chain navigation to list groups
for g in data.groups:
    print(g.name)

# Use attribute-chain navigation to list names and averages in the group
#   'alpha_CKM'
[f'{a.name} : {a.average.value.central}±{a.average.value.uncertainty}' for a in data.groups(name='ASLd and ASLs').averages]
```

```
['ASLd : -0.0021±0.0017', 'ASLs : -0.0006±0.0028']
```

HFLAV API

- There is now an API that can be installed from PyPi
  - `python -m pip install hflav-fair-client`

Example 2: Get the bibtex record for citing these results

Building on the data object from the previous example

```
bibtex = data.reference.bibtex
print(bibtex)
```

```
@article{HeavyFlavorAveragingGroupHFLAV:2024ctg:19446771,
    author = "Banerjee, Sw. and others",
    collaboration = "Heavy Flavor Averaging Group (HFLAV)",
    title = "{Averages of b-hadron, c-hadron, and {\ensuremath{\tau}}-lepton properties as of 2023}",
    eprint = "2411.18639",
    archivePrefix = "arXiv",
    primaryClass = "hep-ex",
    doi = "10.1103/x87q-tld5",
    journal = "Phys. Rev. D",
    volume = "113",
    number = "1",
    pages = "012008",
    year = "2026",
    note = "{with specific result from \href{https://doi.org/10.5281/zenodo.19446771}{{\texttt{doi:10.5281/zenodo.19446771}}}}"
}
```


MONASH University

HFLAV API

- There is now an API that can be installed from PyPi
  - `python -m pip install hflav-fair-client`

Example 3: Search within the loaded record

```
# Search within the loaded data for averages using a specific paper
from hflav_fair_client.models.hflav_data_searching import HflavDataSearching, SearchOperators

searcher = HflavDataSearching(data)
results = searcher.get_data_object_from_key_and_value(
    object_name="groups",
    key_name="doi",
    operator=SearchOperators.EQUALS,
    value="10.1103/PhysRevD.86.072009",
)
[r.name for r in results]
```

```
['ASLd and ASLs']
```


MONASH University