# EOS – A Software for Flavour Physics Phenomenology

SMEFT-Tools – Zurich – 16/09/2022

**Méril Reboud**

*For the EOS authors*

Technische Universität München

# The EOS Software

**EOS:**

- **Main use cases**:

  **1) infer theory parameters** from an extendable database of experimental and theoretical likelihoods;

  2) produce **publication-quality theory predictions** for flavour observables;

  3) produce **high-quality Monte Carlo samples** of flavour processes, e.g. for sensitivity studies.

- **C++** back-end, **python** front-end (Jupyter Notebook)
- Current version: **EOS v1.0.5**
- Developed since ~1000BC by **over 20 contributors**

https://eos.github.io/

# The EOS Software

### How does EOS compare to other software?

https://flav-io.github.io/

http://superiso.in2p3.fr/

FlavBit: 1705.07933

(among others...)

EOS' "unique selling point":

- Simultaneous **inference of hadronic and BSM parameters**
- **Modularity of hadronic matrix elements** (models, parametrizations...)
- **Production of pseudo-events** for use in sensitivity studies and in preparation for experimental measurements
- Prediction of **hadronic matrix elements** from QCD sum rules

# 1. Brief overview

# Installation

- **Installation is very simple** on Linux:

  pip3 install --user eoshep

- Then open a **Jupyter Notebook** and just run

  import eos

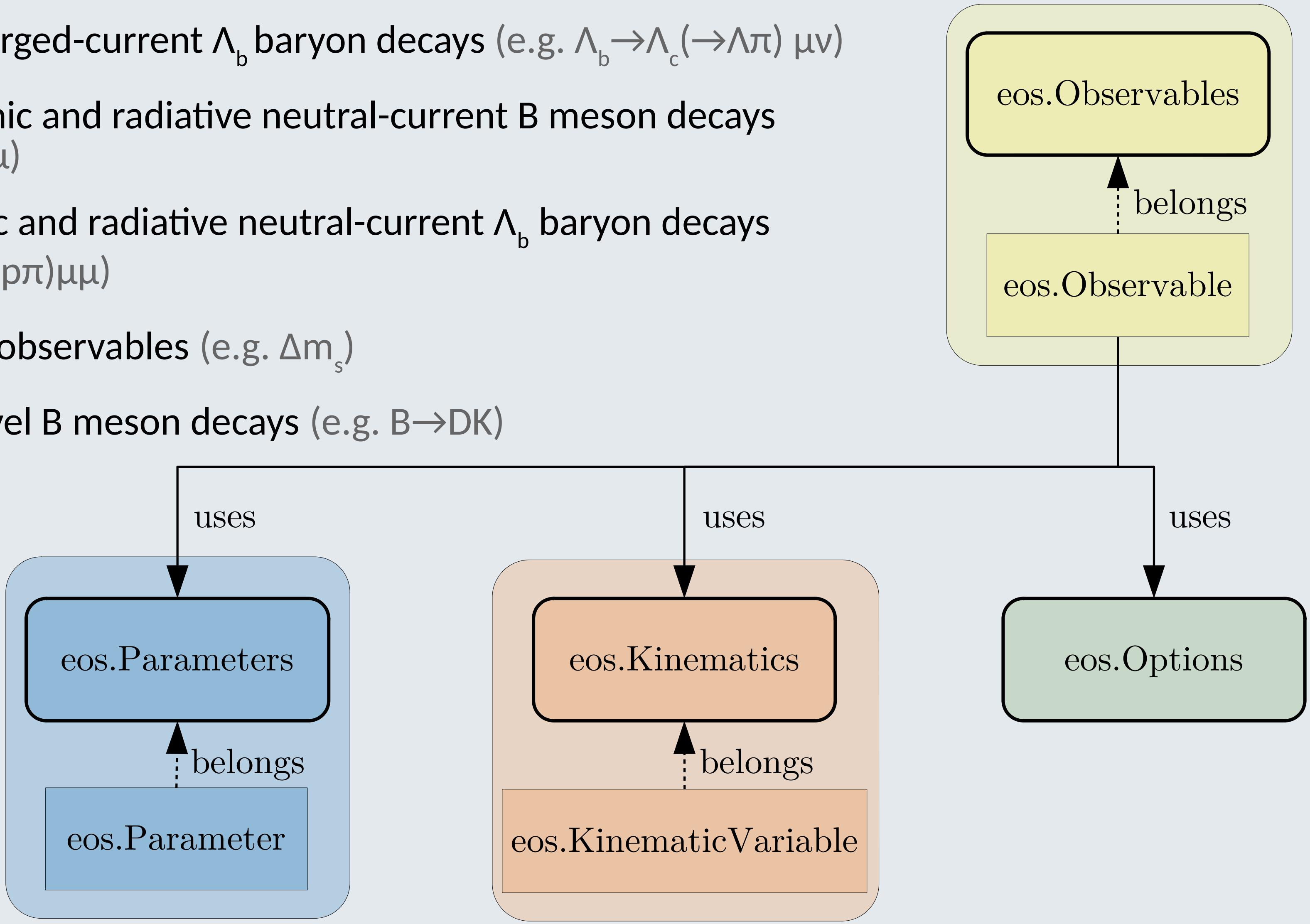
- For other platforms and **further details:**
  https://eos.github.io/doc/installation.html

- **Support** can be found here:
  https://discord.com/invite/hyPu7f7K6W

# Observables and versatility

**~1000 (pseudo-)observables covering:**

- (semi)leptonic charged-current B meson decays (e.g. $B \to D^* \tau\nu$)
- semileptonic charged-current $\Lambda_b$ baryon decays (e.g. $\Lambda_b \to \Lambda_c(\to \Lambda\pi)\,\mu\nu$)
- rare (semi)leptonic and radiative neutral-current B meson decays (e.g. $B \to K^*\mu\mu$)
- rare semileptonic and radiative neutral-current $\Lambda_b$ baryon decays (e.g. $\Lambda_b \to \Lambda(\to p\pi)\mu\mu$)
- B-meson mixing observables (e.g. $\Delta m_s$)
- hadronic tree-level B meson decays (e.g. $B \to DK$)



# Predictions and Uncertainties

EOS

In [3]:
```python
eos.Observable.make('B->Dlnu::BR',
                    eos.Parameters.Defaults(),
                    eos.Kinematics(q2_min=0.02, q2_max=11.60),
                    eos.Options(l = 'mu')
                   )
```
executed in 69ms, finished 18:23:54 2021-11-20

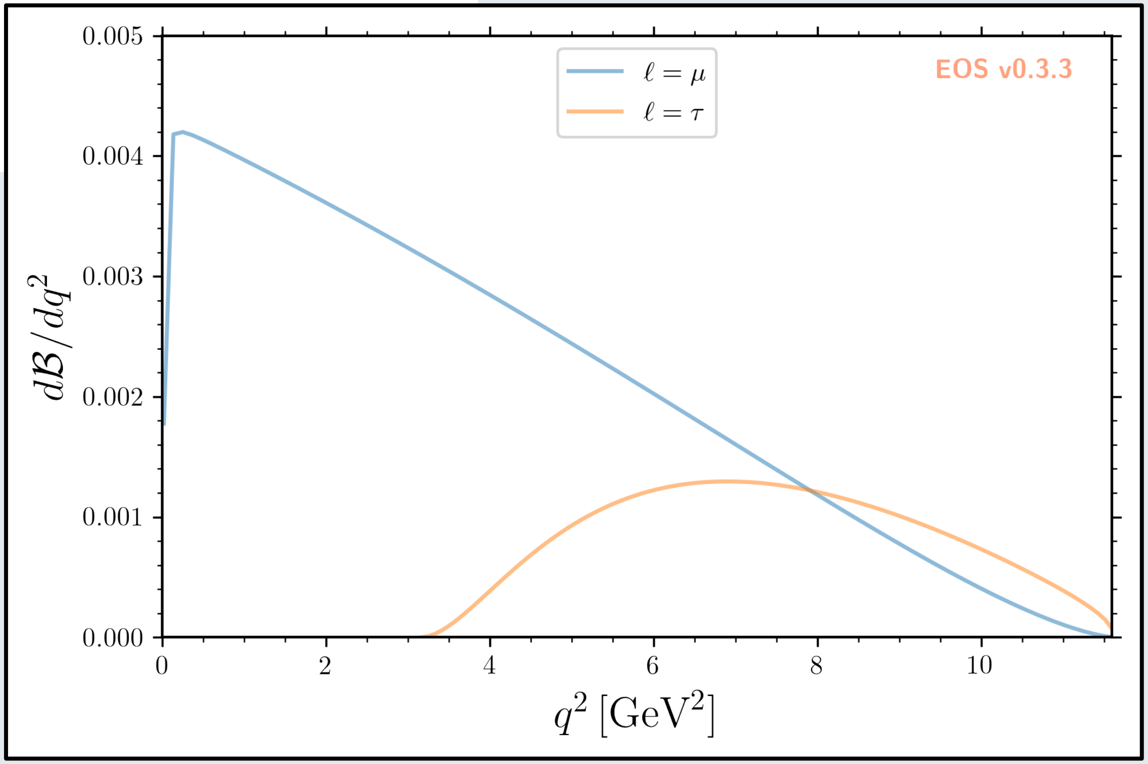
Out[3]:

| B->Dlnu::BR | (eos.Observable) | |
|---|---|---|
| kinematics | q2_min | 0.02 |
| | q2_max | 11.6 |
| options | I | 1/2 |
| | U | c |
| | l | mu |
| current value | 0.02417 | |

**Fast evaluation of observables:**

- Multi-threading
- Observable cache

**Versatile plotting framework** based on matplotlib




EOS Repo: Tutorial 1

# Predictions and Uncertainties

**Theory uncertainties** are estimated using **Monte Carlo techniques**, specifically importance sampling technique using **pypmc** (https://pypmc.github.io/)

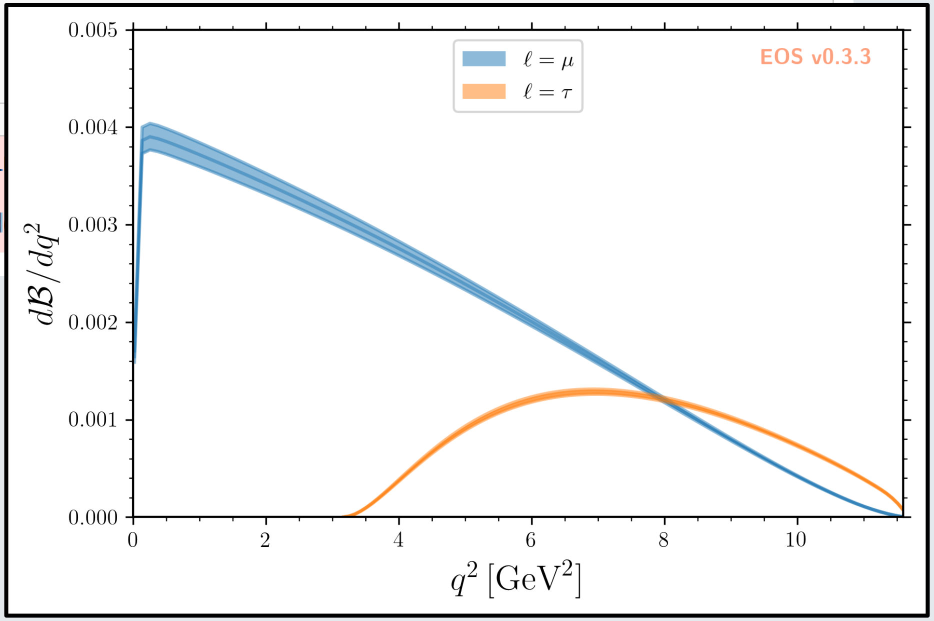
```
In [7]: analysis_args = {
            'priors': [
                { 'parameter': 'B->D::alpha^f+_0@BSZ2015', 'min':  0.0, 'max':  1.0, 'type': 'uniform' },
                { 'parameter': 'B->D::alpha^f+_1@BSZ2015', 'min': -5.0, 'max': +5.0, 'type': 'uniform' },
                { 'parameter': 'B->D::alpha^f+_2@BSZ2015', 'min': -5.0, 'max': +5.0, 'type': 'uniform' },
                { 'parameter': 'B->D::alpha^f0_1@BSZ2015', 'min': -5.0, 'max': +5.0, 'type': 'uniform' },
                { 'parameter': 'B->D::alpha^f0_2@BSZ2015', 'min': -5.0, 'max': +5.0, 'type': 'uniform' }
            ],
            'likelihood': [
                'B->D::f_++f_0@HPQCD:2015A',
                'B->D::f_++f_0@FNAL+MILC:2015B'
            ]
        }
        analysis = eos.Analysis(**analysis_args)
```

executed in 725ms, finished 19:18:37 2021-11-20

```
INFO:EOS:Creating analysis with 5 priors, 2 EOS-wide constrai
nd 0 fixed parameters.
INFO:EOS:likelihood probably depends on 3 parameter(s) that d
```

EOS Repo: Tutorial 1

**Markov Chains**



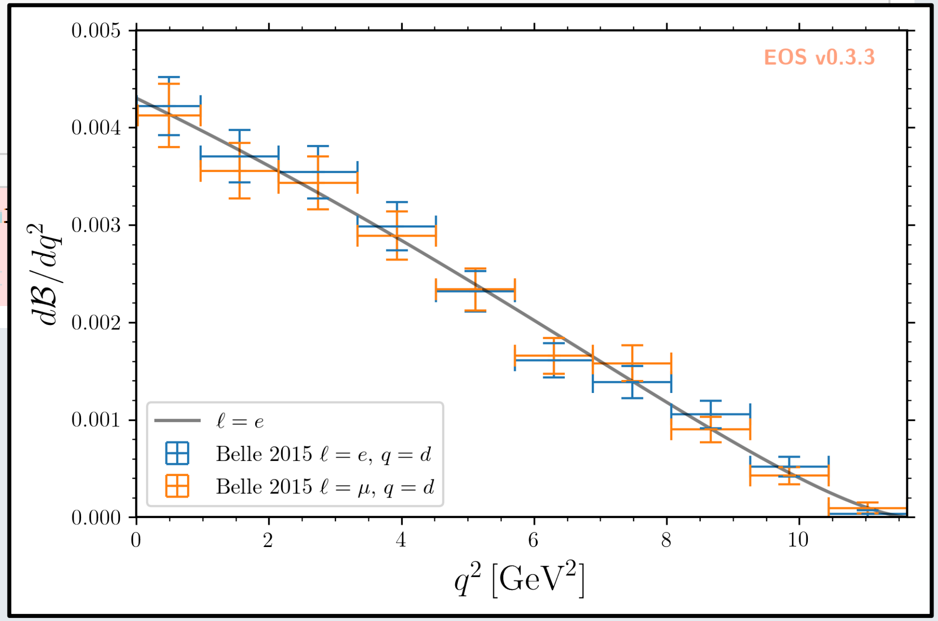
# Parameter Inference

**Parameters** can be inferred from a **database of experimental or theoretical constraints**

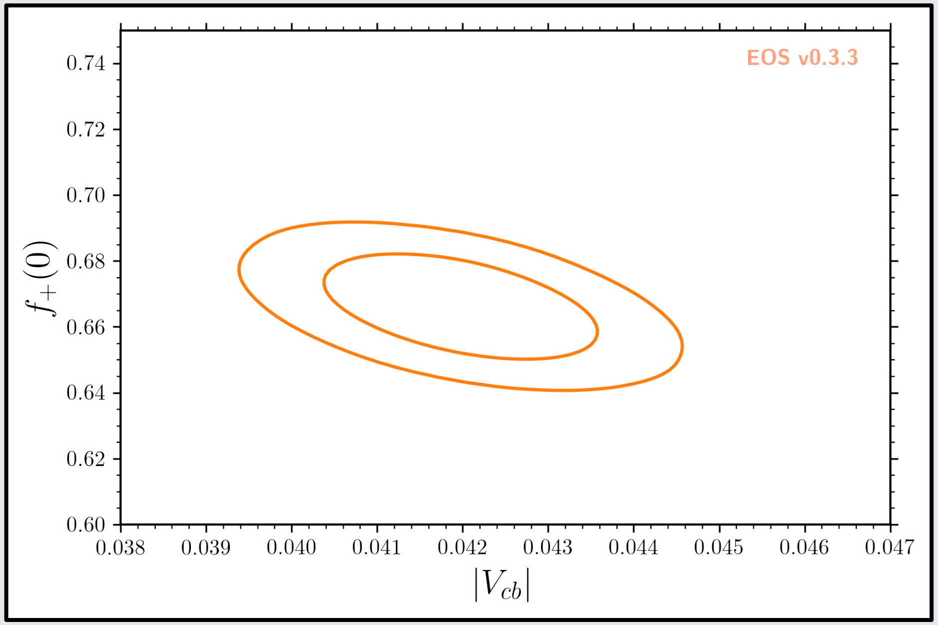
```
In [5]: analysis_args = {
            'global_options': { 'form-factors': 'BSZ2015', 'model': 'CKM' },
            'priors': [
                { 'parameter': 'CKM::abs(V_cb)',           'min':  38e-3, 'max':  45e-3 , 'type': 'uniform'},
                { 'parameter': 'B->D::alpha^f+_0@BSZ2015', 'min':  0.0,   'max':  1.0   , 'type': 'uniform'},
                { 'parameter': 'B->D::alpha^f+_1@BSZ2015', 'min': -4.0,   'max': -1.0   , 'type': 'uniform'},
                { 'parameter': 'B->D::alpha^f+_2@BSZ2015', 'min': +4.0,   'max': +6.0   , 'type': 'uniform'},
                { 'parameter': 'B->D::alpha^f0_1@BSZ2015', 'min': -1.0,   'max': +2.0   , 'type': 'uniform'},
                { 'parameter': 'B->D::alpha^f0_2@BSZ2015', 'min': -2.0,   'max':  0.0   , 'type': 'uniform'}
            ],
            'likelihood': [
                'B->D::f_++f_0@HPQCD:2015A',
                'B->D::f_++f_0@FNAL+MILC:2015B',
                'B^0->D^+e^-nu::BRs@Belle:2015A',
                'B^0->D^+mu^-nu::BRs@Belle:2015A'
            ]
        }
        analysis = eos.Analysis(**analysis_args)
        analysis.parameters['CKM::abs(V_cb)'].set(42.0e-3)

        executed in 310ms, finished 11:23:40 2021-11-21

        INFO:EOS:Creating analysis with 6 priors, 4 EOS-wide constra
        ... parameters.
        ...elihood probably depends on 48 parameter(s) that
```

EOS Repo: Tutorial 2

SciPy optimization



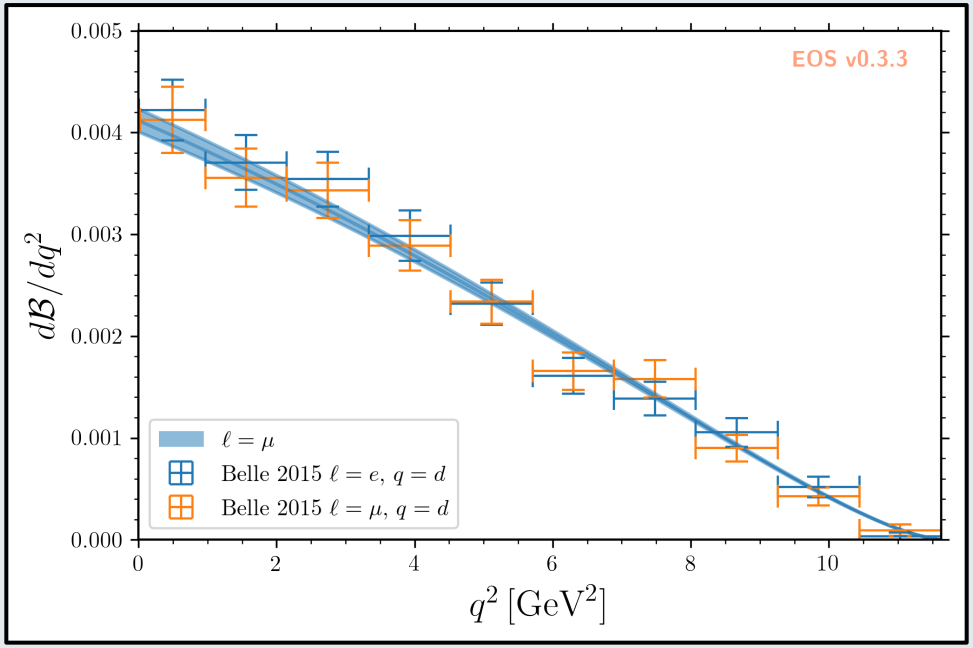
# Parameter Inference




Output of the sampling are **genuine python objects**

$|V_{cb}| = 0.0420$ +/- 0.0009

**Publication-quality plots**



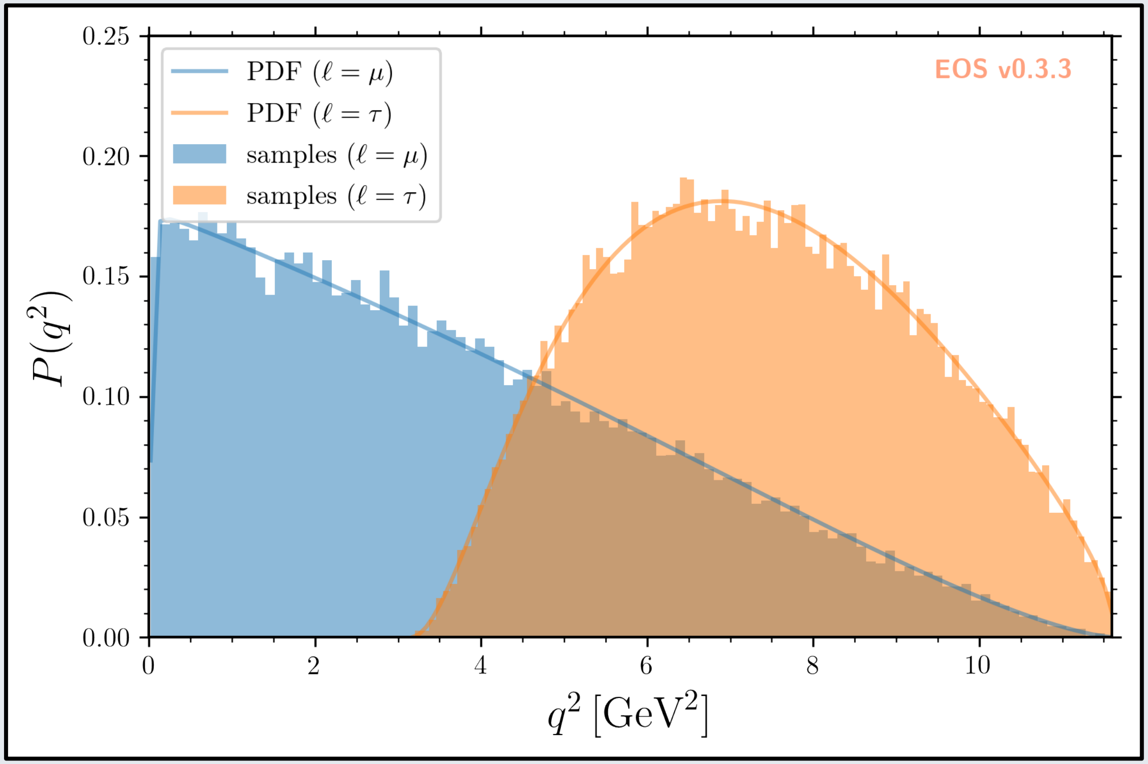

EOS Repo: Tutorial 2

# Simulation of Pseudo Events

Event simulation from a set of **built-in PDFs** using Markov chain Monte Carlo techniques.

→ **Sensitivity studies**

**Excellent matching** between produced samples and built-in PDFs




EOS Repo: Tutorial 3

# 2. Applications

See https://eos.github.io/publications/

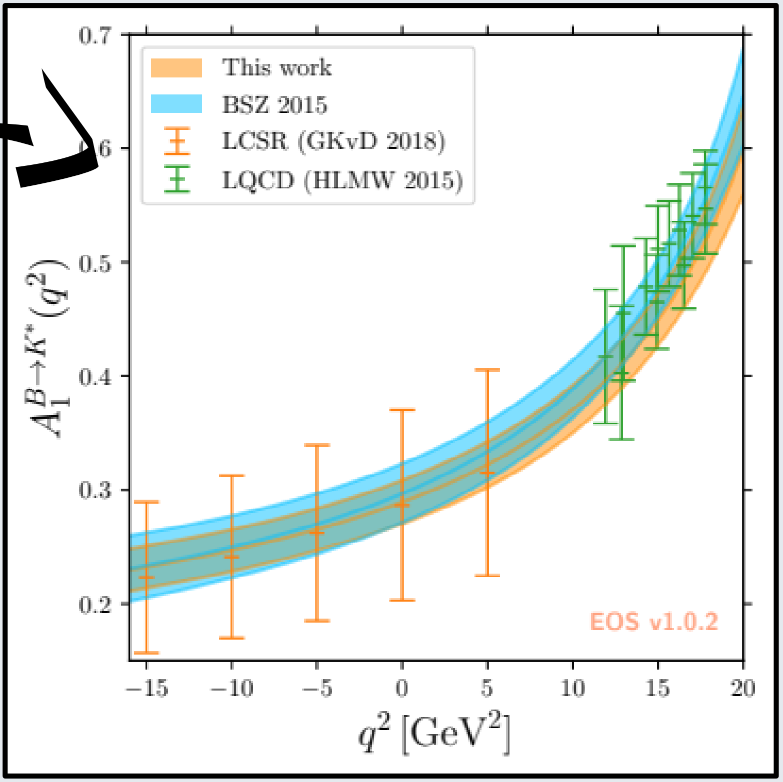
# Multi-step fit

From *Improved Theory Predictions and Global Analysis of Exclusive b→sμ+μ- Processes* (2206.03797)
[N. Gubernari, MR, D. van Dyk, J. Virto]

1) Fit **local** form-factors of $B \to K^{(*)}$, $B_s \to \varphi$ (46 parameters)

GKvD:2018A **Database of references**

Gubernari, N. and Kokulu, A. van Dyk, D. – $B \to P$ and $B \to V$ Form Factors from $B$-Meson Light-Cone Sum Rules beyond Leading Twist – arXiv:1811.00983



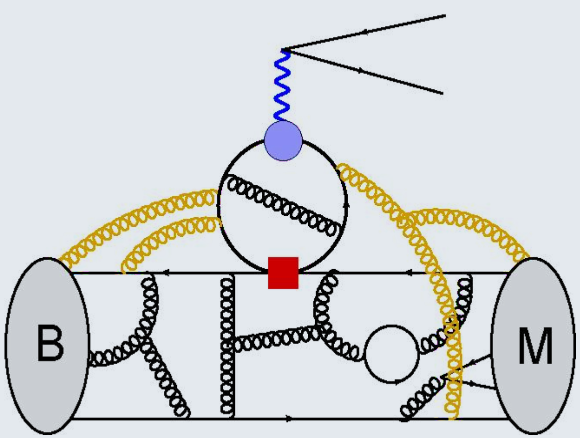
# Multi-steps fits

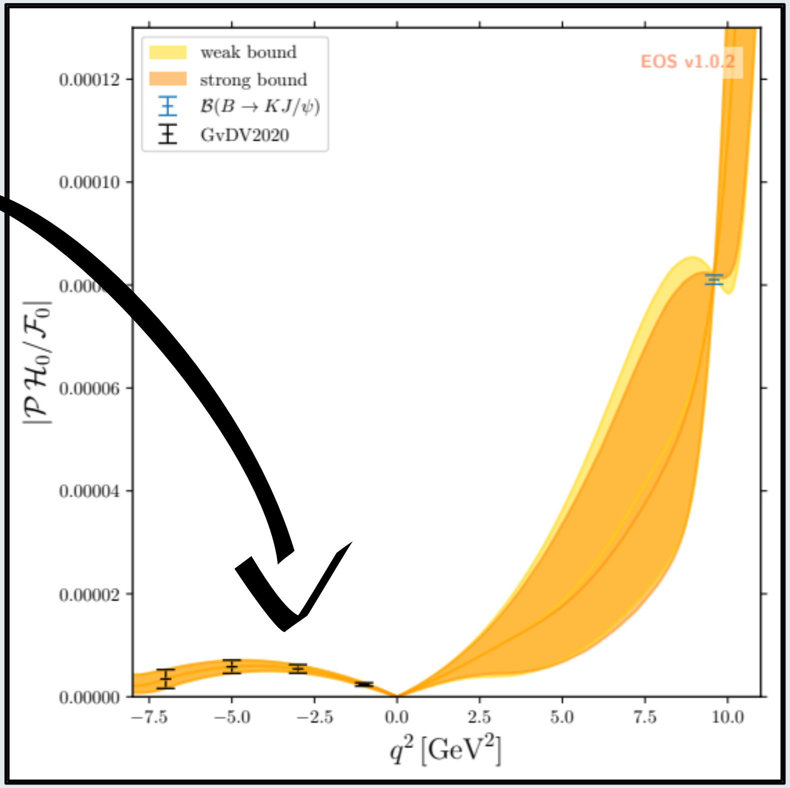
From *Improved Theory Predictions and Global Analysis of Exclusive b→sμ+μ- Processes* (2206.03797)

[N. Gubernari, MR, D. van Dyk, J. Virto]

1) Fit **local** form-factors of $B \to K^{(*)}$, $B_s \to \varphi$ (46 parameters)

2) Use step 1 to **constrain** the **non-local** form factors (charm-loops)



# Multi-steps fits

From *Improved Theory Predictions and Global Analysis of Exclusive b→sμ+μ- Processes* (2206.03797)
[N. Gubernari, MR, D. van Dyk, J. Virto]

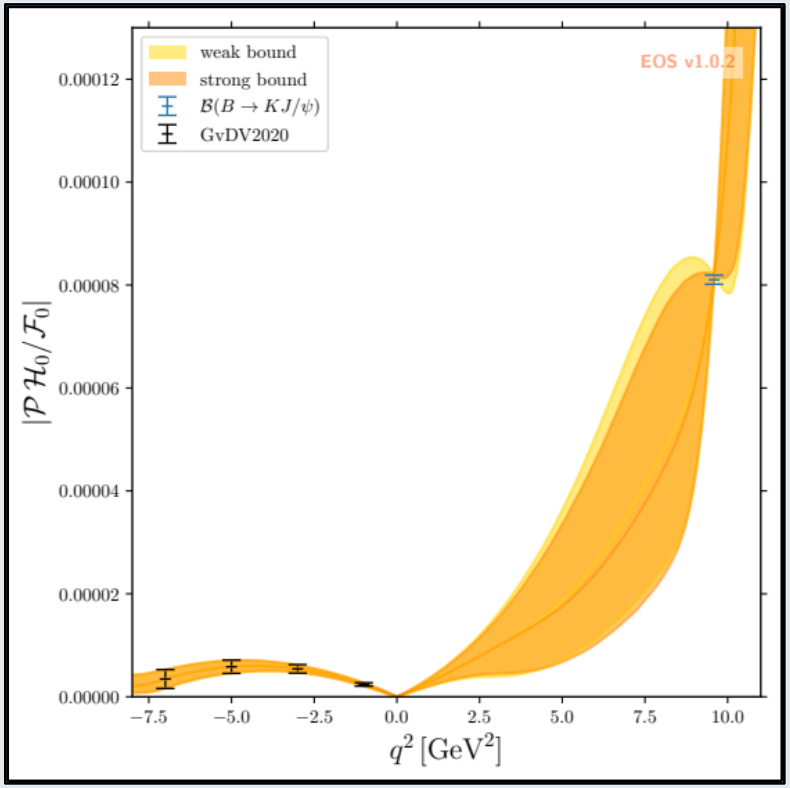
1) Fit **local** form-factors of $B \to K^{(*)}$, $B_s \to \varphi$ (46 parameters)

2) Use step 1 to **constrain** the **non-local** form factors (charm-loops)

**3) Fit** them in a new parametrization (84 parameters)

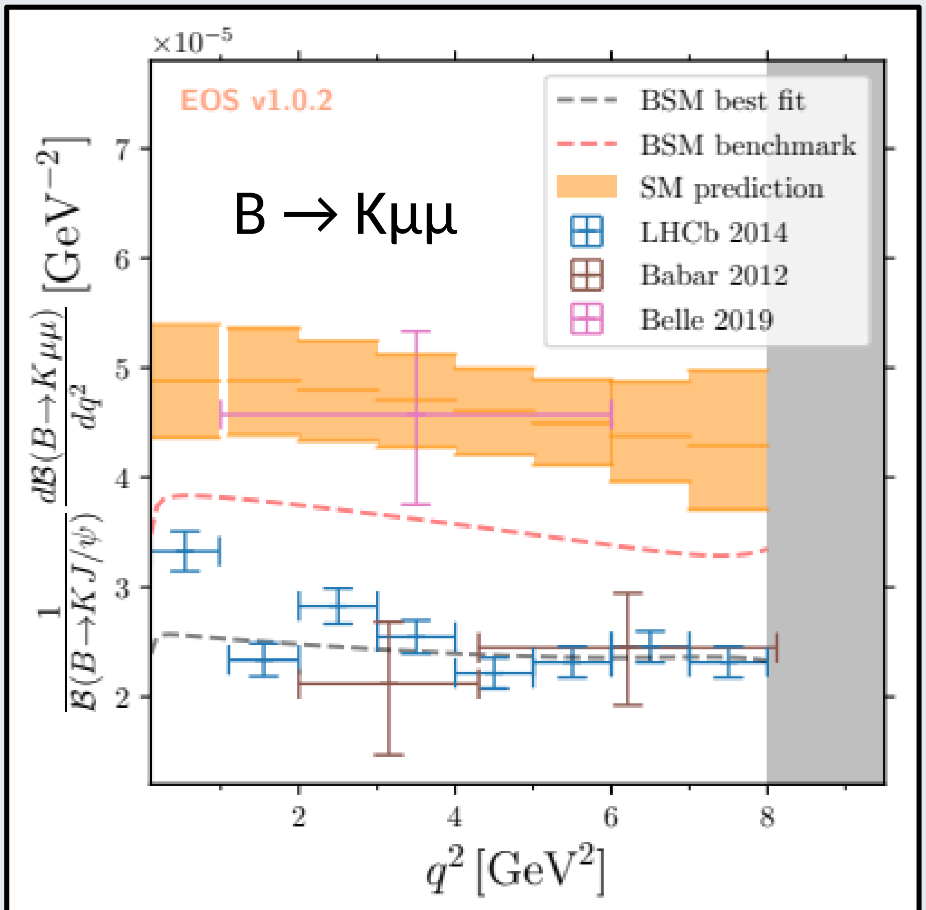
→ **130 nuisance parameters**

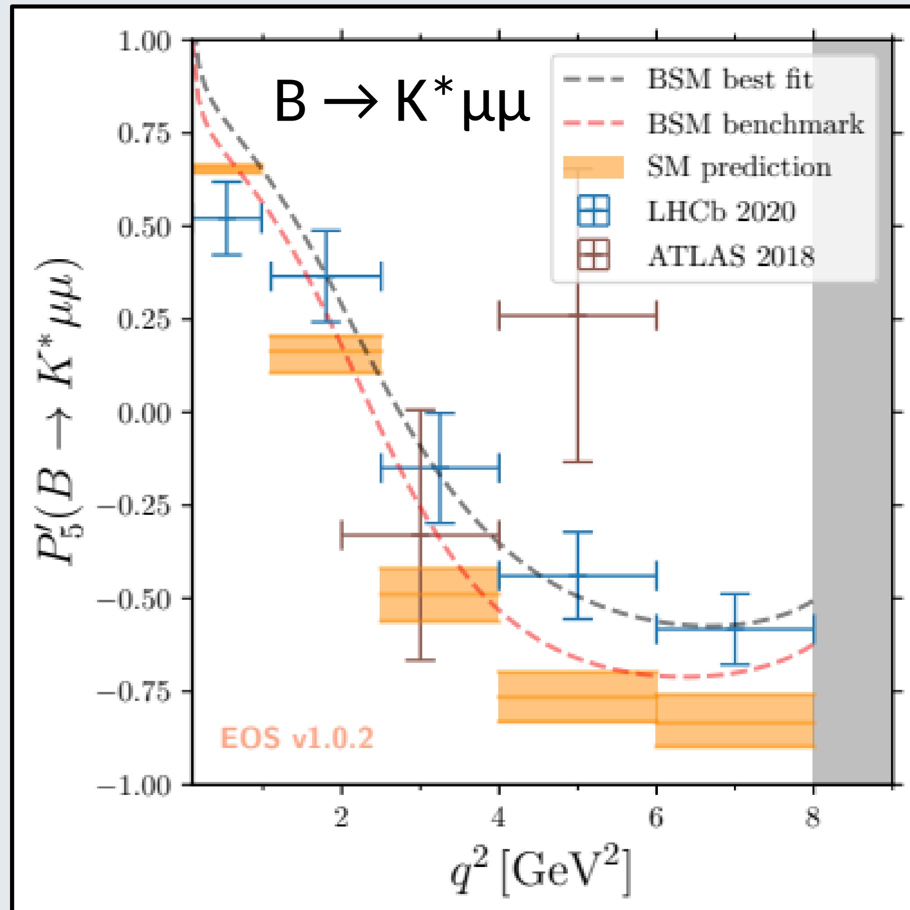
→ ‘Proof of concept’ **fit to the WET’s WC**



# Summary plots

From *Improved Theory Predictions and Global Analysis of Exclusive b→sμ+μ- Processes* (2206.03797)
[N. Gubernari, MR, D. van Dyk, J. Virto]



# WET and SMEFT

From *The Flavor of UV Physics* 2101.07273
[S. Bruggisser, R. Schäfer, D. van Dyk, S. Westhoff]

- **Simplify observables** to keep only the parameters of interest:

$$\mathcal{B}(B_s \to \mu^+\mu^-) \times 10^9 = \left[3.57 - 1.71\,\mathcal{C}_{10} + 0.21\,\mathcal{C}_{10}^2\right] \times \left(1 \pm 1.2\%\big|_{f_{B_s}} \pm 1.5\%\big|_{\mathrm{CKM}}\right),$$

Wilson polynomial

Other uncertainties

$$\mathcal{B}(B \to X_s\gamma) \times 10^4 = \left[3.26 - 15.17\,\mathcal{C}_7 - 0.77\,\mathcal{C}_8 + 1.66\,\mathcal{C}_7\mathcal{C}_8 + 1.36\,\mathcal{C}_{7'}\mathcal{C}_{8'} + 18.03\,(\mathcal{C}_7^2 + \mathcal{C}_{7'}^2) + 0.20\,\mathcal{C}_8^2 + 0.09\,\mathcal{C}_{8'}^2\right] \times (1 \pm 5\%),$$

- Part of **WCxf** (https://wcxf.github.io/assets/pdf/WET.EOS.pdf)

  → Simple **connection to SMEFT tools**

# 3. EOS' Future

# Ongoing Work

- Improve **statistical tools**
  - preparing for global analyses with **O(100) nuisance parameters**
  - interfacing to `dynesty` to make full use of **nested sampling** algorithm https://dynesty.readthedocs.io

- Constantly adding **new observables**, **form-factor parametrizations**, ...

# User friendly pre-analysis

- "**Pre-packaged**" low-energy analyses
  - **Remove nuisance parameters** to reduce complexity
  - Disseminate as a **Gaussian Mixture Model** of the WET coefficients

    → Do the hard work once :)
  - **Interface to wilson and smelli** for ease of use by model-building community

# Conclusion

- **EOS is open source**, the current version is **v1.0.5**
  - **~1000 (pseudo-)observables**, **~1000 parameters**, **~500 constraints**
  - Development on **github** https://github.com/eos/eos/
  - Used in **~35 theory papers** and many experimental papers; part of Belle II external software
- **Online documentation** and nice tutorials: https://eos.github.io/doc/
- We are happy to discuss how to **add further observables**

**Thank you!**