Universität Zürich[UZH]

DsixTools: The SMEFT toolkit

Javier Fuentes-Martín
University of Zurich

Manual: arXiv:1704.04504

Website: https://dsixtools.github.io/

Outline of the talk

Part I: DsixTools

[Celis, JF, Vicente, Virto, 1704.04504]

1. **The package:** What is DsixTools and what it can do for you
2. **Basics:** Modules, installation and some usage examples

Part II: Future directions and developments

I will focus in something that DsixTools cannot do for you… yet

→ Integrating out heavy particles at one loop: A simplified framework

[JF, Portoles, Ruíz-Femenía, 1607.02142]

The Standard Model Effective Theory (SMEFT)

SMEFT $\equiv$ Effective Field Theory with SM fields and symmetries

$$\mathcal{L}_{\rm SMEFT} = \mathcal{L}_{\rm SM} + \frac{1}{\Lambda}\mathcal{L}_5 + \frac{1}{\Lambda^2}\mathcal{L}_6 + \frac{1}{\Lambda^3}\mathcal{L}_7 + \dots$$

$$\mathcal{L}_d = \sum_i \mathcal{C}_i \, O_i^d$$

$\mathcal{C}_i$: Wilson coefficients (WC)

O_i^d: Effective operators

✓ Any effect of New Physics above Λ (with a linearly realised Higgs) can be mapped into a set of Wilson coefficients of the SMEFT

✓ Model independent and systematic approach to New Physics

The SMEFT: operators and WCs

$$\mathcal{L}_{\rm SMEFT} = \mathcal{L}_{\rm SM} + \frac{1}{\Lambda}\mathcal{L}_5 + \frac{1}{\Lambda^2}\mathcal{L}_6 + \dots$$

ν Majorana masses

Weinberg PRL 43 (1979) 1566

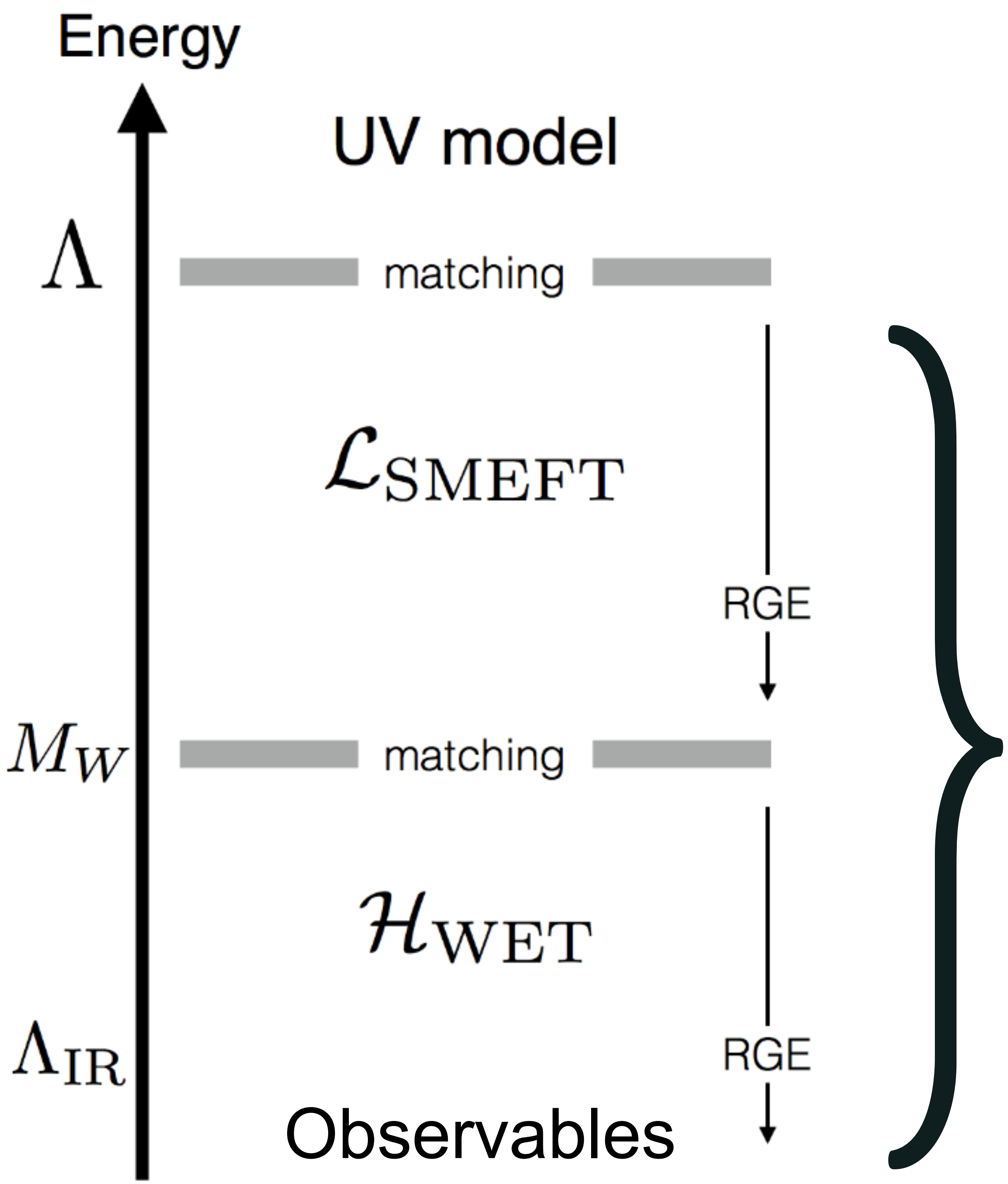
Leading deviations from the SM

but there are 2499 (real) WCs!

Buchmüller, Wyler, Nucl. Phys. B268 (1986) 621
Grzadkowski et al., 1008.4884

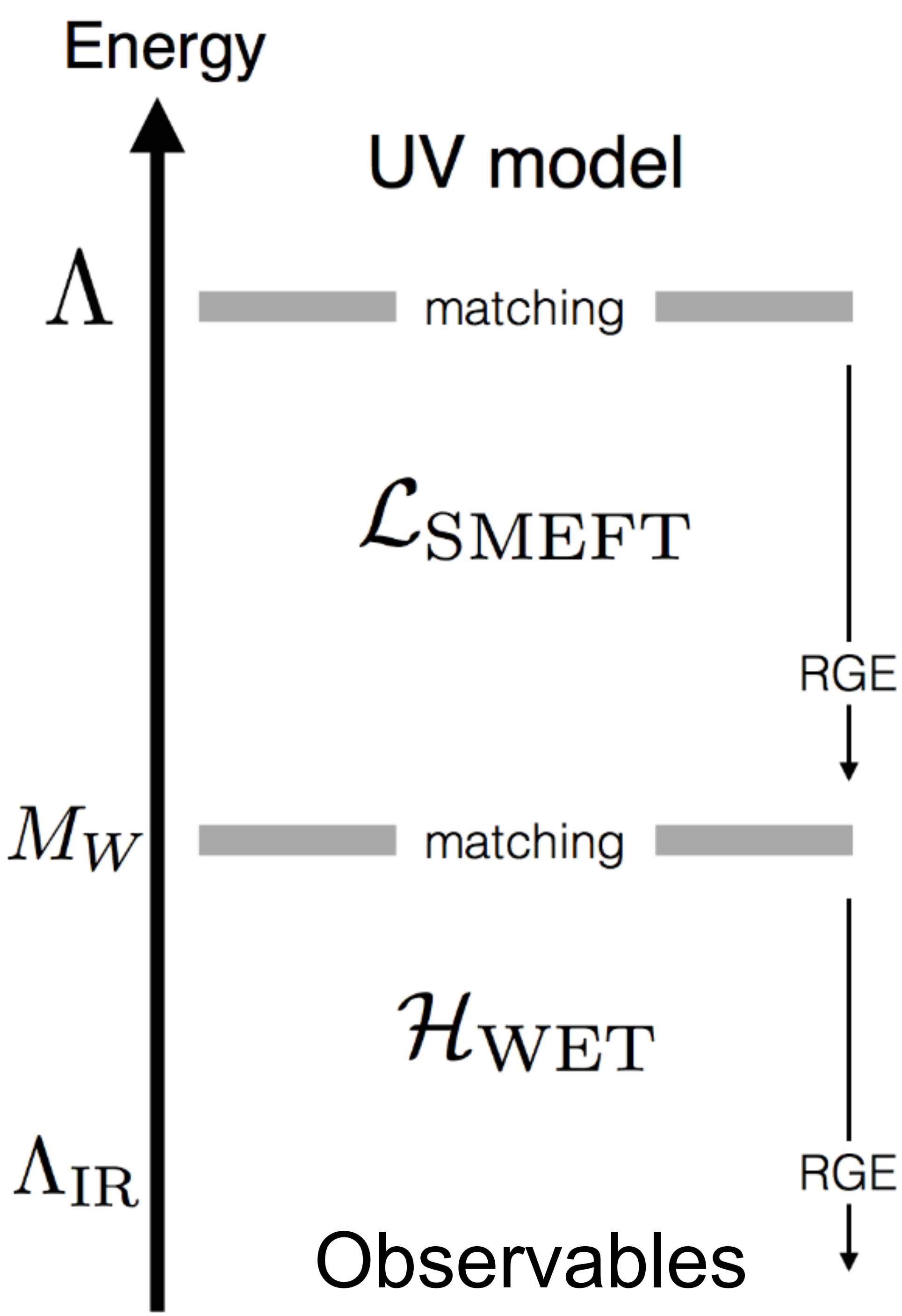
- Imposition of restrictions on the WCs becomes (almost) mandatory
  $\Longrightarrow$ However these restrictions often translate into a loss of generality
- In NP models precise correlations among WCs are expected
- A (simple) systematic framework to handle any UV model in terms of the SMEFT is highly desirable!

The top-down approach in the SMEFT



Once and for all!

Recent progress with the SMEFT



More later (part II)

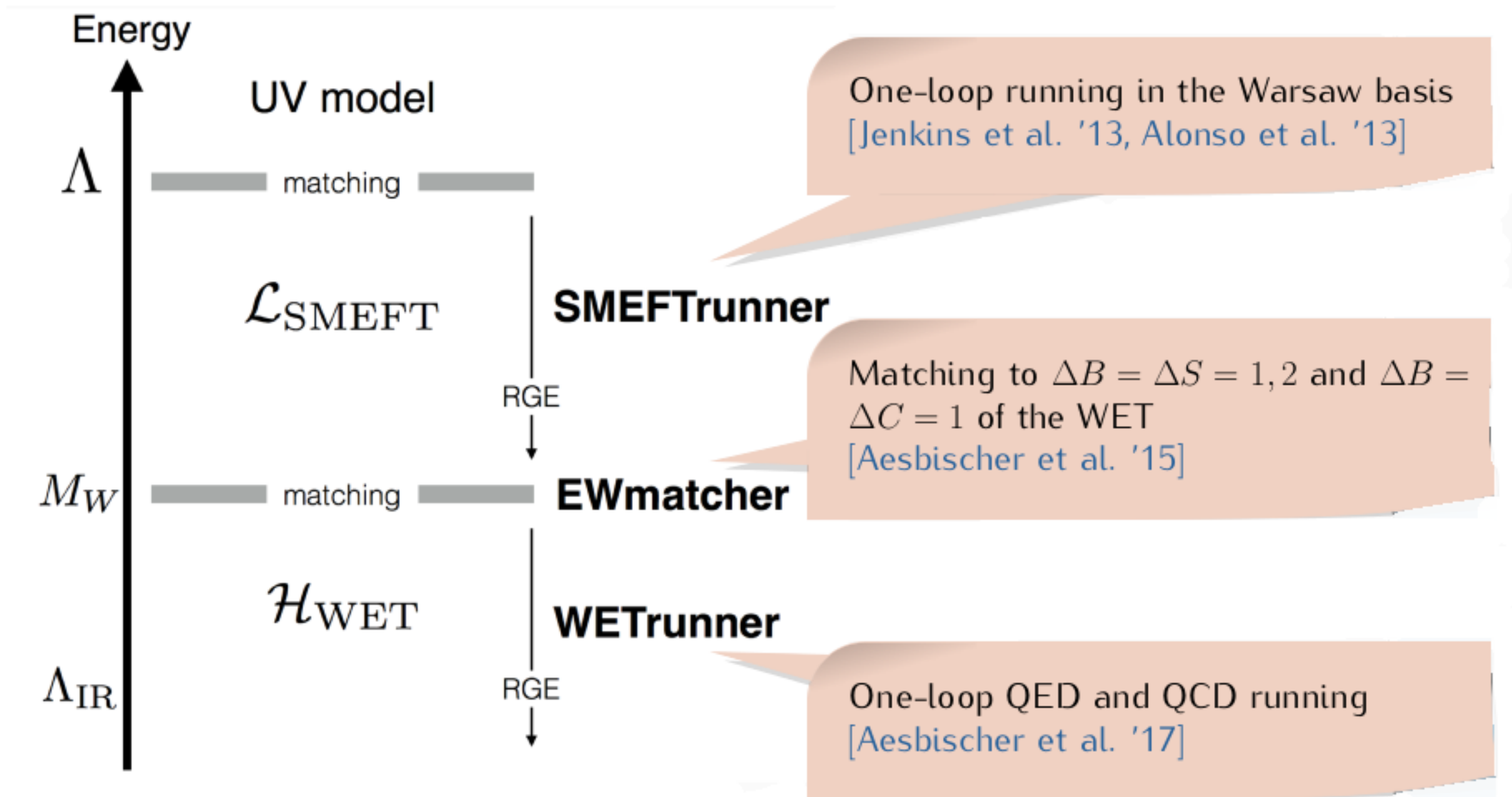
Jenkins et al. 1308.2627, 1310.4838,
Alonso et al. 1312.2014

Aebischer et al. 1512.02830
Jenkins et al. 1709.04486

Aebischer et al. 1704.06639
González-Alonso et al. 1706.00410
Jenkins et al. 1711.05270

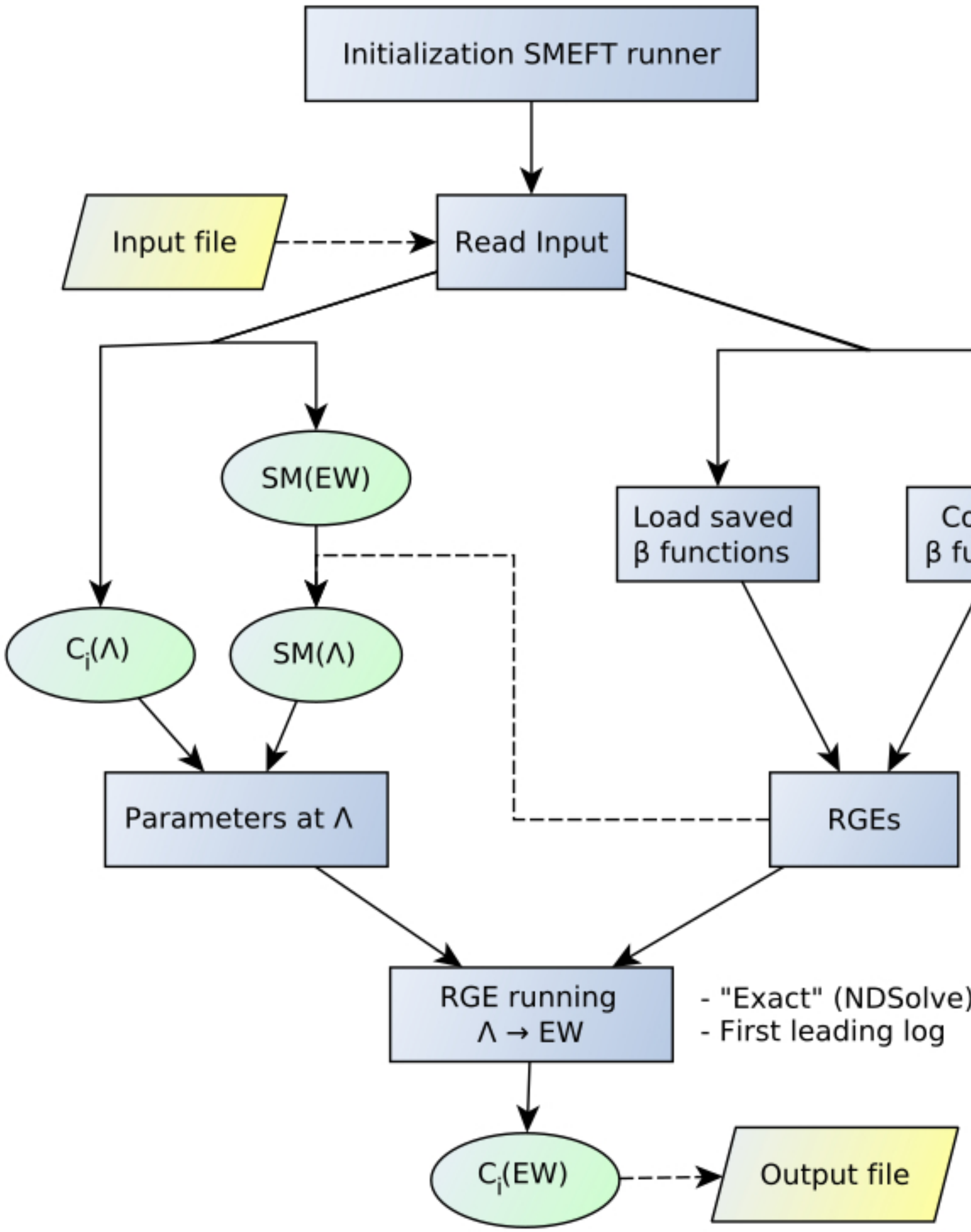
DsixTools: main structure

DsixTools is a **Mathematica** package for the handling of $d \leq 6$ SMEFT operators



Modular structure: Each module can be used independently

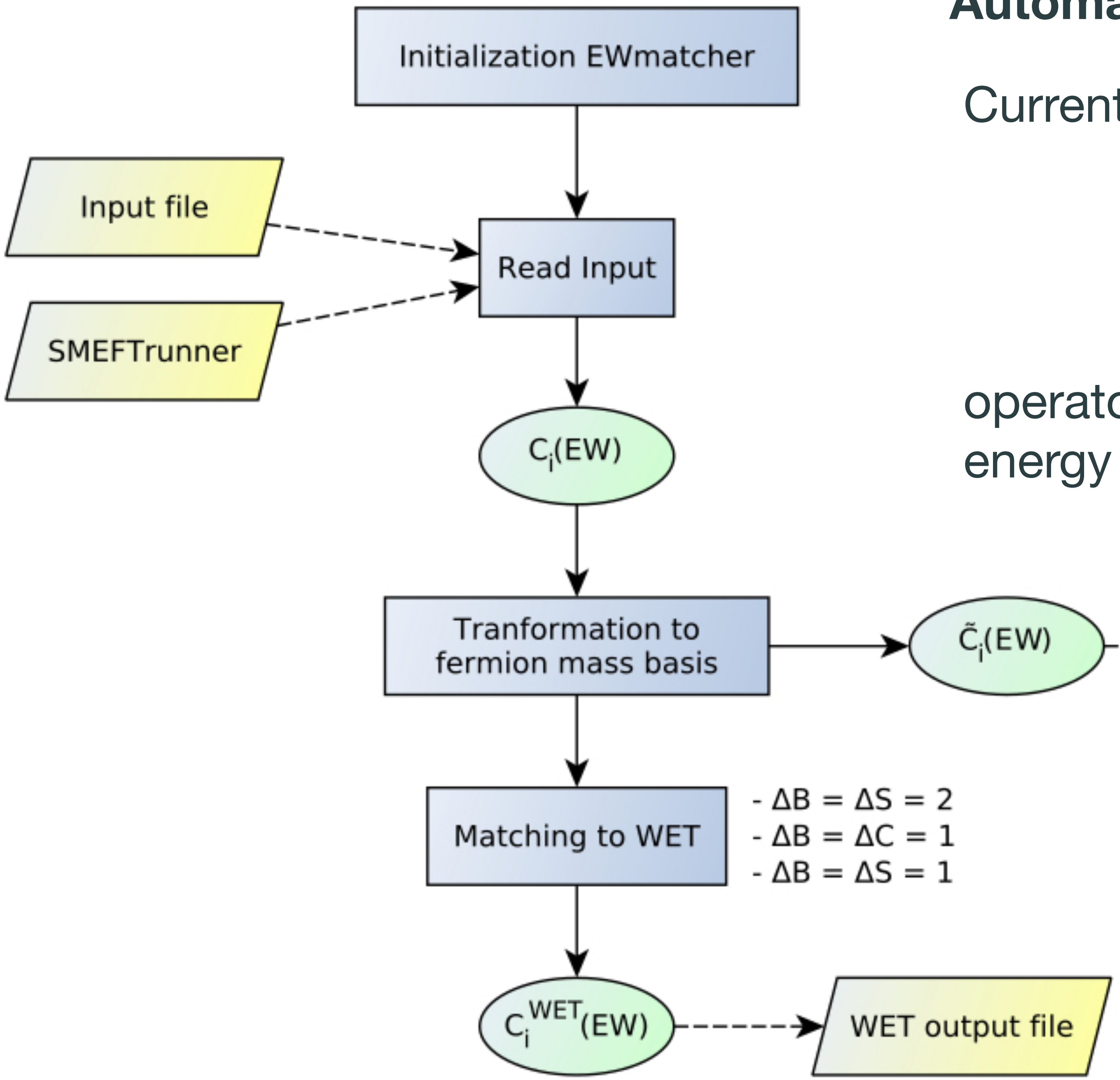
DsixTools modules: SMEFTrunner



Automatic RGE running in the SMEFT

Based on the results in Jenkins et al. 1308.2627, 1310.4838, Alonso et al. 1312.2014

DsixTools modules: EWmatcher



Automatic matching SMEFT ⟶ WET

Currently, matching only to

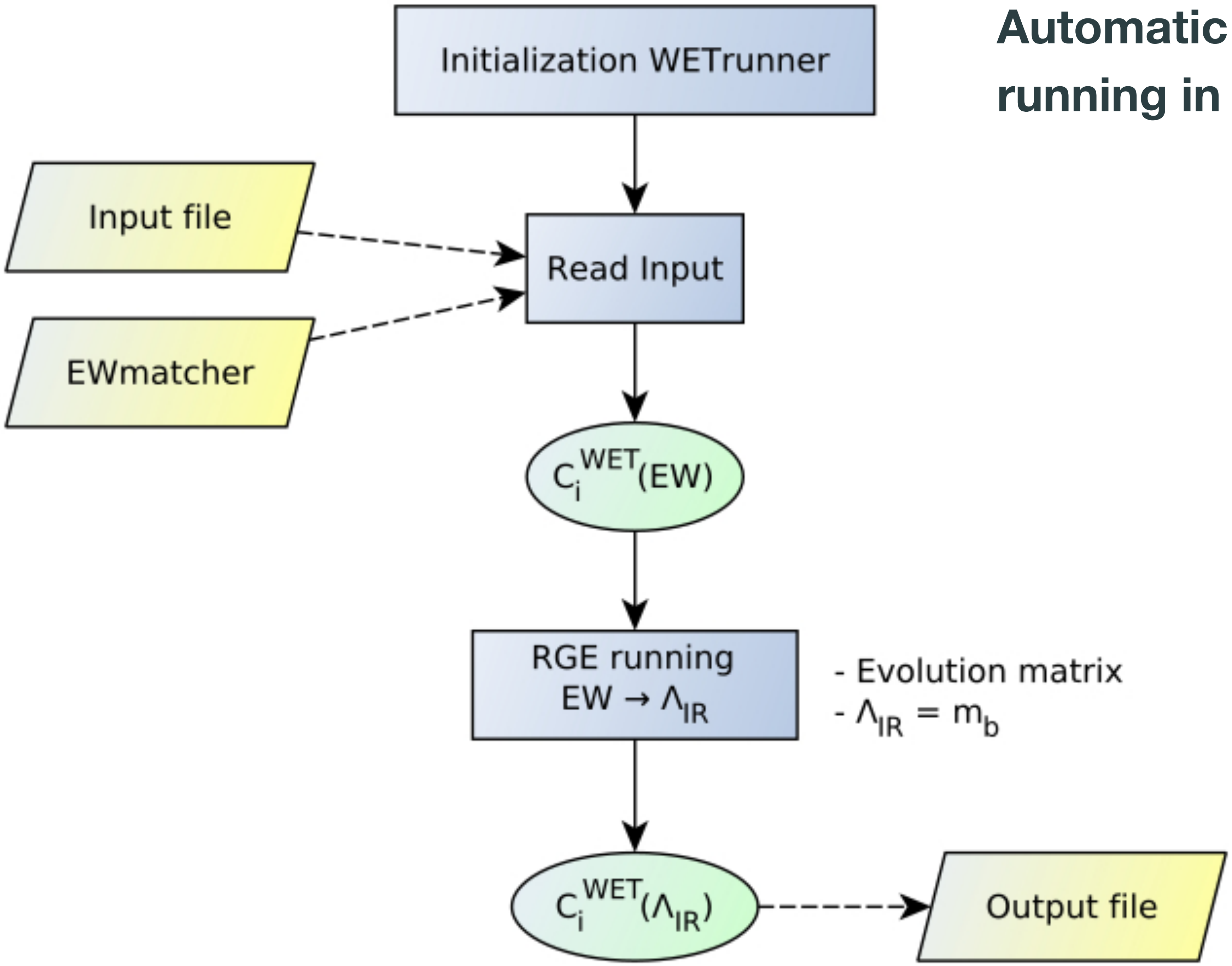
$$\Delta B = \Delta S = 1, 2$$

$$\Delta B = \Delta C = 1$$

operators. Matching to the rest of low energy operators coming soon!

Based on the results in Aebischer et al. 1512.02830

DsixTools modules: WETrunner



Automatic RGE QED and QCD running in the WET

Based on the results in Aebischer et al., 1704.06639

Installing and loading DsixTools

The **DsixTools** Mathematica package can be downloaded from

https://github.com/DsixTools/DsixTools

To use it, simply download it and place the DsixTools folder in the Mathematica base directory. Alternatively, you can place it in any other *directory* and give its path at the beginning of your Mathematica notebook

```
pathtoDsixTools = "<directory>" ; AppendTo [ $Path , pathtoDsixTools ];
```

After that one can easily load DsixTools by running the command

```
Needs["DsixTools`"]
```

Documentation and simple usage examples for each of the modules are available in the Documentation folder!

Simple usage examples: inputs

```
Block WC4
6 1.0          # phiBtilde
Block IMWCDPHI
1 1 0.1        # dphi(1,1)
1 2 0.2        # dphi(1,2)
1 3 0.3        # dphi(1,3)
2 1 0.1        # dphi(2,1)
2 2 0.2        # dphi(2,2)
2 3 0.3        # dphi(2,3)
3 1 0.4        # dphi(3,1)
3 2 0.5        # dphi(3,2)
3 3 0.6        # dphi(3,3)
Block WCDD
2 3 2 3 1.0    # dd(2,3,2,3)
Block WCPHIQ3
1 3 1.0        # phiq3(1,3)
```

WCsInput.dat

Simple text file

Similar format for the output file

Also possible to give the input directly on the Mathematica notebook or in a WCxf file

[Aebischer et al., 1712.05298]

Simple usage examples: SMEFTrunner

A DsixTools Program

This notebook loads DsixTools and shows how to use the SMEFTrunner modules.

Start DsixTools

```
Needs["DsixTools`"]
```

Read input files

```
ReadInputFiles["Options.dat", "WCsInput-CPV-SMEFT.dat", "SMInput-CPV.dat"];
```

Load SMEFTrunner module

```
In[7]:= LoadModule["SMEFTrunner"]
```

Use SMEFTrunner module

```
LoadBetaFunctions;

RunRGEsSMEFT;
```

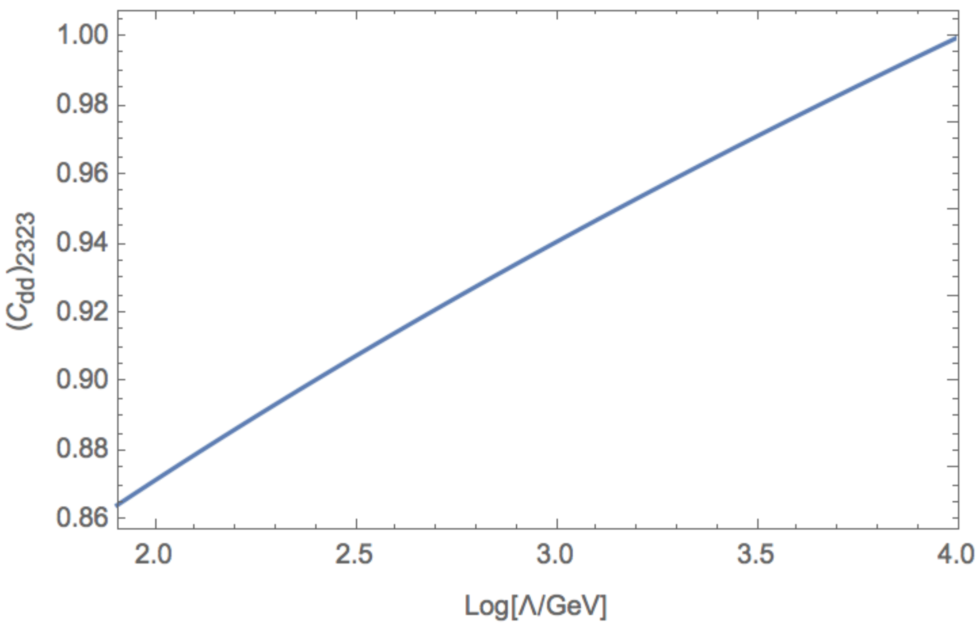
The results obtained with SMEFTrunner are saved in 'outSMEFTrunner', which can be evaluated at different values of $t = \log_{10} \mu$

Simple usage examples: SMEFTrunner

Results after SMEFTrunner

```
In[59]:= (* We find the position of (Cdd)2323 in Parameters *)
         FindParameterSMEFT[DD[2, 3, 2, 3]]

Out[59]= {443}

         (* The output of SMEFTrunner is saved in 'outSMEFTrunner', which can be evaluated at different t=Log[μ/Λ] values *)

In[89]:= outSMEFTrunner[[443]] /. t → tHIGH
         outSMEFTrunner[[443]] /. t → Log[10, 1000]
         outSMEFTrunner[[443]] /. t → tLOW

Out[89]= 1. + 0. i

Out[90]= 0.940835 + 7.52025 × 10^-17 i

Out[91]= 0.864661 + 9.0679 × 10^-16 i

In[97]:= Plot[outSMEFTrunner[[443]], {t, tLOW, tHIGH}, Frame → True, PlotRange → {{tLOW, tHIGH}, Automatic},
          FrameLabel → {"Log[Λ/GeV]", "(Cdd)2323"}]
```

Out[97]=



Simple usage examples: analytic beta functions

This notebook shows how to use the SMEFTrunner module to study SMEFT β functions analytically.

```
(* Let us compute βlq^(1) and βlq^(3) assuming top dominance and no NP effects in the 1st fermion family *)
```

In[14]:=
```
(* Top dominance approximation *)
top = {GD[i_, j_] :> 0, GE[i_, j_] :> 0, GU[i_, j_] :> If[i == j == 3, Vtb yt, If[i == 2 && j == 3, Vts yt, 0]]};
```

In[15]:=
```
(* No NP in 1st family *)
WCs2F = {φL1, φL3, φQ1, φQ3};
WCs4F = {LQ1, LQ3, LU, QE, QU1, QU8, QD1, QD8, QQ1, QQ3};
nofirst2F = Table[Part[WCs2F, i][a_, b_] → If[AnyTrue[{a, b}, # == 1 &], 0, 1] Part[WCs2F, i][a, b], {i, 1, Length[WCs2F]}];
nofirst4F = Table[Part[WCs4F, i][a_, b_, c_, d_] → If[AnyTrue[{a, b, c, d}, # == 1 &], 0, 1] Part[WCs4F, i][a, b, c, d], {i, 1, Length[WCs4F]}];
nofirst = Join[nofirst2F, nofirst4F];
```

In[20]:=
```
βlq1 = β[lq1][[2, 2, 2, 3]] /. top /. nofirst // Expand
```

Out[20]=

$$\frac{1}{2}\mathrm{Vtb\,Vts\,yt}^2\,\mathrm{LQ1}[2,2,2,2] - \frac{1}{3}\mathrm{gp}^2\,\mathrm{LQ1}[2,2,2,3] + \frac{1}{2}\mathrm{Vtb}^2\,\mathrm{yt}^2\,\mathrm{LQ1}[2,2,2,3] + \frac{1}{2}\mathrm{Vts}^2\,\mathrm{yt}^2\,\mathrm{LQ1}[2,2,2,3] + \frac{1}{2}\mathrm{Vtb\,Vts\,yt}^2\,\mathrm{LQ1}[2,2,3,3] + \frac{2}{3}\mathrm{gp}^2\,\mathrm{LQ1}[3,3,2,3] + 9\,\mathrm{g}^2\,\mathrm{LQ3}[2,2,2,3] - \mathrm{Vtb\,Vts\,yt}^2\,\mathrm{LU}[2,2,3,3] + \frac{2}{3}\mathrm{gp}^2\,\mathrm{QD1}[2,3,2,2] + \frac{2}{3}\mathrm{gp}^2\,\mathrm{QD1}[2,3,3,3] + \frac{2}{3}\mathrm{gp}^2\,\mathrm{QE}[2,3,2,2] + \frac{2}{3}\mathrm{gp}^2\,\mathrm{QE}[2,3,3,3] - \frac{2}{9}\mathrm{gp}^2\,\mathrm{QQ1}[2,2,2,3] - \frac{4}{3}\mathrm{gp}^2\,\mathrm{QQ1}[2,3,2,2] - \frac{14}{9}\mathrm{gp}^2\,\mathrm{QQ1}[2,3,3,3] - \frac{2}{3}\mathrm{gp}^2\,\mathrm{QQ3}[2,2,2,3] - \frac{2}{3}\mathrm{gp}^2\,\mathrm{QQ3}[2,3,3,3] - \frac{4}{3}\mathrm{gp}^2\,\mathrm{QU1}[2,3,2,2] - \frac{4}{3}\mathrm{gp}^2\,\mathrm{QU1}[2,3,3,3] + \mathrm{Vtb\,Vts\,yt}^2\,\varphi\mathrm{L1}[2,2] - \frac{1}{3}\mathrm{gp}^2\,\varphi\mathrm{Q1}[2,3]$$

In[21]:=
```
βlq3 = β[lq3][[2, 2, 2, 3]] /. top /. nofirst // Expand
```

Out[21]=

$$3\,\mathrm{g}^2\,\mathrm{LQ1}[2,2,2,3] + \frac{1}{2}\mathrm{Vtb\,Vts\,yt}^2\,\mathrm{LQ3}[2,2,2,2] - \frac{16}{3}\mathrm{g}^2\,\mathrm{LQ3}[2,2,2,3] - \mathrm{gp}^2\,\mathrm{LQ3}[2,2,2,3] + \frac{1}{2}\mathrm{Vtb}^2\,\mathrm{yt}^2\,\mathrm{LQ3}[2,2,2,3] + \frac{1}{2}\mathrm{Vts}^2\,\mathrm{yt}^2\,\mathrm{LQ3}[2,2,2,3] + \frac{1}{2}\mathrm{Vtb\,Vts\,yt}^2\,\mathrm{LQ3}[2,2,3,3] + \frac{2}{3}\mathrm{g}^2\,\mathrm{LQ3}[3,3,2,3] + \frac{2}{3}\mathrm{g}^2\,\mathrm{QQ1}[2,2,2,3] + \frac{2}{3}\mathrm{g}^2\,\mathrm{QQ1}[2,3,3,3] - \frac{2}{3}\mathrm{g}^2\,\mathrm{QQ3}[2,2,2,3] + 4\,\mathrm{g}^2\,\mathrm{QQ3}[2,3,2,2] + \frac{10}{3}\mathrm{g}^2\,\mathrm{QQ3}[2,3,3,3] - \mathrm{Vtb\,Vts\,yt}^2\,\varphi\mathrm{L3}[2,2] + \frac{1}{3}\mathrm{g}^2\,\varphi\mathrm{Q3}[2,3]$$

Summary of part I: DsixTools

A **Mathematica** package for the handling of $d \leq 6$ SMEFT operators

- **Easy RGE running**, with varying WCs and/or energy scales
- Input/output in simple format text files
- Transformation to fermion mass basis at EW scale and **matching to WET operators relevant for B-physics**
- One-loop **QED and QCD running from the EW scale** down to the b-quark scale

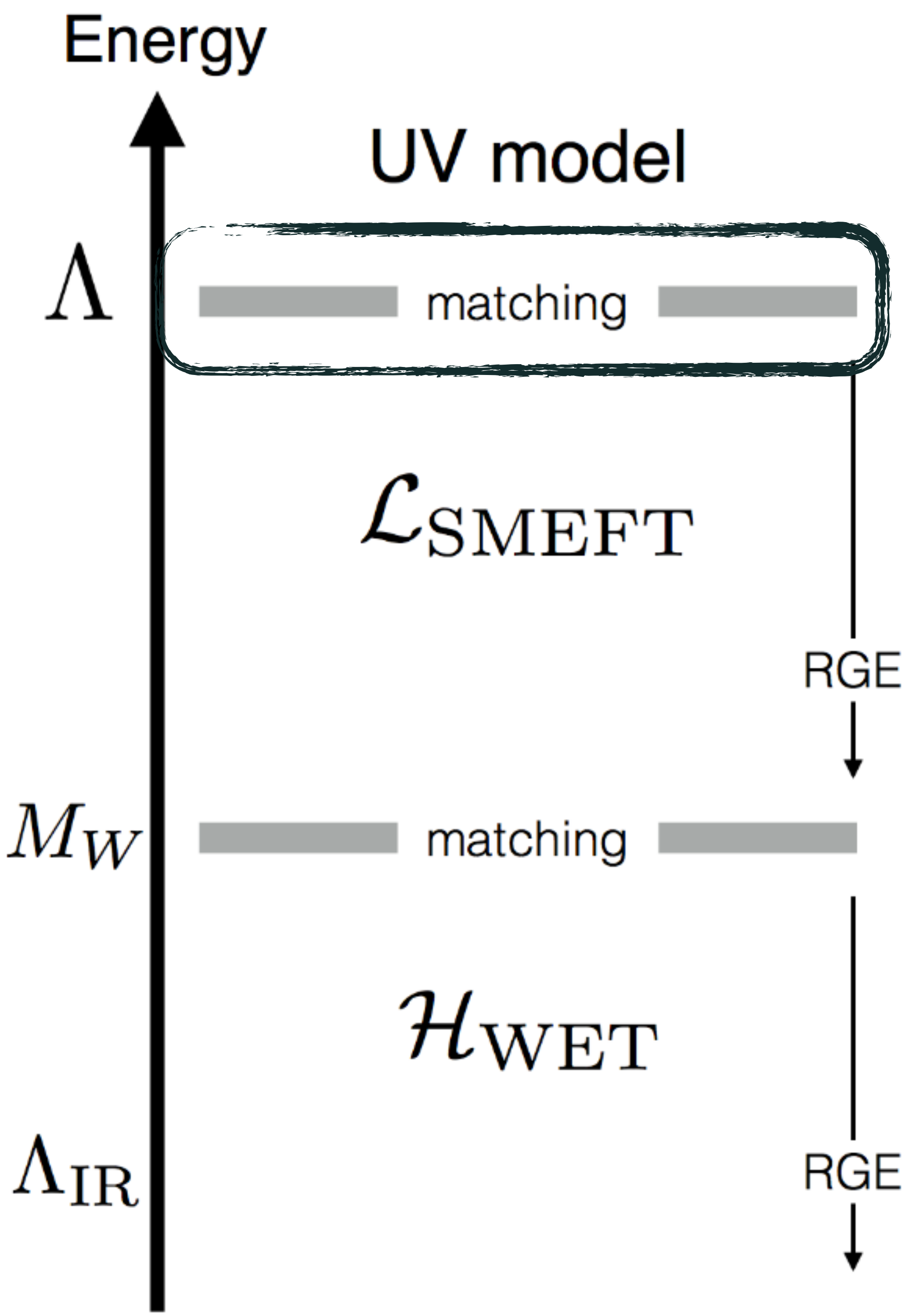
Things that DsixTools cannot do… yet

- Connection to low-energy observables: **Flavio**, **EOS**,…
- Matching of an (arbitrary) UV model and reduction to the EFT basis

Part II: One-loop matching

[JF, Portoles, Ruíz-Femenía, 1607.02142]

Tree vs one-loop matching



A simple and systematic framework to match any UV theory to its EFT would be highly desirable

See also talk by José Santiago

Interesting automated tools for tree-level matching [**MatchingTools**]

Talk by Juan Criado

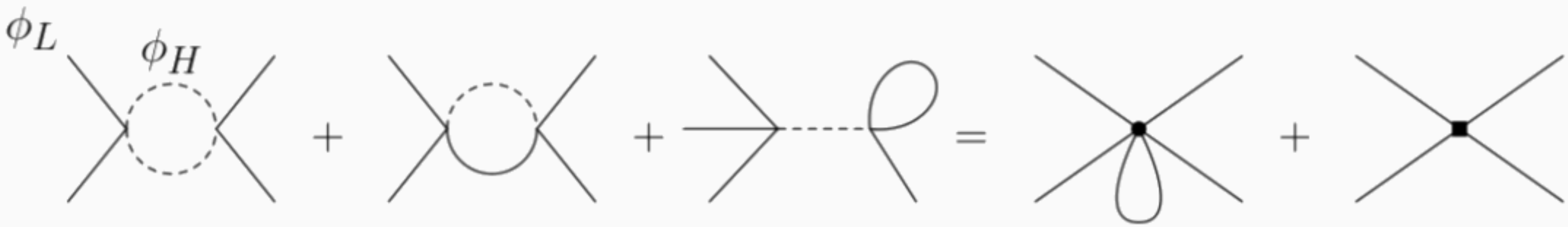
RGEs take care of the (possibly large) logs… But there are cases in which these logs are not so large

→ The inclusion of pieces that are not log-enhanced could be important [**One-loop matching**]

Diagrammatic vs functional matching

Two general approaches for the construction of EFTs:

- **Diagrammatical matching** of the Green Functions with light particles in the external legs in the UV theory and in the EFT



- **Integrate out the heavy fields**, and extract the local contributions

$$e^{iS} = \mathcal{N} \int \mathcal{D}\eta_L \, \mathcal{D}\eta_H \, \exp\left[i \int dx \, \mathcal{L}(\eta_L, \eta_H) \right]$$

Functional integration techniques are **more powerful** than diagrammatic matching when one aims to determine the **full EFT**

Making functional methods great again

Functional techniques for one-loop matching were developed long ago

Fraser ’85; Aitchison, Fraser ’85; Chan ’86; Galliard ’86; Cheyette ’88; Dittmaier, Grosse-Knetter ’95

Functional methods have recently experienced a renaissance

- One-loop EFT with only heavy particles in the loop allow for simple universal closed expressions

  Ball ’89; Bilenki, Santamaria ’94; Henning et al. ’14; Drozd et al. ’15
  See also del Aguila, Kunszt, Santiago ’16

- Several techniques recently developed for the full one-loop EFT, including heavy-light loops

  Boggia et al. ’16; Henning et al. ’16; Ellis et al. ’16

✘ These techniques require complicated subtraction techniques to remove terms already present in the EFT

I will present an alternative functional approach (without infrared subtractions) that rely on the method of expansion by regions

Functional methods: preliminaries

$$e^{iS} = \mathcal{N} \int \mathcal{D}\eta_L \, \mathcal{D}\eta_H \, \exp\left(i \int dx \, \mathcal{L}(\eta_L, \eta_H) \right) \qquad \eta \equiv \begin{pmatrix} \eta_H \\ \eta_L \end{pmatrix}$$

Background field method: Expand the Lagrangian around the solution to the Equations of Motion (EOM), $\hat{\eta}$, i.e. $\eta \to \hat{\eta} + \eta$

$$\mathcal{L}(\eta) = \mathcal{L}^{\text{tree}}(\hat{\eta}) + \left(\eta^\dagger \frac{\delta \mathcal{L}}{\delta \eta^*} + \frac{\delta \mathcal{L}}{\delta \eta} \eta \right)_{\eta = \hat{\eta}} + \frac{1}{2} \eta^\dagger \underbrace{\frac{\delta \mathcal{L}}{\delta \eta^* \delta \eta}}_{} \eta + \mathcal{O}(\eta^3)$$

$$\left. \frac{\delta \mathcal{L}}{\delta \eta^{(*)}} \right|_{\eta = \hat{\eta}} = 0 \quad \text{(EOM)} \qquad \equiv O \quad \text{fluctuation operator}$$

$$S(\hat{\eta}) = \underbrace{\int d^d x \, \mathcal{L}^{\text{tree}}(\hat{\eta})}_{\text{tree level}} \underbrace{- i \ln \left[\mathcal{N} \int \mathcal{D}\eta \, \exp\left(i \int dx \, \frac{1}{2} \eta^\dagger \, O \, \eta \right) \right]}_{\text{one loop}} + \mathcal{O}(\eta^3)$$

$$\hat{\eta}_H = \hat{\eta}_H(\hat{\eta}_L) \quad \text{(EOM)}$$

One-loop effective action

$$S^{1\text{loop}} = -i \ln \left[\mathcal{N} \int \mathcal{D}\eta_L \mathcal{D}\eta_H \exp\left(i \int dx \, \frac{1}{2} \eta^\dagger \, O \, \eta \right) \right] \qquad \eta \equiv \begin{pmatrix} \eta_H \\ \eta_L \end{pmatrix}$$

Generic form of the fluctuation operator:

$$O = \begin{pmatrix} \Delta_H & \mathbf{X}^\dagger_{\mathbf{LH}} \\ \mathbf{X}_{\mathbf{LH}} & \Delta_L \end{pmatrix}$$

Δ_H, Δ_L: heavy and light loops

X_{HL}: heavy-light loops

Our aim: compute the one-loop heavy particle effects and extract their contributions to the Wilson coefficients of the EFT

Shifting the heavy-light loops away

$$S^{1\text{loop}} = -i \ln \left[\mathcal{N} \int \mathcal{D}\eta_L \mathcal{D}\eta_H \exp\left(i \int dx \, \frac{1}{2} \eta^\dagger \, O \, \eta \right) \right] \qquad \eta \equiv \begin{pmatrix} \eta_H \\ \eta_L \end{pmatrix}$$

Generic form of the fluctuation operator:

$$O = \begin{pmatrix} \Delta_H & \mathbf{X}_{\mathbf{LH}}^\dagger \\ \mathbf{X}_{\mathbf{LH}} & \Delta_L \end{pmatrix}$$

Δ_H, Δ_L: heavy and light loops

X_{HL}: heavy-light loops

We can isolate the effect of the heavy particles with a shift of the fields

$$\eta \to P\,\eta \qquad P = \begin{pmatrix} I & 0 \\ -\Delta_L^{-1} X_{LH} & I \end{pmatrix} \implies P^\dagger O P = \begin{pmatrix} \widetilde{\Delta}_H & 0 \\ 0 & \Delta_L \end{pmatrix}$$

$$\widetilde{\Delta}_H = \Delta_H - X_{LH}^\dagger \Delta_L^{-1} X_{LH}$$

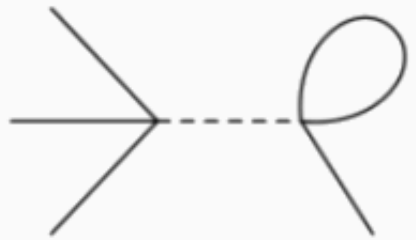
JF, Portolés, Ruiz-Femenía, 1607.02142

Heavy-field one-loop effective action

$$e^{iS^{\text{1loop}}} = \underbrace{\int \mathcal{D}\eta_H \exp\left[i\int dx\, \frac{1}{2}\eta_H^\dagger \widetilde{\Delta}_H \eta_H\right]}_{=\left(\det \widetilde{\Delta}_H\right)^{-c}} \mathcal{N}\int \mathcal{D}\eta_L \exp\left[i\int dx\, \frac{1}{2}\eta_L^\dagger \Delta_L \eta_L\right]$$

$$(c = 1/2, -1 \text{ for bosons/fermions})$$

The integral over η_L gives loops with only light particles

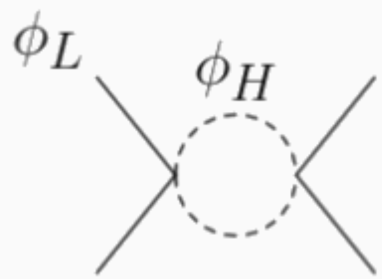


All one-loop heavy and heavy-light contributions contained in $\det \widetilde{\Delta}_H$!

$$S_H = ic \ln \det \widetilde{\Delta}_H$$

Expansion by regions

$$S_H = \frac{i}{2} \ln\det \widetilde{\Delta}_H \qquad \widetilde{\Delta}_H = \Delta_H - X_{LH}^{\dagger} \Delta_L^{-1} X_{LH}$$



Heavy-light loops have **soft** ($p \sim m_L$) and **hard** ($p \sim m_H$) momentum regions but **only the hard region contributes to the EFT**

JF, Portolés, Ruiz-Femenía, 1607.02142

$\longrightarrow$ In dim. reg. one can do a **expansion by regions** by Taylor expanding the loop integrand, and integrating in the full domain.

Beneke, Smirnov '98

$$S_H = \underbrace{S_H^{\text{hard}}}_{p \,\sim\, m_H \,\gg\, m_L} + \underbrace{S_H^{\text{soft}}}_{p \,\sim\, m_L \,\ll\, m_H}$$

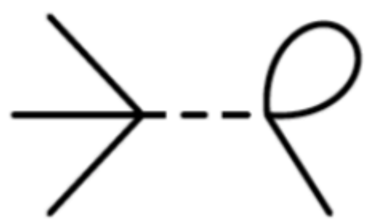
Pure short distance!

Heavy propagator expanded in p/m_H
(same as loops with tree EFT vertices)

A simple example of expansion by regions at work

$$\mathcal{L}_{\rm UV} = \frac{1}{2}\left(\partial_\mu\phi\,\partial^\mu\phi - M^2\,\phi^2\right) + \frac{1}{2}\left(\partial_\mu\varphi\,\partial^\mu\varphi - m^2\,\varphi^2\right) - \frac{\kappa}{4!}\,\varphi^4 - \frac{\lambda}{3!}\,\varphi^3\,\phi \qquad (M \gg m)$$

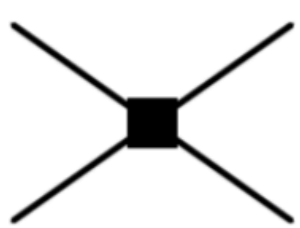
$$= \frac{i}{16\pi^2}\,\lambda^2\left[3 + \frac{s+t+u}{2M^2} + 3\,\frac{m^2}{M^2}\ln\left(\frac{m^2}{M^2}\right)\right] + \mathcal{O}(M^{-4})$$



$$= \frac{i}{16\pi^2}\,\lambda^2\left[-2\,\frac{m^2}{M^2} + 2\,\frac{m^2}{M^2}\ln\left(\frac{m^2}{M^2}\right)\right] + \mathcal{O}(M^{-4})$$

$$\mathcal{L}_{\rm EFT} = \frac{1}{2}\left(\partial_\mu\varphi\,\partial^\mu\varphi - m^2\,\varphi^2\right) + \frac{\lambda^2}{72M^2}\varphi^6 + \frac{1}{16\pi^2}\left[\frac{\alpha}{4!}\varphi^4 + \frac{\beta}{4!M^2}\varphi^2\partial^2\varphi^2 + \frac{\gamma}{6!}\varphi^6\right]$$

$$= \frac{i}{16\pi^2}\,\lambda^2\left[-5\,\frac{m^2}{M^2} + 5\,\frac{m^2}{M^2}\ln\left(\frac{m^2}{M^2}\right)\right] + \mathcal{O}(M^{-4})$$



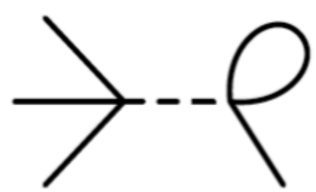
$$= i\,\alpha - i\,\frac{\beta}{3M^2}\,(s+t+u)$$

A simple example of expansion by regions at work

$$\mathcal{L}_{\rm UV} = \frac{1}{2}\left(\partial_\mu\phi\,\partial^\mu\phi - M^2\,\phi^2\right) + \frac{1}{2}\left(\partial_\mu\varphi\,\partial^\mu\varphi - m^2\,\varphi^2\right) - \frac{\kappa}{4!}\,\varphi^4 - \frac{\lambda}{3!}\,\varphi^3\,\phi \qquad (M \gg m)$$

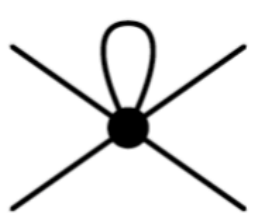
$$= \frac{i}{16\pi^2}\,\lambda^2\left[3 + 3\,\frac{m^2}{M^2} + \frac{s+t+u}{2M^2}\right]\Bigg|_{\rm hard}$$

$$+ \frac{i}{16\pi^2}\,\lambda^2\left[-3\,\frac{m^2}{M^2} + 3\,\frac{m^2}{M^2}\ln\left(\frac{m^2}{M^2}\right)\right]\Bigg|_{\rm soft} + \mathcal{O}(M^{-4})$$

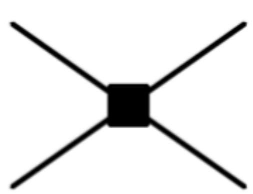


$$= \frac{i}{16\pi^2}\,\lambda^2\left[-2\,\frac{m^2}{M^2} + 2\,\frac{m^2}{M^2}\ln\left(\frac{m^2}{M^2}\right)\right]\Bigg|_{\rm soft} + \mathcal{O}(M^{-4})$$

$$\mathcal{L}_{\rm EFT} = \frac{1}{2}\left(\partial_\mu\varphi\,\partial^\mu\varphi - m^2\,\varphi^2\right) + \frac{\lambda^2}{72M^2}\varphi^6 + \frac{1}{16\pi^2}\left[\frac{\alpha}{4!}\varphi^4 + \frac{\beta}{4!M^2}\varphi^2\partial^2\varphi^2 + \frac{\gamma}{6!}\varphi^6\right]$$



$$= \frac{i}{16\pi^2}\,\lambda^2\left[-5\,\frac{m^2}{M^2} + 5\,\frac{m^2}{M^2}\ln\left(\frac{m^2}{M^2}\right)\right] + \mathcal{O}(M^{-4})$$

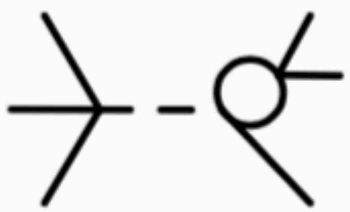


$$= i\,\alpha - i\,\frac{\beta}{3M^2}\,(s+t+u) \qquad \Longrightarrow \qquad \alpha = \tfrac{3}{16\pi^2}\lambda^2\left(1 + \tfrac{m^2}{M^2}\right), \quad \beta = -\tfrac{3}{16\pi^2}\,\tfrac{\lambda^2}{2}$$

A simple example of expansion by regions at work

$$\mathcal{L}_{\rm UV} = \frac{1}{2}\left(\partial_\mu \phi\, \partial^\mu \phi - M^2\, \phi^2\right) + \frac{1}{2}\left(\partial_\mu \varphi\, \partial^\mu \varphi - m^2\, \varphi^2\right) - \frac{\kappa}{4!}\,\varphi^4 - \frac{\lambda}{3!}\,\varphi^3\,\phi$$

$$= \frac{i}{16\pi^2}\, 45\, \frac{\kappa\,\lambda^2}{M^2}\Bigg|_{\rm hard} + \cancel{\frac{i}{16\pi^2}\, 45\, \frac{\kappa\,\lambda^2}{M^2} \ln\left(\frac{m^2}{M^2}\right)\Bigg|_{\rm soft}} + \mathcal{O}(M^{-4})$$



$$= \cancel{\frac{i}{16\pi^2}\, 30\, \frac{\kappa\,\lambda^2}{M^2} \ln\left(\frac{m^2}{M^2}\right)\Bigg|_{\rm soft}} + \mathcal{O}(M^{-4})$$

$$\mathcal{L}_{\rm EFT} = \frac{1}{2}\left(\partial_\mu \varphi\, \partial^\mu \varphi - m^2\, \varphi^2\right) + \frac{\lambda^2}{72 M^2}\varphi^6 + \frac{1}{16\pi^2}\left[\frac{\alpha}{4!}\varphi^4 + \frac{\beta}{4! M^2}\varphi^2 \partial^2 \varphi^2 + \frac{\gamma}{6!}\varphi^6\right]$$

$$= \cancel{\frac{i}{16\pi^2}\, 75\, \frac{\kappa\,\lambda^2}{M^2} \ln\left(\frac{m^2}{M^2}\right)} + \mathcal{O}(M^{-4})$$

$$= \frac{i}{16\pi^2}\, \frac{\gamma}{M^2} \quad \Longrightarrow \quad \gamma = 45\, \kappa \lambda^2$$

No need to compute EFT diagrams!

Combining the two…

$$S_H = ic \ln \det \widetilde{\Delta}_H \qquad \widetilde{\Delta}_H = \Delta_H - X^\dagger_{LH} \Delta_L^{-1} X_{LH}$$

We only need the hard part:

$$\int d^d x \, \mathcal{L}^{\text{1loop}}_{\text{EFT}} = S_H^{\text{hard}} + S_H^{\text{soft}} + S_L - S_{\text{EFT}}^{\text{tree}}$$

Expansion by regions: Taking the hard piece is equivalent to expanding the loop integrand in

$$p_\mu, m_H \sim \zeta$$

and truncate to a given order in ζ^{-n} (related to the dimension of the EFT operators)

Long story short

Applying standard path integral techniques to $S_H = ic \ln \det \widetilde{\Delta}_H$

$$S_H = \mp \frac{i}{2} \int d^d x \sum_{n=1}^{\infty} \frac{1}{n} \int \frac{d^d p}{(2\pi)^d} \,\mathrm{tr} \left\{ \left(\frac{2ip\hat{D} + \hat{D}^2 + U\left(x, \partial_x + ip\right)}{p^2 - m_H^2} \right)^n \mathbb{1} \right\}$$

$$U\left(x, \partial_x\right) = -\hat{D}^2 - m_H^2 - \widetilde{\Delta}_H\left(x, \partial_x\right)$$

Counting: $p_\mu, m_H \sim \zeta$

[Exp. by regions]

One-loop matching essentially reduced to algebra!... but lots of algebra is usually involved

$\Longrightarrow$ **Automation** becomes essential for any realistic model

Work in progress [Celis, JF, Ruíz-Femenía, Vicente, Virto]

A Mathematica package for the matching and RGE evolution from the new physics scale to the scale of low energy observables

Manual: arXiv:1704.04504
Website: https://dsixtools.github.io/

New features are coming

Comments (including critical ones!), questions and suggestions are welcome!

Thanks!