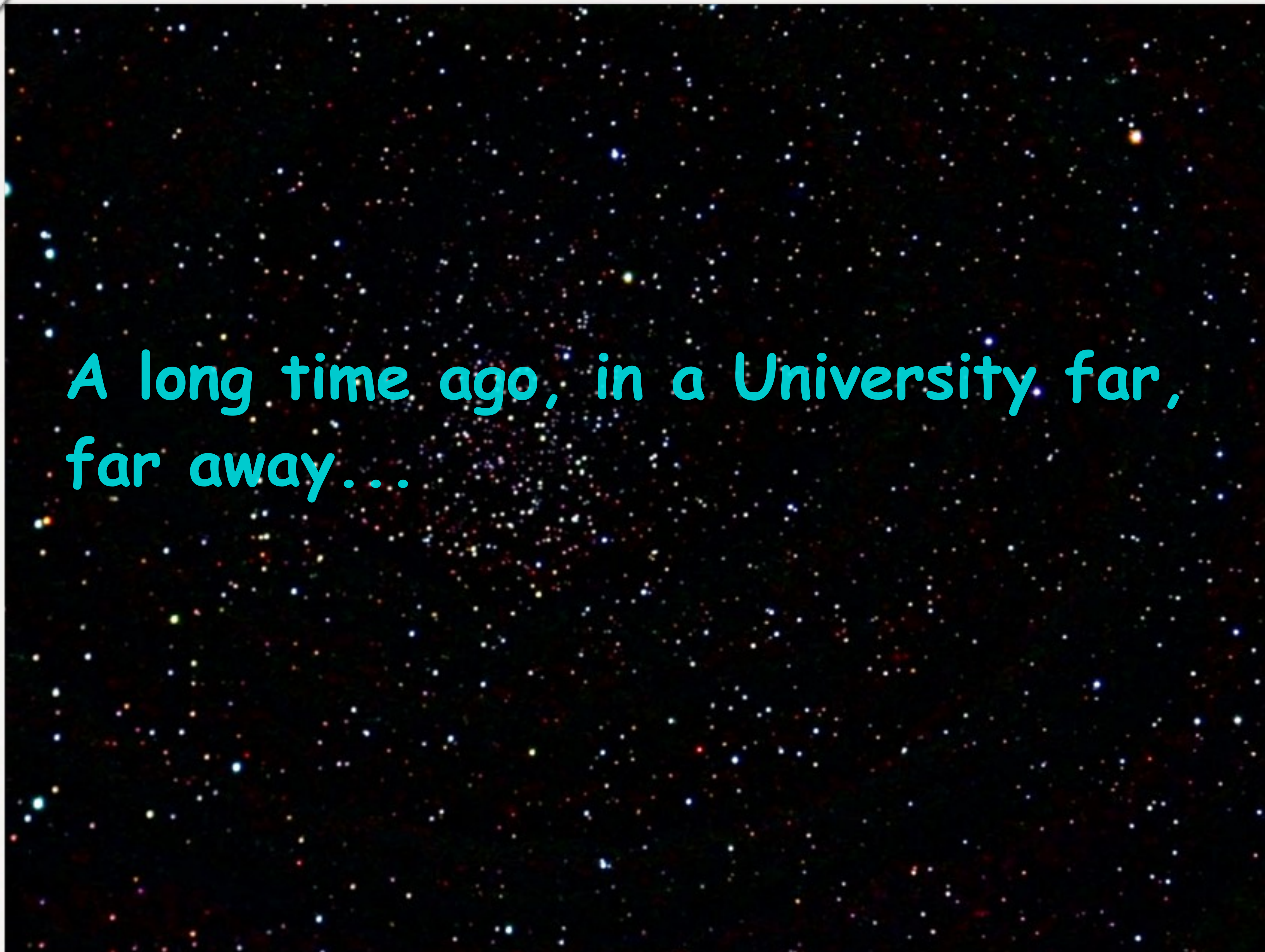
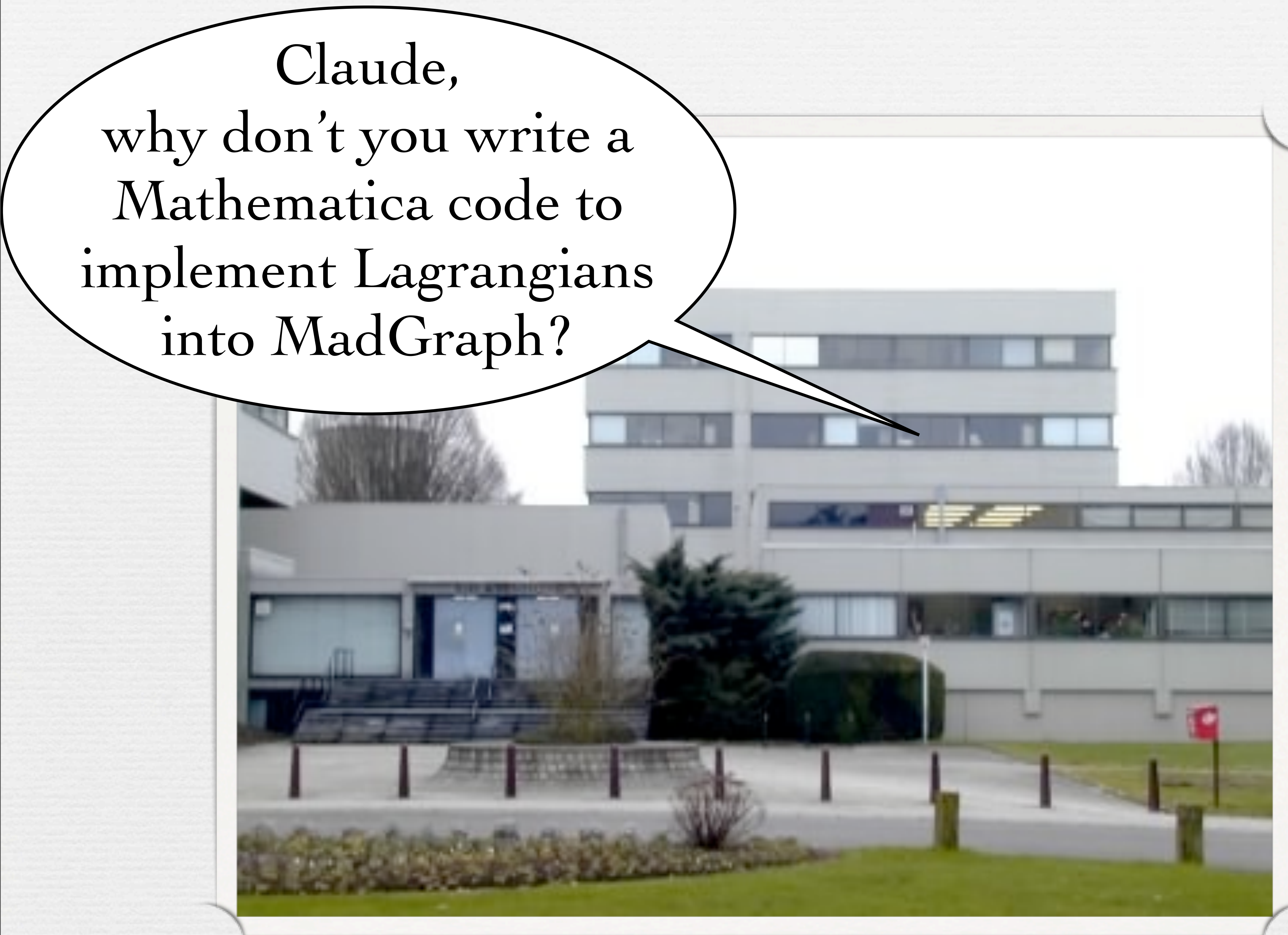
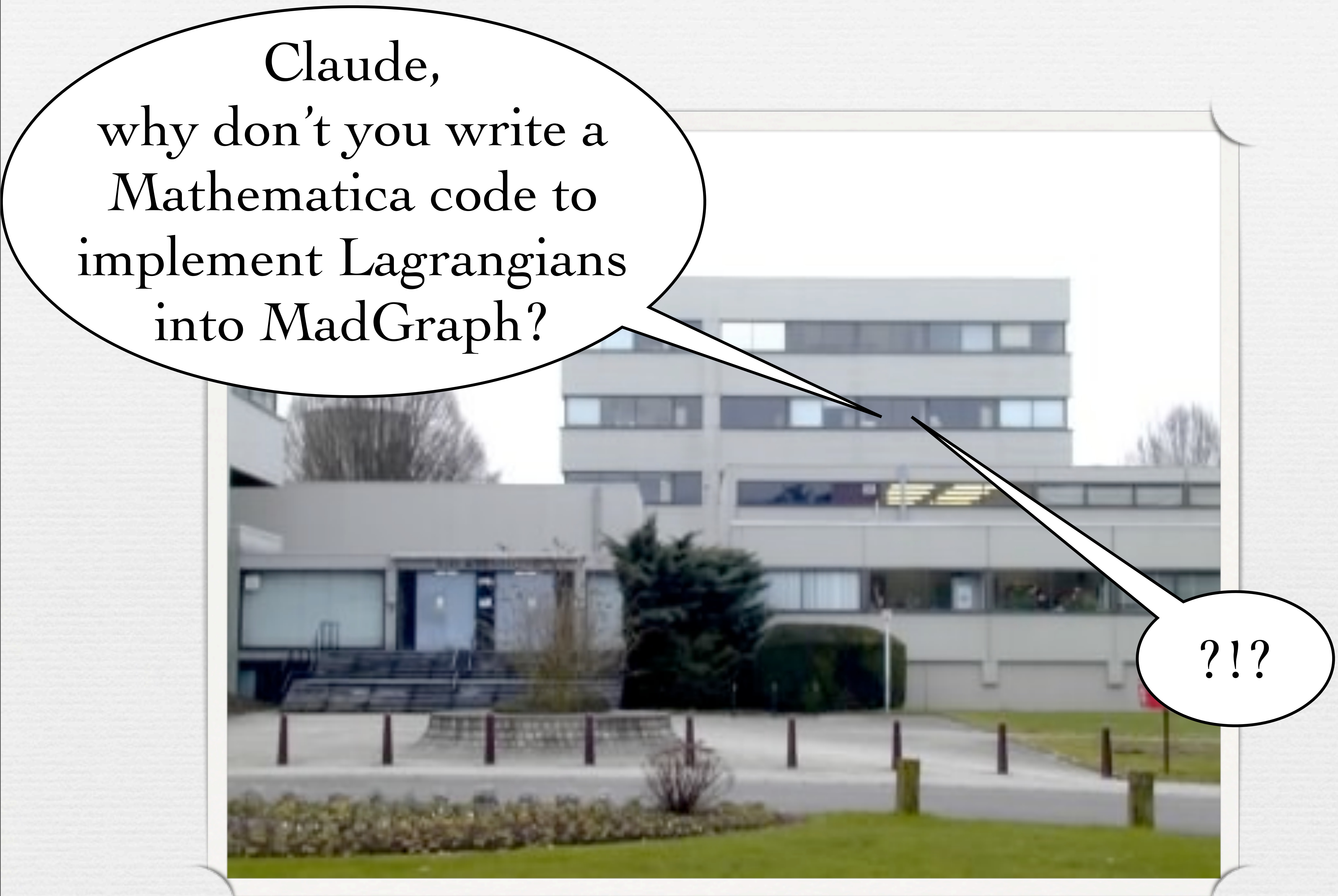
Status and prospects of FeynRules

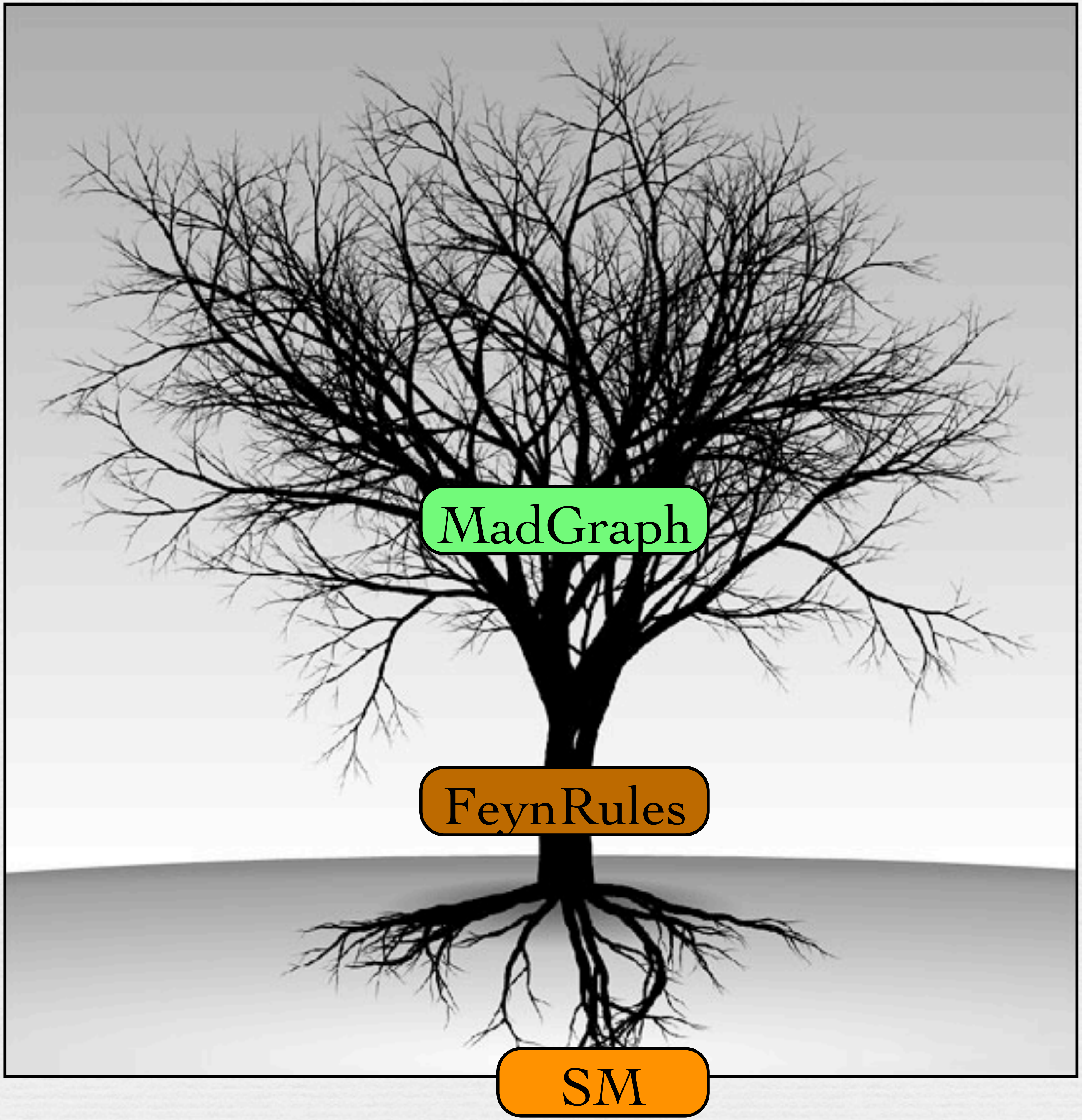
Claude Duhr

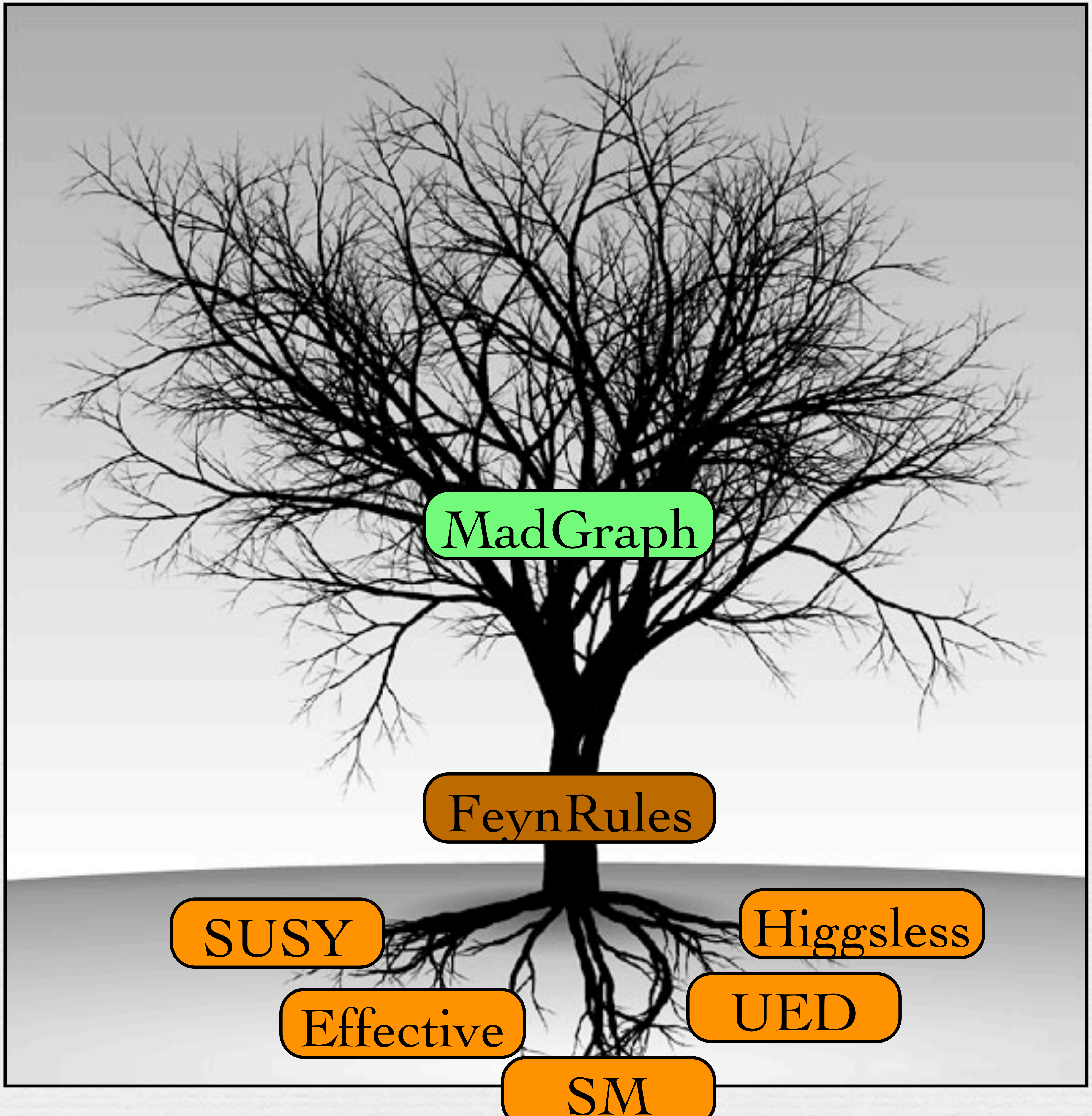
FeynRules 2010
St. Odile, March 15, 2010

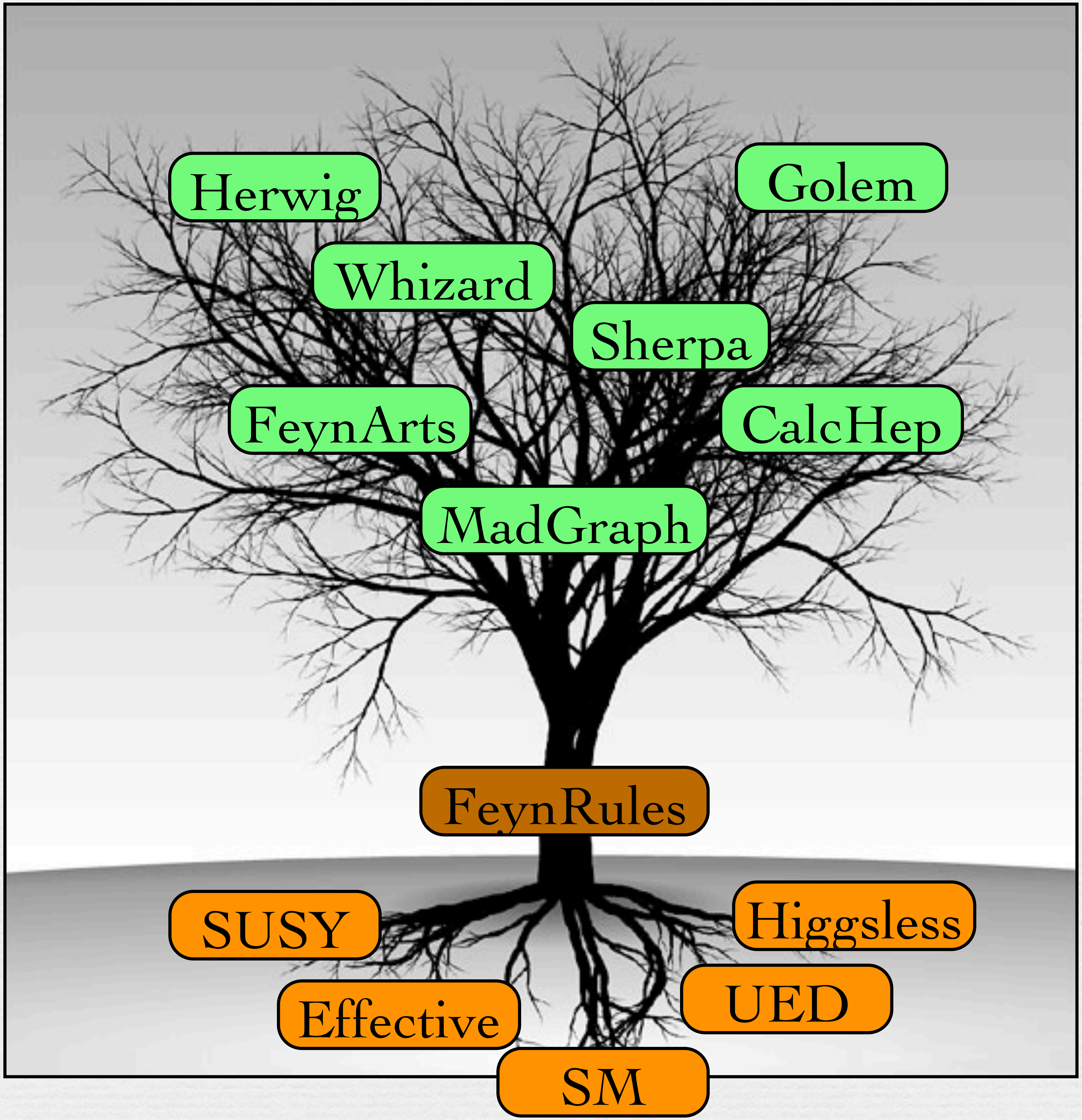

A long time ago, in a University far,
far away...


Claude,
why don't you write a Mathematica code to implement Lagrangians into MadGraph?


Claude, why don't you write a Mathematica code to implement Lagrangians into MadGraph?
?!?


MadGraph
FeynRules
SM


MadGraph
FeynRules
SUSY
Higgsless
Effective
UED
SM


Herwig
Golem
Whizard
Sherpa
FeynArts
CalcHep
MadGraph
FeynRules
SUSY
Higgsless
Effective
UED
SM

- The present: current features and limitations
- The future: new challenges and development plans
- Towards a communication platform between theorists and experimentalists

The present:
current features and limitations

The FeynRules philosophy

- If at the LHC we come to the situation that we have to discriminate between a plethora of competing models, we need an efficient and fast way to simulate all these models.

- We aim to provide the user a framework where new models can be easily implemented into matrix element generators, without having to know the technical details of the generator (conventions, programming language).

- From *one* FeynRules model many different implementations can be obtained:
  - ➡ Try to avoid redoing the same work over and over again.
  - ➡ Each generator has its own strengths, and we want to exploit all of them at once!

FeynRules - current status

- FeynRules can cope with any 4D Lagrangian, the only constraints are gauge and Lorentz invariance, and the field types:
    - ➡ Scalars
    - ➡ Dirac and Majorana fermions
    - ➡ Vectors
    - ➡ Spin 2
    - ➡ ghosts

- Higher dimensional operators are not a problem (at least for FeynRules)!

FeynRules - current status

- The input requested form the user is twofold.
  - ➡ A FeynRules model file: definition of particles and parameters in the Lagrangian:

```
F[1]  ==  {ClassName        ->   q,
             SelfConjugate   ->  False,
             Indices         ->  {Index[Colour]},
             Mass            ->  {MQ,  200},
             Width           ->  {WQ, 5}   }
```

  - ➡ The Lagrangian of the model:

```
L = -1/4 FS[G,mu,nu,a] FS[G,mu,nu,a]
      + I qbar.Ga[mu].del[q,mu] - MQ qbar.q
```

FeynRules - current status

```
L = -1/4 FS[G,mu,nu,a] FS[G,mu,nu,a]
     + I qbar.Ga[mu].DC[q,mu] - MQ qbar.q
```

- FeynRules knows about the gauge groups, *i.e.*, the field strength tensors and covariant derivatives are automatically defined.
  ➡ In quantum theories we need to fix the gauge.
  Can we also generate the gauge fixing and ghost terms automatically?

- The user can now ask FeynRules to compute the Feynman rules:

```
FeynmanRules[ L ];
```

FeynRules - current status

- The Feynman rules can be exported to various matrix element generators via dedicated interfaces.
- Currently implemented interfaces:
  - ➡ CalcHep/CompHep - Micr'Omegas
  - ➡ FeynArts/FormCalc
  - ➡ MadGraph/MadEvent
  - ➡ Sherpa
  - ➡ Whizard/Omega (beta)
  - ➡ Golem and Herwig will be added in the future.
- FeynRules then produces a set of files that can be copied into the matrix element generator and be used in the same way as all the other models («plug 'n' play»).

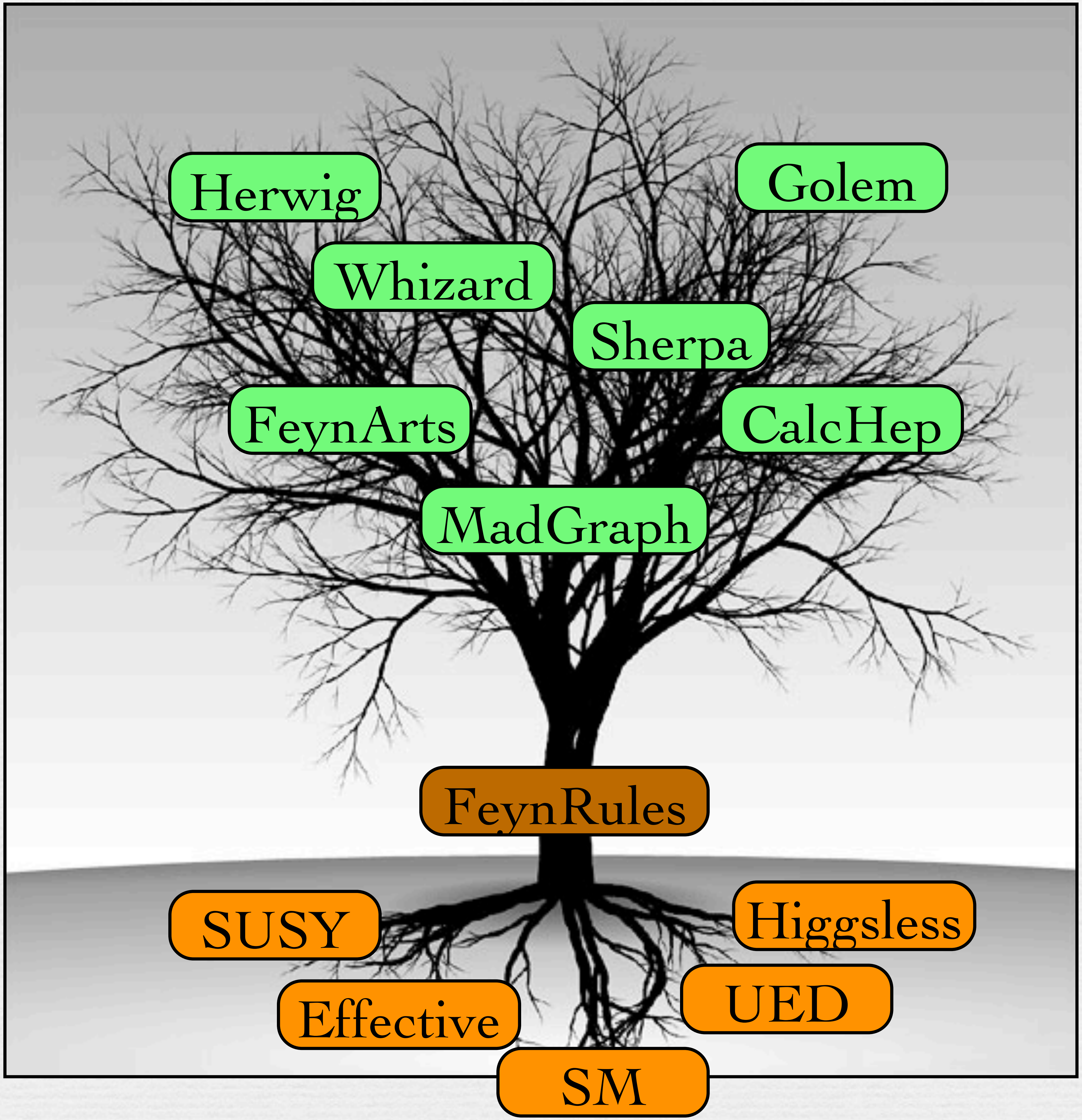
Available models

- We want to provide the users with a 'critical mass' of models from which new models/extensions of existing models can be created.
- Currently implemented models:
  - ➡ SM
  - ➡ complete MSSM (+extensions: NMSSM, RPV, ...)
  - ➡ Universal extra dimensions
  - ➡ Large extra dimensions
  - ➡ Moose models (3-site model) + linear sigma models
  - ➡ Effective operators
- Missing models: Little Higgs theories, Technicolor, Leptoquarks, GUT theories ...

The future:
new challenges and development plans

Limitations

- In principle, every QFT model can be implemented in FeynRules.
- In practice, this is hampered by the fact that
  - ➡ the Lagrangian must be entered in terms of four-component spinors.
  - ➡ the mass matrices must be diagonalized by hand.
  - ➡ supersymmetric theories are most conveniently written in terms of superfields.
- FeynRules so far only deals with tree-level objects (no counterterms).
- Most of the matrix element generators have color and/or Lorentz structures hardcoded, limiting in this way the number of models that can be implemented.



Herwig
Golem
Whizard
Sherpa
FeynArts
CalcHep
MadGraph
FeynRules
SUSY
Higgsless
UED
Effective
SM

General Lorentz structures

- FeynRules can be used to generate the Feynman rules also for higher dimensional operators and arbitrary gauge groups.
- Some generators have the Lorentz and color structures hardcoded, *e.g.*,
  ➡ generic couplings in FeynArts.
  ➡ HELAS library for MadGraph and Herwig.
- Aim: Use the information available in FeynRules to extend the library of Lorentz structures of the matrix element generator.

FeynArts generic couplings

- For FeynArts, we can write the generic couplings file directly from FeynRules (Céline Degrande).
- In other words, for each model we can write out the so-called *classes couplings* as well as the associated *generic couplings*.
- This will allow to implement *any* QFT model into FeynArts.

Automatic generation of HELAS routines

- The interface for MadGraph 5 will be rewritten from scratch, and will output a set of Python files that do not only contain the information on the couplings, but also on the Lorentz structures of each vertex.

- A Python code is being developed (W. Link, O. Mattelaer) that will allow to write Fortran HELAS routines directly from the FeynRules information.

- The same strategy could be followed also for Herwig (writing HELAS routines in C++).

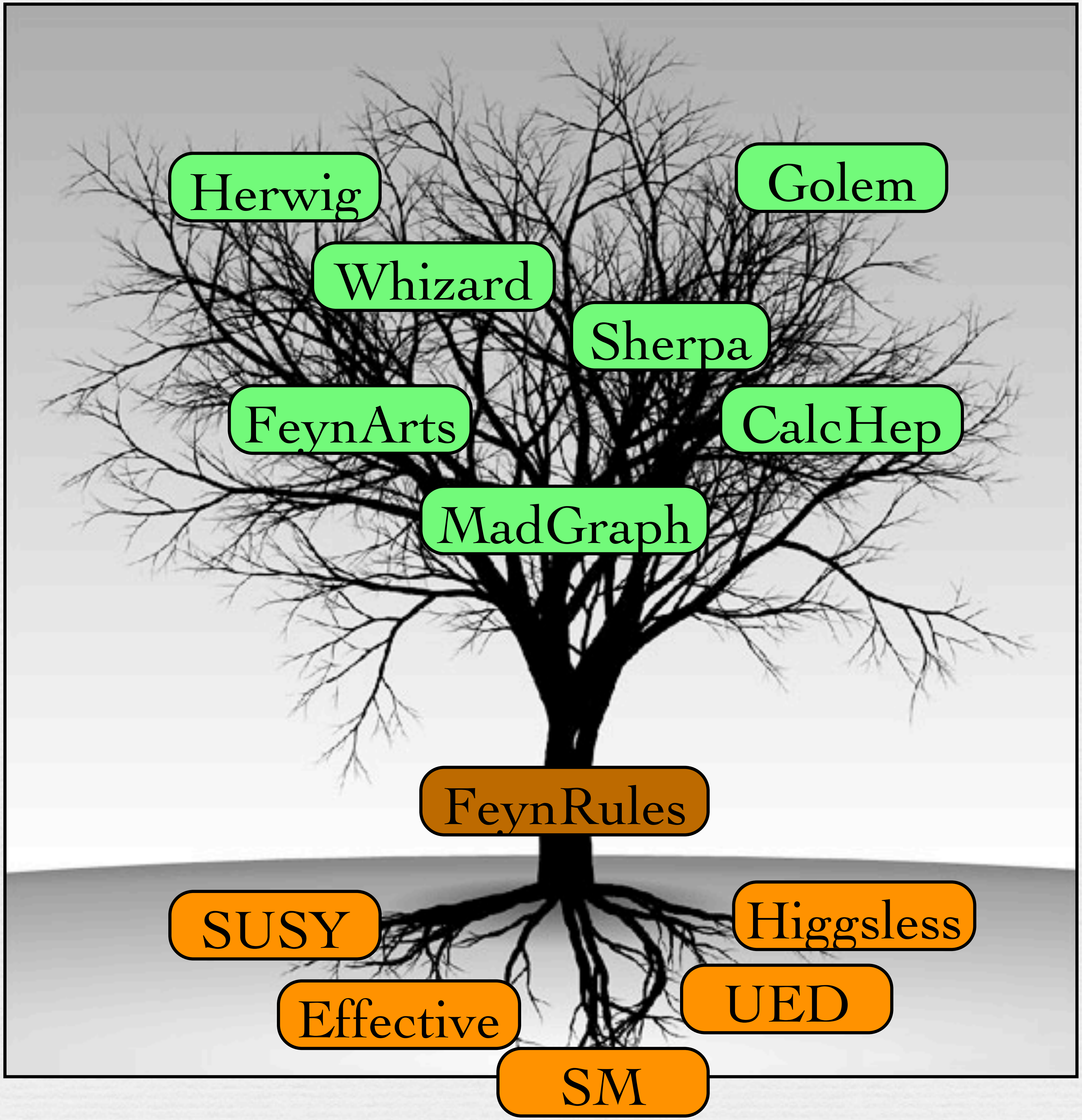
General Lorentz structures

- At the end, every generator can then in principle handle higher-dimensional operators:

  ✓ CalcHep, Golem, Whizard: Lorentz structures are part of model definition.

  ✓ FeynArts: both generic and classes couplings are written by FeynRules.

  ✓ MadGraph & Herwig: Automatic generation of HELAS routines form Python module.


Herwig
Golem
Whizard
Sherpa
FeynArts
CalcHep
MadGraph
FeynRules
SUSY
Higgsless
UED
Effective
SM

Weyl fermions & superfields

- Supersymmetric theories are most conveniently written in terms of superfields.
- Supermultiplets contain Weyl spinors as component fields, rather than four-component spinors.
- As a first step, we have implemented Weyl fermions into FeynRules

```
W[41] ==
  {ClassName -> qL,
   Chirality -> Left,
   SelfConjugate -> False,
   Indices -> {Index[Colour]}},
```

```
F[3] == {
 ClassName -> q,
 SelfConjugate -> False,
 Indices -> {Index[Colour]},
 WeylComponents -> {qL, qRbar}},
```

Weyl fermions & superfields

- FeynRules then replaces the Weyl fermions by there four-component expression,

$$q_L = P_L\, q \qquad\qquad q_R = P_L\, q^c$$
$$\bar{q}_R = P_R\, q \qquad\qquad \bar{q}_L = P_R\, q^c$$

- After this replacement, the Lagrangian (and the Feynman rules) are again expressed in terms of four-component spinors, as required by the matrix element generators.

- We tested this new feature already by rewriting the MSSM completely in terms of Weyl fermions.

Weyl fermions & superfields

- A similar approach can be taken also for superfields:
  - ➡ chiral superfields:

$$\Phi = (\phi, \chi, F)$$

  - ➡ gauge superfields:

$$V = (A, \lambda, D)$$

$$S = \int d^4x d^2\theta d^2\bar{\theta} \Phi^\dagger \Phi + \int d^4x d^2\theta \mathcal{W}(\Phi) + \text{h.c.}$$

➡

$$\mathcal{L} = \partial_\mu \phi^\dagger \partial^\mu \phi + i\chi^\dagger \bar{\sigma}^\mu \partial_\mu \chi + F^\dagger F - \frac{\partial \mathcal{W}}{\partial \phi} F - \frac{1}{2} \frac{\partial^2 \mathcal{W}}{\partial \phi^2} \chi \cdot \chi + \text{h.c.}$$

Weyl fermions & superfields

$$S = \int d^4x d^2\theta d^2\bar{\theta} \Phi^\dagger \Phi + \int d^4x d^2\theta \mathcal{W}(\Phi) + \text{h.c.}$$

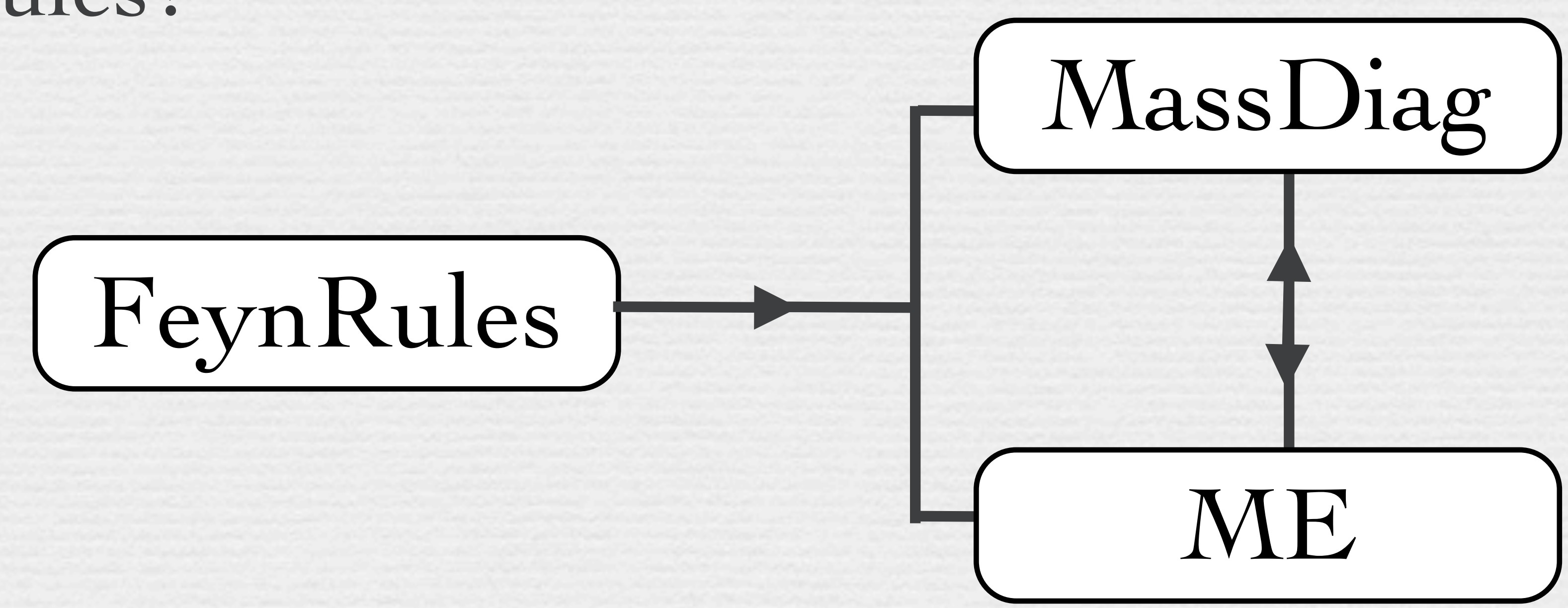
$$\mathcal{L} = \partial_\mu \phi^\dagger \partial^\mu \phi + i\chi^\dagger \bar{\sigma}^\mu \partial_\mu \chi + F^\dagger F - \frac{\partial \mathcal{W}}{\partial \phi} F - \frac{1}{2} \frac{\partial^2 \mathcal{W}}{\partial \phi^2} \chi \cdot \chi + \text{h.c.}$$

- The equations of motion for the F and D terms are trivial, and can be solved 'easily' in Mathematica, allowing to reduce the superspace action completely to a Lagrangian in terms of physical component fields,

$$\mathcal{L} = \partial_\mu \phi^\dagger \partial^\mu \phi + i\chi^\dagger \sigma^\mu \partial \chi - \left| \frac{\partial \mathcal{W}}{\partial \phi} \right|^2 - \frac{1}{2} \frac{\partial^2 \mathcal{W}}{\partial \phi^2} \chi \cdot \chi + \text{h.c.}$$

Mass eigenstates

- At present, the mass matrices must be diagonalized by hand, and the relation between the gauge and mass eigenstates is part of the definition of the model.
- We can extract the mass matrix from the Lagrangian and diagonalize it numerically.
- Should this be done inside or outside Mathematica/FeynRules?

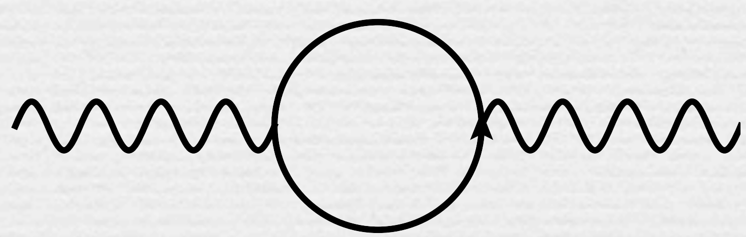


Let's get loopy...

- FeynRules computes only *tree-level* Feynman rules.

- This is of course sufficient for all tree-level matrix element generators.

- For loop-level generators (*cf.* FeynArts, Golem) we also need the UV counterterms.



$$= \frac{1}{\epsilon} \times \ldots$$

  ➡ Which scheme to use?
  ➡ How to deal with mixing of particles?
  ➡ ...

Connecting the high and the low scale

- From the counterterms we can determine the β and γ functions, *i.e.*, the RG evolution,

$$\mu\frac{\partial g_0}{\partial\mu} = 0 \Rightarrow \mu\frac{\partial g}{\partial\mu} = \beta(g)$$

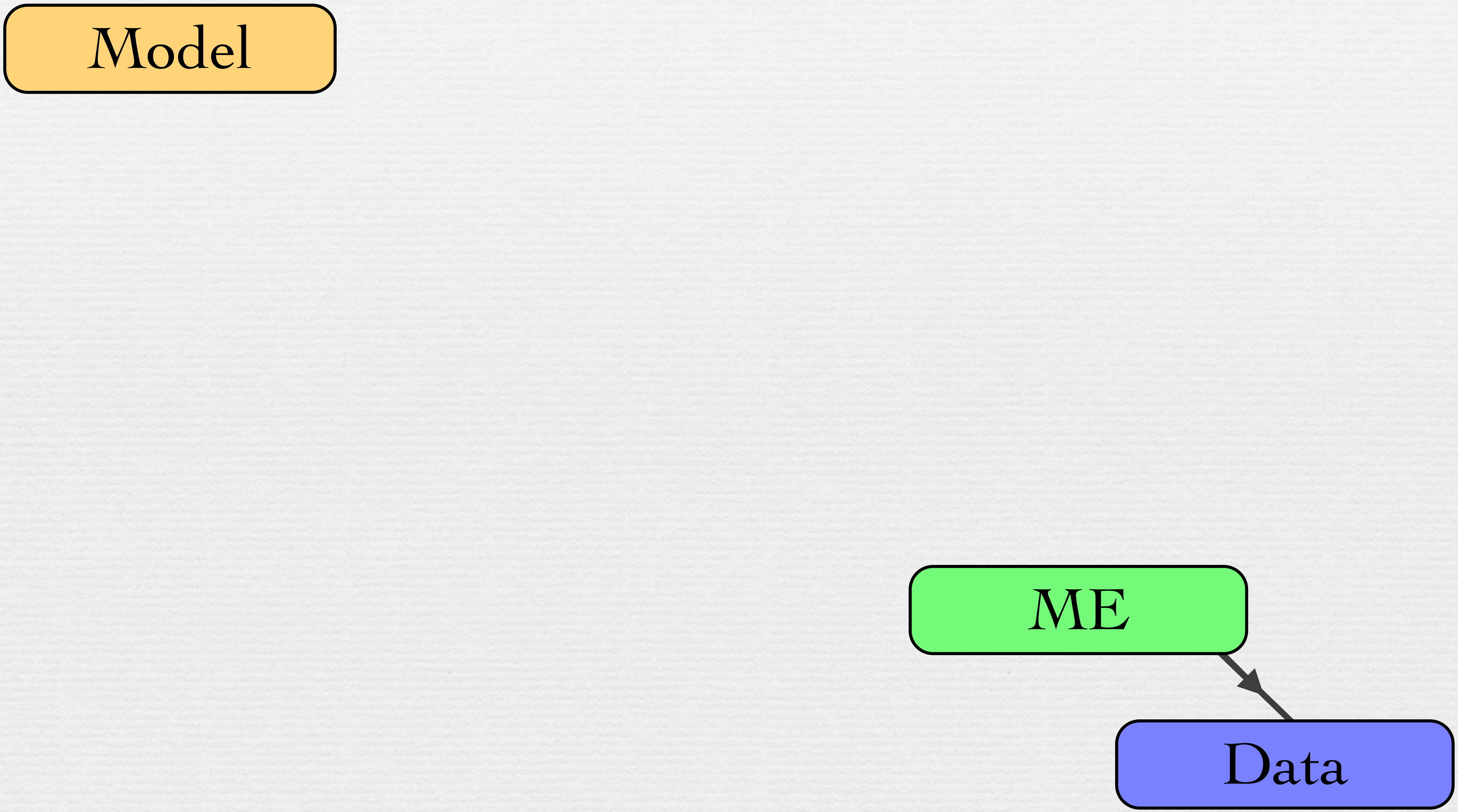
- Plan: have a tool (not in Mathematica) that sets up 1-loop RGE's, and generates the low-scale inputs form the high-scale inputs, at least for some classes of models.
  - ➡ How do deal efficiently with the boudary conditions at different scales?
  - ➡ How to deal with the decoupling of heavy particles (DRbar counterterms are mass independent)?

From theory to phenomenology

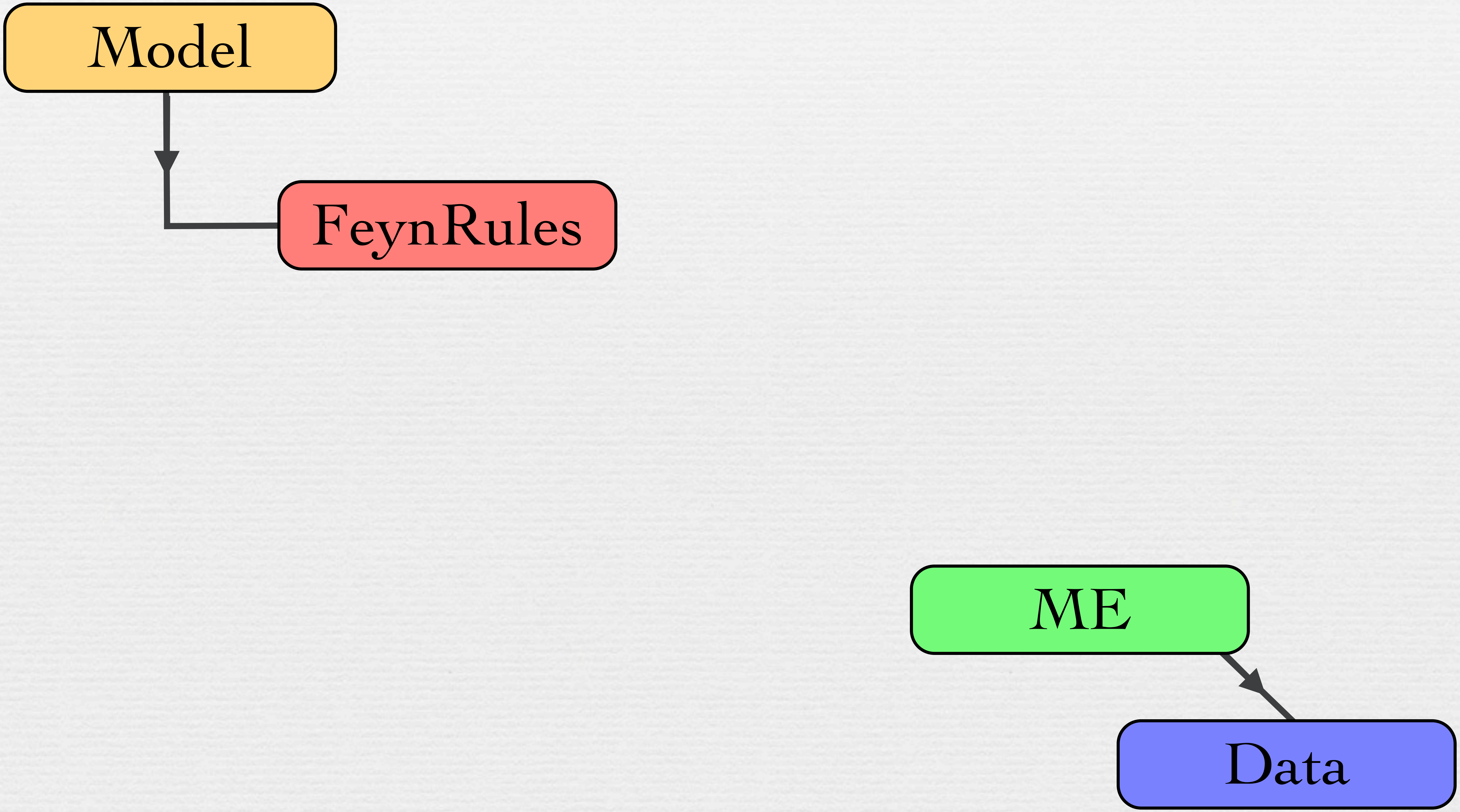
Model

Data

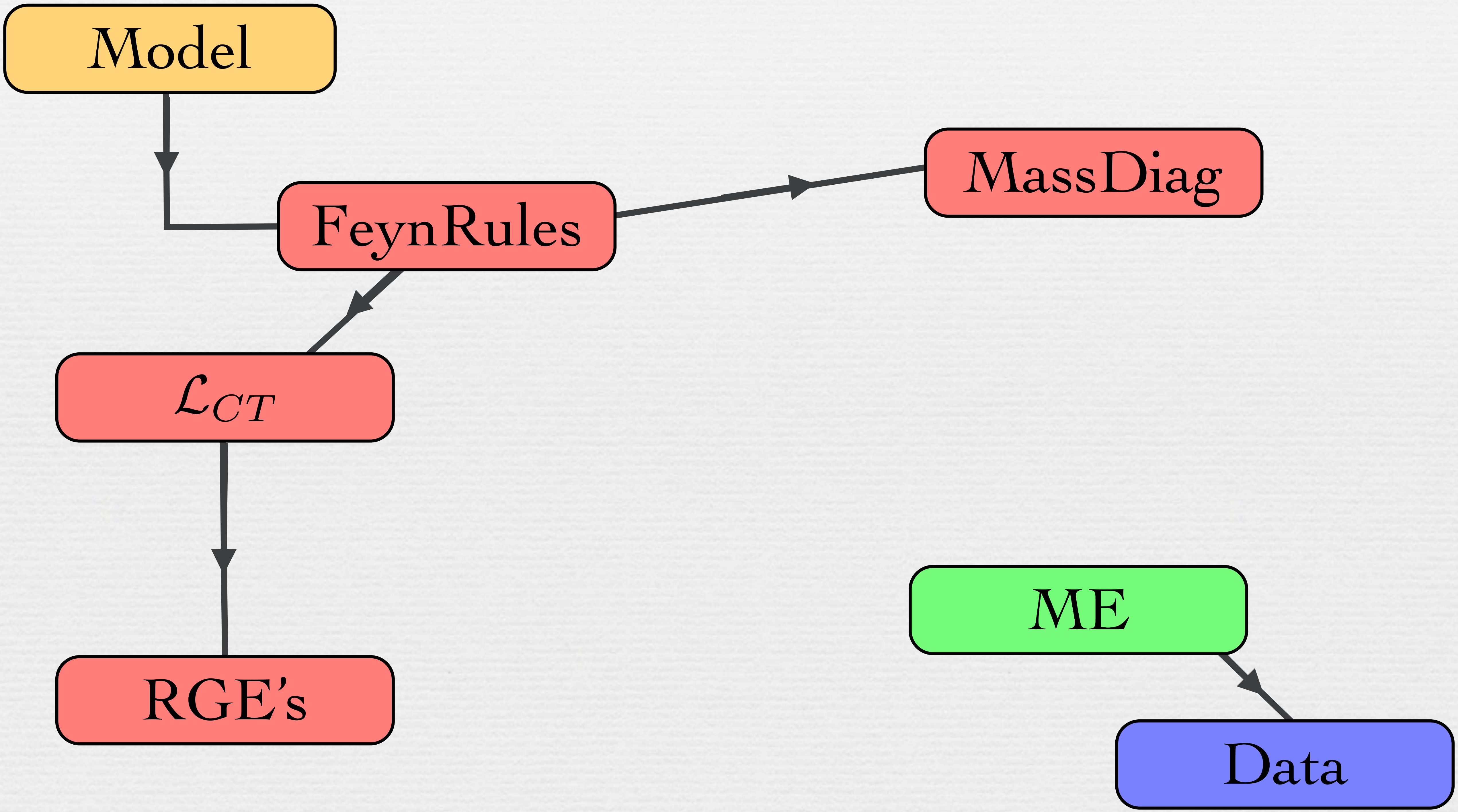
From theory to phenomenology



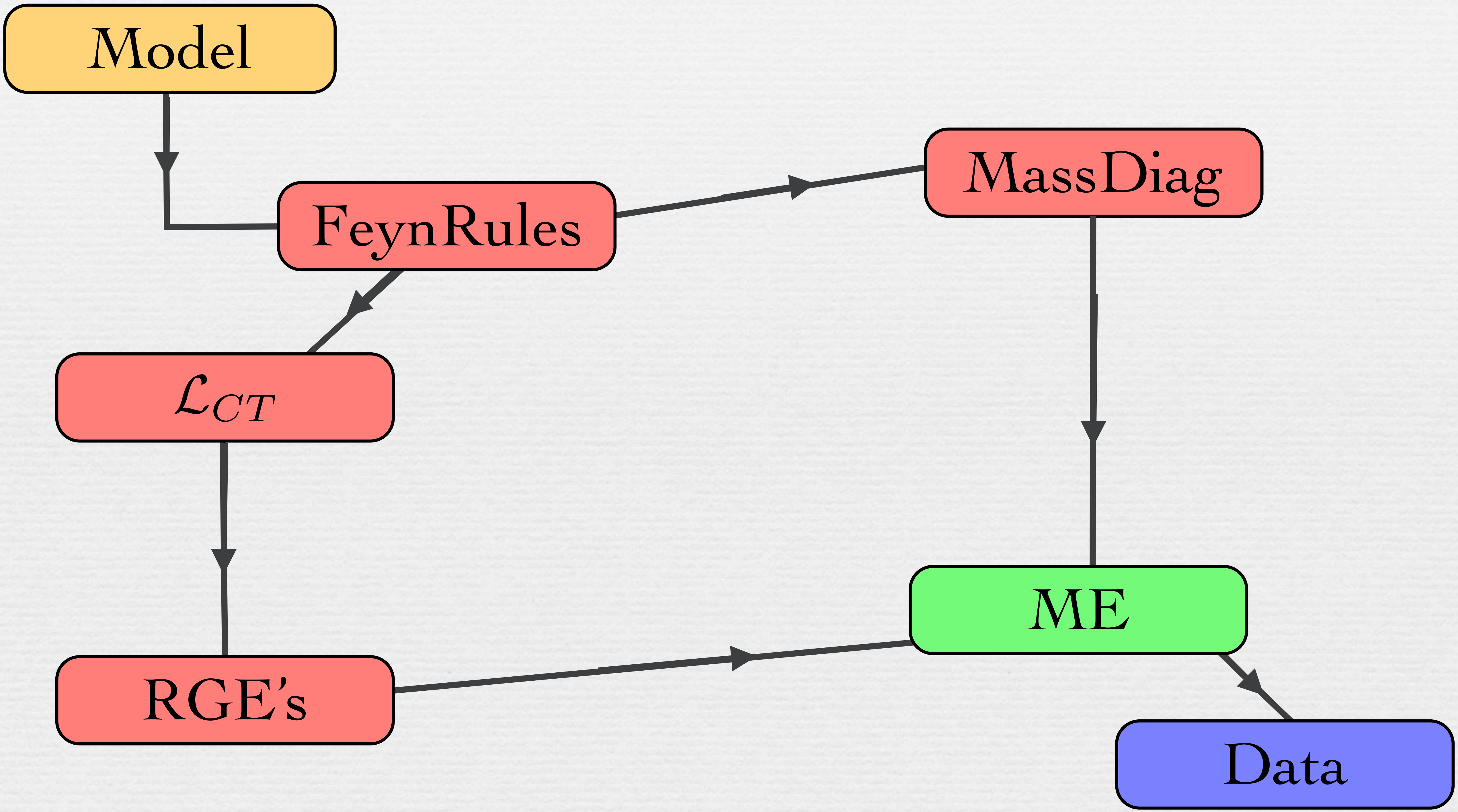
From theory to phenomenology



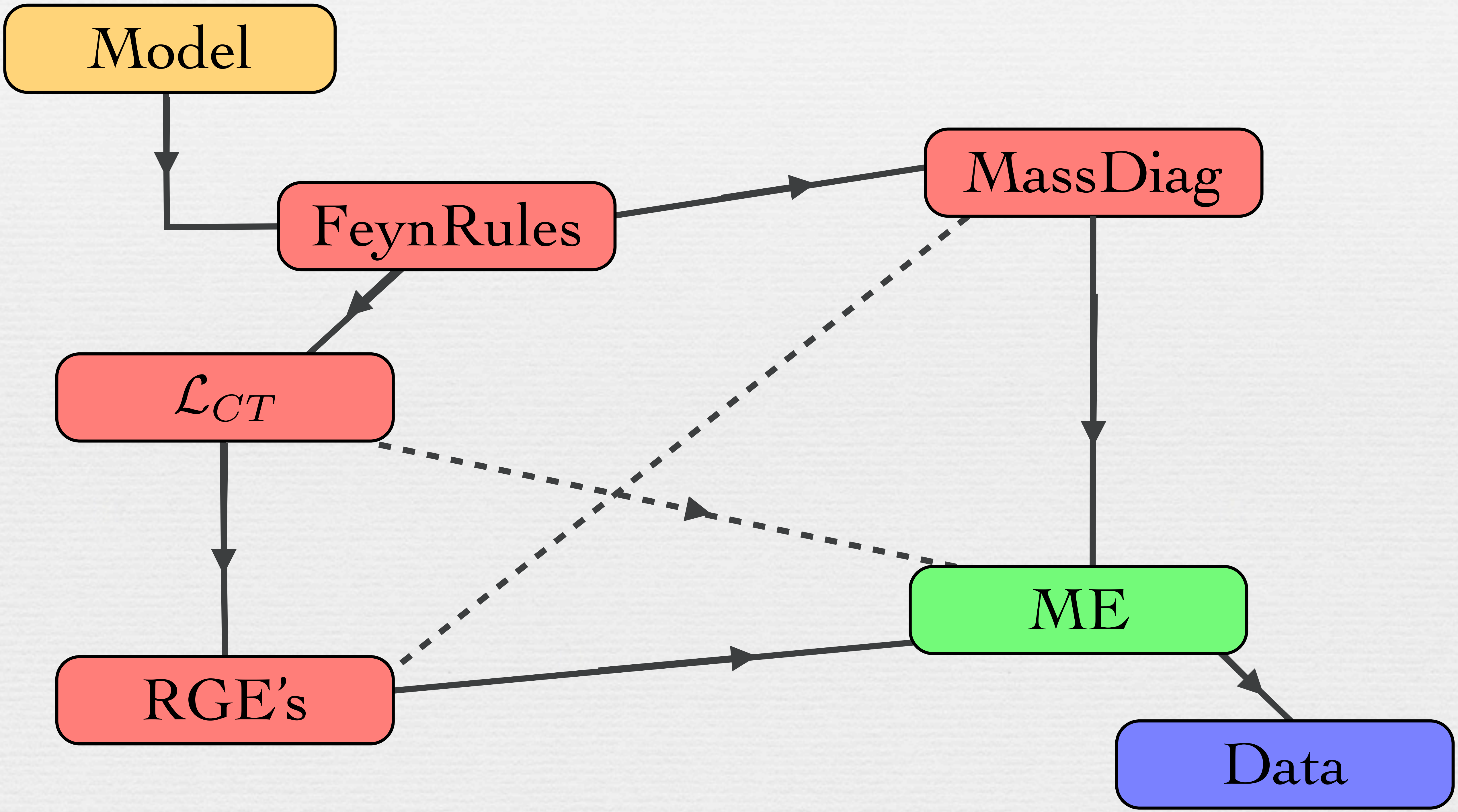
From theory to phenomenology



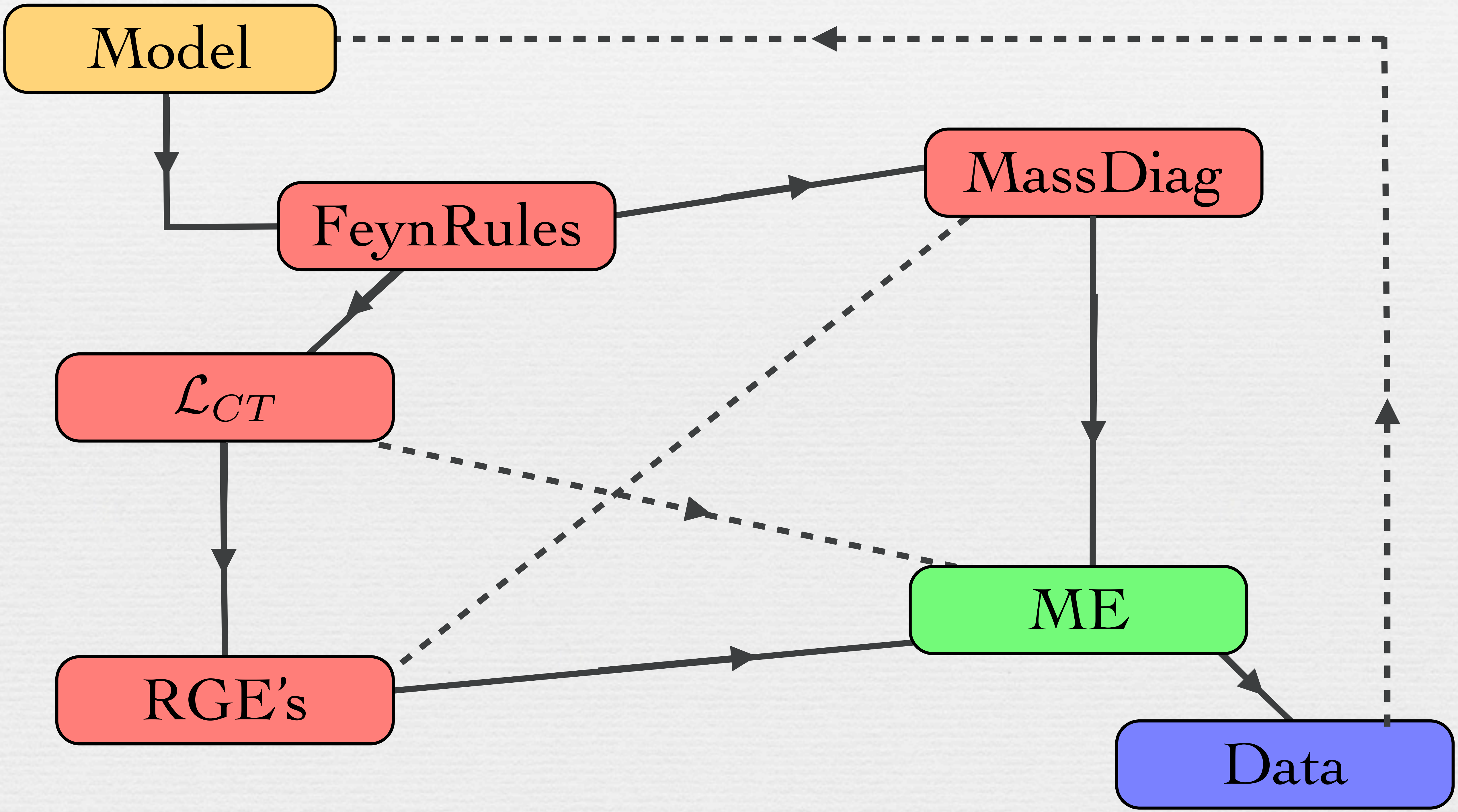
From theory to phenomenology



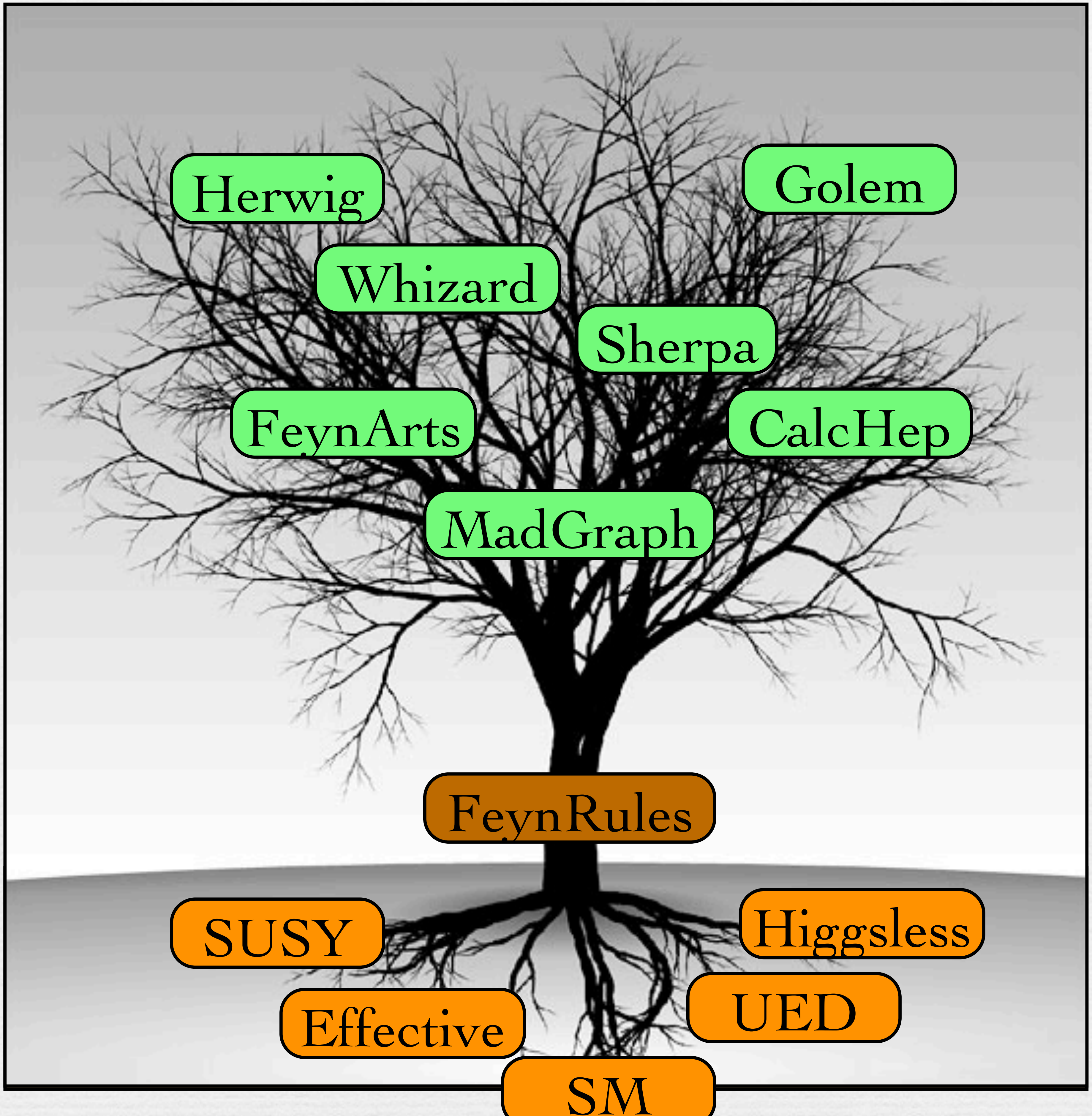
From theory to phenomenology

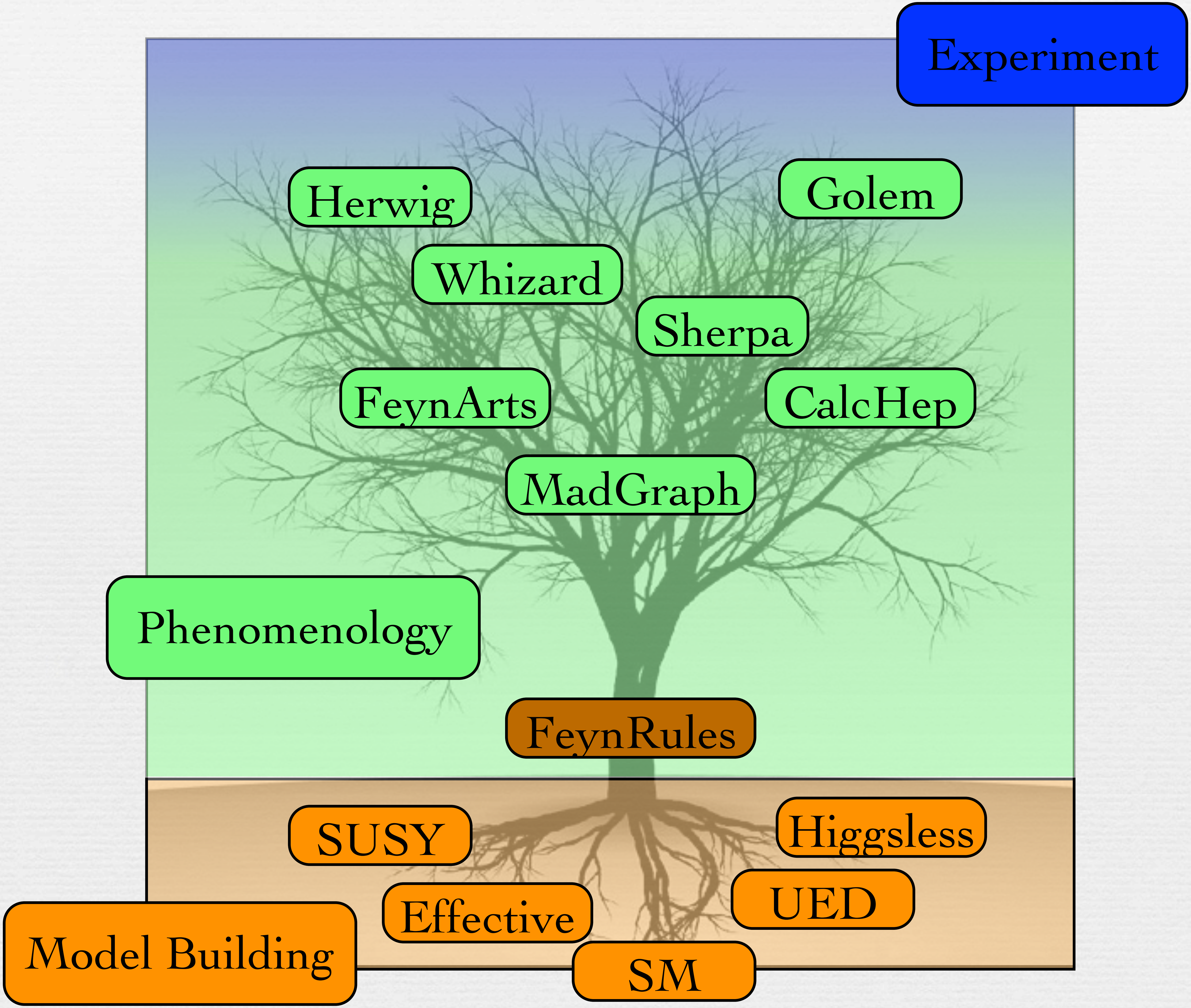


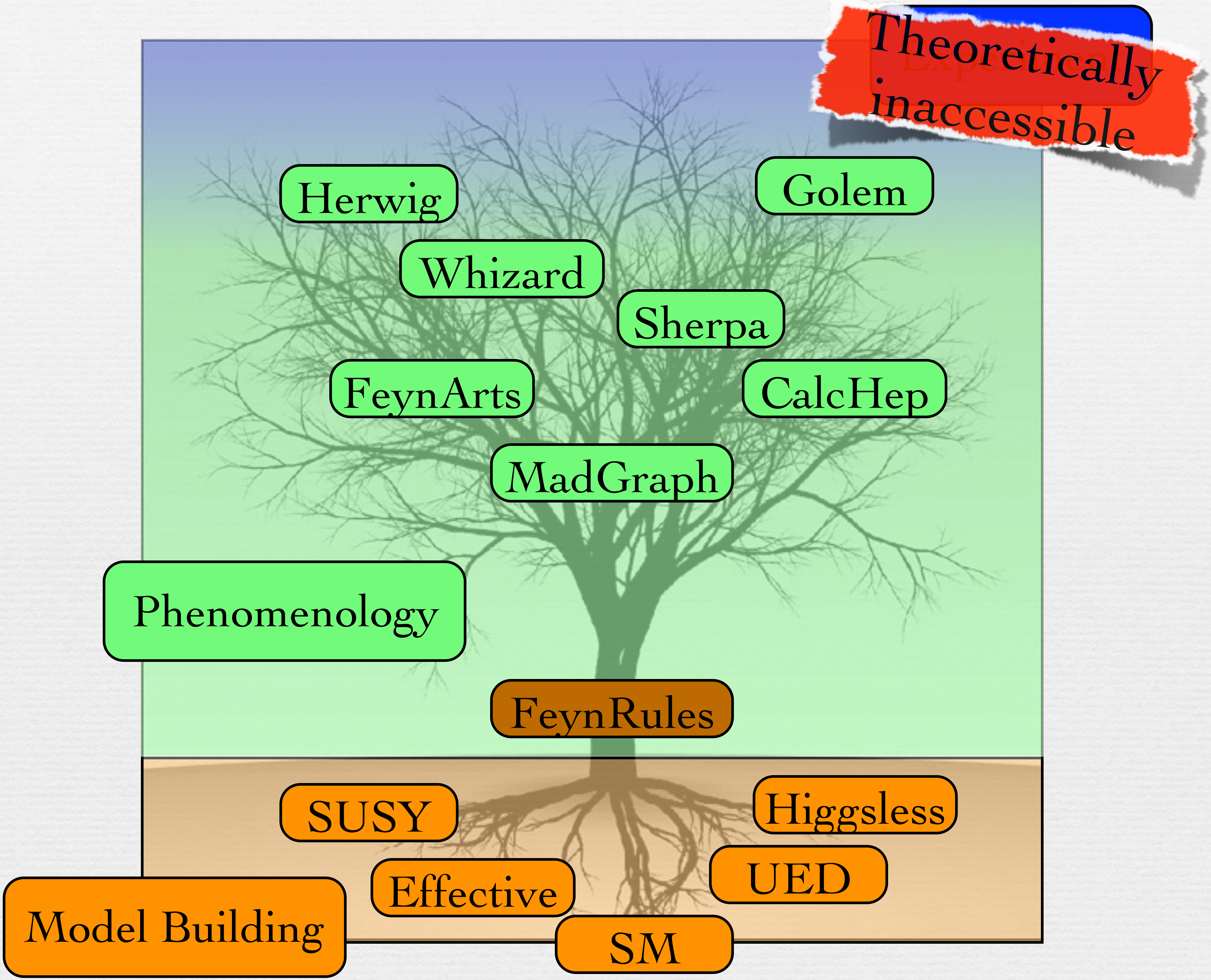
From theory to phenomenology



Towards a communication platform between theorists and experimentalists


Herwig
Golem
Whizard
Sherpa
FeynArts
CalcHep
MadGraph
FeynRules
SUSY
Higgsless
UED
Effective
SM


Experiment
Herwig
Golem
Whizard
Sherpa
FeynArts
CalcHep
MadGraph
Phenomenology
FeynRules
SUSY
Higgsless
UED
Effective
Model Building
SM

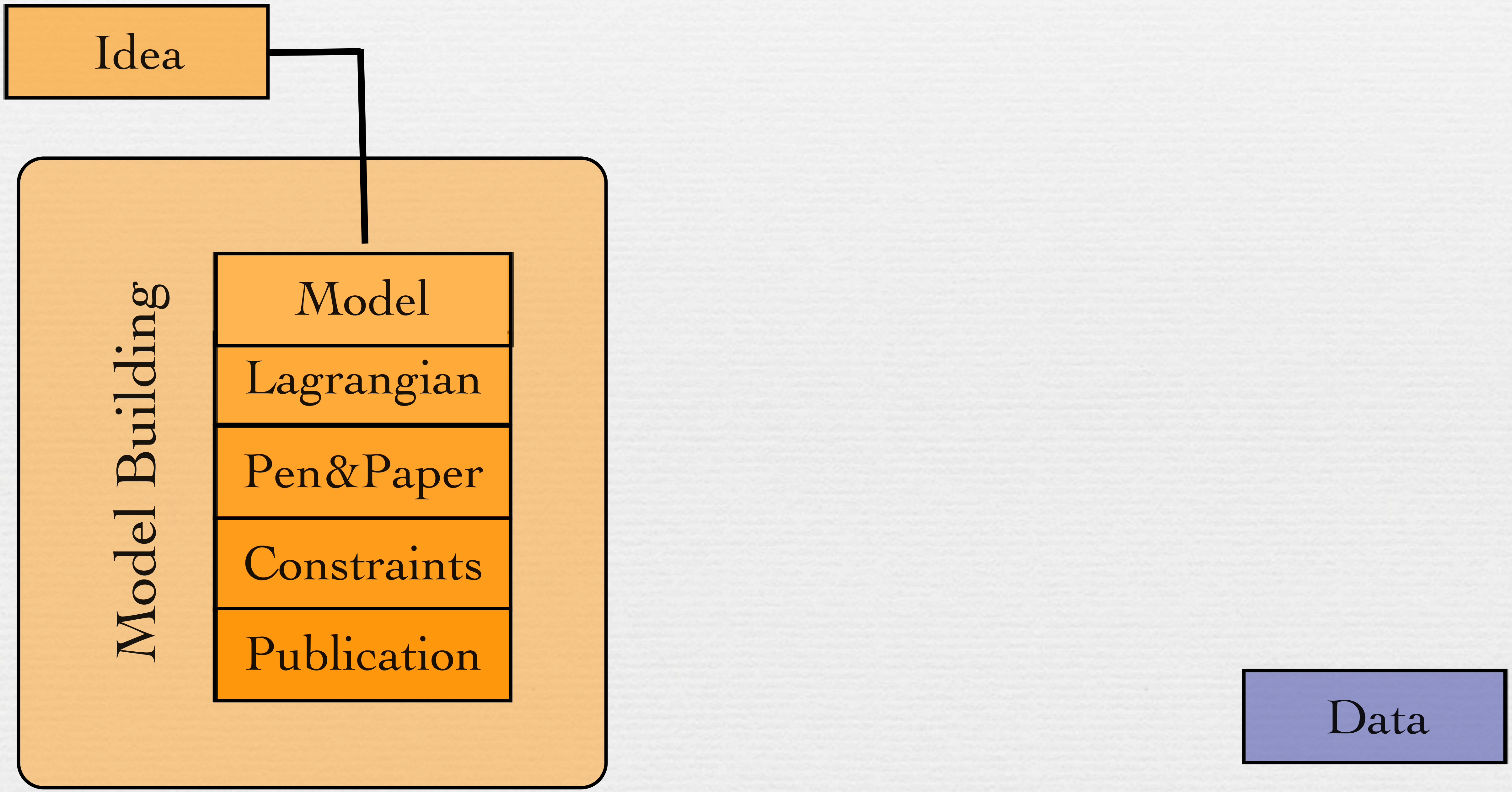

Theoretically inaccessible
Herwig
Golem
Whizard
Sherpa
FeynArts
CalcHep
MadGraph
Phenomenology
FeynRules
SUSY
Higgsless
UED
Effective
Model Building
SM

A Roadmap for BSM @ the LHC

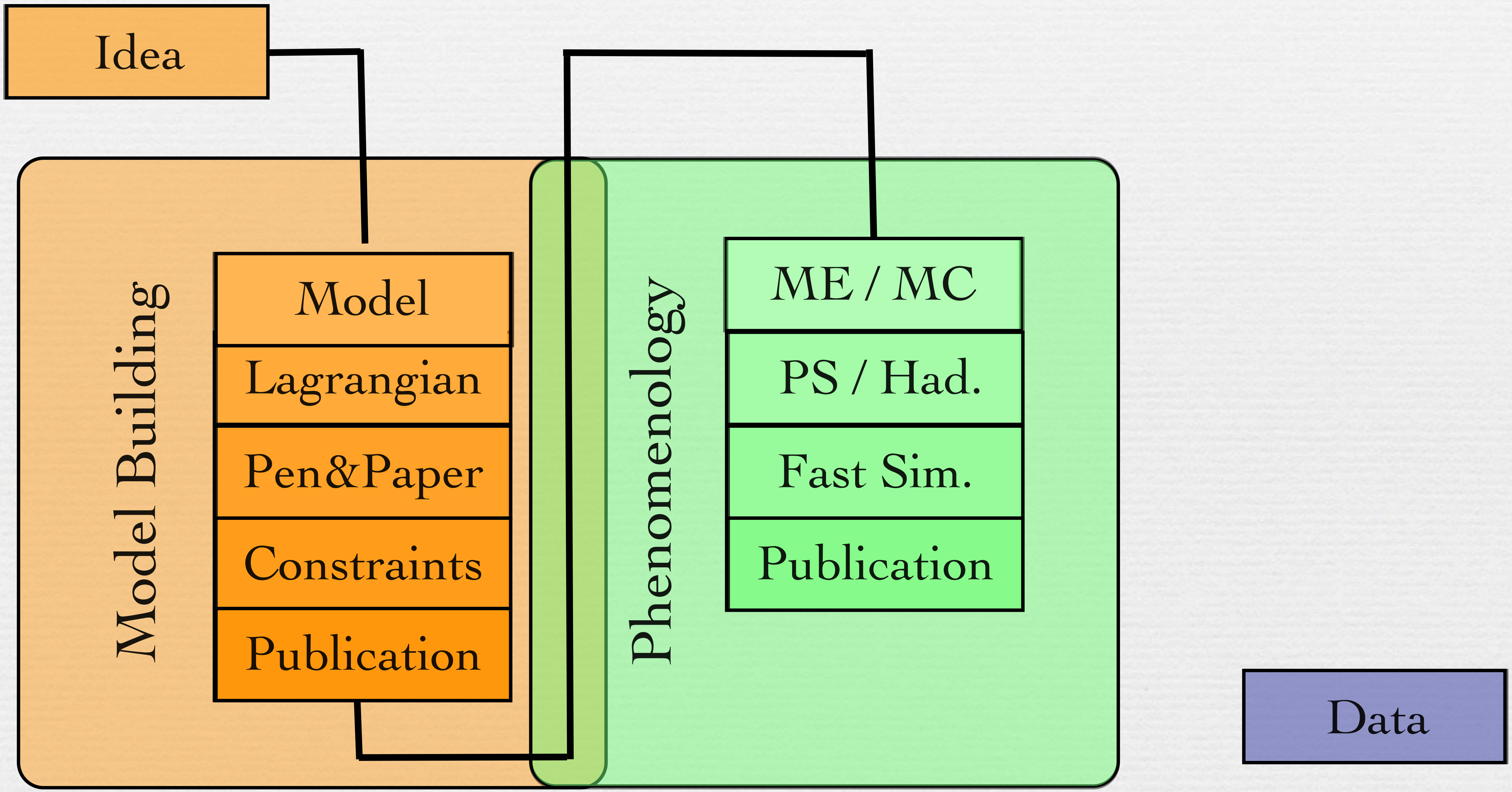
Idea

Data

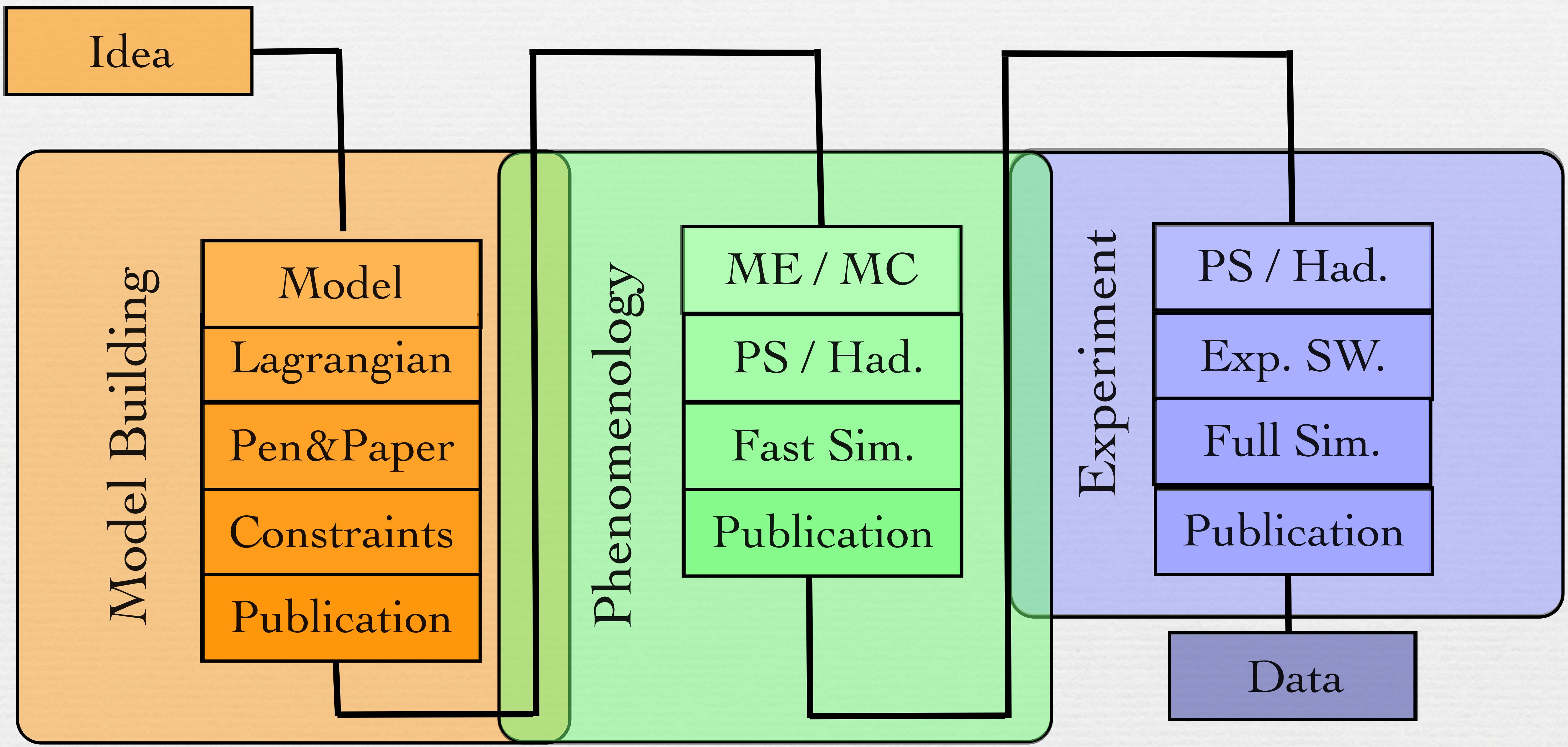
A Roadmap for BSM @ the LHC



A Roadmap for BSM @ the LHC

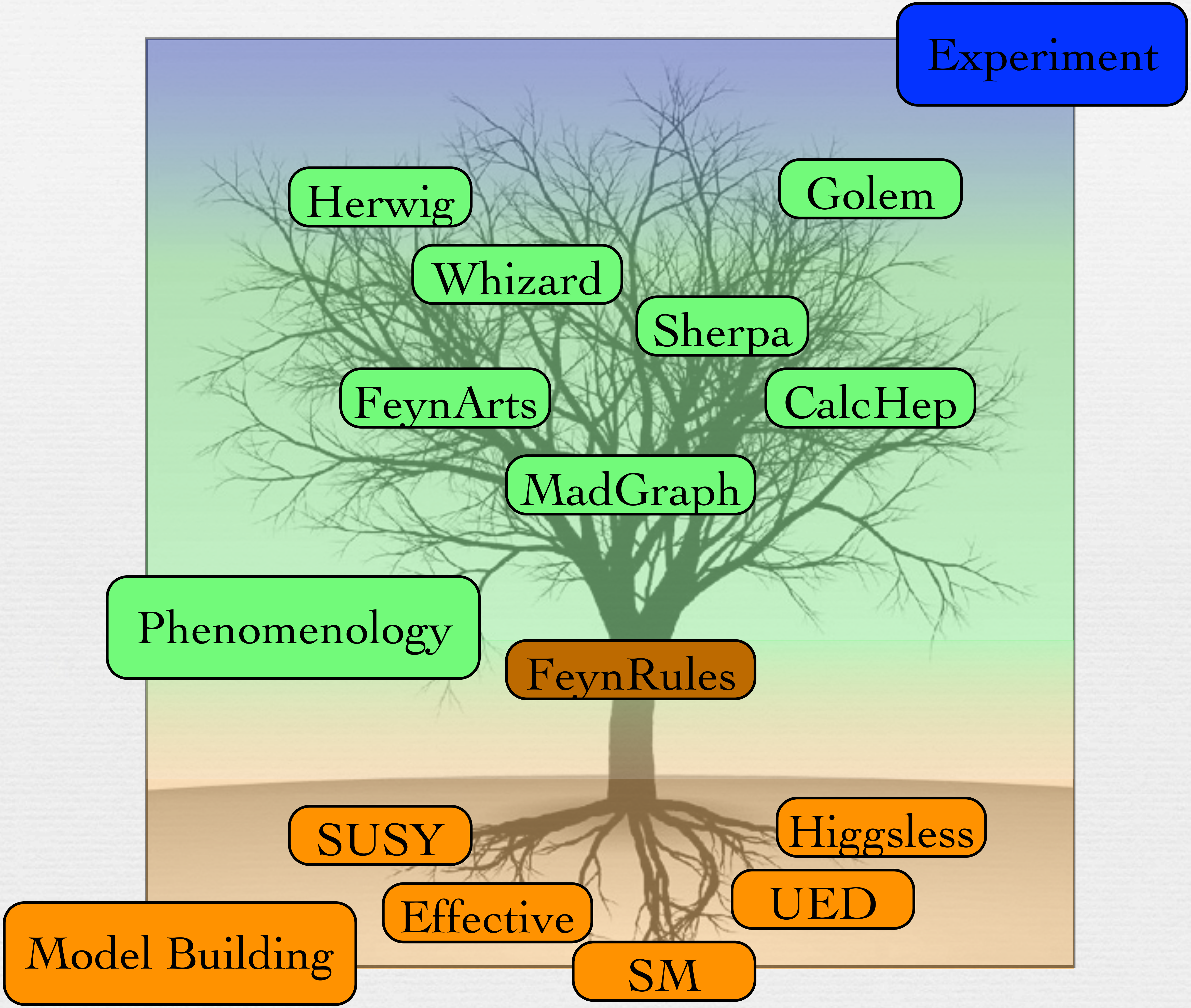


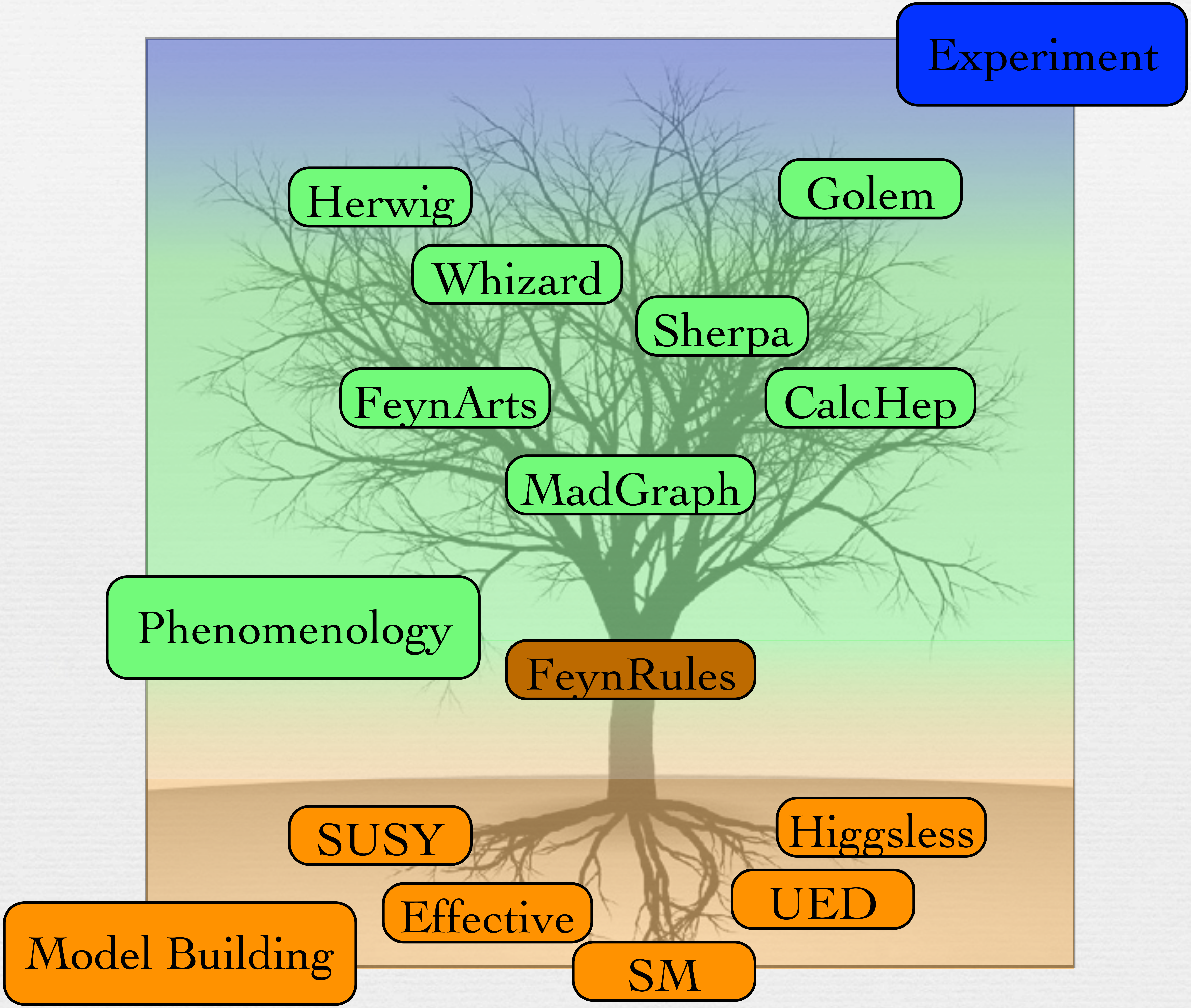
A Roadmap for BSM @ the LHC



A Roadmap for BSM @ the LHC

- Workload is tripled, due to disconnected fields of expertise.
- Error-prone, painful validation at each step.
- Proliferation of private MC's/Pythia tunings:
  - ➡ No clear documentation.
  - ➡ Not traceable.
- We need more than just papers to communicate between theorists and experimentalists!


Experiment
Herwig
Golem
Whizard
Sherpa
FeynArts
CalcHep
MadGraph
Phenomenology
FeynRules
SUSY
Higgsless
UED
Effective
Model Building
SM


Experiment
Herwig
Golem
Whizard
Sherpa
FeynArts
CalcHep
MadGraph
Phenomenology
FeynRules
SUSY
Higgsless
UED
Effective
Model Building
SM

A Roadmap for BSM @ the LHC

- MC's are already integrated into the experimental framework:
  - ➡ no re-validation required!
- All the information about the physics content of the implementation is centered where it belongs, in the Lagrangian
  - ➡ full traceability of all event samples
  - ➡ possibility to create web database for BSM models
- Compatibility with various MC's
  - ➡ Unseen validation power

The Les Houches validation scheme

- ★ Documentation:

  References to the original papers,operating system, ...

- ★ Basic theory sanity checks:

  Hermiticity, gauge invariance, 2-to-2 cross section,...

- ★ Testing one ME generator:

  All possible 2-to-2 cross sections, in different gauges, HE behavior, ...

- ★ Testing several ME generators

Process	MG-FR	MG-ST	CH-FR	CH-ST	SH-FR	SH-ST	WO-FR	WO-ST	Comparison
e+,e->sd1,sd1-	2.85002×10^{-3}	2.85011×10^{-3}	2.8501×10^{-3}	2.8501×10^{-3}	2.85007×10^{-3}	2.85007×10^{-3}	2.85013×10^{-3}	2.85013×10^{-3}	δ = 0.00394796 %
e+,e->sd2,sd2-	4.34049×10^{-4}	4.34207×10^{-4}	4.3415×10^{-4}	4.3415×10^{-4}	4.34145×10^{-4}	4.34145×10^{-4}	4.34155×10^{-4}	4.34155×10^{-4}	δ = 0.0364994 %
e+,e->sd1,sd2-	2.85795×10^{-4}	2.85759×10^{-4}	2.8578×10^{-4}	2.8579×10^{-4}	2.85825×10^{-4}	2.85825×10^{-4}	2.8579×10^{-4}	2.8579×10^{-4}	δ = 0.0229397 %
e+,e->n1,n1	7.45909×10^{-2}	7.45813×10^{-2}	7.4637×10^{-2}	7.4637×10^{-2}	7.46268×10^{-2}	7.46266×10^{-2}	7.463×10^{-2}	7.46338×10^{-2}	δ = 0.0746855 %
e+,e->n1,n2	2.5541×10^{-2}	2.55366×10^{-2}	2.5555×10^{-2}	2.5555×10^{-2}	2.55523×10^{-2}	2.55516×10^{-2}	2.55521×10^{-2}	2.55535×10^{-2}	δ = 0.0719985 %
e+,e->n1,n3	2.08218×10^{-3}	2.08034×10^{-3}	2.081×10^{-3}	2.081×10^{-3}	2.08093×10^{-3}	2.08089×10^{-3}	2.0811×10^{-3}	2.081×10^{-3}	δ = 0.0880299 %
e+,e->n1,n4	3.73046×10^{-3}	3.73254×10^{-3}	3.7325×10^{-3}	3.7325×10^{-3}	3.73208×10^{-3}	3.7321×10^{-3}	3.73223×10^{-3}	3.73238×10^{-3}	δ = 0.0555803 %

Conclusion

- If we need to decide between many competing BSM models at the LHC, a new way of communicating between theorists and experimentalists is needed.
- In such a framework theorists and experimentalists can meet on a common 'platform', that offers a flexible environment how a model can be developed and extended and its phenomenology studied.
- This framework does not only allow the full traceability and reproducibility of all event samples, but also the validation of the models to an unprecedented level.