Durham University

IP3

FeynRules

Claude Duhr

in collaboration with N. D. Christensen and B. Fuks

+ P. de Aquino, C. Degrande, D. Grellscheid, W. Link, F. Maltoni,
+ O. Mattelaer, T. Reiter, C. Speckner, S. Schumann, M. Wiebusch

Tools 2010

Winchester, June 30th 2010

Outline

- What is FeynRules..?
- New developments:
  - ➡ New interfaces (FeynArts, Whizard, UFO)
  - ➡ Support for Weyl fermions and superfields
  - ➡ Diagonalization of mass matrices
- Live demonstration

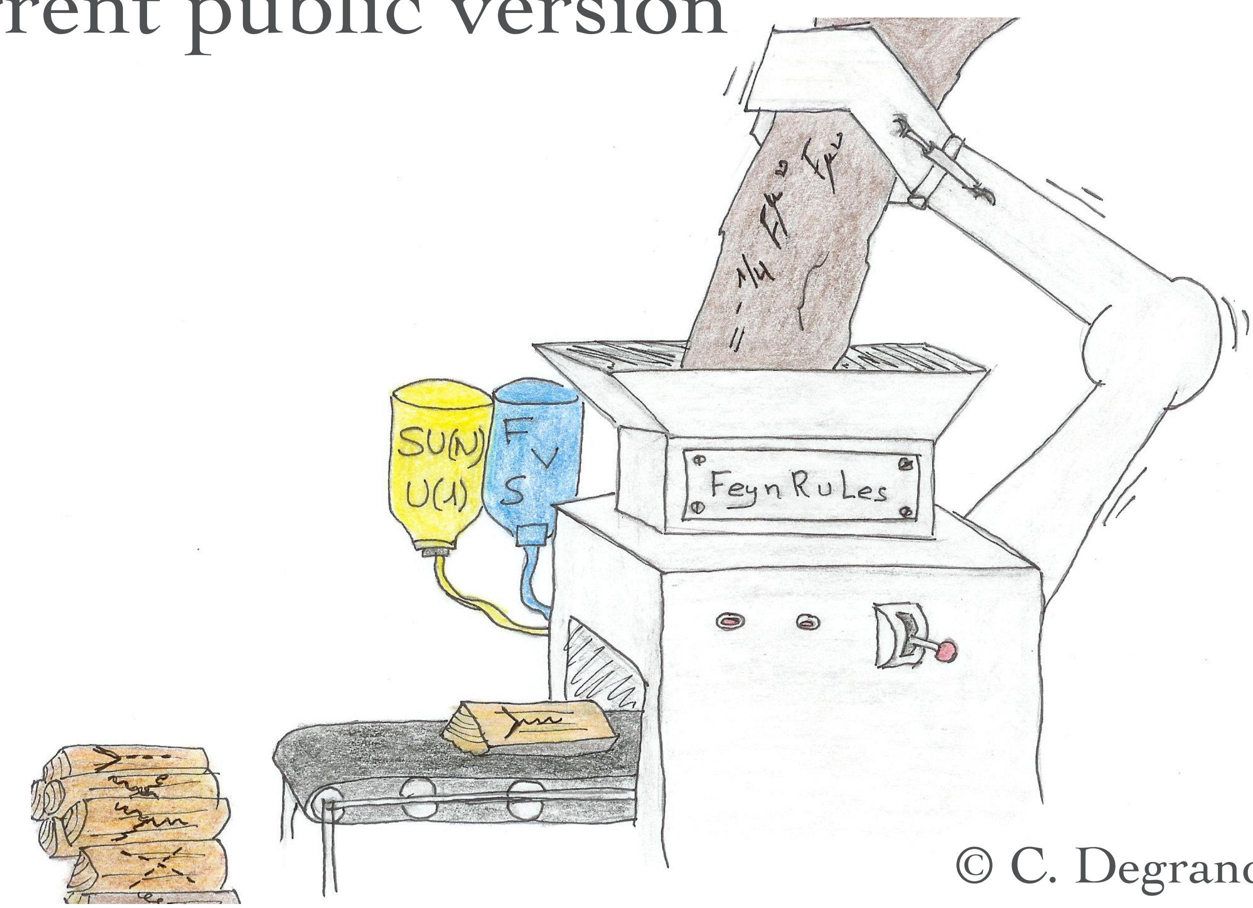
What is FeynRules..?

- FeynRules is a Mathematica package that allows to derive Feynman rules from a Lagrangian.
- Current public version: 1.4.9, available from

  http://feynrules.phys.ucl.ac.be
- The only requirements on the Lagrangian are:
  - ➡ All indices need to be contracted (Lorentz and gauge invariance)
  - ➡ Locality
  - ➡ Supported field types: spin 0, 1/2, 1, 2 & ghosts

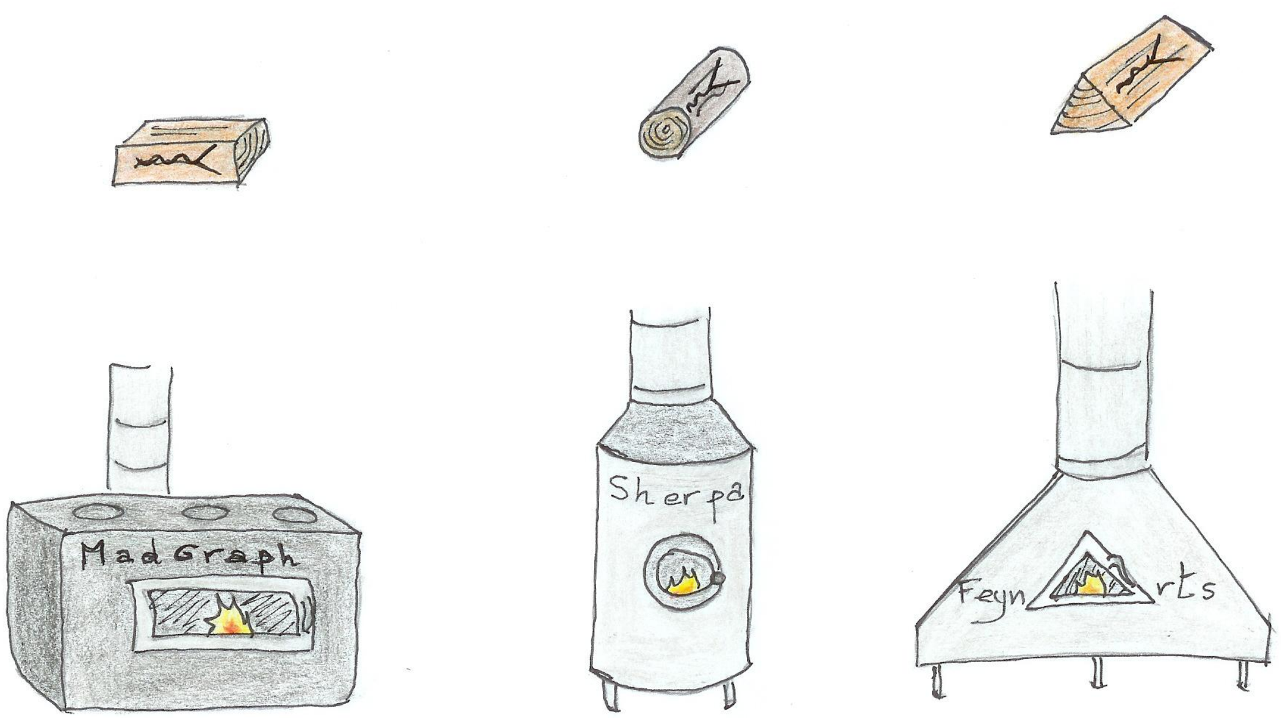
What is FeynRules..?

- FeynRules comes with a set of interfaces, that allow to export the Feynman rules to various matrix element generators.
- Interfaces coming with current public version
  - ➡ CalcHep / CompHep
  - ➡ FeynArts / FormCalc
  - ➡ MadGraph 4
  - ➡ Sherpa
  - ➡ Whizard / Omega



What is FeynRules..?

- FeynRules comes with a set of interfaces, that allow to export the Feynman rules to various matrix element generators.
- Interfaces coming with current public version:
  - ➡ CalcHep / CompHep
  - ➡ FeynArts / FormCalc
  - ➡ MadGraph 4
  - ➡ Sherpa
  - ➡ Whizard / Omega



How to use FeynRules

- The input requested form the user is twofold.

- The Model File:
  Definitions of particles and parameters (e.g., a quark)

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

- The Lagrangian:

$$\mathcal{L} = -\frac{1}{4} G^a_{\mu\nu}\, G_a^{\mu\nu} + i\bar{q}\,\gamma^\mu\, D_\mu q - M_q\, \bar{q}\, q$$

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

How to use FeynRules

- Once this information has been provided, FeynRules can be used to compute the Feynman rules for the model:

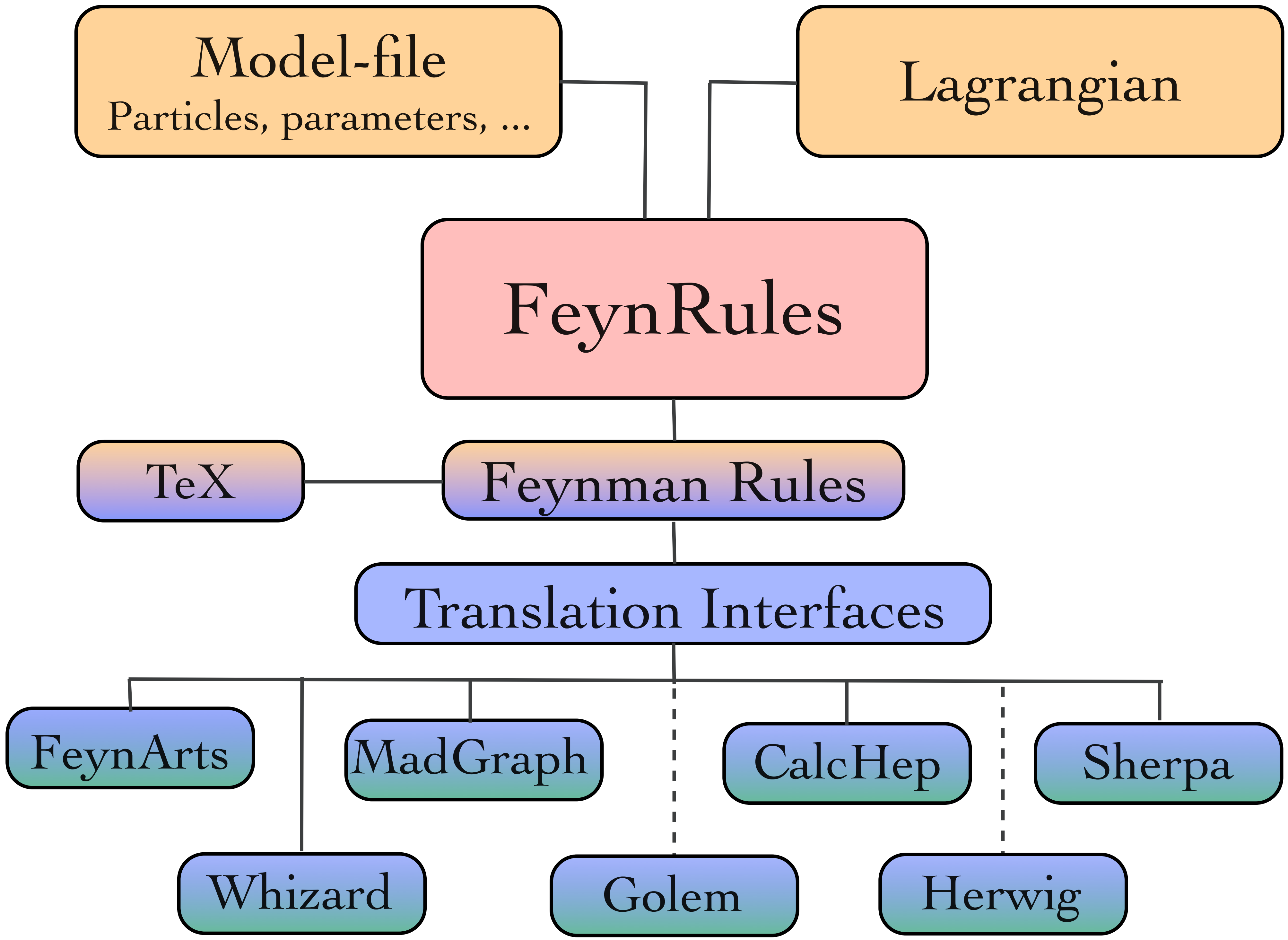
FeynmanRules[L]

- Equivalently, we can export the Feynman rules to a matrix element generator, e.g., for MadGraph 4,

WriteMGOutput[L]

- This produces a set of files that can be directly used in the matrix element generator ("plug 'n' play").

FeynRules



Model-file
Particles, parameters, ...
Lagrangian
FeynRules
TeX
Feynman Rules
Translation Interfaces
FeynArts
MadGraph
CalcHep
Sherpa
Whizard
Golem
Herwig

New Developments

- In spring 2010, we locked up 13 experts in a room in a nice monastry for 5 days, and let them gather new ideas...
- As a result, many new projects got started (in fact too many to review them all here...)

Superfields (B. Fuks)

- In the future, FeynRules will allow the use of superfields.

- Example: Superpotential for left-handed quarks

$$\mathcal{W} = a_i Q_{Li} + M_{ij} Q_{Li} Q_{Lj} + \frac{1}{6}\lambda_{ijk} Q_{Li} Q_{Lj} Q_{Lk}$$

```
W = a[i] QL[i] + 1/2*M[i, j] QL[i] QL[j] + 1/6*l[i, j, k] QL[i] QL[j] QL[k]
```

- FeynRules then converts the superfields into component fields:

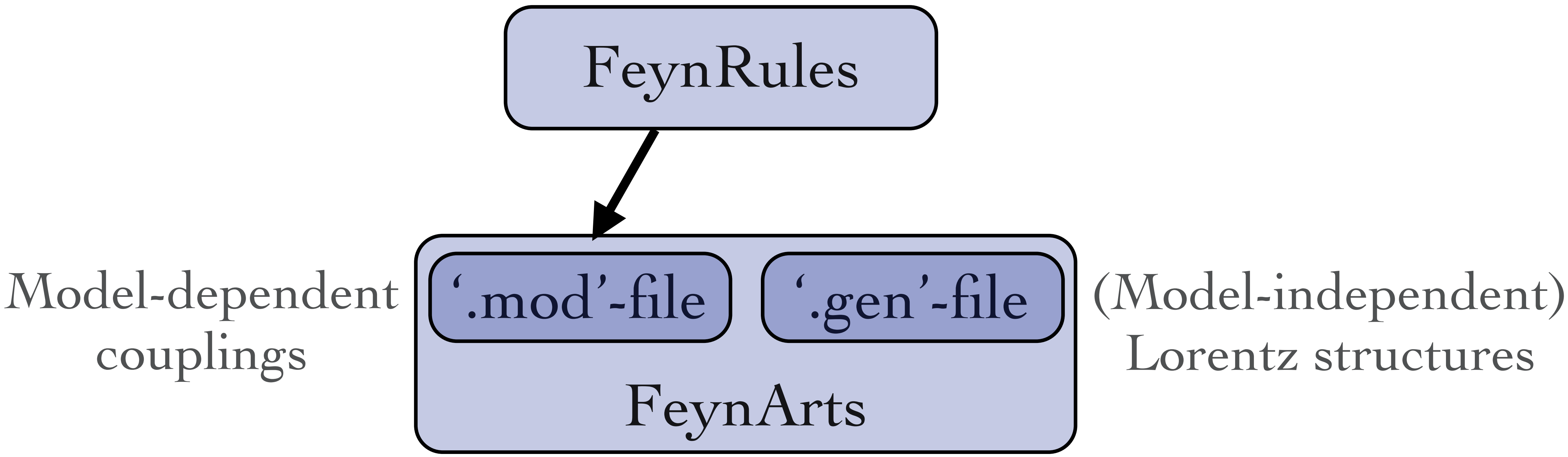
```
SF2Components[ W ]
```

$$-\frac{1}{6}\,\text{FTerm3}_i\,\text{sqL}_j\,\text{sqL}_k\,l_{i,j,k} - \frac{1}{6}\,\text{FTerm3}_k\,\text{sqL}_i\,\text{sqL}_j\,l_{i,j,k} - \frac{1}{6}\,\text{FTerm3}_j\,\text{sqL}_i\,\text{sqL}_k\,l_{i,j,k}$$

$$-\frac{1}{6}\,\text{sqL}_j\,l_{i,j,k}\,\text{qL}_{\text{sp\$2},i}.\text{qL}_{\text{sp\$2},k} - \frac{1}{6}\,\text{sqL}_i\,l_{i,j,k}\,\text{qL}_{\text{sp\$2},j}.\text{qL}_{\text{sp\$2},k} - \frac{1}{2}\,M_{i,j}\,\text{qL}_{\text{sp\$2},i}.\text{qL}_{\text{sp\$2},j}$$

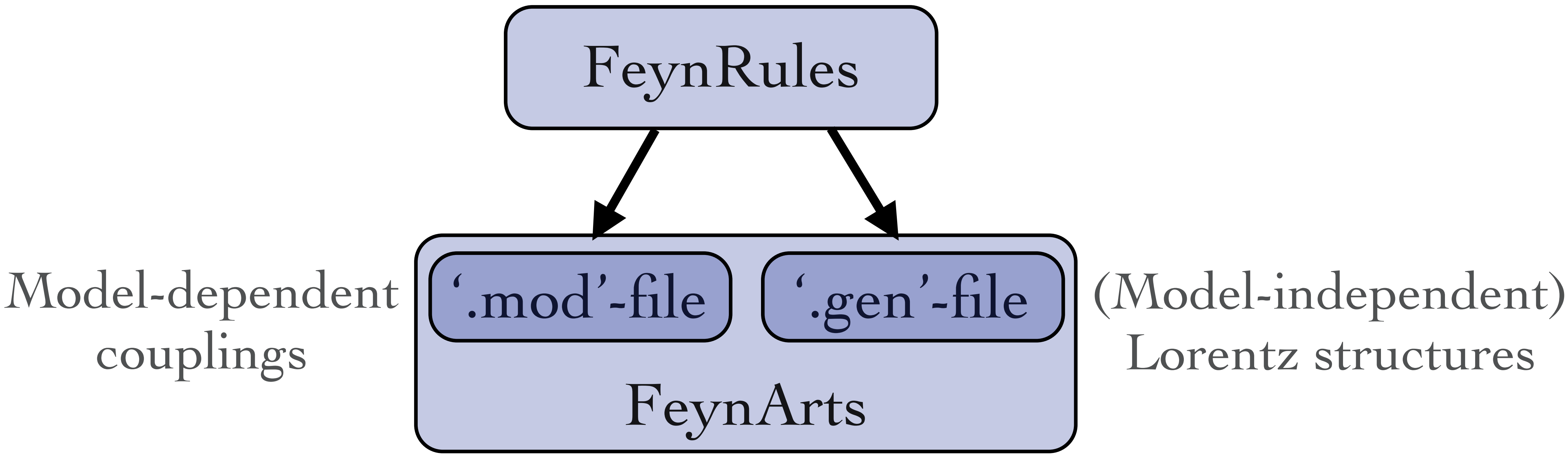
FeynArts interface (C. Degrande, CD)

- A new interface to FeynArts is being developped that allows to implement arbitrary Lorentz structures.



FeynArts interface (C. Degrande, CD)

- A new interface to FeynArts is being developped that allows to implement arbitrary Lorentz structures.



- This development goes along with a new version of FormCalc able to deal with multi-fermion interactions.

Mass Matrix diagonalization (N. Christensen, M. Wiebusch)

- In the future, FeynRules will perform the diagonalization automatically.

Input 1: Lagrangian

qLbar[s1,l1,f1,i].uR[s1,f2,i] Yu[f1,f2] Eps[l1,l2] Phibar[l2]

Input 2: Mixing relations

uq[s, #, o] == CKMU.uqp[s, #, o]

Output 1: Mass matrix to be diagonalized

$$\text{Diagonalization}(\text{HEigensystem}, \text{CKMU}, \{m_u, m_c, m_t\}) \to \begin{pmatrix} -\frac{v\, Yu_{1,1}}{2\sqrt{2}} & -\frac{v\, Yu_{1,2}}{2\sqrt{2}} & -\frac{v\, Yu_{1,3}}{2\sqrt{2}} \\ -\frac{v\, Yu_{2,1}}{2\sqrt{2}} & -\frac{v\, Yu_{2,2}}{2\sqrt{2}} & -\frac{v\, Yu_{2,3}}{2\sqrt{2}} \\ -\frac{v\, Yu_{3,1}}{2\sqrt{2}} & -\frac{v\, Yu_{3,2}}{2\sqrt{2}} & -\frac{v\, Yu_{3,3}}{2\sqrt{2}} \end{pmatrix}$$

Output 2: 'Rotation rules'

$$(P_+)_{s1,s} \left(\text{CKMU}(1,1)^* \, uq_{s,1,o} + \text{CKMU}(2,1)^* \, uq_{s,2,o} + \text{CKMU}(3,1)^* \, uq_{s,3,o}\right)$$

- To be done: numerical code for the diagonlization.

The UFO

(P. de Aquino, CD, D. Grellscheid, W. Link, O. Mattelaer, T. Reiter)

UFO = Universal FeynRules Output

- Idea: Create Python modules that can be linked to other codes and contain all the information on a given model.
- The UFO is a self-contained Python code, and not tied to a specific matrix element generator.
- Golem, MadGraph 5 and Herwig++ will use the UFO.
- The development of the UFO goes hand in hand with the development of ALOHA (Automatic Language-independent Output of Helicity Amplitudes), a code that allows to create HELAS routines from the UFO.

Implemented models

- Standard Model* (CD, N. Christensen)
- Most general two Higgs doublet model* (CD, M. Herquet)
- Minimal Higgsless Model* (N. Christensen)

  * available at http://feynrules.phys.ucl.ac.be

- Validation of the models:

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 %

Implemented models: (Susy)

- Full MSSM* (B. Fuks)
- NMSSM (B. Fuks)
- R-symmetric MSSM (B. Fuks)
- RPV MSSM (B. Fuks)

* available at http://feynrules.phys.ucl.ac.be

Implemented models: (ED)

- Universal Extra Dimensions* (P. de Aquino)
- Large extra dimensions* (P. de Aquino)
- Randall-Sundrum I (P. de Aquino)

* available at http://feynrules.phys.ucl.ac.be

Implemented models: (Effective)

- Strongly interacting Little Higgs (C. Degrande)
- Composite Top model (C. Degrande)
- Chiral perturbation theory (C. Degrande)

* available at http://feynrules.phys.ucl.ac.be

Live demonstration

- Let us consider a simple model, just to get started...
- We will implement a model consisting of a Dirac octet ('Dirac gluino') decaying into triplet scalars ('squarks').

$$\mathcal{L} = \mathcal{L}_{SM} + \mathcal{L}_{gluino} + \mathcal{L}_{squark} + \mathcal{L}_{decay}$$

$$\mathcal{L}_{gluino} = i\bar{\tilde{g}}\gamma^\mu D_\mu \tilde{g} - M_G \bar{\tilde{g}}\tilde{g}$$

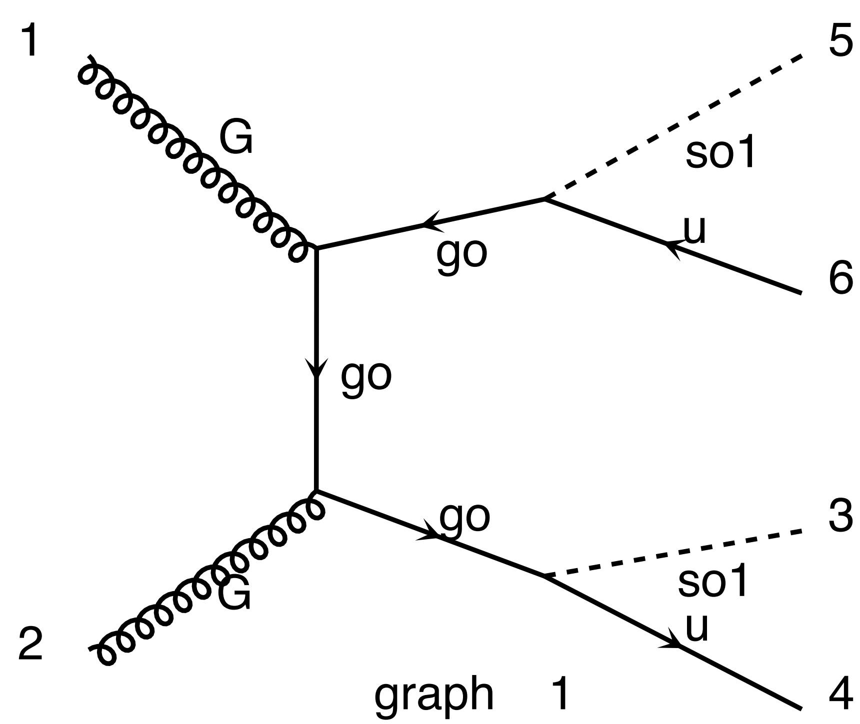
$$\mathcal{L}_{squark} = D^\mu \phi_k^\dagger D_\mu \phi_k - M_\phi^2 \phi_k^\dagger \phi_k$$

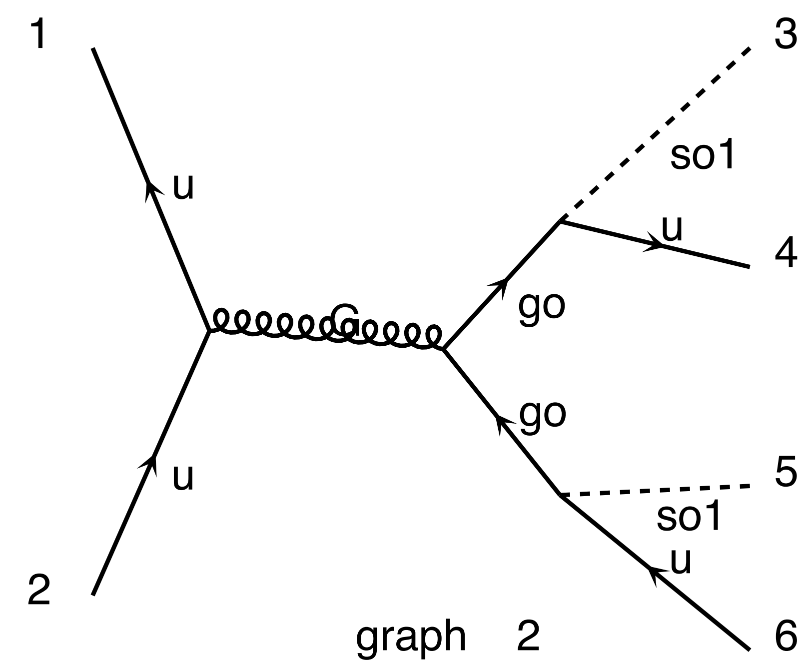
$$k = 1, 2, 3$$

$$\mathcal{L}_{decay} = g_s\, T^a_{ij}\, \bar{\tilde{g}}^a u_{R,jk}\, \phi^\dagger_{ik} + \text{h.c.}$$

	SU(3)	SU(2)	U(1)	Mass (GeV)
$\tilde{g}$	8	1	1	200
ϕ	3	1	2/3	150, 220, 250

Live demonstration





Towards a database of models...

https://moDel.org/1004.0123.html

CP3 FYMA Webmail FeynRules MadGraph Hotmail SPIRES LEO IPPP IP3 Webmail

[1004.0123] MSSM + Z'

moDel.org > SuperSym > moDel:1004.1424

Search or Article-id (Help | Advanced search) All papers Go!

Supersymmetric models

MSSM + Z'

Mr. X

(Submitted on 14 Apr 2010)

We present the FeynRules implementation of the extnsion of the MSSM with a Z' boson. This model was first presented in arXiv:1003.1234.

Comments:	FeynRules model file (3 files) + 2 benchmark points (2 files)
Subjects:	**MSSM - Extensions (SuperSym)**
Cite as:	**moDel:1004.0123v1 [SuperSym]**

Validation

This model implemetation is known to work with

CalcHep
Golem
Herwig
MadGraph
Sherpa

Results of the validation are available here .

Submission history

From: Mr. X [view email]
[v1] Wed, 14 Apr 2010 20:45:35 GMT (13kb)

Which authors of this paper are endorsers?

Download:

- Model files
- Benchmark Points
- Validation Tables
- Other formats

Current browse context:
SuperSym
< prev | next >
new | recent | 1004

References & Citations

- SLAC-SPIRES HEP
  (refers to | cited by)

Bookmark (what is this?)