# FeynRules.

Benjamin Fuks (IPHC Strasbourg / Université de Strasbourg)

In collaboration with N. Christensen, C. Degrande and C. Duhr.

Tools 2012 conference @ AlbaNova University Center, Stockholm
June 18-21, 2012

# Outline

1. FeynRules in a nutshell.
2. The Universal FeynRules output.
3. Supersymmetry developments.
4. Web validation.
5. Future development plans.
6. Conclusions.

# FeynRules in a nutshell.

- **A framework for LHC analyzes based on FeynRules to**:

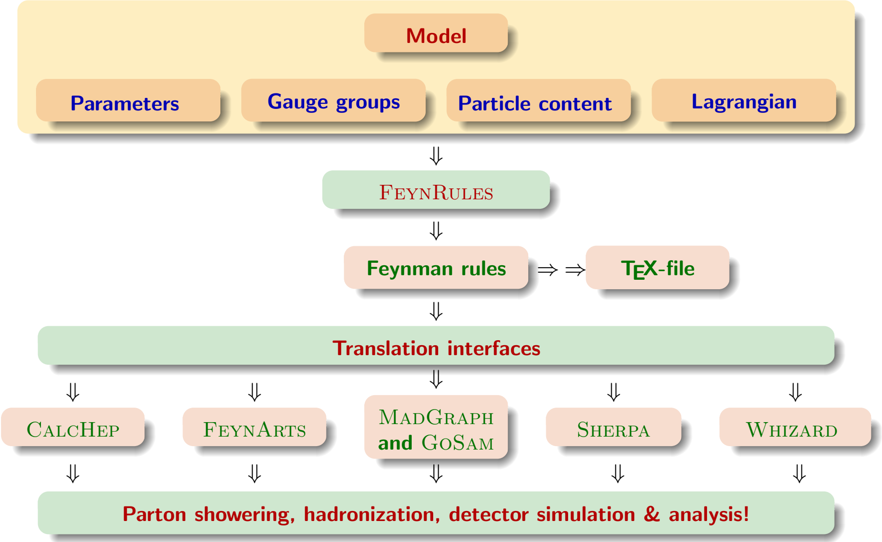
  [Christensen, de Aquino, Degrande, Duhr, BenjF, Herquet, Maltoni, Schumann (EPJC '11)]

  * **Develop new models**.
  * **Implement (and validate)** new models in Monte Carlo tools.
  * Facilitate **phenomenological** investigations of the models.
  * **Test** the models against data.

- **Main features**

  * FeynRules is a Mathematica package.
  * FeynRules derives **Feynman rules from a Lagrangian**.
  * **Requirements**: locality, Lorentz and gauge invariance.
  * **Supported fields**: scalar, fermion, vector, tensor, ghost, superfield.
  * **Interfaces**: export the Feynman rules to Monte Carlo generators.

# The FeynRules scheme.



# Example: QCD - Parameters

**Parameters of the model**

```
aS == {
    Description       -> "Strong coupling constant at MZ"
    TeX               -> Subscript[\[Alpha],s],
    ParameterType     -> External,
    BlockName         -> SMINPUTS,
    OrderBlock        -> 3,
    InteractionOrder -> {QCD, 2}},
gs == {
    Description       -> "Strong coupling constant",
    TeX               -> Subscript[g, s],
    ComplexParameter -> False,
    ParameterType     -> Internal,
    Value             -> Sqrt[4 Pi aS],
    InteractionOrder -> {QCD, 1},
    ParameterName     -> "G"}
```

- * **All the information** needed by the MC codes.
- * **TEX-form** (for the TEX-file).
- * **Complex/real** parameters.
- * **External/internal** parameters.

# Example: QCD - Gauge group and gauge boson

**The $SU(3)_C$ gauge group**

```
SU3C == {
  Abelian              -> False,
  GaugeBoson           -> G,
  StructureConstant -> f,
  DTerm                -> dSUN,
  Representations      -> {T, Colour},
  CouplingConstant     -> gs}
```

**Gluon field definition**

```
V[1] == {
  ClassName           -> G,
  SelfConjugate       -> True,
  Indices             -> Index[Gluon],
  Mass                -> 0,
  Width               -> 0,
  ParticleName        -> "g",
  PDG                 -> 21,
  PropagatorLabel     -> "G",
  PropagatorType      -> C,
  PropagatorArrow     -> None}
```

* **Gauge boson** definition.
* **Gauge group** definition.
* Association of a **coupling constant**.
* Definition of the **structure functions**.
* Definition of the **representations**.

# Example: QCD - Quark fields.

### The quark fields

```
F[1] == {
  ClassName        -> q,
  ClassMembers     -> {d, u, s, c, b, t},
  FlavorIndex      -> Flavour,
  SelfConjugate    -> False,
  Indices          -> {Index[Flavour],Index[Colour]},
  Mass             -> {MQ, MD, MU, MS, MC, MB, MT},
  Width            -> {WQ, 0, 0, 0, 0, 0, WT},
  ParticleName     -> {"d", "u", "s", "c", "b", "t"},
  AntiParticleName -> {"d~", "u~", "s~", "c~", "b~", "t~"},
  PDG              -> {1, 2, 3, 4, 5, 6},
  PropagatorLabel  -> {"q", "d", "u", "s", "c", "b", "t"},
  PropagatorType   -> Straight,
  PropagatorArrow  -> Forward}
```

- **Classes**: implicit sums in the Lagrangian.
- **All the information** needed by the MC codes.

# Example: QCD - Lagrangian

**QCD Lagrangian:**

$$\mathcal{L}_{\rm QCD} = -\frac{1}{4} G^a_{\mu\nu} G^{a\mu\nu} + \sum_f \Big[ \bar{q}_f \big( i\partial\!\!\!/ + g_s G\!\!\!/^a T^a - m_f \big) q_f \Big] .$$

**The QCD Lagrangian**

```
LQCD = -1/4 * FS[G, mu, nu, a] * FS[G, mu, nu, a] +
  I*qbar.Ga[mu].DC[q, mu] -
  MQ[f] * qbar[s,f,c].q[s,f,c] ;
```

* **Implicit summations** ⇒ easy debugging.

# Example: QCD - Results

**Results - let us be ready for (some) phenomenology!**

```
FeynmanRules[LQCD, FlavorExpand->False]

    Vertex 1
    Particle 1 : Vector , G
    Particle 2 : Dirac , q†
    Particle 3 : Dirac , q
    Vertex:
```

$$i\, g_s\, \gamma^{\mu_1}_{s_2,s_3}\, \delta_{f_2,f_3}\, T^a_{m_2,m_3}$$

```
    WriteUFO[LQCD]
    WriteFeynArtsOutput[LQCD]
    WriteCHOutput[LQCD]
    WriteSHOutput[LQCD]
    WriieWOOutput[LQCD]
```

# FeynRules-1.6 - status.

- **Current public version: 1.6.0**.
  - * **To be downloaded on http://feynrules.irmp.ucl.ac.be/**.
  - * Contains the **superspace module**.
  - * Contains the **UFO interface** $\Rightarrow$ MadGraph5, GoSam.
  - * Contains the new FeynArts **interface**.
  - * Interfaced to Whizard. [Christensen, Duhr, BenjF, Reuter, Speckner (EPJC '12)]
  - * Supports **color sextets**.
  - * Other interfaces: CalcHep/CompHep, MadGraph4, Sherpa.
  - * **Manual currently being updated**.
    [Christensen, Duhr (CPC '09); Christensen, Degrande, Duhr, BenjF (in prep)].

- **Current online model database**.
  - * **http://feynrules.irmp.ucl.ac.be/wiki/ModelDatabaseMainPage/** .
  - * Standard Model and simple extensions (12).
  - * Supersymmetric models (4).
  - * Extra-dimensional models (4).
  - * Strongly coupled and effective field theories (4).

# Outline

# Latest developments: the Universal FeynRules Output.



- **The UFO** [Degrande, Duhr, BenjF, Grellscheid, Mattelaer, Reiter CPC '12].
  - UFO ≡ Universal FeynRules output (**not tied** to any Monte Carlo tool).
  - FeynRules interface: creates a Python **module** to be linked.
  - The module contains **all** the model information (particles, vertices, ...).
  - Allows for **generic** color and Lorentz structures (no assumptions).
  - Used by MadGraph **5,** MadAnalysis **5,** GoSam **and (soon by)** Herwig**++**.
- **ALOHA** [de Aquino, Link, Maltoni, Mattelaer, Stelzer (2011)].
  - ALOHA ≡ Automatic Libraries Of Helicity Amplitudes.
  - Exports the UFO; **produces the related** Helas **routines** (C++/Python).
    ⇒ to be used for **Feynman diagram computations**.
  - Used by MadGraph**5** / as a standalone package.

# More details about the UFO (1).

- **Particles in UFO**.

Example of the top quark.

```
t = Particle(pdg_code = 6,
             name = 't',
             antiname = 't~',
             spin = 2,
             color = 3,
             mass = Param.MT,
             width = Param.WT,
             texname = 't',
             antitexname = 't',
             charge = 2/3)
t__tilde__ = t.anti()
```

* Similar to FeynRules.
* Slightly **different attribute names**.
* **Masses and widths** are UFO parameters.
* Special function to define **antiparticles**.

# More details about the UFO (2).

- **External parameters in UFO**.

```
aS = Parameter(name = 'aS',
               nature = 'external',
               type = 'real',
               value = 0.1184,
               texname = '\\text{aS}',
               lhablock = 'SMINPUTS',
               lhacode = [ 1 ])
G = Parameter(name = 'G',
              nature = 'internal',
              type = 'real',
              value = '2*cmath.sqrt(aS)*cmath.sqrt(cmath.pi)',
              texname = 'G')
```

- Similar to FeynRules.
- Let us note the **SLHA structure**.
- **External**: value is numeric.
- **Internal**: value is a formula.

# More details about the UFO (3).

- **Vertices in the UFO**.
  - * Must be decomposed in **the spin $\otimes$ color space**.
  - * Concrete example: the **quartic gluon vertex**:

$$\left(f^{a_1a_2b}f^{ba_3a_4},\ f^{a_1a_3b}f^{ba_2a_4}, f^{a_1a_4b}f^{ba_2a_3}\right) \times \begin{pmatrix} ig_s^2 & 0 & 0 \\ 0 & ig_s^2 & 0 \\ 0 & 0 & ig_s^2 \end{pmatrix} \begin{pmatrix} \eta^{\mu_1\mu_4}\eta^{\mu_2\mu_3} - \eta^{\mu_1\mu_3}\eta^{\mu_2\mu_4} \\ \eta^{\mu_1\mu_4}\eta^{\mu_2\mu_3} - \eta^{\mu_1\mu_2}\eta^{\mu_3\mu_4} \\ \eta^{\mu_1\mu_3}\eta^{\mu_2\mu_4} - \eta^{\mu_1\mu_2}\eta^{\mu_3\mu_4} \end{pmatrix}.$$

  - * One line vector in **color space**.
  - * One column vector with the **Lorentz structures**.
  - * One matrix with the **coupling strengths** $\equiv$ the *coordinates*.

# More details about the UFO (4).

- **Vertices in UFO**.

```
V_2 = Vertex(name = 'V_2',
             particles = [ P.G, P.G, P.G, P.G ],
             color = [ 'f(-1,1,2)*f(3,4,-1)',
                       'f(-1,1,3)*f(2,4,-1)',
                       'f(-1,1,4)*f(2,3,-1)' ],
             lorentz = [ L.VVVV1, L.VVVV2, L.VVVV3 ],
             couplings = {(1,1):C.GC_8,
                          (0,0):C.GC_8,
                          (2,2):C.GC_8})
```

  - `color`: the **color basis**.
  - `lorentz`: the **spin basis**.
  - `couplings`: the **non-zero coupling strengths**.

# More details about the UFO (5).

- **Lorentz structures in UFO**.

```
VVVV1 = Lorentz(name = 'VVVV1',
                spins = [ 3, 3, 3, 3 ],
                structure = 'Metric(1,4)*Metric(2,3) -
                             Metric(1,3)*Metric(2,4)')
```

- **Coupling strengths in UFO**.

```
GC_8 = Coupling(name = 'GC_8',
                value = 'complex(0,1)*G**2',
                order = {'QCD':2})
```

- **Coupling orders**.

```
QCD = CouplingOrder(name = 'QCD',
                    expansion_order = 99,
                    hierarchy = 1)
QED = CouplingOrder(name = 'QED',
                    expansion_order = -1,
                    hierarchy = 2)
```

  * Allows to **speed up** event generation, keeping only the dominant diagrams.

# Outline

1. FeynRules in a nutshell.
2. The Universal FeynRules output.
3. Supersymmetry developments.
4. Web validation.
5. Future development plans.
6. Conclusions.

# Latest developments: FeynRules gets supersymmetric.


- **A superspace module for** FeynRules [Duhr, BenjF (CPC '11)].
  - * Full support for **Weyl fermions and superfields**.
  - * Series expansion in terms of **component fields**.
  - * **Automatic derivation** of supersymmetry-conserving Lagrangians.
  - * **Automatic solution** of the equations of motion for the auxiliaries.
  - * Can be used for **many calculations in superspace**.
- **Renormalization group equations** [Alloul, De Causmaecker, BenjF (to appear)].
  - * Automatic extraction of the **two-loop renormalization group equations**.
  - * Own **numerical framework**.
  - * Future interface to **SuSpect 3**.

# Some details on the supersymmetry module (1).

- **Definition**:

$$\Phi(x,\theta,\bar\theta) = \phi(y) + \sqrt{2}\theta\cdot\psi(y) - \theta\cdot\theta F(y) \qquad \text{with} \qquad y^\mu = x^\mu - i\theta\sigma^\mu\bar\theta \ .$$

  * It describes **matter multiplets**.
  * One scalar field $\phi$, one Weyl fermion $\chi$, one auxiliary field $F$.

- **Declaration of the left-handed quark superfield**.

```
CSF[1] == {
  ClassName      -> QL,
  Chirality      -> Left,
  Weyl           -> uqLw,
  Scalar         -> sqL,
  QuantumNumbers -> {Q->2/3},
  Indices        -> {Index[Gen], Index[Colour]}
}
```

  * **Chiral superfield** $\Rightarrow$ the label is `CSF[1]`.
  * The `Scalar` and `Weyl` components must be declared properly.
  * **The auxiliary field are automatically generated (not explicitly present).**

# Some details on the supersymmetry module (2).

- **In the Wess-Zumino gauge, it is defined as**:

$$\Phi_{W.Z.}(x,\theta,\bar\theta) = \theta\sigma^\mu\bar\theta v_\mu + i\theta\cdot\theta\bar\theta\cdot\bar\lambda - i\bar\theta\cdot\bar\theta\theta\cdot\lambda + \frac{1}{2}\theta\cdot\theta\ \bar\theta\cdot\bar\theta D\ .$$

  * This describes **gauge supermultiplets**: one Majorana fermion $(\lambda, \bar\lambda)$, one (massless) gauge boson $v$, one auxiliary field $D$.

- **Declaration of the $SU(3)_c$ vector superfield**.

```
VSF[1] == {
  ClassName        -> GSF,
  GaugeBoson       -> G,
  Gaugino          -> gow,
  Indices          -> {Index[Gluon]}
}
```

  * **Vector superfield** $\Rightarrow$ the label is `VSF[1]`.
  * The `Gaugino` and `GaugeBoson` components must be declared properly.
  * **The auxiliary field are automatically generated (not explicitly present)**.

- **Vector superfields can be associated to a gauge group**.

# Some details on the supersymmetry module (3).

- **Complete Lagrangian for a model.**

$$\mathcal{L} = \Phi^\dagger e^{-2gV}\Phi\big|_{\theta^2\bar\theta^2} + \frac{1}{16g^2\tau_{\mathcal{R}}}\mathrm{Tr}(W^\alpha W_\alpha)\big|_{\theta^2} + \frac{1}{16g^2\tau_{\mathcal{R}}}\mathrm{Tr}(\bar W_{\dot\alpha}\bar W^{\dot\alpha})\big|_{\bar\theta^2}$$
$$+W(\Phi)\big|_{\theta^2} + W^\star(\Phi^\dagger)\big|_{\bar\theta^2} + \mathcal{L}_{\rm soft}$$

  * **Chiral superfield** kinetic terms: automatic.
  * **Vector superfield** kinetic terms: automatic.
  * **Superpotential**: model dependent.
  * **Soft SUSY-breaking Lagrangian**: model dependent (and often not related to the superspace).

```
Theta2Thetabar2Component[ CSFKineticTerms[] ] +
Theta2Component[ VSFKineticTerms[] + SuperPot ] +
Thetabar2Component[ VSFKineticTerms[] + HC[SuperPot] ] +
LSoft
```

  * `LSoft` and `SuperPot` are the **only** pieces provided by the user.

# Some details on the supersymmetry module (4).

- **Solution of the equation of motions**.
  - * Get rid of the auxiliary $D$-fields and $F$-fields through their **eqs. of motion**.

```
lagr = SolveEqMotionD[ lagr ] ;
lagr = SolveEqMotionF[ lagr ] ;
```

- **Back to four-component fermions**.
  - * Usual FeynRules routine (cf. MC code requirements).

```
lagr = WeylToDirac[ lagr ] ;
```

# Some details on the supersymmetry module (5).

- **Renormalization group equations for generic supersymmetric models**.
  * One or two loops included.

```
RGE[ LSoft, SuperW, NLoops->1 ] ;
```

$$\frac{\mathrm{d}\mu}{\mathrm{d}t} = \mu\Big[ - \frac{3g'^2}{80\pi^2} - \frac{3g_w^2}{16\pi^2} + \frac{3}{16\pi^2}\mathrm{Tr}\big[\mathbf{y^d}^\dagger \mathbf{y^d}\big] + \frac{3}{16\pi^2}\mathrm{Tr}\big[\mathbf{y^u}^\dagger \mathbf{y^u}\big] + \frac{1}{16\pi^2}\mathrm{Tr}\big[\mathbf{y^e}^\dagger \mathbf{y^e}\big]\Big] ,$$

$$\frac{\mathrm{d}b}{\mathrm{d}t} = b\Big[ - \frac{3g'^2}{80\pi^2} - \frac{3g_w^2}{16\pi^2} + \frac{3}{16\pi^2}\mathrm{Tr}\big[\mathbf{y^d}^\dagger \mathbf{y^d}\big] + \frac{3}{16\pi^2}\mathrm{Tr}\big[\mathbf{y^u}^\dagger \mathbf{y^u}\big] + \frac{1}{16\pi^2}\mathrm{Tr}\big[\mathbf{y^e}^\dagger \mathbf{y^e}\big]\Big]$$
$$+ \mu\Big[\frac{3g'^2 M_1}{40\pi^2} + \frac{3g_w^2 M_2}{8\pi^2} + \frac{3}{8\pi^2}\mathrm{Tr}\big[\mathbf{y^d}^\dagger \mathbf{T^d}\big] + \frac{3}{8\pi^2}\mathrm{Tr}\big[\mathbf{y^u}^\dagger \mathbf{T^u}\big] + \frac{1}{8\pi^2}\mathrm{Tr}\big[\mathbf{y^e}^\dagger \mathbf{T^e}\big]\Big] .$$

- **Development plans:**
  * Interface to a own numerical code ⇒ **spectrum generator generator**.
  * Interface to SuSpect **3**.

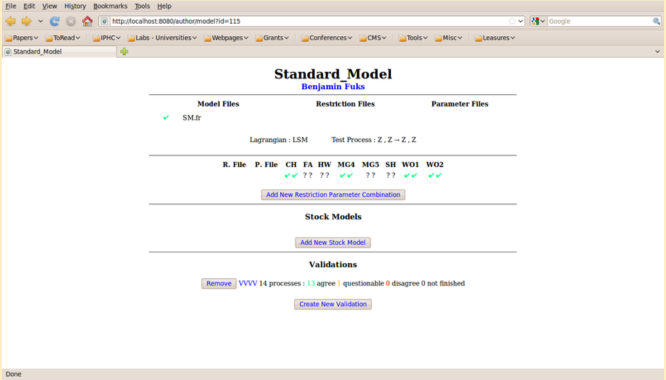
# Outline

# Validation of new models.

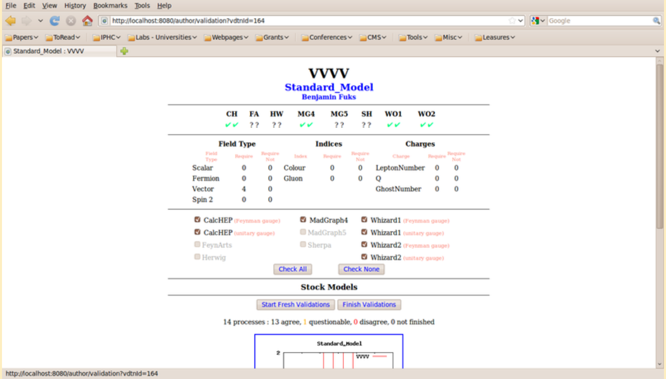
[Christensen, Salmon, Setzer, Speckner, Stefanus (to appear)].

- FeynRules **provides a platform to**:
  - * **Develop** Beyond the Standard Model theories.
  - * **Validate** them to an unprecedented level.
- **Using the different interfaces, we can compare different Monte Carlos**:
  - * Using **different conventions**.
  - * Using **different gauges**.
  - * Using **different way of handling large cancellations**.
- **This can be fully automated**.
- **This can be stored on the Internet**.
- **A full FeynRules web interface is also under development**.

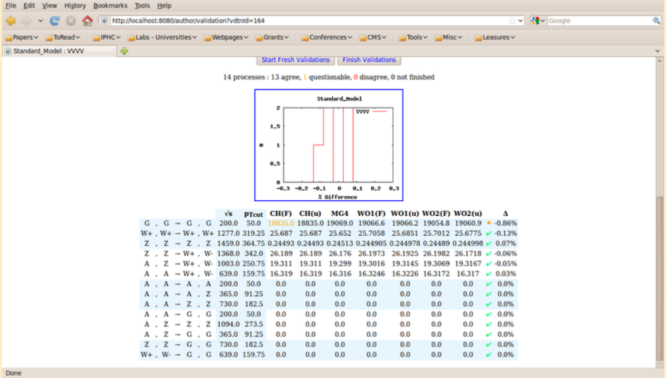
# Web validation.



# Web validation.



# Web validation.




| | √s | PTcut | CH(F) | CH(u) | MG4 | WO1(F) | WO1(u) | WO2(F) | WO2(u) | Δ |
|---|---|---|---|---|---|---|---|---|---|---|
| G , G → G , G | 200.0 | 50.0 | 18835.0 | 18835.0 | 19069.0 | 19066.6 | 19066.2 | 19054.8 | 19060.9 | • -0.86% |
| W+ , W+ → W+ , W+ | 1277.0 | 319.25 | 25.687 | 25.687 | 25.652 | 25.7058 | 25.6851 | 25.7012 | 25.6775 | ✓ -0.13% |
| Z , Z → Z , Z | 1459.0 | 364.75 | 0.24493 | 0.24493 | 0.24513 | 0.244905 | 0.244978 | 0.24489 | 0.244998 | ✓ 0.07% |
| Z , Z → W+ , W- | 1368.0 | 342.0 | 26.189 | 26.189 | 26.176 | 26.1973 | 26.1925 | 26.1982 | 26.1718 | ✓ -0.06% |
| A , Z → W+ , W- | 1003.0 | 250.75 | 19.311 | 19.311 | 19.299 | 19.3016 | 19.3145 | 19.3069 | 19.3167 | ✓ -0.05% |
| A , A → W+ , W- | 639.0 | 159.75 | 16.319 | 16.319 | 16.316 | 16.3246 | 16.3226 | 16.3172 | 16.317 | ✓ 0.03% |
| A , A → A , A | 200.0 | 50.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ✓ 0.0% |
| A , A → A , Z | 365.0 | 91.25 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ✓ 0.0% |
| A , A → Z , Z | 730.0 | 182.5 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ✓ 0.0% |
| A , A → G , G | 200.0 | 50.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ✓ 0.0% |
| A , Z → Z , Z | 1094.0 | 273.5 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ✓ 0.0% |
| A , Z → G , G | 365.0 | 91.25 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ✓ 0.0% |
| Z , Z → G , G | 730.0 | 182.5 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ✓ 0.0% |
| W+ , W- → G , G | 639.0 | 159.75 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | ✓ 0.0% |

# Outline

# FeynRules 2012 @Sainte Odile - the 'official' picture

# NLO with FeynRules.

[Degrande, Duhr, Frederix, BenjF, Hahn, Hirschi, Mattelaer]

- **Real emission**.
  - Must include the appropriate **subtraction terms**.
    $\Rightarrow$ Automated.
  - The tree-level Feynman rules are the **only required components**.

☺ **No particular problem for BSM $\Rightarrow$ problem solved.** ☺
(Use of FeynRules, its interfaces to MC tools)

- **One-loop virtual amplitudes**.
  - Several algorithms have been proposed in the last few years. $\Rightarrow$ some are based on OPP reduction [Ossola, Papadopolous, Pittau (NPB '07)].
  - Requirements:
    - **Tree-level Feynman rules**.
    - **UV renormalization counterterms**.
    - **Rational $R_2$ terms**.

☹ **The two latter must be included by hand.** ☹

# NLO calculations in the context of FeynRules.

- **Counterterms and $R_2$ terms**.
  - * **Non-automatic steps**.
  - * ☼ **Can be derived from the tree-level Lagrangian**.

☺ **All the information is already there at the FeynRules-level.** ☺

- **Automatic renormalization in the $\overline{\text{MS}}$-scheme with FeynRules**
  1. **Automated extraction of the renormalized Lagrangian** ✓.
  2. **Modification of the FeynArts interface to include counterterms** ✓.
  3. **Calculation of the renormalization constants with FormCalc**.
  4. **Re-injection in FeynRules** ✗.

- **Automatic $R_2$ terms** ✗.
  ►**SM**: done.
  ►**MSSM**: on-going

- **The UFO at NLO**: basically there.

# Automatic renormalization with FeynRules.

- **Expansion of the renormalization constants (works with full Lagrangians)**.
  - The **type of the interactions** in the loops can be specified.
  - The **loop-level** can be specified.

```
ExtractCounterterms[l[s,f],{aS,1}]
```

$$\blacktriangleright\, l_{sf} \to l_{sf} + \frac{\alpha_s}{4\pi}\Big[(\delta Z_{ll}^{L(1)})_{ff'}\ (P_L)_{ss'} + (\delta Z_{ll}^{R(1)})_{ff'}\ (P_R)_{ss'}\Big]\ l_{s'f'}$$

```
ExtractCounterterms[ydo,{{aS,2},{aEW,1}}]
```

$$\blacktriangleright\, y_d \to y_d + \frac{\alpha_s}{2\pi}\delta y_d^{(1,0)} + \frac{\alpha}{2\pi}\delta y_d^{(0,1)} + \frac{\alpha_s^2}{4\pi^2}\delta y_d^{(2,0)} + \frac{\alpha_s\alpha}{4\pi^2}\delta y_d^{(1,1)} + \frac{\alpha_s^2\alpha}{8\pi^3}\delta y_d^{(2,1)}$$

- **Treatment of the internal parameters**.
  - Automatic computation of the relations among renormalization constants.
  - **Only** the ren. cnsts of the **external parameters** will have to be computed.

**$g_s$ and $\alpha_s$ at first order in QCD.**

$$g_s = 2\sqrt{\pi\alpha_s} \qquad \Rightarrow \qquad \delta g_s^{(1)} = \frac{\sqrt{\alpha_s}}{2\sqrt{\pi}}\delta\alpha_s^{(1)}$$

# Importing the counterterms to FeynArts.

- **New FeynArts interface** [Degrande, Duhr, BenjF].
  * Allows for **generic** Lorentz structures.
  * Creates both the **model dependent and independent** FeynArts files.
  * Update of FormCalc $\Rightarrow$ **multifermion interactions**.
  * For the counterterms: **the loop level must be specified**.
  * Automated introduction of the **renormalization constants**.

**The FeynRules command**

```
WriteFeynArtsOutput[LSM, LoopOrder -> {aS, 1}];
```

```
C[ S[1] , S[1] , S[1] , S[1] ] == {{(-6*I)*lam,
  ((-3*I)*aS*(deltalam1 + 2*deltaZHH1*lam))/Pi}}

C[ V[3] , -V[3] ] == {{0, ((-I/2)*aS*deltaZWW1)/Pi},
  {0, ((I/2)*aS*deltaZWW1)/Pi}, {0, ((I/8)*aS*EL*vev*
  (2*deltavev1*EL*sw - 2*deltasw1*EL*vev + 2*deltaee1*
  sw*vev + deltaZWW1*EL*sw*vev))/(Pi*sw^3)}}
```

# Spin-3/2 Rarita-Schwinger fields.

[Christensen, de Aquino, Duhr, BenjF, Mattelaer, Mawatari, Oexl]

- **Spin 3/2 fields.**
  - * **Four-component fields** are now fully supported ✔.

**Gravitino implementation.**

```
R[1] == {
  ClassName        -> Gvno,
  SelfConjugate    -> True,
  Mass             -> {MGvno,500},
  Width            -> {WGvno, 10},
  ParticleName     -> "Gvno",
  PDG              -> 1000039}
```

  - * Analytical **validation** with the literature ✔.
  - * Extension to **two-component fermions** ✔.
  - * Automatic extraction of the **gravitino Lagrangian in SUSY** ✔.
  - * Inclusion in the **UFO** ✔.
  - * **Being validated**.

# Automated decay package (1).

[Duhr, BenjF, Mattelaer]

- **Tree-level $1 \to 2$ decay widths.**
  - * Are now **automatically computed** for any model.

**Computation of decay widths.**

```
verts = FeynmanRules[Lag];
vertsexp = FlavorExpansion[verts];
results = ComputeWidths[vertsexp];
```

- **Everything is analytical.**
- **Phase-space closed channels included**.
  - No information on the (numerical values of the) spectrum at this level.
  - **Benchmark scenario independent**.

# Automated decay package (2).

- **Interfaced to the UFO**.

**UFO interface.**

```
WriteUFO[Lag, AddDecays->True];
```

```
Decay_t = Decay(name = 'Decay_t', particle = P.t,
 partial_widths = {
   (P.W__plus__,P.d):'(CKM3x1*ee**2*(MT - MW)**2*(MT + MW)**2*
     (MT**2 + 2*MW**2)*complexconjugate(CKM3x1))/
     (64.*cmath.pi*MT**3*MW**2*sw**2)',
   (P.W__plus__,P.s):'(CKM3x2*ee**2*(MT - MW)**2*(MT + MW)**2*
     (MT**2 + 2*MW**2)*complexconjugate(CKM3x2))/
     (64.*cmath.pi*MT**3*MW**2*sw**2)',
   (P.W__plus__,P.b):'(CKM3x3*ee**2*(MB**4 - 2*MB**2*MT**2 +
     MT**4 + MB**2*MW**2 + MT**2*MW**2 - 2*MW**4)*
     complexconjugate(CKM3x3)*cmath.sqrt(MB**4 - 2*MB**2*MT**2 +
     MT**4 - 2*MB**2*MW**2 - 2*MT**2*MW**2 + MW**4))/
     (64.*cmath.pi*MT**3*MW**2*sw**2)'})
```

  - **Numerically**: checks open channels.
  - **Flexible**: closed formulas (NLO, *n*-body, BSM) can be included.
- **Validation on-going** (SM done, MSSM on its way).

# Outline

# Conclusions.

- FeynRules **provides a platform to**:
  * **Develop** new models.
  * Investigate their **phenomenology**.
- **The current version now includes:**
  * A **superspace module**.
  * A **UFO interface** $\Rightarrow$ MadGraph5, GoSam.
  * Supports **color sextets**.
  * The **web-validation** platform is online.
  * An automated **decay package** (soon).
  * **Spin 3/2** fields (soon).
- **New developments:**
  * Automated **supersymmetric spectrum generator**.
  * An **interface to** SuSpect 3.
  * A **NLO module**.
  * A full **web-interface**.
- **http://feynrules.irmp.ucl.ac.be**