Beyond the Standard Model Physics with FeynRules and Monte Carlo tools.

Benjamin Fuks.

IPHC Strasbourg / Université de Strasbourg.

KIAS School on MadGraph @ KIAS (Seoul, Korea).
October 24-30, 2011.

Outline.

1. FeynRules in a nutshell.
2. A (maybe not so) simple example: implementation of supersymmetric QCD.
3. Using FeynRules with the supersymmetric QCD model.
4. Advanced model implementation techniques.
5. The superspace module.
6. Summary.

Monte Carlo tools and discoveries at the LHC (1).

- **One of the goals of the LHC: which New Physics theory is the correct one?**
  [if any, the LHC might be one ring to rule them all out!]
  - We need **data** [which are finally there].
  - We need **theoretical predictions for any model** [which is the aim of this talk].
    - For Standard Model (SM) backgrounds.
    - For Beyond the Standard Model (BSM) signals.

Confront data and theory.

- **Theoretical predictions:**
  - **Handmade calculations** ☹.
    - Not practical: factorial growth of the number of diagrams.
    - Tedious and error prone.
  - **Automated Monte Carlo tools** ☺.
    - Easy to use!
    - Can be used to simulate the full collision environment.

Monte Carlo tools and discoveries at the LHC (2).

- **Establishing of an excess over the SM backgrounds**.
  - * **Difficult task**.
  - * Use of **Monte Carlo generators** (backgrounds, signals).
- **Confirmation of the excess**.
  - * **Model building activities**.
    - ◇ Bottom-up approach.
    - ◇ Top-down approach.
  - * **Implementation** of the new models in the Monte Carlo tools.
- **Clarification of the new physics**.
  - * **Measurement of the parameters**.
  - * Use of **precision predictions**.
  - * **Sophistication of the analyses ⇔ new physics and detector knowledge**.

Monte Carlo tools play a key role!
But how is new physics presently investigated in particle physics?

A framework for BSM analyzes at the LHC (1).

Idea ⇔

Model building

* **Pen&pencil stage.**
* Leading order, loop calculations, ...
* Electroweak, low energy constraints,...

⇒ **Publications**

⇓

Phenomenology

* **Monte Carlo event generator.**
* Matrix element calculation.
* Parton showering / hadronization.
* Generic detector simulation.

⇒ **Publications**

⇓

Experiment

* **Experimental framework.**
* Matrix elements, parton showering, hadronization.
* Realistic detector simulation.
* **Comparison with data**.

⇒ **Publications**

A framework for LHC analyzes (2).

- **New physics theories**.
  - There are a **lot of different** theories.
  - Based on very **different ideas**.
  - **In evolution** (especially regarding the discoveries).

Implementation in Monte Carlo tools.

- A model consists in:
  - **particles**,
  - **parameters**,
  - **interactions** ($\equiv$ Feynman rules).
- The Feynman rules **have to be derived (from a Lagrangian)**.
  - Must be **translated in a programming language**.
  - **Tedious, time-consuming, error prone**.
  - We need to iterate for each considered model.
  - We need to iterate for each considered MC tool.
  - Beware: **allowed Lorentz and color structures**.

Redundancies of the work.

A framework for LHC analyzes (3).

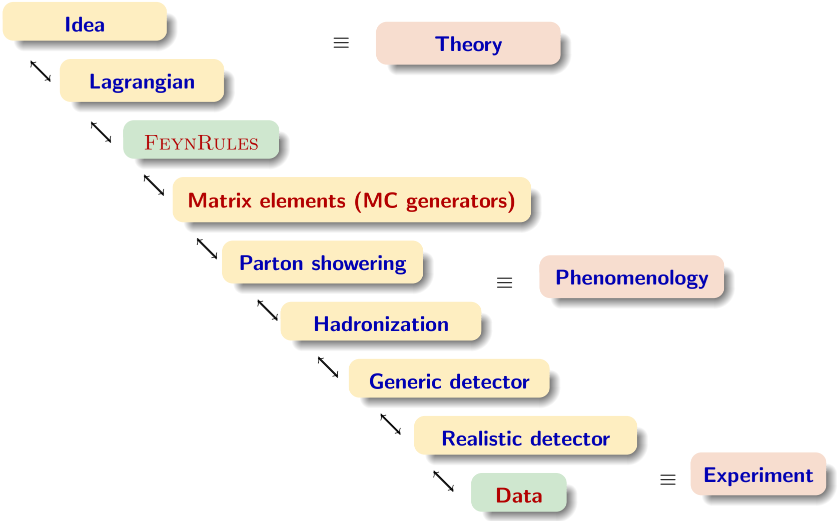
- **Validation**.
  - * **Necessary at each step**.
    - * **Error-prone**.
    - * **Time-consuming**.
  - * **Comparison** with existing analytical and numerical results.
  - * **Non systematic and partial**.
    - ◇ **Restricted set** of available results.
    - ◇ **No dedicated framework**.
    - ◇ **Warning: conventions**.
- **Distribution**.
  - * Many models remain **private**.
  - * **Exception:** popular models, *e.g.*, the MSSM.
  - * Use of **many home-made and hacked versions** of existing models.
    $\Rightarrow$ Issues about documentation, traceability, maintenance, ...

A framework for LHC analyzes (4).

We need an efficient framework:

- To **develop** new models.
- To **implement (and validate)** new models in MC tools.
- To **test** the models against data.
- **Enhancing communication** between theory and experiment.

A FEYNRULES-based framework for LHC analyzes.



The FeynRules approach (1).

- **Starting from physical quantities**.
  - * All the physics is included in the model **Lagrangian**.
    - ◇ Remark: the Lagrangian is **absent in the MC implementation**.
  - * **Traceability.**
    - ◇ **Univocal definition of a model**.
    - ◇ **No dependance on the conventions used** by the MC tools.
  - * **Flexibility.**
    - ◇ A modification of a model $\equiv$ change in the Lagrangian.

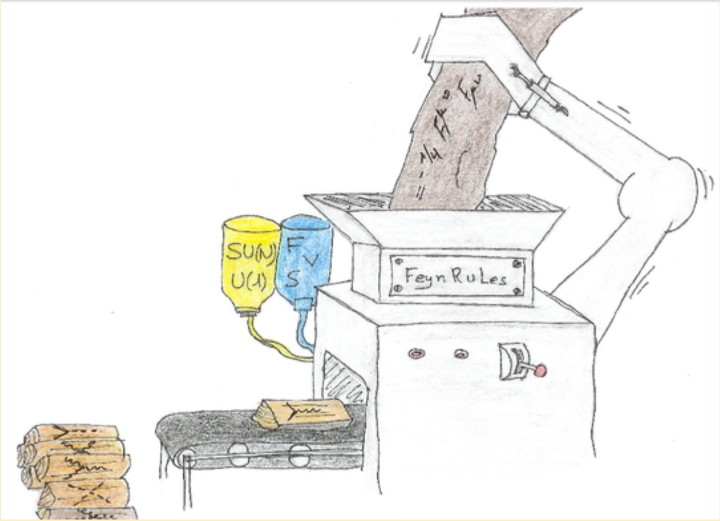
Aims.

- * A **general environment** to implement any Lagrangian-based model.
- * To interface **several Monte Carlo generators**.
- * **Robustness, easy validation and maintenance**.
- * Easy integration in **experimental software frameworks**.
- * Allowing for both **top-down and bottom-up approaches**.

The FEYNRULES approach (2).

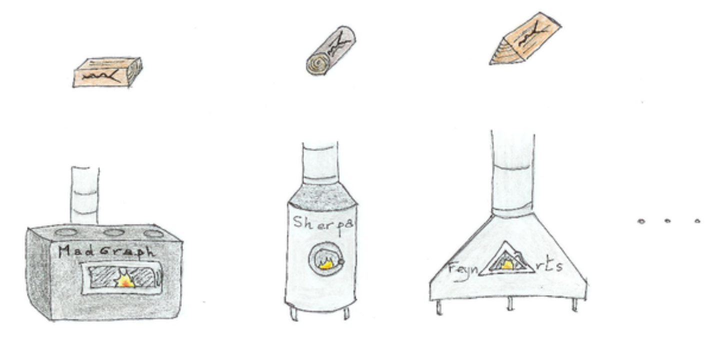
© C. Degrande

The FeynRules approach (3).



© C. Degrande

The FeynRules approach (4).



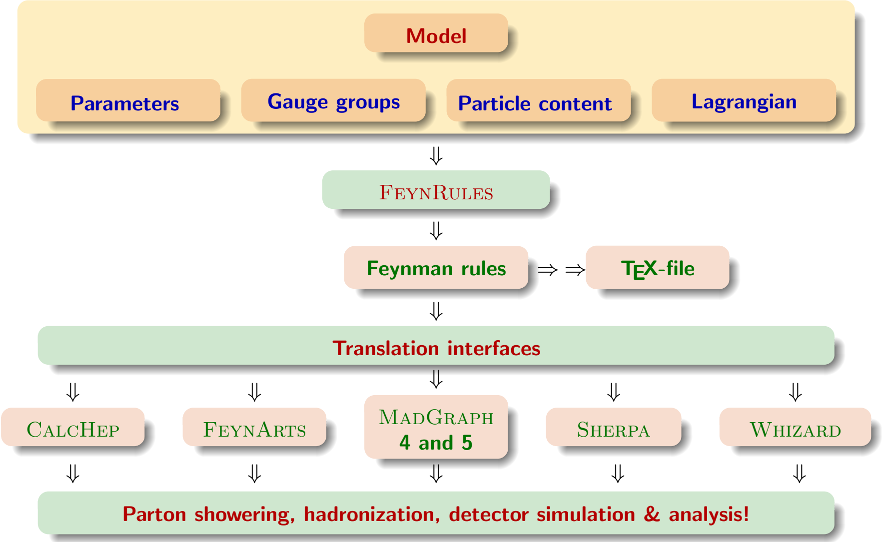
© C. Degrande

FeynRules in one slide.

- **A framework for LHC analyzes based on** FeynRules **to**:
  - * **Develop new models**.
  - * **Implement (and validate)** new models in Monte Carlo tools.
  - * Facilitate **phenomenological** investigations of the models.
  - * **Test** the models against data.

- FeynRules **in a nutshell**
  - * 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, superfields.
  - * **Interfaces**: export the Feynman rules to Monte Carlo generators.

Main features of FeynRules (1).



Main features of FeynRules (2).

- **The working environment is MATHEMATICA.**
  - * **Flexibility** for symbolic manipulations.
    - ◇ **Routines** to check a Lagrangian.
    - ◇ ...
  - * Various **built-in features.**
    - ◇ **Matrix diagonalization**.
    - ◇ **Pattern recognition functions**.
    - ◇ ...
  - * **New additional functions** can easily be added by users.
    - ◇ **Model spectrum calculator**.
    - ◇ ...
- **Interfaces to Monte Carlo codes**.
  - * The philosophy, architecture and aim of the codes can be **different**.
  - * **Maximization** of probability to have (at least) one (working) MC per model.
  - * FeynRules **translates** models in terms of files readable by the MC tools.

Outline.

Supersymmetric QCD - general features.

- **Field content.**
  - * **Matter multiplets.**
    - ◇ Three generations of up-type left-handed quarks and squarks.
    - ◇ Three generations of up-type right-handed quarks and squarks.
  - * **The $SU(3)_c$ vector multiplet.**
    - ◇ Gluino and gluon fields.
- **Symmetries of the theory.**
  - * $SU(3)_c$ **gauge invariance.**
  - * **Supersymmetry.**
- **The dynamics of the system is given by the Lagrangian**

$$\begin{aligned}\mathcal{L} \quad = \quad & -\frac{1}{4} g^a_{\mu\nu} g_a^{\mu\nu} + \frac{i}{2} \bar{\tilde{g}}^a \not{D} \tilde{g}^a + D_\mu \tilde{q}^\dagger_{Li} D^\mu \tilde{q}_{Li} + D_\mu \tilde{q}^\dagger_{Ri} D^\mu \tilde{q}_{Ri} + i \bar{q} \not{D} q \\ & - m^2_{\tilde{q}_i} \tilde{q}^\dagger_i \tilde{q}_i - m_q \bar{q} q - \frac{1}{2} m_{\tilde{g}} \bar{\tilde{g}}^a \tilde{g}^a \\ & - \frac{g_s^2}{2} \Big[- \tilde{q}^\dagger_{Li} T^a \tilde{q}_{Li} + \tilde{q}^\dagger_{Ri} T^a \tilde{q}_{Ri} \Big] \Big[- \tilde{q}^\dagger_{Lj} T^a \tilde{q}_{Lj} + \tilde{q}^\dagger_{Rj} T^a \tilde{q}_{Rj} \Big] \\ & + \sqrt{2} g_s \Big[- \tilde{q}^\dagger_{Li} T^a \big(\tilde{g}^a P_L q \big) + \big(\bar{q} P_L \tilde{g}^a \big) T^a \tilde{q}_{Ri} \Big] + \text{h.c.} \ ,\end{aligned}$$

with $i, j = 1, 2, 3$.

How to write FeynRules model files.

- **A FeynRules model file follows the Mathematica syntax.**
- **It is a .fr text file containing:**
  - * A **preamble**.
    - ◇ Author information.
    - ◇ Model information.
    - ◇ Definitions of the indices.
  - * The declaration of the model **gauge group**.
    - ◇ Abelian or not.
    - ◇ Representation matrices, structure constants.
    - ◇ Associated coupling constant.
    - ◇ Associated gauge boson or vector superfield.
  - * The declaration of the **particle** content.
    - ◇ Names, spins, PDG-ids, carried indices.
    - ◇ Self-conjugate or not, quantum numbers.
    - ◇ Masses, widths.
    - ◇ Particles of the same type can be grouped in **classes**.
  - * The declaration of the model **parameters**.
  - * The **Lagrangian** itself.

Preamble of the model file (1).

- **The preamble of the model file contains**:
  - * **Author and model information**.

```
M$ModelName = "SUSYQCD";

M$Information = {
  Authors      -> {"Benjamin Fuks"},
  Date         ->  "24.10.11",
  Version      ->  "1.0.0",
  Institutions -> {"IPHC Strasbourg / U. of Strasbourg"},
  Emails       -> {"benjamin.fuks@iphc.cnrs.fr"}
};
```

Other possible options: `References`, `URLs`.

Preamble of the model file (2).

- **The preamble of the model file contains**:
  * The definitions of the **dimension of the indices**.

```
IndexRange[Index[Gluon ]] = NoUnfold[Range[8]];
IndexRange[Index[Colour]] = NoUnfold[Range[3]];
IndexRange[Index[Gen   ]] = Range[3];
```

  - ◇ `Gluon` ⇔ $SU(3)_c$ adjoint index, **reserved keyword**
  - ◇ `Colour` ⇔ $SU(3)_c$ fundamental index, **reserved keyword**.
  - ◇ `Gen` ⇔ Generation index.

  * The definitions of the **style to be used for the indices**.

```
IndexStyle[Colour, m];
IndexStyle[Gluon,  a];
IndexStyle[Gen,    f];
```

- `Color` **and** `Gluon` **are special names**.
  * Strong interactions have special significance in MC tools.
  * Same for the gluon field name (`G`), the strong coupling constant (`gs`, `aS`), the fundamental color matrices (`T`), the structure constants (`f`).

Declaration of the gauge group.

- **Declaration of the $SU(3)_c$ gauge group (in `M$GaugeGroups`)** .

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

  * The group is **non-Abelian**.
  * The associated **gauge boson** is the **gluon field** `G` (▶ see later).
  * The associated coupling constant is the **parameter** `gs` (▶ see later).
  * The **structure constants** `f` are associated to the adjoint representation.
  * **Representation matrices** `T` are associated to the index type `Colour`.

- **Consequences: easier Lagrangian building**.
  * Automated definition of the **field strength tensor** for the gluon `FS[G,mu,nu,a]`.
  * Automated definition of a **covariant derivative** for all fields `DC[field[...],mu]`.

Field declaration - the gluon field.

- **Declaration of the gauge boson** `G` (in `M$ClassesDescription`).

```
V[1] == {
  ClassName        -> G,
  SelfConjugate    -> True,
  Indices          -> {Index[Gluon]},
  Mass             -> 0,
  Width            -> 0,
  PDG              -> 21
}
```

- **Vector field** ⇒ the label is `V[1]`.
- **Symbol** to be used in the Lagrangian: `G`.
- Its **own antiparticle** ⇒ `SelfConjugate -> True`.
- **Adjoint representation** of $SU(3)_c$ ⇒ `Indices -> {Index[Gluon]}`.
  - This relates (internally) the index `Gluon` to the adjoint representation.
- **Vanishing mass and widths**.
- PDG-id $\equiv 21$ ⇒ `PDG -> 21`.
- **Other possible options for vector fields**: `Unphysical`, `Definitions`, `PropagatorLabel`, `PropagatorType`, `PropagatorArrow`, `ParticleName`, `AntiParticleName`, `QuantumNumbers`.

Field declaration - the gluino field (1).

- **Declaration of the gluino field $\tilde{g}$ (in `M$ClassesDescription`)**.

```
F[1] == {
  ClassName        -> go,
  SelfConjugate    -> True,
  Indices          -> {Index[Gluon]},
  PDG              -> 1000021,
  Mass             -> {Mgo,500},
  Width            -> {Wgo,10}
}
```

* **Four-component fermion** ⇒ the label is `F[1]`.
* **Symbol** to be used in the Lagrangian: `go`, `gobar`.
* Its **own antiparticle** ⇒ `SelfConjugate -> True`.
* **Adjoint representation** of $SU(3)_c$ ⇒ `Indices -> {Index[Gluon]}`.
* PDG-id ≡ 1000021 ⇒ `PDG -> 1000021`.

Field declaration - the gluino field (2).

- **Declaration of the gluino field $\tilde{g}$ (in `M$ClassesDescription`)**.

```
F[1] == {
  ClassName         -> go,
  SelfConjugate     -> True,
  Indices           -> {Index[Gluon]},
  PDG               -> 1000021,
  Mass              -> {Mgo,500},
  Width             -> {Wgo,10}
}
```

* **The gluino mass**.
  - Symbol to be used in the Lagrangian: `Mgo`.
  - Chosen numerical value: **500 GeV**.
  - Can be set to `Internal` $\Leftrightarrow$ link to an internal parameter.
* **The gluino width**.
  - Symbol to be used: `Wgo`.
  - Chosen numerical value: **10 GeV**.

Parenthesis: representations of the Lorentz algebra (1).

- **The left-handed Weyl spinor representation** $(1/2, 0)$.
  - Action on **complex left-handed spinors** ψ_α $(\alpha = 1, 2)$.
  - **Generators**: a set of 6 2×2 matrices based on the Pauli matrices.

$$(\sigma^{\mu\nu})_\alpha{}^\beta = -\frac{i}{4}\Big(\sigma^\mu\bar\sigma^\nu - \sigma^\nu\bar\sigma^\mu\Big)_\alpha{}^\beta\ .$$

  - A **finite Lorentz transformation** is given by

$$\Lambda_{(\frac{1}{2},0)} = \exp\left[\frac{i}{2}\omega_{\mu\nu}\sigma^{\mu\nu}\right]\ .$$

- **The right-handed Weyl spinor representation** $(0, 1/2)$.
  - Action on **complex right-handed spinors** $\bar\chi^{\dot\alpha}$ $(\dot\alpha = \dot1, \dot2)$.
  - **Generators**: a set of 6 2×2 matrices based on the Pauli matrices.

$$(\bar\sigma^{\mu\nu})^{\dot\alpha}{}_{\dot\beta} = -\frac{i}{4}\Big(\bar\sigma^\mu\sigma^\nu - \bar\sigma^\nu\sigma^\mu\Big)^{\dot\alpha}{}_{\dot\beta}\ .$$

  - A **finite Lorentz transformation** is given by

$$\Lambda_{(0,\frac{1}{2})} = \exp\left[\frac{i}{2}\omega_{\mu\nu}\bar\sigma^{\mu\nu}\right]\ .$$

- **Complex conjugation maps left-handed and right-handed spinors**.

Parenthesis: representations of the Lorentz algebra (2).

- **A Dirac spinor is defined as**

$$\psi_D = \begin{pmatrix} \psi_\alpha \\ \bar{\chi}^{\dot{\alpha}} \end{pmatrix} ,$$

which is a **reducible** representation of the Lorentz algebra.

  - **Generators of the Lorentz algebra**: a set of 6 4×4 matrices

$$\gamma^{\mu\nu} = -\frac{i}{4}\left[\gamma^\mu, \gamma^\nu\right] = \begin{pmatrix} \sigma^{\mu\nu} & 0 \\ 0 & \bar{\sigma}^{\mu\nu} \end{pmatrix}$$

  - A **finite Lorentz transformation** is given by

$$\Lambda_{(\frac{1}{2},0)\oplus(0,\frac{1}{2})} = \exp\left[\frac{i}{2}\omega_{\mu\nu}\gamma^{\mu\nu}\right] = \begin{pmatrix} \Lambda_{(\frac{1}{2},0)} & 0 \\ 0 & \Lambda_{(0,\frac{1}{2})} \end{pmatrix} .$$

- **A Majorana spinor is defined as**

$$\psi_M = \begin{pmatrix} \psi_\alpha \\ \bar{\psi}^{\dot{\alpha}} \end{pmatrix} ,$$

$\Leftrightarrow$ a Dirac spinor with **conjugate left- and right-handed components**.

$$\bar{\psi}^{\dot{\alpha}} = \varepsilon^{\dot{\alpha}\dot{\beta}}\bar{\psi}_{\dot{\beta}} \qquad \text{with} \qquad \bar{\psi}_{\dot{\beta}} = \left(\psi_\beta\right)^\dagger .$$

Field declaration - the gluino field (3).

- **Declaration of the gluino field $\tilde{g}$ (in `M$ClassesDescription`)**.

```
F[1] == {
  ClassName         -> go,
  SelfConjugate     -> True,
  Indices           -> {Index[Gluon]},
  PDG               -> 1000021,
  Mass              -> {Mgo,500},
  Width             -> {Wgo,10}
}
```

  - * The `WeylComponents` option for fermionic fields (example later).
    - ◇ Definition of a two-component fermion: `W[1]` instead of `F[1]`.
    - ◇ **Chirality**: `Chirality -> Left` or `Chirality -> Right`.
    - ◇ Linking **Dirac** and **two-component** fermions: `WeylComponents->{psi, chibar}`.
    - ◇ Linking **Majorana** and **two-component** fermions: `WeylComponents->gow`.
  - * **Other possible options for fermionic fields**: `Unphysical`, `Definitions`, `PropagatorLabel`, `PropagatorType`, `PropagatorArrow`, `AntiParticleName`, `QuantumNumbers`.

Field declaration - the quark fields.

- **Declaration of the up-type quark fields u_q (in `M$ClassesDescription`).**

```
F[2] == {
  ClassName        -> uq,
  SelfConjugate    -> False,
  Indices          -> {Index[Gen], Index[Colour]},
  FlavorIndex      -> Gen,
  QuantumNumbers   -> {Q -> 2/3},
  ClassMembers     -> {u, c, t},
  Mass             -> {Mu, {MU,2.55*^-3}, {MC,1.42}, {MT,172}},
  Width            -> {0, 0, {WT,1.50833649}},
  PDG              -> {2, 4, 6}
}
```

- Similar to the gluino declaration.
- Introduction of **particle classes**.
  - `uq` **and** `uqbar`: generic up-type quark.
  - `Gen` is the **flavor index** ⇒ defines class members.
  - Particle attributes consist now in **lists**.
- Remark: we assign an **electric charge quantum number**.

Field declaration - the squark fields (1).

- **Declaration of the left up-type squarks $\tilde{q}_{Li}$ (in `M$ClassesDescription`).**

```
S[1] == {
  ClassName         -> sqL,
  SelfConjugate     -> False,
  Indices           -> {Index[Gen],Index[Colour]},
  FlavorIndex       -> Gen,
  QuantumNumbers    -> {Q -> 2/3},
  ClassMembers      -> {suL, scL, stL},
  Mass              -> {MsqL, {MsuL,300}, {MscL,300}, {MstL,300}},
  Width             -> {{WsuL,5}, {WscL,5}, {WstL,5}},
  PDG               -> {1000002, 1000004, 1000006}
}
```

  * **Similar as for the other particles**.
  * **Scalar field** ⇒ the label is `S[1]`.
  * **Symbol** to be used in the Lagrangian: `sqL` and `sqLbar`.

Field declaration - the squark fields (2).

- **Declaration of the right up-type squarks $\tilde{q}_{Li}$ (in `M$ClassesDescription`)**.

```
S[2] == {
  ClassName        -> sqR,
  SelfConjugate    -> False,
  Indices          -> {Index[Gen],Index[Colour]},
  FlavorIndex      -> Gen,
  QuantumNumbers   -> {Q -> 2/3},
  ClassMembers     -> {suR, scR, stR},
  Mass             -> {MsqR, {MsuR,300}, {MscR,300}, {MstR,300}},
  Width            -> {{WsuR,5}, {WscR,5}, {WstR,5}},
  PDG              -> {2000002, 2000004, 2000006}
}
```

  - **Similar as for the other particles**.
  - **Scalar field** $\Rightarrow$ the label is `S[2]`.
  - **Symbol** to be used in the Lagrangian: `sqR` and `sqRbar`.

Declaration of the model parameters (1).

- **Masses and widths**.
  - * **Already taken into account** at the particle declaration time.
  - * **No need to declare them a second time**
- **The Lagrangian is**:

$$
\begin{aligned}
\mathcal{L} \quad = \quad & -\frac{1}{4} g^a_{\mu\nu} g_a^{\mu\nu} + \frac{i}{2} \bar{\tilde{g}}^a \not{D} \tilde{g}^a + D_\mu \tilde{q}^\dagger_{Li} D^\mu \tilde{q}_{Li} + D_\mu \tilde{q}^\dagger_{Ri} D^\mu \tilde{q}_{Ri} + i \bar{q} \not{D} q \\
& - m^2_{\tilde{q}_i} \tilde{q}^\dagger_i \tilde{q}_i - m_q \bar{q} q - \frac{1}{2} m_{\tilde{g}} \bar{\tilde{g}}^a \tilde{g}^a \\
& - \frac{g_s^2}{2} \Big[- \tilde{q}^\dagger_{Li} T^a \tilde{q}_{Li} + \tilde{q}^\dagger_{Ri} T^a \tilde{q}_{Ri} \Big] \Big[- \tilde{q}^\dagger_{Lj} T^a \tilde{q}_{Lj} + \tilde{q}^\dagger_{Rj} T^a \tilde{q}_{Rj} \Big] \\
& + \sqrt{2} g_s \Big[- \tilde{q}^\dagger_{Li} T^a \big(\bar{\tilde{g}}^a P_L q \big) + \big(\bar{q} P_L \tilde{g}^a \big) T^a \tilde{q}_{Ri} \Big] + \mathrm{h.c.} \ ,
\end{aligned}
$$

with $i, j = 1, 2, 3$.

- * We only need to declare the **strong coupling constant**.
- * Requirement from the MC tools: declaration of both `gs` and `aS`.

Declaration of the model parameters (2).

- **Declaration of the parameters (in `M$Parameters`).**

```
aS == {
  ParameterType     -> External,
  Value             -> 0.1184,
  InteractionOrder -> {QCD, 2}
},
gs == {
  ParameterType     -> Internal,
  Value             -> Sqrt[4 Pi aS],
  InteractionOrder -> {QCD, 1},
  ParameterName     -> G
}
```

* We have `Internal` and `External` parameters.
  - `External`: free parameter of the theory ⇒ a numerical value must be provided (`Value`).
  - `Internal`: dependent parameter of the theory ⇒ a formula must be provided (`Value`)
* `InteractionOrder`: specific to MadGraph.
* `ParameterName`: specific to MC tools.

Declaration of the model parameters (3).

- **Declaration of the parameters (in `M$Parameters`).**

```
aS == {
  ParameterType    -> External,
  Value            -> 0.1184,
  InteractionOrder -> {QCD, 2}
},
gs == {
  ParameterType    -> Internal,
  Value            -> Sqrt[4 Pi aS],
  InteractionOrder -> {QCD, 1},
  ParameterName    -> G
}
```

  * **Other possible options for parameters**: `TeX`, `Definitions`, `ComplexParameter`, `Description`, `BlockName`, `OrderBlock`.
  * **Other possible options for matrices**: `Indices`, `Unitary`.

Outline.

Implementing the vector Lagrangian.

- **The vector multiplet (gluino and gluon) Lagrangian reads**:

$$\mathcal{L}_{\rm vector} \quad = \quad -\frac{1}{4} g^a_{\mu\nu} g_a^{\mu\nu} + \frac{i}{2} \bar{\tilde{g}}^a \not{D} \tilde{g}^a - \frac{1}{2} m_{\tilde{g}} \bar{\tilde{g}}^a \tilde{g}^a$$

  - * **Kinetic and mass terms for the gluon and the gluino fields**.
  - * **Gauge interaction terms for the gluon and the gluino fields**.

- **Use of predefined functions**.

```
LVector = -1/4 FS[G,mu,nu,a] FS[G,mu,nu,a] +
           I/2 Ga[mu,s1,s2] gobar[s1,a].DC[go[s2,a],mu] -
           1/2 Mgo gobar[s1,a].go[s1,a]
```

Loading the model in MATHEMATICA (1).

- **Testing the (partial) model in** MATHEMATICA.
- **Step 1: loading** FEYNRULES.
  - * Setting up **the** FEYNRULES **path**.
  - * **Loading the program** itself.

```
$OldDir = Directory[];
$FeynRulesPath =
  SetDirectory["~/FeynRules/trunk/feynrules-development"];
<< FeynRules`
```

- MATHEMATICA **ouput messages**.

```
In[1]:= $OldDir = Directory[];
        $FeynRulesPath = SetDirectory["~/FeynRules/trunk/feynrules-development"];
        << FeynRules`

         - FeynRules -

        Authors: C. Duhr, N. Christensen, B. Fuks


        Please cite: Comput.Phys.Commun.180:1614-1641,2009 (arXiv:0806.4194).
        http://feynrules.phys.ucl.ac.be


        The FeynRules palette can be opened using the command FRPalette[].
```

Loading the model in MATHEMATICA (2).

- **Step 2: loading the model file**.
  - * It contains all the **information above**.

```
SetDirectory[$OldDir];
LoadModel["susyqcd.fr"];
```

- MATHEMATICA **ouput messages**.

```
In[4]:= SetDirectory[$OldDir];
        LoadModel["susyqcd.fr"];

       This model implementation was created by
       Benjamin Fuks
       Model Version: 1.0.0
       For more information, type ModelInformation[].

          - Loading particle classes.
          - Loading gauge group classes.
          - Loading parameter classes.

       Model SUSYQCD loaded.
```

  - * Printing the information included in the **preamble** of the model file.

Checking the implementation in MATHEMATICA (1).

- **Step 3: Printing the Lagrangian.**

```
LVector
```

- MATHEMATICA **ouput messages**.

In[6]:= **LVector**

Out[6]= $-\frac{1}{2}\,\mathrm{Mgo}\,\mathrm{go}_{s1,a}\cdot\mathrm{go}_{s1,a}-\frac{1}{4}\left(-\partial_{nu}[G_{mu,a}]+\partial_{mu}[G_{nu,a}]+\mathrm{gs}\,f_{a,bb\$658,cc\$658}\,G_{mu,bb\$658}\,G_{nu,cc\$658}\right)^2+$
$\frac{1}{2}\,i\,\mathrm{go}_{s1,a}\cdot\left(\partial_{mu}[\mathrm{go}_{s2,a}]-i\,\mathrm{gs}\,\mathrm{FSU3C}^{a\$659}_{a,i\$659}\,G_{mu,a\$659}\,\mathrm{go}_{s2,i\$659}\right)\gamma^{mu}_{s1,s2}$

  - * Automated generation of the **field strength tensor**.
  - * Automated generation of the **adjoint representation matrices**.
    ►Included in the gluino covariant derivative terms.

- **Reminder:**

$$\mathcal{L}_{\text{vector}} = -\frac{1}{4}g^a_{\mu\nu}g_a^{\mu\nu} + \frac{i}{2}\bar{\tilde{g}}^a \not{D} g^a - \frac{1}{2}m_{\tilde{g}}\bar{\tilde{g}}^a\tilde{g}^a$$

Checking the implementation in MATHEMATICA (2).

- **Step 4: Checking the Lagrangian**.
  - * The Lagrangian must be **hermitian**.

```
CheckHermiticity[LVector];
```

```
In[7]:= CheckHermiticity[LVector];
   Checking for hermiticity by calculating the Feynman rules contained in L-HC[L].
   If the lagrangian is hermitian, then the number of vertices should be zero.
   Starting Feynman rule calculation.
   Collecting the different structures that enter the vertex...
   Found 1 possible non zero vertices.
   Start calculating vertices...

   0 vertices obtained.
   The lagrangian is hermitian.
```

Checking the implementation in MATHEMATICA (3).

- **Step 4: Checking the Lagrangian** (cntn'd).
  - * The kinetic terms must be **correctly normalized**.
  - * The kinetic terms must be **diagonal**.

```
CheckKineticTermNormalisation[LVector];
```

```
In[8]:= CheckKineticTermNormalisation[LVector];
    Neglecting all terms with more than 2 particles.
    All kinetic terms are diagonal.
    Neglecting all terms with more than 2 particles.
    All kinetic terms are correctly normalized.
```

- **Other similar checks**: `CheckDiagonalQuadraticTerms`, `CheckDiagonalKineticTerms`, `CheckDiagonalMassTerms`.

Checking the implementation in MATHEMATICA (4).

- **Step 4: Checking the Lagrangian** (cntn'd).
  * Investigation of the **mass spectrum**.
  * Extracting the masses **from the Lagrangian**.
  * Comparing with the values provided in **the declaration of particles**.

```
CheckMassSpectrum[LVector];
```

```
In[9]:= CheckMassSpectrum[LVector]
    Neglecting all terms with more than 2 particles.
    All mass terms are diagonal.
    Neglecting all terms with more than 2 particles.
    Getting mass spectrum.
    Checking for less then 0.1% agreement with model file values.
Out[9]//TableForm=
     Particle Analytic value Numerical value Model-file value
     go       Mgo            500.            500.
```

Feynman rules (1).

- **Printing the Feynman rules**.

```
FeynmanRules[LVector];
```

- MATHEMATICA **ouput messages**.

```
In[10]:= FeynmanRules[LVector];
    Starting Feynman rule calculation.
    Collecting the different structures that enter the vertex...
    Found 3 possible non zero vertices.
    Start calculating vertices...
    3 vertices obtained.
    (* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *)
    Vertex 1
    Particle 1 : Vector , G
    Particle 2 : Vector , G
    Particle 3 : Vector , G
    Vertex:
```

$$gs\, f_{a_1,a_2,a_3}\, p_1^{\mu_3}\, \eta_{\mu_1,\mu_2} - gs\, f_{a_1,a_2,a_3}\, p_2^{\mu_3}\, \eta_{\mu_1,\mu_2} - gs\, f_{a_1,a_2,a_3}\, p_1^{\mu_2}\, \eta_{\mu_1,\mu_3} + gs\, f_{a_1,a_2,a_3}\, p_3^{\mu_2}\, \eta_{\mu_1,\mu_3} + gs\, f_{a_1,a_2,a_3}\, p_2^{\mu_1}\, \eta_{\mu_2,\mu_3} - gs\, f_{a_1,a_2,a_3}\, p_3^{\mu_1}\, \eta_{\mu_2,\mu_3}$$

* **Adjoint color indices** a_i are related to the $i^{\rm th}$ particle.
* **Lorentz indices** μ_i are related to the $i^{\rm th}$ particle.

Feynman rules (2).

- **Printing the Feynman rules**.

```
FeynmanRules[LVector];
```

- MATHEMATICA **ouput messages** (cntn'd).

```
(* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *)
Vertex 2
Particle 1 : Vector , G
Particle 2 : Vector , G
Particle 3 : Vector , G
Particle 4 : Vector , G
Vertex:
```

$$i\,gs^2 f_{a_1,a_3,c1} f_{a_2,a_4,c1} \eta_{\mu_1,\mu_4} \eta_{\mu_2,\mu_3} + i\,gs^2 f_{a_1,a_2,c1} f_{a_3,a_4,c1} \eta_{\mu_1,\mu_4} \eta_{\mu_2,\mu_3} + i\,gs^2 f_{a_1,a_4,c1} f_{a_2,a_3,c1} \eta_{\mu_1,\mu_3} \eta_{\mu_2,\mu_4} - i\,gs^2 f_{a_1,a_2,c1} f_{a_3,a_4,c1} \eta_{\mu_1,\mu_3} \eta_{\mu_2,\mu_4} - i\,gs^2 f_{a_1,a_4,c1} f_{a_2,a_3,c1} \eta_{\mu_1,\mu_2} \eta_{\mu_3,\mu_4} - i\,gs^2 f_{a_1,a_3,c1} f_{a_2,a_4,c1} \eta_{\mu_1,\mu_2} \eta_{\mu_3,\mu_4}$$

* **Adjoint color indices** a_i are related to the $i^{\rm th}$ particle.
* **Lorentz indices** μ_i are related to the $i^{\rm th}$ particle.
* The **adjoint color index** `c1` is a summed (repeated) index.

Feynman rules (3).

- **Printing the Feynman rules**.

```
FeynmanRules[LVector];
```

- MATHEMATICA **ouput messages** (cntn'd).

```
(* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *)
Vertex 3
Particle 1 : Vector , G
Particle 2 : Majorana , go
Particle 3 : Majorana , go
Vertex:
```

$gs\, f_{a_1,a_2,a_3}\, \gamma^{\mu_1}_{s_2,s_3}$

```
(* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *)
```

* The **adjoint color index** a_1 is related to the 1^{st} particle, the gluon.
* The **Lorentz index** μ_1 ies related to the 1^{st} particle, the gluon.
* The **spin indices** c_2, c_3 are related to the 2^{nd} and 3^{rd} particles (gluinos).

Implementing the matter Lagrangian (1).

- **The matter multiplet (quark and squarks) Lagrangian reads**:

$$\begin{aligned}\mathcal{L}_{\rm matter} \quad = \quad & D_\mu \tilde{q}^\dagger_{Li} D^\mu \tilde{q}_{Li} + D_\mu \tilde{q}^\dagger_{Ri} D^\mu \tilde{q}_{Ri} + i\bar{q}\not{D}q - m^2_{\tilde{q}_i} \tilde{q}^\dagger_i \tilde{q}_i - m_q \bar{q} q \\ & - \frac{g_s^2}{2}\Big[- \tilde{q}^\dagger_{Li} T^a \tilde{q}_{Li} + \tilde{q}^\dagger_{Ri} T^a \tilde{q}_{Ri}\Big]\Big[- \tilde{q}^\dagger_{Lj} T^a \tilde{q}_{Lj} + \tilde{q}^\dagger_{Rj} T^a \tilde{q}_{Rj}\Big] \\ & + \sqrt{2} g_s \Big[- \tilde{q}^\dagger_{Li} T^a \left(\tilde{g}^a P_L q\right) + \left(\bar{q} P_L \tilde{g}^a\right) T^a \tilde{q}_{Ri}\Big] + \text{h.c.}\end{aligned}$$

  - * **Kinetic and mass terms for the (s)quark fields**.
  - * **Gauge interaction terms for (s)quark fields**.
  - * The so-called ***D*-terms**.
  - * Supersymmetric **quark-squark-gluino interactions**.

Implementing the matter Lagrangian (2).

- **Kinetic, mass and gauge interaction terms**:

$$\mathcal{L}_{\rm matter,kin} = D_\mu \tilde{q}_{Li}^\dagger D^\mu \tilde{q}_{Li} + D_\mu \tilde{q}_{Ri}^\dagger D^\mu \tilde{q}_{Ri} + i\bar{q}\not{D} q - m_{\tilde{q}_i}^2 \tilde{q}_i^\dagger \tilde{q}_i - m_q \bar{q} q$$

- **Use of predefined functions**.

```
Lkin = DC[sqLbar[cc,ff],mu] DC[sqL[cc,ff],mu] +
       DC[sqRbar[cc,ff],mu] DC[sqR[cc,ff],mu] +
       I Ga[mu,s1,s2] uqbar[s1,ff,cc].DC[uq[s2,ff,cc],mu] -
       MsqL[ff]^2 sqLbar[ff,cc] sqL[ff,cc] -
       MsqR[ff]^2 sqRbar[ff,cc] sqR[ff,cc] -
       Mu[ff] uqbar[s1,ff,cc].uq[s1,ff,cc]
```

* `sqLbar` and `sqRbar` denote the hermitian-conjugate fields.
* **Implicit summation over flavor indices** (`ff`).
* **Covariant derivatives** for both squarks and quarks (`DC`).

Implementing the matter Lagrangian (3).

- *D*-**terms**:

$$\mathcal{L}_{\rm matter,D} = -\frac{g_s^2}{2}\Big[-\tilde{q}^\dagger_{Li}T^a\tilde{q}_{Li} + \tilde{q}^\dagger_{Ri}T^a\tilde{q}_{Ri}\Big]\Big[-\tilde{q}^\dagger_{Lj}T^a\tilde{q}_{Lj} + \tilde{q}^\dagger_{Rj}T^a\tilde{q}_{Rj}\Big]$$

- **Straightforward implementation**.

```
LD = -1/2 gs^2 *
     (sqRbar[ff1,cc1] T[a,cc1,cc2] sqR[ff1,cc2] -
      sqLbar[ff1,cc1] T[a,cc1,cc2] sqL[ff1,cc2]) *
     (sqRbar[ff2,cc3] T[a,cc3,cc4] sqR[ff2,cc4] -
      sqLbar[ff2,cc3] T[a,cc3,cc4] sqL[ff2,cc4])
```

* **Implicit summation over repeated indices**.
  ► Compact form for the Lagrangian.

* **BEWARE: do not use a specific index more than twice** (here).

Implementing the matter Lagrangian (4).

- **The gluino-quark-squark interaction terms**:

$$\mathcal{L}_{\rm matter,gosqq} = \sqrt{2} g_s \Big[- \tilde{q}^\dagger_{Li} T^a \big(\tilde{g}^a P_L q \big) + \big(\bar{q} P_L \tilde{g}^a \big) T^a \tilde{q}_{Ri} \Big] + {\rm h.c.}$$

- **Straightforward implementation**.

```
Lgosqq =  Sqrt[2] gs ProjM[s1,s2] *(
  - sqLbar[ff, cc1] T[a,cc1,cc2] gobar[s1,a].uq[s2,ff,cc2] +
    uqbar[s1,ff,cc1].go[s2,a] T[a,cc1,cc2] sqR[ff,cc2]);
```

  * introduction of the **chirality projectors** (ProjM, ProjP).

- The complete **matter Lagrangian reads**:

```
LMatter = Lkin + LD + Lgosqq + HC[Lgosqq];
```

  * The HC **function**: automatic derivation of the hermitian-conjugate pieces.

Check of the matter Lagrangian.

- **The new pieces of the Lagrangian can be tested as** `LVector`.
- **Example: the mass spectrum**.

```
In[6]:= CheckMassSpectrum[LMatter]

    Neglecting all terms with more than 2 particles.

    All mass terms are diagonal.

    Neglecting all terms with more than 2 particles.

    Getting mass spectrum.

    Checking for less then 0.1% agreement with model file values.

Out[6]//TableForm=
```

Particle	Analytic value	Numerical value	Model-file value
c	MC	1.42	1.42
t	MT	172.	172.
u	MU	0.00255	0.00255
scL	$\sqrt{\text{MscL}^2}$	300.	300.
scR	$\sqrt{\text{MscR}^2}$	300.	300.
stL	$\sqrt{\text{MstL}^2}$	300.	300.
stR	$\sqrt{\text{MstR}^2}$	300.	300.
suL	$\sqrt{\text{MsuL}^2}$	300.	300.
suR	$\sqrt{\text{MsuR}^2}$	300.	300.

Manipulating Feynman rules (1).

- **Calculating all Feynman rules**.

```
In[7]:= FR = FeynmanRules[LMatter, ScreenOutput → False];

Starting Feynman rule calculation.

Collecting the different structures that enter the vertex...

Found 12 possible non zero vertices.

Start calculating vertices...

12 vertices obtained.
```

* The option `ScreenOutput` renders FeynRules **silent**.
* **Flavor indices are kept understood**.
  ▶ *e.g.*, we will get one single quark-antiquark-gluon vertex, and not three.

```
In[8]:= FR2 = FeynmanRules[LMatter, ScreenOutput → False, FlavorExpand → True];

Starting Feynman rule calculation.

Collecting the different structures that enter the vertex...

Found 48 possible non zero vertices.

Start calculating vertices...

48 vertices obtained.
```

* All vertices have now been **expanded in flavor space**.
  ▶ *e.g.*, we have here three quark-antiquark-gluon vertices.

Manipulating Feynman rules (2).

- **Selecting given Feynman rules**.

```
SelectVertices[FR, Contains -> {uq}, Free -> {go}]
```

* We select the Feynman rules containing **quarks** (`Contains`).
* We select the Feynman rules not containing any **gluino** (`Free`).

In[9]:= `SelectVertices[FR, Contains → {uq}, Free → {go}]`

`Applying seclection rules...`

Out[9]= $\{\{\{\{G, 1\}, \{uq, 2\}, \{\bar{uq}, 3\}\}, i\, gs\, \gamma^{\mu_1}_{s_3,s_2}\, \delta_{f_2,f_3}\, T^{a_1}_{m_3,m_2}\}\}$

* The list of particles contain the particle **names** and **numbers**.
► relating indices to particles.
* The **color index** a_1 is related to the 1st particle (gluon).
* The **Lorentz index** μ_1 is related to the 1st particle (gluon).
* The **color indices** m_2, m_3 are related to the 2nd and 3rd particles (quarks).
* The **spin indices** s_2, s_3 are related to the 2nd and 3rd particles (quarks).
* The **flavor indices** f_2, f_3 are related to the 2nd and 3rd particles (quarks).
* QCD interactions are **diagonal in flavor space** (δ_{f_2,f_3}).

Manipulating Feynman rules (3).

- **Selecting given Feynman rules (cntn'd)**.

```
SelectVertices[FR2, Contains -> {G},
  Free -> {suR, scR, stR, suL, stL, scL}]
```

* We select the Feynman rules containing a **gluon** (`Contains`).
* We select the Feynman rules not containing any **squark** (`Free`).

In[10]:= `SelectVertices[FR2, Contains → {G}, Free → {suR, scR, stR, suL, stL, scL}]`

`Applying seclection rules...`

Out[10]=

$$\left\{\left\{\left\{\{c,1\},\{\bar{c},2\},\{G,3\}\right\}, i\, gs\, \gamma^{\mu_3}_{s_2,s_1}\, T^{a_3}_{m_2,m_1}\right\}, \left\{\left\{\{G,1\},\{t,2\},\{\bar{t},3\}\right\}, i\, gs\, \gamma^{\mu_1}_{s_3,s_2}\, T^{a_1}_{m_3,m_2}\right\}, \left\{\left\{\{G,1\},\{u,2\},\{\bar{u},3\}\right\}, i\, gs\, \gamma^{\mu_1}_{s_3,s_2}\, T^{a_1}_{m_3,m_2}\right\}\right\}$$

* The list of particles contain the particle **names** and **numbers**.
* The **color index** a_i is related to the i^{th} particle (gluon).
* The **Lorentz index** μ_i is related to the i^{th} particle (gluon).
* The **color index** m_i is related to the i^{th} particles (quark).
* The **spin index** s_i is related to the i^{th} particles (quark).
* **No more explicit flavor indices**.

From FeynRules to phenomenology (1).

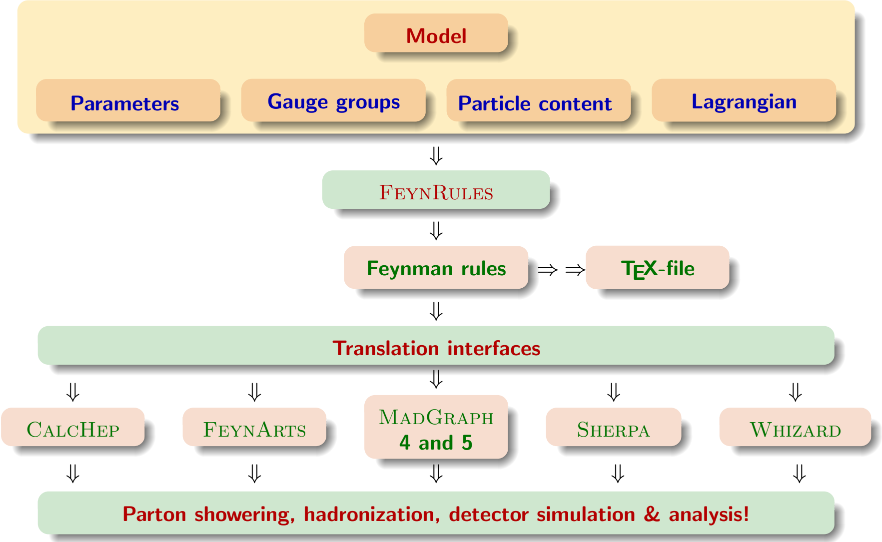
We are now ready to do phenomenology.

* The model is (correctly) **implemented in** FeynRules.
  - The particle content.
  - The parameters.
  - The Feynman rules.
* The Feynman rules can be **automatically** derived.
* Model information can be **automatically** exported to MC's.
  - CalcHep/CompHep.
  - FeynArts/FormCalc.
  - MadGraph version 4.
  - Sherpa.
  - **The UFO format** $\Rightarrow$ MadGraph **version 5**.
  - Whizard/Omega.

From FeynRules to phenomenology (2).

- **The UFO** [arXiv:1108.2040].
  - UFO ≡ Universal FeynRules output (**not tied** to any Monte Carlo tool).
  - Allows for **generic** color and Lorentz structures.
  - Used by MadGraph**5,** Golem **and** Herwig++.
  - FeynRules interface: creates a Python **module** to be linked.
  - The module contains **all** the model information.
- **ALOHA** [arXiv:1108.2041].
  - 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**.

From FeynRules to phenomenology (3).



Limitations of the Monte Carlo generators (1).

- **Some names are hard-coded at the MC level.**
  - * Issues related to the **strong interactions**.
    - ◇ The names of the **color indices**: `Colour` **and** `Gluon`.
    - ◇ The names of the **strong coupling constants**: `aS` **and** `gs`.
    - ◇ The numerical value of `aS` is given at the *Z*-pole (*cf.* **running**).
    - ◇ The **gluon field** name is `G`.
    - ◇ The **structure constants** are denoted by `f`.
    - ◇ The **fundamental representation** is given by `T`.
  - * **Weak interactions**: Fermi coupling and the *Z* mass.
  - * **Hypercharge and the weak coupling constant**.
  - * More: see the manual...

- **Some generators have hard-coded color structures.**
  - * The interfaces reject the **unsupported structures**.
    - ◇ CalcHep: $\underset{\sim}{1}$, $\underset{\sim}{3}$, $\underset{\sim}{8}$ (limited).
    - ◇ FeynArts: all.
    - ◇ MadGraph 4: $\underset{\sim}{1}$, $\underset{\sim}{3}$, $\underset{\sim}{8}$ (limited).
    - ◇ MadGraph **5: $\underset{\sim}{1}$, $\underset{\sim}{3}$, $\underset{\sim}{6}$, $\underset{\sim}{8}$**.
    - ◇ Sherpa: $\underset{\sim}{1}$, $\underset{\sim}{3}$, $\underset{\sim}{8}$.
    - ◇ Whizard: $\underset{\sim}{1}$, $\underset{\sim}{3}$, $\underset{\sim}{8}$.

Limitations of the Monte Carlo generators (2).

- **Some generators have hard-coded Lorentz structures.**
  - * The interfaces reject the **unsupported structures**.
    - ◇ CalcHep: all (theoretically).
    - ◇ FeynArts: all.
    - ◇ MadGraph 4: MSSM-like.
    - ◇ MadGraph **5: all**.
    - ◇ Sherpa: SM-like.
    - ◇ Whizard: MSSM-like.
- **Not all spin states are allowed.**
  - * The interfaces reject the **unsupported structures**.
    - ◇ CalcHep: scalar, spinor, vector, tensor.
    - ◇ FeynArts: scalar, spinor, vector.
    - ◇ MadGraph 4: scalar, spinor, vector (+ Rarita-Schwinger, tensor).
    - ◇ MadGraph **5: scalar, spinor, vector, tensor**.
    - ◇ Sherpa: scalar, spinor, vector.
    - ◇ Whizard: scalar, spinor, vector, tensor.

Running the interfaces (1).

- **Using the** CalcHep **interface**.

```
WriteCHOutput[{LVector,LMatter}];
```

  * Arguments: **a list of Lagrangians**.
  * Main options: `Exclude4Scalars`, `CHSimplify`, `ModelNumber`, `Output`.
  * Complete list of options: see the manual.

- Mathematica **output messages**:

```
In[11]:= WriteCHOutput[{LVector, LMatter}];

    - - - FeynRules interface to CalcHep/CompHEP
    - - - Authors: N. Christensen, C. Duhr
    - - - Please cite: arXiv:0906.2474

   Writing files to /home/bfuks/FeynRules/trunk/models/SUSYQCD/SUSYQCD-CH.
...
    Warning! The following vertex is not implemented in FeynRules->CH yet.
        {{suL, 1}, {suL, 2}, {suL~, 3}, {suL~, 4}}
        You can add this vertex by hand after importing into CalcHEP.
...
 Done in 0.07min!
```

  * All the generated files are stored in **a single directory**.
    $\Rightarrow$ **to be copy-pasted in** CalcHep.
  * Non-supported vertices have been **automatically rejected**.

Running the interfaces (2).

- **Using the FeynArts interface**.

```
WriteFeynArtsOutput[{LVector,LMatter}];
```

  * Arguments: **a list of Lagrangians**.
  * Main options: `FlavorExpand`, `Output`, `CouplingRename`, `GenericFile`.
  * Complete list of options: see the manual.

- Mathematica **output messages**:

```
In[12]:= WriteFeynArtsOutput[{LVector, LMatter}];
    - - - FeynRules interface to FeynArts - - -
          C. Degrande C. Duhr, 2010
...
   Writing FeynArts model file into directory SUSYQCD_FA
   Writing FeynArts generic file on SUSYQCD_FA.gen.
```

  * All the generated files are stored in **a single directory**.
    ⇒ **to be copy-pasted in** FeynArts.
  * A **generic** model file (.gen) and a **model-dependent** file (.mod) are created.

Running the interfaces (3).

- **Using the other interfaces works in the same fashion**.

```
WriteSHOutput[{LVector,LMatter}];
WriteMGOutput[{LVector,LMatter}];
WriteWOOutput[{LVector,LMatter}];
```

* Arguments: **a list of Lagrangians**.
* Complete list of options: see the manual.
* All the generated files are stored in **a single directory**.
  ⇒ **to be copy-pasted in the corresponding Monte Carlo tool**.
* **All generated models by** FeynRules **are plug 'n' play**.

From FeynRules to MadGraph 5 (1).

- **Extracting UFO files (works as for the other interfaces)**.

```
WriteUFO[{LVector,LMatter}];
```

  - Arguments: **a list of Lagrangians**.
  - Main options: `Exclude4Scalars`, `RemoveGhosts`, `Input`, `Output`.
  - Complete list of options: see the manual.

- Mathematica **output messages**:

```
In[13]:= WriteUFO[{LVector, LMatter}];

    --- Universal FeynRules Output (UFO) v 0.1 ---
...
      - Saved vertices in InterfaceRun[ 1 ].
   Preparing Python output.
      - Splitting vertices into building blocks.
      - Optimizing: 51/51 .
      - Writing files.
   Done!
```

  - All the generated files are stored in **a single directory** (SUSYQCD_UFO).

From FEYNRULES to MADGRAPH 5 (2).

- **The UFO format is a PYTHON translation of the FEYNRULES format.**
  - * Generic, **model-independent** files.
    - ◇ `__init__.py`: initialization of the lists of particles, vertices, ...
    - ◇ `object_library.py`: definition of all classes (`Particle`, ...)
    - ◇ `function_library.py`: definition of user-defined functions.
    - ◇ `write_param_card.py`: exporting the UFO parameters to a standard MG `param_card.dat`.
  - * **Model-independent** files.
    - ◇ `particles.py`: particles of the model.
    - ◇ `parameters.py`: parameters of the model.
    - ◇ `vertices.py`: Feynman rules, with the color structures explicit.
    - ◇ `couplings.py`: the coupling strengths appearing in the vertices.
    - ◇ `lorentz.py`: the Lorentz structures appearing in the vertices.
    - ◇ `coupling_orders.py`: Coupling orders.

DISCLAIMER

In these lectures, only the basic features of the UFO will be covered. For more information: arXiv:1108.2040. Please investigate the UFO files produced during the tutorial sessions.

From FeynRules to MadGraph 5 (3).

- **The particles in UFO**.

```
G = Particle(pdg_code = 21,
             name = 'G',
             antiname = 'G',
             spin = 3,
             color = 8,
             mass = Param.ZERO,
             width = Param.ZERO,
             texname = 'G',
             antitexname = 'G',
             charge = 0)
```

* Similar to FeynRules.
* Slightly **different attribute names**.
* **Spin** is $2s + 1$.
* Special keyword for **zero**.

From FeynRules to MadGraph 5 (4).

- **The particles in UFO (cntn'd)**.

```
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**.
  - **Spin** is $2s + 1$.
  - **Masses and widths** are UFO parameters.
  - Special function to define **antiparticles**.

From FeynRules to MadGraph 5 (5).

- **External parameters in UFO**.

```
aS = Parameter(name = 'aS',
               nature = 'external',
               type = 'real',
               value = 0.1184,
               texname = '\\text{aS}',
               lhablock = 'FRBlock',
               lhacode = [ 1 ])
```

* Similar to FeynRules.
* Let us note the **SLHA structure**.
* `value` is numeric.

From FeynRules to MadGraph 5 (6).

- **Internal parameters in UFO**.

```
G = Parameter(name = 'G',
              nature = 'internal',
              type = 'real',
              value = '2*cmath.sqrt(aS)*cmath.sqrt(cmath.pi)',
              texname = 'G')
```

  * Similar to FeynRules.
  * value is a formula.

From FeynRules to MadGraph 5 (7).

- **Vertices in the UFO**.
  - * Must be decomposed in **the spin $\otimes$ color space**.
  - * Concrete example: the **quartic gluon vertex** (slide 45):

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

becomes:

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

From FeynRules to MadGraph 5 (8).

- **Vertices in the UFO (cntn'd).**

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

* One line vector in **color space**.
* One column vector with the **Lorentz structures**.
  Stored in `lorentz.py`.
* One matrix with the **coupling strengths** $\equiv$ the *coordinates*.
  Stored in `couplings.py`.

From FeynRules to MadGraph 5 (9).

- **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**.

From FeynRules to MadGraph 5 (10).

- **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** MadGraph, keeping only the dominant diagrams.

From FEYNRULES to MADGRAPH 5 (11).

- **Exporting the UFO into MADGRAPH 5.**
  - **All generated models by FEYNRULES are plug 'n' play**.
  - A **single copy-paste** is enough.
  - In a shell:

```
cp -r SUSYQCD_UFO ~/Tools/madgraph5/models/
```

- **Running MADGRAPH 5.**
  - **Disclaimer:** for more advanced MADGRAPH usage, see the MG lectures!
  - The generated UFO model **can be used as any other MADGRAPH model**. (**the UFO is the standard model format for MADGRAPH 5**).
  - In a shell (no PYTHIA here, default cards):

```
cd ~/Tools/madgraph5
./bin/mg5
...
mg5> import model SUSYQCD_UFO -modelname
mg5> generate g g >  go go
mg5> output
mg5> launch -f
```

- **The cross section is 14.8 pb.**

Outline

Other useful tips for implementing models in FeynRules.

- **Gauge-eigenstates and mass-eigenstates.**
  - * **Gauge-eigenstates**: compact Lagrangian, easier to implement.
  - * **Mass-eigenstates**: physical fields, complicated Lagrangian (in general).
  - * Relation through **unitary rotation matrices**.
- **Two- and four-component fermions.**
  - * **Four-component fermions**: complications due to chirality projectors.
  - * **Weyl fermions**: easier, no projector (cf. SUSY theories).
- **Extending existing FeynRules models.**
  - * **Adding/changing/removing particles and operators**.
  - * Implementing the new model from scratch: **not efficient**.
- **Restricting more general existing FeynRules models.**
  - * **Setting** some parameters to 0 or 1.
  - * Implementing the new model from scratch: **not efficient**.
- **Simplifying implementations with Mathematica.**
  - * Implementing supersymmetric models in **superspace** (see below).
  - * Implementing D-dimensional models in D dimensions.

    [Not treated here: see arXiv:0906.2474]

Implementing particle mixing in FeynRules (1).

- **Concrete example: supersymmetric QCD**.
- **After supersymmetry (and electroweak symmetry) breaking**:
  * Particles with **same color representation, spin, quantum numbers mix**.
  * The mass matrices must be **diagonalized** through unitary rotations.

$$\begin{pmatrix} \tilde{u}_1 \\ \tilde{u}_2 \\ \tilde{u}_3 \\ \tilde{u}_4 \\ \tilde{u}_5 \\ \tilde{u}_6 \end{pmatrix} = R^{\tilde{u}} \begin{pmatrix} \tilde{u}_L \\ \tilde{c}_L \\ \tilde{t}_L \\ \tilde{u}_R \\ \tilde{c}_R \\ \tilde{t}_R \end{pmatrix}$$

  * **The squarks $\tilde{u}_i$ are the physical states**.

How to minimally modify the model file to implement the mixing?

- **Remark: this situation happens in many models**.
  * B/W boson mixing to photon/Z in the Standard Model.
  * Higgs mixing in Two-Higgs-Doublet models.
  * *etc...*

Implementing particle mixing in FeynRules (2).

- **No change to the Lagrangian**.
  - * **Easier to implement with gauge-eigenstates**.
  - * We do not want to make it **more complicated**.
- **Modifications at the particle level**.
  - * Use of the options `Unphysical` and `Definitions` of the particle class.
- **Modifications of `susyqcd.fr`**.
  - * Implementation of the **mass eigenstates**.
  - * Implementation of the **mixing matrix**.
  - * Modification of the fields `sqL` and `sqR` to render them **unphyiscal**.
  - * Modification of the fields `sqL` and `sqR` to **add the mixing relations**.
- **This procedure holds for any model**.

Squark mixing in SUSY QCD (1).

- **Our benchmark scenario: only top squarks do mix**.

$$\begin{pmatrix}\tilde u_1\\ \tilde u_2\\ \tilde u_3\\ \tilde u_4\\ \tilde u_5\\ \tilde u_6\end{pmatrix} = R^{\tilde u} \begin{pmatrix}\tilde u_L\\ \tilde c_L\\ \tilde t_L\\ \tilde u_R\\ \tilde c_R\\ \tilde t_R\end{pmatrix} = \begin{pmatrix}1 & 0 & 0 & 0 & 0 & 0\\ 0 & 1 & 0 & 0 & 0 & 0\\ 0 & 0 & \frac{\sqrt{2}}{2} & 0 & 0 & -\frac{\sqrt{2}}{2}\\ 0 & 0 & 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 0 & 1 & 0\\ 0 & 0 & \frac{\sqrt{2}}{2} & 0 & 0 & \frac{\sqrt{2}}{2}\end{pmatrix} \begin{pmatrix}\tilde u_L\\ \tilde c_L\\ \tilde t_L\\ \tilde u_R\\ \tilde c_R\\ \tilde t_R\end{pmatrix}$$

- **Modification of the model file**.
  1. Adding a **six-dimensional index**.
  2. Adding the **mixing matrix** $R^{\tilde u}$ in `M$Parameters`.
     - as well as left-handed and right-handed blocks.
  3. Adding the **physical squarks** $\tilde u_i$ in `M$ClassesDescription`.
  4. **Modifying the squark gauge-eigenstates**.

Squark mixing in SUSY QCD (2).

- **Step 1: a six dimensional-index**.

```
IndexRange[Index[Squark]] = Range[6];
IndexStyle[Squark, i];
```

Squark mixing in SUSY QCD (3).

- **Step 2: the mixing matrix**.

```
Ru == {
  ParameterType -> External,
  Indices       -> {Index[Squark], Index[Squark]},
  Value         ->  { ... },
  Unitary       -> True
},
RuL == {
  ParameterType -> Internal,
  Indices       -> {Index[Squark], Index[Gen]},
  Definitions   -> {RuL[i_,j_]:>Ru[i,j]/;NumericQ[j]}
},
RuR == {
  ParameterType -> Internal,
  Indices       -> {Index[Squark], Index[Gen]},
  Definitions   -> {RuR[i_,j_]:>Ru[i,j+3]/;NumericQ[j]}
}
```

* The `Squark` and `Gen` indices **do not have the same range**.
  - The definition is applied only if the second index is **numeric**.

* `RuL` **will be used for left-handed squark mixing**.

* `RuR` **will be used for right-handed squark mixing**.

Squark mixing in SUSY QCD (4).

- **From the mixing matrix to the fields.**

$$\begin{pmatrix}\tilde u_1\\ \tilde u_2\\ \tilde u_3\\ \tilde u_4\\ \tilde u_5\\ \tilde u_6\end{pmatrix} = R^{\tilde u} \begin{pmatrix}\tilde u_L\\ \tilde c_L\\ \tilde t_L\\ \tilde u_R\\ \tilde c_R\\ \tilde t_R\end{pmatrix} \Leftrightarrow \begin{pmatrix}\tilde u_L\\ \tilde c_L\\ \tilde t_L\\ \tilde u_R\\ \tilde c_R\\ \tilde t_R\end{pmatrix} = (R^{\tilde u})^\dagger \begin{pmatrix}\tilde u_1\\ \tilde u_2\\ \tilde u_3\\ \tilde u_4\\ \tilde u_5\\ \tilde u_6\end{pmatrix} = \begin{pmatrix}(R^{\tilde u}_L)^\dagger\\ \\ (R^{\tilde u}_R)^\dagger\end{pmatrix} \begin{pmatrix}\tilde u_1\\ \tilde u_2\\ \tilde u_3\\ \tilde u_4\\ \tilde u_5\\ \tilde u_6\end{pmatrix}$$

- **Mixing relations between gauge- and mass-eigenstates.**
  * `RuL` **will be used for left-handed squark mixing.**
  * `RuR` **will be used for right-handed squark mixing.**

$$\begin{pmatrix}\tilde u_L\\ \tilde c_L\\ \tilde t_L\end{pmatrix} = (R^{\tilde u}_L)^\dagger \begin{pmatrix}\tilde u_1\\ \tilde u_2\\ \tilde u_3\\ \tilde u_4\\ \tilde u_5\\ \tilde u_6\end{pmatrix} \qquad \begin{pmatrix}\tilde u_R\\ \tilde c_R\\ \tilde t_R\end{pmatrix} = (R^{\tilde u}_R)^\dagger \begin{pmatrix}\tilde u_1\\ \tilde u_2\\ \tilde u_3\\ \tilde u_4\\ \tilde u_5\\ \tilde u_6\end{pmatrix}$$

Squark mixing in SUSY QCD (5).

- **Step 3: declaration of the physical squark field.**

```
S[3] == {
  ClassName          -> su,
  SelfConjugate      -> False,
  Indices            -> {Index[Squark],Index[Colour]},
  FlavorIndex        -> Squark,
  QuantumNumbers     -> {Q -> 2/3},
  ClassMembers       -> {su1, su2, su3, su4, su5, su6},
  Mass               -> {Msu, {Msu1,300}, {Msu2,300},
                           {Msu3,300}, {Msu4,300},
                           {Msu5,300}, {Msu6,300}},
  Width              -> {{Wsu1,5}, {Wsu2,5}, {Wsu3,5},
                           {Wsu4,5}, {Wsu5,5}, {Wsu6,5}},
  PDG                -> {1000002, 1000004, 1000006,
                           2000002, 2000004, 2000006}
}
```

* We have now **six states**.

Squark mixing in SUSY QCD (6).

- **Step 4a: Modifying** sqL.

```
S[1] == {
  ClassName          -> sqL,
  Unphysical         -> True,
  SelfConjugate      -> False,
  Indices            -> {Index[Gen],Index[Colour]},
  FlavorIndex        -> Gen,
  QuantumNumbers     -> {Q -> 2/3},
  Definitions        -> { sqL[ff_,cc_] :> Module[{ff2},
    Conjugate[RuL[ff2,ff]] su[ff2,cc]] }
}
```

* The option Unphysical is set to True.
* The option Definitions relating sqL to su is **provided**.
  - ► This involves RuL.

Squark mixing in SUSY QCD (7).

- **Step 4b: Modifying** `sqR`.

```
S[2] == {
  ClassName          -> sqR,
  Unphysical         -> True,
  SelfConjugate      -> False,
  Indices            -> {Index[Gen],Index[Colour]},
  FlavorIndex        -> Gen,
  QuantumNumbers     -> {Q -> 2/3},
  Definitions        -> { sqR[ff_,cc_] :> Module[{ff2},
    Conjugate[RuR[ff2,ff]] su[ff2,cc]] }
}
```

  * The option `Unphysical` is set to `True`.
  * The option `Definitions` relating `sqR` to `su` is **provided**.
    - This involves `RuR`.

Manipulating Feynman rules (1).

- **Calculating all Feynman rules**.

```
In[6]:= FR = FeynmanRules[LVector + LMatter, ScreenOutput → False];
    Starting Feynman rule calculation.
    Collecting the different structures that enter the vertex...
    Found 9 possible non zero vertices.
    Start calculating vertices...
    9 vertices obtained.
```

- **Selecting given Feynman rules**.

```
SelectVertices[FR, Contains -> {go, su}]
```

  * We select Feynman rules containing **gluinos and squarks** (`Contains`).

Manipulating Feynman rules (2).

- **Reminder**:

$$\mathcal{L} = \sqrt{2} g_s \Big[- \tilde{q}^\dagger_{Li} T^a \big(\tilde{g}^a P_L q \big) + \big(\bar{q} P_L \tilde{g}^a \big) T^a \tilde{q}_{Ri} \Big] + \text{h.c.}$$

- **Selecting given Feynman rule**.

```
In[7]:= SelectVertices[FR, Contains → {go, su}]

     Applying seclection rules...

Out[7]= {{{{go, 1}, {su, 2}, {ūq, 3}}, i √2 gs (RuR*_{i2,f3} (P-)_{s3,s1} - RuL*_{i2,f3} (P+)_{s3,s1}) T^{a1}_{m3,m2}}}
```

  * The Feynman rule shows **squark related indices**.
  - A `Squark` index i_2 (from $1 \to 6$).
  - A `Colour` index m_2.

  * The Feynman rule depends on the **mixing matrices**.
  - `RuL` has one `Squark` and one `Gen` index (i_2 and f_3).
  - `RuR` has one `Squark` and one `Gen` index (i_2 and f_3).

- **The rotations have been performed automatically by FeynRules.**

Two-component and four-component fermions (1).

- **Some models are easier to implement using Weyl fermions**.
  - as any supersymmetric model.

- **Concrete example: supersymmetric QCD**.

  * We have a **four-component** version of the model file.
  * We want a **two-component** version of the model file.

How to minimally modify the model file to implement this?

1. We need to modify the **quark and gluino implementations** (fermions).
2. We need to provide **new Lagrangian terms** $\Rightarrow \ \mathcal{L}_4 \rightarrow \mathcal{L}_2$.
3. Cross-check: `FeynmanRules[L4-L2]` **must be empty**.

Two-component and four-component fermions (2).

- **Step 1a: Implementing a Weyl gluino $\chi_{\tilde{g}}$ (in `M$ClassesDescription`).**

```
W[1] == {
  ClassName      -> gow,
  Unphysical     -> True,
  Chirality      -> Left,
  SelfConjugate -> False,
  Indices        -> {Index[Gluon]},
  Definitions    -> {gow[inds__]->-I*goww[inds]}
}
```

* **Two-component fermion** ⇒ the label is `W[1]`.
* Defined **symbols**: `gow` (left-handed), `gowbar` (right-handed).
* `Unphysical`: Weyl fermion are not physical states.
  ► **contrary to Dirac and Majorana fields**.
* `Definitions`: *cf.* SLHA ⇒ i factor absorbed in gaugino definitions.

$$\Psi_{\tilde{g}} = \begin{pmatrix} i\chi_{\tilde{g}} \\ -i\bar{\chi}_{\tilde{g}} \end{pmatrix}$$

► definition of the Weyl fermion `goww`.

Two-component and four-component fermions (3).

- **Step 1a (ctn'd): Definition of the Weyl fermion** `goww`.

```
W[2] == {
  ClassName     -> goww,
  Unphysical    -> True,
  Chirality     -> Left,
  SelfConjugate -> False,
  Indices       -> {Index[Gluon]}
}
```

Two-component and four-component fermions (4).

- **Step 1b: Relating Weyl and Dirac gluinos**.

```
F[1] == {
  ClassName         -> go,
  WeylComponents    -> goww,
  SelfConjugate     -> True,
  Indices           -> {Index[Gluon]},
  PDG               -> 1000021,
  Mass              -> {Mgo,500},
  Width             -> {Wgo,10}
}
```

  * Through the `WeylComponents` option.
  * One single component $\equiv$ Majorana fermion.

Two-component and four-component fermions (5).

- **Step 1a: Implementing a left-handed Weyl quark** uqLw.

```
W[3] == {
  ClassName      -> uqLw,
  Unphysical     -> True,
  Chirality      -> Left,
  SelfConjugate  -> False,
  Indices        -> {Index[Gen],Index[Colour]},
  FlavorIndex    -> Gen,
  QuantumNumbers -> {Q-> 2/3}
}
```

- * **Two-component fermion** ⇒ the label is W[2].
- * Defined **symbols**: uqLw (left-handed), uqLwbar (right-handed).
- * Unphysical: Weyl fermion are not physical states.
  - ► **contrary to Dirac and Majorana fields**.
- * The electric charge is 2/3 (QuantumNumbers).

Two-component and four-component fermions (6).

- **Step 1a: Implementing a left-handed Weyl antiquark** `uqRw`.

```
W[4] == {
  ClassName       -> uqRw,
  Unphysical      -> True,
  Chirality       -> Right,
  SelfConjugate   -> False,
  Indices         -> {Index[Gen],Index[Colour]},
  FlavorIndex     -> Gen,
  QuantumNumbers -> {Q-> 2/3}
}
```

* **Two-component fermion** ⇒ the label is `W[4]`.
* Defined **symbols**: `uqRwbar` (left-handed), `uqRw` (right-handed).
* `Unphysical`: Weyl fermion are not physical states.
  - ▶ **contrary to Dirac and Majorana fields**.
* The electric charge is 2/3 (`QuantumNumbers`).

Two-component and four-component fermions (7).

- **Step 1b: Relating Weyl and Dirac quarks**.

```
F[2] == {
  ClassName       -> uq,
  WeylComponents -> {uqLw,uqRw},
  SelfConjugate  -> False,
  Indices         -> {Index[Gen], Index[Colour]},
  FlavorIndex     -> Gen,
  QuantumNumbers -> {Q -> 2/3},
  ClassMembers    -> {u, c, t},
  Mass            -> {Mu, {MU,2.55*^-3}, {MC,1.42}, {MT,172}},
  Width           -> {0, 0, {WT,1.50833649}},
  PDG             -> {2, 4, 6}
}
```

  * Through the `WeylComponents` option.
  * Two components (one left-handed + one right-handed) $\equiv$ Dirac fermion.

Implementing Lagrangians using Weyl fermions (1).

- **Kinetic and gauge interaction terms for quarks**:

$$\mathcal{L}_{\rm matter,kin} = \frac{i}{2}\Big[\chi^i_{qL}\sigma^\mu D_\mu\bar\chi_{qL,i} - D_\mu\chi^i_{qL}\sigma^\mu\bar\chi_{qL,i}\Big] + \frac{i}{2}\Big[\chi^i_{QR}\bar\sigma^\mu D_\mu\bar\chi_{QR,i} - D_\mu\chi^i_{QR}\bar\sigma^\mu\bar\chi_{QR,i}\Big] - m_q\Big[\bar\chi^i_{QR}\cdot\chi^i_{qL} + \chi^i_{QR}\cdot\bar\chi^i_{qL}\Big] + \text{squark terms}$$

- **Step 2: implementation**.

```
LkinW = ... +
      I/2 si[mu, sp1, sp2] (
        uqLw[sp1, ff, cc].DC[uqLwbar[sp2, ff, cc], mu] -
        DC[uqLw[sp1, ff, cc], mu].uqLwbar[sp2, ff, cc]) +
      I/2 sibar[mu,sp1,sp2] (
        uqRw[sp1, ff, cc].DC[uqRwbar[sp2, ff, cc], mu] -
        DC[uqRw[sp1, ff, cc], mu].uqRwbar[sp2, ff, cc]) -
       Mu[ff] (uqLw[sp, ff, cc].uqRwbar[sp, ff, cc] +
        uqLwbar[sp, ff, cc].uqRw[sp, ff, cc])
```

Implementing Lagrangians using Weyl fermions (2).

- **Checking the implementation** via the Feynman rules.

```
LK = Lkin - WeylToDirac[LkinW];
LK = OptimizeIndex[Expand[LK]];
FeynmanRules[LK,ScreenOutput->False]
```

* We compute the **difference of the two Lagrangians**.
* We transform **Weyl fermions to Dirac fermions** (`WeylToDirac`).
* We **optimize the index naming scheme** (`OptimizeIndex`).
  ► Renaming consistently the summed indices.
* We derive the Feynman rules.

Implementing Lagrangians using Weyl fermions (3).

- **Checking the implementation** via the Feynman rules.
- MATHEMATICA **ouput messages**.

```
In[6]:= LK = OptimizeIndex[Expand[WeylToDirac[Lkin - LkinW]]];
        FeynmanRules[LK, ScreenOutput → False]

    Starting Feynman rule calculation.
    Collecting the different structures that enter the vertex...
    Found 1 possible non zero vertices.
    Start calculating vertices...

    1 vertex obtained.
```

Out[7]= $\{\{\{\{G, 1\}, \{uq, 2\}, \{\bar{uq}, 3\}\}, i\, gs\, \delta_{f_2,f_3}\, T^{a_1}_{m_3,m_2} \left(\gamma^{\mu_1}_{s_3,s_2} - (\gamma^{\mu_1}.P_-)_{s_3,s_2} - (\gamma^{\mu_1}.P_+)_{s_3,s_2}\right)\}\}$

  * **It works!**

Implementing Lagrangians using Weyl fermions (4).

- **Checking the implementation** via the mass spectrum.

```
LK = Lkin - WeylToDirac[LkinW];
LK = OptimizeIndex[Expand[LK]];
Simplify[GetMassTerms[LK]]
```

  * We compute the **difference of the two Lagrangians**.
  * We transform **Weyl fermions to Dirac fermions** (`WeylToDirac`).
  * We **optimize the index naming scheme** (`OptimizeIndex`).
    - Renaming consistently the summed indices.
  * We extract the mass terms.

Implementing Lagrangians using Weyl fermions (5).

- **Checking the implementation** via the mass spectrum.
- MATHEMATICA **ouput messages**.

```
In[12]:= GetMassTerms[LK] // Simplify
    Neglecting all terms with more than 2 particles.
Out[12]= 0
```

- **We could also use `GetKineticTerms`, ...**
- **Exercise: implement the rest of the Weyl Lagrangian**.

Extending existing models.

- **Investigated models are often extensions of other, more minimal, models.**
  - Two-Higgs-Doublets models are a simple extensions of the SM.
  - R-parity violating supersymmetry extends R-parity conserving SUSY.
  - Additional $U(1)'$ interactions within the Standard Model.
  - *etc.*..
- FEYNRULES **offers an efficient way to implement extensions to models.**
  - The smaller model is **taken as it is.**
  - We implement a **new** FEYNRULES **model file.**
    - It contains the **additional gauge group/particles/operators.**
    - It is loaded **together** with the smaller model.

```
LoadModel["SmallModel.fr", "Extension.fr"];
```

- **No need to re-invent the wheel**...

Restricting existing models (1).

- **Restricted versions of a more general model**.
  * The Standard Model with vanishing light masses.
  * The Standard Model with vanishing CKM matrix.
  * The cMSSM (5 free parameters) vs. the MSSM (105 parameters).
  * *etc*...
- **Phenomenology: often enough to consider restricted models, not full ones**.
  * The full model renders the MC **slower**.
    - *e.g.*: the general MSSM has more than 10.000 vertices.
  * Many vertices are **subleading**.
    - *e.g.*: CKM suppression.
- FeynRules **offers an efficient way to implement restrictions to models**.

Restricting existing models (2).

- **The restrictions are implemented in a restriction file.**
  - The file contains **additional definitions** for parameters.
  - To be replaced **before** passing the information to the MC.
  - The restricted parameters **do not appear** at the MC level.
  - **The MC implementation is lighter $\Rightarrow$ more efficient**.
- **Example: a diagonal CKM matrix in** `DiagonalCKM.rst`.

```
M$Restrictions = {
  CKM[ i_, i_ ] -> 1,
  CKM[ i_?NumericQ, j_?NumericQ ] :> 0 /; (i =!= j)
};
```

- **The restrictions are loaded after the model.**

```
LoadModel["SM.fr"];
LoadRestriction["DiagonalCKM.rst"];
```

Restricting existing models (3).

- **The restrictions can be implemented at the MadGraph level.**
  - The restriction file is a **param_card**, with:
    - some parameters set to **zero**.
    - some parameters set to **unity**.
  - The filename is on the form `restrict_`*`restrictionname`*`.dat`.
  - It is loaded as

```
mg5> import model modelname-restrictionname
```

- **Effects in MadGraph.**
  - MadGraph replaces the zeros and ones by their numerical values (**removal of the associated symbols**).
  - MadGraph maps couplings with the **same value**.
  - MadGraph removes **vanishing couplings**.

- **Example: list the directory `models/sm` in MadGraph.**
  - By **default**, the file `restrict_default` is used.
  - To bypass all possible restrictions:

```
mg5> import model modelname-full
```

i.e., `sm-full`: **complete Standard Model** (CKM, non-zero masses, ...).

Outline

Fields and superfields (1).

- **Supported fields**.
  - Scalar fields.
  - Weyl, Dirac and Majorana fermions.
  - Vector (and ghost) fields.

Is this relevant / enough for the implementation of supersymmetric theories.

Yes, but ... let us investigate two short examples.

Fields and superfields (2).

- **Example 1: the superpotential for (s)leptons in the MSSM**.
  - Terribly expressed in terms of **components fields**, *i.e.*, scalars, **Dirac and Majorana fermions**, vector fields:

$$\mathcal{L}_{\rm W} = (\mathbf{y^e})_{ij}\Big[\widetilde{E}_R{}^i \widetilde{L}^j \cdot F_{H_D} + \widetilde{E}_R{}^i H_D \cdot F_L{}^j + \widetilde{L}^j \cdot H_D F_E^i + \widetilde{E}_R{}^i (\bar\psi_L^{cj} P_L \psi_{H_D}) + \widetilde{L}^j \cdot (\bar\psi_{H_D} P_L \psi_e{}^i) + (\bar\psi_e{}^i P_L \psi_L{}^j) \cdot H_D\Big]$$

  - Not very nicely expressed in terms of **components fields**, *i.e.*, scalars, **Weyl fermions**, vector fields:

$$\mathcal{L}_{\rm W} = (\mathbf{y^e})_{ij}\Big[\widetilde{E}_R{}^i \widetilde{L}^j \cdot F_{H_D} + \widetilde{E}_R{}^i H_D \cdot F_L{}^j + \widetilde{L}^j \cdot H_D F_E^i + \widetilde{E}_R{}^i (\chi_L{}^j \cdot \widetilde{H}_D) + \widetilde{L}^j \cdot (\widetilde{H}_D \cdot \chi_E{}^i) + (\chi_E{}^i \cdot \chi_L{}^j) \cdot H_D\Big]$$

  - Naturally expressed in terms of **superfields (1 terms)**:

$$\mathcal{L}_{\rm W} = \Big[- (\mathbf{y^e})_{ij}\ E^i (L^j \cdot \mathbf{H_D})\Big]\Big|_{\theta\cdot\theta}$$

Fields and superfields (3).

- **Example 1: the superpotential for (s)leptons in the MSSM**.
  - * Terribly expressed in terms of **components fields**, *i.e.*, scalars, **Dirac and Majorana fermions**, vector fields:

$$\mathcal{L}_{\rm W} = (\mathbf{y^e})_{ij}\Big[\widetilde{E}_R{}^i \widetilde{L}^j \cdot F_{H_D} + \widetilde{E}_R{}^i H_D \cdot F_L{}^j + \widetilde{L}^j \cdot H_D F_E^i$$
$$+ \widetilde{E}_R{}^i (\bar{\psi}_L^{cj} P_L \psi_{H_D}) + \widetilde{L}^j \cdot (\bar{\psi}_{H_D} P_L \psi_e{}^i) + (\bar{\psi}_e{}^i P_L \psi_L{}^j) \cdot H_D\Big]$$

* Are the **charge conjugated fields** correct?
* Are the signs in the **fermion flows** correct?
* **The superfield formalism seems more convenient...**

$$\mathcal{L}_{\rm W} = \Big[-(\mathbf{y^e})_{ij}\ E^i (L^j \cdot \mathbf{H_D}) \Big]\Big|_{\theta\cdot\theta}$$

Fields and superfields (4).

- **Kinetic terms and gauge interactions for left-handed (s)quarks in the MSSM**.
  - Terribly expressed in terms of **components fields**: *i.e.*, scalars, Dirac and Majorana fermions, vector fields (**13 terms)**:

$$\mathcal{L}_{\rm kin} \supset \ldots \qquad \text{[Censured: too ugly to appear on a slide].}$$

  - Not very nicely expressed in terms of **components fields**, i. e. scalars, Weyl fermions, vector fields (**13 terms)**:

$$\mathcal{L}_{\rm kin} \supset D_\mu \widetilde{Q}_i^\dagger D^\mu \widetilde{Q}^i + \frac{i}{2}(\chi_Q^i \sigma^\mu D_\mu \bar{\chi}_{Q\,i} - D_\mu \chi_Q^i \sigma^\mu \bar{\chi}_{Q\,i}) + F_{Q\,i}^\dagger F_Q^i$$
$$+ i\sqrt{2}\Big[\frac{1}{6} g' \widetilde{Q}^i \bar{\widetilde{B}} \cdot \bar{\chi}_{Q\,i} + g \bar{\widetilde{W}}^k \cdot \bar{\chi}_{Q\,i} \frac{\sigma^k}{2} \widetilde{Q}^i + g_s \bar{\widetilde{G}}^a \cdot \bar{\chi}_{Q\,i} T^a \widetilde{Q}^i + \text{h. c.}\Big]$$
$$- g' D_B \widetilde{Q}_i^\dagger \widetilde{Q}^i - g D_{W^k} \widetilde{Q}_i^\dagger \frac{\sigma^k}{2} \widetilde{Q}^i - g_s D_{G^a} \widetilde{Q}_i^\dagger \frac{T^a}{2} \widetilde{Q}^i$$

  - Naturally expressed in terms of **superfields (1 terms)**:

$$\mathcal{L}_{\rm kin} \supset \Big[Q_i^\dagger e^{-2\frac{1}{6}g' \mathbf{V_B}} e^{-2g \mathbf{V_{W^k}} \frac{\sigma^k}{2}} e^{-2g_s \mathbf{V_{G^a}} \frac{T^a}{2}} Q^i\Big]\Big|_{\theta\cdot\theta\bar\theta\cdot\bar\theta}$$

Fields and superfields (5).

- **Kinetic terms and gauge interactions for left-handed (s)quarks in the MSSM**.
  * Not very nicely expressed in terms of **components fields**, i. e. scalars, Weyl fermions, vector fields (**13 terms)**:

$$\mathcal{L}_{\rm kin} \supset D_\mu \widetilde{Q}_i^\dagger D^\mu \widetilde{Q}^i + \frac{i}{2}(\chi_Q^i \sigma^\mu D_\mu \bar\chi_{Qi} - D_\mu \chi_Q^i \sigma^\mu \bar\chi_{Qi}) + F_{Qi}^\dagger F_Q^i$$
$$+ i\sqrt{2}\Big[\frac{1}{6} g' \widetilde{Q}^i \bar{\widetilde{B}} \cdot \bar\chi_{Qi} + g \bar{\widetilde{W}}^k \cdot \bar\chi_{Qi} \frac{\sigma^k}{2} \widetilde{Q}^i + g_s \bar{\widetilde{G}}^a \cdot \bar\chi_{Qi} T^a \widetilde{Q}^i + \text{h. c.}\Big]$$
$$- g' D_B \widetilde{Q}_i^\dagger \widetilde{Q}^i - g D_{W^k} \widetilde{Q}_i^\dagger \frac{\sigma^k}{2} \widetilde{Q}^i - g_s D_{G^a} \widetilde{Q}_i^\dagger \frac{T^a}{2} \widetilde{Q}^i$$

  * Are all **relative signs and factors of i** correct (especially in the non-gauge-like interactions)?
  * **Four-component fermions**... (They are a pain, but required for MCs).
  * **The superfield formalism is more convenient...**

$$\mathcal{L}_{\rm kin} \supset \Big[Q_i^\dagger e^{-2\frac{1}{6}g' \mathbf{V_B}} e^{-2g \mathbf{V_{W^k}} \frac{\sigma^k}{2}} e^{-2g_s \mathbf{V_{G^a}} \frac{T^a}{2}} Q^i \Big]\Big|_{\theta\cdot\theta\bar\theta\cdot\bar\theta}$$

A superspace module in FeynRules.

Motivation for the superspace module in FeynRules

* **Natural** to implement any supersymmetric theory.
* **Zero probability** to introduce wrong signs, i factors,...
* Could be **a useful tool** for model building.
(not only a Lagrangian translator).

Superspace basics (1).

- **Superspace: adapted space to write down SUSY transformations naturally.**
- **Basic objects and their FeynRules (hardcoded) implementation.**
  - * The **Majorana spinor** $(\theta, \bar{\theta}) \Rightarrow$ a superspace point $\equiv G(x, \theta, \bar{\theta})$.
    - ◇ `theta` is defined internally as a regular **Weyl spinor**.
    - ◇ `theta` is a **mathematical object** $\Rightarrow$ `Unphyiscal->True`.

```
W[x1000] == {
 TeX              -> \[Theta],
 ClassName        -> theta,
 Chirality        -> Left,
 SelfConjugate    -> False,
 Unphysical       -> True}
```

  - * SUSY transformation parameters: **Majorana spinors** $(\varepsilon_1, \bar{\varepsilon}_1)$, $(\varepsilon_2, \bar{\varepsilon}_2)$, ...
    - ◇ The `epsx` are defined internally as a regular **Weyl spinor**, *e.g.*:

```
W[x1006] == {
  ClassName       -> eps6,
  Chirality       -> Left,
  SelfConjugate   -> False,
  Unphysical      -> True}
```

Superspace basics (2).

- **The supercharges $(\mathbf{Q}, \bar{\mathbf{Q}})$**: action to the left $\equiv G(0,\varepsilon,\bar\varepsilon)G(x,\theta,\bar\theta)$.
  - * Reminder: calculated by identifying the variations of the coordinates.
- **The superderivatives $(\mathbf{D}, \bar{\mathbf{D}})$**: action to the right $\equiv G(x,\theta,\bar\theta)G(0,\varepsilon,\bar\varepsilon)$.
  - * Reminder: calculated by identifying the variations of the coordinates.

$$Q_\alpha = -i(\partial_\alpha + i\sigma^\mu{}_{\alpha\dot\alpha}\bar\theta^{\dot\alpha}\partial_\mu) \quad \text{and} \quad \bar Q_{\dot\alpha} = i(\bar\partial_{\dot\alpha} + i\theta^\alpha\sigma^\mu{}_{\alpha\dot\alpha}\partial_\mu)\ ,$$
$$D_\alpha = \partial_\alpha - i\sigma^\mu{}_{\alpha\dot\alpha}\bar\theta^{\dot\alpha}\partial_\mu \quad \text{and} \quad \bar D_{\dot\alpha} = \bar\partial_{\dot\alpha} - i\theta^\alpha\sigma^\mu{}_{\alpha\dot\alpha}\partial_\mu\ .$$

$Q_\alpha(\exp)$ and $\bar Q_{\dot\alpha}(\exp)$

```
QSUSY    [exp_, alpha_]
QSUSYBar[exp_, alphadot_]
```

$D_\alpha(\exp)$ and $\bar D_{\dot\alpha}(\exp)$

```
DSUSY    [exp_, alpha_]
DSUSYBar[exp_, alphadot_]
```

- * **These operators can be used on any superspace expressions** (see below).

Superspace expressions: the general superfield (1).

- **Definition of a generic superfield.**
  - Most general (reducible) **expansion in the $\theta, \bar\theta$ variables**.
  - Can be expressed as,

$$\Phi(x,\theta,\bar\theta) = z(x) + \theta\cdot\xi(x) + \bar\theta\cdot\bar\zeta(x) + \theta\cdot\theta f(x) + \bar\theta\cdot\bar\theta g(x) + \theta\sigma^\mu\bar\theta\ v_\mu(x) + \bar\theta\cdot\bar\theta\theta\cdot\omega(x) + \theta\cdot\theta\cdot\bar\theta\cdot\bar\rho(x) + \theta\cdot\theta\bar\theta\cdot\bar\theta d(x).$$

  - **16 bosonic degrees of freedom**.
    - Four complex scalar fields z, f, g, d.
    - One complex vector field v_μ.
  - **16 fermionic degrees of freedom**.
    - Four Weyl fermions ξ, ζ, ω, ρ.

- **Reminder: spinor scalar product**.

$$\psi\cdot\chi = \psi^\alpha\chi_\alpha \qquad \text{and} \qquad \bar\psi\cdot\bar\chi = \bar\psi_{\dot\alpha}\bar\chi^{\dot\alpha}\ .$$

Superspace expressions: the general superfield (2).

$$\Phi(x,\theta,\bar\theta) = z(x) + \theta\cdot\xi(x) + \bar\theta\cdot\bar\zeta(x) + \theta\cdot\theta f(x) + \bar\theta\cdot\bar\theta g(x) + \theta\sigma^\mu\bar\theta\ v_\mu(x) + \bar\theta\cdot\bar\theta\theta\cdot\omega(x) + \theta\cdot\theta\cdot\bar\theta\cdot\bar\rho(x) + \theta\cdot\theta\bar\theta\cdot\bar\theta d(x).$$

- **Can be implemented in FeynRules-superfields**.
  - **Use of the `nc` environment** (keep the fermion ordering).
  - All the fermions are carrying **lower indices**.
    - We can define a **metric acting on spin space**.

$$\psi_\alpha = \varepsilon_{\alpha\beta}\psi^\beta\ , \qquad \psi^\alpha = \varepsilon^{\alpha\beta}\psi_\beta\ ,$$
$$\bar\chi_{\dot\alpha} = \varepsilon_{\dot\alpha\dot\beta}\bar\chi^{\dot\beta}\ , \qquad \bar\chi^{\dot\alpha} = \varepsilon^{\dot\alpha\dot\beta}\bar\chi_{\dot\beta}\ .$$

      (Beware of conventions: summation on the second index).
    - Use of the ε **rank-two antisymmetric tensors** (`Ueps` and `Deps`).
  - Remark: all the components must be declared properly and explicitely.

```
z + nc[theta[sp],xi[sp2]] Ueps[sp2,sp] + ...
```

Superfields.

- **The most general superfield contains too many degrees of freedom to describe the SUSY multiplets**.
- **We will put constraints on it**.
  * Definition of **chiral superfields**.
  * Definition of **vector superfields**.
- **SUSY multiplets for right-handed quarks**.
  * One **left-handed** spinor for the charge-conjugate right-handed quark.
  * The corresponding **charge-conjugate scalar** (anti)squark.
- **To be adapted in** `susyqcd.fr`.
  * Creation of the **antifundamental color representation** `{Tb,Colourb}`.
  * Definition of `UQRw`, a right-handed antiquark, *i.e.*, a left-handed spinor.
  * Definition of `SQR`, the corresponding antisquark.

`susyqcd.fr` **is now ready to include superfields**.
► **if you need help:** `addon.fr` **on the school wikipage**.

Superfields: chiral superfields (1).

- **Definition: the most general expansion in $\theta, \bar{\theta}$ satisfying $\bar{D}_{\dot{\alpha}}\Phi(x,\theta,\bar{\theta}) = 0$.**

$$\Phi(x,\theta,\bar{\theta}) = \phi(y) + \sqrt{2}\theta\cdot\psi(y) - \theta\cdot\theta F(y) \ .$$

  - $y^\mu = x^\mu - i\theta\sigma^\mu\bar{\theta}$.
  - It describes **matter multiplets**.
  - One scalar field ϕ, one Weyl fermion χ, one auxiliary field F.
    - On-shell: F is eliminated, 2 fermionic, 2 bosonic degrees of freedom.
    - Off-shell: 4 fermionic, 4 bosonic degrees of freedom.
    - **F is an unphysical complex scalar field**.

Superfields: chiral superfields (2).

- **Declaration of the left-handed quark superfield (in `M$Superfieds`).**

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

  - **Chiral superfield** ⇒ the label is `CSF[1]`.
  - The `Scalar` and `Weyl` components must be declared properly.
  - **The auxiliary field are automatically generated (not explicitely present).**
  - `Indices` and `QuantumNumbers` must match those of the components.

Superfields: chiral superfields (3).

- **Expansion in superspace:** $\Phi(y,\theta) = \phi(y) + \sqrt{2}\theta\cdot\psi(y) - \theta\cdot\theta F(y)$.

```
In[6]:= GrassmannExpand[QL]
```

$$\text{Out[6]=}\ \mathtt{sqL} - \mathtt{FTerm1}\,\theta_{sp\$1}.\theta_{sp\$1} + \sqrt{2}\,\theta_{sp\$1}.\mathtt{uqLw}_{sp\$1} - \frac{1}{4}\,\partial_{mu\$1}[\partial_{mu\$1}[\mathtt{sqL}]]\,\theta_{sp\$1}.\theta_{sp\$1}\,\bar\theta_{sp\$1dot}.\bar\theta_{sp\$1dot} - i\,\partial_{mu\$1}[\mathtt{sqL}]\,\theta_{sp\$1}.\bar\theta_{sp\$1dot}\left(\sigma^{mu\$1}\right)_{sp\$1,sp\$1dot} + \frac{i\,\partial_{mu\$1}[\mathtt{uqLw}_{sp\$1}].\bar\theta_{sp\$1dot}\,\theta_{sp\$2}.\theta_{sp\$2}\left(\sigma^{mu\$1}\right)_{sp\$1,sp\$1dot}}{\sqrt{2}}$$

- `GrassmannExpand` **expands a superfield expression** in terms of θ, $\bar\theta$.
- The **auxiliary** term, `FTerm1`, was automatically generated by FeynRules.
- **Automatic y-expansion** ($y^\mu = x^\mu - i\theta\sigma^\mu\bar\theta$) $\Rightarrow$ the fields depend on x.

Superfields: chiral superfields (4).

- **Extraction of the coefficients of the expansion:** $\Phi = \phi + \sqrt{2}\theta\cdot\psi - \theta\cdot\theta F + \ldots$.

```
In[7]:= ScalarComponent[QL]
Out[7]= sqL

In[8]:= ThetaComponent[QL]
Out[8]= √2 uqLw_alpha$882

In[9]:= Theta2Component[QL]
Out[9]= -FTerm1
```

  * Extraction of the **first three coefficients** (SUSY degrees of freedom).
  * Existing functions:
    - ◇ `ScalarComponent`
    - ◇ `ThetaComponent`
    - ◇ `Theta2Component`
    - ◇ `ThetabarComponent`
    - ◇ `Thetabar2Component`
    - ◇ `ThetaThetabarComponent`
    - ◇ `Theta2ThetabarComponent`
    - ◇ `Thetabar2ThetaComponent`
    - ◇ `Theta2Thetabar2Component`

Superfields: chiral superfields (5).

- **Declaration of the right-handed quark superfield (in `M$Superfieds`).**

```
CSF[2] == {
  ClassName      -> UR,
  Chirality      -> Left,
  Weyl           -> UQRw,
  Scalar         -> SQR,
  QuantumNumbers -> {Q->-2/3},
  Indices        -> {Index[Gen], Index[Colourb]}
}
```

  * **Chiral superfield** $\Rightarrow$ the label is `CSF[2]`.
  * The `Scalar` and `Weyl` components must be declared properly.
  * **The auxiliary field are automatically generated (not explicitely present).**
  * `Indices` and `QuantumNumbers` must match those of the components.
  * The components fields are the **charge-conjugate fields**.
    $\Rightarrow$ antifundamental color representation, opposite electric charge.

Using some superspace basic objects (1).

- **Transformation laws for a chiral superfield and its components**:
  - * In terms of **superfields**: $\delta_\varepsilon \Phi(x,\theta,\bar\theta) = i(\varepsilon\cdot Q + \bar Q \cdot \bar\varepsilon)\cdot \Phi(x,\theta,\bar\theta)$.
  - * In terms of **component fields** (depending on y, not x):

$$\delta_\varepsilon \phi = \sqrt{2}\varepsilon\cdot\psi \ , \quad \delta_\varepsilon \psi = -i\sqrt{2}\sigma^\mu \bar\varepsilon \partial_\mu \phi - \sqrt{2} F \varepsilon \ , \quad \delta_\varepsilon F = \ -i\sqrt{2}\partial_\mu \psi \sigma^\mu \bar\varepsilon \ .$$

  - * This depends on the **supercharges** `QSUSY` **and** `QSUSYBar`.
  - * The function `DeltaSUSY` is a better option...

```
DeltaPHI = DeltaSUSY[UR, eps1];
```

  - * `eps1` is the **transformnation parameter**.
  - * The `DeltaSUSY` **operator** corresponds to the superfield equation above.

Using some superspace basic objects (2).

- **The components of `DeltaPHI` read**:

$$\delta_\varepsilon\phi = \sqrt{2}\varepsilon\cdot\psi\ ,\quad \delta_\varepsilon\psi = -i\sqrt{2}\sigma^\mu\bar{\varepsilon}\partial_\mu\phi - \sqrt{2}F\varepsilon\ ,\quad \delta_\varepsilon F = -i\sqrt{2}\partial_\mu\psi\sigma^\mu\bar{\varepsilon}\ .$$

```
In[7]:= DeltaPHI = DeltaSUSY[UR, eps1];

In[8]:= ScalarComponent[Tonc[DeltaPHI]]

Out[8]= √2 UQRw_sp$1 . eps1_sp$1

In[10]:= Expand[ThetaComponent[Tonc[DeltaPHI]] / Sqrt[2]]

Out[10]= -√2 FTerm2 eps1_alpha$1783 - i √2 ∂_mu$1[SQR] eps1†_sp$1dot (σ^mu$1)_alpha$1783,sp$1dot

In[12]:= Expand[Theta2Component[Tonc[DeltaPHI]] / (-1)]

Out[12]= -i √2 ∂_mu$1[UQRw_sp$1] . eps1†_sp$1dot (σ^mu$1)_sp$1,sp$1dot
```

* `Tonc` **breaks dot products** and restore the `nc` structure (fermion ordering).
* This is **mandatory** in order to have the `xxxComponent` to work properly.
* The $\sqrt{2}$ and the minus sign are related to:

$$\Phi = \phi\ +\ \sqrt{2}\theta\cdot\psi - \theta\cdot\theta F + \ldots$$

Superfields: vector superfields (1).

- **We apply the constraint $\Phi = \Phi^\dagger$ on a general superfield.**
- **In the Wess-Zumino gauge, we have**:

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

  - This describes **gauge supermultiplets**.
  - One Majorana fermion $(\lambda,\bar\lambda)$, one (massless) gauge boson v, one auxiliary field D.
    - On-shell: D eliminated, 2 fermionic, 2 bosonic degrees of freedom.
    - Off-shell: 4 fermionic, 4 bosonic degrees of freedom.
    - **D is an unphysical real scalar field**.

Superfields: vector superfields (2).

- **Declaration of the $SU(3)_c$ vector superfield (in `M$Superfieds`)**.

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

* **Vector superfield** ⇒ the label is `VSF[1]`.
* The `Gaugino` and `GaugeBoson` components must be declared properly.
* **The auxiliary field are automatically generated (not explicitely present)**.
* `Indices` and `QuantumNumbers` must match those of the components.

Superfields: vector superfields (3).

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

```
SU3C == {
  Abelian           -> False,
  Superfield        -> GSF,
  CouplingConstant  -> gs,
  StructureConstant -> f,
  Representations   -> {{T,Colour}, {Tb,Colourb}}
}
```

* Through the option `Superfield`.
* This replace the option `GaugeBoson`.

Superfields: vector superfields (4).

- **Expansion in superspace with FeynRules:**

$$\Phi = \theta\sigma^\mu\bar\theta v_\mu + i\theta\cdot\theta\bar\theta\cdot\bar\lambda - i\bar\theta\cdot\bar\theta\theta\cdot\lambda + \frac{1}{2}\theta\cdot\theta\;\bar\theta\cdot\bar\theta D\ .$$

In[14]:= `GrassmannExpand[GSF[aa]]`

Out[14]= $i\,\text{gow}^\dagger_{sp\$1dot,aa}\cdot\bar\theta_{sp\$1dot}\,\theta_{sp\$1}\cdot\theta_{sp\$1} - i\,\theta_{sp\$1}\cdot\text{gow}_{sp\$1,aa}\,\bar\theta_{sp\$1dot}\cdot\bar\theta_{sp\$1dot} + \frac{1}{2}\theta_{sp\$1}\cdot\theta_{sp\$1}\,\bar\theta_{sp\$1dot}\cdot\bar\theta_{sp\$1dot}\,\text{DTerm3}_{aa} + \theta_{sp\$1}\cdot\bar\theta_{sp\$1dot}\,G_{mu\$1,aa}\left(\sigma^{mu\$1}\right)_{sp\$1,sp\$1dot}$

In[9]:= `ThetaThetabarComponent[GSF[aa]]`

Out[9]= $G_{mu\$743,aa}$

In[11]:= `Theta2ThetabarComponent[GSF[aa]]`

Out[11]= $i\,\text{gow}^\dagger_{alphadot\$822,aa}$

In[12]:= `Thetabar2ThetaComponent[GSF[aa]]`

Out[12]= $-i\,\text{gow}_{alpha\$900,aa}$

In[13]:= `Theta2Thetabar2Component[GSF[aa]]`

Out[13]= $\frac{\text{DTerm3}_{aa}}{2}$

* `DTerm3` was **automatically generated**.

Superfields: vector superfields (5).

- **Some properties of vector superfields in the Wess-Zumino gauge**:

$$\Phi^2_{W.Z.} = \frac{1}{2}\theta\cdot\theta\bar\theta\cdot\bar\theta v^\mu v_\mu, \quad \Phi^3_{W.Z.} = 0.$$

```
In[7]:= GrassmannExpand[GSF[aa] GSF[bb]]
Out[7]= 1/2 θsp$1 . θsp$1 θ̄sp$1dot . θ̄sp$1dot Gmu$1,aa Gmu$1,bb

In[8]:= GrassmannExpand[GSF[aa] GSF[bb] GSF[cc]]
Out[8]= 0
```

- **The superfield strength tensor is built from associated spinorial superfields**:

$$W_\alpha = -\frac{1}{4}\bar D\cdot\bar D\ e^{2gV} D_\alpha e^{-2gV}, \quad \overline{W}_{\dot\alpha} = -\frac{1}{4} D\cdot D\ e^{-2gV}\bar D_{\dot\alpha} e^{2gV}.$$

W_α, $(W_\alpha)_{ij}$, W^a_α, $\overline{W}_{\dot\alpha}$, $\overline{W}^a_{\dot\alpha}$, $(\overline{W}_{\dot\alpha})_{ij}$

```
SuperfieldStrengthL[ SF, lower spin index ]
SuperfieldStrengthL[ SF, spin index, gauge index/indices ]
SuperfieldStrengthR[ SF, lower spin index ]
SuperfieldStrengthR[ SF, spin index, gauge index/indices ]
```

Superfields: vector superfields (6).

- **Spinorial superfields**:

$$W_\alpha(y,\theta) = -2g\left(-i\lambda_\alpha + \left[-\frac{i}{2}(\sigma^\mu\bar\sigma^\nu\theta)_\alpha F_{\mu\nu} + \theta_\alpha D\right] - \theta\cdot\theta(\sigma^\mu D_\mu\bar\lambda)_\alpha\right) .$$

```
In[19]:= ScalarComponent[SuperfieldStrengthL[GSF, alpha, aa]]

Out[19]= 2 i gs nc[gow_{alpha,aa}]

In[20]:= ThetaComponent[SuperfieldStrengthL[GSF, alpha, aa]]

Out[20]= -2 gs DTerm3_{aa} - i gs ∂_{mu$2}[G_{mu$1,aa}] nc[TensDot2[(σ^{mu$1})_{alpha,sp$1dot}, (σbar^{mu$2})^{sp$1dot,alpha$7592}][
           {down, Left, alpha}, {up, Left, alpha$7592}]] +
         i gs^2 f_{aa,Gluon$1,Gluon$2} G_{mu$1,Gluon$1} G_{mu$2,Gluon$2} nc[TensDot2[(σ^{mu$1})_{alpha,sp$1dot}, (σbar^{mu$2})^{sp$1dot,alpha$7592}][
           {down, Left, alpha}, {up, Left, alpha$7592}]] +
         i gs ∂_{mu$2}[G_{mu$1,aa}] nc[TensDot2[(σ^{mu$2})_{alpha,sp$1dot}, (σbar^{mu$1})^{sp$1dot,alpha$7592}][
           {down, Left, alpha}, {up, Left, alpha$7592}]]

In[21]:= Theta2Component[SuperfieldStrengthL[GSF, alpha, aa]]

Out[21]= 2 gs nc[TensDot2[(σ^{mu$1})_{alpha,sp$1dot}, ∂_{mu$1}[gow†_{sp$1dot,aa}]][down, Left, alpha]] +
         2 gs^2 f_{aa,Gluon$1,Gluon$2} G_{mu$1,Gluon$1}
          nc[TensDot2[(σ^{mu$1})_{alpha,sp$1dot}, gow†_{sp$1dot,Gluon$2}][down, Left, alpha]]
```

- * FeynRules has performed the ***y*-expansion**.
- * Spinors with **non-lower spin index** are embedded in a `TensDot2` structure.

Vector Lagrangians (1).

- **Each vector superfield is attached to one gauge group.**
- **Vector superfield interactions are obtained by calculating superfield strengths.**
  - * **Abelian groups.**

$$\begin{aligned}\mathcal{L} = &\ \frac{1}{4} W^\alpha W_{\alpha}{}_{|\theta\theta} + \frac{1}{4} \bar{W}_{\dot\alpha} \bar{W}^{\dot\alpha}{}_{|\bar\theta\bar\theta} \\ = &\ -\frac{1}{4} F_{\mu\nu} F^{\mu\nu} + i \bar\lambda \bar\sigma^\mu \partial_\mu \lambda + \frac{1}{2} D^2 \ .\end{aligned}$$

  - * **Non-abelian groups.**

$$\begin{aligned}\mathcal{L} = &\ \frac{1}{16 g^2 \tau_{\mathcal{R}}} \mathrm{Tr}(W^\alpha W_\alpha)_{|\theta\theta} + \frac{1}{16 g^2 \tau_{\mathcal{R}}} \mathrm{Tr}(\bar{W}_{\dot\alpha} \bar{W}^{\dot\alpha})_{|\bar\theta\bar\theta} \\ = &\ -\frac{1}{4} F^a_{\mu\nu} F_a^{\mu\nu} + i\, \bar\lambda_a \, \bar\sigma^\mu \, D_\mu \lambda^a + \frac{1}{2} D_a D^a\end{aligned}$$

  $\Rightarrow$ Interactions between gauge-bosons and gauginos.

- **Automatic extraction of the vector Lagrangian of a model:**

```
(* all vector superfields *) VSFKineticTerms[]
(* one vector superfield  *) VSFKineticTerms[ GSF ]
```

Vector Lagrangians (2).

- **Non-abelian superfield strengths (with Weyl fermions)**:

$$\mathcal{L} = \frac{1}{16 g^2 \tau_{\mathcal{R}}} \mathrm{Tr}(W^\alpha W_\alpha)_{|\theta\theta} + \text{h.c.} = -\frac{1}{4} F^a_{\mu\nu} F_a^{\mu\nu} + i\,\bar\lambda_a \bar\sigma^\mu D_\mu \lambda^a + \frac{1}{2} D_a D^a \ .$$

- **In SUSY-QCD**:.

```
In[28]:= LVectorSF = Theta2Component[VSFKineticTerms[]];
         LVectorSF = LVectorSF + HC[LVectorSF]
```

$$\text{Out[29]=}\ -\frac{1}{2}\,\partial_{\text{mu\$2}}[G_{\text{mu\$1,Gluon\$1}}]^2 + \frac{1}{2}\,\partial_{\text{mu\$2}}[G_{\text{mu\$1,Gluon\$1}}]\,\partial_{\text{mu\$1}}[G_{\text{mu\$2,Gluon\$1}}] +$$
$$\frac{\text{DTerm3}^2_{\text{Gluon\$1}}}{2} + gs\,\partial_{\text{mu\$2}}[G_{\text{mu\$1,Gluon\$1}}]\, f_{\text{Gluon\$1,Gluon\$2,Gluon\$3}}\, G_{\text{mu\$1,Gluon\$2}}\, G_{\text{mu\$2,Gluon\$3}} -$$
$$\frac{1}{4}\, gs^2\, f_{\text{Gluon\$1,Gluon\$2,Gluon\$3}}\, f_{\text{Gluon\$4,Gluon\$5,Gluon\$3}}\, G_{\text{mu\$1,Gluon\$1}}\, G_{\text{mu\$1,Gluon\$4}}\, G_{\text{mu\$2,Gluon\$2}}\, G_{\text{mu\$2,Gluon\$5}} +$$
$$\frac{1}{2}\, i\, gow_{\text{sp\$1,Gluon\$1}} \cdot \partial_{\text{mu\$1}}[gow^\dagger_{\text{sp\$1dot,Gluon\$1}}]\,(\sigma^{\text{mu\$1}})_{\text{sp\$1,sp\$1dot}} -$$
$$\frac{1}{2}\, i\, gs\, gow_{\text{sp\$1,Gluon\$1}} \cdot gow^\dagger_{\text{sp\$1dot,Gluon\$2}}\, f_{\text{Gluon\$1,Gluon\$2,Gluon\$3}}\, G_{\text{mu\$1,Gluon\$3}}\,(\sigma^{\text{mu\$1}})_{\text{sp\$1,sp\$1dot}} -$$
$$\frac{1}{2}\, i\, \partial_{\text{mu\$1}}[gow_{\text{sp\$1dot,Gluon\$1}}] \cdot gow^\dagger_{\text{sp\$1,Gluon\$1}}\,(\sigma^{\text{mu\$1}})_{\text{sp\$1dot,sp\$1}} +$$
$$\frac{1}{2}\, i\, gs\, gow_{\text{sp\$1dot,Gluon\$2}} \cdot gow^\dagger_{\text{sp\$1,Gluon\$1}}\, f_{\text{Gluon\$1,Gluon\$2,Gluon\$3}}\, G_{\text{mu\$1,Gluon\$3}}\,(\sigma^{\text{mu\$1}})_{\text{sp\$1dot,sp\$1}}$$

Is this correct?

Vector Lagrangians (3).

- **Checking the implementation** via the Feynman rules.

```
LV = LVectorW - LVectorSF;
LV = WeylToDirac[LV];
LV = OptimizeIndex[Expand[LV]];
rules = FeynmanRules[LV,ScreenOutput->False];
rules = {#[[1]], OptimizeIndex[Expand[#[[2]]]]} &/@ rules
```

  * We compute the **difference of the two Lagrangians**.
  * We transform **Weyl fermions to Dirac fermions** (`WeylToDirac`).
  * We **optimize the index naming scheme** (`OptimizeIndex`).
    - ► Renaming consistently the summed indices.
  * We derive the Feynman rules.

Vector Lagrangians (4).

- **Checking the implementation** via the Feynman rules.
- MATHEMATICA **ouput messages**.

```
Starting Feynman rule calculation.
Collecting the different structures that enter the vertex...
Found 2 possible non zero vertices.
Start calculating vertices...
1 vertex obtained.
{{{{G,1}, {G,2}, {G,3},{G,4}}, 0}}
```

Matter Lagrangians (1).

- **Lagrangian associated to the chiral superfield content of the theory**.
  - Contains gauge interactions and kinetic terms for **chiral superfields**.
  - Is entirely **fixed by SUSY and gauge invariance**
  - **Example for** $SU(3)_c \times SU(2)_L \times U(1)_Y$.

$$\mathcal{L} = \left[\Phi^\dagger(x,\theta,\bar\theta) e^{-2y_\Phi g' \mathbf{V_B}} e^{-2g\mathbf{V_W}} e^{-2g_s \mathbf{V_G}} \Phi(x,\theta,\bar\theta)\right]\Big|_{\theta\cdot\theta\bar\theta\cdot\bar\theta}$$

(Non-abelian vector superfields contains group representation matrices.)

- **Automatic extraction of the matter Lagrangian of a model:**

```
CSFKineticTerms[]
CSFKineticTerms[ UR ] + CSFKineticTerms[ QL ]
```

Matter Lagrangians (2).

- **Generic matter kinetic Lagrangian**:

$$\mathcal{L} = D_\mu\phi^\dagger D^\mu\phi - \frac{i}{2}\left(D_\mu\bar\psi\bar\sigma^\mu\psi - \bar\psi\bar\sigma^\mu D_\mu\psi\right) + i\sqrt{2}g\bar\lambda^a\cdot\bar\psi T_a\phi - i\sqrt{2}g\phi^\dagger T_a\psi\cdot\lambda^a + FF^\dagger - gD^a\phi^\dagger T^a\phi\ .$$

```
In[65]:= LMatterSF = Theta2Thetabar2Component[CSFKineticTerms[]]
```

Out[65]=

$$\tfrac{1}{2}\,\partial_{\text{mu\$1}}[\text{sqL}_{\text{Gen\$1,Colour\$1}}]\,\partial_{\text{mu\$1}}[\text{sqL}^\dagger_{\text{Gen\$1,Colour\$1}}] + \tfrac{1}{2}\,\partial_{\text{mu\$1}}[\text{SQR}_{\text{Gen\$1,Colourb\$1}}]\,\partial_{\text{mu\$1}}[\text{SQR}^\dagger_{\text{Gen\$1,Colourb\$1}}] + \text{FTerm1}_{\text{Gen\$1,Colour\$1}}\,\text{FTerm1}^\dagger_{\text{Gen\$1,Colour\$1}} +$$

$$\text{FTerm2}_{\text{Gen\$1,Colourb\$1}}\,\text{FTerm2}^\dagger_{\text{Gen\$1,Colourb\$1}} - \tfrac{1}{2}\,i\,\partial_{\text{mu\$1}}[\text{uqLw}_{\text{sp\$1,Gen\$1,Colour\$1}}].\text{uqLw}^\dagger_{\text{sp\$1dot,Gen\$1,Colour\$1}}\,(\sigma^{\text{mu\$1}})_{\text{sp\$1,sp\$1dot}} -$$

$$\tfrac{1}{2}\,i\,\partial_{\text{mu\$1}}[\text{UQRw}_{\text{sp\$1,Gen\$1,Colourb\$1}}].\text{UQRw}^\dagger_{\text{sp\$1dot,Gen\$1,Colourb\$1}}\,(\sigma^{\text{mu\$1}})_{\text{sp\$1,sp\$1dot}} +$$

$$\tfrac{1}{2}\,i\,\text{uqLw}_{\text{sp\$2,Gen\$1,Colour\$1}}.\partial_{\text{mu\$1}}[\text{uqLw}^\dagger_{\text{sp\$2dot,Gen\$1,Colour\$1}}]\,(\sigma^{\text{mu\$1}})_{\text{sp\$2,sp\$2dot}} +$$

$$\tfrac{1}{2}\,i\,\text{UQRw}_{\text{sp\$2,Gen\$1,Colourb\$1}}.\partial_{\text{mu\$1}}[\text{UQRw}^\dagger_{\text{sp\$2dot,Gen\$1,Colourb\$1}}]\,(\sigma^{\text{mu\$1}})_{\text{sp\$2,sp\$2dot}} - \tfrac{1}{4}\,\partial_{\text{mu\$1}}[\partial_{\text{mu\$1}}[\text{sqL}^\dagger_{\text{Gen\$1,Colour\$1}}]]\,\text{sqL}_{\text{Gen\$1,Colour\$1}} -$$

$$\tfrac{1}{4}\,\partial_{\text{mu\$1}}[\partial_{\text{mu\$1}}[\text{sqL}_{\text{Gen\$1,Colour\$1}}]]\,\text{sqL}^\dagger_{\text{Gen\$1,Colour\$1}} - \tfrac{1}{4}\,\partial_{\text{mu\$1}}[\partial_{\text{mu\$1}}[\text{SQR}^\dagger_{\text{Gen\$1,Colourb\$1}}]]\,\text{SQR}_{\text{Gen\$1,Colourb\$1}} -$$

$$\tfrac{1}{4}\,\partial_{\text{mu\$1}}[\partial_{\text{mu\$1}}[\text{SQR}_{\text{Gen\$1,Colourb\$1}}]]\,\text{SQR}^\dagger_{\text{Gen\$1,Colourb\$1}} + i\sqrt{2}\,\text{gs}\,\text{gow}^\dagger_{\text{sp\$1dot,Gluon\$1}}.\text{uqLw}^\dagger_{\text{sp\$1dot,Gen\$1,Colour\$1}}\,\text{sqL}_{\text{Gen\$1,Colour\$2}}\,T^{\text{Gluon\$1}}_{\text{Colour\$1,Colour\$2}} -$$

$$i\,\text{gs}\,\partial_{\text{mu\$1}}[\text{sqL}^\dagger_{\text{Gen\$1,Colour\$1}}]\,G_{\text{mu\$1,Gluon\$1}}\,\text{sqL}_{\text{Gen\$1,Colour\$2}}\,T^{\text{Gluon\$1}}_{\text{Colour\$1,Colour\$2}} -$$

$$\text{gs}\,\text{uqLw}_{\text{sp\$2,Gen\$1,Colour\$1}}.\text{uqLw}^\dagger_{\text{sp\$2dot,Gen\$1,Colour\$2}}\,G_{\text{mu\$1,Gluon\$1}}\,(\sigma^{\text{mu\$1}})_{\text{sp\$2,sp\$2dot}}\,T^{\text{Gluon\$1}}_{\text{Colour\$2,Colour\$1}} -$$

$$i\sqrt{2}\,\text{gs}\,\text{uqLw}_{\text{sp\$2,Gen\$1,Colour\$1}}.\text{gow}_{\text{sp\$2,Gluon\$1}}\,\text{sqL}^\dagger_{\text{Gen\$1,Colour\$2}}\,T^{\text{Gluon\$1}}_{\text{Colour\$2,Colour\$1}} +$$

$$i\,\text{gs}\,\partial_{\text{mu\$1}}[\text{sqL}_{\text{Gen\$1,Colour\$1}}]\,G_{\text{mu\$1,Gluon\$1}}\,\text{sqL}^\dagger_{\text{Gen\$1,Colour\$2}}\,T^{\text{Gluon\$1}}_{\text{Colour\$2,Colour\$1}} - \text{gs}\,\text{DTerm3}_{\text{Gluon\$1}}\,\text{sqL}_{\text{Gen\$1,Colour\$1}}\,\text{sqL}^\dagger_{\text{Gen\$1,Colour\$2}}\,T^{\text{Gluon\$1}}_{\text{Colour\$2,Colour\$1}} +$$

$$\text{gs}^2\,G_{\text{mu\$1,Gluon\$1}}\,G_{\text{mu\$1,Gluon\$2}}\,\text{sqL}_{\text{Gen\$1,Colour\$1}}\,\text{sqL}^\dagger_{\text{Gen\$1,Colour\$2}}\,T^{\text{Gluon\$1}}_{\text{Colour\$2,Colour\$3}}\,T^{\text{Gluon\$2}}_{\text{Colour\$3,Colour\$1}} +$$

$$i\sqrt{2}\,\text{gs}\,\text{gow}^\dagger_{\text{sp\$1dot,Gluon\$1}}.\text{UQRw}^\dagger_{\text{sp\$1dot,Gen\$1,Colourb\$1}}\,\text{SQR}_{\text{Gen\$1,Colourb\$2}}\,\text{Tb}^{\text{Gluon\$1}}_{\text{Colourb\$1,Colourb\$2}} -$$

$$i\,\text{gs}\,\partial_{\text{mu\$1}}[\text{SQR}^\dagger_{\text{Gen\$1,Colourb\$1}}]\,G_{\text{mu\$1,Gluon\$1}}\,\text{SQR}_{\text{Gen\$1,Colourb\$2}}\,\text{Tb}^{\text{Gluon\$1}}_{\text{Colourb\$1,Colourb\$2}} -$$

$$\text{gs}\,\text{UQRw}_{\text{sp\$2,Gen\$1,Colourb\$1}}.\text{UQRw}^\dagger_{\text{sp\$2dot,Gen\$1,Colourb\$2}}\,G_{\text{mu\$1,Gluon\$1}}\,(\sigma^{\text{mu\$1}})_{\text{sp\$2,sp\$2dot}}\,\text{Tb}^{\text{Gluon\$1}}_{\text{Colourb\$2,Colourb\$1}} -$$

$$i\sqrt{2}\,\text{gs}\,\text{UQRw}_{\text{sp\$2,Gen\$1,Colourb\$1}}.\text{gow}_{\text{sp\$2,Gluon\$1}}\,\text{SQR}^\dagger_{\text{Gen\$1,Colourb\$2}}\,\text{Tb}^{\text{Gluon\$1}}_{\text{Colourb\$2,Colourb\$1}} +$$

$$i\,\text{gs}\,\partial_{\text{mu\$1}}[\text{SQR}_{\text{Gen\$1,Colourb\$1}}]\,G_{\text{mu\$1,Gluon\$1}}\,\text{SQR}^\dagger_{\text{Gen\$1,Colourb\$2}}\,\text{Tb}^{\text{Gluon\$1}}_{\text{Colourb\$2,Colourb\$1}} -$$

$$\text{gs}\,\text{DTerm3}_{\text{Gluon\$1}}\,\text{SQR}_{\text{Gen\$1,Colourb\$1}}\,\text{SQR}^\dagger_{\text{Gen\$1,Colourb\$2}}\,\text{Tb}^{\text{Gluon\$1}}_{\text{Colourb\$2,Colourb\$1}} +$$

$$\text{gs}^2\,G_{\text{mu\$1,Gluon\$1}}\,G_{\text{mu\$1,Gluon\$2}}\,\text{SQR}_{\text{Gen\$1,Colourb\$1}}\,\text{SQR}^\dagger_{\text{Gen\$1,Colourb\$2}}\,\text{Tb}^{\text{Gluon\$1}}_{\text{Colourb\$2,Colourb\$3}}\,\text{Tb}^{\text{Gluon\$2}}_{\text{Colourb\$3,Colourb\$1}}$$

Is this correct? ⇒ NO!!!

Full supersymmetric Lagrangians (1).

- **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.

Full supersymmetric Lagrangians (2).

- **In the case of a supersymmetric QCD theory**:

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

```
LMatterSF = Theta2Thetabar2Component[CSFKineticTerms[]];
LVectorSF = Theta2Component[VSFKineticTerms[]];
LVectorSF = LVectorSF + HC[LVectorSF];
LSoft = - Mu[ff] (uqLw[sp,ff,cc].UQRw[sp,ff,cc] +
             uqLwbar[sp,ff,cc].UQRwbar[sp,ff,cc]) -
          MsqL[ff]^2 sqLbar[ff,cc] sqL[ff,cc] -
          MsqR[ff]^2 SQRbar[ff,cc] SQR[ff,cc] -
          1/2 Mgo (goww[s1,a].goww[s1,a] +
             gowwbar[s1,a].gowwbar[s1,a]);
lagr = LMatterSF + LVectorSF + LSoft;
```

* There is **no superpotential**.

Full supersymmetric Lagrangians (3).

- **Solution of the equation of motions**.
  - Get rid of the auxiliary *D*-fields and *F*-fields.
  - Through their **equations of motion**.

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

- **Back to four-component fermions**.
  - Usual FeynRules routine.
  - We **replace antifundamental color representations by fundamental ones**. (cf. MC code requirements).

```
Colourb=Colour;
lagr = lagr/.Tb[aa_,ii_,jj_]->-T[aa,jj,ii];
lagr = WeylToDirac[ lagr ] ;
```

Full supersymmetric Lagrangians (4).

- **Checking the implementation** via the Feynman rules.

```
LL = lagr - LVector - LMatter;
LL = OptimizeIndex[Expand[LL]];
rules = FeynmanRules[LL,ScreenOutput->False];
rules = {#[[1]], OptimizeIndex[Expand[#[[2]]]]} &/@ rules;
rules = DeleteCases[rules, {_, 0}];
```

  - We compute the **difference of the two Lagrangians**.
  - We **optimize the index naming scheme** (`OptimizeIndex`).
    - Renaming consistently the summed indices.
  - We derive the Feynman rules.

Full supersymmetric Lagrangians (5).

- **Checking the implementation** via the Feynman rules.

```
In[6]:= LMatterSF = Theta2Thetabar2Component[CSFKineticTerms[]];
        LVectorSF = Theta2Component[VSFKineticTerms[]];
        LVectorSF = LVectorSF + HC[LVectorSF];
        LSoft = -Mu[ff] (uqLw[sp, ff, cc].UQRw[sp, ff, cc] + uqLwbar[sp, ff, cc].UQRwbar[sp, ff, cc]) -
           MsqL[ff] ^ 2 sqLbar[ff, cc] sqL[ff, cc] - MsqR[ff] ^ 2 SQRbar[ff, cc] SQR[ff, cc] -
           1 / 2 Mgo (goww[s1, a].goww[s1, a] + gowwbar[s1, a].gowwbar[s1, a]);
        lagr = LMatterSF + LVectorSF + LSoft;
        lagr = SolveEqMotionD[lagr];
        lagr = SolveEqMotionF[lagr];

In[13]:= Colourb = Colour;
         lagr = lagr /. Tb[aa_, ii_, jj_] → -T[aa, jj, ii];
         lagr = WeylToDirac[lagr];

In[30]:= LL = lagr - LVector - LMatter;
         LL = OptimizeIndex[Expand[LL]];
         rules = FeynmanRules[LL, ScreenOutput → False];
         rules = {#[[1]], Simplify[OptimizeIndex[Expand[#[[2]]]]]} & /@ rules;
         rules = DeleteCases[rules, {_, 0}]

        Starting Feynman rule calculation.
        Collecting the different structures that enter the vertex...
        Found 9 possible non zero vertices.
        Start calculating vertices...
        6 vertices obtained.
```

Out[34]= $\left\{\left\{\left\{\{G, 1\}, \{uq, 2\}, \{\bar{uq}, 3\}\right\}, -i\, gs\, \delta_{f_2, f_3}\, T^{a_1}_{m_3, m_2} \left(\gamma^{Ext[1]}_{s_3, s_2} - \left(\gamma^{Ext[1]}.P_-\right)_{s_3, s_2} - \left(\gamma^{Ext[1]}.P_+\right)_{s_3, s_2}\right)\right\}\right\}$

Outline.

Summary.

- **Implementing new physics into a Monte Carlo tools can be a tedious task**.
- FeynRules **provides a platform to**:
  * **Develop** new models.
  * Investigate their **phenomenology**.
  * **Validate** their implementation in commonly used tools.
- **Restrictions**:
  * **Lorentz and gauge invariance**.
  * **Locality**.
  * **Spins**.
- **Website: http://feynrules.phys.ucl.ac.be**
  * **Includes a large model database**.
  * **Add your favorite model!**