

# Towards a common format for analysis implementations with MADANALYSIS 5

**Fuks Benjamin**

**CERN - IPHC - U. Strasbourg**

**Coordinating a simplified models effort workshop**

**LPCC meeting @ CERN  
October 29 - 30, 2013**

# The status

- ◆ Simplified models cover many topologies
- ◆ Several groups are recasting the experimental results. For a given topology:
  - ❖ Parameter space scans including event generation
  - ❖ **Implementing (some of) the existing experimental analyses**
  - ❖ **Validating the implementations**
  - ❖ Applying the searches to different frameworks
- ◆ We need to iterate for each topology
  - ❖ The task is huge
  - ❖ We may want to share the effort
  - ❖ **Some of the analyses may have been implemented (and validated) by several groups**

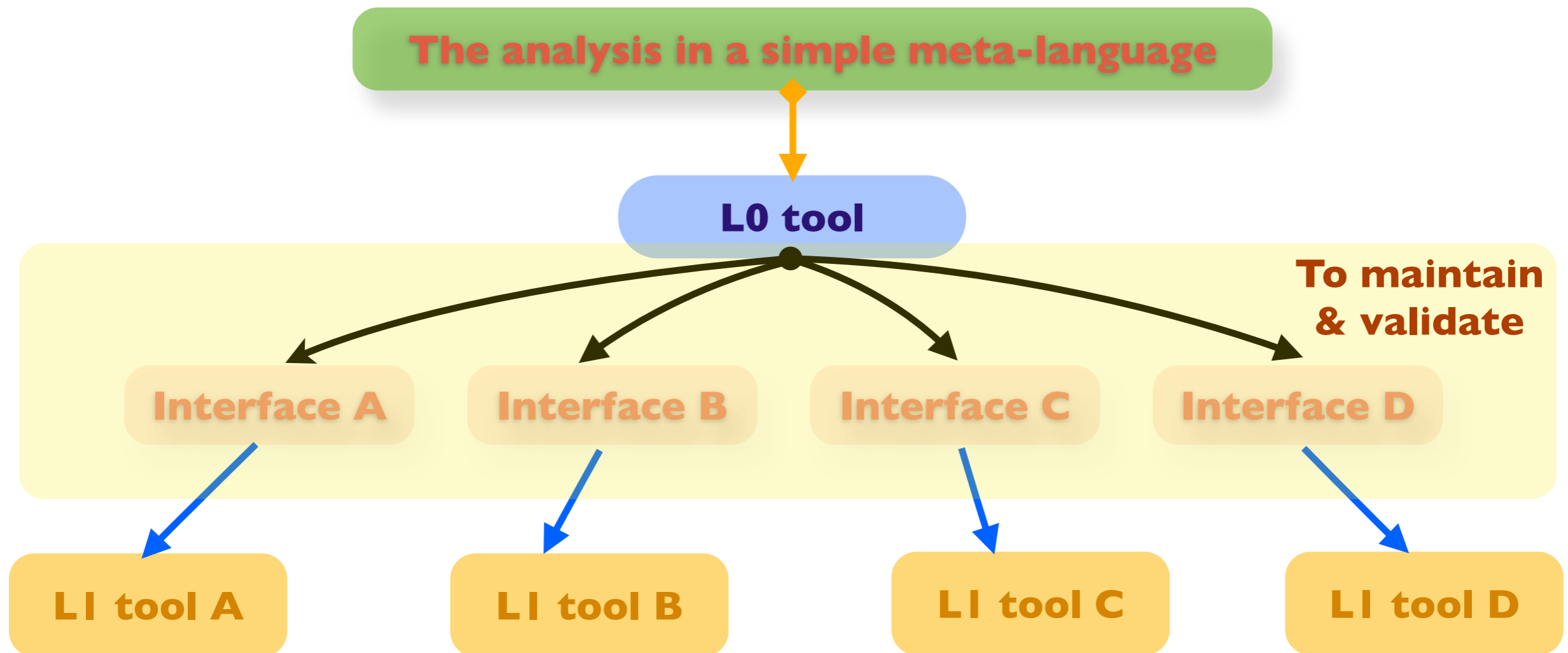
⇒ **Important redundancy of the work!**

# The drawbacks in short

- ◆ Can an analysis implemented and validated by a group A be used by a group B?
  - ❖ Each tool has its own programming language
  - ❖ Each tool has its own internal set of conventions
  - ❖ **The answer is no (at least not easily)**
  
- ◆ Moreover:
  - ❖ An implemented analysis is a routine in a given programming language
  - ❖ The code can be pretty long
  - ❖ **It cannot be understood easily by eyes (not human readable)**

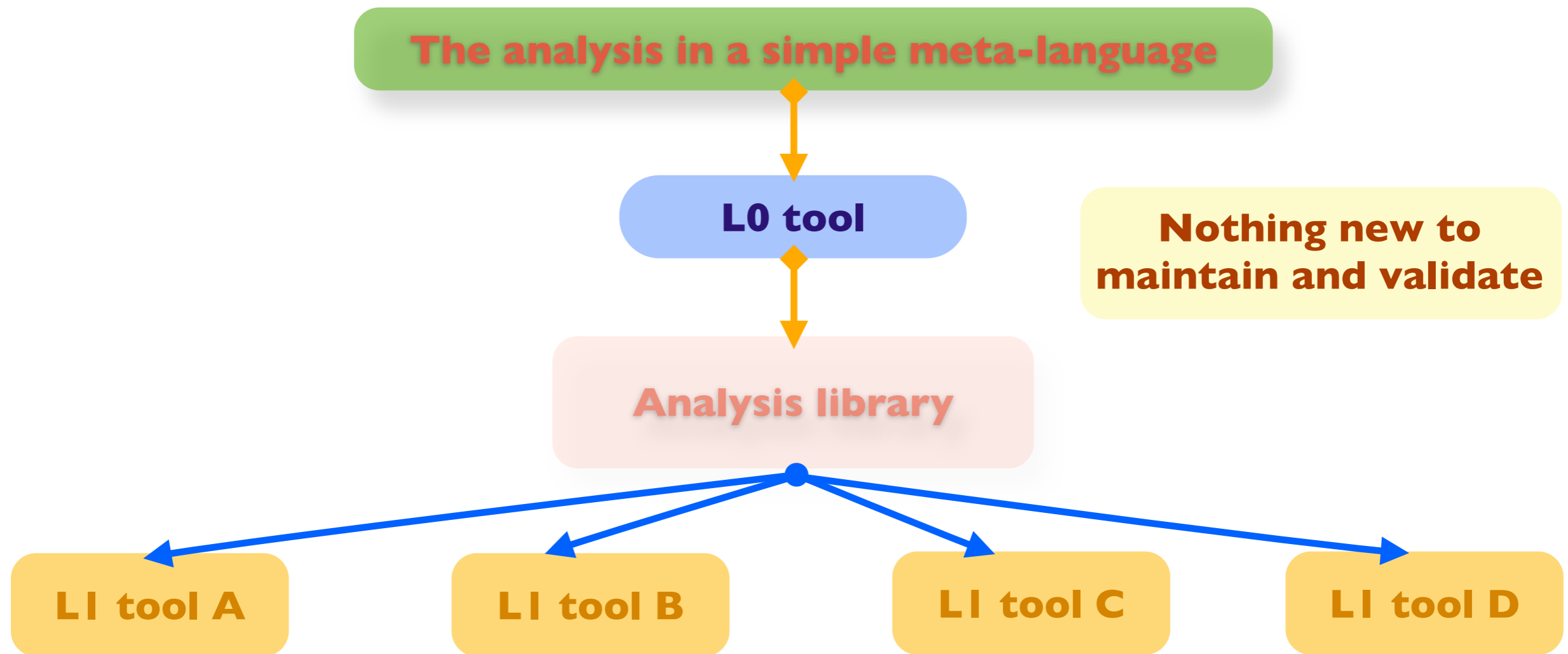
# Possible improvements (I)

- ◆ The needs (cf. my experience with automation and the FeynRules to MC generators connection):
  - ♣ A simple meta language to implement an analysis (tool at level-0)
  - ♣ A series of interfaces linking the tool at L0 to each tool at level-1
  - ♣ Automatic export of an analysis coded in the meta-language to any format of the LI tools



# Possible improvements (2)

- ◆ The needs (cf. my experience with the UFO for MC generator model formats):
  - ♣ A simple meta language to implement an analysis (tool at level-0)
  - ♣ A single interface exporting automatically the analysis as a library
  - ♣ We need to define the input/output, as well as the name of the functions



# Why MADANALYSIS 5 as a level 0 tool?

- ◆ It contains a PYTHON interpreter with a very simple language for analysis implementation

```
ma5>reject MET < 150
ma5>select THT > 100
ma5>reject N(1) != 2
ma5>select 2 <= N(j) <= 3
```

```
ma5>set main.stacking_method = stack
ma5>set main.lumi = 20
ma5>plot NAPID [logY]
ma5>plot MET 50 0 500 [logY]
ma5>plot THT 50 0 500 [logY]
```

```
ma5>plot DELTAR(1[1],1[2]) 15 0 15 [logY]
ma5>plot MT_MET(1[1]) 50 0 500 [logY]
ma5>plot MT_MET(j[2]) 50 0 500 [logY]
```

```
ma5>plot PT(1[1]) 50 0 500 [logY]
ma5>plot MT(j[1]) 50 0 500 [logY]
```

- ◆ The results are stored in the so-called SAF format

- ❖ Text-based
- ❖ Easily further processable

- ◆ Current developments

- ❖ Being able to export an analysis as a shared library
- ❖ A SAF reader to automatically generate histograms and cut-flow charts (to be included as a shared library)

# Triggering the discussions

- ◆ Does this proposal sound reasonable?
- ◆ What is missing (that could be useful)?
- ◆ What should be done (if one goes for the shared library)
  - ♣ The list of the needed functions
  - ♣ The arguments they take
  - ♣ MADANALYSIS 5 has its own internal data-format that could be used as a starting point (see, e.g., the technical discussions)
  - ♣ *etc...*
- ◆ No shared library: what other option would be reasonable (beware of the time/manpower constraint)?