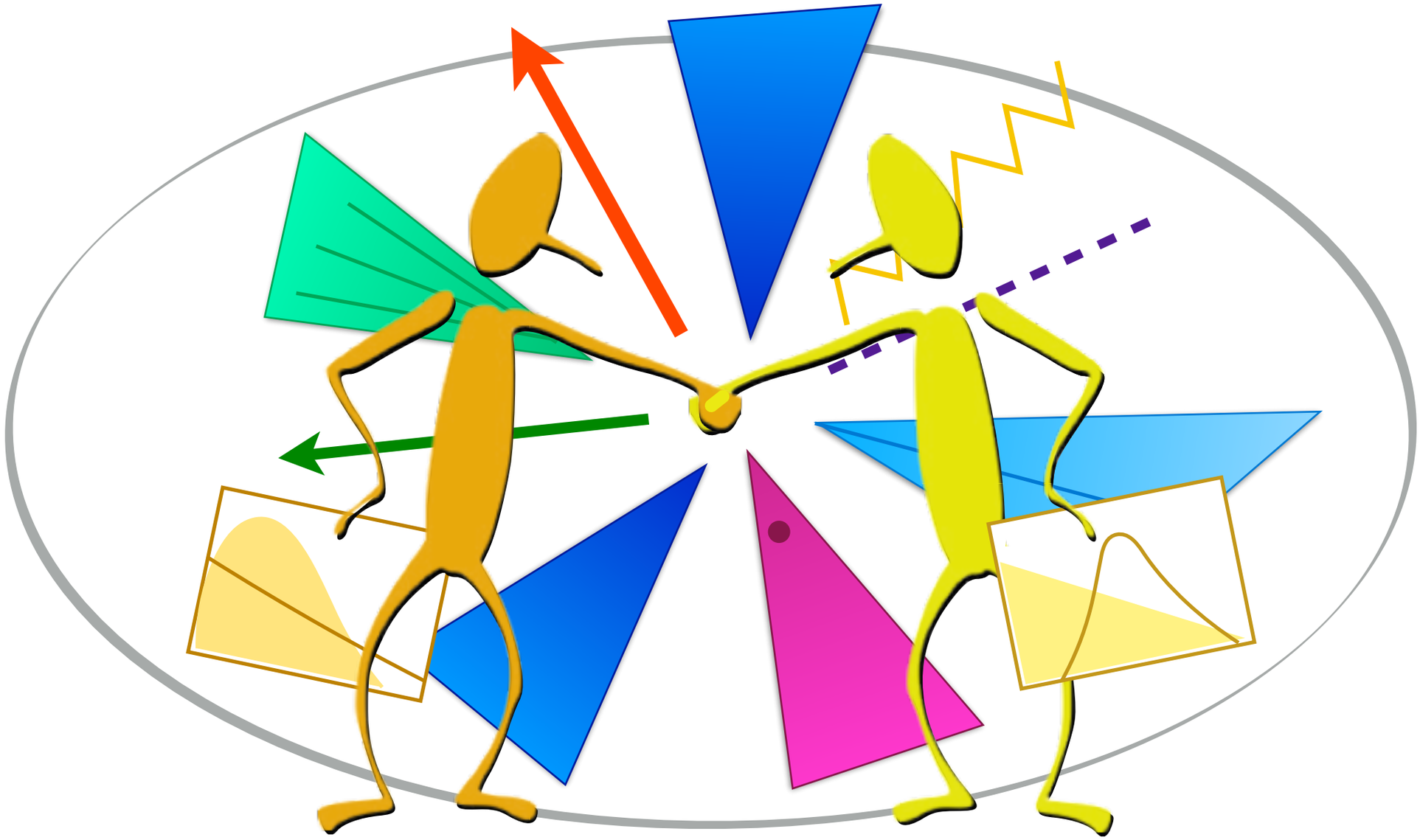
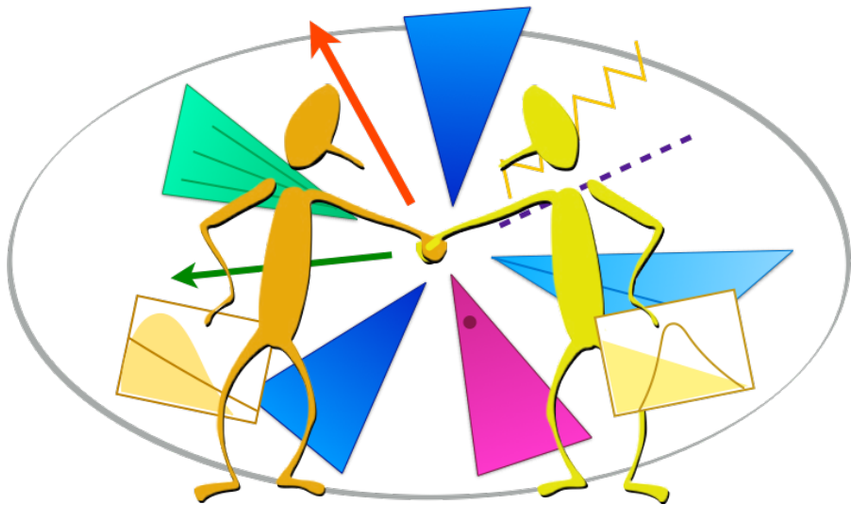


Welcome / Work plans

Sezen Sekmen (KNU)

LHADA Working Meeting
CERN, 16-18 November 2016



Welcome to the LHADA working meeting!

- Brief introduction / reminder to LHADA (for potential newcomers)
- Goals of this working meeting
- Program and work plan

An LHC analysis description accord

We set out to design an analysis description accord for the LHC,

- which is an LHC standard capable of describing the contents of an analysis in an unambiguous way.
  - analysis details include all object and event selections, as well as quantities such as efficiencies, analytic and algorithmic observables, and advanced multivariate selections.
- which should be exploited by the whole LHC physicists to abstract, visualize, validate, combine, reproduce, interpret, and communicate the contents of LHC analyses.

Earlier similar efforts proved to be very successful and useful:

- Les Houches Event Accord
- SUSY Les Houches Accord

An LHC analysis description accord

We defined the principles of an analysis description accord in the Les Houches 2015 new physics WG report (arXiv:1605.02684)

Towards an analysis description accord for the LHC

D. Barducci, A. Buckley, G. Chalons, E. Conte, N. Desai, N. de Filippis, B. Fuks, P. Gras, S. Kraml, S. Kulkarni, U. Laa, M. Papucci, C. Pollard, H. B. Prosper, K. Sakurai, D. Schmeier, S. Sekmen, D. Sengupta, J. Sonneveld, J. Tattersall, G. Unel, W. Waltenberger, A. Weiler.

Abstract: We discuss the concept of an “analysis description accord" for LHC analyses, a format capable of describing the contents of an analysis in a standard and unambiguous way. We present the motivation for such an accord, the requirements upon it, and an initial discussion of the merits of several implementation approaches. With this, we hope to initiate a community-wide discussion that will yield, in due course, an actual accord.

Motivations / use cases for the accord

- Analysis preservation
- Analysis design
- Analysis review
- Interpretation studies and analysis reimplementation
- Easier comparison of analyses

Overall enhancement of LHC data usage, by experimentalists, theorists and the public.

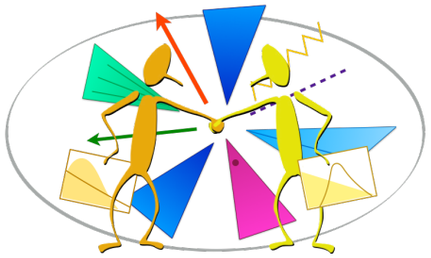
Properties of the accord

Basic requirements:

- Publicly available
- Complete
- Long lasting
- Correct and validatable

Desirable features:

- Human readable and writeable
- Self-contained
- Language-independent
- Framework-independent
- Supports combination of analyses



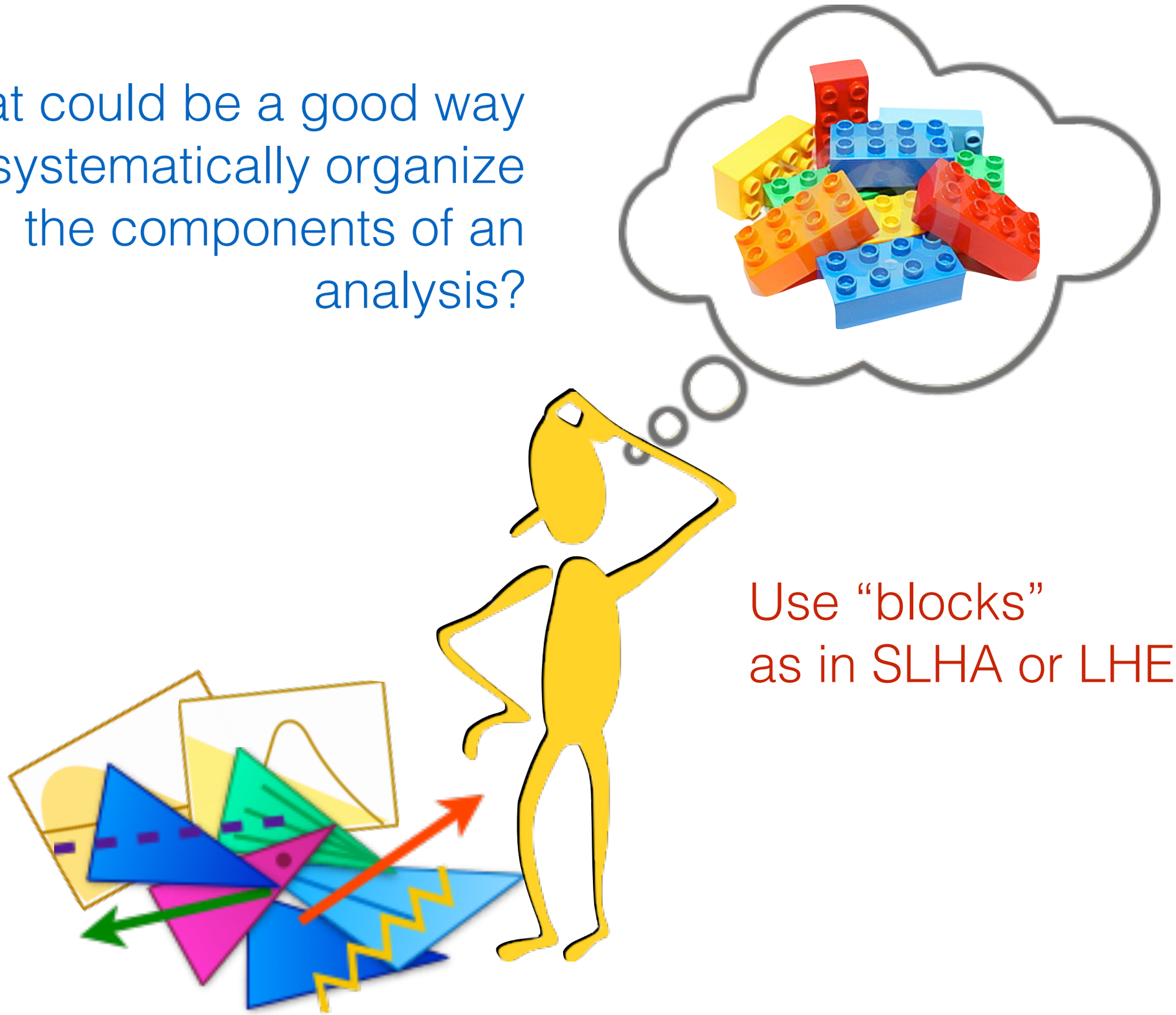
Different approaches to an accord

	Analysis description language	Pseudocode (or real standalone code)	Analysis framework code +metadata
Definition	New language tailored for analysis description. Requires parser.	Express full detail in pseudocode or actual executable code	Standard public analysis framework.
Pros	Easiness, control, human readability	Algorithmic completeness	Algorithmic completeness, unambiguous, validatable.
Cons	difficult to express full algorithmic detail	Pseudocode not runnable. Full code not universal and human readable.	Not universal and human readable.

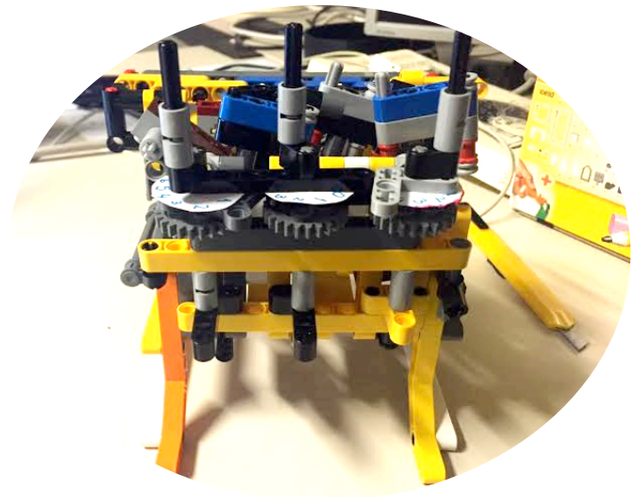
We devised a first proposal based on the analysis description language approach.

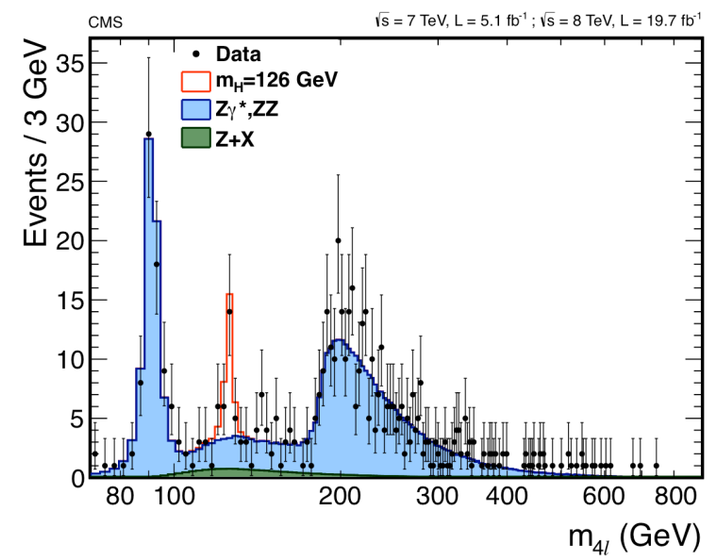
A proposal based on the “analysis description language” way:

What could be a good way to systematically organize the components of an analysis?



Use “blocks” as in SLHA or LHE.





A proposal for a Les Houches Analysis Description Accord

D. Barducci, G. Chalons, N. Desai, N. de Filippis, P. Gras, S. Kraml, S. Kulkarni, U. Laa, M. Papucci, H. B. Prosper, K. Sakurai, D. Schmeier, S. Sekmen, D. Sengupta, J. Sonneveld, J. Tattersall, G. Unel, W. Waltenberger, A. Weiler.

Abstract: We present the first draft of a proposal for “a Les Houches Analysis Description Accord" for LHC analyses, a formalism that is capable of describing the contents of an analysis in a standard and unambiguous way independent of any computing framework. This proposal serves as a starting point for discussions among LHC physicists towards an actual analysis description accord for use by the LHC community.

LH 2015 New Phys WG report (arXiv:1605.02684)

A proposal for a “LHADA”

This LHADA proposal aims to primarily fulfil

- human readability
- computer language independence.
- framework-independence, BUT…

This approach does not oppose analysis frameworks.
On the contrary, it aims to offer a standard analysis input to frameworks (just like SLHA does to SUSY calculators).
It gives us the freedom to develop or use whatever framework we like!

It consists of

- a plain text file describing the analysis, which uses a dedicated language with a strict set of syntax rules and a limited number of operators.
- library of self-contained functions encapsulating variables nontrivial to express (e.g. MVAs, advanced kinematic variables, …)

LHADA file syntax

An analysis description file is a plain text file that consists of easy-to-read blocks with a key value structure.

```
blocktype blockname
  # general comment
  key value
  key2 value2
  key3 value3 # comment about value3
```

- Allows clear separation of individual analysis components.
- Same key can appear multiple times.
- 5 types of blocks (for now): `info` `function` `object` `cut` `table`
- 18 keywords (for now):

```
apply  arg  bin  code  columns  cut  doc  entry  function
hepdata  info  object  reject  select  table  take  type  weight
```

- Can be automatically converted into any computing language format.

Where are we now?

- LHADA proposal was presented at the 1st [Re]Interpretation Forum workshop at CERN in June 2016
  https://indico.cern.ch/event/525142/timetable/
  and has met very positive reviews.
  —> we are encouraged to take the effort to the next step.
  —> since there was sufficient interest, LHADA became a subgroup of the Reinterpretation Forum:
  https://twiki.cern.ch/twiki/bin/view/LHCPhysics/LHADAparser
- CERN Analysis Preservation Support Group is informed about LHADA, and interested in discussing how LHADA can serve CERN analysis preservation plans (discussion Thursday morning).
- ATLAS and CMS are informed about LHADA (all new physics conveners, and CMS new physics groups). They are interested in watching the progress, and opening a discussion after the Moriond rush.

How to proceed?

"Action speaks louder than words."

Now we need to demonstrate to ourselves and to the LHC community that LHADA works; is actually useful, and makes analysis handling easier and more efficient.

Two threads of work to pursue:

- Write several LHC analyses in the LHADA format to test the capability of LHADA in describing analyses in full detail.
- Devise code to parse LHADA and interface it with various analysis frameworks.

Work in both fronts is already underway.

In this meeting…

We will hear about the progress between the June Reinterpretation Forum meeting and now.

…and will do practical work to build upon the current state.

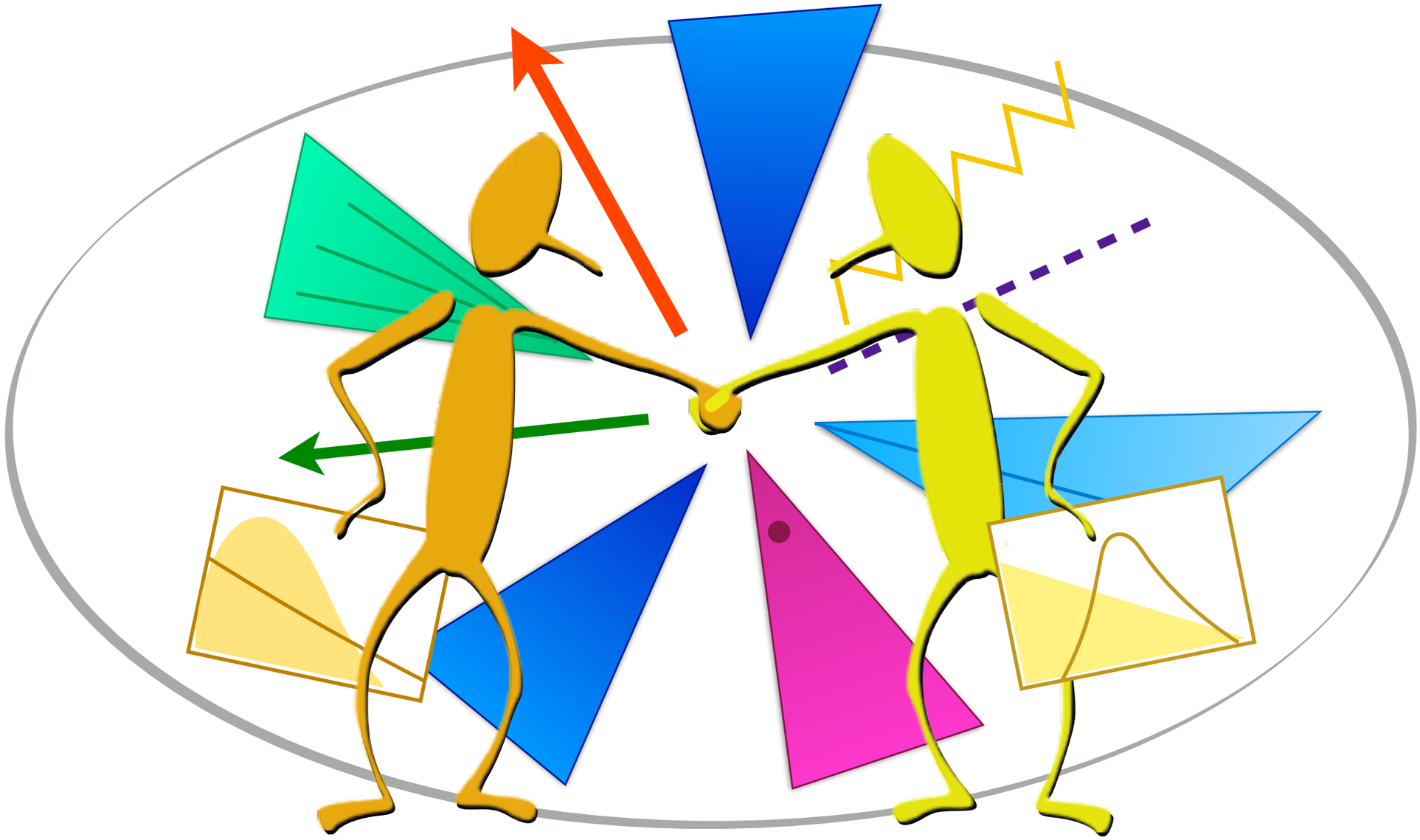
A loose structure for these 3 days:

- 16 Nov morning: analysis implementation plans
- 16 Nov afternoon: LHADA parsers
- 17 Nov morning: discussion with CERN Analysis preservation support group
- 17 Nov afternoon, 18 Nov morning: Work!

Many thanks to…

- *Les Houches PhysTeV 2015 organizers and participants*
- *LPSC Grenoble theory group for hosting the first LHADA meeting in February 2016*
- *LHC [Re]interpretation forum for welcoming LHADA discussions*
- *LPCC, especially Michelangelo, and Nanie for support with the organization of this working meeting.*
- *and you all for volunteering to work for creating a useful tool for the LHC.*

Let's make the most of this meeting to make progress!



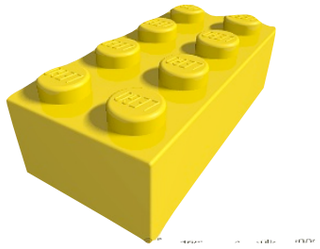
BACKUP: LHADA blocks description

Blocks: `info`

Provides general information about the analysis, e.g. publication information, benchmark scenarios, event generators used, validation material, etc.

(Minimally required or optional keys to be decided later.)

```
info analysis
  # Details about experiment
  experiment ATLAS
  id SUSY-2013-15
  publication JHEP11(2014)118
  sqrtS 8.0
  lumi 20.0
  arXiv 1407.0583
  hepdata http://hepdata.cedar.ac.uk/view/ins1304456
```



Blocks: `function`

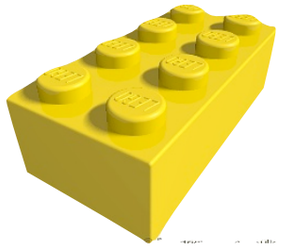
Defines all non-trivial operations that are calculated during the analysis.

- e.g.: advanced kinematic variables, e.g. transverse mass or variables created using machine learning methods, and filtering algorithms, e.g. lepton isolation definitions.

```
function function_name
  arg name1
  arg name2
  code link-to-code-repository
  doc link-to-documentation
```

arg: arguments to be respectively called by their names

- Individual function codes can be automatically updated as languages evolve, but LHADA file itself stays constant and valid.
- Allows to create a library of reusable, sharable functions.



Blocks: function

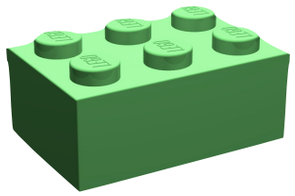
```
function mT2
  # stransverse mass
  arg vis1 # First visible 4-momentum vector
  arg vis2 # Second visible 4-momentum vector
  arg invis # Invisible transverse 4-momentum vector
  arg mass # Assumed mass of the invisible particle
  doc http://inspirehep.net/record/617472?ln=en # original publ.
  code http://goo.gl/xLyfN0 # code example from oxbridge package

function isol
  # Sums up activity in the vicinity of a given candidate
  arg cand # object whose isolation is to be computed
  arg src # "calo", "tracks", "eflow"
  arg dR # dR cone to be probed
  arg relIso # divide by candidate’s pt?
  code ...
  doc ...
```

Blocks: `object`

Defines all objects that are used in event selection.

- Objects can be defined at the truth or detector level depending on the purpose of the analysis.
- Some objects can be processed versions of other objects (electron —> isolated electron)
- Some objects can be provided from external sources.

```
object jets
  # clustered jets from the calorimeter cells
  take external
  apply antikt(dR=0.4, ptmin=20, etamax=2.5)
  code ...
  doc ...
```



Blocks: object

```
object mu
  # Muons
  take external
  apply detector_muons(workingpoint=combined)
  select pt > 10
  select |eta| < 1.5
  select isol(src=tracks, dR=0.4, reliso=true)<0.1
  doc ...

object ele_l
  # loose electrons
  take external
  apply detector_electrons(workingpoint=loose)
  select pt > 5
  select |eta| < 2.5
  select isol(src=tracks, dR=0.4, relIso=true)<0.1
  reject overlaps(neighs=mu, dR=0.4)
  doc 10.1140/epjc/s10052-014-2941-0 # doi to ATLAS ID def.
```

Blocks: `cut`

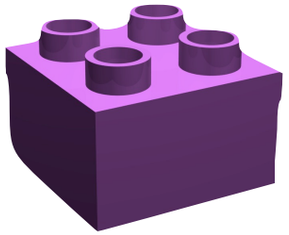
Defines event selection criteria that are applied to a given event in order to define analysis regions, e.g. preselection, signal, control.

- select and reject are boolean keys.
  Complex logical statements are possible.
- modular selection: a cut block itself can be considered as a boolean constraint in another block
- weight key enables event reweighting
- bin key enables binning of analysis regions

```
cut preselect
  # Pre-selection cuts
  weight triggerefficiency(leptonpt = lep[1].pt)
  reject lep.size > 1
  select lep[1].pt > 75
  select jets.size > 2
```

Blocks: **cut**

```
cut leadjets_1
  select jets[1].pt >= 60
  select jets[2].pt >= 40

cut SRBtoF
  select preselect
  select leadjets_2
  bin met.pt = 100,125,150,200

cut noZ
  # define a region outside the Z mass range
  select mll < 70 or mll > 100

cut razor
  # Define the ladder-like razor region
  select (MR>100 and R2>0.8) or (MR>300 and R2>0.5) or
(MR>500 and R2>500)
```



Blocks: `table`

Tabular collection of numerical information related to the analysis.

Examples: analysis results, cutflows, histograms, numerical efficiencies or other numerical functions, etc.

```
table results_events
  # Table for basic observed-signal and background events
  type events
  columns name obs bkg dbkg
  entry SRA 3452 3452 59
  entry SRBtoF[0] 1712 1720 161
  entry SRBtoF[1] 313 295 50
  entry SRBtoF[2] 201 235 34
```

Blocks: `table`

```
table result_corr
  # Correlation matrix for signal regions
  type corr
  columns name SRA SRBtoF[0] SRBtoF[1] SRBtoF[2] SRBtoF[3]
  entry SRA 1 0.2 0.1 0.15 0.14
  entry SRBtoF[0] 0.2 1 0.5 0.4 0.3
  entry SRBtoF[1] 0.1 0.5 1 0.3 0.2
  entry SRBtoF[2] 0.15 0.4 0.3 1 0.7
  entry SRBtoF[3] 0.14 0.2 0.2 0.7 1

table result_bkg
  # Breakdown of background in different signal regions
  type bkg
  columns name Z_jets Z_jets_err W_jets W_jets_err ...
  entry SRA 1726 254 1151 178 ...
  entry SRBtoF[0] 856 89 571 76 ...
  entry SRBtoF[1] 157 27 105 18 ...
  entry SRBtoF[2] 101 19 67 12 ...
  entry SRBtoF[3] 1009 156 674 56 ...
```