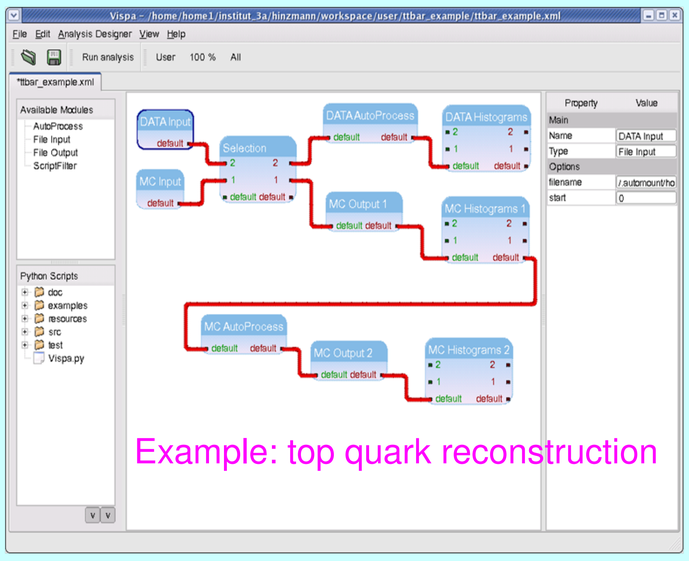
Visual Physics Analysis VISPA

Tatsiana Klimkovich for the **VISPA** group
(O.Actis, M.Brodski, M.Erdmann, R.Fischer, A.Hinzmann, T.Klimkovich, G.Müller, T.Münzer, M.Plum, J.Steggemann, T. Winchen)

Physics Institute 3a, RWTH Aachen University, 52056 Aachen, Germany
email contact: klimkovich@physik.rwth-aachen.de

GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung

Analysis Designer

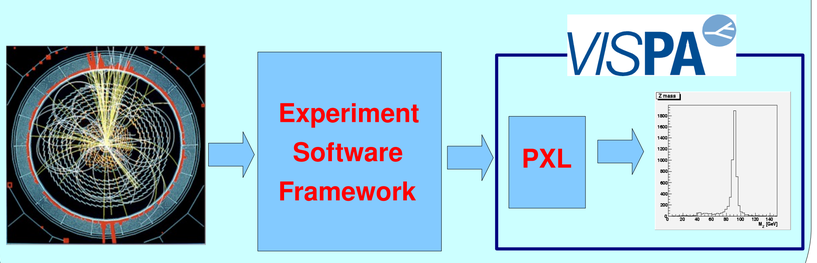


Example: top quark reconstruction

- Use GUI to design analysis
- Multi-path analysis flows
- Build analyses combined from C++ and Python modules
- Interactive creation of modules

The PXL Toolkit

- C++ toolkit for high level physics analysis [3]. Has been developed since 2006
- Version 2.1 (2009)
- It is the successor of the PAX (Physics Analysis Expert) toolkit, which was developed from 2002 to 2007
- PXL provides all necessary features for an experiment-independent high level physics analysis with emphasis on an easy user syntax



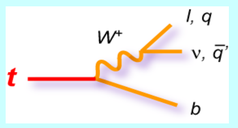
PXL Components

- **Event Container pxl::Event**

Particles (pxl::Particle)
Vertices (pxl::Vertex)
Collisions (pxl::Collision)
User data (pxl::UserRecord)
Their **relations** and **roles**

} Physics objects

} Event Interpretation pxl::EventView



- **pxl::Event** represents an entire physics event
- **pxl::Event** can hold several **pxl::EventViews**
- **pxl::EventView** is a special view of this event
- Copies of these classes preserve all contained information such as the relations between particles

- **User Record pxl::UserRecord**

All major PXL objects provide UserRecord for storage of arbitrary user data

- **Input/Output System pxl::Serializable**

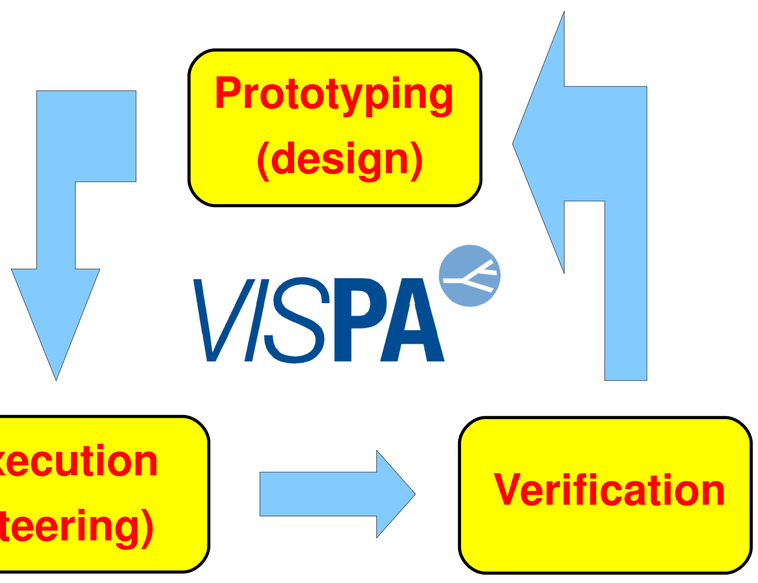
Fast, flexible, **small file size** (uses **ZLIB** library)

General Features of VISPA

- Aim: support design, execution and verification of HEP analysis [1,2]
- Multi-purpose window
- Visualization of analysis data and analysis flow in one Graphical User Interface
- PXL C++ toolkit as an underlying analysis software

Downloads and Literature

1. http://vispa.sourceforge.net/
2. O. Actis et al., Visual Physics Analysis (VISPA) - Concepts and First Applications, arXiv:0810.3609
3. http://pxl.sourceforge.net/
4. O. Actis et al., Automated Reconstruction of Particle Cascades in High Energy Physics Experiments, arXiv:0801.1302



Novel Concept of making physics analysis:

- Combination of graphical and textual programming
- Module steering
- For application in any HEP experiment

Python Interface

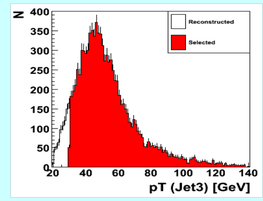
To enable the usage of all PXL objects and their methods within Python programs, a Python extension PyPXL is provided:

- Python code is easy to read
- Less code compared to C++
- Dynamic typing
- Automatic memory management

```
for particle in eventview.getParticles():
   if (particle.getName() == "Jet" and particle.getPt() > 30)
      selected.setObject(particle)
```

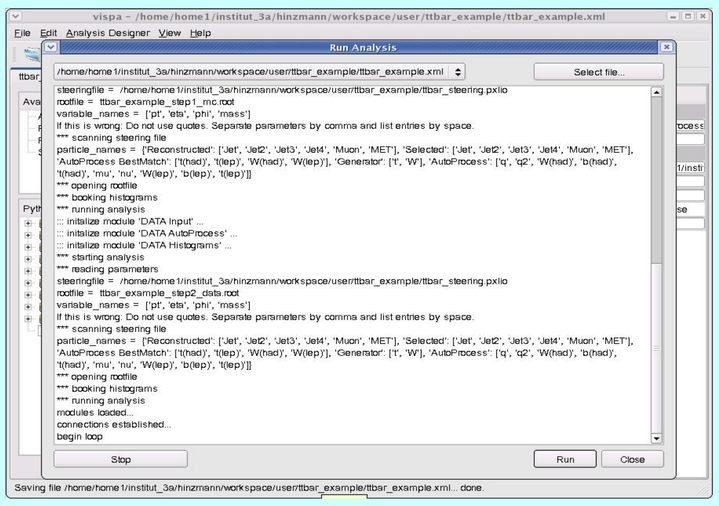
Use of SWIG for automatic interface of C++ to Python

Histogramming: PyROOT



Run Analysis

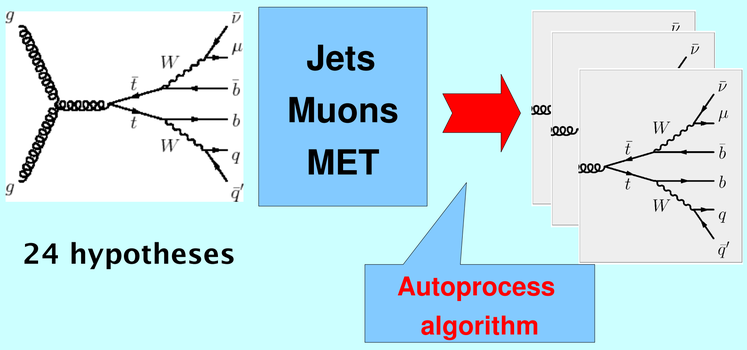
- Run analysis interactively:



- Or export the analysis as XML or Python steering and run it on the laptop, desktop or GRID

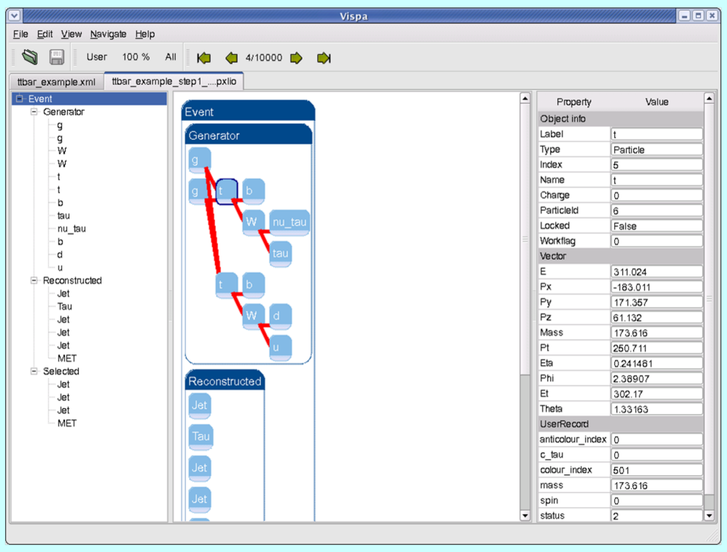
Autoprocess

Steering event container | Data event container | Output event container



- In various physics analyses (Top, Higgs, SUSY) a reconstruction of the whole decay chain is needed
- Several possible configurations need to be built
- Autoprocess is a module for automated reconstruction of particle cascades [4]

Event Browser



- Browsing physics data on an event-by-event basis
- Visualization of decay trees
- Inspecting properties of each object