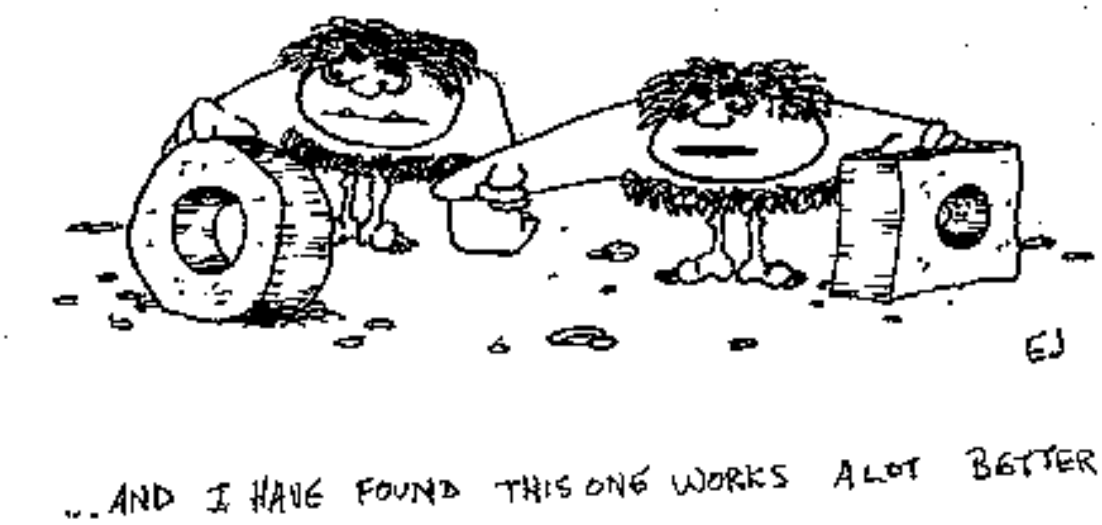
GammaLib

A versatile toolbox for high-level analysis of astronomical gamma-ray data

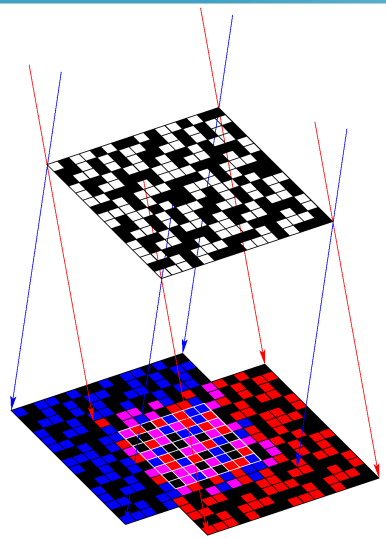
… or: stop reinventing the wheel



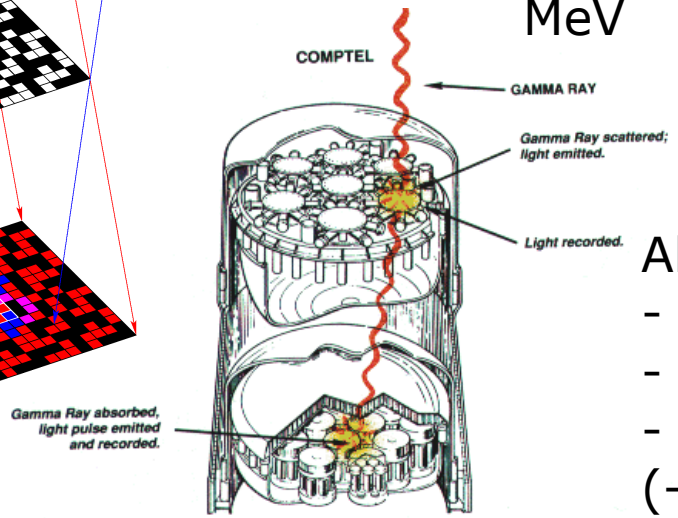
Jürgen Knödlseder (Toulouse, France)

irap
astrophysique & planétologie

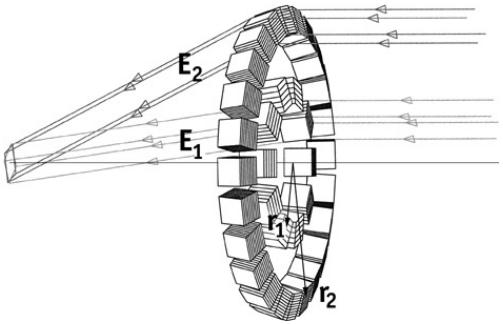
Observing gamma rays



keV

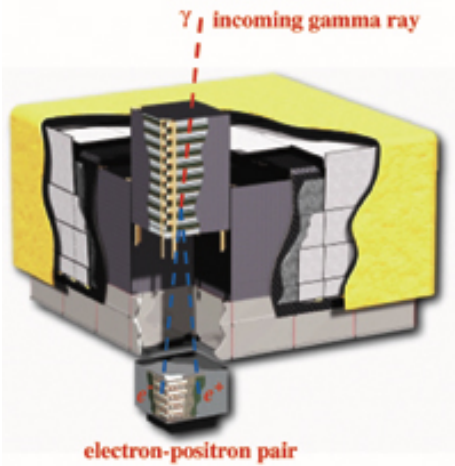


MeV



GeV

All measure photons !
- localization
- time
- energy
(- polarization)



TeV

Analysing gamma rays

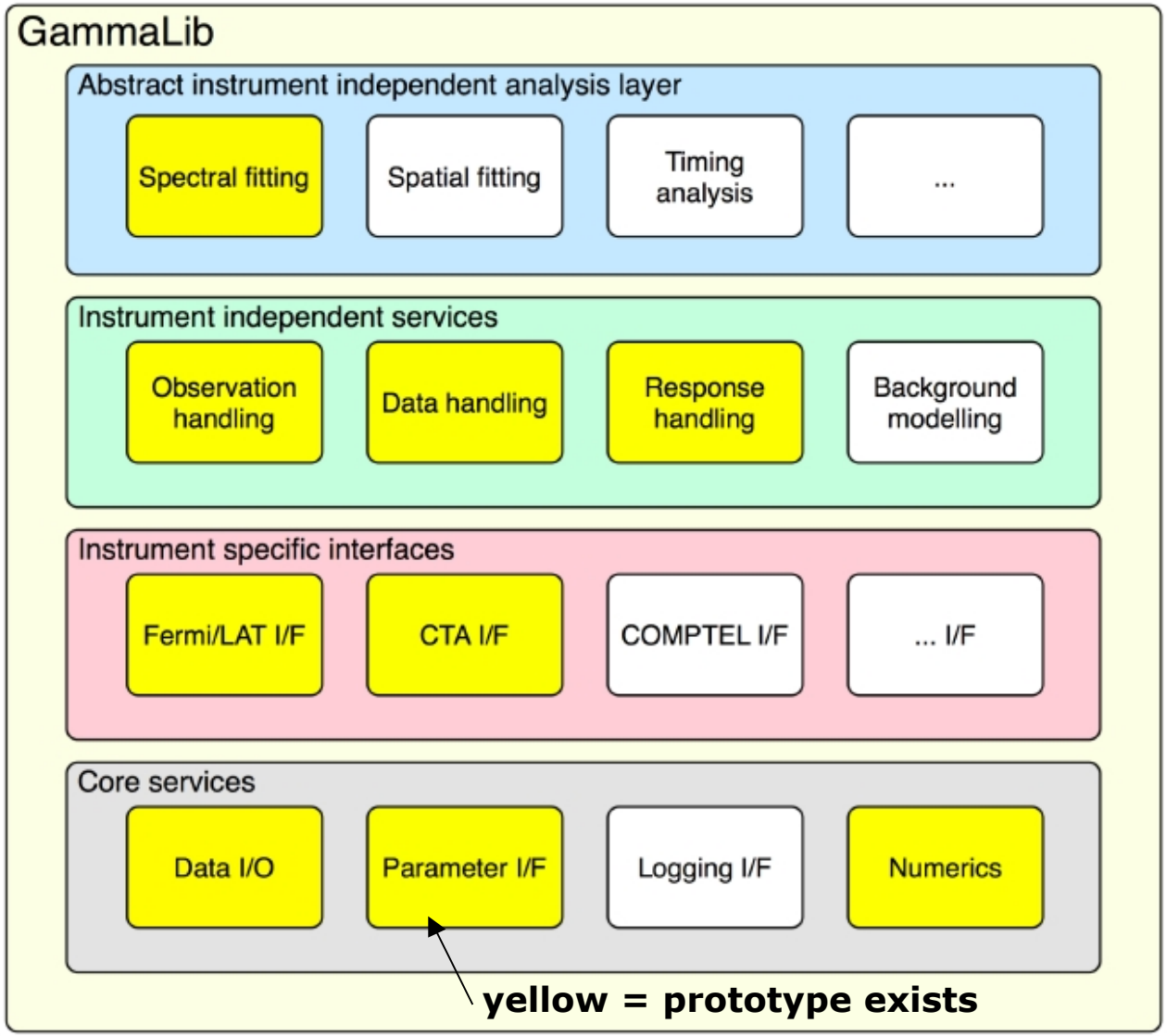
Common features

- Data are photon based (event lists or event histograms)
- Background is important or dominating (reduction techniques are crucial)
- Instruments are complex (time variable background and instrument properties, indirect imaging)
- Support guest observers

Common formats, methods and issues

- All community data are in FITS format (OGIP)
  http://heasarc.gsfc.nasa.gov/docs/heasarc/ofwg/ofwg_recomm.html
  (ftools, DS9, …)
- Most analysis systems break down analysis tasks in executables
  (bricks of science analysis)
  (ftools like executables; analysis pipeline by chaining executables)
- Many missions use the IRAF parameter interface
  (ftools, Chandra, INTEGRAL, SWIFT, Fermi, …)
- Most systems rely on a maximum likelihood optimizer (Poisson statistics)
  (INTEGRAL/SPI, Fermi, COMPTEL, …)
- Execution time and data volume **is** an issue …

The GammaLib concept



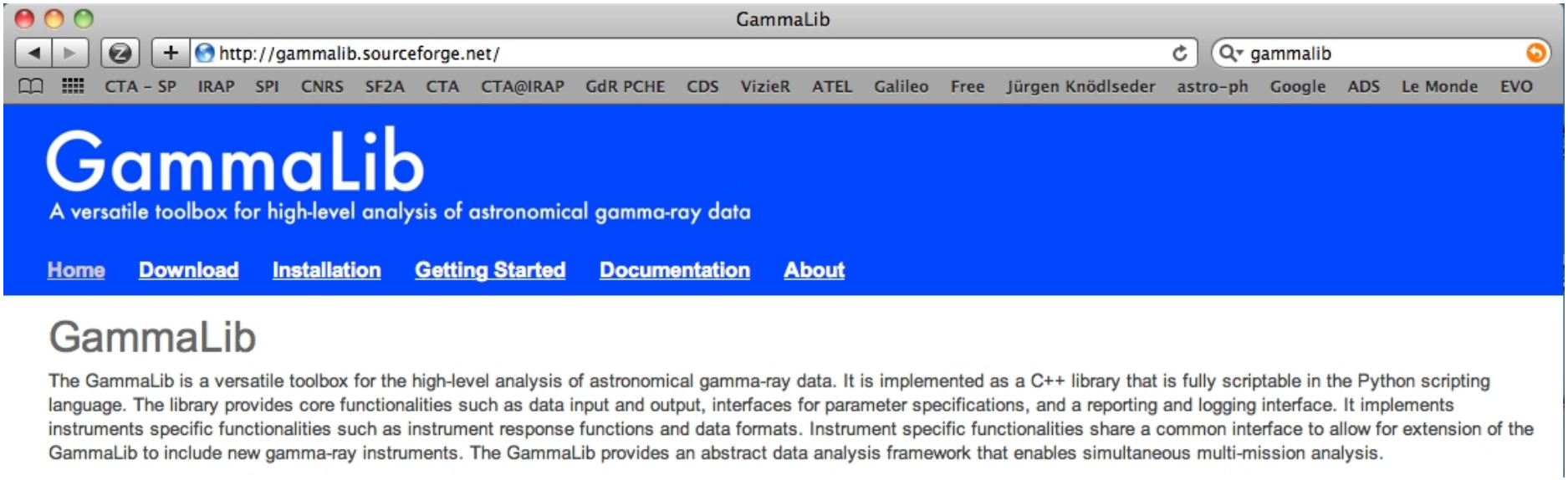
High-level analysis support (model fitting)

Generic analysis services (abstract combination of observations and events)

Instrument specific support (data content, response function, background handling)

Generic core services such as input/output in standard formats and numerical support (e.g. matrix handling/solving)

Summary



- **generic:** GammaLib core code does not depend on a specific instrument type or architecture
- **extandable:** new instrument interface implemented as derived classes
- **self-contained:** does not depend on any other library (considerable cost-reduction for maintenance; exception: cfitsio)
- **open source:** GNU licence (http://sourceforge.net/projects/gammalib/)
- **multi-platform C++:** should compile on any POSIX-compliant platform (32 and 64 bit)
- **API and Python scriptable** (SWIG based)