Go4 Analysis design

J.Adamczewski-Musch, H.G.Essel, S.Linev

Go4 Workshop 2010

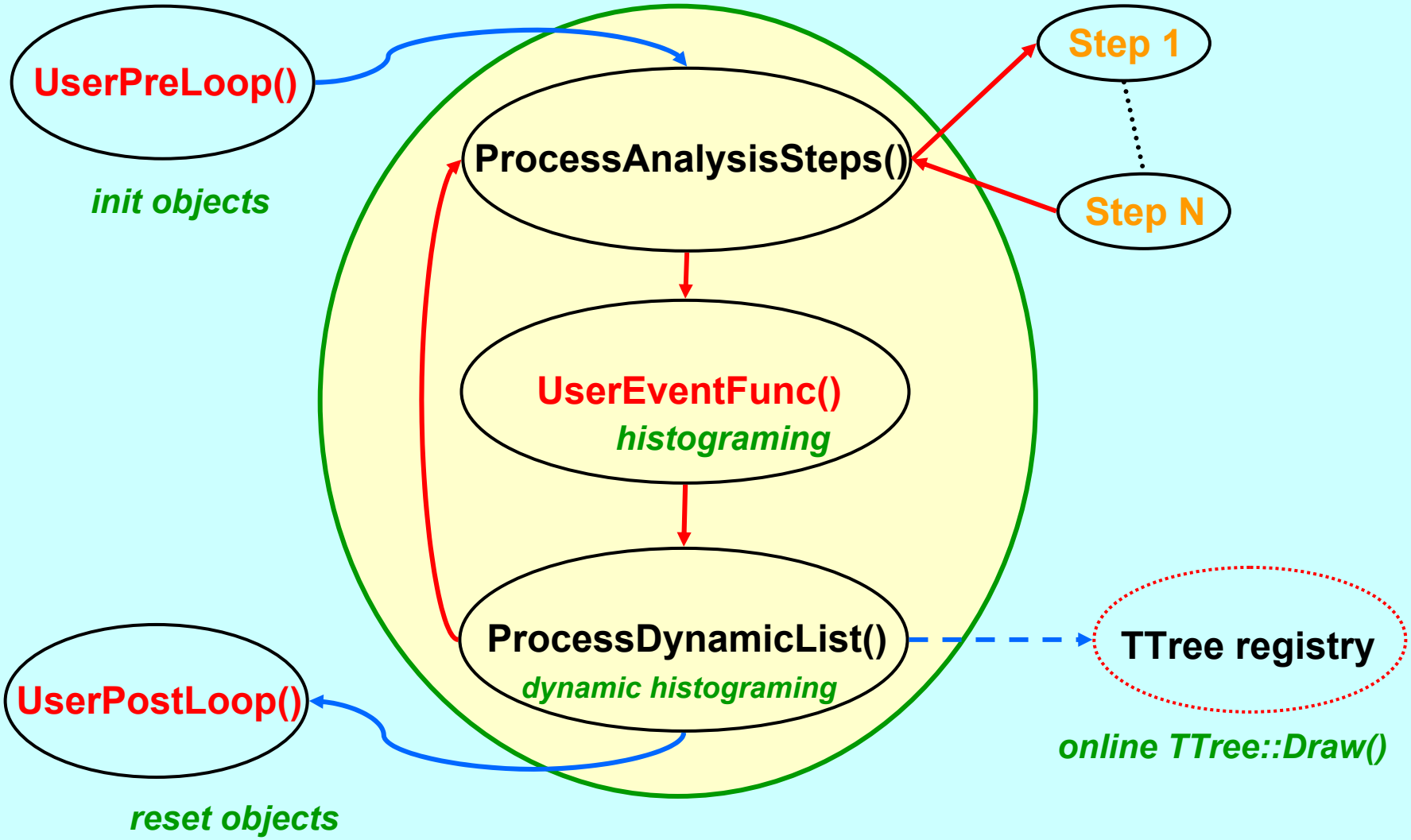
Analysis framework

- **TGo4Analysis baseclass:**
  - **Analysis setup (chain of analysis steps)**
  - **Object organization, dynamic histograms**
  - **Implicit / explicit event loop**
  - **Virtual methods to be defined in user analysis subclass**
- **TGo4AnalysisStep:**

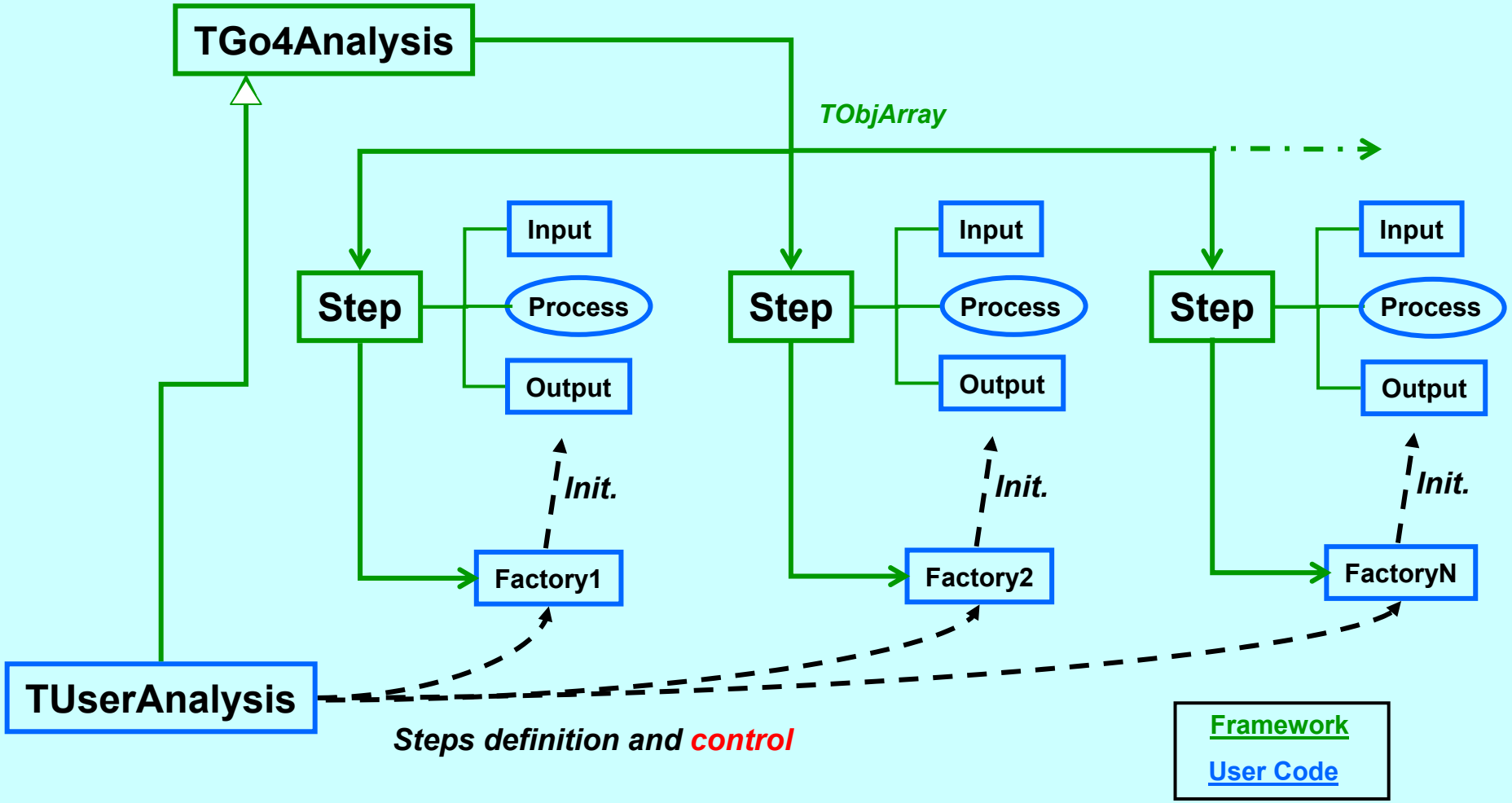
  defines one stage of the analysis, implements event classes

User may design own subclass of TGo4Analysis

Analysis event loop



Analysis steps

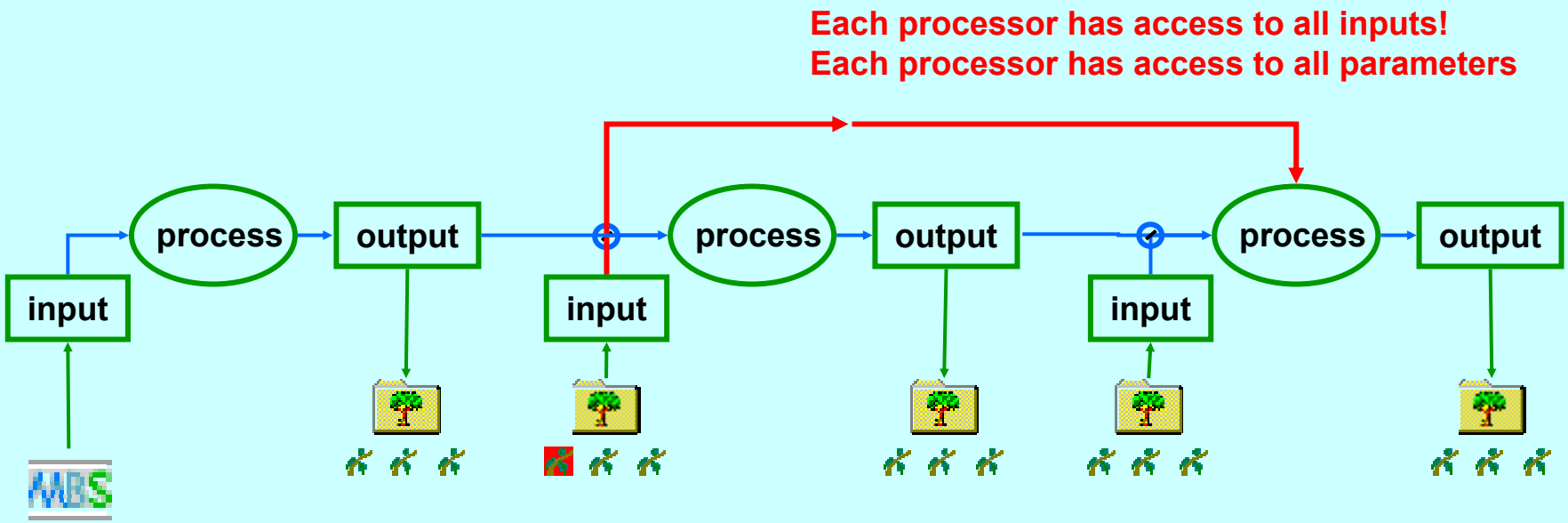


Analysis steps

Chain of analysis steps processed sequentially

Each step can be en/disabled (framework)

Input/output can be switched (framework)



Event Base Classes

- **TGo4EventElement:**
  Event structure (input and output event)
  User methods: *Clear(), Fill()* (optional)
- **TGo4EventSource:**
  Fills event structure with data;
  Framework! User event source optional
- **TGo4EventStore:**
  Stores event structure; Framework!
- **TGo4EventProcessor:**
  Converts input event into output event;
  User methods: *BuildEvent(TGo4EventElement* e)*
  any other methods optional
- **TGo4EventFactory:**
  Defines the user implementations of all the above at initialization
  Framework: TGo4StepFactory. User factory optional.

Implemented Services

- **GSI standard DAQ (Mbs)**

  TGo4MbsEvent, TGo4MbsSubEvent (format 10,1)

  TGo4MbsFile (read from *.lmd)

  TGo4MbsEventServer, TGo4MbsStream, TGo4MbsTransport
  (connect to Mbs)

  TGo4RevServ (connect to remote event server mrevserv)

- **Complex event structures „toolbox“:**

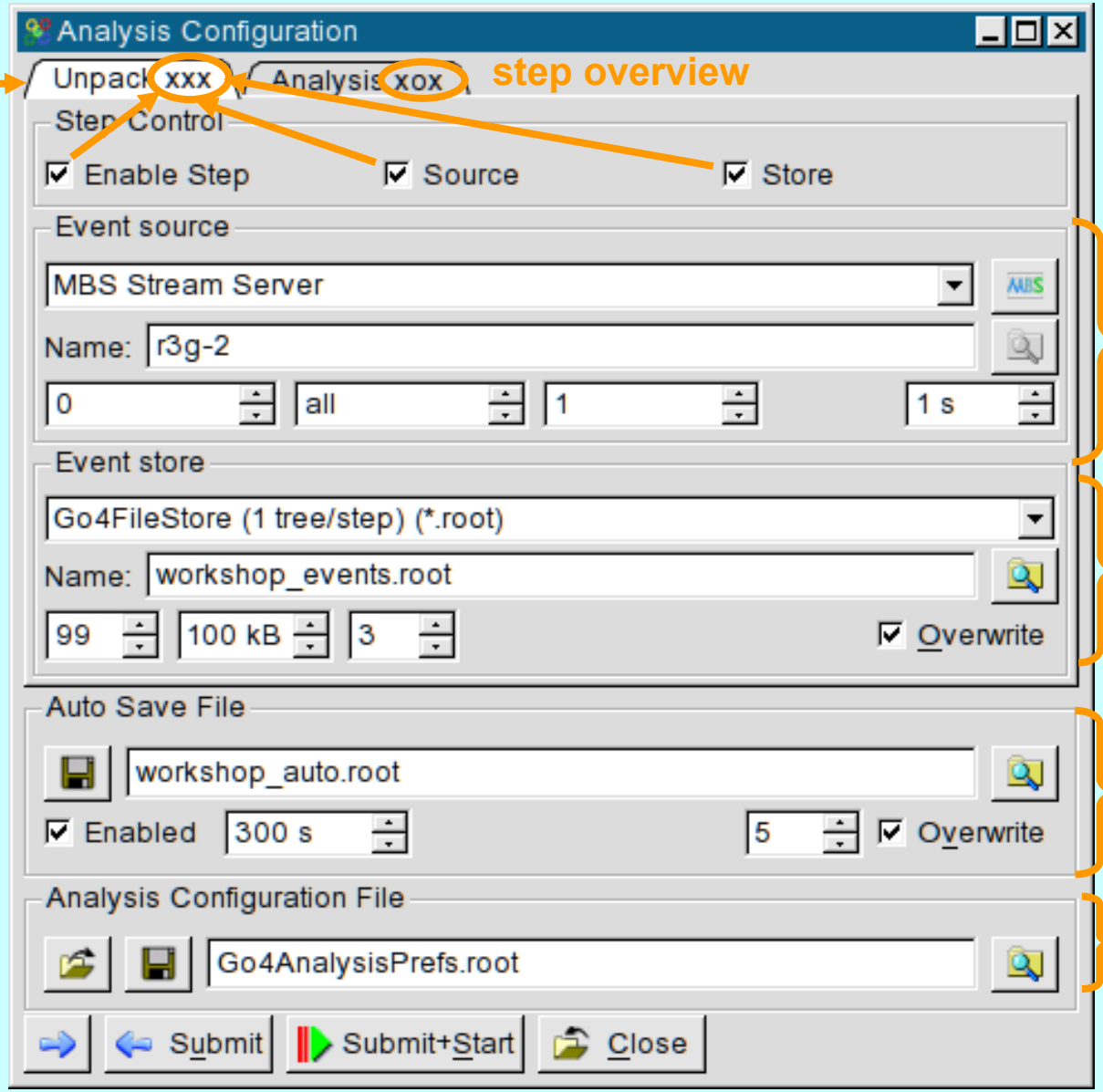
  TGo4CompositeEvent

- **Root I/O :**

  - **TGo4FileSource, TGo4FileStore :**
    wraps 1 TTree in 1 TFile

  - **TGo4BackStore :**
    1 TTree in memory only (<- online TTree::Draw())

Analysis Setup from GUI



Analysis Setup from command line

standard executable go4analysis

- **provided by framework**
  (no user compilation of MainUserAnalysis program required anymore!)
- **generic command line arguments for analysis set up**
- **optional arguments passed to user's TGo4Analysis subclass**

examples:

• go4analysis -stream r3g2 -step 0 -store workshop_events.root
-step 1 –store workshop_analyzed.root -asf workshop_auto.root
(setup as from gui: mbs stream server as data source, 1. and 2. step storage files, autosave file specified)

• go4analysis –file gauss.lmd –disable_store
–step Analysis -store analyzed_events.root –number 100000 -disable_asf
(process 100000 events from file gauss.lmd, no store of first step, storage file of step „Analysis", autosave disabled)

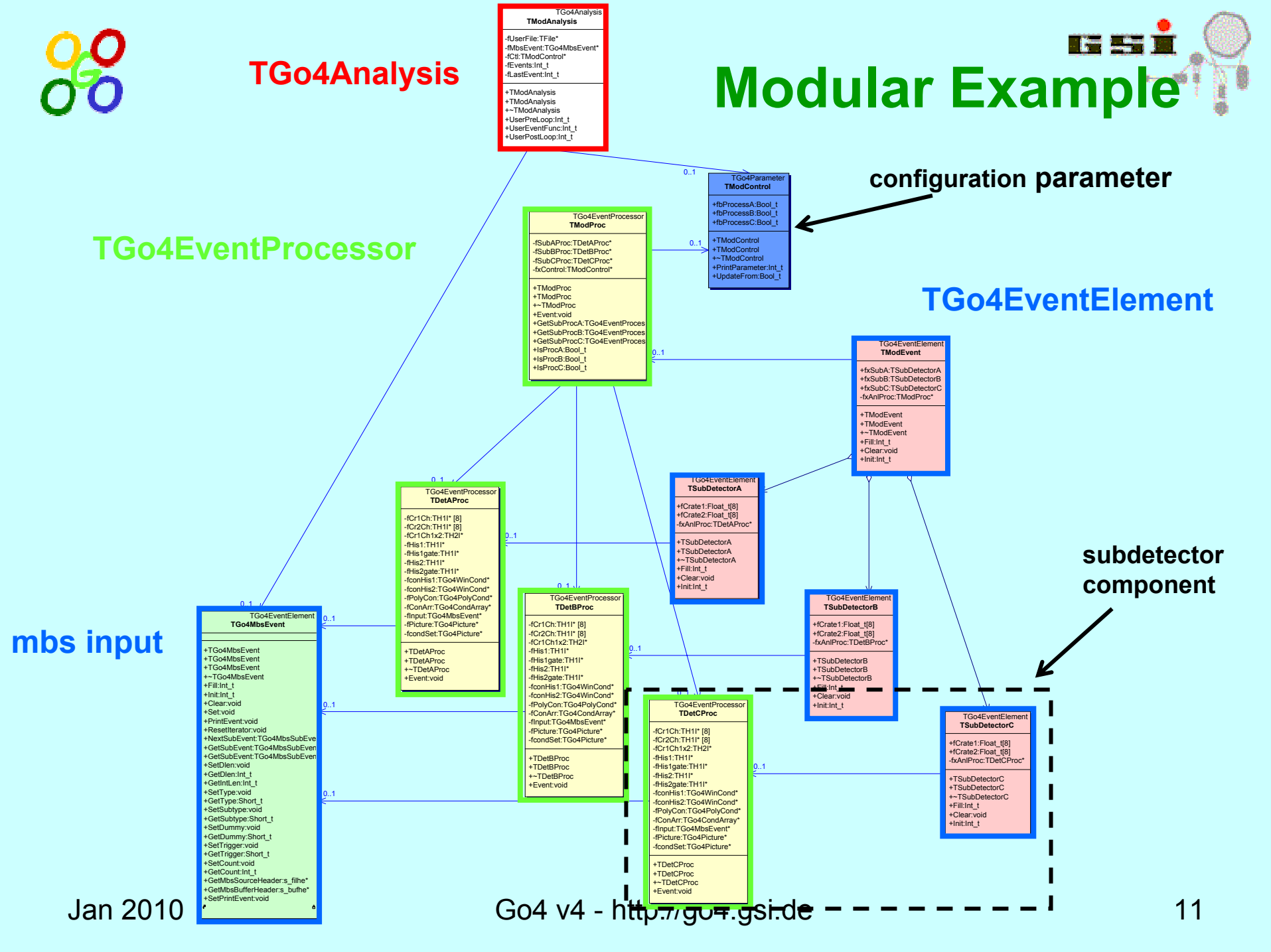
• go4analysis –user mydaq.gsi.de -server dataserver –norun –v 2 –log logfile.txt
(start analysis as data server, user defined DAQ event source, do not start processing, log output to file with verbosity 2)

• go4analysis –file gauss.lmd –args result.root 1 5000 7.0 42
(process file and pass user specific arguments to analysis constructor)

• go4analysis –help
(show all options)

Examples of Go4 analyses

- **Go4ExampleSimple:**
  - **1 analysis step, dummy output event**
  - **no user analysis class, framework TGo4StepFactory**
- **Go4Example1Step (-> ExampleModular):**
  - **1 analysis step with filled output event**
  - **user analysis class, framework TGo4Stepfactory**
- **Go4ExampleAdvanced (previous demo)**
  - **2 analysis steps, 2 output events**
  - **user analysis class, framework TGo4Stepfactory**
- **FRS framework analysis**
  - **4 standard steps + 1 user step, all with output events**
  - **user analysis class, user step factories, processor user baseclass**

TGo4Analysis

Modular Example