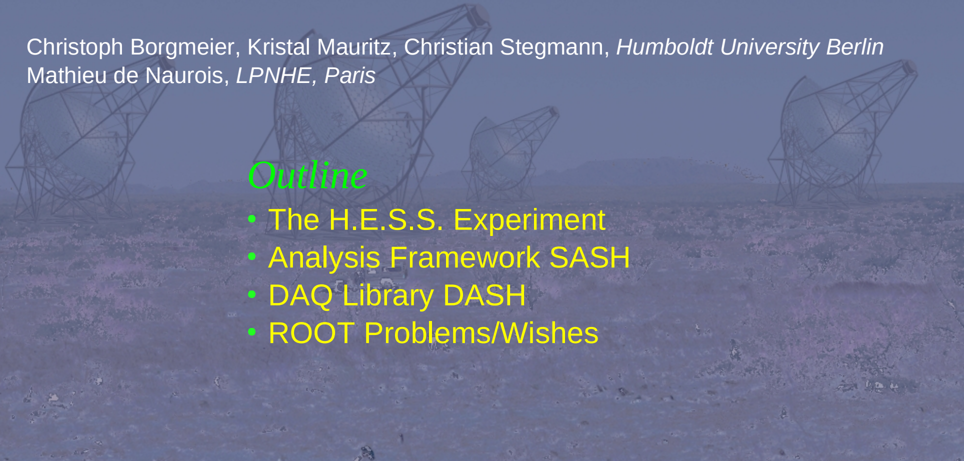
DAQ and Analysis at H.E.S.S.

ROOT Users Workshop 3
16 June 2001

Christoph Borgmeier, Kristal Mauritz, Christian Stegmann, *Humboldt University Berlin*
Mathieu de Naurois, *LPNHE, Paris*

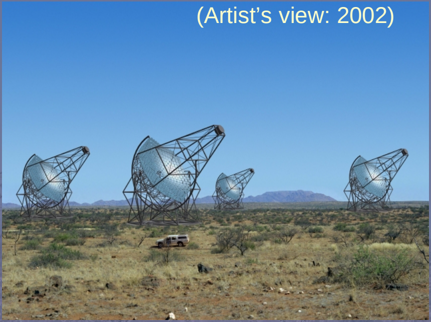


Outline

- The H.E.S.S. Experiment
- Analysis Framework SASH
- DAQ Library DASH
- ROOT Problems/Wishes

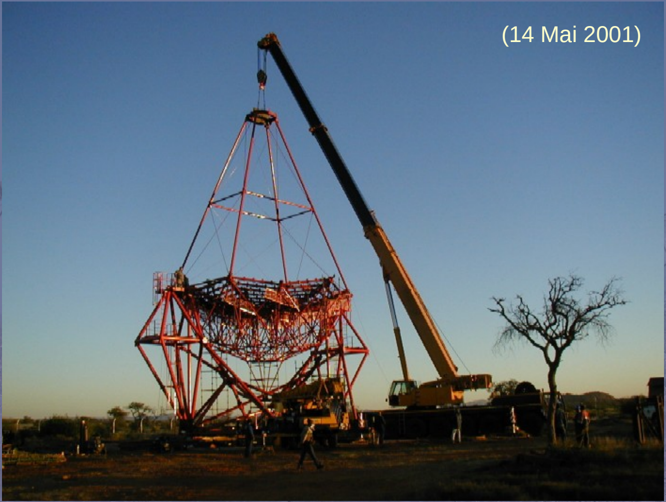
The High Energy Stereoscopic System (H.E.S.S.)

- Analyse gamma−rays above 100 GeV
  - Pulsars
  - Active Galactic Nuclei, etc
- Detect Cherenkov radiation of air showers, 3d reconstruction
- Construction in the Khomas Highland in Namibia
- First telescope: Fall 2001, Phase I (4 Telescopes): 2002


(Artist's view: 2002)

The H.E.S.S. collaboration:
18 Institutes in Europe
and Africa
`http://www-hfm.mpi-hd.mpg.de/HESS/HESS.html`

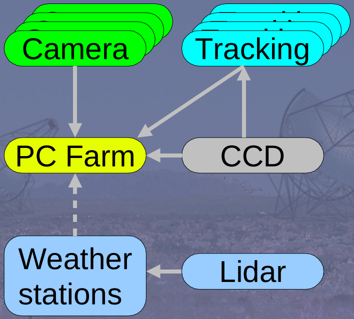
Current Construction



(14 Mai 2001)

Data Sources at H.E.S.S.

- Event data from PMTcamera
  - 1 kHz event rate (6 MB/s)
- Different kinds of monitoring data taken independently
  - CCDs
  - Cloud scanner etc.
  - Optical telescopes
  - Etc.



Camera

Tracking

PC Farm

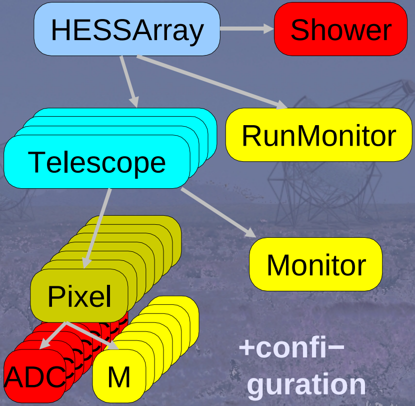
CCD

Weather stations

Lidar

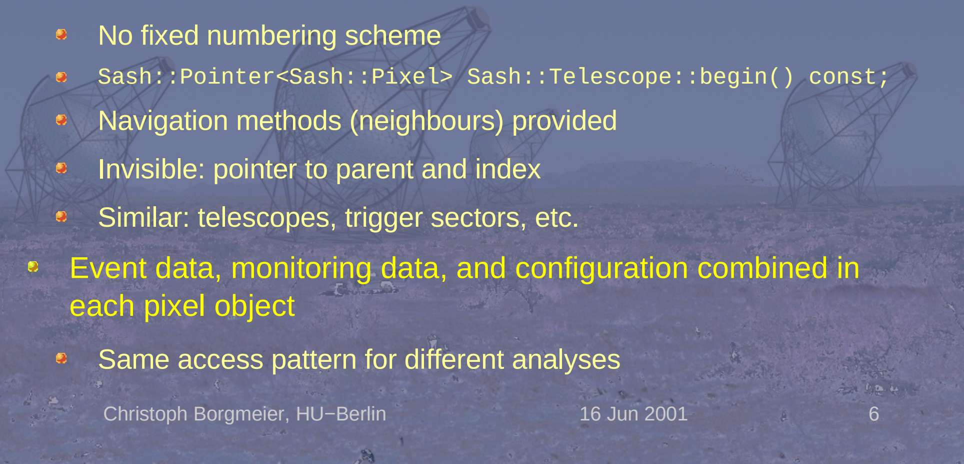
SASH (Storage and Analysis Software at H.E.S.S.)

- Relies completely on ROOT
- Representation of all data–producing parts connected by `Sash::HESSArray`
- Different ‘data rates’
- Transparent combination by analysis framework
  - Time stamp



+confi–guration

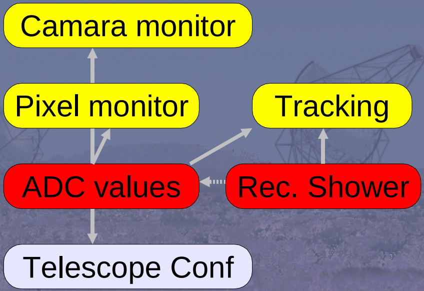
Containers in SASH



- Access to container data eg. Pixel only via iterator (`Sash::Pointer<Sash::Pixel>`)
  - No fixed numbering scheme
  - `Sash::Pointer<Sash::Pixel> Sash::Telescope::begin() const;`
  - Navigation methods (neighbours) provided
  - Invisible: pointer to parent and index
  - Similar: telescopes, trigger sectors, etc.
- Event data, monitoring data, and configuration combined in each pixel object
  - Same access pattern for different analyses

Sash::DataSet

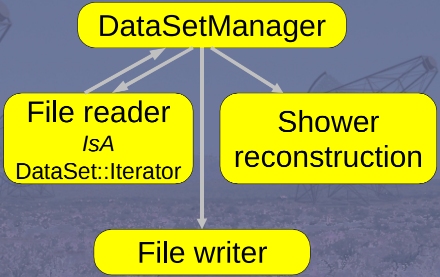
- Extension to ROOT TTree
  - Knows where to map into HESSArray
    - SASH does not need to know HESSArray details
  - Provide iterator
- Maintain connection to reachable (secondary) DataSets
- Iterator synchronization



DataSet examples

SASH Makers

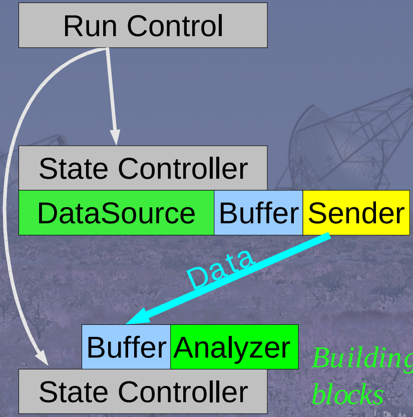
- Work on a Sash::HESSArray
- Common interface: `Process` function
  - Called subsequently by `Sash::DataSetManager`
- Callback feature for updates of secondary iterators



Maker examples

Data Acquisition Software at H.E.S.S. (DASH)

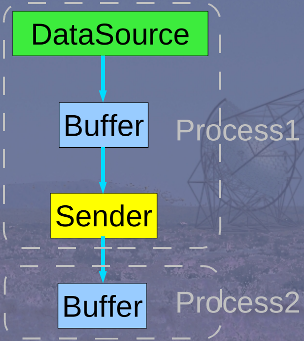
- Provides building blocks
  - Server
  - Communication
  - Configuration
  - Data processing
- CORBA (omniORB) for Inter-Process−Commumication
  - Built−in multi−threading facility



Building blocks

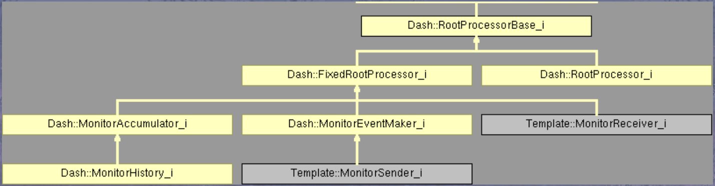
Data Block Processor

- Push architecture:
  - `TakeBlock` call between processes
- Arbitrary byte sequence data blocks
  - Streamed ROOT objects
    - Polymorphic
  - Other binary formats
    - Fixed C structs



Root Processor Hierarchy

- `RootProcessor_i`: handle general Root objects
  - New memory location for each object
- `FixedRootProcessor_i<T>`: template class
  - Accepts only non−polymorphic ROOT object
  - Constant memory location



Dash::RootProcessorBase_i

Dash::FixedRootProcessor_i

Dash::RootProcessor_i

Dash::MonitorAccumulator_i

Dash::MonitorEventMaker_i

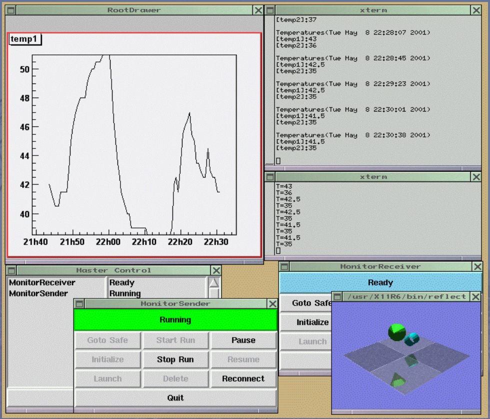
Template::MonitorReceiver_i

Dash::MonitorHistory_i

Template::MonitorSender_i

Monitor Event Class

- Dynamic monitor class
  - Holds map of numbers and ROOT objects
  - Qualified by name
- Example:
  CPU temperature
  - RootDrawer
  - RootPrinter
  - RootTreeWriter

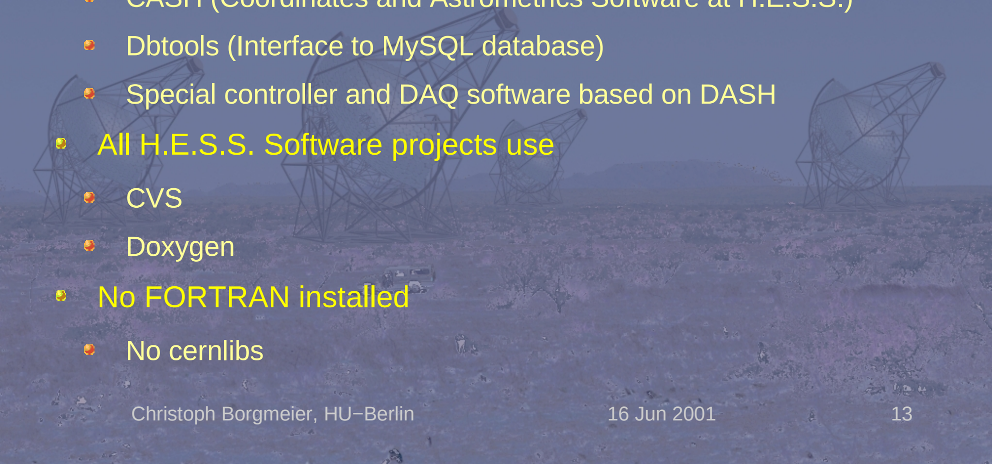


```
[temp2]:37

Temperatures(Tue May  8 22:28:07 2001)
[temp1]:43
[temp2]:36

Temperatures(Tue May  8 22:28:45 2001)
[temp1]:42.5
[temp2]:35

Temperatures(Tue May  8 22:29:23 2001)
[temp1]:42.5
[temp2]:35

Temperatures(Tue May  8 22:30:01 2001)
[temp1]:41.5
[temp2]:35

Temperatures(Tue May  8 22:30:38 2001)
[temp1]:41.5
[temp2]:35
```

```
T=43
T=36
T=42.5
T=35
T=42.5
T=35
T=41.5
T=35
T=41.5
T=35
```

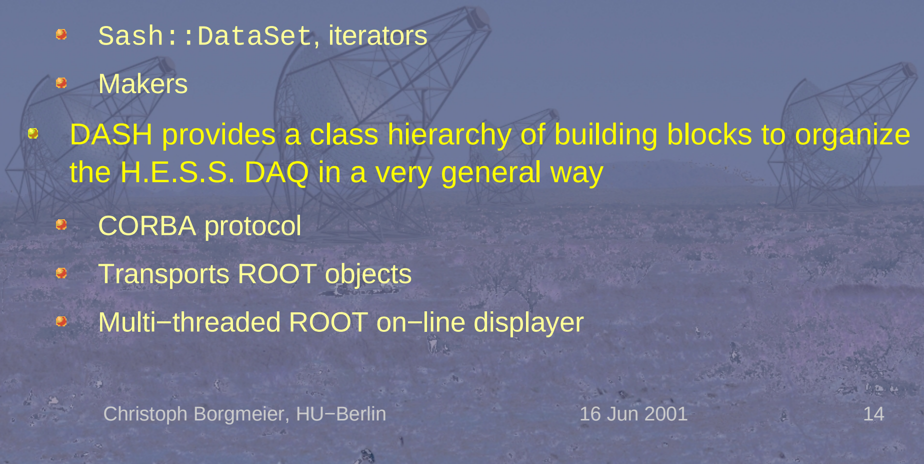
MonitorReceiver Ready
MonitorSender Running

Miscellaneous

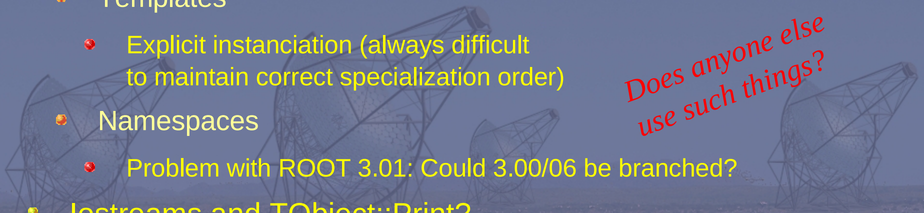


- H.E.S.S. Software consists of ~10 modules by ~8 developers
  - CASH (Coordinates and Astrometrics Software at H.E.S.S.)
  - Dbtools (Interface to MySQL database)
  - Special controller and DAQ software based on DASH
- All H.E.S.S. Software projects use
  - CVS
  - Doxygen
- No FORTRAN installed
  - No cernlibs

Summary



- SASH implements a general way to combine different ROOT trees
  - `Sash::DataSet`, iterators
  - Makers
- DASH provides a class hierarchy of building blocks to organize the H.E.S.S. DAQ in a very general way
  - CORBA protocol
  - Transports ROOT objects
  - Multi−threaded ROOT on−line displayer



ROOT Problems/Wishes

- Frequent code breaks with new ROOT versions
  - Templates
    - Explicit instanciation (always difficult to maintain correct specialization order)

Does anyone else use such things?

  - Namespaces
    - Problem with ROOT 3.01: Could 3.00/06 be branched?
- Iostreams and TObject::Print?
- ROOT web site search engine: e.g. `ref.txt` and dates

Experience with ROOT

- Impressive support from the ROOT team
  - Quick fixes
- Interesting new ROOT developments

Thank you to everyone who contributed.