Data acquisition and data quality control in a High Energy Physics experiment

Maria Lorenza Ferrer and the KLOE collaboration

Abstract—
The data acquisition system for the KLOE experiment, presently running at the DAΦNE collider in Frascati, requires a set of processes distributed in a complex network of computers and storage devices, which cooperate to sustain an acquisition throughput of 50 MBytes/sec and an event rate of 10 kHz. Methodologies used in data moving, processes control and data management are described. A short presentation of the physics program and of the KLOE detector is given for completeness.

Keywords— **DAQ, SNMP, ADSM, DB2, switched network, High Energy Physics experiment, high data rate.**

I. Introduction

The KLOE experiment is installed at the Frascati DAΦNE Φ-factory [1], a high luminosity double-ring e^+e^- collider designed to attain a peak luminosity of $5 \times 10^{32} cm^{-2} sec^{-1}$ with 120 bunches and an interbunch crossing time of 2.7 ns. DAΦNE was approved and funded by the Istituto Nazionale di Fisica Nucleare, INFN, in June 1990. The collider is optimized for operating at a total center of mass energy of 1020 MeV, the mass of the ϕ meson. The cross section for the process $e^+e^- \rightarrow \phi$ at the energies of the ϕ mass is approximately 5 μb which corresponds to a production rate of 2500 ϕ mesons per second for the nominal luminosity of DAΦNE . The e^+e^- collisions produce other final states and in particular the well known Bhabha scattering, $e^+e^- \rightarrow e^+e^-$, which due to the high rate of produced events, 20×10^3/sec at the nominal luminosity, is used for luminosity measurements. The ϕ-meson decay channels are listed in table I, that shows also branching ratios (BR),the maximum momentum of the produced particles and the number of candidates expected per year.

TABLE I
ϕ DECAYS.

Mode	BR %	P_{max}	#/year
		(MeV/c)	
K^+K^-	49	127	1.2×10^{10} pairs
K_SK_L	34	110	8.5×10^9 pairs
$\rho\pi$	13	182	3×10^9
$\pi^+\pi^-\pi^0$	2	462	5×10^8
$\eta\gamma$	1.3	362	2.5×10^8
$\eta'\gamma$	$\sim$ 0.01	60	2.5×10^6
$f_0\gamma, a_0\gamma$	$< 5 \times 10^{-3}$	39	$\sim 10^6$
other	< 1	-	10^8

As shown in the table DAΦNE is a good K^+K^- and K_SK_L factory, a reasonable η factory, most likely a good source of η' mesons. In addition DAΦNE has the unique possibility of producing pure K_S, K_L, K and $\bar{K}$ beams.

The KLOE experiment [2] intends to do a precision study of CP (charge conjugation x parity) violation in the decays $K_L, K_S \rightarrow \pi^0\pi^0, \pi^+\pi^-$. Defining the usual amplitude ratios and epsilon parameters:

$$\frac{\langle\pi^+\pi^-|K_L\rangle}{\langle\pi^+\pi^-|K_S\rangle} = \epsilon + \epsilon', \frac{\langle\pi^0\pi^0|K_L\rangle}{\langle\pi^0\pi^0|K_S\rangle} = \epsilon - 2\epsilon'$$

experimental observation of $\epsilon' \neq 0$ would be proof that CP is violated in the decay amplitude.

The traditional way to measure direct CP violation is through the double ratio

$$R = \frac{N_L^{\pm}/N_S^{\pm}}{N_L^0/N_S^0} \approx 1 + 6\ \Re(\epsilon'/\epsilon)$$

where each N refers to the numbers of $K_{L,S}$ decaying to two charged or neutral pions.

Many experiments have attempted to measure $\Re(\epsilon'/\epsilon)$ with continuously improving sensitivity. The results can be summarized as: $0 \leq \Re(\epsilon'/\epsilon) \leq 4 \times 10^{-3}$. The goal for KLOE is to measure $\Re(\epsilon'/\epsilon)$ with sensitivity of the order of 10^{-4}.

As an example, Table II compares the branching ratio of the neutral K_S and K_L mesons decaying into two charged and two and three neutral pions. Also the expected average decay paths are presented.

TABLE II
K_S AND K_L COMPARISON.

	K_S^0	K_L^0
BR($\pi^+\pi^-$)	68.6 %	2×10^{-3} %
BR($\pi^0\pi^0$)	31.4 %	9×10^{-4} %
BR($\pi^0\pi^0\pi^0$)	$< 1.4 \times 10^{-5}$ %	21 %
decay path (cm)	0.59	344

Neutral pions decay into 2 photons (γ) with a probability of 99 % in a very short decay path (25 nm), while for instance the process $\pi^{\pm} \rightarrow \mu^{\pm}\nu_{\mu}$ has almost a 100 % probability with a very long decay path of 7.8 m.

Given the very different decay paths of K_S^0 and K_L^0, see Table II, the typical $\phi \rightarrow K_S^0K_L^0$ event shows a K_S^0 decaying, near to the e^+ e^- colliding region, into two charged or neutral pions. Neutral pions decay immediately into 2 γ, while $\pi^{\pm}$ survive for a long path without decaying into other particles (Fig.8 in Section VII is a typical example).

M.L. Ferrer is with the Laboratori Nazionali di Frascati, Istituto Nazionale di Fisica Nucleare, Frascati, Italy. E-mail: Maria.Lorenza.Ferrer@lnf.infn.it .

The KLOE collaboration is in Appendix

Other relevant consequences of Tables I and II are:

- the statistical errors in measuring $\Re(\epsilon'/\epsilon)$ depends essentially on the number of $K_L^0 \rightarrow \pi\pi$ events,
- the sensitivity of 1×10^{-4} requires 2 years of running time at the peak luminosity of DAΦNE,
- the systematic errors strongly depend on the efficiency of the photon detection.

Other interesting channels to be studied in KLOE are the Kaon form factors, K_S rare decays, K_S semileptonic asymmetry never measured so far and radiative ϕ decays.

II. The KLOE detector

The KLOE detector has a cylindrical structure surrounding the collider beam pipe, see Figure 1, and consists of

- a large highly efficient drift chamber for detecting the charged K^0 decay,
- an electromagnetic calorimeter (Barrel and End Cap) with excellent timing capability, to measure the energy and impact point of photons,
- a second electromagnetic calorimeter (QCAL) fitting in the narrow space between the drift chamber and the beam focusing quadrupoles to control systematic effects in background subtraction, a particularly dangerous one being $K_L \rightarrow \pi^0\pi^0\pi^0$.

The detector is immersed in the magnetic field of a superconducting coil of 2.5 m inner radius and 4.2 m length; the value of the magnetic field being 0.6 Tesla.

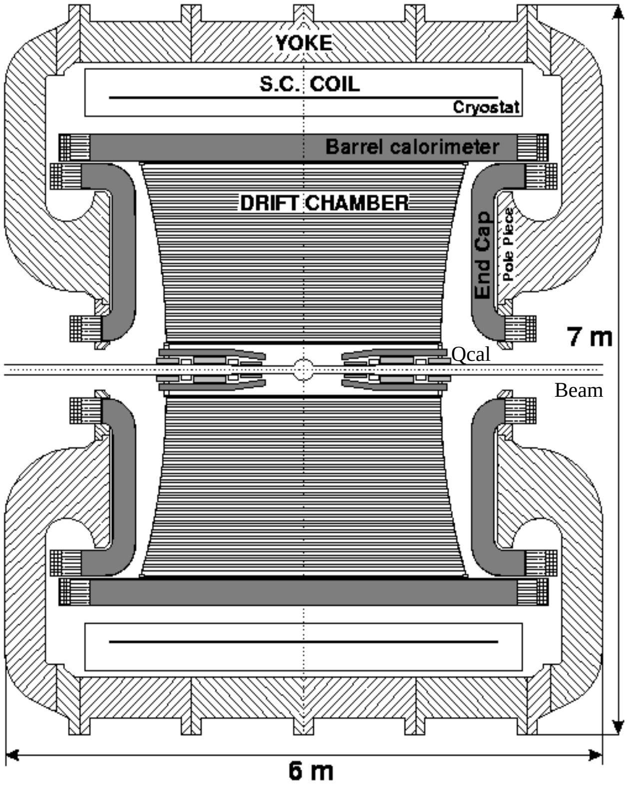


Fig. 1. Vertical cross section of the KLOE detector along the beam line

The basic structure of the electromagnetic calorimeter [3] is obtained by gluing lead plates of 0.5 mm, encapsulating 1 mm diameter scintillating fibers in rectangular or half disk modules with fibers running parallel to the beam or in parallel half circles. At both ends of any module, the light from the fibers belonging to trapezoidal adjacent elements, is collected by photomultipliers, a total of about 5000 for the whole calorimeter. Each photomultiplier signal is split three ways to provide the following functions:

- amplitude measurements via ADC, to provide energy deposit information,
- time measurement via TDC
- trigger generation after analog sum of signals

Both ADC's and TDC's have been developed for KLOE.

The main characteristics of the KLOE electromagnetic calorimeter are the following:

- good energy resolution $\sim 5\%/\sqrt{E(GeV)}$,
- almost full efficiency to detect γ^s in the range of momentum (20-300) MeV, giving negligible probability of a $K_L \rightarrow 6\gamma$ decay being confused with a $K_L \rightarrow 4\gamma$,
- excellent time resolution $\sim 70\ ps/\sqrt{E(GeV)}$,
- determination of the γ conversion point with $\sim 1\ cm$ accuracy, allowing to determine the vertex of $K_{L,S}$ with an accuracy of few millimeters,
- fast triggering response to suppress the 20 kHz Bhabha rate.

The drift chamber[4] provides tracking of charged particles in three dimensions with a resolution in the determination of the $K_{L,S}$ decay vertex of $200\ \mu m \times 1\ mm$ over the whole sensitive volume. It also provides a good momentum resolution ($\delta p/p \sim 0.5\%$) for low momentum tracks, and a fast trigger to complement the calorimeter based one. The drift chamber is a cylindrical structure of carbon fibers, providing 13000 drift cells, which is filled with a very light gas mixture, $90\% He - 10\% iC_4H_{10}$. The digitization of the drift time is performed by the chamber TDC's, based on a fully digital chip that was developed for KLOE.

The QCAL [5] is a tile calorimeter designed to offer a good photon detection (92 %) in the energy range (20 - 280) MeV with a time resolution of $240\ ps/\sqrt{E(GeV)}$. QCAL is readout by 32 photomultipliers.

The KLOE experiment startup was in 1999.

III. Data acquisition overview

The data acquisition, DAQ, of the KLOE detector was designed to sustain a rate of 10^4 events per seconds or equivalently an average readout time of 100 μs/event. Half of the acquisition rate is due to the ϕ decays expected at two times the maximum design luminosity of DAΦNE, the remainder is from down scaled Bhabha scattering and non vetoed cosmic rays. Bhabha events are fundamental to periodically calibrate the calorimeter scales for time and energy measurements and to determine the tracking chamber parameters. An average event size of 5 KBytes was also estimated, corresponding to a total bandwidth requirement of 50 MBytes/sec.

About 10^{11} events should be acquired, stored and analyzed during the KLOE lifetime. This requirement in data handling capability represents a challenge in the field of typical collider experiments.

Since the major aim of KLOE is to perform studies at sensitivities of the order of 10^{-4}, DAQ error rates, specially if dependent on event configuration, which would be quite acceptable in other situations, must be minimized. In particular the possible build-up of instantaneous backlogs that might result in losses of parts of an event has to be avoided.

A. Data acquisition architecture

The DAQ readout system involves some 23,000 channels of front end electronics, FEE, from calorimeters, tracking chamber and trigger system. For each event, relevant data coming from the whole FEE system have to be concentrated in a single CPU where a dedicated process is in charge of building the complete event. A three level data concentration scheme has been implemented.

The first two levels, that work on an event per event base, relie on custom buses and hardware controllers, in order to achieve high bandwidth without software penalties and CPU overheads.

Data transmission between CPUs, where in particular the 3^{rd} level of data concentration is implemented, relies on standard network media and protocols and works with packets of events in order to optimize the use of the network channels.

FEE and readout controllers use standard VME and custom boards. The VME protocol specification is used for FEE initialization and monitoring procedures while data readout uses a custom bus, AUXbus, built around the free pins in the VME P2 connector.

The DAQ components and the information flow are schematically presented in Figure 2. After the receiving a trigger signal, each KLOE FEE channel performs analog signal conditioning and digitisation during a fixed dead time of 2.2 μs after which the system is ready to receive a new trigger. To cope with this specification, every FEE board contains a local trigger counter and a buffer of appropriate length where each event number and the related data are memorized. A crate readout controller, ROCK[6], memorizes trigger counter in a local buffer and acquires the FEE event data sequentially and asynchronously with respect to the trigger signal. The number of boards per crate is optimized, according to the expected average occupancy, in order to assure a readout time much lower than 100 μs, as required by the expected rate of 10^4 events/s. A variable number of readout controllers are chained using a custom bus, Cbus[6]. The chain is terminated in a custom manager of readout controllers, ROCKM[6], that concentrates the data of each event in a local buffer. This buffer is read out by a VME CPU (a Compaq Alpha based processor). Event data coming from a chain will be called sub-event in the following.

The interface between the trigger system and the acquisition chains is concentrated in a single VME crate, the Trigger Supervisor. The busy and fault conditions are managed by the Trigger Supervisor which inhibits the trigger system at their occurrence. This is connected by mean of fan-in/fan-out units to the readout controllers. Periodic synchronization cycles are started by the Trigger Supervisor in order to check the alignment between trigger counters in all the acquisition boards.

The KLOE DAQ has been designed to sustain a throughput of 50 MBytes/s all along the 'DAQ path' from the readout controllers to the storage system. This path is implemented by means of a 22 ports FDDI switch that interconnects in particular the VME CPUs of 10 chains and an online farm of seven SMP 4-ways servers where the event building, formating, monitoring and disk storage processes are implemented.

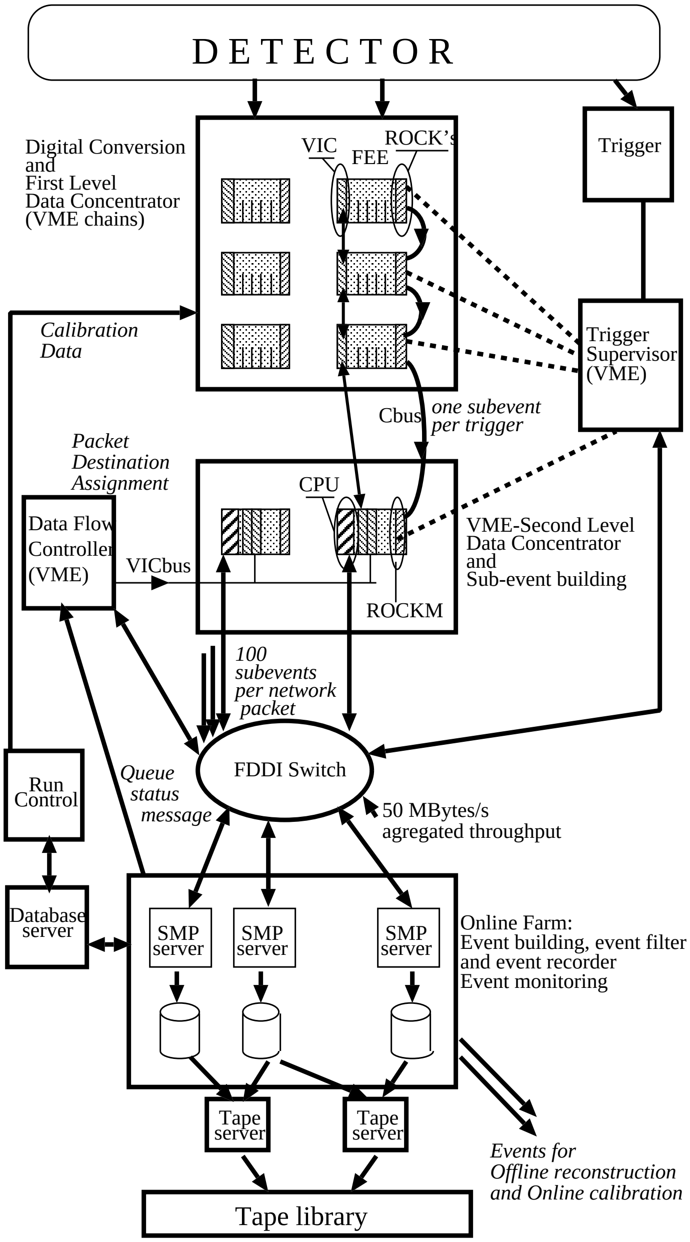


Fig. 2. DAQ architecture

Packets of 100 sub-events are buffered in the VME CPUs and then transported using TCP/IP connections over FDDI. The destination node in the farm is assigned cyclically by a central Data Flow Controller, DFC, that distributes trigger-destination assignment tables to the VME CPUs using a commercial interconnect bus with mirrored memories, VICbus. Each DFC table contains a flag for any node in the online farm enabling or disabling the node cooperation to the data acquisition, and a word indicating the maximum trigger number for which the table is still valid. This capability of redirecting the network traffic in case of sporadic high occupancy of a node, strongly reduces DAQ dead times.

The Run Control node, based in a SMP 4-ways server, centralizes the functions related to the control of the run activity and provides the graphic operator interfaces for DAQ monitoring and error logs. This node acts also as the NFS (Network File Systems) disk server of the acquisition network where software, calibration data and history files containing run conditions and detector/accelerator parameters are recorded.

A Database server maintains track of run conditions, accelerator and detector status for any run. Moreover relevant parameters related to any acquired or analyzed event file are entered in the database.

The online farm provides functionalities of event building and event recording in local disks. In parallel, specialized processes are in charge for selecting interesting events that are used to monitor the quality of the acquired data and the goodness of the detector calibration constants. Feedback to the collider team (e.g. instantaneous luminosity, beam energy) is provided online using this early data analysis.

Each node in the farm exports via NFS its own disk to an offline farm of servers where the full off-line analysis is started as soon as the goodness of the calibration data is confirmed.

A Gigabit and Fast Ethernet switched network interconnects all the KLOE computers and is used in parallel with the 'DAQ path' for traffic not correlated with the data acquisition. Two SMP servers with Gigabit Ethernet interfaces are dedicated to archive files from the online disks to a Tape Library.

IV. Software components

The acquisition software has been developed having in mind performances, scalability and independence from processor and network platforms. In fact machines running HPUX versions 9 and 10, AIX v4.3 and OSF/1 v4.0 operating systems are presently used in the KLOE data acquisition and functionalities can be easily moved among different platforms with minor effort.

The core of the data acquisition software was designed from scratch, but commercial and public domain packages are used for many applications. Examples of them are presented in Table III.

TABLE III

Commercial and public domain software.

Package	Application
DB2[7]	Online database
ADSM[8]	Archiving and Backup
HEPDB[9]	Calibration database
ROOT[10]	Histograms and Plots
ONX[11]	Event display
TCL/TK	Monitoring tools
HTML	Slow Control interface

A. *The message system*

The online processes are distributed in a large set of nodes of the acquisition network and have therefore to intercommunicate by means of an adequate messaging system. All processes have to change coherently their state of activity according to local or remote commands. Monitoring of the process activity should not interfere with its cooperation in moving data. Having these ideas in mind, an acquisition node was designed as a network device that can controlled using the standard Simple Network Management Protocol, SNMP.

The SNMP was developed to provide a general purpose internetworking management protocol. It allows to access remote information about network configuration, traffic, faults, accounting and security. Privileged access are also defined to force changes in remote devices or software applications. The information is made available as a tree of conceptual variables, defined in a Manager Information Base (MIB) [12] using elements of the ASN.1 notation. Well maintained, easy-to-use public domain software is available, implementing both dedicated daemons and utilities for remote access.

The structure of a typical data acquisition process and its interaction with the message system are described in the following. At the startup the process subscribes itself by specifying a set of its more significant variables to a shared process table implemented as a UNIX shared memory. A process message queue is created and its identification is also saved in that table. During the process activity the value of its variables are periodically updated in the shared memory.

A KLOE command server running in any acquisition node is implemented as a private SNMP daemon. A special MIB tree has been defined that maps the structure of the process table in one-indexed and two-indexed lists. The command server implements this mapping.

The command server is able to respond to "set", "get" and "getnext" queries. The "get" and "getnext" queries allow to scan remotely the process table and their variables, thus performing a sort of generalized shared memory. Remote "set" operations allow to modify the value of single variables, which is extremely useful for debugging purposes, and can be converted into local commands addressed to one or more processes in the node when the "set" function is directed to a particular "last command" variable. The command server uses message queues and process signaling to distribute messages to its managed processes.

This message system allows to create centralized utilities which are able to monitor remote node activity by only invoking the command server cooperation in that node.

A two level acknowledgment over the SNMP protocol has been implemented [13] to manage remote commands. Moreover, in order to speed up the execution time of commands to be distribute to more processes, parallel command execution and delayed acknowledgment requests are defined. It is very useful in implementing for instance run control commands that take times as long as seconds to be executed by single nodes.

As a reference, the average time to get a remote variable in the KLOE DAQ network is ∼ 1.2 ms and the remote command completion time is ∼ 4 ms, including the acknowledgment waiting time.

SNMP traps are used to implement the Data Flow Control protocol [13].

Examples of utilities walking in a remote list of DAQ processes and variables responding to the two commands specified below are given in the following:

1) Command: *procs fibm05*

```
process:cmdsrv        status:alive runstate:INIT
process:lpcheck       status:alive runstate:INIT
process:receiver      status:alive runstate:INIT
process:build         status:alive runstate:INIT
process:recorder      status:alive runstate:INIT
process:trgmon        status:alive runstate:INIT
process:filter        status:alive runstate:INIT
process:recorder_BHA  status:alive runstate:INIT
process:recorder_COS  status:alive runstate:INIT
```

2) Command: *varnames trgmon@fibm05*

```
runnr = 18871
L1 = 4621
L2 = 3046
dead = 5.533692
ratephi = 30.638260
..................
```

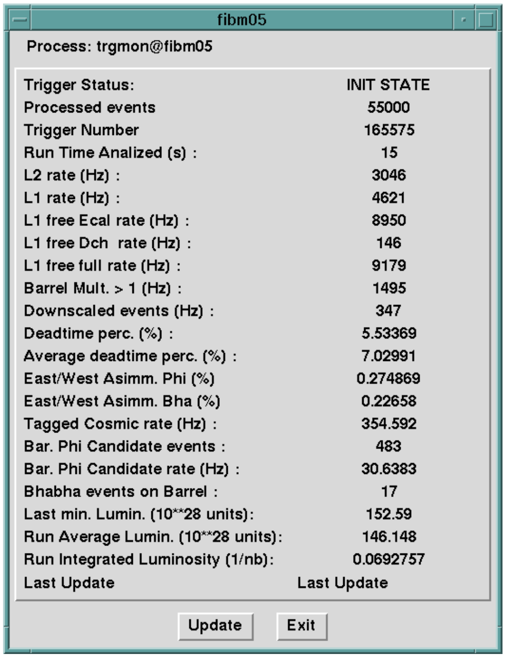
Graphic tools have been built using the tcl/tk package in order to help full process control. Only two new tcl commands have been implemented:

- a “get variable” command
- a “send message” command

As an example, Figure 3 gives a graphic presentation of the “varnames” utility.

B. Process template

Processes taking part in the data acquisition are built according to a well defined template implementing the DAQ state machine. After the initialization phase, a DAQ process enters in a main loop where sequentially executes its private actions and reacts to external commands. A set of libraries written in “C” language provides the interfaces to the Message System, hardware components, TCP/IP sockets, shared memories, and so on, thus keeping in the process code only its specific functionalities.



fibm05

Process: trgmon@fibm05

Trigger Status:	INIT STATE
Processed events	55000
Trigger Number	165575
Run Time Analized (s) :	15
L2 rate (Hz) :	3046
L1 rate (Hz) :	4621
L1 free Ecal rate (Hz) :	8950
L1 free Dch rate (Hz) :	146
L1 free full rate (Hz) :	9179
Barrel Mult. > 1 (Hz) :	1495
Downscaled events (Hz) :	347
Deadtime perc. (%) :	5.53369
Average deadtime perc. (%) :	7.02991
East/West Asimm. Phi (%)	0.274869
East/West Asimm. Bha (%)	0.22658
Tagged Cosmic rate (Hz) :	354.592
Bar. Phi Candidate events :	483
Bar. Phi Candidate rate (Hz) :	30.6383
Bhabha events on Barrel :	17
Last min. Lumin. (10**28 units):	152.59
Run Average Lumin. (10**28 units):	146.148
Run Integrated Luminosity (1/nb):	0.0692757
Last Update	Last Update

Fig. 3. Trigger monitor

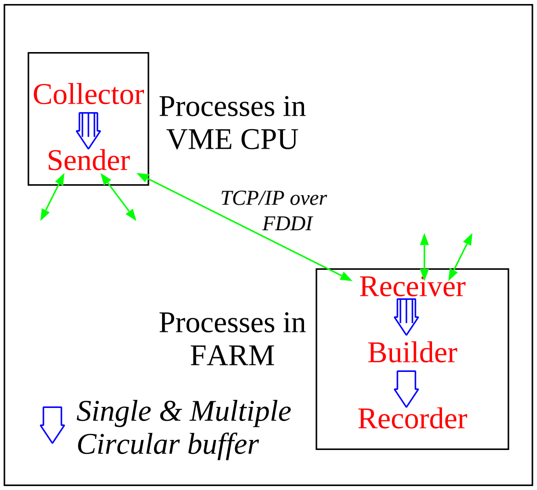


Fig. 4. Data moving processes

V. Data moving processes

Processes involved in the ‘DAQ path’ from the ROCKM buffer to the online farm disks are presented in Figure 4.

The *Collector* initializes the hardware modules in its VME chain and executes debugging tests in case of failure in order to identify the error source. In the main loop it also reads the ROCKM buffer and builds packets containing 100 sub-events. Direct memory access or single-word VME read operations are used according to the real occupancy of the buffer, that can change during the single run time, which is usually less than 1 hour. Packets are put directly in a multiple shared circular buffer according to the destination node in the online farm, as defined in the corresponding DFC table. The structure and characteristics of the KLOE circular buffer will be described below in this section.

The *Sender* extracts packets from the circular buffer and sends them using TCP/IP connections that remain open during all the run activity. TCP/IP parameters are tuned in order to reduce dead times due to network latencies. In fact a hardware busy condition will be originated in a VME chain if the *Collector* detects a buffer full status.

The *Receiver* reads data coming from the 10 *Sender*s and puts them in a multiple circular buffer according to the data source. Moreover it continuously monitors the buffer occupancy and keeps it under control using the following procedure:

- when the occupancy exceeds a given upper threshold the *Receiver* sends an “almost full” SNMP traps to the DFC,
- the DFC eliminates the almost full node from the acquisition chain by changing the validity word of the last flow table according to the ‘actual’ trigger number obtained from the Trigger Supervisor, increased by a delta that is dynamically calculated as a function of the trigger rate and the DFC internal latencies. Finally a new flow table with infinite validity trigger is distributed to the VME CPUs,
- when the *Receiver* detects a buffer occupancy below a given lower threshold, it sends to DFC an “almost empty” trap signal, that will originate a new DFC procedure in order to reinsert the node in the DAQ path.

Thresholds can be dynamically modified during the run according to the real trigger rate. The average time needed to receive and react to a trap is ∼1.2 ms, while the average total reaction time of the DFC system is ∼ 10 ms, equivalent to the time of 100 events at the maximum expected trigger rate of 10 kHz.

The *Builder* puts together all sub-event data related to the same trigger number, checks the consistency and formats the event structure in YBOS[14] banks. Events are put in a new circular buffer where they are subsequently extracted from by the *Recorder* in order to be written on disk files, which have presently the size of 1 GB, or to be simply thrown away during debugging or monitoring runs. Files are finally archived by remote nodes that mount NFS the online farm disks.

A. The multiple circular buffer

The circular buffer is a relevant part of the KLOE DAQ architecture. In order to maintain a high degree of global performance, processes that cooperate in moving data have to reduce to the minimum the required number of buffer copies. On the other hand, monitoring and calibrations tools need access to event data in order to check continuously data quality and calibration constants, and to recalculate them if needed.

The circular buffer library has been designed and implemented having in mind high performance requirement. Some peculiarities are listed in the following:

- Only one master puts data in the buffer;
- Only one client extracts data from the buffer;
- The master reserves space every time a new packet of events has to be acquired, received or built, and puts the relative data directly into the buffer. Excess of space is released when the buffer is validated and becomes available to the clients. A short time of memory locking is requested, long enough to modify internal pointers;
- Many clients can copy in a local space events from the circular buffer, to be used as random samples for statistical analysis. These will be called spy events in the following.

At present the KLOE circular buffer library is supported on AIX, HP-UX, Lynux, LynxOS, OSF1 and Solaris platforms. It can be configured as a single or multi-thread environment using SYSV or POSIX compliant routines to access shared memories and semaphores.

B. Processes in a node of the online farm

Figure 5 presents the full software configuration of a node in the online farm.

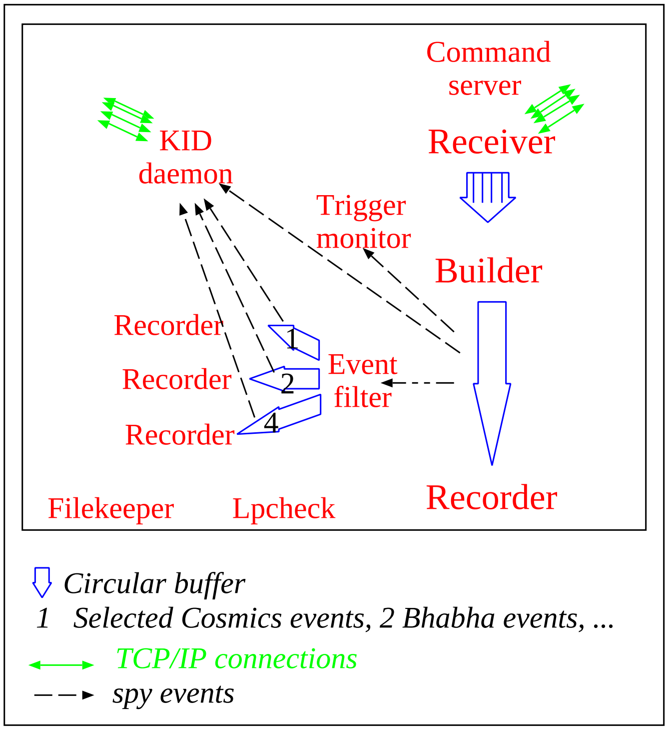


Fig. 5. Processes in a farm node

The *Trigger monitor* uses the information collected by the trigger chains in order to measure trigger rates, acquisition dead times, Cosmic and Bhabha event rates, DAΦNE instantaneous luminosity and other relevant parameters. The *Event filter* selects different event categories using information from the whole detector to allow for remote control of calibration constants and data quality.

The *KID daemon*, that will be described in the next section, distributes spy events to remote processes. The *Filekeeper* maintains free space on disk deleting files according to a policy that assures that data acquisition is never stopped due to lack of space but assures also that offline processes can proceed analyzing files without requiring access to tapes.

The *Lpcheck* monitors the status on life of the processes from the data acquisition point of view, by requesting the change of an internal variable.

Figure 6 shows the distribution of CPU usage in a node that sustains event rates of 0.8 and 1.6 kHz. A single process can only use up to 25 % of the total CPU power which is a natural limit to the resources used by processes with monitoring function. A fine tune of the operating system and NFS parameters has been required in order to keep control in system dead times that could be originated by lack of resources in the farm.

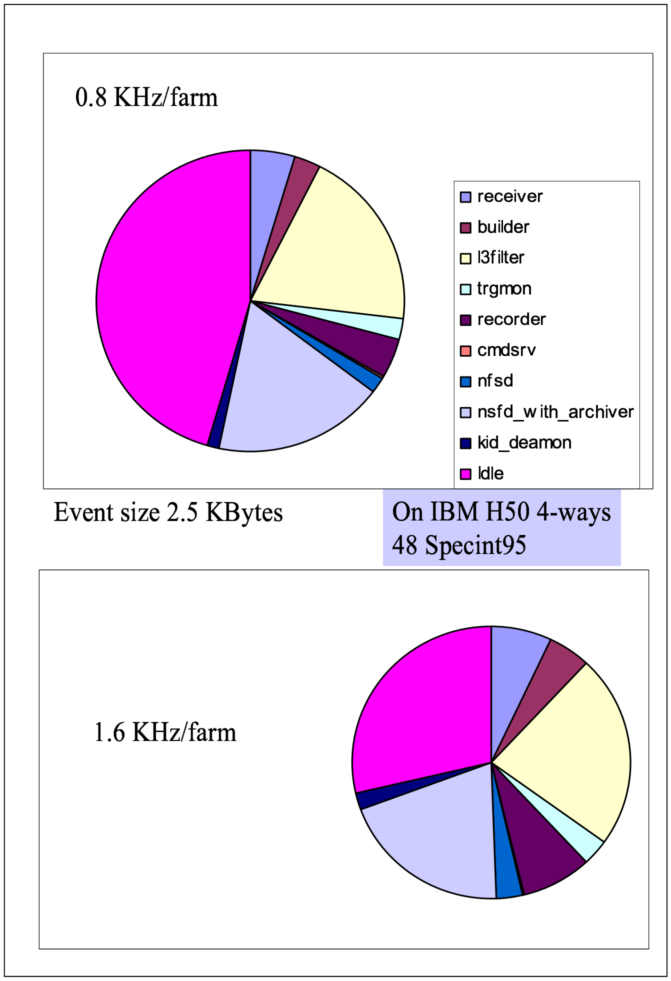


Fig. 6. CPU utilization in a node of the online farm

VI. The KLOE Integrated Dataflow (KID)

KLOE data are stored using millions of large files containing raw, reconstructed or simulated events. All data files are indexed using a Relational Database that keeps track of file parameters and actual file location. Tapes are used for long-term storage and large disk areas are used for short-range data staging. Events can also be peeked directly from data acquisition pipes.

The KID package[15] provides an unique application programming interface to access data in KLOE. The source of data can be selected using simple and easy to remember text strings in the form of URLs, Universal Resource Identifier. Several data sources can be combined together like circular buffers or a set of files implicitly selected by SQL queries to the Database.

KID also optimizes data access by moving files to the right storage medium (e.g., from tape to disk) so that there is no need to track down explicitly the location of the files as this is done transparently by KID.

The KID environment is a very powerful tool that was first designed for offline event reconstruction and data analysis. By adding the interface to the DAQ multiple circular buffer, KID allows the use of the same analysis tools (e.g., event display, histograming, calibration) both in online and offline environments by simply defining the right data source.

Examples of URL:

- *spydaq:ALL*

to receive raw unfiltered events from all the online farms,

- *spydaq:L3BHA?multi=n?wait_run=no*

to ask for raw events that are acquired in all the online farms, being identified by the online filters as Bhabha events. The process asking for this input will receive an End-of-file at the end of the run,

- *dbdatarec:run_nr between 19005 and 19011 and stream_code= 'rad'?report=true*

to analyze all files of type *rad* reconstructed from raw files in the given range of run numbers.

The KLOE Event display uses KID to access raw or reconstructed data. It is built as two cooperating processes. One of them reads a single event from the requested data source, reconstructs its components (clusters in the calorimeters, tracks and vertices in the drift chamber) extending the event data with new banks, or uses already reconstructed values depending on the user request. The event is written to a shared memory. The second process reads the event from the shared memory and implements the graphic interface.

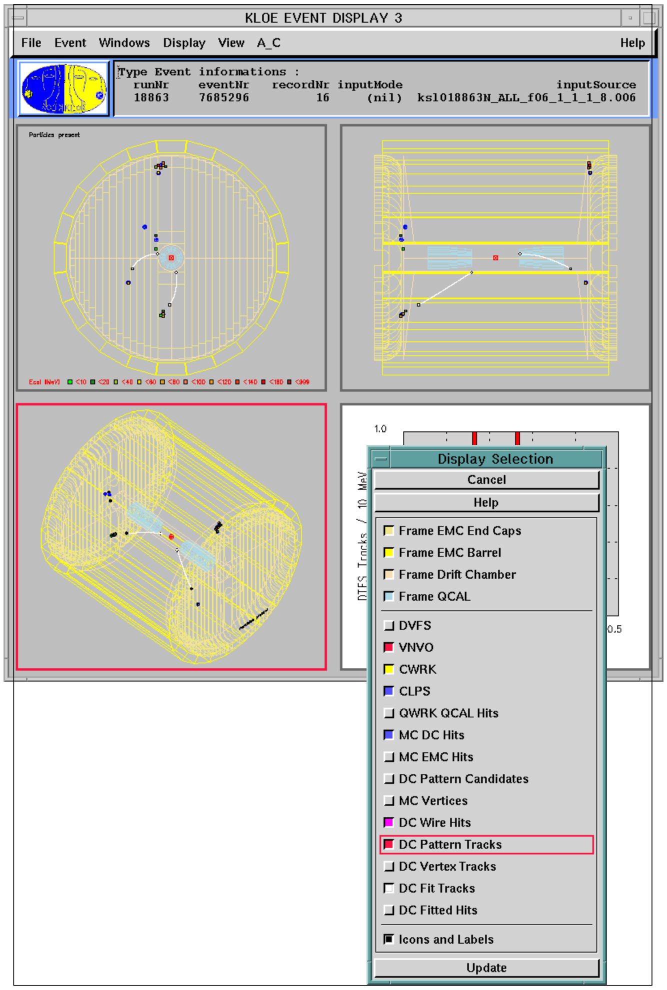
Figure 7 shows partially the KLOE Event display interface. Information not appearing in the figure is related in particular to reconstructed track momentum and coordinates, energy and time of the calorimeter clusters and reconstructed decay vertices, that can be obtained in a separated window just simple clicking the object in the display.

Figure 8 shows the identification of the particles produced in the event presented in Figure 7.

VII. Database and archiving systems

An IBM 3494 Tape library with 5500 cartridges of 40 Gbytes uncompressed capacity, six (up to 26) tape drives and the ADSM software package are the components of the KLOE archive and backup system. In order to improve system availability, tape drives are physically connected to two H80 servers and each server manages one half of the library capacity maintaining its own database.

A set of dedicated processes that use the ADSM Application Programming Interface, API, implement the auto-



KLOE EVENT DISPLAY 3

```
Type Event informations :
  runNr     eventNr    recordNr inputMode                     inputSource
  18863     7685296          16     (nil)  ksl018863N_ALL_f06_1_1_1_8.006
```

Display Selection

- ☑ Frame EMC End Caps
- ☑ Frame EMC Barrel
- ☑ Frame Drift Chamber
- ☑ Frame QCAL
- ☐ DVFS
- ☑ VNVO
- ☑ CWRK
- ☑ CLPS
- ☐ QWRK QCAL Hits
- ☑ MC DC Hits
- ☐ MC EMC Hits
- ☐ DC Pattern Candidates
- ☐ MC Vertices
- ☑ DC Wire Hits
- ☑ DC Pattern Tracks
- ☐ DC Vertex Tracks
- ☐ DC Fit Tracks
- ☐ DC Fitted Hits
- ☑ Icons and Labels

Fig. 7. Event display

matic files archive at completion of each data acquisition run. The same processes provide archiving and disk staging facilities for offline analysis.

An external database based on the DB2 software package is used to better identify the main characteristics of any archived event file. This DB2 database also contains the history of run and detector conditions from the very beginning of the KLOE activity in DAΦNE.

The ADSM client option is installed in any KLOE node that requires backup functionalities. On the contrary, access to the DB2 database is provided by a custom multithread client-server system that requires DB2 software running only in the database server.

A 2 GB ADSM database size has been successfully tested. KLOE is presently using 300 MB per server for about 600.000 entries, while the DB2 database size is now 800 MBytes.

VIII. Slow Control

The KLOE slow control system is responsible for setting and/or monitoring constants related to high and low voltage power suppliers, gas, temperature, currents and noise channels in the drift chamber and many other parameters that help operators to monitor the full detector and DAΦNE performances during the run. It is managed by a CPU in a VME crate and uses CAENET[16] modules interconnected via serial lines.

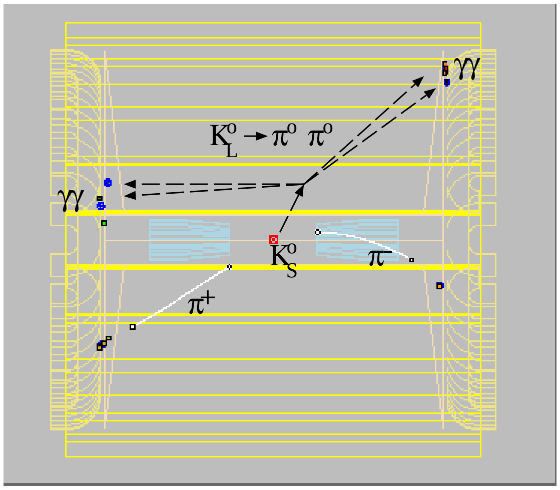
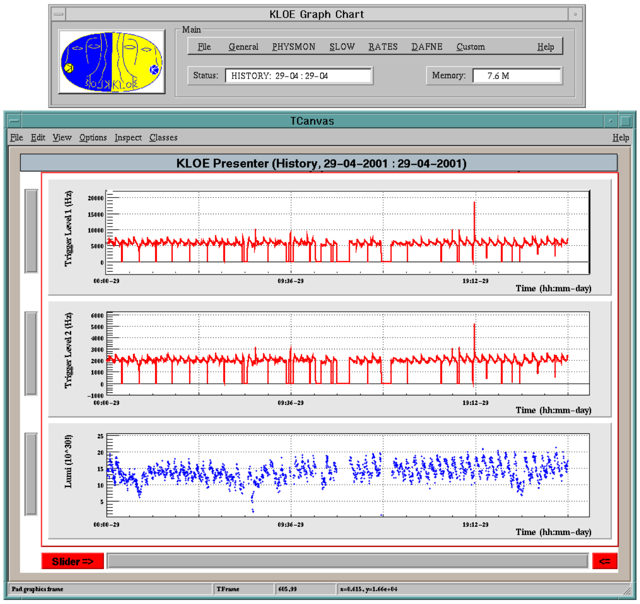


Fig. 8. Event display with a complete reconstructed event

History files containing slow control variables are periodically updated. The *Presenter* utility (see Figure 9) is used to monitor continuously sets of these variables both during the run activity and all along the KLOE history. Moreover HTML pages (see an example in figure 10) are used as slow control operator interface.



KLOE Graph Chart

Status: HISTORY: 29-04 : 29-04

Memory: 7.6 M

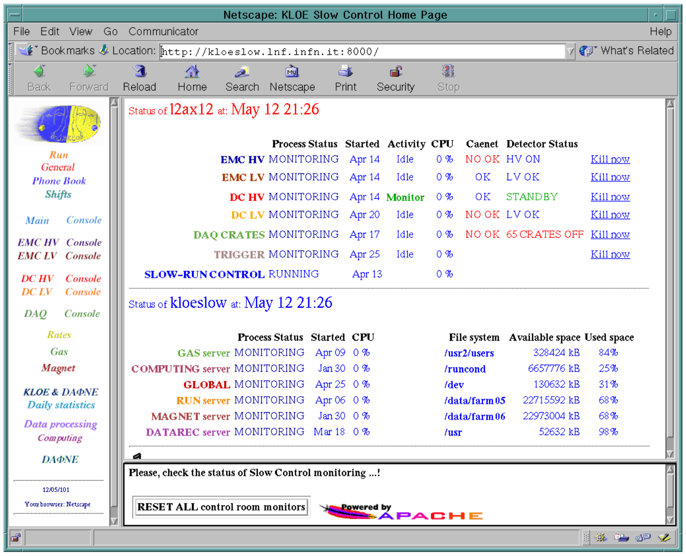
TCanvas

Fig. 9. Presenter

IX. KLOE and DAΦNE status

As a reference, Figure 9 shows the DAΦNE luminosity achieved during the KLOE run period in April 2001. The collider was working with 45 bunches, and the peak luminosity was $\sim 2 \times 10^{31}\ cm^{-2}\ sec^{-1}$. The two KLOE trigger rates are also shown. The average data acquisition rate, L2, was about 2.5 kHz.

KLOE is going to publish the results about physics channels not requiring very high integrated luminosity, while DAΦNE is periodically increasing performances.



Status of l2ax12 at: May 12 21:26

	Process Status	Started	Activity	CPU	Caenet	Detector Status	
EMC HV	MONITORING	Apr 14	Idle	0 %	NO OK	HV ON	Kill now
EMC LV	MONITORING	Apr 14	Idle	0 %	OK	LV OK	Kill now
DC HV	MONITORING	Apr 14	Monitor	0 %	OK	STANDBY	Kill now
DC LV	MONITORING	Apr 20	Idle	0 %	NO OK	LV OK	Kill now
DAQ CRATES	MONITORING	Apr 17	Idle	0 %	NO OK	65 CRATES OFF	Kill now
TRIGGER	MONITORING	Apr 25	Idle	0 %			Kill now
SLOW-RUN CONTROL	RUNNING	Apr 13		0 %			

Status of kloeslow at: May 12 21:26

	Process Status	Started	CPU	File system	Available space	Used space
GAS server	MONITORING	Apr 09	0 %	/usr2/users	328424 kB	84%
COMPUTING server	MONITORING	Jan 30	0 %	/runcond	6657776 kB	25%
GLOBAL	MONITORING	Apr 25	0 %	/dev	130632 kB	31%
RUN server	MONITORING	Apr 06	0 %	/data/farm05	22715592 kB	68%
MAGNET server	MONITORING	Jan 30	0 %	/data/farm06	22973004 kB	68%
DATAREC server	MONITORING	Mar 18	0 %	/usr	52632 kB	98%

Please, check the status of Slow Control monitoring ...!

Fig. 10. Slow control interface

X. CONCLUSIONS

The core of the KLOE data acquisition has been designed almost from scratch having in mind high performance, computer platform independence and distributed monitoring capability.

The use of custom buses and a fine tuning of network and operating system parameters allows to maintain a low acquisition dead time even when the available computing resources are partially dedicated to activities like event selection, data quality control and detector calibration.

The KLOE message system, based on standard UNIX mechanisms and the SNMP protocol, proved to be a robust and reliable system for data acquisition and process control.

Commercial software for archiving and database accessing have been successfully integrated with the custom KLOE dataflow software, which allows a transparent access to event files independently of the storage location.

APPENDIX

The KLOE Collaboration

M. Adinolfi[m], A. Aloisio[g], F. Ambrosino[g], A. Andryakov[f], A. Antonelli[c], M. Antonelli[c], F. Anulli[c], C. Bacci[n], G. Barbiellini[q], F. Bellini[n], G. Bencivenni[c], S. Bertolucci[c], C. Bini[k], C. Bloise[c], V. Bocci[k], F. Bossi[c], P. Branchini[n], S. A. Bulychjov[f], G. Cabibbo[k], A. Calcaterra[c], R. Caloi[k], P. Campana[c], G. Capon[c], G. Carboni[m], A. Cardini[k], M. Casarsa[q], V. Casavola[e], G. Cataldi[e], F. Ceradini[n], F. Cervelli[j], F. Cevenini[g], G. Chiefari[g], P. Ciambrone[c], S. Conetti[r], E. De Lucia[k], G. De Robertis[a], R. De Sangro[c], P. De Simone[c], G. De Zorzi[k], S. Dell'Agnello[c], A. Denig[c], A. Di Domenico[k], C. Di Donato[g], S. Di Falco[j], A. Doria[g], E. Drago[g], O. Erriquez[a], A. Farilla[n], G. Felici[c], A. Ferrari[n], M. L. Ferrer[c], G. Finocchiaro[c], C. Forti[c], A. Franceschi[c], P. Franzini[ki], M. L. Gao[b], C. Gatti[c], P. Gauzzi[k], A. Giannasi[j], S. Giovannella[c], V. Golovatyuk[e], E. Gorini[e], F. Grancagnolo[e], W. Grandegger[c], E. Graziani[n], P. Guarnaccia[a], H. G. Han[b], S. W. Han[b], X. Huang[b], M. Incagli[j], L. Ingrosso[c], Y. Y. Jiang[b], W. Kim[o], W. Kluge[d], C. Kuo[d], V. Kulikov[f], F. Lacava[k], G. Lanfranchi[c], J. Lee-Franzini[co], T. Lomtadze[j], C. Luisi[k], C. S. Mao[b], M. Martemianov[d], A. Martini[c], M. Matsyuk[f], W. Mei[c], A. Menicucci[m], L. Merola[g], R. Messi[m], S. Miscetti[c], A. Moalem[h], S. Moccia[c], M. Moulson[c], S. Mueller[d], F. Murtas[c], M. Napolitano[g], A. Nedosekin[cf], M. Panareo[e], L. Pacciani[m], P. Pagès[c], M. Palutan[n], L. Paoluzi[m], E. Pasqualucci[k], L. Passalacqua[c], M. Passaseo[k], A. Passeri[n], V. Patera[lc], E. Petrolo[k], G. Petrucci[c], D. Picca[k], G. Pirozzi[g], M. Pollack[o], L. Pontecorvo[k], M. Primavera[e], F. Ruggieri[a], P. Santangelo[c], E. Santovetti[m], G. Saracino[g], R. D. Schamberger[o], C. Schwick[j], B. Sciascia[k], A. Sciubba[lc], F. Scuri[q], I. Sfiligoi[c], J. Shan[c], P. Silano[k], T. Spadaro[k], S. Spagnolo[e], E. Spiriti[n], C. Stanescu[n], G. L. Tong[b], L. Tortora[n], E. Valente[k], P. Valente[c], B. Valeriani[d], G. Venanzoni[j], S. Veneziano[k], A. Ventura[e], Y. Wu[b], Y. G. Xie[b], P. F. Zema[j], P. P. Zhao[b], Y. Zhou[c]

[a] Dipartimento di Fisica dell'Università e Sezione INFN, Bari, Italy.

[b] Institute of High Energy Physics of Academica Sinica, Beijing, China.

[c] Laboratori Nazionali di Frascati dell'INFN, Frascati, Italy.

[d] Institut für Experimentelle Kernphysik, Universität Karlsruhe, Germany.

[e] Dipartimento di Fisica dell'Università e Sezione INFN, Lecce, Italy.

[f] Institute for Theoretical and Experimental Physics, Moscow, Russia.

[g] Dipartimento di Scienze Fisiche dell'Università e Sezione INFN, Napoli, Italy.

[h] Physics Department, Ben-Gurion University of the Negev, Israel.

[i] Physics Department, Columbia University, New York, USA.

[j] Dipartimento di Fisica dell'Università e Sezione INFN, Pisa, Italy.

[k] Dipartimento di Fisica dell'Università e Sezione INFN, Roma I, Italy.

[l] Dipartimento di Energetica dell'Università, Roma I, Italy.

[m] Dipartimento di Fisica dell'Università e Sezione INFN, Roma II, Italy.

[n] Dipartimento di Fisica dell'Università e Sezione INFN, Roma III, Italy.

[o] Physics Department, State University of New York at Stony Brook, USA.

[q] Dipartimento di Fisica dell'Università e Sezione INFN,

Trieste, Italy.
[r] Physics Department, University of Virginia, USA.
[*] Associate member

References

[1] The DAΦNEProject Team, *Proposal for a ϕ-factory*, LNF-90/031 1990;
C. Milardi *et al.*, *Status of DAΦNE*, Proceedings of the Workshop on Physics and Detectors for DAFNE, Frascati, Italy (1999)
[2] The KLOE Collaboration, *The KLOE Detector Technical Proposal*, LNF-93/002, (1993);
The KLOE Collaboration, *The KLOE Data Acquisition System, addendum to the Technical Proposal*, LNF-95/014, 1995;
S.Bertolucci for the KLOE Collaboration, *A status report of KLOE at DAΦNE*, Proceedings of the 19th International Symposium on Lepton and Photon Interactions at High Energies, Stanford, USA (1999), hepex/002030.
[3] M. Adinolfi *et al.*, The electromagnetic calorimeter of KLOE, Submitted to Elsevier Preprint, (2001).
[4] M. Adinolfi *et al.*, The Tracking Detector of the KLOE Experiment, Submitted to Elsevier Preprint, (2001)
[5] M. Adinolfi *et al.*, The QCAL tile calorimeter of KLOE, Submitted to Elsevier Preprint, (2001).
[6] A. Aloisio *et al.*, *Level-1 DAQ system for the KLOE experiment*, CHEP'95, Rio de Janeiro, 1995
[7] DB2 Product Family, http://www-4.ibm.com/software/data/db2
[8] Storage Solutions http://www.storage.ibm.com/
[9] HepDB, a High Energy Physics Database Management package, http://wwwinfo.cern.ch/asdoc/hepdb/
[10] ROOT, a Object-Oriented interactive data analysis tool, CERN Main Page http://root.cern.ch/
[11] OnX, a package to handle GUI with XML Main WEB page, http:// www.lal.in2p3.fr/SI/OnX
[12] AA. VV., *The Simple Web - The source for SNMP and Internet management info* http://www.simpleweb.org
[13] E. Pasqualucci *et al.*, Process and Data Flow Control in KLOE, CHEP'2000, Padova Italy (2000)
[14] http://www-cdf.fnal.gov/offline/ybos/ybos.html, Fermilab CDF library
[15] I. Sfiligoi *et al.*, The KLOE Integrated Dataflow (KID), Submitted to CHEP'2001, Beijing, September 20001
[16] see for instance http://www.caen.it/

Maria Lorenza Ferrer received the Ph.D. degree in physics from the University of Madrid, Spain, in 1972. She collaborated on several experiments in particle physics at the Junta de Energia Nuclear (Spain), Laboratori Nazionali di Frascati, INFN (Italy), Fermilab (U.S.A.) and CERN (Switzerland). Since 1977 she has worked at the National Laboratory of the INFN where she was coordinator of the Computing Center group and collaborated on computing and network research programs: microprocessor for on-line control processes, fiber optical networks and internetwork gateways for file transfer, access and management. She was a member of the INFN National Commission for network setup and management. Since 1990 she coordinates the Data Acquisition group of the KLOE experiment and is involved in the setting and management of the KLOE computing resources.