The data acquisition system for the ANTARES neutrino telescope

J.A. Aguilar [j], A. Albert [u], F. Ameli [x], M. Anghinolfi [i], G. Anton [g], S. Anvar [y], E. Aslanides [e], J-J. Aubert [e], E. Barbarito [b], S. Basa [r], M. Battaglieri [i], Y. Becherini [c,2], R. Bellotti [b], J. Beltramelli [y], V. Bertin [e], A. Bigi [w], M. Billault [e], R. Blaes [u], N. de Botton [y], M.C. Bouwhuis [v,*], S.M. Bradbury [t], R. Bruijn [v,ab], J. Brunner [e], G.F. Burgio [f], J. Busto [e], F. Cafagna [b], L. Caillat [e], A. Calzas [e], A. Capone [x], L. Caponetto [f], E. Carmona [j], J. Carr [e], S.L. Cartwright [z], D. Castel [u], E. Castorina [w], V. Cavasinni [w], S. Cecchini [c,m], A. Ceres [b], P. Charvis [h], P. Chauchot [k], T. Chiarusi [x], M. Circella [b], C. Colnard [v], C. Compère [k], R. Coniglione [s], N. Cottini [w], P. Coyle [e], S. Cuneo [i], A-S. Cussatlegras [d], G. Damy [k], R. van Dantzig [v], C. De Marzo [b,1], I. Dekeyser [d], E. Delagnes [y], D. Denans [y], A. Deschamps [h], F. Dessages-Ardellier [y], J-J. Destelle [e], B. Dinkespieler [e], C. Distefano [s], C. Donzaud [y], J-F. Drogou [ℓ], F. Druillole [y], D. Durand [y], J-P. Ernenwein [u], S. Escoffier [e], E. Falchini [w], S. Favard [e], F. Feinstein [e], S. Ferry [n], D. Festy [k], C. Fiorello [b], V. Flaminio [w], S. Galeotti [w], J-M. Gallone [n], G. Giacomelli [c], N. Girard [u], C. Gojak [e], Ph. Goret [y], K. Graf [g], G. Hallewell [e], M.N. Harakeh [q], B. Hartmann [g], A. Heijboer [v,ab], E. Heine [v], Y. Hello [h], J.J. Hernández-Rey [j], J. Hößl [g], C. Hoffman [n], J. Hogenbirk [v], J.R. Hubbard [y], M. Jaquet [e], M. Jaspers [v,ab], M. de Jong [v], F. Jouvenot [y], N. Kalantar-Nayestanaki [q], A. Kappes [g], T. Karg [g], S. Karkar [e], U. Katz [g], P. Keller [e], H. Kok [v], P. Kooijman [v,aa], C. Kopper [g], E.V. Korolkova [z], A. Kouchner [a], W. Kretschmer [g], A. Kruijer [v], S. Kuch [g], V.A. Kudryavstev [z], D. Lachartre [y], H. Lafoux [y], P. Lagier [e], R. Lahmann [g], G. Lamanna [e], P. Lamare [y], J.C. Languillat [y], H. Laschinsky [g], Y. Le Guen [k], H. Le Provost [y], A. Le Van Suu [e], T. Legou [e], G. Lim [v,ab], L. Lo Nigro [f], D. Lo Presti [f],

H. Loehner [q], S. Loucatos [y], F. Louis [y], F. Lucarelli [x],
V. Lyashuk [p], M. Marcelin [r], A. Margiotta [c], R. Masullo [x],
F. Mazéas [k], A. Mazure [r], J.E. McMillan [z], R. Megna [b],
M. Melissas [e], E. Migneco [s], A. Milovanovic [t], M. Mongelli [b],
T. Montaruli [b], M. Morganti [w], L. Moscoso [y,a], M. Musumeci [s],
C. Naumann [g], M. Naumann-Godo [g], V. Niess [e], C. Olivetto [n],
R. Ostasch [g], N. Palanque-Delabrouille [y], P. Payre [e], H. Peek [v],
C. Petta [f], P. Piattelli [s], J-P. Pineau [n], J. Poinsignon [y],
V. Popa [c,o], T. Pradier [n], C. Racca [n], N. Randazzo [f],
J. van Randwijk [v], D. Real [j], B. van Rens [v], F. Réthoré [e],
P. Rewiersma [v,1], G. Riccobene [s], V. Rigaud [ℓ], M. Ripani [i],
V. Roca [j], C. Roda [w], J.F. Rolin [k], M. Romita [b], H.J. Rose [t],
A. Rostovtsev [p], J. Roux [e], M. Ruppi [b], G.V. Russo [f], F. Salesa [j],
K. Salomon [g], P. Sapienza [s], F. Schmitt [g], J-P. Schuller [x,y],
R. Shanidze [g], I. Sokalski [b], T. Spona [g], M. Spurio [c],
G. van der Steenhoven [v], T. Stolarczyk [y], K. Streeb [g],
D. Stubert [u], L. Sulak [e], M. Taiuti [i], C. Tamburini [d], C. Tao [e],
G. Terreni [w], L.F. Thompson [z], P. Valdy [ℓ], V. Valente [x],
B. Vallage [y], G. Venekamp [v], B. Verlaat [v], P. Vernin [y],
R. de Vita [i], G. de Vries [v,aa], R. van Wijk [v],
P. de Witt Huberts [v], G. Wobbe [g], E. de Wolf [v,ab], A-F. Yao [d],
D. Zaborov [p], H. Zaccone [y], J.D. Zornoza [j], J. Zúñiga [j]

[a] *AstroParticule et Cosmologie, UMR 7164 (CNRS, Université Paris 7, CEA, Observatoire de Paris), 11, place Marcelin Berthelot, 75005 Paris, France*

[b] *Dipartimento Interateneo di Fisica e Sezione INFN, Via E. Orabona 4, 70126 Bari, Italy*

[c] *Dipartimento di Fisica dell'Università e Sezione INFN, Viale Berti Pichat 6/2, 40127 Bologna, Italy*

[d] *Centre d'Océanologie de Marseille, CNRS/INSU Université de la Méditerranée Aix-Marseille II, Station Marine d'Endoume-Luminy, Rue de la Batterie des Lions, 13007 Marseille, France*

[e] *CPPM – Centre de Physique des Particules de Marseille, CNRS/IN2P3 Université de la Méditerranée Aix-Marseille II, 163 Avenue de Luminy, Case 907, 13288 Marseille Cedex 9, France*

[f] *Dipartimento di Fisica ed Astronomia dell'Università e Sezione INFN, Viale Andrea Doria 6, 95125 Catania, Italy*

[g] *Friedrich-Alexander-Universität Erlangen-Nürnberg, Physikalisches Institut, Erwin-Rommel-Str. 1, D-91058 Erlangen, Germany*

[h] *UMR GéoScience Azur, Observatoire Océanologique de Villefranche, BP48, Port de la Darse, 06235 Villefranche-sur-Mer Cedex, France*

[i] *Dipartimento di Fisica dell'Università e Sezione INFN, Via Dodecaneso 33, 16146 Genova, Italy*

[j] *IFIC – Instituto de Física Corpuscular, Edificios Investigación de Paterna, CSIC – Universitat de València, Apdo. de Correos 22085, 46071 Valencia, Spain*
[k] *Centre de Brest, BP 70, 29280 Plouzané, France*
[ℓ] *Centre de Toulon/La Seyne Sur Mer, Port Brégaillon, Chemin Jean-Marie Fritz, 83500, La Seyne sur Mer, France*
[m] *INAF-IASF, via P. Gobetti 101, 40129 Bologna, Italy*
[n] *IPHC–Institut Pluridisciplinaire Hubert Curien, UMR 7178 Université Louis Pasteur Strasbourg 1 and IN2P3/CNRS, 23 rue du Loess, BP 28 – 67037 Strasbourg CEDEX 2, France*
[o] *Institute for Space Sciences, 77125 Bucharest, Magurele, Romania*
[p] *ITEP – Institute for Theoretical and Experimental Physics, B. Cheremushkinskaya 25, 117259 Moscow, Russia*
[q] *Kernfysisch Versneller Instituut (KVI), University of Groningen, Zernikelaan 25, 9747 AA Groningen, The Netherlands*
[r] *Laboratoire d'Astrophysique de Marseille, CNRS/INSU - Université de Provence Aix-Marseille I, Traverse du Siphon – Les Trois Lucs, BP 8, 13012 Marseille Cedex 12, France*
[s] *INFN – Labaratori Nazionali del Sud (LNS), Via S. Sofia 44, 95123 Catania, Italy*
[t] *School of Physics & Astronomy, University of Leeds LS2 9JT, UK*
[u] *GRPHE – Groupe de Recherche en Physique des Hautes Energies, Université de Haute Alsace, 61 Rue Albert Camus, 68093 Mulhouse Cedex, France*
[v] *Nationaal Instituut voor Kernfysica en Hoge-Energiefysica (NIKHEF), Kruislaan 409, 1098 SJ Amsterdam, The Netherlands*
[w] *Dipartimento di Fisica dell'Università e Sezione INFN, Largo B. Pontecorvo 3, 56127 Pisa, Italy*
[x] *Dipartimento di Fisica dell'Università "La Sapienza" e Sezione INFN, P.le Aldo Moro 2, 00185 Roma, Italy*
[y] *DSM/DAPNIA – Direction des Sciences de la Matière, Département d'Astrophysique de Physique des Particules de Physique Nucléaire et de l'Instrumentation Associée, CEA/Saclay, 91191 Gif-sur-Yvette Cedex, France*
[z] *Dept. of Physics and Astronomy, University of Sheffield, Sheffield S3 7RH, UK*
[aa] *Universiteit Utrecht, Faculteit Betawetenschappen, Princetonplein 5, 3584 CC Utrecht, The Netherlands*
[ab] *Universiteit van Amsterdam, Instituut voor Hoge-Energiefysica, Kruislaan 409, 1098 SJ Amsterdam, The Netherlands*

Abstract

The ANTARES neutrino telescope is being constructed in the Mediterranean Sea. It consists of a large three-dimensional array of photo-multiplier tubes. The data acquisition system of the detector takes care of the digitisation of the photo-multiplier tube signals, data transport, data filtering, and data storage. The detector is operated using a control program interfaced with all elements. The design and the implementation of the data acquisition system are described.

Key words: neutrino telescope, data acquisition system

---

1 Introduction

The ANTARES neutrino telescope [1] is used to study astrophysical sources by detecting the high-energy neutrinos that these sources may emit. The detector is deployed on the seabed at a depth of 2.5 km, and consists of a large three-dimensional array of 900 light-sensitive photo-multiplier tubes (PMTs [2], [3]). Neutrinos are detected indirectly, after an interaction inside or in the vicinity of the detector. The produced charged particles emit Cherenkov light, which can be detected by the PMTs. From the known positions of the PMTs, and the measured arrival times of the Cherenkov photons, the signal produced by these charged particles can be distinguished from the background.

The main purpose of the data acquisition (DAQ) system is to convert the analogue signals from the PMTs into a format suitable for the physics analysis. To achieve this, the DAQ system has the task to prepare the detector for data taking, convert the analogue signals from the PMTs into digital data, transport the data to shore, filter the different physics signals from the background, store the filtered data on disk, and archive the run settings. The DAQ hardware and software are described in Sections 2 and 3 respectively, and the overall performance of the ANTARES DAQ system is summarised in Section 4.

2 DAQ hardware

The hardware of the DAQ system is a network consisting of hundreds of processors, as shown schematically in Figure 1. A significant part of this network is located off shore. The off-shore processors are integrated in custom made electronics. The off-shore part of the DAQ system is connected to the on-shore part by a single electro-optical cable. The on-shore processors are standard PCs and are located in the shore station.

2.1 Off-shore DAQ hardware

Most of the off-shore electronics modules are used to read out the photo-multiplier tubes (PMTs). These modules are indicated in Figure 1 by 'sector module' and 'storey module'. A detector string is divided into 25 storeys, each consisting of a triplet of PMTs and a titanium container. This container houses

* Corresponding author
Email address: mieke.bouwhuis@nikhef.nl (M.C. Bouwhuis).
[1] Deceased
[2] Now at: y

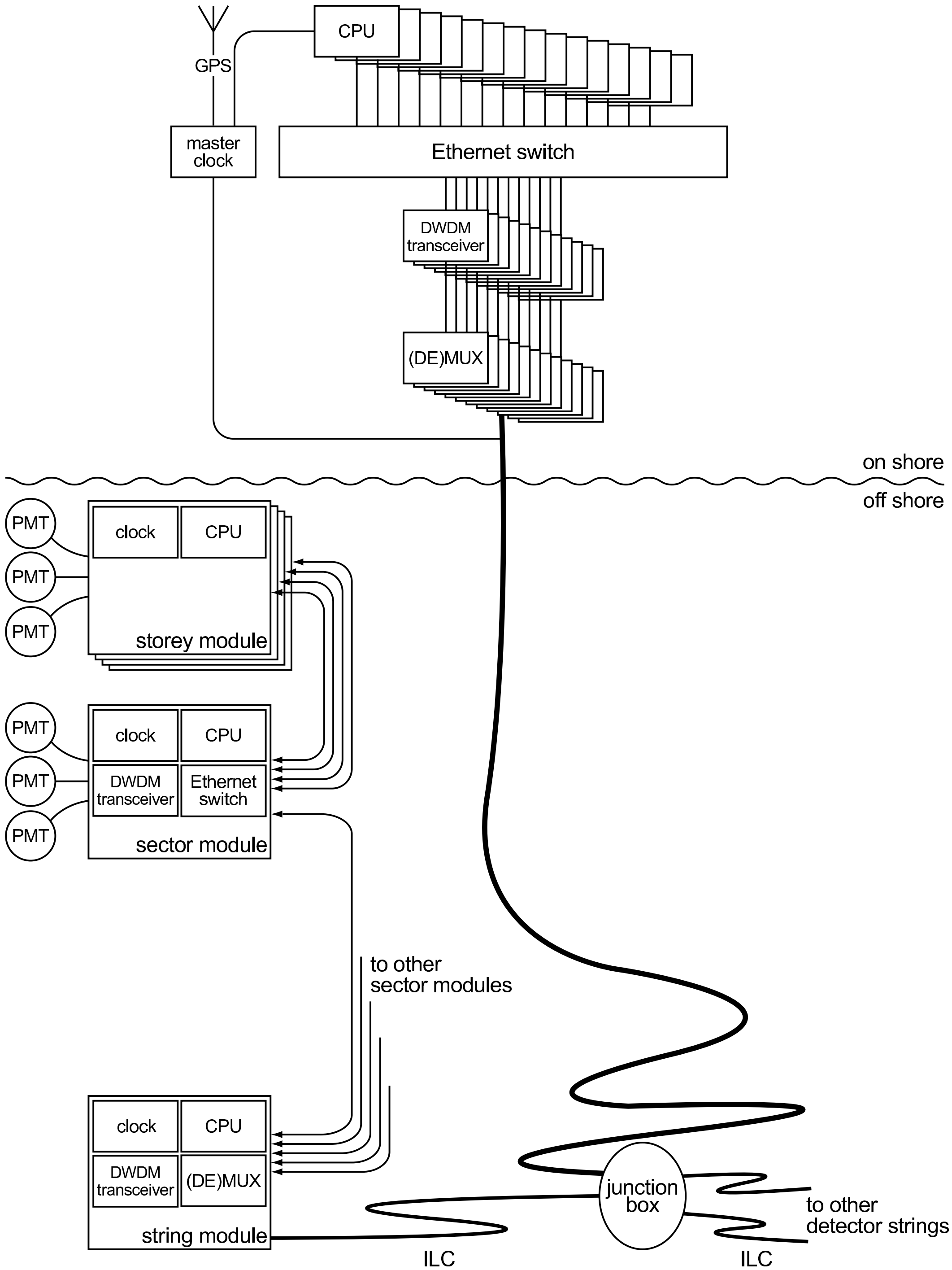

Fig. 1. *Schematic picture of the DAQ hardware. The detector consists of 12 vertical strings, each consisting of 25 storeys. The structure of only one detector string is shown. Each storey contains three PMTs, and an electronics module with a local clock and a CPU. The module on every fifth storey (sector module) contains in addition an Ethernet switch and a DWDM transceiver (see Section 2.1). Each detector string has at the bottom a string module that contains an optical (de)multiplexer ((DE)MUX). Each detector string is connected to the junction box with an interconnecting link cable (ILC). From the junction box a single cable leads to the on-shore PC farm and the master clock. The storey, sector, and string module are in ANTARES also referred to as LCM, MLCM, and SCM respectively.*

the electronics needed for the digitisation of the analogue signals from the PMTs, and the data transport. The hardware components that play a key role in the DAQ system are shown schematically in Figure 2. These components include several custom designed analogue ring sampler (ARS) chips [4], a local clock, a field programmable gate array (FPGA), an SDRAM of 64 MB, and a processor with an Ethernet port.

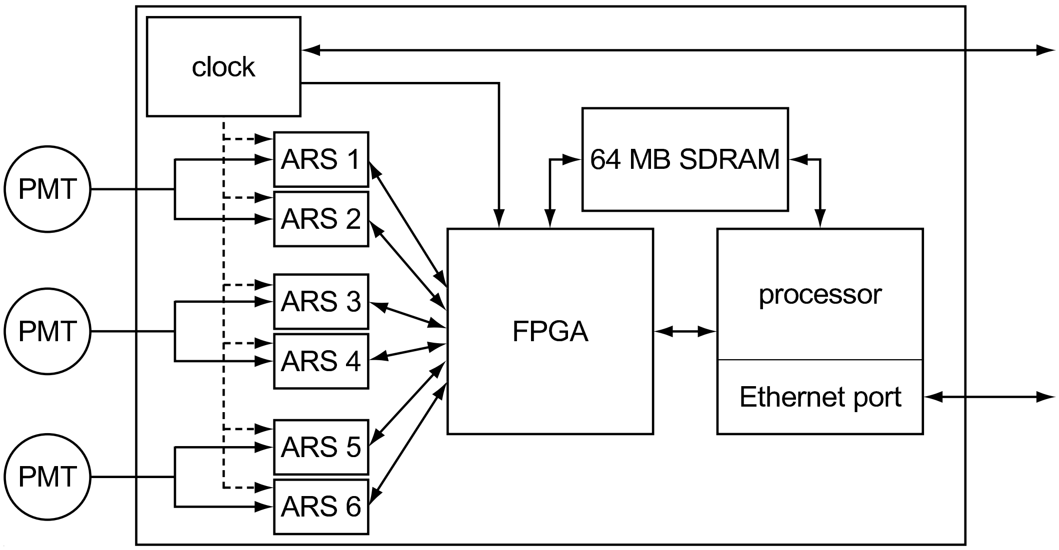

Fig. 2. *Schematic picture of the main hardware components in the electronics module of a storey.*

During data taking the analogue signals from the PMTs are processed by the ARS chips. This includes the digitisation of the timing of each PMT signal and the total charge of the pulse. The settings of each ARS chip can be adjusted, including the two main parameters: threshold and integration gate. The voltage threshold is set to eliminate small pulses due to the dark current in the PMT. Its typical value corresponds to 0.3 photo-electrons. The gate is set to integrate most of the PMT signal, while limiting the contribution of electronic noise. The typical width of the integration gate is 35 ns. A local clock is used by the ARS chips to timestamp each PMT signal above threshold. The combined time and charge information of the PMT signal within the integration time is referred to as a single photo-electron hit. After the integration time, the ARS chip has a dead time of about 200 ns due to the limited transfer speed to the analogue pipeline. To compensate for this dead time, two ARS chips are connected to each PMT. These ARS chips operate in a token ring scheme. After the integration time of the first ARS chip, the second takes over. Only after the integration time of the second ARS chip, the dead time of the first plays a role.

For special cases, like large or double pulses, the pulse shape can be read out by the ARS chip with a sampling frequency that is tunable between 150 MHz and 1 GHz. This feature is particularly useful for calibration and tuning of detector parameters, but can also be enabled for physics triggers. The pulse shape information is not used in the data filter algorithms (see Section 3.4), but it can be stored on disk with the associated hit information.

The readout system of the ARS chips is implemented in a high density FPGA XilinX Virtex-E XCV1000E. The data from the ARS chips are buffered by the FPGA in the 64 MB SDRAM into separate frames covering a certain period. This period can be set to values between 10 and 100 ms, and corresponds to a predefined number of clock cycles. The clock system is described in more detail in Section 2.3.

The processor in the off-shore electronics modules is the interface between the ARS chips and the online data processing system. It is the Motorola MPC860P, which has a fast Ethernet port (100 Mb/s). The operating system on these processors is VxWorks[3]. TCP/IP is used for the data transport.

The module on every fifth storey on a detector string, in Figure 1 specified by 'sector module', has some additional functionality. These modules contain an Ethernet switch, and the transceiver that plays a role in the data transport to and from shore. The Ethernet port on each storey is connected to a sector module using an optical bidirectional 100 Mb/s link. The links from five storeys are merged by the Ethernet switch in the sector module into a single Gb/s Ethernet link. The group of storeys with a common Gb/s Ethernet link is referred to as a sector. For a total of 25 storeys on a detector string, each detector string has five such sectors.

Each detector string has at the bottom an extra container which houses an electronics module, in Figure 1 specified by 'string module'. The string module is used for the slow control of the electrical power system and the calibration systems of the detector string, and also for the distribution of the clock signal. This module has a separate 100 Mb/s link to shore.

For the data transport between the off-shore and on-shore parts of the detector the dense wavelength division multiplexing (DWDM) technique is used [5]. This is an optical technology that uses multiple wavelengths to transmit different streams of data along a single fibre. The DWDM system can be considered as a large fibre-optic network. The DWDM transceivers are used to extend the Ethernet link to shore. The five DWDM channels from (to) the sectors and the DWDM channel from (to) the string module are optically (de)multiplexed in the string module. Each sector and each string module has a unique pair of wavelengths that is used to transmit the data.

[3] http://www.windriver.com/

Each string is connected with an electro-optical cable to the so-called junction box, the container where the cables from the 12 detector strings meet. The junction box is connected to the shore station with a single 40 km long electro-optical cable. On shore, there is a multiplexer and a demultiplexer for each detector string. The DWDM system is also used for the transmission of the slow control data, and the distribution of initialisation and configuration data. The data that are sent from the shore station to the detector are multiplexed on shore, and demultiplexed in the string modules.

2.2 On-shore DAQ hardware

The on-shore part of the DAQ system is located in the shore station. It consists of a farm of standard PCs, a large Ethernet switch, the DWDM hardware, and the master clock system (described in Section 2.3).

All PCs in the farm have software programs running that are needed for the detector operation, data transfer and communication, data processing, monitoring, and data storage. These programs are all part of the DAQ software, discussed in Section 3. The PC farm consists of some tens of PCs, all connected to the Ethernet switch. With 12 detector strings, each consisting of 25 storeys and an electronics module at the bottom, there are 312 off-shore processors, each with an Ethernet port for the connection to shore. All on-shore and off-shore processors that are part of the DAQ system are connected to the same on-shore Ethernet switch. Together they form a large Ethernet network. Each of these processors can be considered as a node in the DAQ network, addressable by its unique IP address. All off-shore processors can be reached by a telnet connection. This allows system tests to be performed and new software and firmware to be downloaded. The Ethernet network makes the distribution of the data from the ARS chips to the on-shore processes completely transparent, and it enables the communication with all processors in the system.

Like in every string module, a (de)multiplexer and DWDM transceiver is available on shore for each detector string. The multiplexers are used for multiplexing the data streams that are meant for the initialisation and configuration of the detector, and the demultiplexers are used for demultiplexing the data streams from the ARS chips, and slow control data for monitoring purposes.

2.3 Clock system

The main purpose of the clock system is to provide a common clock signal to all ARS chips. It consists of a clock signal generator on shore, a clock signal distribution system, and a clock signal transceiver in each detector storey. The on-shore master clock generates a 20 MHz clock signal that is synchronised internally to the GPS time with an absolute accuracy of 100 ns. The clock signal is distributed to all off-shore clock transceivers using the available optical fibre network. Therefore each detector storey has one local clock that is synchronised to the master clock. The phase offset of each local clock can be determined by measuring the return time of a calibration signal. The local clock is used to synchronise the ARS chips. The relative time accuracy is about 50 ps. The clock system operates in parallel and independent of the DAQ system. As a result, the internal clock calibration does not induce any dead time.

3 DAQ software

The main software processes in the DAQ system are indicated schematically in Figure 3. Part of the processes run on the off-shore processors, part of them run on the on-shore processors. There are basically three types of pro-

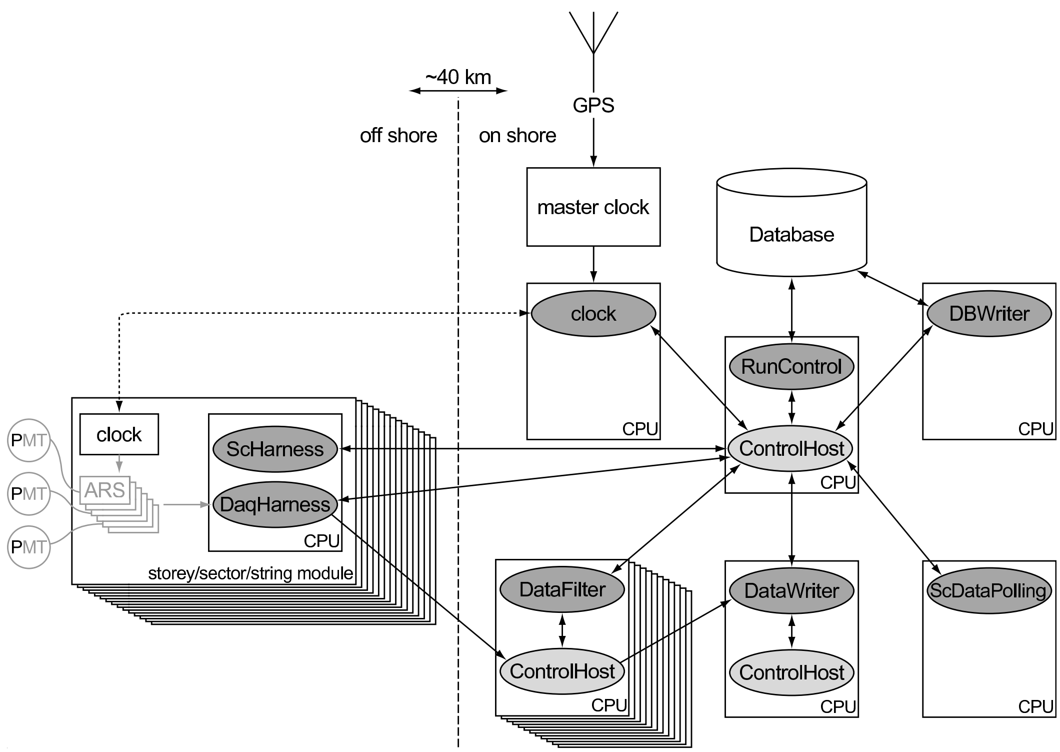

Fig. 3. *Schematic overview of the main processes in the DAQ system. The solid lines indicate the links between processes, for which TCP/IP is used. The dotted line indicates the distribution of the reference clock signal. The processors in each off-shore electronics module, whether this module is connected to PMTs or not, has the same processes running. See text for an explanation of the processes.*

cesses: processes that are used for the data transfer and communication (see Section 3.2), processes that take care of the operation of the detector (see Section 3.3), and processes that are involved in the data taking and data handling (see Sections 3.3–3.5). In total, with hundreds of off-shore processors and the on-shore PC farm, there will be several hundreds of processes in the DAQ system. A DAQ model, that is implemented in each process, has been defined to synchronise all processes.

3.1 DAQ model

The hundreds of processes in the distributed multiprocessor system operate independently. In order to synchronise all processes in the system, a finite state machine has been designed with a fixed set of states and transitions, represented by the state diagram shown in Figure 4. Each of the processes that participates in the DAQ system, except for the data transfer package ControlHost (see Section 3.2), has this state machine implemented, for which the CHSM language [6] is used. The state transitions are accompanied by actions that are specific to each type of process in the system. These actions should be performed before the data taking can start and after the data taking has stopped. The state machine ensures that these actions are executed for all processes in the required order (as specified by the state transitions).

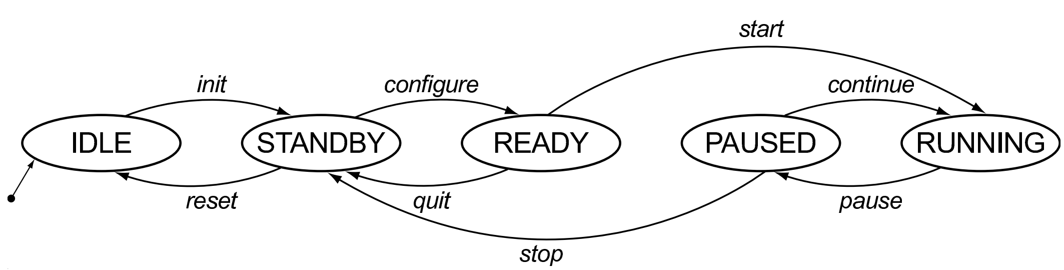

Fig. 4. *The finite state machine of the DAQ system. The ellipses indicate the different states, the arrows indicate the state transitions. When the state machine is entered it starts in the* IDLE *state.*

The steps involved in the preparation of the DAQ system for data taking include the configuration of the hardware components and software processes. The required steps for stopping the data taking (and switching off the detector) include the handling of the data that are still being processed, closing the data files, and storing the run information in the database. Data are only taken in state RUNNING. Each start transition corresponds to a new data taking run, identified by a unique run number. The run number is added to the raw data by the off-shore processes, and is also used by the data storage program to archive the data.

Processes in the DAQ system can only be in one of the predefined states. State transitions are initiated from a central application (see Section 3.3), and apply to all processes in the system. During a state transition, all processes perform the necessary actions independently. Consequently, the actual state of the system is associated with all actions that took place during the state transitions. Normally all processes are in the same state at any time.

3.2 Data transfer and communication

All data transfer and communication between processes in the DAQ system are done with the ControlHost package [7]. ControlHost implements the tagged data concept, which means that any program in the system can send data accompanied by a tag to a ControlHost server. The server will distribute the data to all processes that subscribed to the specific tag using TCP sockets. Most of the PCs in the shore station have a ControlHost server running. The ControlHost package has been optimised for Gb/s Ethernet, and its overhead is typically 5% of the total bandwidth.

ControlHost is used for the data transfer between processes at three different stages in the DAQ: data processing, data storage, and detector operation. The data from the off-shore processes (DaqHarness) are transferred to the on-shore processes that take care of the processing of the data (DataFilter, see Section 3.4). On each PC in the shore station that has such a data processing program running, a ControlHost server is running for passing the data. All data that passed the data processing, and that have to be stored on disk, are transferred from the data processing PCs via one ControlHost server to the data storage program (DataWriter, see Section 3.5). The RunControl program, that is used to operate the detector (see Section 3.3), communicates with all processes in the DAQ system. All state transitions are initiated by the RunControl. The commands for the state transitions, as well as the messages for the handshaking mechanism are distributed via a central ControlHost server.

ControlHost takes care of the distribution of a single message to multiple processes on multiple hosts. Without knowing how many processes should receive a certain message, each will receive the message if it subscribed to the corresponding tag. The implementation of ControlHost also enables the exchange of information between processes written in different programming languages (C^{++} and Java™ are used in the DAQ system). The RunControl program is written in Java™ because of the straightforward possibility for remote operation of the detector, the user friendly interface with the Oracle® database that is used in the DAQ system, and the advanced toolkit for developing graphical user interfaces. The on-shore programs used for data processing and data handling are written in C^{++} because of the required execution speed. The required compatibility of the off-shore software with the related firmware also implies the use of the C^{++} programming language.

3.3 Detector operation

The main program in the DAQ system that is used to operate the detector is the RunControl. It has a graphical user interface from which the operator submits the requests for the execution of state transitions by all processes in the system (shown in Figure 4). All processes that are subjected to these state transitions can be considered as client processes. With these state transitions the detector can be prepared for data taking, and the necessary actions are performed when the data taking has ended.

All data needed by the DAQ system to operate the detector are stored in the central database system, for which the relational database system Oracle® is used. These data are retrieved from the database by the RunControl only. These data include the detector information (the location of each component in the detector), the client process list, the different configurations for each client setup, the initialisation and configuration data, and the information about the data taking. The client process list determines which processes participate in the system, and thus which parts of the detector are active during data taking. The RunControl retrieves the necessary information from the database by using the input from the operator and the relations between the database tables.

A separate graphical user interface is used to control the power system in each string module. When the off-shore electronics modules are powered on, the off-shore processors boot. The RunControl launches each client process that is part of the client setup chosen by the operator by executing its start command on the host with its associated IP address (both retrieved from the database). At startup the client processes enter the general state machine.

Once the client processes are started, the general mechanism that is used for the communication between the RunControl and its clients is done with a handshaking method based on the unique ID of each process. This ID consists of the nickname of the process (which corresponds to the client type), and the IP address of its host. The implementation of the handshaking method is that, after the request for a state change from the RunControl via the central ControlHost server, each client sends a message containing this unique ID, and the state it just entered, to the same ControlHost server that indicates that the state change has succeeded for this client. As the RunControl uses the same unique coding for each client, it is aware of the current state of each client. As a consequence, the RunControl also has a monitoring task, visualising the state of each client. The ControlHost package is designed in such a way that it broadcasts a message when a process connects to or disconnects from a ControlHost server. These messages are also intercepted by the RunControl. As each client connects to the ControlHost server when it is started, and disconnects when it terminates, the RunControl also visualises the running state of each client. In addition, the RunControl provides the means to modify the client setup during operation in case of a client error to prevent interruption of the data taking.

Depending on the type of data that will be taken, a predefined configuration for the chosen client setup is selected by the operator. The specific initialisation and configuration data required by each process are retrieved from the database by the RunControl, and sent, together with the corresponding state transition request, via the ControlHost server to the particular client process. Along with the state transitions for starting and stopping a run, the master clock on shore sends a hardware signal to the off-shore clocks for the synchronisation of the ARS chips.

The ScHarness process that runs on each off-shore processor controls the high voltage applied to the PMTs during the initialisation and configuration phase. It is also used for the readout of various instruments. The data from these instruments are sent to shore, and stored in the database by the DBWriter program. These data are then used to monitor the detector and its environment. The ScDataPolling program schedules the read requests of all these instruments in the detector.

The RunControl updates the database with the run information and the reference to the detector settings to be able to couple the detector settings to the data when they are eventually analysed. A new run is automatically started after a few hours of data taking, or when the output data file, created and updated by the data storage program (see Section 3.5), reaches its maximum size (typically 1 GB). The RunControl then triggers the necessary state transitions automatically to stop the run, and start a new run.

During data taking, the raw data are monitored online using dedicated software. For this, the data are taken from ControlHost and histogrammed using the ROOT package [8]. A custom made presentation tool based on ROOT displays and compares histograms, showing the basic measurements such as the time and amplitude information from the PMTs.

During data taking, all signals that are recorded by the PMTs and digitised off shore are transported to shore without any off-shore data selection. This implementation is known as the ‘all-data-to-shore’ concept. As a result, all raw data are available on shore where the required data processing method can be applied to the data. This minimises the overall loss of physics signal, and allows different processing methods to be applied for specific physics analyses.

The on-shore handling of all raw data is the main challenge in the ANTARES DAQ system, because of the high background rates, mainly caused by bioluminescence and ^{40}K decay. As a result of this background, the average photon detection rate per PMT varies between 60 kHz and 90 kHz during periods with low bioluminescence activity [9]. The data output rate of the detector is then 0.3 GB/s to 0.5 GB/s. The on-shore handling of the data is done with the DataFilter processes that run on each PC in the on-shore data processing farm. The processing of the data involves the separation of signal from background, and as a result the necessary reduction of the data flow for storage. During data taking, the DataFilter programs process all data online as the detector is in principle operating continuously. The DataFilter programs apply various algorithms, each aimed at the search for specific physics signals. This includes a muon filter that looks in all directions for the signals in the detector compatible with a muon. Depending on the random background rate, the filter settings can be adjusted so that the trigger rate is dominated by muons passing the sensitive volume of the detector. For an average data size of a physics signal of 2.5 kB, and an expected atmospheric muon rate of about 10 Hz, the data reduction is then about 10^5. The bandwidth required for the data transport scales linearly with the background photon detection rate. The data processing speed scales with some power of this rate, this power being the required minimum number of photons to identify a physics signal (see Section 4).

The DataFilter processes receive the raw data from the off-shore DaqHarness processes, running on each off-shore processor in the detector. All data that are produced by each ARS chip in a certain time window are buffered in the SDRAM by the FPGA in a so-called frame. The length of this time window can be set to values between 10 and 100 ms. A separate thread that is spawned by the DaqHarness process takes the buffered data from the SDRAM, and sends each frame as a single packet to shore using TCP/IP (see Figure 5). The frames from all DaqHarness processes that belong to the same time window are sent to the same PC in the on-shore data processing system, identified by its IP address. The large Ethernet switch thus operates as a so-called level 2 switch. The list of IP addresses as possible destinations for the raw data is provided to the DaqHarness processes by the RunControl. The collection of

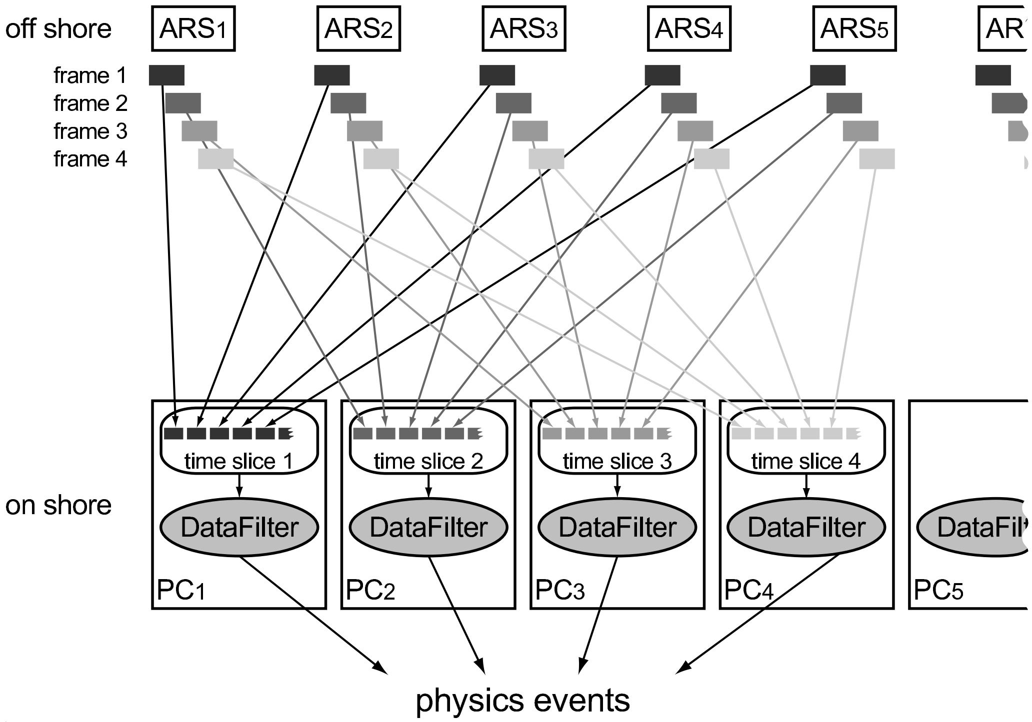

Fig. 5. *The processing of the data based on time slices. All frames belonging to the same time window are sent to a single PC and form a time slice. The DataFilter program running on each PC processes the data in the time slice. All physics events are stored on disk (see Section 3.5).*

frames belonging to the same time window is called a time slice. As a result, a time slice contains all data that were registered during the same time interval by all ARS chips in the detector. The DataFilter programs alternately receive the frames belonging to the same time window from the off-shore processes via the ControlHost server that runs on the same PC. Each DataFilter program has to be finished with processing a time slice before it receives the next. This imposes an optimisation of the configuration of the DataFilter programs in terms of processing speed, and it determines the number of PCs required for online data processing and the specifications of these PCs.

When a sufficient number of correlated single photo-electron (SPE) hits is found that is consistent with a specific physics signal, the data are considered as a physics event, that is written to disk. The period covered by a time slice is chosen such that its duration is long compared to the duration of a physics event in the detector (approximately 1 μs), in order to minimise the chance of having an event crossing the boundaries of a time slice.

The algorithms implemented in the data processing software are designed to find a physics signal by looking for space-time correlations in the data. From the time of the SPE hits, and the positions of the PMTs, these algorithms calculate in real time if hits could originate from light produced by this signal. In the case of standard data processing, when the muon filter is used, this is the signal produced by a muon traversing the detector. Apart from the standard muon filter there are various other algorithms implemented that are used for the search for specific physics signals. The configuration of the DataFilter programs, that indicates which algorithm is used, is done by the RunControl, as described in Section 3.3. Among the presently available filter algorithms is a magnetic monopole filter, an optical beacon filter, and a directional filter. The monopole filter looks for space-time correlations in the data that are caused by particles that traverse the detector at lower speeds than a muon. The optical beacon filter is used when the optical beacons, that are attached to the detector strings, are fired. These optical beacons produce a typical light signal, that is used for time calibration. The directional filter is used when searching for a muon coming from a specific direction. This filter method takes into account the known direction of the neutrinos, and is used for astrophysical sources with known positions on the sky. For gamma-ray bursts, a subgroup of these sources, this filter method can be used offline. The combination of the all-data-to-shore concept, the use of an on-shore PC farm, the specific features of gamma-ray bursts, and the information provided by external satellite triggers, make the ANTARES detector particularly sensitive to neutrinos from gamma-ray bursts [10].

Any other possible filter algorithm for a specific physics signal can easily be implemented in the data processing system. As the whole data processing system is implemented in software, a new filter algorithm only involves adding the specific code and the corresponding configuration option in the database. If necessary, the computing power can be increased by adding PCs to the on-shore system.

3.5 Data storage

The physics events selected by the DataFilter programs are passed via one ControlHost server to the DataWriter, which formats the events and writes them to disk in ROOT format [8]. These disks are located in the shore station. For each run, the data are stored in a separate file. The event files are used for the physics analyses, and are copied regularly from the shore station to a computer centre elsewhere.

4 Performance of the DAQ system

The performance of the DAQ system has been evaluated on the basis of the operation of a prototype detector [9], and has been quantified using a detailed simulation of the response of the complete detector to muons [11]. During periods of low bioluminescence activity (when the average photon detection rate is 60–90 kHz per PMT) all data are transmitted from the off-shore electronics modules to the shore station. The total data output rate of the complete detector is then 0.3–0.5 GB/s. However, at higher background rates congestion of the data flow can occur at the processors in the off-shore storey modules. Although these off-shore processors have a 100 Mb/s Ethernet port, they cannot make use of the full available bandwidth. The maximum data throughput of a single storey is at present 20 Mb/s. This corresponds to a photon detection rate of about 150 kHz per PMT. With the zero-copy feature of the VxWorks operating system running on the off-shore processors, this maximum data flow could be increased to about 40 Mb/s. This corresponds to a maximum manageable photon detection rate of about 300 kHz per PMT.

The sector modules have a Gb/s Ethernet link to shore which can easily accommodate the maximum rate. On shore, there is a single large Ethernet switch to which all data processing PCs, each with a Gb/s Ethernet link, are connected. To avoid congestion of the data flow at the data processing PCs, a barrel shifting mechanism of the data has been implemented in the off-shore DaqHarness processes. In this mechanism, the data sending threads do not send the data for a given time slice simultaneously to the same on-shore PC, but desynchronise the data sending according to a predefined scheme.

The filter programs that run on the data processing PCs have to filter the data in real time. The number of required filter programs, which corresponds to the number of data processing PCs involved in the DAQ system, depends mainly on the background rate in the sea. The standard muon filter is configured such that it triggers when it finds 5 or more space-time correlated level 1 hits (two hits within 20 ns in one detector storey, or one hit with an amplitude larger than 2.5 photo-electrons). Therefore a minimum of 10 detected photons is required to trigger an event. The trigger efficiency of this filter has been evaluated using a simulation of the detector response to muons [11], and the result is shown in Figure 6 as a function of the number of detected photons for a given event. The efficiency rises rapidly above 10 detected photons, and reaches 1 at 40 detected photons.

During periods of low bioluminescence activity (see above), and using a trigger based on a majority logic, the trigger rate due to random background is a few hundred Hz. Simulations have shown that with a more sophisticated trigger, this rate can be reduced to less than 1 Hz, without loss of efficiency. With an expected atmospheric muon trigger rate of about 10 Hz, the standard muon filter then has a purity of 90%. With the filter conditions mentioned above, 10 PCs with a processing speed of 3 GHz are needed. Different filter configurations are used for the various physics filters described in Section 3.4. For physics filters that require more computing power, or that use less strict filter

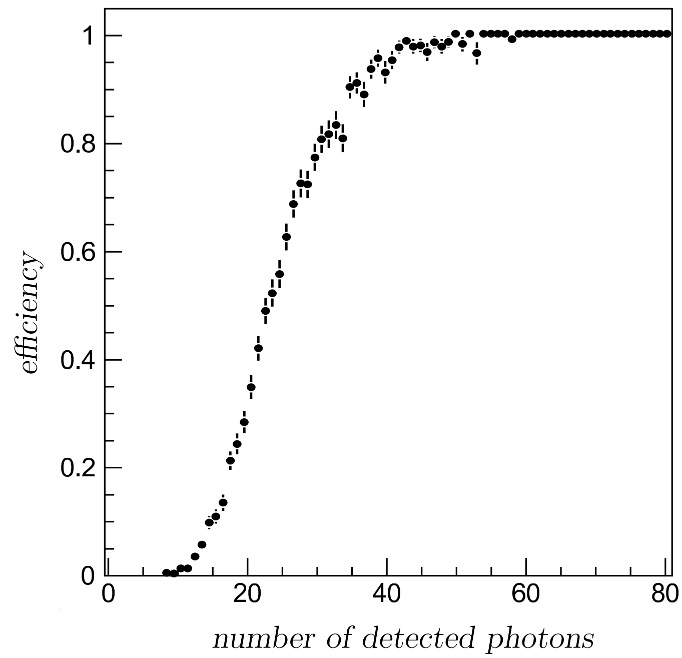

Fig. 6. *Efficiency of the standard muon filter as a function of the number of detected photons for a given event, using a simulation of the detector response to muons. The efficiency is defined as the ratio between the number of events found by the filter and the number of generated events.*

conditions (as used for example in the directional filter), more data processing PCs are used to be able to filter the data online.

In principle the data taking and online data processing take place continuously. There is no dead time in the data transport, nor in the online filtering of the data. Only a small dead time is encountered in the ARS chips, such that a third photon is lost when it is detected by one PMT within 235 ns (i.e. integration gate and dead time of the ARS chip). The effective dead time is then less than 165 ns, taking into account the integration gate of both ARS chips. However, at high background photon detection rates, the maximum bandwidth of the off-shore processors is reached (see above). Part of the data is then lost, and this will result in a loss of the overall detection efficiency. With the prototype detectors and the first complete detector string that have been operational, significantly higher background rates than the maximum manageable background rate have been measured, especially during spring.

5 Conclusions

The ANTARES DAQ system has been operational since 2003, and has been used for various systems that have been deployed in the deep-sea environment. The all-data-to-shore concept offers a flexible data filtering system. A simulation of the detector response to muons has shown that a high detection efficiency can be achieved. The results obtained with one of the prototype detector strings have been published [9]. For this detector, the data taking was more than 90% efficient during periods of low bioluminescence activity. The DAQ system has also been used to read out the first complete detector string. By enlarging the on-shore computer farm, the same system will be used to read out the complete detector, that will consist of twelve detector strings.

Acknowledgements

The authors would like to thank R. Gurin and A. Maslennikov for providing the ControlHost package. The authors also acknowledge the financial support of the funding agencies: Centre National de la Recherche Scientifique (CNRS), Commissariat à l'Energie Atomique (CEA), Commission Européenne (FEDER fund), Région Alsace (contrat CPER), Région Provence-Alpes-Côte d'Azur, Département du Var and Ville de La Seyne-sur-Mer, in France; Bundesministerium für Bildung und Forschung (BBF), in Germany; Istituto Nazionale di Fisica Nucleare (INFN), in Italy; Stichting voor Fundamenteel Onderzoek der Materie (FOM), Nederlandse organisatie voor Wetenschappelijk Onderzoek (NWO), in The Netherlands; Russian Foundation for Basic Research (RFBR), in Russia; Ministerio de Educación y Ciencia (MEC), in Spain.

References

[1] E. Aslanides et al., the ANTARES proposal, astro-ph/9907432

[2] P. Amram et al., Nucl. Instrum. Meth. A484 (2002) 369

[3] J.A. Aguilar et al., Nucl. Instrum. Meth. A555 (2005) 132

[4] F. Feinstein et al., Nucl. Instrum. Meth. A504 (2003) 258

[5] J.M. Senior, Optical Fiber Communications: Principles and Practice, Prentice Hall, 1992

[6] P.J. Lucas, An Object-Oriented Language System for Implementing Concurrent, Hierarchical, Finite State Machines, M.S. Thesis, Technical Report: UIUCDCS-R-94-1868, University of Illinois at Urbana-Champaign, Department of Computer Science, 1993

[7] R. Gurin and A. Maslennikov, ControlHost: Distributed Data Handling Package, 1995

[8] R. Brun and F. Rademakers, Nucl. Instrum. Meth. A389 (1997) 81

[9] J.A. Aguilar et al., First results of the Instrumentation Line for the deep-sea ANTARES neutrino telescope, astro-ph/0606229, accepted for publication in Astroparticle Physics

[10] M.C. Bouwhuis, Detection of neutrinos from gamma-ray bursts, PhD Thesis, University of Amsterdam, 2005

[11] D.J.L. Bailey, Monte Carlo tools and analysis methods for understanding the ANTARES experiment and predicting its sensitivity to Dark Matter, PhD Thesis, University of Oxford, 2002