Standard-based Middleware Platform for Medical Sensor Networks and u-Health

I. Martínez, J. Escayola, M. Martínez-Espronceda,
L. Serrano *Senior Member, IEEE*, J. Trigo, S. Led, and J. García *Member, IEEE*

***Abstract*—Advances in Information and Communication Technologies, ICT, are bringing new opportunities in the field of middleware systems oriented to ubiquitous environments and wearable devices used for patient telemonitoring. At a time of such challenges, this paper arises from the need to identify robust technical telemonitoring solutions that are both open and interoperable in home or mobile scenarios. These middleware systems demand standardized solutions to be cost effective and to take advantage of standardized operation and interoperability. Thus, a fundamental challenge is to design a plug-&-play platform that, either as individual elements or as components, can be incorporated in a simple way into different telecare systems, perhaps configuring a personal user network. Moreover, there is an increasing market pressure from companies not traditionally involved in medical markets, asking for a standard for Personal Health Devices (PHD), which foresee a vast demand for telemonitoring, wellness, Ambient Assisted Living (AAL) and applications for ubiquitous-Health (u-Health). However, the newly emerging situations imply very strict requirements for the protocols involved in the communication. The ISO/IEEE11073 (X73) family of standards is adapting to new personal devices, implementing high quality sensors, and supporting wireless transport (e.g. Bluetooth) and the access to faster and reliable communication network resources. Its optimized version (X73-PHD) is adequate for this new technology snapshot and might appear the best-positioned international standards to reach this goal. This work presents an updated survey of this standard and its implementation in a middleware telemonitoring platform.**

***Index Terms*—Middleware platform, communication protocol, agent-manager model, wearable devices, ISO/IEEE11073 (X73) standards, ubiquitous-Health (u-Health).**

I. Introduction

PATIENT telemonitoring is one of the most common practices in telemedicine in both indoor and outdoor scenarios, and it is hoped that it can increase the quality of the care and the efficiency of services provided. In fact, it should facilitate a continuous or event monitoring of chronic, elderly, under palliative care or have undergone surgery, without them occupying the beds that would be necessary for monitoring in-situ (leaving the beds for the use of patients in a more critical condition). In addition, telemonitored patients can continue to live in their own homes with the subsequent advantages: comfort, more favorable environment, less need for trips to the hospital, etc. Telemonitoring, used appropriately, is expected to decrease healthcare costs.

The communications and interfaces among components of patient monitoring systems and between these systems, become now very important in exploiting all the possibilities offered by the information gathered [1]-[3]. However, different manufacturers use their own software and communication protocols: building proprietary solutions that can only work alone or inside a single-vendor system. As each device speaks a different language, an interoperability problem emerges, leading to difficulties when a part of a system must be replaced as well as high costs [4]. Furthermore, the information acquired cannot be easily integrated into and exchanged with the electronic healthcare record (EHR).

There is a need for developing open sensors and middleware components that allow transparent integration and plug-and-play interoperability of monitoring devices and systems (see Fig. 1). The use of communications standards seems to be an efficient way to solve these problems.

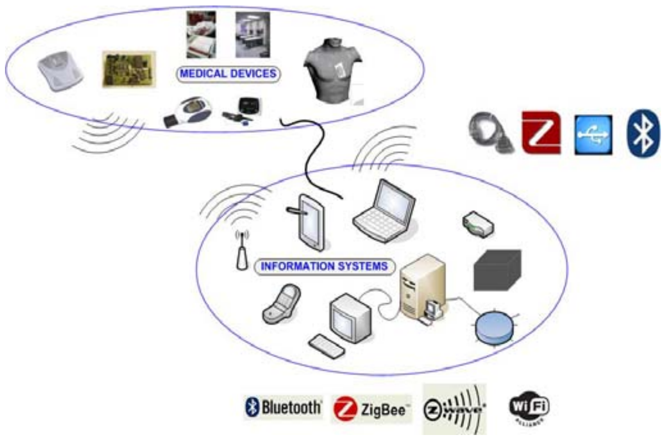

Fig. 1. Middleware need: medical devices interoperability.

Manuscript received April 15, 2008. This research work has been partially supported by projects TSI2007-65219-C02-01 and TSI2005-07068-C02-01 from *Comisión Interministerial de Ciencia y Tecnología* (CICYT) and European Regional Development Fund (ERDF), PET2006-0579 from *Programa de Estímulo de Transferencia de Resultados de Investigación* (PETRI), and FPI grant to M.Martínez-Espronceda (Res. 1342/2006 from Public University of Navarre).

I. Martínez, J. Escayola, J. Trigo, and J. García are with the Communications Technologies Group (GTC), Aragon Institute for Engineering Research (I3A), University of Zaragoza (UZ), c/ María de Luna, 3, 50018 Zaragoza, Spain (e-mail: imr@unizar.es). M. Martínez-Espronceda, L. Serrano, and S. Led are with the Electrical and Electronics Engineering Dep., Public University of Navarra, Campus de Arrosadía s/n, 31006 Pamplona, Spain (e-mail: miguel.martinezdeespronceda@unavarra.es).

As it is shown in Fig. 1, the devices used most frequently in telemedicine applications to measure parameters and biological signals are glucose meters, blood pressure and heart rate meters, pulse-oximeters, ECG monitors, digital scales, etc. Moreover, in last years, it is desirable that non-patient oriented devices that form part of a spectrum of use from fitness and wellness monitoring, though devices in support of both independent and assisted living and into self-managed informal monitoring, are also capable of playing a part in such an interoperable continuum of care. As the paradigms for health management change, in the face of societal and economic pressures, this continuity and flexibility will become increasingly important. Thus, it is common for medical devices to be wireless or wearable (with sensors incorporated into clothing, bracelets, etc.), that makes their use more comfortable. These collections of sensors around the patient make up what can be usually described as either a Body/Personal Area Network (BAN/PAN). Often, for monitoring elderly patients or those with limited mobility, these PAN or BAN networks are completed with presence detectors, movement sensors, or similar telecare devices, which combine to form a Home Area Network (HAN).

The challenge of having telemonitoring systems that can interoperate and communicate with a middleware platform based on an open standard is complicated, somehow, because of the features of the devices that are usually implied. Moreover, for these telemonitoring devices, the physical way of transmission is not always the same; it is possible to be wired or wireless: (e.g. Bluetooth, Zigbee, Wibree, USB, etc.). Furthermore, they coexist with other medical devices and network devices such as PCs, routers, modems, mobile phones, etc. that are using different technologies. Then a modular layer design of the standard should have specializations for different low layer communications that can be used.

To place the standard in context, we summarize other standards in the field of healthcare information systems oriented towards the encoding of signals and biomedical parameters, the standardization of the electronic healthcare record, or the communication between medical applications using standardized messages. Some of these standards are: POCT-1A2 (communication protocols between the device and an access point [5]), Health Level 7 (HL7, for the exchange, management and integration of electronic EHR information [6]), DICOM-Digital Imaging and Communications in Medicine [7], and EN13606 (for EHR communication [8]).

In this paper we propose a middleware platform for ubiquitous patient telemonitoring based on standards. Section II provides a starting point and survey of ISO/IEEE11073 as the best-positioned standard for Plug and Play interoperability of Personal Health Devices (PHD). In Section III the suggested design and its architecture is shown, and its implementation progress is detailed in Section IV. Finally, Section V evaluates the results and new functionalities oriented to X73-PHD.

II. ISO/IEEE11073 (X73) AS A MIDDLEWARE

The ISO/IEEE11073 Point-of-Care (also known as X73-PoC) [9] is an internationally harmonized family of standards, produced by a grouping of manufacturers, institutions and IEEE Institute. It consolidates previous IEEE-1073 Medical Information Bus (MIB) [10] and CEN (VITAL [11] and INTERMED [12]) standards, to cover different levels of the ISO Model, with models for access to the data and with services and communication protocols for interoperability between medical devices. They are considered European standards via the TC251 of the European Committee for Standardization (CEN) [13]. Technical Committee 251 (TC251) is responsible for health informatics and constitutes the only Europe-wide forum for consensus and standardization of computer science applied to healthcare [14]. It liaises closely with the International Standards Organization (ISO), the principal world standardization body.

It is also important to mention that a standard for medical device (MD) communications in telemonitoring scenarios can change the market and is critical for competitiveness between the different companies, manufacturers and service providers. At this point emerges Continua Alliance, which is a group of technology, healthcare and fitness companies that wish to increase compatibility of e-healthcare devices using the existing standards to create an interoperable framework. Their objectives are to design the guidelines to achieve interoperability of sensors and systems [15].

In accordance with the X73 standards [9], interoperability in the local level of monitoring devices can be solved by connecting all of them with a central element that acts as a main connection integrated Computer Engine (CE) with the telemonitoring server (see Fig. 2). This CE must control the interaction with different MDs that form the BAN/PAN/HAB network, and monitor the patient (by means of the configuration of the sending and reception of data and control information). In the same way, CE will be in charge of connecting the patient network with the telemonitoring server as a middleware access server to MDs. The greatest need for standardization arises (homogenizing the interface between MDs and CE, and if widespread use is to be achieved economically) of these connections and in the communication with the telemonitoring MDs that compose the patient network.

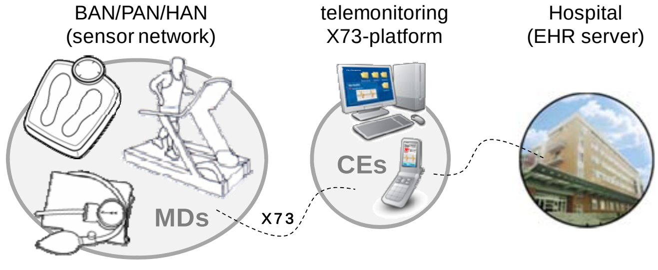

Fig. 2. Generic telemedicine integrating heterogeneous systems

In the other critical field of interoperability introduced earlier, integration of a telemedicine system into mainstream healthcare workflow and practice, the main challenge is in being able to incorporate information from perhaps disparate telemonitoring services that themselves include different vendor's MDs and CEs, managed by the telemonitoring servers; each telemedicine system being connected to the generic EHR. In this scenario, middleware technologies provide portability (a telemonitoring system can be connected to different telemedicine systems) and interoperability (medical applications in different clinical environments can exchange information between devices connected to a patient).

A common shortcoming, even when considering use of new technologies, is to overlook the importance of consistent representation of content. This has been a significant problem in the health sector with a number of attempts at achieving consistent representation of meaning having been attempted in the last 20 years or so [16]. For MD communication the problem was recognized as being of major importance when a pan-European project team started work on VITAL– is was simply not possible to correctly interpret between languages the extremely detailed terms being used. The concept of semantic links was adopted to build up language-independent means of describing these detailed concepts. This, allied to a robust information model of the domain facilitated production of a globally usable MD data language, is crucial in a global industry for both devices and health software systems.

The rigorous and extensible nature of the MD data language has been recognized and adopted to enable large databases to contain physiognomic data for research and regulatory purposes. Work is currently underway to link these detailed representations to the less detailed terms clinicians customarily use and that are represented in SNOMED CT [17]. It appears likely that only with true semantic interoperability from MD to health record will it be possible to use operational health information alongside genomic and adverse event databases for data-mining and research to improve practice.

Finally, the developments of new personal and wearable devices have brought X73 to an optimized version: X73-PHD [18]. There are previous contributions [19], developed in the USA from the research group headed by Dr. Warren, but no European contributions in this field nor proposed end-to-end solutions to cover new use cases for patient monitoring at ubiquitous environments and design-oriented to be compatible with the new standard version X73-PHD. X73-PHD describes the landscape of transport-independent applications and information profiles for personal telehealth. These profiles define data exchange/representation and terminology for communication between MDs or agents (e.g., glucometer, thermometer, weighing scale, blood pressure, etc.), and CEs or managers (e.g., cell phones, personal computers, etc.).

As shown in Fig. 3, the architecture is divided into three main levels that are detailed as follows:

- **Device Specializations**. A set of model descriptions which collects the total of objects and attributes related to the device components, like an overall system's configuration (Medical Device System, MDS), Persistent Metric (PM-Store and Segments) or Metric Specifications. New MDs are continuously being added, by developing its MDS.
- **Optimized Exchange Protocol**. The main part of the standard consists of a medical and technical terminology framework (Domain Information Model, DIM) which will be encapsulated inside the Protocol Data Unit (PDU). The first version of X73 defined this part as the Medical Device Data Language (MDDL). Next, a Service Model defines a set of messages and instructions to retrieve data from the agent based on the DIM. In addition to this, it provides a data conversion from an Abstract Syntax Notation (ASN.1) to a Transfer Syntax, using optimized Encoding Rules (ER) denoted as Medical Device ER (MDER), as well as standard Binary ER (BER) and even more effective Packet ER (PER) support. Service Elements taken from the previous X73 version for this purpose are: Remote Operation (ROSE, optimized for MDER), Association Control (ACSE) and Common Management (CMISE). The communication model describes a point-to-point connection based on manager-agent architecture through a Finite State Machine (FSM).
- **Transport Layer**. Data transmission will be held over a transport technology due to X73-PHD identifies assumptions that require direct support by this layer, allowing various transports to be implemented (X73-PoC established higher dependency between transport and upper and lower layers). Thus, transport specifications are out of the scope of X73-PHD, while other Special Interest Groups (SIG) are working towards profile definition for Bluetooth, USB, ZigBee, etc.

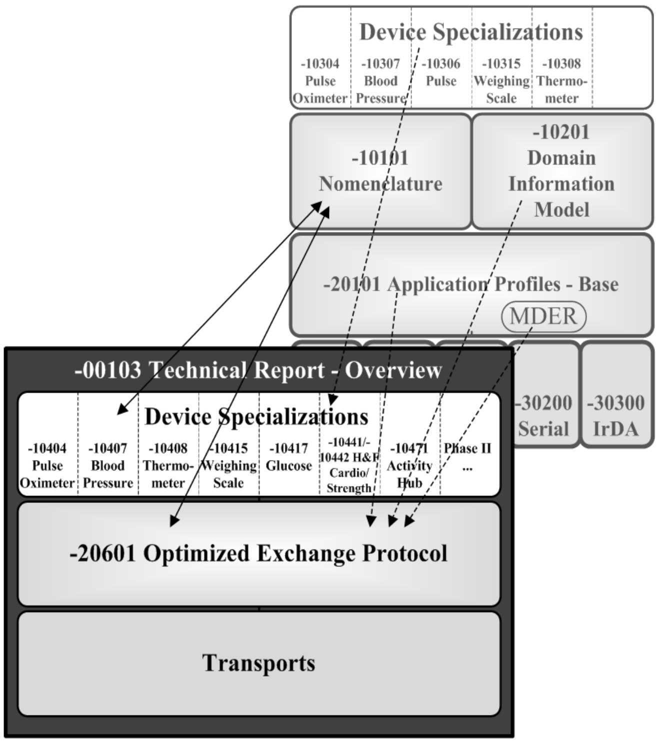

Fig. 3. ISO/IEEE11073 Personal Health Devices (X73-PHD) protocol map

III. System Architecture and Design

Following the X73-design rules, the system architecture is shown in Fig. 4. The programming languages used have been Java and C/C++, and various tools also were needed such as ANTLR 2.7, Java SDK 5.0 and ASN.1c 0.9.22 compiler. ASN.1 is a language to define standards regardless of implementation (it defines what is a "type", a "module", how to "label" a type so it can be correctly coded, etc.). BER is a set of rules to codify ASN.1 data in a sequence of octets that can be transmitted through a communications link (it defines the methods to code ASN.1 values, rules to decide the use of each method, the data format, etc.). Thus, with the ASN.1c 0.9.22 compiler, the X73 specifications for message exchange among layers are translated into BER codification and ASN.1 structures and functions (which are packed into a static ASNX library). Moreover, MDER are the proposed codification rules for X73 MDs (they define methods to transform ASN.1 in a byte stream ready for the communication and optimized for managing X73 objects). Because the ASN.1c compiler does not support the MDER transfer syntax, an *ad-hoc* translator (ANTLR-based) has been developed for working with MDER and BER. Thus, the translator generates a syntactic tree from an ASN.1 grammar that allows defining the ASN.1 structures (specified by X73) and writing the C++ code and Java utilities.

Due to the complexity of X73-communication model, the design attempts to reduce this complexity by simplifying the non-critical protocol layers, including already created data definitions and optimizing the code for memory requirement adjustments. In recent advances and to avoid programming language diversity to collapse and ease the process of improving the system, the system is implemented in C++ under Windows XP using the Microsoft Foundation Classes (MFC), which allows the further addition of new services (multiple transport technologies, database, remote access) and improvements oriented to multimedia environments.

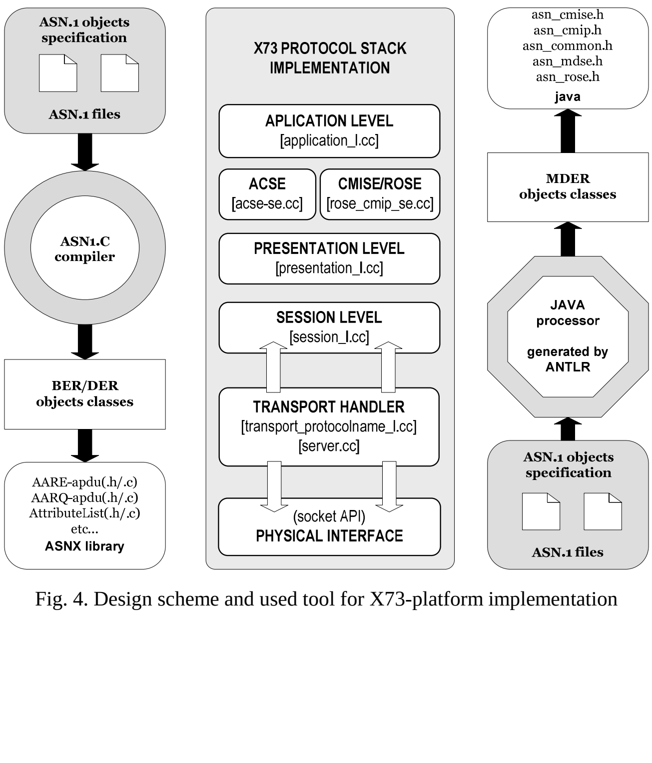

Fig. 4. Design scheme and used tool for X73-platform implementation

IV. X73-Platform Implementation

Our first approach to upgrade the system was based on a system design in which two protocol stacks were implemented into the same application. This way we can manage every process and message manipulation without the additional potential error source due to the communication layers (e.g. sockets, Bluetooth, serial port, etc.). Each one of the protocols stack corresponds to a MD and a CE. A main program will manage both stacks behavior, the information flow in terms of *packets (st_packet and st_buffer),* and provide a basic user interface to interact with the communication process.

Further research advances have optimized this system towards two independent protocol stacks in two different applications (MD and CE) thinking in its real implantation (see Fig. 5). These stacks are X73-compliant and represent the entire agent-manager communication model in several steps:

- Firstly, all the variables and parameters have to be properly initialized. The program will follow a single step routine whose user interface will show the communication status, the output/input buffer if necessary, protocol information and any other relevant execution progress report for the user to evaluate and decide to go to the next step or check any of the last parameters. With a MD connected to the computer, the program will lead the user to acquire vital samples and transmit the values to CE.
- From *main* we call a method that generates an event in the application layer of the MD (1). This event consists of a message that will be encapsulated through its path across the layers until it becomes a *st_packet* at the session layer. This *st_packet* is re-formatted as a *st_buffer* used by the *handler* layer (2). This *st_buffer* contains the initial message send by the MD to the CE.
- The program execution control is recovered by main that receives *st_buffer* as a return parameter.
- The *st_buffer* is encapsulated once leaves the MD as a buffer for the CE (3). This *buffer* goes through the CE layers from the lowest to the highest one by calling the *handler layer*. The *st_buffer* will be later de-encapsulated and processed by the application layer (4).
- The response from the CE to the MD message will be processed by the stack, leaving as a result a *st_buffer* that the main process will receive (5).
- Reply message reaches the MD (6), being firstly processed by the *handler*, goes up to the application layer and generates another message, user decision request or another action, depending on the communication stage. The entire process begins again from this stage (7).

One of the key points of this new implementation is the possibility of managing bits *on-the-wire*, capturing the bits being transmitted and shared between systems. This way, standard compliance tests can be run to evaluate any interoperability issue, one of the objectives of the CEN that will allow data and alarm management.

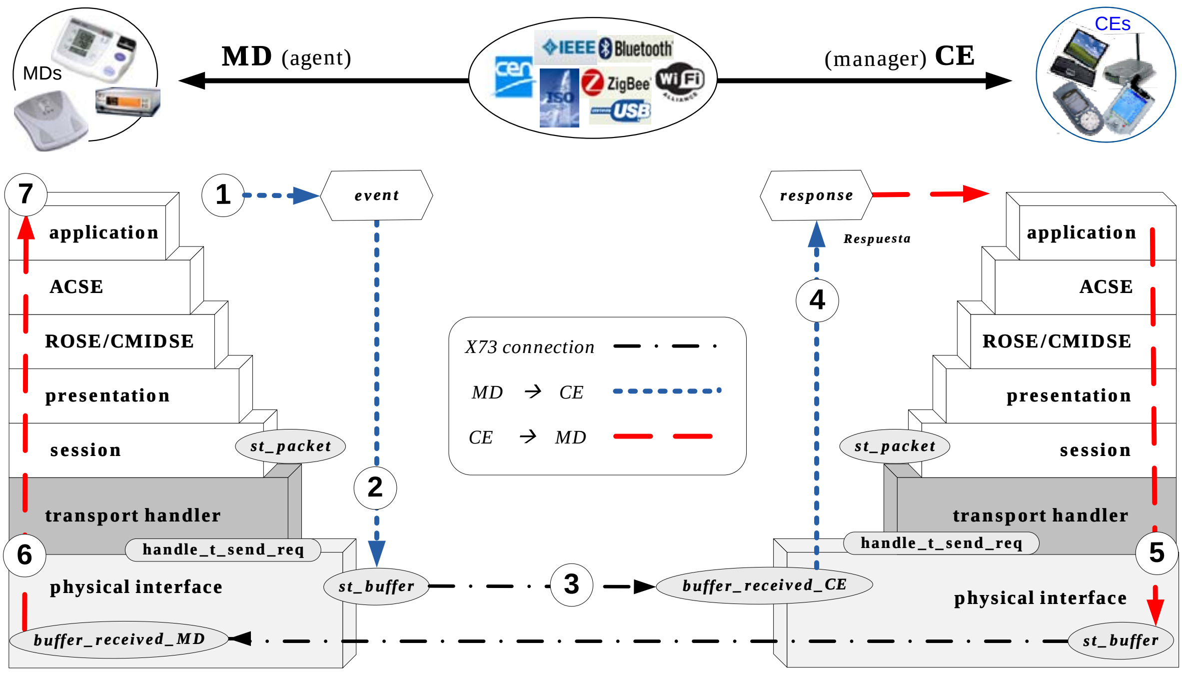

Fig. 5. MD/CE communication model X73-compliant

V. Results and Discussion

This platform implementation is X73-compliant and constitutes a middleware solution for patient telemonitoring. Moreover, the developed system can be used as a useful testbed for demonstrating the X73-communication model. Further system evolution, including the following open points, will permit to transfer this solution to the healthcare system.

The X73 demonstrator is shown in Fig. 6. It starts asking the user which MD wants to be used from an available list. After the selection of MD, its information and types of measures, a menu to control FSM are shown. From here, MD is initiated and the stack layers, operating interfaces and MDIB structure are created: MDS object, VMD and sub-branches of the tree.

Later, the transport system that supports the communication is required, getting the handler ready to support corresponding protocols. Moreover, information of the execution is shown in the screen, helping the engineer to know the methods of the layers. It is also shown how the buffers send the X73-PHD information and the other configuration parameters of the events and responses interchanged between MD and CE.

After the association, MD enters to the configuration point; CE sends the MDS object to the MD, without measurements yet. In CE a context of data reception is created (episodic or polling, regarding MD model). Thus, MD is now ready for the measurement acquirement (always under user request), and enters into the OPERATING state of the FSM.

By submitting data, MD updates the MDS object with the acquired measurements, and sends them to CE to be also updated. The received measurements are shown, detailing the X73-compliant identifications (in this example: 19230, 19229 y 18442; corresponding to the blood pressure device: diastolic pressure, systolic pressure and pulse, respectively). Finally, it is asked if more measurements are going to be made or, on the contrary, a menu is used to disassociate MD and CE or disconnect them as indicated by FSM.

Further advances in the system consist in an enhanced user interface which manage the protocol execution with buttons, display a lot of process information and implements customizable communication layers for testing (error tolerance tests, packet losing, etc.). It will also include a database with list of known devices that will reduce the association process if possible. Following lighter versions will focus on portable devices and its possibilities to be used as a Mobile CE.

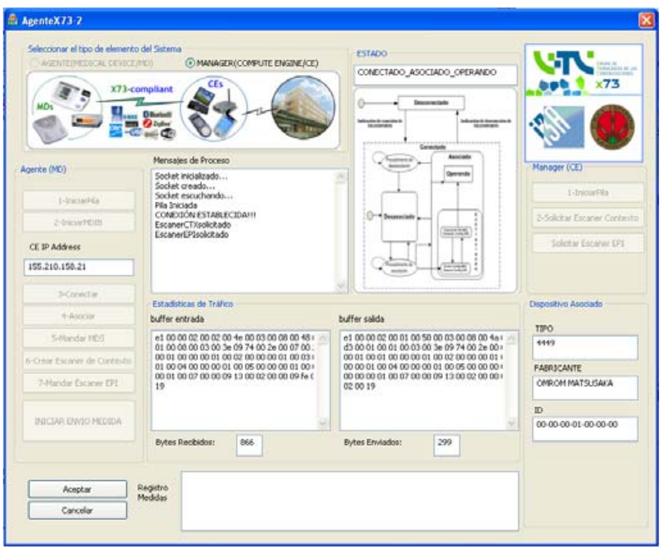

Fig. 6. MD/CE communication model X73-compliant

From this implementation results, several technical targets are opened, which are key objectives of CEN to advance on standard-based solutions for u-Health. These open points are:

- Implementation of the protocols stack into micro-controller devices (X73-compliant MDs) and wireless devices (mobile X73-CE), considering their appropriate technologies of transport level and physical interface from handler layer (connections manager). For example, when several MDs with different physical interfaces (e.g. USB and Bluetooth) connect with the same CE, handler layer communicates in transparent mode with both MDs independently of each transport protocol. The designed architecture for handler layer, with the implemented methods and parameters, has to be completed with each new connection profile in order to manage, in real execution time, each service access interface. This implies the design of private program methods through *sockets* and *threads* intro the handler layer to obtain a better management. Moreover, this design could be integrated with the implementation of allocation models and scheduling methods for particular MDs (under study from CEN).
- Support of multiple connections between several MDs and CEs, optimizing the different MDIBs creation and interchange, and implementing a FSM states manager that can read the specific MD configuration parameters and add them to a database in order to guarantee P&P functionalities. This is directly related with the previous open point from X73-PHD perspective. This problem, also worked in CEN, requires a MDs adapter/concentrator to multiplex the communication of different devices. Thus, if a single MD is updated, changing the entire platform is not necessary due to the MDs adapter can connect via Internet to manufacturer site, download the required software, and send the correct parameters for creating the new MDIB of the updated MD. Anyway, if the MD is aX73-compliant device, this process is transparent into the micro-controller and the multiplex function is implemented directly in CE to integrate MDs that are not X73-compliant but needed in clinical routine.
- Migration of the designed GUI to a completely interactive and customized user environment for being able to incorporate in the daily living of the patient. The environment has to be configurable regarding the device model (miniPC, mobile phone, smart phones, PDA, etc.), and in a transparent and modular mode. The designed platform as X73 demonstrator is a very useful tool for engineers and developers, but its commercial implantation requires an evolution towards a multimedia u-Health system. The design rules for the proposed architecture are ready for this evolution through Microsoft Foundation Classes (MFC), even through integrated graphic structures based Java, .Net, or Web 2.0, adapted for specific requirements of each application scenario, use case and OS (*Windows Mobile*, *Android*, *Symbian*, etc.) depending on the device model.

VI. Conclusion

The need of standardization on middleware solutions has derived to an end-to-end X73-compliant platform that allows achieving a ubiquitous and plug-and-play solution, ready for its integration with wearable sensor networks. Besides, it can be seen as an X73-PHD tester to prove the challenges currently under discussion in the CEN: flow and errors control, errors and alarms management, multiple MD connection with one or multiple CEs, or implantation on micro-controllers.

Acknowledgment

The authors wish to thank X73-PHD Working Group, CEN, and Mr. Melvin Reynolds, *convenor* of the CEN TC251 WGIV, for the contributions to this research. We appreciate the contribution of Miguel Galarraga to the excellent results carried out during the last years, and also Adolfo Muñoz (*Instituto de Salud Carlos III*), Paula de Toledo (*Univ. Carlos III*), y Silvia Jiménez (*Univ. Politécnica de Madrid*).

References

[1] W. W. Stead, R. A. Miller, M. A. Musen and W. R. Hersh, "Integration and beyond: Linking information from disparate sources and into workflow," Journal of the American Medical Informatics Association, 2000, vol. 7, pp. 135-145+146
[2] S. Pedersen and W. Hasselbring, "Interoperability for information systems among the health service providers based on medical standards," Informatik - Forschung Und Entwicklung, 2004. vol. 18, pp. 174-188
[3] R. J. Kennelly, "Improving acute care through use of medical device data," Int. J. Med. Inform. , Feb. 1998, vol. 48, pp. 145-149
[4] S. Sengupta, "Heterogeneity in health care computing environments," Proceedings - Annual Symposium on Computer Applications in Medical Care, 1989, pp. 355-359.
[5] POCT-1A2. "Point-of-Care Connectivity; Approved Standard-2nd Ed. Preview Sample Pages," http://www.clsi.org/source/orders.
[6] HL7. Health Level Seven, Devices Special Interest Group. http://www.hl7.org/Special/committees/healthcaredevices/index.cfm.
[7] DICOM. Digital Image Commun. in Medicine http://medical.nema.org/.
[8] ENV13606CEN/TC251. Electronic Healthcare Record Communication. Parts 1, 2, 3 and 4, Pre-standard," http://www.medicaltech.org.
[9] ISO/IEEE11073 Point-of-Care Medical Device Communication standard (X73-PoC). Health informatics. [Part 1. Medical Device Data Language (MDDL)] [Part 2. Medical Device Application Profiles (MDAP)] [Part 3. Transport and Physical Layers]. http://www.ieee1073.org. See also the previous standards: IEEE13734-VITAL and ENV13735-INTERMED of CEN/TC251, http://www.medicaltech.orgh.
[10] IEEE 1073:1996, IEEE Standard for Medical Device Communications - Overview and Framework. IEEE, ISBN: 1-5593-7755-0
[11] ENV 13734:2000, Health informatics - Vital signs information representation (also known as VITAL). CEN/TC251.
[12] ENV 13735:2000, Health informatics - Interoperability of patient connected medical devices (also known as INTERMED). CEN/TC251.
[13] Comiteé European de Normalisation (CEN). http://www.cen.eu.
[14] Comitée Européen de Normalisation, Technical Committee 251 –Health informatics (CEN/TC251). http://www.centc251.org.
[15] Continua Alliance web site: http://www.continuaalliance.org/home/.
[16] NHS Clinical Terms (also known as Read Codes), http://www. connectingforhealth.nhs.uk/terminology/readcodes/earlierversions.
[17] SNOMED (formerly Systematized Nomenclature of Medicine), http://www.snomed.org.
[18] ISO/IEEE11073 - Personal Health Devices standard (X73-PHD). Health informatics. [P11073-00103. Technical report - Overview] [P11073-104xx. Device specializations] [P11073-20601. Application profile - Optimized exchange protocol]. IEEE Standards Association webpage: http://standards.ieee.org/.
[19] J. Yao and S. Warren, "Applying ISO/IEEE 11073 standards to wearable home health monitoring systems," *Journal of Clinical Monitoring and Computing*, vol.19, pp.427-36, 2005.