ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 5, Issue 10, April 2016

Developing Software for IEEE 11073 Device Specializations using Antidote Library

Aravind S. Nair, V. Natarajan
Dept. of Electronics & Communication Engineering, SRM University, Chennai, India

***Abstract*— World of healthcare IoT have seen radical changes over the past few years with the surge in demand for wearables and health monitoring applications. One of the key features required for improving and popularizing these devices in the future will be interoperability between devices. The widely accepted standard to implement interoperability between agents and managers is the ISO/IEEE 11073 PHD standards. This paper discusses about the standard, its implementation using Antidote open library and development of software for new device specializations using the same**

***Index Terms*— Antidote open library, Device specializations, IEEE 11073, interoperability Personal Health Devices.**

I. INTRODUCTION

In the recent years, the technologies and frameworks used for Information Technology in healthcare have been through radical changes. The heath devices are designed to be connected to different compute engines for real time data collection. The surge in demand for wearable devices for fitness and health monitoring applications over the past years is a very good example for the growing demand in this field. Health devices exclusively for personal use like sleep apnea breathing therapy, ECG, Glucometers, insulin pump etc. which, unlike the devices which were used for Point-of-Care, are widely available in the market. These devices are manufactured by various manufacturers and most of them uses their own proprietary protocols for communication of data. The data thus obtained may be studied or utilized for further analysis using specific algorithms or saved for future reference.

Healthcare information technology initially focused majorly on Point of Care (PoC). PoC is a medical diagnosis technique where clinicians provide the necessary healthcare equipment and services to the patients at the time of need [1]. The diagnosis is done using the provided equipment and data thus obtained is collected and added to their health records, which may be maintained in electronic form called Electronic Health Records (EHR). The record provides information on history of medication and treatment, patient's progress during the treatment, reactions or allergies to medications and information on the healthcare providers and their observations. This data can be further used for performing quality analysis which may be useful for future diagnosis. The Personal Health Devices (PHD) uses similar principles as that of PoC but evolved itself into something better [2]. PHDs are devices personally owned by the user which can be used at any time and are designed for monitoring and providing real time data on specific health parameters to the user. The measured parameter may be stored, transmitted to or analyzed by a certified algorithm and referred to a physician for further diagnosis [3]. Thus an Internet of Things (IoT) for healthcare is formulated here. Various transport techniques like Bluetooth, Wi-Fi, USB etc. could be used for the transmitting the data to the remote gateway. Usually manufacturers use their own proprietary communication formats for transmitting the measured data to the gateway and to remote services [2]. This makes it difficult for devices from different manufacturers to communicate with each other and be a part of the desired healthcare IoT framework. It's also important not to lose the precision and validity of the data while it is being transmitted through various stages.

Interoperability of the health devices is a significant feature which ensures that the devices from different manufacturers can exchange and interpret the shared data by following a common standard protocol for data transmission. This ensures that any device or computing agent following the standard can communicate and understand the data without any modification in their software irrespective of the manufacturer. Continua Health Alliance, a not-for-profit industry organization, suggests use of a wide variety of protocols which can be used at various stages of its end-to-end (E2E) architecture designed for healthcare IoT [4]. The 'Agents' are the devices that measures and transmits the data (Source) to the 'Manager', which may host an application to collect the data (Sink). This transmission of data is achieved using a Personal Area Network (PAN) of Wired (USB) or Wireless transport technologies [4]. ISO/IEEE 11073 Personal Health Device standard, recommended by Continua for PAN, is one such group of standards that enables interoperability for PHDs. The standard defines their own object exchange protocol for effective communication of data between agents and managers [6]. The ISO/IEEE 11073, also known as X73, group of standards were proposed for the point-of-care systems (PoC) but the recent requirements and developments have made the standard to focus more on PHDs rather than PoC [2]. For these application oriented devices, the standard specifies device specializations (11073-1040zz), a group of standards in which each focusing on a specific type of health device. They support three domains: Disease management (10400-10439) for devices like pulse oximeter, ECG 1-3 lead, weighing scale etc. used for monitoring and collection of data. Health and Fitness (10440-10469) which supports heart rate monitor, Thermometers etc. used for fitness applications and Independent Living (10470-10499) in which devices supports features like disease management,

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 5, Issue 10, April 2016

independent living and activity hub and medication monitor. The specializations provides the attributes, data types and other associated information required while developing a communication stack based on X73 standard for that device.

This paper discusses about the ISO/IEEE 11073 standard, focusing on the fundamental information required by a developer for developing a communication stack of an 11073 PHD. For implementation of the standard, we are using the Antidote library, an open source library developed by Signove Corporation, implementing 11073 for health devices. The paper provides a layout on the important aspects of the Object Exchange protocol of 11073 standard, discusses on the implementation of the communication stack using Antidote and methodology to develop stack for new device specializations.

II. FUNDAMENTALS

A. ISO/IEEE 11073 PHD STANDARDS

The IEEE 11073 standard is a group of standards which emphasizes interoperability between agents and managers. The standard provides a framework which includes an efficient communication protocol for the transfer of data [1]. Agents only generate the data and hence have limited processing power [5]. They can be configured to send data either on demand or on availability to the manager. Managers usually have good processing power since the reception, analysis and further processing of the data is done here. In order to ensure ease of code reusability and independency on underlying transport technologies, the protocol are defined using an object oriented model. The devices are defined as objects which will have multiple attributes that may be configured by the manager. Among the group of standards, probably the most significant one is the IEEE 11073-20601 Optimized Exchange Protocol. This standard uses a lot of elements and methodologies defined in its predecessor standards to create a standardized and optimized protocol which is efficient for interoperability. Hence it is also considered as the base standard. Other standards in the group includes some fundamental standards like 11073 - 10101 Nomenclature, 11073 - 10201 Domain Information Model etc. which defines fundamental types and models which were used for developing the base standard. Another category is the device specializations which provides information on the object classes and attributes required for representing a PHD.

1. ISO/IEEE 11073-20601 STANDARD

This standard provides a framework for making an abstract model of the health data in a transport independent format such that the information profile can be converted into an interoperable and transmissible format [6]. It deals with the application layer services and the data exchange protocol. The application layer service provides information on association process and reliable transfer of actions and data between agent and managers. Data exchange protocol describes the protocol and associated commands and configurations required for successful data transmission. The overall IEEE 11073 is divided into 3 components: Domain Information model (DIM), Service model and Communication model [1, 6]. DIM is the key element in implementing the object oriented paradigm by characterizing the data from agents as a set of objects with multiple attributes. The fundamental object called Medical Device System (MDS) object class defines the properties and services offered by the device and is mandatory for all kinds of devices. Another class called Metric class is used for representing measurement and status information. Different sub-classes of Metric class like Numeric, Real Time Sample Array, Enumeration, Persistent Metric store etc. are used depending upon the type of data the device represents. The attributes associated with these classes can collectively represent the measured data, status or control flags. Service model on the other hand, provides primitives for data access that are exchanged between agents and manager. These entities are used for sending commands such as SET, GET and generate event reports. The event report provides the manager with the configuration information, current status of the device, measured data and other information. The third component Communication model, which deals with the communication topology established between agent and manager. The State machine for 11073 is shown below (Fig. 1.) [7]. which has 2 main states: Disconnected and Connected. Connected state have sub-states called unassociated (transport layer established), associating, associated (session established), configuring, configured, operating (data exchange occurs here), disassociating and disassociated states.

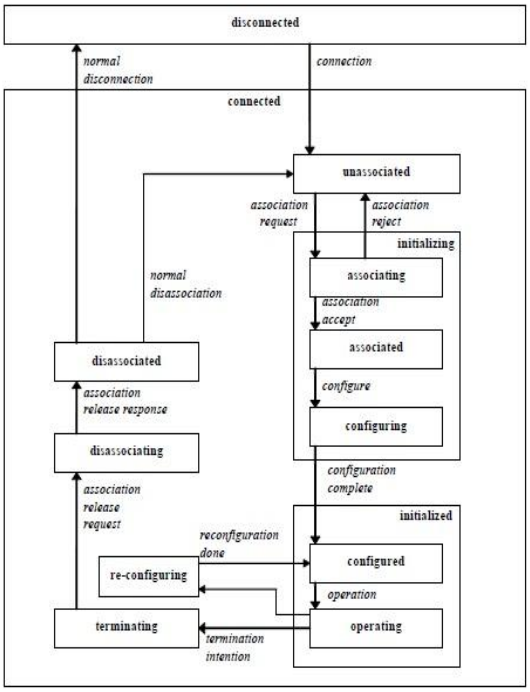

Fig. 1. State Machine of 11073 PHD

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 5, Issue 10, April 2016

The entire protocol messages are written using ASN.1 (Abstract Syntax Notation), a form of notation used for representing messages in telecommunication and networking. In order to convert them into bytes suitable data transmission and encoding rules should be used. Out of the many encoding rules, Medical Device Encoding Rules (MDER) is used, which is custom built for the collection of restricted data types used in 11073 protocol.

B. ANTIDOTE: IEEE 11073 LIBRARY

Antidote is an Open source library which can be used for developing and implementing communication stack for IEEE 11073 based PHDs. It includes files which defines the types, DIM, specializations (blood pressure monitor, glucometer, pulse oximeter, weighing scale) and transport plugins [8]. The stack is designed to be highly portable and is platform and transport independent to an extent. Antidote, which is written in C language and C standard library, supports almost all transport technologies like Bluetooth, USB, TCP/IP etc. provided the plugins for implementing them are added to the stack. These transport technologies have specific profiles designed to support 11073 PHDs. For e.g. Bluetooth have HDP (Health Device profile) [9]. These profiles are desired to provide the required reliable and best effort virtual channels used for communication. It comes with a Bluez plugin with support for HDP, which can be utilized in Linux environment for using Bluetooth. Though the stack is designed not to have any dependency related issues, the dependency associated with the transport plugins have to be taken care of by the developer.

The architecture of Antidote is shown below (Fig. 3.) [8]. Stack supports modularity so that the modules are reusable and independent of each other at the same time reducing the complexities of the 11073 standards. Among these modules manager applications, communication plugins and transport plugins are not part of the standard. Antidote encapsulates all data (measurements, attributes, and configuration) on Datalist structures for the ease of transverse and interpretation. The communication plugin designed should provide services like notification to stack on connection status, maintaining unique id for each connection, transfer of APDU (Application Protocol Data Unit, which is the 'packet' in 11073) and close a connection when requested by the stack. The plugin communicates directly with the application and is not interfered by the Antidote thus maintaining the independency.

Another important feature of Antidote is that it supports Transcoding. Any non-11073 device can communicate to a client application built for 11073 (like 11073 manager) through a transcoding plugin custom made for that device. The plugin allows the non-11073 device to communicate with the application in the same base protocol. The downside is that the plug-in needs to be built by the developer which requires strong knowledge on its operation & dependencies.

They are device dependent and hence there is no universal plug-in which can be designed to be used for all devices. This is because, unlike the communication plug-in where it only needs to send the APDU back and forth, transcoding plug-in will have to parse the protocol and re-encode the messages in IEEE 11073 structure.

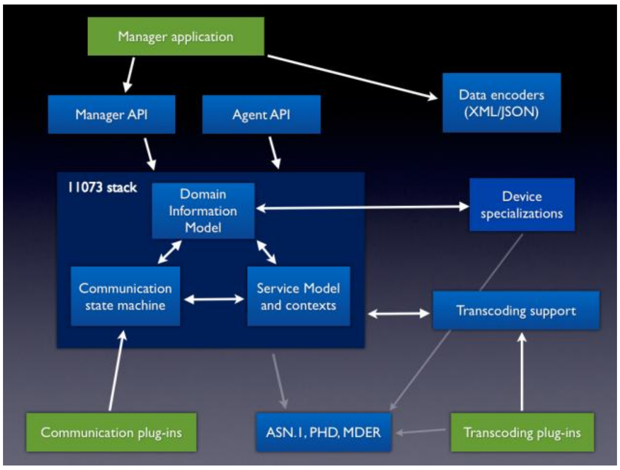

Fig. 2. Antidote Architecture

III. METHODOLOGY & IMPLEMENTATION

In 11073, after associating, the device will prepare to transmit the measured data. As part of this, it will send a device configuration, which will explain to the manager what all kinds of objects exist in MDS, their attributes, types, handles etc. [6, 8]. If it is a standard configuration, the manager will be familiar with this predefined configuration and enters into operating state. If it is an extended configuration, i.e. a configuration custom designed by the developer, then manager need to learn about it first before entering into operating state. The standard configurations have their own unique device configuration id and will have certain mandatory objects which need to be established. For e.g.: 11073-10406 basic 1-3 lead ECG device, have a standard configuration with id 0x258 for its heart rate profile [10]. It must support one numeric object for measuring heart rate. In Antidote, the specialization for 4 devices are available and anything new need to be added by the developer.

Antidote needs to be installed in a PC/Laptop which will act as the manager. The standard Antidote is built in Linux using auto tools [8]. We recommend installing the following packages before installing Antidote: auto make, libtool, libdbus-1-dev, libdbus-glib-1-dev and libusb-1.0-0-dev. A full-fledged manager application called 'healthd' is available along with the library which uses D-Bus IPC (Inter Process Communication) for its operation. A copy of the CONF file associated with the service file (healthd.conf) is required to be available at the dbus source path to ensure that healthd exports the D-Bus service to other applications. Once the installation is completed, the healthd application can be executed in a terminal which will listen for any incoming connections from paired medical devices. Since an 11073 compatible ECG device was not available to us, we used another laptop with Antidote installed, where a script

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 5, Issue 10, April 2016

simulating an ECG agent will be running. The ECG specialization is added to this script and it will generate some dummy data for the standard specialization 0x258. The data is obtained at the terminal running a python script on the manager.

In order to develop software for new device specialization, the developer needs to have access to the specialization document (like 10406 [10]). The document provides information on the configuration procedure performed by the agent and provides examples for generating event reports for data transmission after association (Fig.3). The configuration procedure is performed using a ROIV (Remote Operation Invoke) Confirmed event report and the format and contents of the device configuration is added. In Antidote device specialization, header file defines the nomenclatures for the device (available from specialization), declares the standard configuration and defines the structure for the event report data. The source file will define the configuration structure using StdConfiguration data structure for standard configurations, configuration procedure using ConfigObjectList and generates the event report using DATA_apdu data structure. The type of the parameter to be measured is specified while defining event report generation. Once the device specialization is created, they need to be registered in the manager initialization function in manager source file.

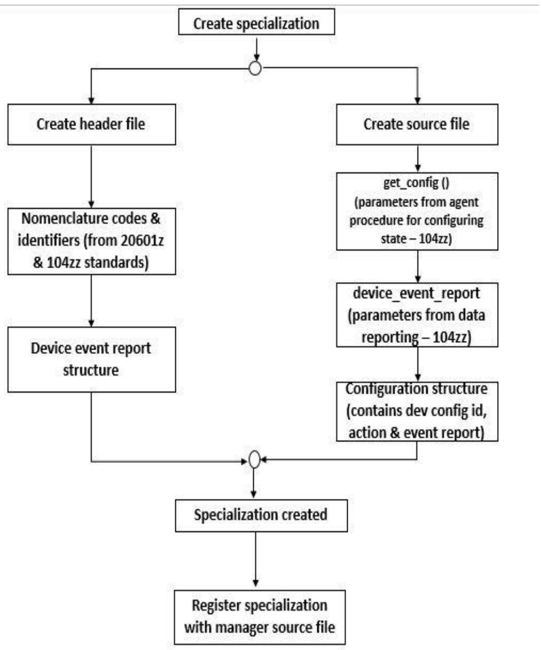

Fig.3. Procedure for adding new specialization

IV. RESULTS

The project is executed in two computers where one acts as the Agent, simulating a basic ECG device and the other acts as the Manager. The specialization created is added to a sample test file which simulates an agent communication over Bluetooth. The exchange of simulated data occurs after association and activity is seen at the manager terminal.

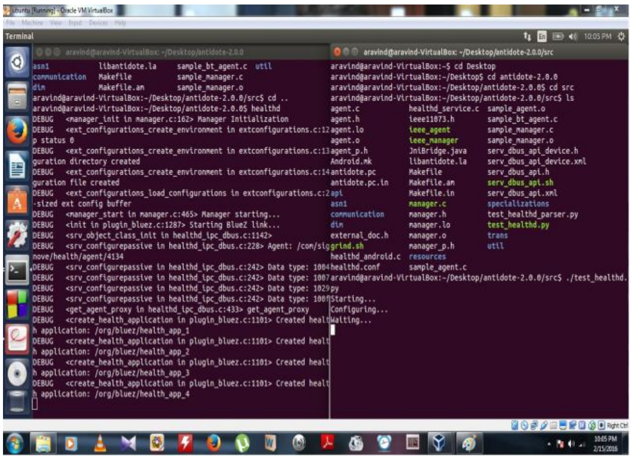

Fig.4. Execution of manager application

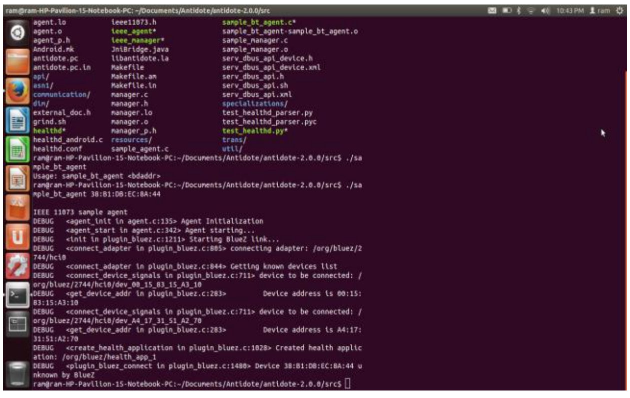

Fig. 5. Execution of Agent application

Fig. 6 Experimental Setup

V. CONCLUSION & FUTURE WORK

IEEE 11073 standard is the de facto standard for enabling interoperability in Personal Health Devices and are adopted by certifying organizations like Continua Health Alliance in their design guidelines. This paper provides a detailed overview on the important concepts required for a developer and discusses on the implementation experiences of the stack using Antidote. Future work will focus on developing a transcoding plug-in which will enable a non-11073 agent to communicate to an 11073 manager. Further, the porting of the stack to different platforms other than Linux can also be performed.

ACKNOWLEDGEMENT

The authors would like to express their sincere gratitude to Department of Electronics and Communication Engineering of SRM University, Chennai, India for providing us the required facilities and undeterred support throughout the duration of the project work. We would also like to acknowledge the efforts of Signove Corporation for developing the 'Antidote' 11073 open source library, considering the complicated nature of the standard. Author Aravind Nair would also like to thank his family and friends for their support during the course of the work.

REFERENCES

[1] Santiago Led, Miguel Martínez-Espronceda, Javier Redondo, Alfonso Baquero, Luis Serrano, Luis Cabezas, Maciej Niegowski., Wearable Electrocardiogram (ECG) Recorder for a Mobile Point-of-Care based on Recent Interoperability Standards in 2013 IEEE Point-of-Care Healthcare Technologies (PHT) Conference Bangalore, 2013, ISBN: 978-1-4673-2767-1.

[2] Hector Gilberto Barrón-González, Miguel Martínez-Espronceda, J. D. Trigo, Santiago Led and Luis Serrano. Recent Advances in mHealth: An Update to Personal Health Device Interoperability Based on ISO/IEEE11073. In: Springer Series in Bio-/Neuroinformatics Vol.5, pp 375-397.

[3] B. Moorman, "Medical device interoperability: Standards overview," IT World, pp. 132–138, March/April 2010.

[4] Continua Health Alliance. Interoperability Design Guidelines for Personal Health Systems 2014 [Online]: http://www.continuaalliance.org/products/design-guidelines .

[5] Miguel Mart´ınez-Espronceda, Ignacio Mart´ınez, Luis Serrano, Santiago Led, Jes´us Daniel Trigo, Asier Marzo, Javier Escayola, and Jos´e Garc´ıa. Implementation Methodology for Interoperable Personal Health Devices With Low-Voltage Low-Power Constraints in IEEE Transactions on Information Technology in Bimedicine, Vol. 15, No. 3, May 2011.

[6] Health Informatics-Personal Health Device Communication Part 20601: Application Profile–Optimized Exchange Protocol. (2008). IEEE Standard 11073-20601-2008. [Online]. Available: http://standards.ieee.org/findstds/standard/11073-20601-2008.html.

[7] Health informatics -- Point-of-care medical device communication -- Part 10201: Domain information model (2004). IEEE Standard 11073-10201-2004. [Online]. http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=36341 .

[8] Signove Corporation. Antidote Development Guide. [Online]: http://oss.signove.com/index.php/File:AntidoteProgramGuide.pdf.

[9] Bluetooth SIG. Personal Health Devices Transcoding Whitepaper, v1.6 2014. [Online]: https://www.bluetooth.com/develop-with-bluetooth/white-papers .

[10] Health informatics - Personal health device communication Part 10406: Device specialization--Basic electrocardiograph (ECG) (1- to 3-lead ECG). [Online]: https://standards.ieee.org/findstds/standard/11073-10406-2011.html.