

# VITAL: Use and Implementation of a Medical Communication Standard in Practice

C Weigand

Fraunhofer Institute for Integrated Circuits, Erlangen, Germany

## Abstract

Modern medicine, especially intensive care, demands the automated interaction of multiple devices for the monitoring and therapy of patients. At present, the communication between devices from various manufacturers is often impossible due to different interfaces and transmission protocols. Open system architecture and standardized communication protocols like the CEN standard ENV 13734/35 (VITAL) will resolve this issue. VITAL (now also part of the IEEE 11073 standards family) describes the design of communication specifications for the transmission of device and vital parameters, independent of device type or manufacturer as well. Additionally, VITAL defines a general data and communication model enabling access to required data – even under real-time conditions. VITAL also enables the automatic configuration and coordination of networked devices. Obviously, VITAL covers a very wide range of applications. Thus device manufacturers need assistance to implement on different platforms and systems compliant devices. To meet this requirement not only a library was designed but also a framework with a special plug-in interface was developed. This allows to program simple plug-ins for all different purposes medical devices have. Using this framework a device manufacturer is in the position to produce VITAL compliant devices without the need of knowing details of VITAL itself. The framework is platform independent and therefore usable on different systems especially small embedded devices.

## 1. Introduction

VITAL [1] describes the design of communication specifications for the transmission of device and vital parameters, independent of device type or manufacturer as well. Additionally, this open communication standard defines a general data and communication model enabling access to required data – even under real-time conditions. VITAL also enables the automatic configuration and coordination of networked devices. As an example, VITAL can be used to log the bio signals gathered on one

diagnostic device and to produce a real-time curve with the data on a separate display. Such technology is ideal for intensive care, sleep research centres, or at an intracardiac catheter measuring station. VITAL also enables the simultaneous accumulation, monitoring, and when needed, the graphical representation of diverse bio signals from several patients.

VITAL enables users to operate systems consisting of pieces of equipment from different manufacturers with little additional input. VITAL also simplifies the expansion of manufacturers' equipment portfolios. Due to straightforward software and hardware extensions, stand-alone measuring devices can be made network-compatible and therefore can be integrated into existing networks in a manner that enables them to recognize the correlation between patient data and to detect critical states of patients. For the user in the medical field, improvements can be realized concerning multi-parameter analysis, long-term monitoring and finally, advanced compliance. Economical and high quality disease management is also now possible. Patients' benefit due to expanded possibilities for individually optimised, more reliable diagnostics and therapies.

## 2. Methods

The VITAL communication architecture [2] is based on the »Agent/Manager« concept of the ISO-Management-System and therefore defines the user specific layers in the ISO/OSI Layer Model (figure 1), especially the »Application Layer«.

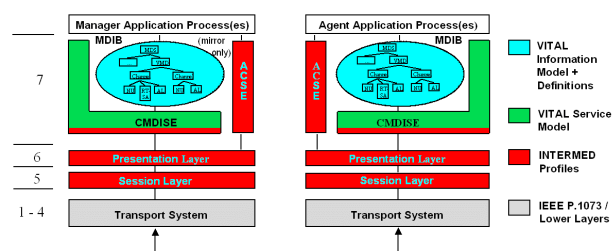


Figure 1: ISO/OSI layer model

This provides for flexibility in the choice of the protocols for the underlying – transport-oriented – layers. This allows the application of VITAL, for example, in combination with Bluetooth, TCP/IP, IrDA, or other transmission systems.

To facilitate automated ad-hoc hardware communication, all transmitted elements of communication must be transformable in appropriate codes. Therefore, VITAL uses an object oriented information model and a substantial nomenclature [1] [3] [4] for all usable model elements, device types, dimensions, and measuring units, as well as medical values and conditions. The following example depicts a term identifying the first lead of an ECG curve, which is assigned the code 256.

Electrical Potential | ECG I | Heart | CVS 256

The four parts are separated by the Symbol '|'. Base of the term is the measurement of an electrical potential. The next part describes the *origin*, which means the first lead of an ECG curve. The part 'Heart' describes the place the measurement *concerns*. And the last part (CVS) characterizes the *context*, in this case the cardiovascular system. These 16 Bit codes are context sensitive and allow to characterize not only medical values and conditions but also all useable model elements.

In figure 2, a thermometer and a blood pressure meter with their measured values, as well as patient-specific data are depicted in an object hierarchy and merged into a complete system as an example for the VITAL-specific modeling of medical equipment.

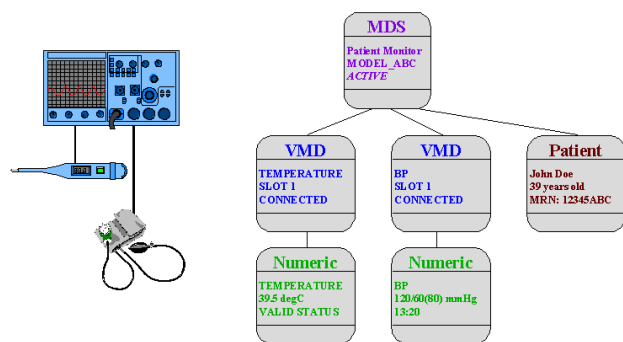


Figure 2: Medical device modeled with VITAL

The IEEE 1073 series of standards [5], which provides a detailed description of various medical devices, exist to select the appropriate model representing the medical information of a specific device, (complementary to VITAL). These documents focus on how to incorporate several virtual medical devices (VMD) (like ECG [6], Blood pressure [7], CO<sub>2</sub>, SpO<sub>2</sub> [8], Temperature, etc.) to create compliant and complete medical devices.

### 3. Results

VITAL covers a very large domain and a huge variety of scenarios and therefore almost every medical device can be modelled with objects from the so called Domain Information Model (DIM). Developing medical devices means that development has to happen on small embedded systems with few memory and slow micro controllers. These micro controllers have to be programmed in C/C++ or in assembler at all. This means a very slim implementation of VITAL is needed which scales to the requirements of an embedded system. But not every medical device developer wants to implement VITAL himself or further more is able to port it to his micro controllers becoming the implementation very slim.

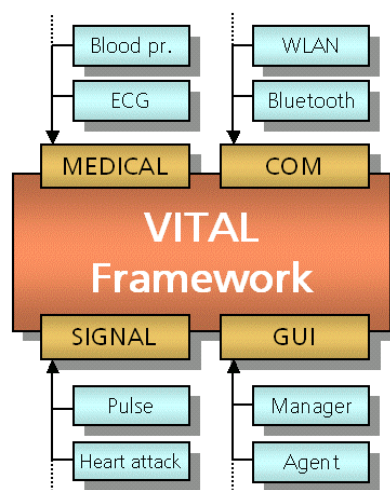


Figure 3: VITAL framework with plug-in interfaces

Fraunhofer IIS therefore provides not only support in implementing VITAL on embedded systems or other platforms, but build up a sizeable library to match everybody's needs. The library itself is surrounded by a framework, which provides several interfaces (API) for plug-ins with different purposes. Figure 3 shows the framework with the four different plug-in APIs *MEDICAL*, *COM*, *SIGNAL* and *GUI*. The *COM* API interfaces the lower layer communication systems such as WLAN, Bluetooth, Infrared or Ethernet. Depending on the medical device communication interface, only one plug-in is needed to establish communication. To model a medical device the *MEDICAL* API provides easy-to-use methods for the design of medical devices, which use the objects in the Domain Information Model. No in-depth knowledge of the VITAL standard is needed. Only the

Objects of the DIM have to be known and used. For signal processing the *SIGNAL* interface can be used. Several different signal processing algorithms can be wrapped in a plug-in and used on the data measured by the medical device. There is no need to re-program the whole device for new or faster signal processing algorithms. Only plug-ins have to be exchanged. For devices with a display and a user interface – or on the other side of the communication link for a monitoring unit – a graphical user interface is essential. This is covered by the plug-in interface called *GUI*. Depending on the data to be visualized and the communication end point (agent or manager) different *GUI*-plug-ins are available. A once implemented *GUI* plug-in for visualization of the acquired data is reusable in different monitor applications to display the same data.

In advantage of a platform independent, easy scaleable VITAL library with an easy to use plug-in interface medical devices manufacturers are now able to provide a standardized communication link with their devices. To prove functionality of the VITAL framework several sample applications were implemented.



Figure 4: wireless VITAL application

The overview of such a sample application illustrates figure 4. In the lack of a VITAL compliant ECG an ECG holter off the shelf was used. This is also a positive proof for making existing medical devices VITAL compliant. The ECG holter is connected to a PDA and provides its data via a serial cable. A VITAL agent application using the framework is reading the data and provides it via a WLAN connection to a network. Medical personal in range of this wireless network is now able to connect to this VITAL agent and to visualize the data measured by the ECG device. This can be done by a computer or laptop or just another handheld device like a PDA.

The next two pictures illustrate the user interface of the VITAL Manager application. Figure 5 depicts the Medical Data Information Base (MDIB) of the two-lead ECG device (two real time sample arrays with one

numeric for the hart rate).

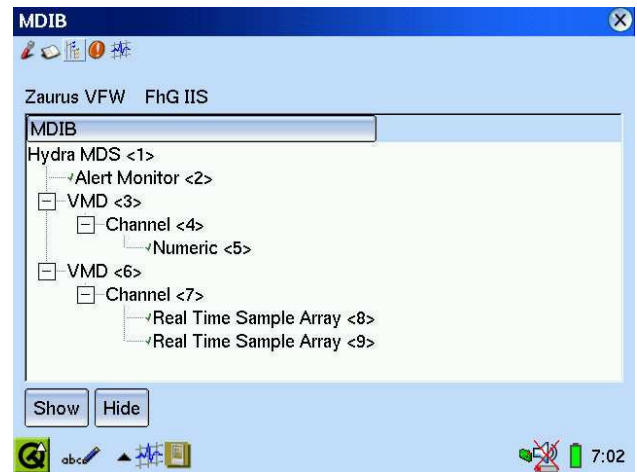


Figure 5: View of Medical Data Information Base on the Linux PDA

This MDIB is similar to the MDIB shown in the device model of figure 2. The implementation of this view on a VITAL-Manager application was done on a Linux PDA using the VITAL framework. In correspondence to the *GUI* plug-in interface a *GUI* plug-in was developed to view the ECG-curves of the two-lead ECG device online and in real-time (Figure 6).

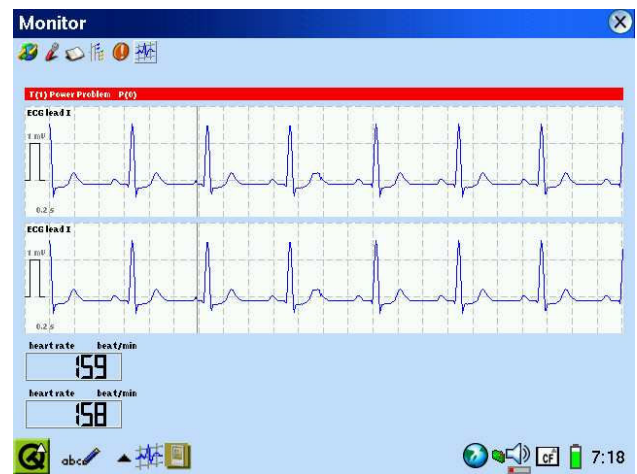


Figure 6: Visualization of ECG-Data on a PDA

Using this application for instance in a home-care environment by utilizing the Internet allows physicians to stay in close contact to their patients. It allows real-time monitoring of patients vital data and endorses disease management. Standardized, open communication interfaces, like the one provided by VITAL, help to broaden this basis system for disease management in a

simple manner. For the measurement of additional bio-signals other VITAL compliant devices are integrable with on effort.

#### 4. Discussion and conclusions

In advantage of a platform independent, easy scaleable VITAL library with an easy to use plug-in interface medical devices manufacturers are able to provide a standardized communication link with their devices. Using the library through the framework and the plug-in APIs can increase time to market. Device developers can concentrate on their main business – developing a new medical device – and do not have to work oneself in VITAL internals.

Interoperability in the medical sector is becoming more and more important and therefore not only the patient data has to be transferred between several institutions but also the measured vital parameters of the patients have to be transferred from each medical device through several networks into a patient record. All medical devices have to be connected to each other establishing a network. And therefore devices of many different manufacturers have to work together exchanging their data. This can only be done by an open communication standard like VITAL.

For the domain, VITAL is defined in, VITAL provides interoperability and a standardized interface to achieve connectivity everywhere. The VITAL nomenclature makes sure that every medical data is well coded and can be understood by every communication partner. But to bring VITAL in medical devices easy to use tools like the Fraunhofer-IIS VITAL framework are needed. This enables manufacturers to produce VITAL compliant medical devices without the need to understand the whole complexity of the VITAL standard.

#### References

- [1] CEN/TC251/PT5-021. Health informatics—Vital signs information representation—VITAL. CEN ENV13734, Final Draft, July 1999.
- [2] CEN/TC 251/PT 35. Interoperability of patient connected medical devices. CEN ENV13735, May 1999.
- [3] Zywietz C, Kraemer M, Fischer R, Widiger B. Integrating the ECG Enterprise - HES-EKG with the Built-in Vital Signs Information Nomenclature. IEEE Computers in Cardiology 2004;31:41-44.
- [4] Anagnostaki A, Pavlopoulos P, Koutsouris D. A Novel Codification Scheme Based on the “VITAL” and “DICOM” Standards for Telemedicine Applications. IEEE Transactions on Biomedical Engineering 2002;49:12:1399-1411
- [5] IEEE 1073.1. Medical Device Data Language (MDDL) framework and overview. in Standard for Medical Device Communications, Apr. 1997.
- [6] IEEE 1073.1.6. Medical Device Data Language (MDDL) virtual medical device, specialized—ECG component. in Standard for Medical Device Communications, Dec. 1998.
- [7] IEEE 1073.1.3.7. Medical Device Data Language (MDDL) virtual medical device, specialized—Blood pressure. in Standard for Medical Device Communications, Mar. 1999.
- [8] IEEE 1073.1.3.4. Medical Device Data Language (MDDL) virtual medical device, specialized—Pulse oximeter. in Standard for Medical Device Communications, Apr. 1999.

Address for correspondence

Christian Weigand  
Fraunhofer-Institute for Integrated Circuits  
Am Wolfsmantel 33  
D-91058 Erlangen, Germany  
[Christian.Weigand@iis.fraunhofer.de](mailto:Christian.Weigand@iis.fraunhofer.de)