

# VITAL Schnittstelle zur Persistierung von Echtzeitdatenströmen

Dipl.-Inf. Matthias Struck, Svetoslav Pramatarov, Dipl.-Inf. Christian Weigand  
Fraunhofer Institut für Integrierte Schaltungen (IIS), 91058 Erlangen, Germany

## Kurzfassung

Die drahtlose Kommunikation zwischen Sensoren, die kontinuierlich Vitalparameter von Patienten erfassen, und mobilen Endgeräten zur Visualisierung und zur Weiterverarbeitung dieser Parameter wird zunehmend zum Stand der Technik in der klinischen Routine. Um den Zugriff und den Austausch der akquirierten Daten auch in Zukunft von bzw. zwischen unterschiedlichen Institutionen zu gewährleisten, müssen standardisierte Methoden zur Persistierung der Daten geschaffen werden. Zu diesem Zweck wurde eine neue Schnittstelle im VITAL-Framework definiert (siehe **Bild 1**), welche eine flexible Persistierung von Echtzeitdatenströmen ermöglicht. Um eine Aussage über die Tauglichkeit verschiedener Dateiformate treffen zu können, wurden die folgenden Formate evaluiert: EDF, EDF+, FEF und SCIPHOX. FEF hat den Vorzug, dass es die vollständige VITAL-Objekthierarchie speichern kann. Im Gegensatz zu FEF ist EDF/EDF+ in den medizinischen Einrichtungen allerdings beinahe zum Standard geworden.

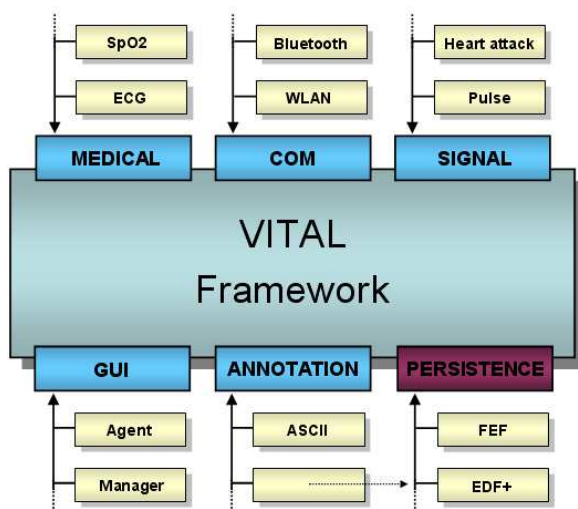
## 1 Einleitung

Für die Integration von Sensoren unterschiedlicher Hersteller in ein Netzwerk sind eine standardisierte Datenrepräsentation und eindeutige Kommunikationsprotokolle erforderlich. Der CEN-Standard ENV 13734/35 „Vital Signs Information Representation“ (VITAL) beschreibt eine gerätetyp- und herstellernabhängige Darstellung und Übertragung von Vitalparametern. Die Implementierung dieses Standards wurde in ein Framework eingebettet, welches Schnittstellen zur Integration verschiedener medizinischer Geräte, Kommunikationsprotokolle wie z.B. WLAN und Bluetooth, Signalverarbeitungs- und Annotierungseinheiten sowie graphischer Benutzeroberflächen bereithält (siehe **Bild 1**).

## 2 Material und Methoden

### 2.1 Anforderungen und Design

Beim Design einer Schnittstelle zur Datenpersistierung gibt es einige Aspekte zu beachten: Die Schnittstelle muss Methoden definieren und ihre Funktionalität beschreiben. Die konkrete Implementierung erfolgt anschließend für jedes Dateiformat separat, wird als Plugin realisiert und in das Framework integriert. Des Weiteren müssen die Dateiformate hardwareunabhängig sein, um auch zwischen verschiedenen Systemen austauschbar zu sein. Es muss sichergestellt sein, dass ein schneller Zugriff auf die persistierten Informationen möglich ist und dass diese eine möglichst geringe Redundanz besitzen. Der Grad des Datenverlusts nach einem möglichen Hard- bzw. Softwareabsturz des Systems muss minimal sein. Daher müssen die erfassten Vitalparameter kontinuierlich auf das Speichermedium geschrieben werden. Um all diese Anforderungen zu erfüllen, wurden drei Methoden für die Persistierungsschnittstelle deklariert, die von dem Framework aufgerufen und für die jeweiligen Plugins implementiert werden müssen: Die erste Methode speichert die demographischen Patientendaten, welche über einen individuellen Benutzerdialog eingegeben werden können. Die zweite ermittelt die Objekthierarchie des VITAL Standards, um die konkrete Struktur der Datei festzulegen, in welche die zu persistierenden Daten geschrieben werden sollen. Die dritte Methode wird stets dann aufgerufen, wenn der interne Puffer, welcher die erfassten Messwerte enthält, voll ist. Sie wird parallel zu der Methode der graphischen Benutzeroberfläche, welche die Vitalparameter in Echtzeit visualisiert, aufgerufen und garantiert folglich einen minimalen Datenverlust bei einem möglichen Systemabsturz.



**Bild 1** Schnittstellen des VITAL-Frameworks

## 2.2 Grundlagen der ASN.1

Die Abstract Syntax Notation One erlaubt die programmiersprachenunabhängige Beschreibung von Datenstrukturen. Nach der Spezifizierung aller erforderlichen Strukturen können diese mittels eines ASN.1 Compilers<sup>1</sup> in eine beliebige Programmiersprache wie z.B. C++ übersetzt werden. Außerdem generiert der Compiler entsprechende Methoden, um die Datenstrukturen zu serialisieren bzw. zu deserialisieren. Daher können die Daten einfach auf ein Speichermedium geschrieben bzw. in einem Netzwerk kommuniziert werden. Nach einem kleinen Beispiel, wie eine einfache Datenstruktur in C++ Code transformiert wird, werden im folgenden Abschnitt einige Dateiformate vorgestellt und bewertet:

ASN.1 Spezifikation:

```
DataStructure ::= SEQUENCE
{
    int_value_1  INTEGER,
    int_value_2  INTEGER OPTIONAL
}
```

Generierter C++ Code:

```
typedef struct DataStructure
{
    int  int_value_1;
    int* int_value_2; /*OPTIONAL*/
} DataStructure_t;
```

## 2.3 Evaluierung von Dateiformaten

Bei der Wahl geeigneter Dateiformate für die Persistierung von medizinischen Echtzeitdatenströmen stößt man auf weitere Anforderungen, welche es zu beachten gilt: Zu diesen gehören der Grad der aktuellen Verbreitung und Anwendung der Dateiformate in der klinischen Routine sowie die Qualität der Tools, mit denen die Dateiformate ausgelesen und visualisiert werden können. Das sogenannte *European Data Format* (EDF) ist ein einfaches Format zum Austausch und zur Persistierung von Biosignalen mit mehreren Kanälen. Neben der einfachen Struktur bestehen die Hauptvorteile dieses Formats in seiner großen Verbreitung in den Kliniken und in der Existenz vieler nicht-kommerzieller Visualisierungstools. Eine Erweiterung des EDF ist EDF+, welches innerhalb derselben Datei zusätzlich auch unterbrochene Signale verwalten kann. Des Weiteren kann ein Kanal des EDF+ derart kodiert werden, dass dieser medizinische Alarmer wie z.B. ein niedriger Batteriestatus oder Annotationen des Arztes speichern kann. Das *File Exchange Format* (FEF) ist universeller als EDF/EDF+. Es basiert auf dem VITAL CEN ENV13734/35 Standard und wurde mit der CEN/TS 14271 zu einem ei-

genen Standard, welche dieselbe Nomenklatur zur Repräsentation medizinischer Geräte und Vitalparameter verwendet wie VITAL. Daher erfährt die Persistierung mittels FEF keinerlei Informationsverlust im Sinne des VITAL Standards. Außerdem definiert der FEF Standard einen sogenannten Multimediaabschnitt, in dem entweder Audio- und Videodaten direkt abgelegt werden bzw. referenziert werden können. Im Gegensatz zu den vorherigen ist *das Standardized Communication of Information in Physician Offices and Hospitals using XML* (SCIPHOX) kein binäres Format, sondern basiert wie der Name bereits andeutet auf der XML Syntax. Aufgrund der hohen Redundanz bei der Speicherung von numerischen Daten konzentriert sich SCIPHOX im Wesentlichen auf die Persistierung demographischer und administrativer Daten. Nichtsdestotrotz hat SCIPHOX den Vorteil, dass sein Inhalt von beliebigen XML Parsern gelesen und interpretiert werden kann.

## 3 Ergebnisse

Es wurde eine generische Schnittstelle zur Persistierung medizinischer Informationen vorgestellt, welche eine beliebige Anzahl an Biosignalen in Echtzeit speichern kann. Unterschiedliche Dateiformate können als Plugins realisiert und in das VITAL-Framework integriert werden. Die Formate können mittels eines ASN.1 Compilers implementiert bzw. generiert werden. Das Design der Schnittstelle zur Persistierung garantiert ein Minimum an Informationsverlust im Fall eines Systemabsturzes.

## 4 Diskussion

Weil FEF konsistent zum VITAL Standard ist, stellt die Persistierung medizinischer Daten mit diesem Format eine adäquate Lösung dar. Unglücklicherweise wurde der Standard erst verabschiedet, nachdem bereits viele „Insellösungen“ in die klinischen Arbeitsprozesse eingeführt wurden. Daher ist es umso wichtiger, dass das VITAL Framework gleichzeitig auch mit anderen Formaten umgehen kann. Um sowohl die größtmögliche Datenerfassung als auch den unmittelbaren praktischen Einsatz zu gewährleisten, sollte man für die Persistierung FEF und zusätzlich EDF/EDF+ verwenden.

## 5 Literatur

- [1] Walkin, L. Open Source ASN.1 Compiler. Webdokument: <http://lionet.info/asn1c>, 2006