



**xDELIA**

Project Number 231830

Xcellence in Decision-making through Enhanced Learning  
in Immersive Applications



---

Document Name:	<b>Monitoring framework: implementation and evaluation (D22-4.3)</b>
Document Date:	<i>Feb 29 2012</i>
Document Owner:	<i>xDELIA Consortium</i>
Document Author/s:	<i>Kristina Schaaff FZI Forschungszentrum Informatik</i>
Contributors:	<i>Stephan Heuer FZI Forschungszentrum Informatik</i>
Deliverable Number:	<i>D22-4.3</i>
Work Package:	<i>WP4</i>
Deliverable Type:	<i>Report</i>
Version	<i>2</i>



## COPYRIGHT AND CONFIDENTIALITY NOTICE

The work described in this document was performed as part of the xDELIA project ('Boosting Deliberate Practice and Handling Biases through Immersive Cognitive and Emotional Reinforcement Strategies & Tools') which is funded under contract No. 231830 of the European Community. The project is a collaboration between CIMNE (coordinating partner), Forschungszentrum Informatik, Open University, Blekinge Tekniska Högskola (Game and Media Arts Laboratory), Erasmus University Rotterdam (Erasmus Centre for Neuroeconomics), University of Bristol (Personal Finance Research Centre), and Saxo Bank A/S. The opinions, findings and conclusions expressed in this report are those of the authors alone and do not necessarily reflect those of the EC or any other organisation involved in the project.

No part of this publication may be reproduced or transmitted in any form or by any means, whether electronic, mechanical, photocopying, recording or otherwise; nor stored in any information retrieval system of any kind; nor used for tendering or manufacturing; nor communicated to any person, without the prior permission in writing of the copyright owner.

The contents of this document must be treated as confidential and protected as defined in the terms and conditions of the xDELIA Consortium Agreement.



# Appendices

<b>Document Name</b>	<b>Date</b>
Appendix A. References	February 29, 2012



## Executive Summary

This deliverable describes the development of an online psychophysiological monitoring component called xAffect and the data management framework that represent the outcomes of T4.6 and T4.7.

The online psychophysiological monitoring component provides information about the current state of arousal of a subject to the games that have been developed to use physiological data as input. These games form an integral part of the xDelia learning journey. Commercial sensor technologies were selected to provide the appropriate physiological data, the xAffect framework interfaces these sensors and abstracts the hardware layer by a middle-ware architecture. The latter also provides the interfaces for third party applications such as the the serious games. The xAffect software has a modular structure which can easily be extended by additional components such as new sensors or new algorithms.

Additionally, the document describes the data management framework which was developed within T4.7. The goal of this data management framework was to enable easy management of the data acquired during the psychophysiological studies conducted in xDelia. The data management framework not only manages the data, but also supports selective search for specific information in the study data to support the test of different hypotheses.



# Contents

<b>1 INTRODUCTION .....</b>	<b>7</b>
1.1 DOCUMENT PURPOSE AND SCOPE .....	7
1.2 LIST OF ACRONYMS .....	8
<b>2 CONTEXT OF THE MONITORING FRAMEWORK .....</b>	<b>9</b>
<b>3 IMPLEMENTATION OF A MONITORING FRAMEWORK FOR GAMES INTERVENTIONS 11</b>	
3.1 REQUIREMENTS.....	11
3.1.1 <i>Requirements to the Software Framework</i> .....	11
3.1.2 <i>Sensor Requirements</i> .....	12
3.2 THE xAFFECT SOFTWARE.....	13
3.2.1 <i>xAffect Core</i> .....	14
3.2.2 <i>xAffect Components</i> .....	15
3.2.3 <i>xAffect Control Framework</i> .....	16
3.2.4 <i>xAffect Setup</i> .....	17
3.2.5 <i>Licensing</i> .....	18
3.3 PHYSIOLOGICAL SENSORS .....	19
3.3.1 <i>EkgMove</i> .....	19
3.3.2 <i>Varioport-e</i> .....	20
3.3.3 <i>Pressure Sensitive Computer Mouse</i> .....	20
3.4 DATA MANAGEMENT AND EVALUATION TOOLKIT .....	21
3.4.1 <i>Unisens Data Base</i> .....	21
3.4.2 <i>Matlab Scripts for Data Analysis from Games</i> .....	26
3.4.3 <i>Matlab Scripts for Data Analysis from Interoception Study</i> .....	27
<b>4 VALIDATION OF THE SYSTEM USABILITY.....</b>	<b>29</b>
<b>5 CONCLUSIONS.....</b>	<b>30</b>
<b>6 APPENDIX A – REFERENCES.....</b>	<b>31</b>



# Figures

<i>Figure 3-1 – xDelia Monitoring Framework</i> .....	11
<i>Figure 3-2 – Architecture of the xAffect software</i> .....	14
<i>Figure 3-3 – Example for cascading of data processors</i> .....	15
<i>Figure 3-4 – Signal visualizer showing an ECG signal</i> .....	16
<i>Figure 3-5 – xAffect GUI</i> .....	17
<i>Figure 3-6 – Setup for arousal computation (discrete or continuous)</i> .....	18
<i>Figure 3-7 – Setup for marker button</i> .....	18
<i>Figure 3-8 – ekgMove</i> .....	20
<i>Figure 3-9 – Varioport-e</i> .....	20
<i>Figure 3-10 – Pressure sensitive mouse</i> .....	21
<i>Figure 3-11 – Categorization of Unisens entries</i> .....	22
<i>Figure 3-12 – Relevant Tables for Data Analysis</i> .....	22
<i>Figure 3-13 – Graphical user interface for Unisens data base</i> .....	24
<i>Figure 3-14 – Creating a new data base</i> .....	24
<i>Figure 3-15 – Example output for interoception data analysis</i> .....	28



# 1 Introduction

---

## 1.1 Document Purpose and Scope

The purpose of this document is to describe the implementation of the xDelia monitoring framework “xAffect”. The necessity to develop this framework was a result of the market analysis of available monitoring solutions illustrated in D15-5.1.1 and D15-5.1.2. The framework architecture has been described in detail in D19-5.2. The monitoring framework includes a description of the software framework which has been implemented for the learning interventions. Additionally, the commercial hardware, which has been selected for the studies according to the previous analysis of available commercial sensor technologies, is described.

The document is divided into the following sections:

- **Section 1 “Introduction”:** describes the structure and scope of this document.
- **Section 2 “Context of the Monitoring Framework”:** gives a description of the context in which the monitoring solution has been developed.
- **Section 3 “Implementation of a Monitoring Framework for Games Interventions”:** illustrates how the monitoring framework has been implemented to be used for the games interventions.
- **Section 4 “Validation of the System Usability”:** presents the results of the usability of the whole monitoring framework.
- **Section 5 “Conclusions”:** gives a brief summary about the key aspects of the deliverable.



## 1.2 List of Acronyms

API	Application programming interface
DSP	Digital signal processing
EC	European commission
ECG	Electrocardiogram
EDA	Electrodermal activity
HR	Heart rate
HRV	Heart rate variability
PPG	Photoplethysmogram
PTT	Pulse transit time
UDP/IP	User datagram protocol / internet protocol



## 2 Context of the Monitoring Framework

---

Within the project xDelia, several of the conducted studies require a customized solution for psychophysiological monitoring. Within the 1<sup>st</sup> and the 2<sup>nd</sup> year of the project, the focus of the conducted studies was set on research applications. The monitoring solutions provided for these applications have been described in previous deliverables (D15-5.1.1, D15-5.1.2 and D19-5.2). In year 3 the focus of studies has been shifted towards learning interventions. This document concentrates on the description of the monitoring framework which has been developed in order to fulfill the specific requirements from these learning interventions.

For the xDelia learning interventions several games have been developed. Two of these games implement online physiological feedback (“biofeedback”) which influences the gameplay:

- **Auction game:** The auction game simulates a market where a player has to buy or sell goods. For every good the player gets several price estimations from which he can calculate the true price of a good. With increasing arousal level of a person the game will get more difficult, e.g. the variance of the price estimations will increase. Therefore, the player should try to down-regulate his or her emotions in order to perform better in this game.
- **Space investors:** In the space investors game (formerly aiming game) the player has to shoot asteroids. If the player gets too aroused, the target will get unstable and get blurred which makes it even harder to shoot the asteroids. Down-regulating his or her emotions will help the player to hit the asteroids easier.

A more detailed description of the games developed for the learning interventions can be found in D14-4.2.2.

Besides the scenarios where the online feedback influences the course of the game, there is also an offline scenario where arousal values are not fed back into the game. This scenario can be used for control purposes.

In addition to commercial sensor technologies, this setup requires a software framework which consists of interface components and a hardware abstraction layer for the integration of the commercial sensors. Moreover, the framework has to provide an implementation of several analysis components as well as a communication and control interface to the games.

Although there are some existing commercial products as well as open source projects available for monitoring and analysis of certain biosignals (see D15-5.1.1 for an overview), a closer look at these technologies reveals several reasons that make them unsuitable within the context of xDelia. The following main deficits have been identified in the systems which are currently commercially available:

- Insufficient functionality (e.g. low sampling rate, limited number of available parameters)
- No documentation about the measuring methods and algorithms used for the data analysis
- No possibilities for online data analysis
- Closed source solutions which do not allow modification and do not provide an API for custom data access and analysis
- Unacceptable license terms and conditions concerning the rights to publish results and the ownership of acquired data



At least one of these deficits applies to all available solutions within the field. Thus, the requirements identified within the project use cases cannot be met by any available commercial or open source software tool, framework or tool chain. Therefore, the development of a custom solution is necessary (see section 4.2), as no commercial software components exist which covers all the needs arising from the requirements from the learning interventions.



## 3 Implementation of a Monitoring Framework for Games Interventions

For the game based studies conducted in year three, a modular monitoring framework had to be developed. The requirements for the monitoring framework are described in section 3.1 . Subsequently, section 3.2 illustrates the implementation of the monitoring framework. Section 3.2.5 provides a description of the sensors which have been integrated into the framework in order to record psychophysiological signals.

### 3.1 Requirements

The monitoring framework which has been provided for the learning interventions consists of two main parts. Section 3.1.1 outlines the requirements which mainly drove the development of the software. In section 3.1.2 the requirements for the sensors which were selected for the learning interventions are described.

#### 3.1.1 Requirements to the Software Framework

The framework which has been developed for the game based studies in xDelia had to fulfill different requirements arising from the usage of serious games for the learning interventions.

Figure 3-1 illustrates the scenario in which the software framework will be used. The digital signal processing (DSP) component in the middle connects the physiological sensors with the game. Purpose of this component is to receive signals from multiple physiological sensors and to extract arousal features from these signals. After computing the arousal features, the values are sent to the game in order to influence the course of the game and to provide feedback to a person.

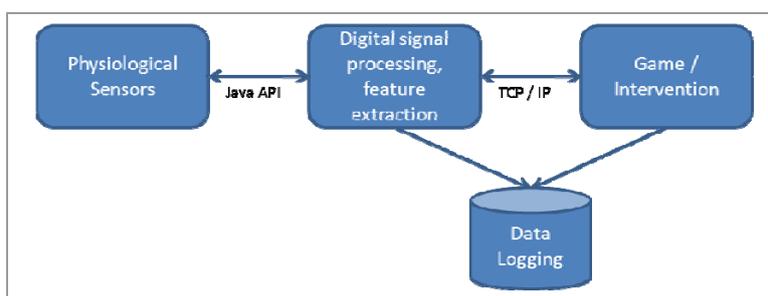


Figure 3-1 – xDelia Monitoring Framework

As the figure shows, the main functionality of the DSP component has to include the following general features:

- Connectivity for different sources: It has to be able to connect to and process data from different physiological sensors (e.g. ECG, EDA, etc.)



- An application programming interface (API): It must provide a connection to third party applications (such as a game) and communicate information about the current state of arousal of a user to these applications.
- Logging functionality: It has to be possible that the DSP component logs to the same data set as the game to facilitate later data analysis.

In detail, this implies the following functional requirements:

- Algorithms for preprocessing of incoming physiological data have to be provided.
- It has to be possible to compute arousal values from physiological data.
- There has to be a possibility to visualize data.
- It has to be possible to receive control commands by the game.
- Arousal values have to be sent to the game.
- It has to be possible to annotate data the data during a recording session.
- There has to be a possibility to provide predefined setups for different games.
- It has to be possible to log raw data as well as data from any processing step.
- Data logging has to be done in Unisens data format.
- There has to be a graphical user interface to use the application outside the games.

Moreover, the following non-functional requirements have to be fulfilled:

- The algorithms used by the system have to be optimized for high performance in order to provide appropriate online feedback during the game.
- There has to be a standardized interface to communicate with the games.
- There has to be a standardized interface in order to integrate different kinds of sensors and to handle their respective data formats.
- The developed software has to be able to run on a standard Windows PC that is equipped with the required sensor peripherals.
- The architecture has to be modular which allows exchanging individual components (e.g. algorithms for signal processing) if required.
- If a component is exchanged this does not interfere with other components.

### 3.1.2 Sensor Requirements

In the course of the restructuring of xDelia after the Interim Review, the work in WP5 has been discontinued and parts of it were integrated into WP4. One of the consequences was that it was decided that no specific sensor hardware development should take place in the context of xDelia. Instead, commercially available sensors had to be used for the psychophysiological monitoring within the project.

The recording system used for the game based studies had to be unobtrusive and discrete, so that participants are not disturbed during the game or even influenced by the system. Therefore, a sensor solution had to be provided which is unobtrusive yet still able to record data with a quality which is high enough for further analysis. The system should at least be able to record ECG with sufficient quality to enable post-hoc HRV analysis. For instance, the American Heart Association



recommended that the ECG signal should at least be recorded with a sampling frequency of 500 Hz and a resolution of 12 bits (AHA, 1975). The Task Force of the European Society of Cardiology and the North American Society of Pacing (1996) also recommended a sampling rate between 250 – 500 Hz.

As using a multi-modal sensor setup in the gaming environment is likely to provide a more complex and robust measure for the emotion regulation capabilities, later versions of the monitoring solution should be able to measure additional parameters like EDA or pulse transit time (PTT) as well. Moreover, new algorithms developed by other research groups can easily be implemented.

## 3.2 The xAffect software

As a consequence of the requirements illustrated in section 3.1.1 , the xAffect software for affect recognition has been designed. The architecture of the xAffect software was driven by the requirement that the component has to be very flexible to configure. The software therefore consists of a central core and several modules that deal with a specific aspect of functionality. This allows that xAffect can be modified such that different sensors can be integrated as well as signal processing algorithms and the software can easily be integrated into the learning interventions. Moreover, the software has been implemented such that the algorithms which are used for feature computation can easily be extended or exchanged.

The xAffect software has been implemented in Java as a single user application. While Java itself is platform independent, the usage of native libraries is bound to a specific platform. Nevertheless, Java provides a large number of interfaces to other languages and there are implementations for most common communication protocols. The Unisens library is implemented in Java as well. This simplifies the implementation of data storage and data logging.

Figure 3-2 shows the architecture developed for the xAffect software. With its modular design consisting of data sources, processors and sinks on the component side, it can easily be extended to integrate new sensors (data sources) and analysis algorithms (data processors). A new sensor can be integrated by writing the corresponding sensor adapter. If the sensor can send data to the data dispatcher, the data will be available to all data processors / algorithms that can deal with the data content. The limitations imposed by the different data formats are solved before the physiological signals enter the core of this system. A new data processor with a specific signal processing adapter can be used for all kinds of relevant sensors. Moreover each data processor can use the output of other data processors. The components can be arranged in a cascade based on accompanying metadata objects. In consequence different data processing tool chains as well as different sensor setups are possible.

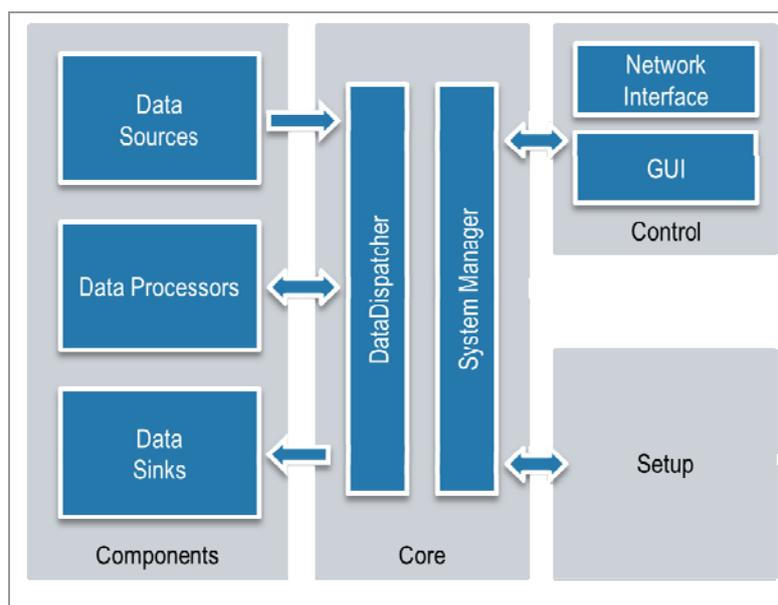


Figure 3-2 – Architecture of the xAffect software

The main design decision for the architecture of the framework was to use a common data format. The specification of the data format consists of two parts. The first part provides a general container for sensor and game data and the corresponding timestamps. The second part is a metadata object that describes the content in the container. The structure of the metadata object builds on the existing experience with the definition of the Unisens data format. The advantages of the Unisens data format have been described in D19-5.2.

In summary, the main benefits of the xAffect software are that the system can integrate a variety of sensors and (if necessary) real time analysis algorithms, thus supporting configurations for very complex scenarios. The integration effort of a new component is limited to writing an adapter for this component and adjusting the corresponding setup. Broader changes of the system or any changes concerning the core will not be necessary.

### 3.2.1 xAffect Core

Purpose of the core of the xAffect software is to initialize all elements and connecting them by routing the data. It consists of two main parts: the system manager and the data dispatcher. The system manager is responsible to initialize the setup. The data dispatcher takes care that data is passed to the respective components as specified in the selected setup. All components from the different elements can register for a specific set of data (e.g. the component which computes heart rate can register for the raw ECG signal). They provide a metadata object to specify the data they can process. In consequence, the data dispatcher can compare registered meta-documents with the incoming data descriptions and thus decide about the routing of the data.



## 3.2.2 xAffect Components

Three different kinds of components exist which communicate with the core of the application: data sources, data processors and data sinks. The components which have been integrated into xAffect are described in the following sections.

### 3.2.2.1 Data Sources

The data sources reflect the sensors, signal generators or timers which are integrated into the system. For every device that is connected to the core, a sensor adapter implements an interface in order to convert the data such that they can be processed by the data processors. Moreover, the sensor adapter transmits the control commands from the core to the sensor in the format which is required by the sensor. So far, sensor adapters have been implemented to integrate the ekgMove sensor, the Varioport-e and a pressure sensitive mouse. Moreover, there exists a marker source, a signal generator and a timer to trigger other processes.

### 3.2.2.2 Data Processors

The data processors include a large number of signal processing adapters. Each adapter implements an interface to a signal processing algorithm. Data is converted such that it can be processed by the corresponding algorithm and the result of the algorithm is in the following transferred in a format which can be understood by the core of the xAffect software.

The online processing of the incoming data requires parallel computation of features wherever possible. Therefore, each data processor is running in an independent thread. The data dispatcher (see section 3.2.1) is the synchronizing object. If new data is sent by a sensor, the data dispatcher starts all matching data processors at the same time. When an algorithm finishes, it returns the result again to the data dispatcher. This can be repeated several times before the final features are sent to the data dispatcher.

For xDelia, a large number of data processors have been implemented. For instance there exists a large number of processors in order to extract information from incoming ECG data like QRS detection or computation of heart rate.

It is possible to combine multiple data processors in order to achieve the desired output. Figure 3-3 shows an example how different data processors can be cascaded in order to compute HRV features from the raw ECG signal. Moreover, it is also possible to combine the arousal values computed by different processors for multimodal arousal recognition.

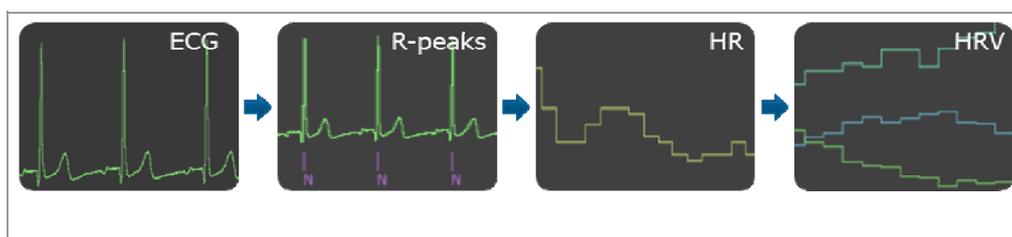


Figure 3-3 – Example for cascading of data processors



### 3.2.2.3 Data Sinks

The data sinks are responsible for data logging and data transmission. Currently there exist sinks for data logging in Unisens data format, signal visualization and an interface to submit data to the games interventions via UDP/IP. Figure 3-4 shows the data viewer displaying an ECG signal.



*Figure 3-4 – Signal visualizer showing an ECG signal*

Figure 3-4 shows an ECG signal displayed by the signal visualizer.

## 3.2.3 xAffect Control Framework

The control framework provides an interface to configure the xAffect software. The control framework requests the information about the configuration from the setup element. There are two different ways how the control framework can be used:

- a) A graphical user interface (GUI) allows the configuration (e.g. start or stop of the recording) of the whole digital signal processing component. Resulting features can be printed on a console.

*Figure 3-5 shows the graphical user interface of the xAffect software*

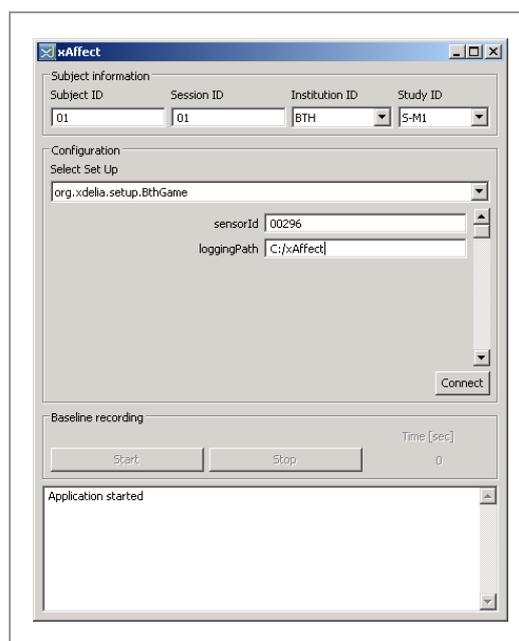


Figure 3-5 – xAffect GUI

- b) A network interface which provides the same functionality to external components like the space investors game or the auction game.

The control interface does not generate new data for the dispatcher but sends the data to a console or via UDP protocol to a game.

### 3.2.4 xAffect Setup

The setup provides the required setup parameters for sensors, algorithms and data sinks. It is initialized by the SystemManager. The setup is implemented such that several predefined configurations can easily be selected via the GUI or the network control.

For the games interventions developed in WP4 the following setups have been predefined:

- **Discrete arousal computation:** The raw ECG signal which can either be recorded by the ekgMove or the Varioport-e sensor is processed by the QRS-detection processor. After QRS detection, heart rate is computed. From the heart rate information, discrete arousal values ranging from 1 to 5 are derived.
- **Continuous arousal computation:** Same setup like discrete arousal computation but on a continuous scale from 0 to 0.99.
- **Marker Setup:** In this setup only the ECG signal is recorded. When this setup is chosen a separate window will open which allows setting labeled markers with a timestamp.

Figure 3-6 illustrates how the components are cascaded for the computation of arousal values. Depending on whether discrete or continuous arousal values have to be computed, either the data processor for discrete arousal computation or the data processor for continuous arousal computation is used.

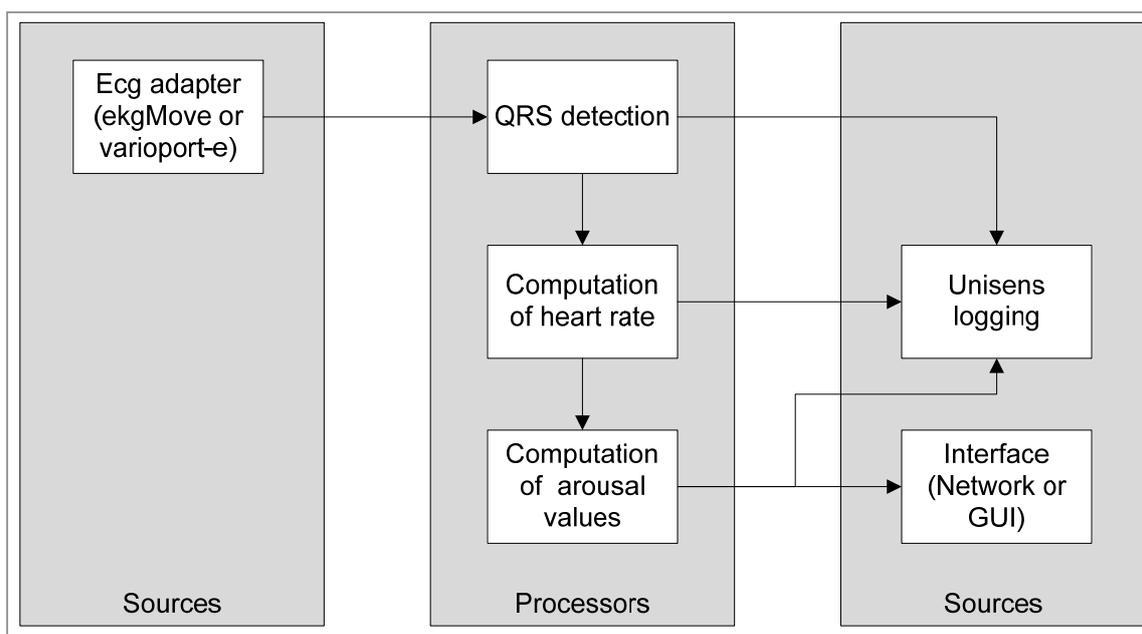


Figure 3-6 – Setup for arousal computation (discrete or continuous)

The setup to use the marker button is illustrated in Figure 3-7. In the current configuration, the only data sink is the Unisens logging adapter as no data needs to be provided via the GUI or an interface.

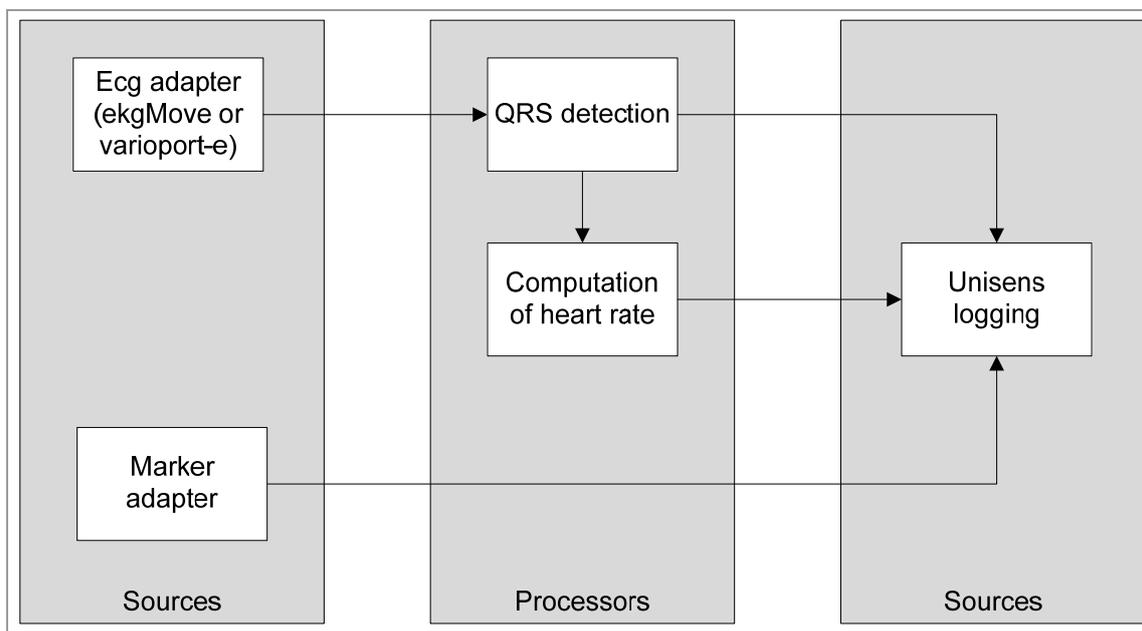


Figure 3-7 – Setup for marker button

### 3.2.5 Licensing

The core components of the xAffect software will be released under original BSD license. The most important modules of the framework will be provided for download at <http://www.xaffect.org>. The additional components such as the computation of arousal from heart rate will be provided to the



project partners separately as closed source plugins. Also, components and modules that are going to be developed after the xDelia project will be made available via the website as they evolve.

The following paragraphs reproduce the licensing conditions:

“Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement: "This product includes software developed by FZI Forschungszentrum Informatik, Karlsruhe, Germany and its contributors."
4. Neither the name of the FZI Forschungszentrum Informatik nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE FZI FORSCHUNGSZENTRUM INFORMATIK AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE FZI FORSCHUNGSZENTRUM INFORMATIK OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.”

## 3.3 Physiological Sensors

This section provides a description of the physiological sensors which have been integrated into the xAffect software so far. Besides the ekgMove sensor a small mobile polygraph as well as a pressure sensitive mouse have been integrated.

### 3.3.1 EkgMove

The ekgMove sensor is a wearable ECG belt including a 3-axis-acceleration sensor which is worn at the chest. In contrast to most medical ECG holter recorders, this sensor is very comfortable and unobtrusive to wear. Compared to other wearable systems which use dry electrodes, evaluations have shown that the obtained signal quality is very high. Furthermore, raw data (not preprocessed) is accessible without problems. The Bluetooth interface allows wireless data transfer to a PC / Smartphone. As the system is open, it can easily be configured according to the needs of a specific study.



Figure 3-8 – ekgMove

### 3.3.2 Varioport-e

In contrast to ekgMove, the varioport-e allows to record multiple parameters during the game interventions. Varioport-e allows to record ECG, EDA, BVP and 3-axis acceleration simultaneously. The system is designed such that EDA can be recorded at the plantar region such that no electrodes have to be attached at the hand. Moreover, the system is very small and unobtrusive. For real time data analysis the recorder can be connected to the PC with a fiber-optic cable, this way the subject's safety is also guaranteed.



Figure 3-9 – Varioport-e

The system will be used in order to evaluate suitable features for multiparameter arousal recognition.

### 3.3.3 Pressure Sensitive Computer Mouse

The pressure sensitive mouse was developed at FZI Forschungszentrum Informatik. The mouse is able to measure the pressure with which a person presses the mouse button. Moreover, the squeeze which is applied to the side of the mouse is monitored.



*Figure 3-10 – Pressure sensitive mouse*

One main advantage of this mouse is that from the outside it looks like a normal computer mouse (see Figure 3-10). Therefore, the user does not see that mouse pressure is recorded and will not manipulate the pressure on the mouse button intentionally.

## 3.4 Data Management and Evaluation Toolkit

The studies conducted within xDelia will include a large number of subjects of which data will be recorded within multiple sessions. This causes a large amount of data to be generated. To have a possibility to search for certain information in this data the data management and evaluation toolkit has been developed. The data management and evaluation toolkit is based on the Unisens format, a relational database and a collection of Matlab scripts that provides the means for evaluation of the data collected during the game interventions.

### 3.4.1 Unisens Data Base

Originally, the Unisens Data Base has been developed in order to index all meta-information within the data sets. For xDelia it has been extended such that it is now also possible to search for specific events from the studies and export the results. The meta data as well as the event specific content is stored in a data base.

#### 3.4.1.1 System Requirements

The Unisens data base works on every windows PC where Java 1.6 (x86) or higher is installed. Moreover, the database requires MySQL Server 5.5 (MySQL Community Server, Typical Installation) to be installed on the PC or an accessible MySQL server in the network. MySQL is a powerful relational data base management system which is available open source.

#### 3.4.1.2 Structure of the Unisens Data Base

Entries in a Unisens data set can be categorized as shown in Figure 3-11. Timed entries contain information which is related with timing while custom entries are general entries for a specific data set. Timed entries can be subdivided into measurement entries and annotations. Measurement entries can either be signals (i.e. evenly sampled data such as ECG) or values (i.e. unevenly sampled data like heart rate). Annotation entries include event entries. In contrast to measurement entries which



include only information about time and measurement values, event entries include annotation information. In the games event entries have for instance been used to mark beginning and end of a level.

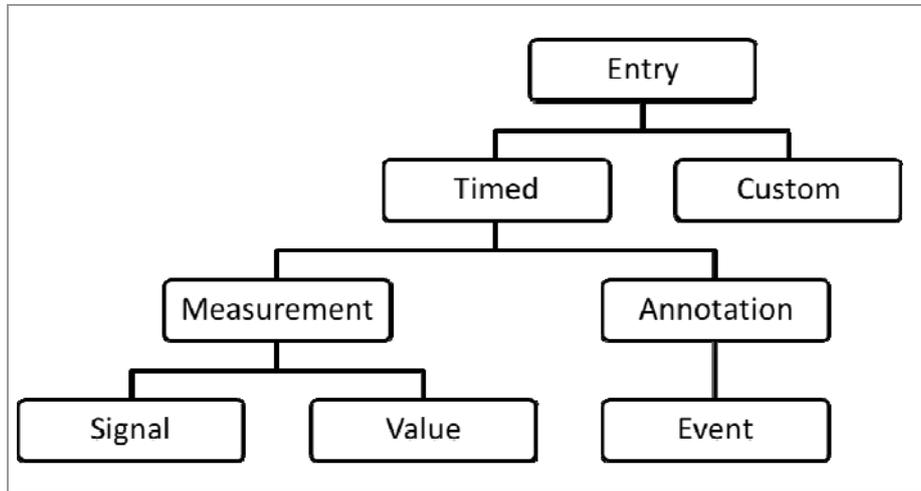


Figure 3-11 – Categorization of Unisens entries

The event categorization is also reflected in the Unisens data base which is organized in a several tables. Figure 3-12 shows the tables which are relevant for later data analysis.

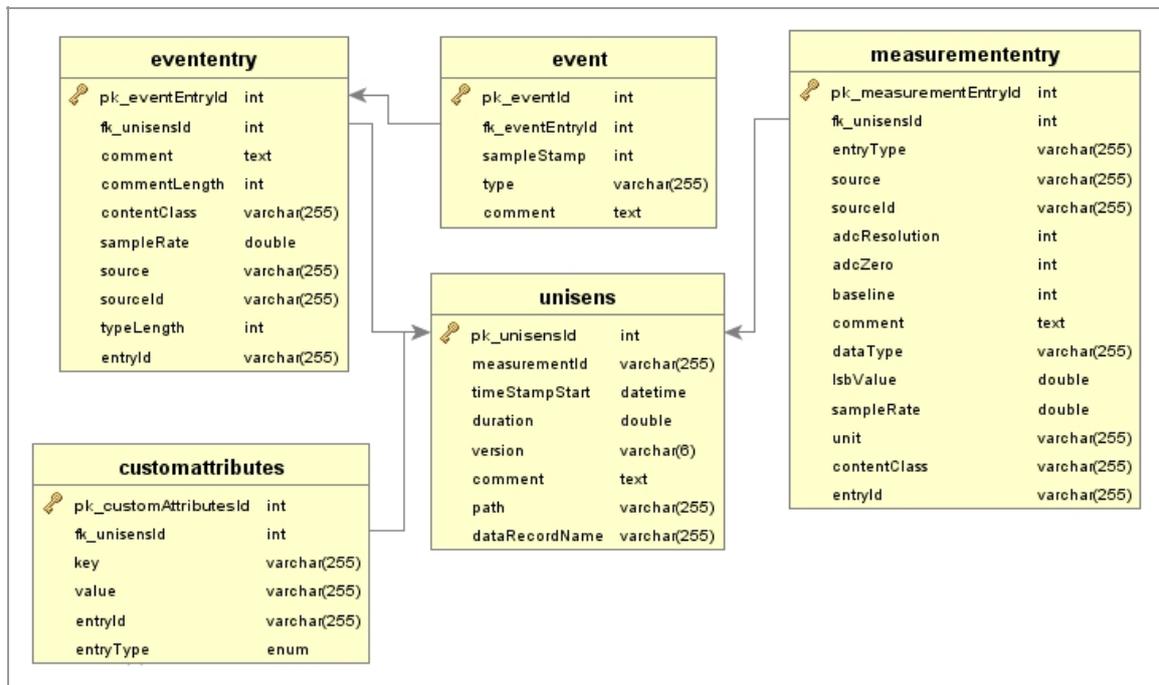


Figure 3-12 – Relevant Tables for Data Analysis

The tables are organized as follows:

- unisens: contains an entry for every data set which was found in the folder. Every data set has a unique ID to be referenced by other tables.



- **measuremententry:** all timed entries found in the Unisens data sets (stored in table `unisens`) are listed in this table. They can be assigned to the corresponding Unisens data set using the attribute `fk_unisensId` which corresponds to the primary key `pk_unisensId` of the table `unisens`. The table contains all information about the entries like entry type (i.e. signal or value), content class (e.g. ECG), source (e.g. `ekgMove`) or the data type of the entry.
- **evententry:** this table contains information about annotation entries for the data sets, i.e. one row for every event entry. The attribute `fk_unisensId` in this table corresponds to the attribute `pk_unisensId` in the table `unisens`. The table includes information like the content class of the event and the data source.
- **event:** in this table all events which are included in the event entries are listed. Using the `fk_eventEntryId` they can be assigned to the corresponding evententry. For every event, the `samplestamp`, the type of the event and the comment are stored in the table.
- **customattributes:** custom attributes stored in the `unisens.xml` are stored in this table. The custom attributes contain for instance information like the number of decisions taken correctly in the auction game or the overall money earned. The attributes can be assigned to the corresponding data set using the `fk_unisensId`. Custom attributes can be related to the hole data set or to a specific entry.

As the overview shows, it is possible to join all tables using the unique keys of each table (e.g. the `unisensId`). This allows building a large number of queries for data analysis.

### 3.4.1.3 Using the Unisens Data Base

To provide an easy access to the Unisens data base, a MySQL client has been developed in Java. It provides a graphical user interface to view the data. Data from study can be imported to the data base using the import function. After specifying a folder, all information from the Unisens data sets within this folder will be imported. Once data is imported, the desired information can be extracted using SQL queries.

*Figure 3-13* shows the user interface of the Unisens data base. It consists of two main parts. In the upper part of the window queries can be executed to receive information from the data base. There are three different ways to execute a query:

- **Predefined queries:** The Unisens data base provides the possibility to save predefined queries. The predefined queries can be selected from the pull down menu. It is also possible to edit predefined queries before executing them.
- **Query history:** The query history shows the queries which have been executed since the program has started. It is emptied once the program is closed.
- **Query window:** In the query window all kind of queries can be defined using SQL syntax. This allows to specify a large number of different queries for data analysis. All queries can be saved to the list of predefined queries for later usage.

The lower part of the window shows the results from the SQL queries. In the example illustrated in *Figure 3-13*, all events where R-peaks in the trigger file have been labelled as normal (i.e. event type = 'N') are listed with the corresponding sample stamp. Using the `eventId` it is possible to assign them to the corresponding data set.



The results from the result window can easily be exported by copy and pasting them to another application such as Excel or a text file.

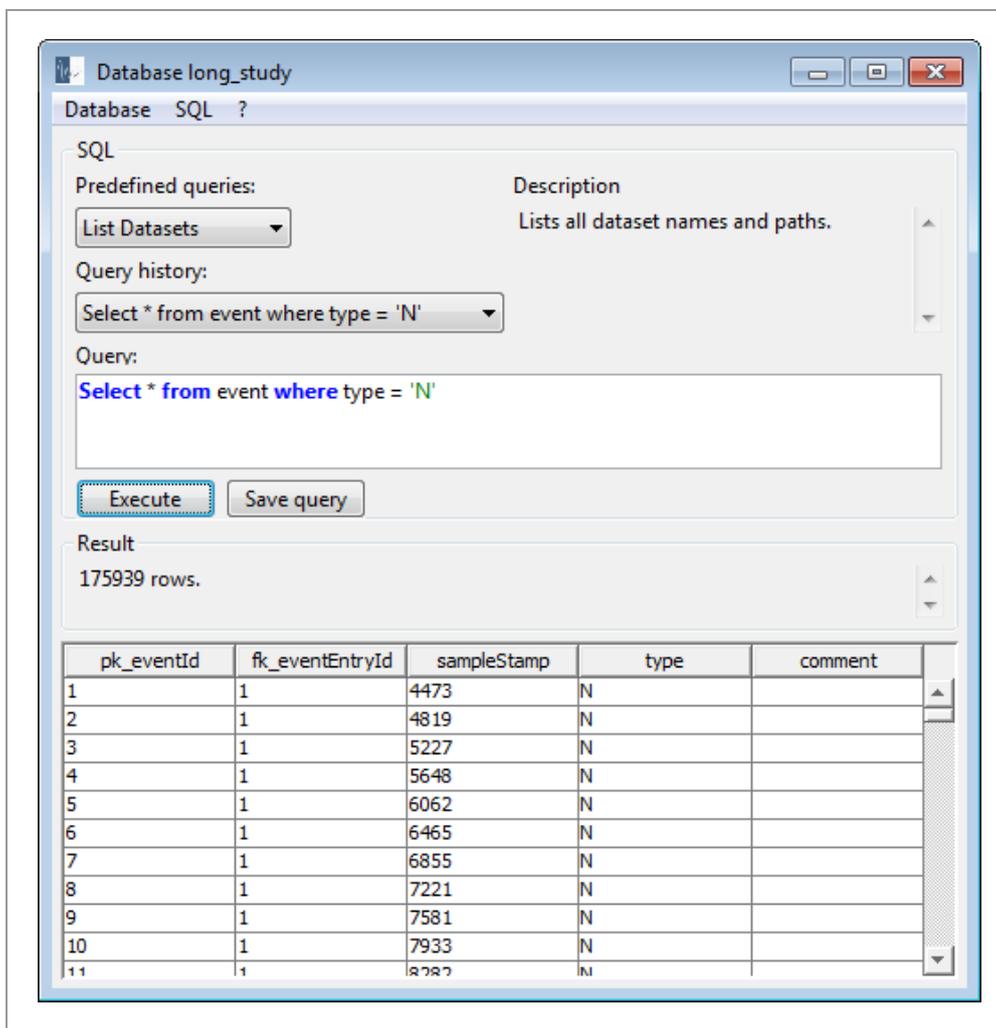


Figure 3-13 – Graphical user interface for Unisens data base

To import data for analysis, first a new data base has to be created. This can be done by selecting Database -> Create Database. In the next step a name for the data base has to be defined.

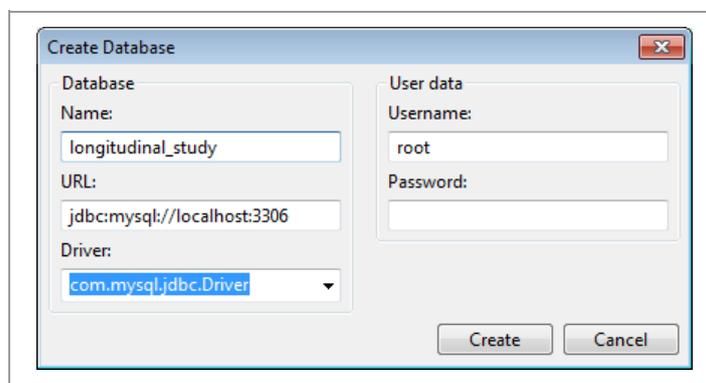


Figure 3-14 – Creating a new data base



Figure 3-14 shows how to fill in the form to create a data base named 'longitudinal\_study'. In the next step, the data from the longitudinal study has to be imported. This can be done by selecting Database -> Import XML. Next, the folder which contains the data from the study has to be selected. If events shall be imported as well, the Box 'Import event-data' has to be checked. Events are only available for later analysis if they are imported to the data base.

### 3.4.1.4 Example Queries

The following query lists the mean value of the SAM ratings during the auction game per level including the number of participants who rated this level. The number of participants might be an interesting information as not all participants reach higher levels.

```
select type as round, avg(event.comment) as 'mean SAM rating', count(*) as count
from event, evententry
where entryId = 'samdata.csv'
and fk_eventEntryId = pk_eventEntryId
and source = 'BTH-Auction Game'
group by type
```

A query which returns the mean SAM rating per participant during the auction game including the number of levels he or she rated (i.e. reached) during the game can be done as follows:

```
select dataRecordName as 'subject ID', avg(event.comment) as 'mean SAM
rating', count(*) as count
from event, evententry, unisens
where entryId = 'samdata.csv'
and fk_eventEntryId = pk_eventEntryId
and fk_unisensId = pk_unisensId
and source = 'BTH-Auction Game'
group by dataRecordName
```

To list the mean heart rate for each level and each participant with the corresponding sam rating during this level, a query can be formulated like the following example:

```
select sam.dataRecordName as 'subject ID', sam.type as round, sam.comment as
'SAM rating', hr.comment as 'mean HR'
from
(select dataRecordName, type, event.comment
from event, evententry, unisens
where entryId = 'offline_level_hr.csv'
and fk_eventEntryId = pk_eventEntryId
and fk_unisensId = pk_unisensId) as hr,
(select dataRecordName, type, event.comment
from test_db.event, test_db.evententry, test_db.unisens
where entryId = 'samdata.csv'
and fk_eventEntryId = pk_eventEntryId
and fk_unisensId = pk_unisensId
and source = 'BTH-Auction Game') as sam
where sam.type = hr.type
and sam.dataRecordName = hr.dataRecordName
```



## 3.4.2 Matlab Scripts for Data Analysis from Games

We developed Matlab analysis tools for the offline feature computation of the data, which was collected in the psychophysiological studies conducted in xDelia.

### 3.4.2.1 Configuration

For the data analysis based on the Matlab scripts there exists a configuration file where several parameters of the processing steps can be specified. For instance it can be selected which computation steps shall be performed. Moreover the threshold for the artifact detection as well as window size and window shift for computation of HRV can be defined. Additionally the configuration file allows to specify the names of the output files.

### 3.4.2.2 Functionalities

The xAffect analysis tools provide the following functionalities:

- **Re-computation of R-peaks:** the algorithm used for R-peak detection during the recording is optimized to high performance. For detailed ex-post analysis of heart rate and HRV it is important that R-peaks have been detected accurately. As computation time is not crucial for offline analysis, an improved version of the OSEA algorithm has been implemented which detects the R-peaks more accurately than the online version used in the xAffect framework. The R-peak detection also includes a detection of ectopic beats.
- **Re-computation of heart rate:** based on the re-computed R-peaks, heart rate is computed again to correct possible errors based on the online R-peak detection.
- **Computation of heart rate per level:** this function computes the mean heart rate for each level played in the space investors or the auction game. Moreover, the mean heart rate for the baseline phase is computed. Information about start and end of a level are taken from the files logged by the games.
- **Computation of HRV parameters:** for a given window size and a given window shift (specified in the configuration file), the following HRV parameters are computed: VLF, LF, HF and LF/HF-Ratio. According to the suggestions from the Task Force of The European Society of Cardiology and The North American Society of Pacing and Electrophysiology (1996): the minimum window size is 300sec. As most levels last shorter than 300s, a level based computation of HRV parameters was not implemented.

There are two different modalities to process the data. Using the function `xaffected_convert_data` single data sets can be converted. The function `xaffected_batch_convert` can be used to convert multiple data sets at one time. In the batch mode a specified folder is browsed for Unisens data sets. All Unisens data sets containing valid data are then processed according to the settings specified in the configuration.

### 3.4.2.3 Output files

All data computed in the respective processing steps is saved to the same Unisens dataset as the original data set. Depending on the configuration settings the following output files are produced (note: file names can be changed in the configuration file):



- `offline_trigger256.csv`: contains an event for every R-peak detected from the raw ECG signal. First column contains `samplestamp`, second column the type of the R-peak.
- `offline_hearttrate.csv`: contains the heart rate computed from the trigger list. Heart rate values are saved as `eventEntries` every time an R-peak is detected. The first column of the file denotes the sample stamp, the second column the value of the heart rate at this point.
- `offline_level_hr.csv`: gives information about the mean heart rate of a level or the baseline period. At the end of the baseline / each level, a `valuesEntry` is created. Again, the first column contains the sample stamp while the second column contains the mean heart rate during the time frame. The last column gives information for which time frame (i.e. baseline or level no.) the mean value has been computed.
- `offline_hrv_power_lomb.csv`: this file contains frequency based information about hrv parameters. Each value represents a window of the window size specified in the configuration file. The values are set at the beginning of each window. The HRV data is saved as an evenly sampled signal with a sampling frequency of  $1/\text{window\_shift}$ . The signal consists of three channels: VLF, LF and HF. Each channel is saved in a separate column.
- `Offline_lfhf_ratio_lomb.csv`: contains the LF/HF-ratio based on the HRV computations for the window size and window shift specified in the configuration file.

### 3.4.3 Matlab Scripts for Data Analysis from Interoception Study

In the interoception experiment participants were asked to press the mouse button every time they think their heart is beating. After 35 perceived beats (i.e. after 35 mouse button presses) they had to pause for one minute before they could try again.

For automated analysis of the data from the study, Matlab based scripts have been programmed. In the analysis scripts first a specified path is parsed for valid Unisens data sets. For the data sets an R-peak detection is performed. After the R-peaks are detected, the number of heart beats between the first and the 35th mouse button press is calculated. In the input folder a file containing the results for all participants is generated.

	Subject ID	Trial No.	# beats	Error
	A	B	C	D
1	LONG_OU_UKLN37-1	1	102	191.428571
2	LONG_OU_UKLN37-1	2	80	128.571429
3	LONG_OU_UKLN37-1	3	64	82.857143
4	LONG_OU_UKLN37-2	1	64	82.857143
5	LONG_OU_UKLN37-2	2	49	40
6	LONG_OU_UKLN37-2	3	33	-5.714286



---

*Figure 3-15 – Example output for interoception data analysis*

---

Figure 3-15 shows how the result file looks like after parsing data of two participants with three trials each.



## 4 Validation of the System Usability

---

For the learning interventions the xAffect software had to be able to transfer data from the ekgMove sensors to the games developed for the learning interventions.

To receive the sensor data, an adapter has been developed which implements an interface to the xAffect software. This enables xAffect to receive the raw ECG data for further signal processing.

Moreover, the xAffect software has successfully been integrated into the games for the learning interventions which use arousal to influence the course of the game. The games are able to send control commands to the xAffect software for instance to start the recording or to initialize the baseline period. Additionally, the initialization of parameters like the study ID or the sensor ID can be passed to the xAffect. On the other side the xAffect software sends information about the computed arousal level to the games. Using the xAffect framework several studies have successfully been conducted with a large number of subjects.

Further details about the evaluation of the monitoring framework will be presented in D20.



## 5 Conclusions

---

In this deliverable the monitoring framework developed for the learning interventions conducted in year 3 of the project is described. The document starts with an analysis of the requirements arising from the games. In the following the monitoring framework which resulted from these requirements is described. The monitoring consists of the xAffect software and sensor hardware for physiological monitoring. The xAffect software provides a modular framework which is able to compute arousal values from incoming physiological data and pass this arousal information to the games. The hardware selected for the monitoring framework was selected in order to be as unobtrusive as possible but it still had to provide a high signal quality. The document concludes with first results of the integration of the monitoring framework into the learning interventions.



## 6 Appendix A – References

---

AHA Committee, “Recommendations for standardizations of leads and specifications for instruments in ECG and VCG,” *Circulation*. Vol. 52. July-Dec. 1975.

Task Force of The European Society of Cardiology and The North American Society of Pacing and Electrophysiology (1996): Heart rate variability. Standards of measurement, physiological interpretation, and clinical use. *Eur Heart J* 17: 354-381.