University of West Bohemia
Faculty of Applied Sciences
Department of Computer Science
and Engineering

Master Thesis

Software tool for management of neuroinformatics data

Pilsen, 2013 Jan Řeřicha

The original assignment

ACKNOWLEDGEMENT

I would like to thank Roman Mouček for his kind supervision of my master thesis. I appreciate his helpful advice and generous help as well as useful materials on the subject. Also, I would like to thank Pavel Mautner for data analyzing advises and also the volunteers who participated in the experiment.

DECLARATION

I hereby declare that this master thesis is completely my own work and that I used only the cited sources.

Pilsen, May 15, 2013 ……………………….

Jan Řeřicha

ABSTRACT

This master thesis deals with design, performance and evaluation of driver's attention experiment. During monotonous driving the EEG signal, respiratory frequency and skin conductance are recorded. The goal is to find relation between rising fatigue and changes in human physiological functions. In relation with recorded experimental data the declaration and implementation of a new data model for electroencephalography is defined. The purpose of model specification is to collect a set of complex meta-data information necessary to interpret measured data correctly. The data model is realized using Hierarchical Data Format (HDF5). The developed software realizing conversion of existing recorded data into data model in the HDF5 format is implemented using C# and .NET wrapper.

CONTENTS

1 INTRODUCTION

One of the priorities and key factors of research in driver's attention is to limit the number of accidents caused by fatigue or drowsiness of drivers. During driving the emphasis is focused on driver's fast reaction and quick decision. Over time driving is more boring and more tiring. Fatigue increases in comparing with the beginning of driving. Attention gradually decreases and the driver's reaction time is prolonged.

The experiment realizes the simulation of real tiring environment - driving a car simulator along a monotonous highway track. The analysis is based on measuring the latency time of the auditory ERP P3 component. Together with analyzing brain activity the respiratory frequency is measured and evaluated. Correlation between these two physiological signals is analyzed. Biomedical signal is especially useful for collection of information about the body physiological processes and their response to stimulation. Provided information could be useful in detection of risk situation mainly reaching the fatigue limit after that the micro sleeping is probable. The assumption is that during experimental driving, decreasing of driver attention will occur and the peak latency of reaction to target auditory stimuli will prolong and in relation with prolongation of latency the respiratory frequency will decrease.

Experimentation requires storing recorded data. There are numerous models or their concepts representing and storing electrophysiological data. Goal of this work is to merge ideas and create a new model of electroencephalography data which are recorded in the neuroinformatics laboratory at University of West Bohemia. The model could be used as a base or an element of a more generic and complex model for electrophysiological data.

At the beginning of the thesis theoretical background about the human brain and EEG[1]/ERP[2] experiment is presented; then the basics about respiratory measurement are mentioned. The next chapter deals with experiment design and realization. This part is followed by evaluation of the experiment with summary of the results and diagrams. After chapters about experiment, the chapters about the data model and software implementation are situated. There is a chapter about data model design and chapter focuses the software architecture and implementation. The last chapter summarizes the course and results of the experiment.

---

[1] EEG – Electroencephalography is the recording of electrical activity along the scalp

[2] ERP – Event-related potentials are a brain electrophysiological response to a stimuli

2 THEORETICAL BACKGROUND

2.1 Human brain

At the beginning of the text it is good to acquire and be familiar with essential theoretical facts about the human brain and its parts.

The brain consists of several main parts [1]. For EEG experiments the largest portion – cerebrum is the most important. It lies in front-top of the brainstem and in humans is the best-developed of the major divisions of the brain. Cerebrum can be divided into two parts – right and left hemisphere. Further it has four sections known as (as shown in Figure 2:1) lobes: temporal, occipital, parietal and frontal [2], [3].

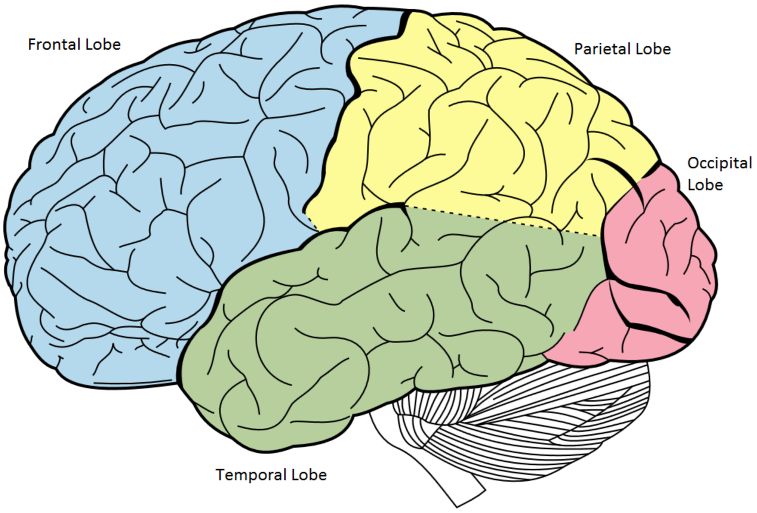


Figure 2:1 - The lobes of the brain [4]

Frontal lobe

This lobe has many functions most of which center on regulation of social behavior and the emotional control, logical thinking, planning and creativity. Other functions that the frontal lobes do are control of sequencing of events, which is the ability to plan a series of movements needed to perform a multi-step task.

The lobe is located at the front of the central sulcus where it receives information from other lobes of the brain.

Parietal lobe

This lobe is responsible for processing sensory information from various parts of the body. The next functionality includes sensation, perception and spatial reasoning. For example, here are some of the functions: speech and processing the human’s five senses.

The parietal lobe resides in the middle section of the brain, which is located in the cerebral hemisphere, behind the central sulcus, above the occipital lobe.

Occipital lobe

Occipital lobe is the smallest lobe of brain. The center of visual perception system is here. It is a primary visual processing center of the brain and it includes recognition of colors. This lobe is located near the back of the skull.

Temporal lobe

It responses for the interpretation of sounds and smell, as well as for formation of visual, verbal and long-term memories. But the primary function is processing of semantics in vision and speech. The lobe is on the bottom of the brain and it is located on both sides of the brain close to the ears.

2.2 EEG

Electroencephalography (EEG) is a diagnostic method used for recording electrical activity from the brain along the scalp. The technique is based on sensing changes of electrical potentials using electrodes and following evaluation of obtained EEG records.

2.2.1 Measurement

Experimental measurement is realized using noninvasive electrodes attached to a special cap. Usually, measurement is unipolar, which means that the second pole of electrodes is the same for all channels and it is called a reference electrode. The distribution of the electrodes usually used to measurement is the 10-20 international system. This is the standard naming and positioning scheme for EEG applications (Figure 2:2).

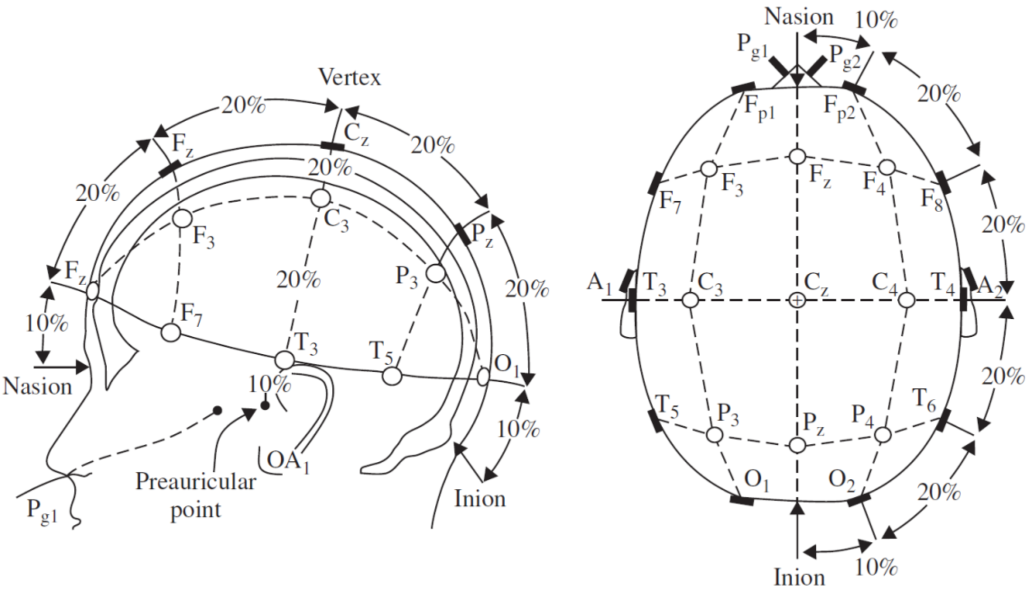


Figure 2:2 -The international 10-20 system [5]

A raw record of the EEG signal is very complex; it contains signals of many brain neural activities. One possibility to get specific neural responses and requested data from the record is to focus on specific neuronal responses connected to specific sensory stimuli. These specific target responses are known as event-related potentials (ERP). Introduction and more information about this issue can be seen in the next chapter.

2.2.2 Brain rhythms

The brain electrical activities are displayed in the form of brainwaves. There are four categories of these brain waves [6]. Rhythms can be seen in Figure 2:3.

Beta

Beta rhythm is often associated with thinking and active concentration. It occurs when the brain is aroused and actively engaged in mental activities. The frequency of beta waves ranges from 13 to 30 Hz.

Alpha

The origin of these waves is during wakeful relaxation with closed eyes. The rhythm is reduced with open eyes and during sleep.

Theta

Theta waves appear as consciousness slips towards drowsiness. Theta waves have been associated with access to unconscious material, creative inspiration and deep meditation.

Delta

These waves are high amplitude rhythms. They are slow, but the highest amplitude. They never go down to zero because that would mean that you were brain dead. Primarily they are associated with deep sleep and may be present in the waking state.

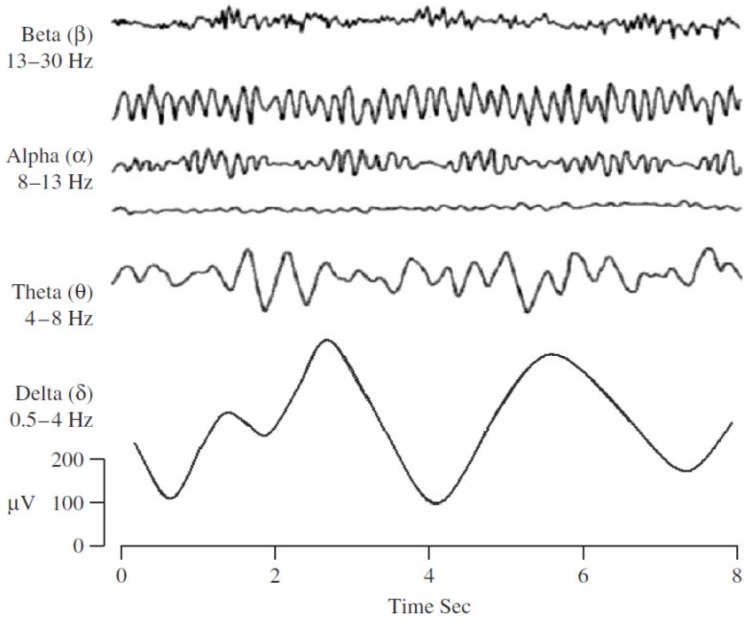


Figure 2:3 - Four typical brain rhythms [5]

2.3 ERP

Event Related Potentials are the parts of the EEG signal. They are electrical responses of the cortex to sensory, cognitive, or affective events. They usually occur as the brain response to external stimulations, and appear as auditory, visual or somatosensory brain potentials. The ERPs consist of a sequence of positive and negative voltage fluctuations called components. Because of their relatively small amplitudes (1-30 μV) in compare with the background EEG activity, a signal-averaging procedure is often needed to use [5].

ERPs can be classified according to nature of the stimulus [7]:

- Auditory
- Visual
- Somato-sensory

Potentials are also classified according to the latency at which occur after stimulus:

- Short latency (< 100 ms)
- Long latency

Long latency potentials are referred by the components which appear 100 ms after stimulus. Because of their affecting by level of attention they are used to evaluation in this thesis and components related to long latency potentials are mentioned in the next chapter.

2.3.1 ERP components

The ERP waves are either positive or negative. Positives are labeled as P and negatives are represented by the letter N. P or N are followed by digits representing either an ordinal position after stimulus or latency in milliseconds. The most common are P1 ~ P100, N2 ~ N200, P3 ~ P300 and N4 ~ N400. Common convention of plotting ERP waveforms has positive polarity of voltage downward and negative voltages upward [8]. Extraction and averaging of the ERP waveform can is shown in Figure 2:4.

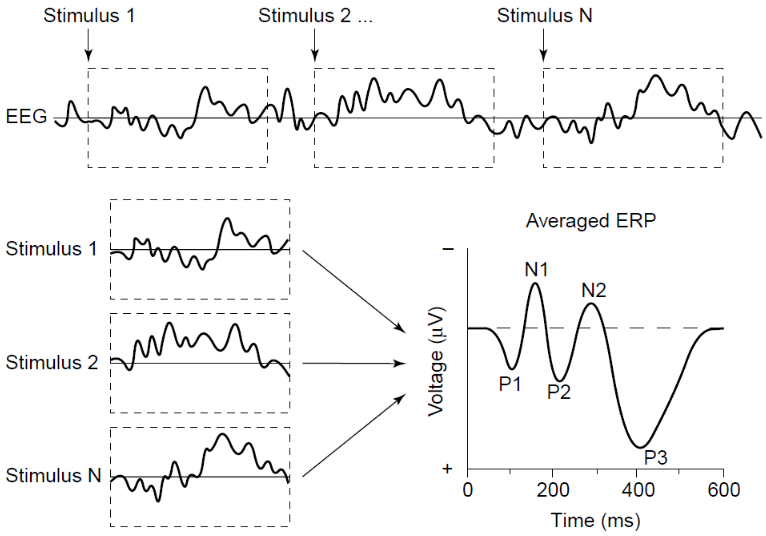


Figure 2:4 - Extraction of the ERP waveform from EEG [9]

P300 component

The P300 component is a positive ERP component occurring at about 300 milliseconds after stimulus. It depends on the type of stimulation and duration of stimulus. It can be between 300 – 450 msec. The signal is typically measured most strongly on the parietal lobe. This component is one of the most experimented wave in the ERP. Nowadays is known a lot of about the effects of various manipulations on P3 amplitude and latency, but still there are no clear consensus about reflection of neural or cognitive process in the P3 wave [8].

P3 is not a unitary phenomenon. Research has shown that it contains two distinguishable subcomponents. The term P3 is usually used to refer the P3b component. Different shape of waveform can be seen in Figure 2:5.

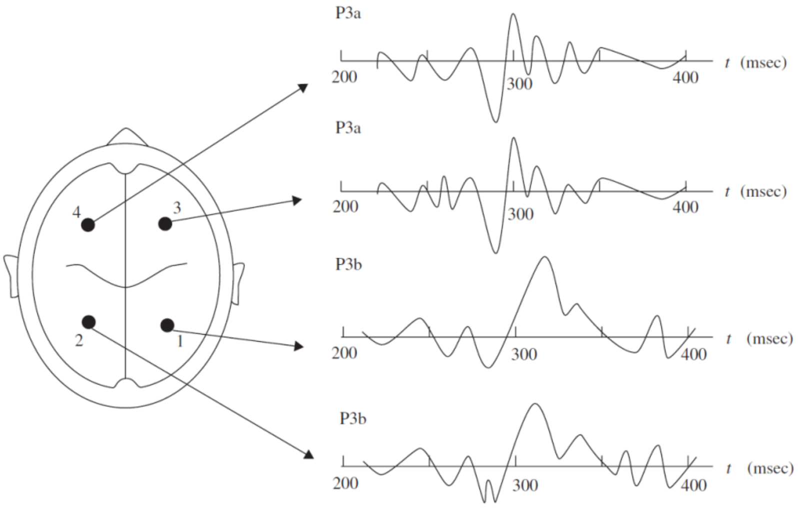


Figure 2:5 - P3 subcomponents P3a and P3b [5]

The morphology feature of P3 wave is its relation with probability of target stimulus. Amplitude of P3 increases with smaller probability of stimulus. The next hallmark of the wave is its dependency on a number of non-target stimuli preceding the target. P3 is larger when more non-targets are before the target. P3 amplitude is also larger when the subject pays more attention to a task.

On the other hand, P3 amplitude is smaller if the subject does not know whether a given stimulus is not a target. It means that more difficult tasks can increase P3 amplitude because the subject pays more attention to these tasks and simultaneously decrease P3 amplitude because the subject is not certain of the stimulus category [10].

Ideas and assumptions related to the latency of the P3 component are associated with stimulus categorization. If stimulus categorization is postponed, P3 latency is increased. While P3 latency depends on the time required to stimulus categorization it does not depend on consequent processes. Thus P3 latency can be used to determine if a performed experiment influences the processes of stimulus categorization or processes related to a response [10].

2.4 Respiration

The next physiological signal utilized in this thesis is respiratory rate. The typical rate of a healthy adult is 12-22 breaths per minute. The measurement of respiratory parameters has grown in importance in physiological research. Depending on the type of research, these signals can be useful, often in neuroscience. For example, respiration plays a role in the MR environment, where it may be a source of related artifacts – the signal can be linked to movement artifact due to action of breathing. Changes in respiratory behavior could be related to many psychological conditions and processes (e.g. mental load). In the following text the used method is described [11].

Respiratory Inductance Plethysmography (RIP[3]) is a non-invasive method of measurement respiration by analyzing changes in movement of the chest. RIP uses respiration elastic belts to detect the expansion and contraction of the lungs based on changes of inductance. The belts can be used in two forms [12].

- **Two belts** – Pair of belts is worn around the chest and abdomen.
- **One belt** – Only one belt is used – this method was used in this thesis. In compare to the first variant (pair of belts) using of one belt is less accurate; furthermore this limits quantification of many useful respiratory indices and limits utility to only respiration rates and other basic timing indices.

Due to the measuring only the respiratory frequency, one belt variant is used sufficient.

2.5 Skin conductance

Skin resistance is very informative indicator of consciousness an emotional states. For measuring skin conductance the galvanic skin response (GSR[4]) method was used. Conductivity is measured from the fingers via two electrodes. The GSR reflects changes in the sympathetic nervous system and sweat gland activity. The activity of the sweat glands in response to nervous stimulation is accompanied by an increase the conductance level. GSR measures a change in the electrical properties of the skin in response to different kinds of stimuli. In measurements changing the voltage measured from the surface of the skin are recorded [13].

GSR habituation is a well-known, the amplitudes of GSRs decrease during repeated stimulations. Waveforms are long with simple shape, usually biphasic [14]. As mentioned in [15] previous studies have shown that the amplitude of the late positive components of ERP and GSR measurements correlates.

---

[3] RIP – it is a method of evaluation ventilation by measuring the movement of the chest

[4] GSR – it is a method of measuring the electrical conductance of the skin

3 EXPERIMENT BACKGROUND

This chapter discusses several general best-practices to construct a successful experiment. These are followed by introduction to measurement of attention by using ERPs.

3.1 Strategies and rules of experiments

List of strategies [16]**:**

1) Focus on a specific component.
2) Use well-studied experimental manipulations.
3) Focus on large components.
4) Isolate components with difference waves.
5) Focus on components that are easily isolated.
6) Use component-independent experimental designs.
7) Hijack useful components from other domains.

List of rules [16] **:**

1) Peaks and components are not the same thing. There is nothing special about the point at which the voltage reaches a local maximum.

2) It is impossible to estimate the time course or peak latency of a latent ERP component by looking at a single ERP waveform—there may be no obvious relationship between the shape of a local part of the waveform and the underlying components.

3) It is dangerous to compare an experimental effect (i.e., the difference between two ERP waveforms) with the raw ERP waveforms.

4) Differences in peak amplitude do not necessarily correspond with differences in component size, and differences in peak latency do not necessarily correspond with changes in component timing.

5) Never assume that an averaged ERP waveform accurately represents the individual waveforms that were averaged together. In particular, the onset and offset times in the averaged waveform will represent the earliest onsets and latest offsets from the individual trials or individual subjects that contribute to the average.

6) Whenever possible, avoid physical stimulus confounds by using the same physical stimuli across different psychological conditions (the Hillyard Principle). This includes ‘‘context’’ confounds, such as differences in sequential order.

7) When physical stimulus confounds cannot be avoided, conduct control experiments to assess their plausibility. Never assume that a small physical stimulus difference cannot explain an ERP effect.

8) Be cautious when comparing averaged ERPs that are based on different numbers of trials.

9) Be cautious when the presence or timing of motor responses differs between conditions.

10) Whenever possible, experimental conditions should be varied within trial blocks rather than between trial blocks.

11) Never assume that the amplitude and latency of an ERP component are linearly or even monotonically related to the quality and timing of a cognitive process. This can be tested, but it should not be assumed.

The Hillyard Principle: Always compare ERPs elicited by the same physical stimuli, varying only the psychological conditions.

3.3 State of the Art

3.3.1 Study 1

Comparison of P300 from passive and active tasks for auditory and visual stimuli [17]

Introduction

One of the most common paradigm uses two different tones – target and non-target. The target "oddball" stimulus is presented with the low frequency of occurrence in compare with non-target presented higher frequently. The subject is asked to distinguish between two tones by responding to the target usually mentally counting or concentrating and no responding to the standard.

Another point of view was reported in several studies – waveforms can be obtained with "passive" oddball paradigm; Intentional discrimination between the two tones is not required.

However, passive and active tasks do not elicit identical P300 components. In summary only the auditory part of experiment is used.

Method

The study compared auditory and visual stimuli using the same stimulus sequence paradigm played in passive and active condition. Testing was realized on the 16 subjects, university students, without psychiatric or neurologic problems. EEG activity was measured at Fz, Cz and Pz electrodes in 10-20 international system.

Auditory stimulation presentation consisted of 10 tones with 2 seconds interval. After six non-target 1000Hz stimuli, 2000Hz target tone followed on the one of four random positions (7th, 8th, 9th, 10th) – remaining 3 position again were used for standard non-target stimuli.

Passive tasks were presented before active. It is due to avoid of inducing a response to the target stimulus during the passive task, which could occurred, when the active task would be presented before the passive one. The tested subject was instructed to listen to the tones without making responses.

Active task conditions were set according to oddball paradigm. The subject had to respond to the target stimuli.

Result

Only data from the target stimuli were considered. In the case when subjects responded to the target stimulus within the stimulus paradigm, the amplitude was larger than when subjects were passive. P300 latency was marginally shorter for the passive in compare to active condition. Waveforms are compared in Figure 3:1.

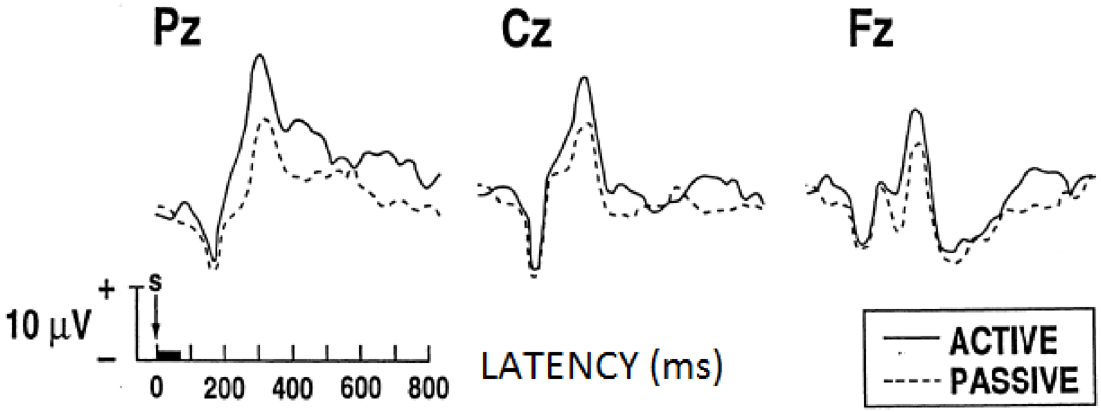


Figure 3:1 - Averaged ERPs from active and passive task conditions [17]

3.3.2 Study 2

Analysis of P3 in a continuous 40-min auditory oddball task [18]

Introduction

The experiment deals with analyzing latency and amplitude of the N1 and P3 peaks for non-target as well as the target stimuli. The investigation of P3 latencies and amplitudes for the target ERP is more important for this thesis. Analysis was realized in a continuous 40-min auditory oddball task.

Method

A group of tested subjects consisted of eight men and eight women aged between 22 and 25 years. The subjects had to detect 10 percent of target stimuli tones (2kHz). These targets were pseudo randomly interspersed among common non-target stimuli (1kHz). As a method of recognizing the target stimulus, pressing a button as quickly and accurately as possible was used. Tones were presented in an inter stimulus interval 1800 millisecond. Presentation of sounds was done throw a speaker. Each experiment consisted of 4 minutes training period followed by next three 12 minutes experimental sessions. The number of 35 EEG targeted responses was obtained from all sessions (except training). The averaging of ERP waves was set from 100 ms before the stimulus to 900 ms after the stimulus.

Result

The study provided observation that the N1 amplitude of the target as well as non-target stimulus was significantly larger at the electrode Fz than at Pz. In the P3 context it was observed that P3 amplitude of target was significantly larger at Pz that at Fz in every session. Furthermore, decreasing of P3 amplitudes in the 3rd session against to the 1st session was observed.

3.3.3 Study 3

Controlled inducement and measurement of drowsiness in a driving simulator [19]

Introduction

This study looked for patterns in biomedical variables that could be used to detect and characterize the drowsiness and its phases.

Method

During the 105 minutes testing 20 volunteers were measured. Eyes closure, pressures on the seat and longitudinal and lateral control of the vehicle were recorded in a simulator.

Result

The drowsiness was induced in 80% of the subjects. Increase of fatigue affected performance of the driving task such as biomedical signals. In the end, using of respiration and heart rate measurement is supposed. In the future, it can be used to drowsiness detection in a similar experiment.

3.4 Summary of observations

In chapter 3.3 difference of shape in passive and active task during auditory stimulation were presented. As a result, experimental design can be based on the passive task ensuring the same quality of analysis options. The next study confirmed that 40 minutes testing should be sufficient to get credible results. The last study confirmed existing correlation between physiological sensors.

3.5 Laboratory equipment

The experiment was realized in the laboratory at Department of Computer Science and Engineering at University of West Bohemia. There is all of necessary hardware and software equipment for experimenting in the laboratory (Figure 3:2).

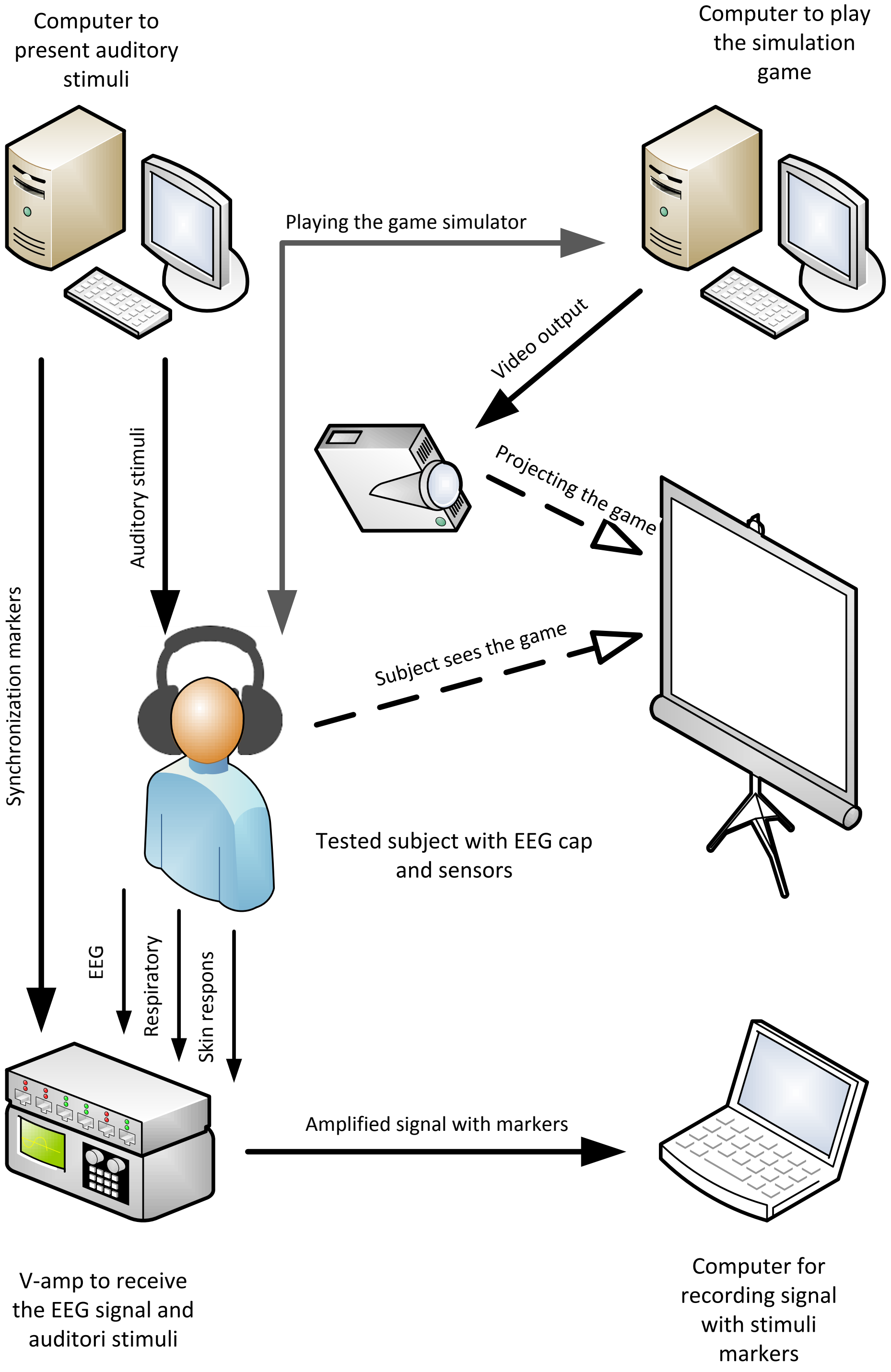


Figure 3:2 - Laboratory schema

In the schema of laboratory equipment (Figure 3.2) the main parts of hardware equipment are shown. The detailed list of used hardware includes:

- EEG Brain Products equipment: V-amp, EEG cap and physiological sensors – GSR sensor and respiratory belt.
- Driving simulator – a cockpit of Skoda Octavia car with embedded steering wheel and pedals Logitech G27.
- 2 PCs with 2 monitors – the first is used to present auditory stimuli, the second is connected to a projector and it is used to play the game.
- Laptop – used for recording and storing EEG signals by BrainVision Recorder software.
- Projector – used to project the game in front of the driver
- Headphones – to listen to auditory stimuli from the Presentation software.

3.5.1 Software

Simulation

As environment for simulation of driving computer game the World Racing 2 produced by Synetic [20] was chosen. Although the game itself does not provide an optimal scenario with monotonous road, there are other options to add an acceptable scenario into the game.

Presentation

Presentation [21] is a control software system for neuroscience. Software is designed for physiological experiments include ERP, reaction time and other issue. The Presentation is optimized to deal with two separate priorities, performance and accuracy. That is necessary to synchronize markers on pc output with presented stimuli (sounds, video). The recorded signal has to be synchronized in milliseconds.

Synchronization markers (binary representation of played stimuli) are transmitted throw parallel port from PC to V-amp. V-amp is connected by USB with PC with Brain Vision Recorder software. The program uses SDL[5] and PCL[6] to describe presentation of stimuli. An example of used code (Figure 3:3) shows filling of an array with target and non-target stimuli.

---

[5] SDL – descriptive language used to specify stimuli presentation

[6] PCL – interpreted programming language used to implement of scenarios

```
trial {
start_delay = $interstimulus_interval; #time between stimuli
stimulus_event {
   sound nontarget;
   port_code = 1; #numeric port_code of stimulus
   code = 1; #numeric code of stimulus
}event1;
}main;

begin_pcl;
array <int> stim[100]; #array for 100 stimuli
array <int> out_code[100]; #array of 100 out_code values
int st, en, pos;
int count_of_cycles = 5; #one cycle takes 84 seconds -> 5*84
= 420 = 7 minutes

sub #procedure stop presentation for entered time (ms)
  wait(int time)
   begin
     loop #cycle
     int end_of_waiting = clock.time() + time*1000;
     until
     clock.time() >= end_of_waiting
   begin
   # empty body
   end

end;
```

Figure 3:3 - Example of the Presentation software code

Recording

Brain Vision Recorder software [22] records EEG signals from an electrode cap. In addition, it can be used to record the signal from two external physiological sensors.

Analyzing

EEGLAB [23] is used for processing EEG and ERP signals and other electrophysiological data. It is an interactive Matlab toolbox and is considered as an alternative to BrainVision Analyzer software.

EEGLAB allows user to use standard averaging methods, as well as time and frequency analysis and independent component analysis.

ERPLAB Toolbox [24] extends EEGLAB's options to provide robust tools for ERP processing and analysis.

3.5.2 Hardware

EEG cap

An electrode cap is used to apply electrodes to the human scalp. The 10-20 international system of electrode placement is used. The cap (Figure 3:4) contains unipolar electrodes to create connection it is necessary to be connected to the second pole electrode called the reference electrode. This electrode is shared by all of cup electrodes as a second pole. [25]

Figure 3:4 - EEG cap [25]

BrainVision V-amp

This amplifier disposes a lot of features. Used connection is shwon in the Figure 3:5. V-amp has 16 monopolar channel inputs for the EEG cap and 2 auxiliary bipolar channels inputs for sensors. In the display, current configuration of electrode conductivity can be seen.

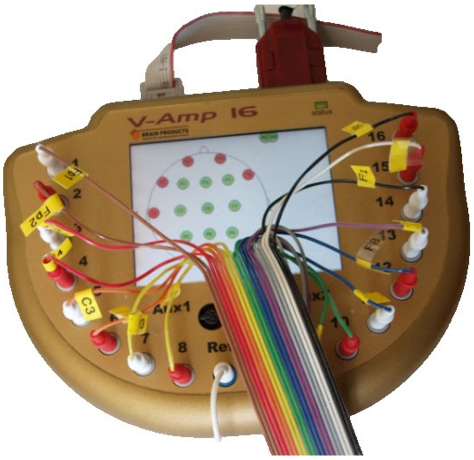


Figure 3:5 - V-amp with connected channels

GSR sensor

The used GSR-MR module can be used during medical and non-medical research in the field of physiological process. It is used to convert the conductance of the human skin to a voltage record by a bipolar amplifier input [26].

To establish contact between the human skin and two electrodes the usage of special gel is necessary. These sensors are used only non-invasively.

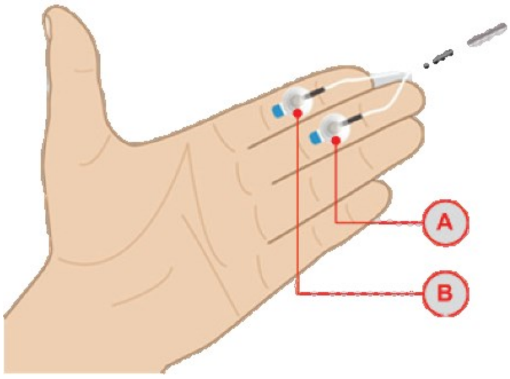


Figure 3:6 - Position of the electrodes [26]

The electrodes are connected to the non-dominant hand on the middle of the index finger (B) and middle finger (A) – this position is showed in Figure 3:6.

Respiratory belt

Respiration belt is a non-intrusive comfortable sensor; it is not perceived as disturbing by the tested subject.

The sensor (Figure 3:7) consists of a disposable belt with an internal sinuous wire. At the ends of the belt there are two connectors for making electrical contact of wire in the belt which serves as the inductance of an oscillation circuit. Stretching the belt changes the frequency of the oscillator slightly and therefor they are amplified and filtered. The sensor can be used for thorax and abdomen [27].

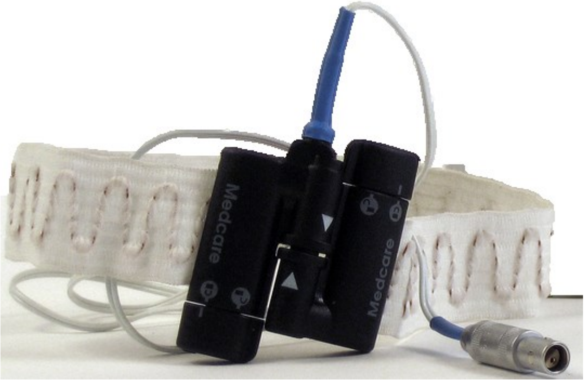


Figure 3:7 - Respiratory belt [46]

4 EXPERIMENT REALIZATION

4.1 Experimental design

The experiment deals with investigation of driver's attention during monotonous drive. The proposal of experiment is based on the observation of scientific articles; some of them are summarized in chapter 3.3. I have used experience and best practices obtained during realization of my bachelor thesis [28] and used in the article [10].

Assumption is based on prolongation of reaction time of the tested subjects caused by monotonous driving and on gradually decreasing attention. This should cause extension of response time with incoming fatigue. As compromise of necessary time to bring to drowsiness and tolerability for tested subjects 50 minutes experiment schedule was chosen.

As an evaluation method of auditory ERPs, comparison of peak latency of P300 components was chosen. EEG measurement was supplemented by sensors for recording other physiological functions; specifically a respiration belt and a GSR module were used. The respiration belt is used for measurement of frequency of breathing; the GSR module records skin conductance.

The experiment was realized on a group of 10 drivers, 22 - 25 years old (2 women, 8 men). For 50 minutes they were sitting in a car simulator and were connected to electrodes for EEG recording and to sensors. During driving tested subjects had headphones for better transmission of sounds and for minimization of audio noise from surroundings. Their mission was to drive the car simulator on a monotonous endless highway track without traffic. As the track highway built by Jan Rada in his bachelor thesis [29] was used. The track (Figure 4:1) did not contain any sharp turning; the tested drivers did not have to do any fast movements which could produce artifacts to the signal. The next positive was that driver could drive by using only one hand; this was ideal for measuring by using GSR sensor or heart rate sensor. The left hand could be placed on the armrest of car's door.

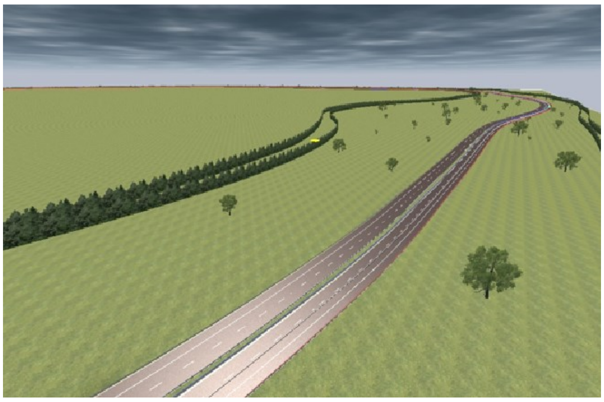


Figure 4:1 - Used track [29]

Auditory stimulation was realized using common paradigm. In this task two stimuli were presented in a random series with one of the two occurring relatively infrequently. This auditory paradigm used two different tones which were used as two types of stimuli – target and non-target. Target stimulus had lower frequency of occurrence than basic – non-target stimulus. Inter-stimulus interval was 800 milliseconds. The frequency of target stimulus occurrence was 7%. It was realized using array of 100 non-target stimuli. Over these stimuli targets tones were written; in every 14 non-targets, one item was randomly overwritten onto target. Between two targets at least 3 non-targets were placed. The schedule of experiment is showed in Figure 4:2.

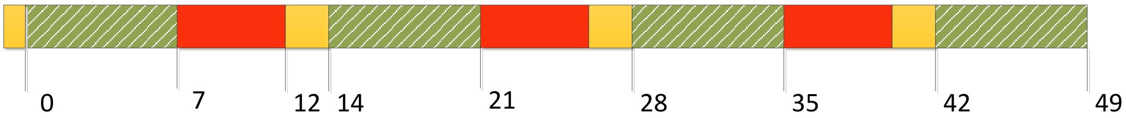


Figure 4:2 - Experimental schedule

In Figure 4:2 50 minutes timeline can be seen. In summary, there are four 7-minutes intervals (green color) of stimulation during driving and three periods without stimulation (red color). The first minute is not included in the line because recording is not activated. After the first minute, (it is point "0" in the timeline) recording is started. There are seven minutes periods of stimulation and relaxation. Green color shows stimulation which is followed by a red section. At these 5 minutes time-box driver does not drive, he/she only sits and relax. After that he/ she is asked to continue in driving (but without stimulation) for next 2 minutes (yellow time-box). Then 7 minutes stimulation period starts again.

2-minute driving without stimulation is designed to decrease the drivers' brain activity before a new stimuli presentation. Only the periods with stimulation are analyzed and are important to experiment result interpretation.

The goal of using the respiration belt is to record respiratory rate. It is the number of breaths per one minute. The sum of this number in each of 4 measured experiment periods (7 minutes) will be used for comparing number of breaths in across the periods. The relation with prolongation of ERP P3 latency is expected. As well as correlation of respiratory frequency with ERP the skin response correlation is expected.

4.2 Method of measurement

11 channels were used for recording. Together with these channels reference and ground electrodes were attached. Furthermore, two physiological sensors were connected – the respiration belt for measuring respiration frequency and

the galvanic skin response sensor to skin conductance measurement. The electrodes were located in accordance with 10-20 system.

Signals were obtained from the frontal, central, parietal and occipital areas. An amplifier for measuring EEG was set to continuous recording with a sampling rate of 1000Hz during the auditory stimulation. No filter was used during recording – all filtering was processed during signal processing.

4.3 Preparation for experiment

Because of the relevance of recorded signals and analyzed result, it is necessary to ensure that tested subjects comply basic requirement. One day before the testing, the subjects were asked to keep in their standard biorhythm. They were not allowed to drink a caffeine drinks and alcohol 12 hours before testing. The chosen people were did not use any medicaments affecting their basic physiological functions.

4.3.1 First step of experiment

Before testing and recording the subjects were acquainted with the car simulator and tried their driving technique for a short time. After that the document of declaration of agreement was presented and signed by the tested subject.

Then basic introduction to the EEG measurement was realized. The subject was informed about elementary rules, like reduction of facial expression, talking and mainly eye blinking. The rules were a standard breathing rhythm and smooth moving with steering wheel.

Before driving the simulator is set to the most simulation mode, all sounds of the game are turned off. The traffic and car damage is set to off.

4.3.2 Second step of experiment

After familiarization of the subject with the simulator and after setting the simulator options, the measurement can start. The first step is arranging the EEG electrode cap and establishing of connection between the scalp and electrodes. During arrangement the subject has to sit out of the simulator because there is not enough space

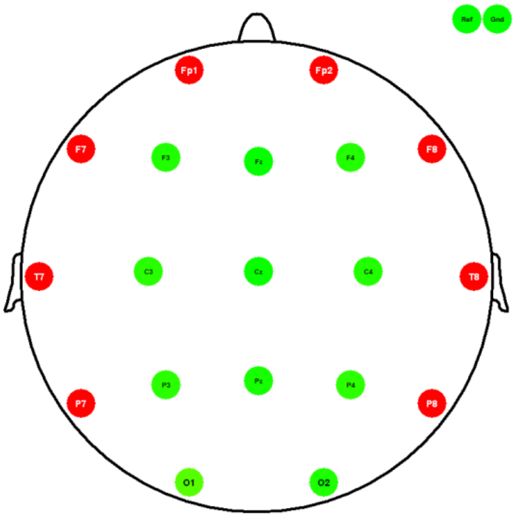


Figure 4:3 - Position of measured electrodes

inside above the subject's head. The EEG cap's output cables are connected into V-amp.

After putting on the electrode cap, the reference electrode and the ground electrode are used. The reference is attached on the forehead between eyebrows on the root of the nose. The ground clip electrode is attached to the ear lobe.

When all electrodes are attached, conductivity has to be established (Figure 4:3). This is realized using a special electrolyte gel.

4.3.3 During experiment

When conductivity is established, application of other two physiological sensors (GSR, Respiration belt) has to be realized. GSR is attached to the left forefinger and left middle finger using a special conductive gel. After that the respiration belt is dressed up.

Figure 4:4 - The subject during driving

After that the subject is asked to move into the simulator. The simulator is shown in Figure 4:4. Then revision of conductance is done. If everything is correct, the headphones are arranged on his/her head.

Finally the experiment can be started. In first minute the tested person only drives without recording and presentation of stimuli. After the first minute, presentation and recording start. Stimuli are presented for seven minutes, then there is 5 minutes break without driving. After that the experimenter gives a command to continue in driving but without stimuli for next 2 minutes. The next presentation cycle starts 7 minutes after the end of the first one.

4.3.4 End of experiment

After 49 minutes of recording and 50 minutes totally from the start of driving, the experiment ends. Then the examiner has to disconnect all cables from the cap and sensors. Then the tested driver can leave the simulator and he/she is asked to fill a short questionnaire (Appendix E). It contains data and metadata serving to other scientist to understand and evaluated recorded data.

5 EXPERIMENT EVALUATION

The output created by Brain Vision Recorder consist three types of files. The raw binary recorded signals are saved into *.eeg* file. This file can be interpreted using information from the header file *.vhdr*. Finally, the third file *.vmrk* is composed of experiment stimuli (markers) positions.

In these experiments the main binary file size is around 210 MB. The file contains signal records from totally 15 sources (11 eeg cap electrodes, ground, reference and 2 physiological sensors). Recording of these 15 signals took around 50 minutes.

Processing of measured experimental data has been doing in the EEGLAB software. This software was extended with the ERPLAB and Grand Average plugins. The ERPLAB was used to filter the imported raw data. Grand Average was used to average the result of ERP components across the time period and across the tested subjects.

5.1 Steps of data processing

This chapter describes the steps of data processing.

1) Import the recorded data from the folder to the EEGLAB:
   In the EEGLAB menu choose: File - Import data – Using EEGLAB functions and plugins - From Brain Vision Recorder .vhdr file:
   - In the dialog set channel numbers for analysis.

2) Filter the imported data using ERPLAB filters:
   In the EEGLAB menu choose: ERPLAB – Filter & frequency tools – Filters for EEG data:
   - There are two sliders to set High-Pass and Low-Pass filters in the filters dialog window.
   - High-Pass filter set to 0.1 Hz
   - Low-Pass filter set to 20 Hz

3) Now the data are filtered, the splitting to the time period with stimulation can be done:
   In the EEGLAB menu choose: Edit – Select data:
   - Set the time range of required time interval (in seconds)

4) From now the processing is realized on each of the selected time intervals.

5) Because the artifacts are in the raw records, their rejecting is necessary. In the EEGLAB menu choose: Tools – Reject continuous data by eye:
    - The continuous signals are displayed and using marking of the section, deleting this section is processed.

6) The selection of data using events has to be done, in the EEGLAB menu choose: Edit – Select data using events:
    - Select the type of target stimulus (S2).
    - Set time limits interval [-0.3 0.8] in seconds around the target stimulus.

7) The Channel locations setting has to be done, in the EEGLAB menu choose: Edit – Channel location:
    - Click OK button in the window to automatic settings.

8) The last execute step is extraction of epochs, in the EEGLAB menu choose: Tools – Extract epochs:
    - Epoch limits set to [-0.05 0.8] seconds

9) Finally, the visualization of the ERP can be done, in the EEGLAB menu choose: Plot – Channel ERP Image

5.2 Data processing

The recorded raw data are not usable as they are. The EEG must be filtered, because without the filtering almost nothing can be observed in the continuous raw signal. After the application of the specific filter the signal changes the shape to the form analyzable by human eye. Unfortunately, the artifacts recorded during measurement occur. Various artifacts may contaminate the EEG, they are caused by various factors. Rejection of artifacts is necessary; otherwise it the epoch extracted from the contaminated data could be distorted.

The most common artifact in the signal is the artifact of eye blinking. A typical one (Figure 5.1) has abnormally high and opposite peak against the ERP P3 component. Although the subjects are instructed to limit the blinking when the target stimuli are presented, it is the most common reason for non-accepting stimulus. The most problems with eye-blinking the people using glasses or eye contact lenses have.

Average contamination of target stimuli epochs by artifacts was about 50%. It means in absolute numbers that 13-16 stimuli were used from total number of 30 target stimuli.

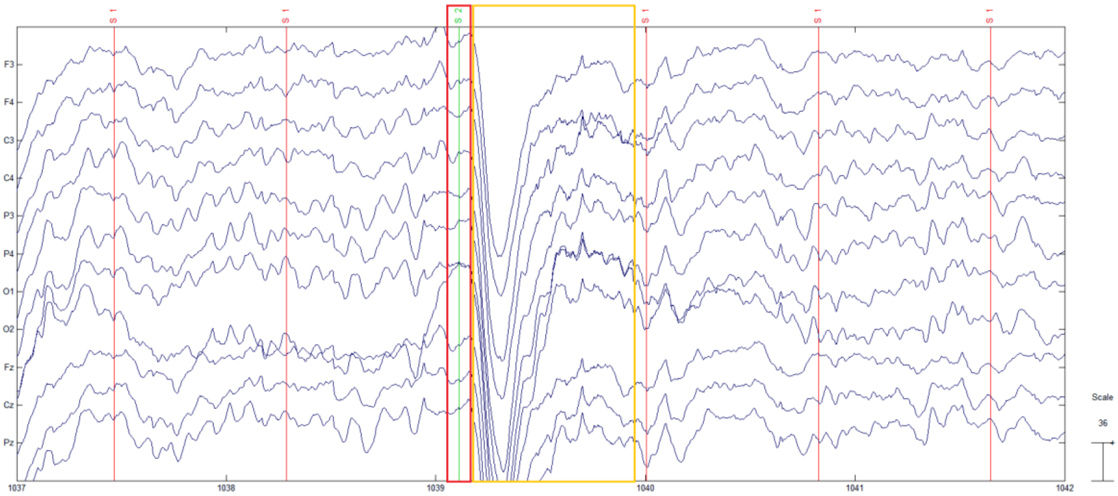


Figure 5:1 - Typical artifact

The target stimuli affected by artifacts have to be rejected from the continuous signal. As a common interval of affecting is considered a half second time interval before and one second interval after the stimulus. If the artifact interferes into this interval, the stimulus must not be used, because it could distort extracted ERP components. In Figure 5:1 the target stimulus affected by the eye-blink artifact is bordered by red color, the artifact (highlighted in orange rectangle) occurred immediately after the stimulus. This target has to be rejected.

Rejection of artifacts was realized with the signal splitted into the time periods according to the stimuli presentation. The experiment is splitted into 7-minutes periods (as it was mentioned in 4.1 Experiment Design). The periods were defined as 410 seconds intervals (420 seconds is 7 minutes minus 10 second tolerance).

Each time period (P1, P2, P3, P4) was analyzed separately. For each of them the EEGLAB's dataset (*.set*) is created. Within each period the extraction of events according to the target stimuli is done. The setting of the time interval around the target is [-0.3 ; 0.8] second. After that the epoch extraction is processed. There is a time interval for extracting set to [-0.05 ; 0.8] second. 50 milliseconds pre-stimulus is used to set the baseline.

When the epochs are extracted, the location of the P3 component has to be executed. In the EEGLAB the mode called plot is available. In this mode the epochs are drawn as a chart and the tool "data cursor" is used for taking the values on an x-axis. This axis represents the time in milliseconds. The axis starts with the value 0 (the stimulus location) and epoch time is measured from there up to 800 milliseconds where interval of extraction ends. In this interval, around

300 ms the P3 component is located and x-axis value is taken and inserted into the evaluation table in the next chapter.

5.3 Experimental results

The 49-minutes experiment was divided into four seven-minute intervals. Every interval has a column in Table 5:1 (Table 5:2 and Table 5:3). The last column contains the averages of each electrode from every seven-minute interval. The last row is average over all subjects on the specific electrode. This row is used to visualization the Figure 5:3. The averaging over the time periods required to use 4 different datasets. This type of averaging needs to add feature into the EEGLAB in the GranAverage plugin functionality.

Measured latencies were grouped according to the electrodes location. Three tables with different groups of electrodes were created. Electrodes O1 and O2 were analyzed too. Because of their ambiguous shape of epochs they were not considered in evaluation of experiment.

Nine of ten measured subjects were evaluated. One of the subjects (subject 6) had too many artifacts in the signal; it was not able to analyze it. Occasionally the P3 component was not located because of problematic shape of epoch. Instead of these unreadable latencies symbols “x” are in the tables.

Interval	[0-7]			[14-21]			[28-35]			[42-49]			Avg [P1,P2,P3,P4]		
Electrode	Fz	Cz	Pz	Fz	Cz	Pz	Fz	Cz	Pz	Fz	Cz	Pz	Fz	Cz	Pz
subject 1	325	317	324	322	322	318	329	333	336	329	333	336	329	319	319
subject 2	300	299	299	344	343	344	300	301	305	320	319	319	316	300	317
subject 3	350	348	347	326	323	323	353	351	341	346	343	346	352	345	346
subject 4	308	306	309	329	328	330	340	338	333	319	320	321	342	316	322
subject 5	305	283	284	307	306	306	319	302	300	292	282	282	310	288	282
subject 7	314	309	313	328	336	338	352	353	350	321	315	326	333	334	345
subject 8	352	352	365	338	350	351	353	333	331	353	357	352	344	353	354
subject 9	344	346	347	332	328	332	304	298	x	290	288	287	307	289	289
subject 10	276	276	277	293	293	294	297	288	291	285	284	283	278	273	276
Average	**321**	**319**	**323**	**330**	**329**	**334**	**333**	**312**	**321**	**328**	**328**	**329**	**327**	**321**	**328**

Table 5:1- Table of P300 latencies on Fz, Cz, Pz

In the next Table 5:2 values from F3, C3, P3 electrodes were evaluated:

Interval	[0-7]			[14-21]			[28-35]			[42-49]			Avg [P1,P2,P3,P4]		
Electrode	F3	C3	P3	F3	C3	P3	F3	C3	P3	F3	C3	P3	F3	C3	P3
subject 1	326	324	325	319	320	321	319	327	321	319	327	321	319	321	324
subject 2	300	299	300	344	345	346	311	305	310	331	328	328	321	317	319
subject 3	349	348	348	326	325	324	353	351	350	344	343	346	353	346	350
subject 4	308	307	310	331	324	328	338	337	338	324	324	323	339	321	322
subject 5	307	294	289	312	307	307	321	308	302	294	285	283	317	303	290
subject 7	313	313	313	335	336	347	352	353	354	314	310	315	330	334	335
subject 8	351	350	367	337	341	349	348	324	344	356	362	356	348	355	352
subject 9	346	348	347	332	333	333	305	303	304	290	289	287	292	289	288
subject 10	277	276	277	296	296	295	287	288	293	285	284	270	276	277	278
Average	**321**	**323**	**326**	**334**	**333**	**336**	**332**	**320**	**321**	**330**	**329**	**330**	**329**	**328**	**328**

Table 5:2 - Table of P300 latencies on F3, C3, P3

In the last Table 5:3 electrodes F4, C4, P4 were analyzed:

Interval	[0-7]			[14-21]			[28-35]			[42-49]			Avg [P1,P2,P3,P4]		
Electrode	F4	C4	P4	F4	C4	P4	F4	C4	P4	F4	C4	P4	F4	C4	P4
subject 1	324	317	323	330	322	320	344	333	332	344	333	332	327	318	316
subject 2	298	298	299	344	348	249	300	300	301	317	323	319	316	316	317
subject 3	350	335	346	325	323	322	353	351	341	347	344	345	353	345	347
subject 4	317	308	309	317	318	330	336	338	331	324	322	321	343	339	325
subject 5	300	285	284	307	305	306	319	301	299	293	284	284	305	290	288
subject 7	305	307	x	328	336	352	349	352	354	327	328	323	334	336	337
subject 8	354	354	360	339	352	349	347	344	333	360	359	358	351	355	352
subject 9	344	345	347	329	323	328	302	299	304	290	286	283	293	289	288
subject 10	277	276	277	293	295	295	298	293	293	284	283	273	277	272	280
Average	**318**	**320**	**325**	**330**	**329**	**330**	**339**	**332**	**331**	**329**	**329**	**327**	**323**	**326**	**329**

Table 5:3 - Table of P300 latencies of F4, C4, P4

The respiratory rate evaluation was realized by manual counting peaks of breaths. Each record was split into the intervals as well as during the EEG signal processing. It means that from the 49 minutes record, four 7-minutes intervals were selected and analyzed. A sample of signal is shown in Figure 5:2.

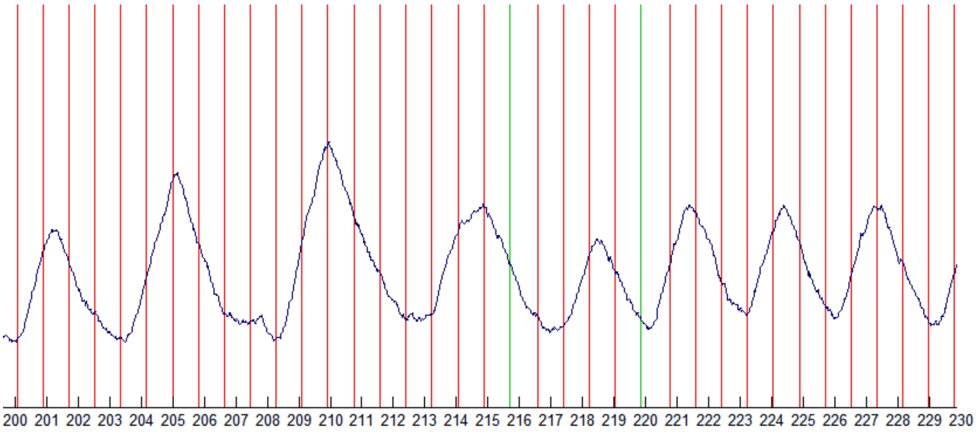


Figure 5:2 – 30-seconds illustration of breathing frequency

Each record was analyzed (and peaks were counted) in 60-seconds intervals (the last interval had only 50 seconds). Then number of breaths from these 60-seconds interval was summarized into the result in the table. The row "sum" represents the sum of subtotal count of breaths.

Table 5:4 shows the values of subject 2 as an example.

subject 2	P1	P2	P3	P4
60	x	x	x	x
120	17	12	16	15
180	14	12	16	12
240	13	16	14	11
300	13	18	13	10
360	15	21	13	12
410	12	17	12	11
sum	**84**	**96**	**84**	**71**

Table 5:4 - Subject 2 - numbers of breaths in

The same table as Table 5:4 was filled for all subjects (except subject 6 – he was discarded because of bad quality of the signal). From the sums from each table averages were calculated (Table 5:5).

Average of each period of all subjects			
P1	P2	P3	P4
109	**110**	**100**	**100**

Table 5:5 - Average of number of breaths across all subjects

5.4 Results interpretation

The chart of changes of averaged latency in each period was plotted from averaged values in Table 5:1. The averages of one electrode (Fz) were used in Figure 5:3.

After the first period the latency increases, but the maximum of latency is not in the last period as could be expected. The maximum latency come in the third period and then decreases again.

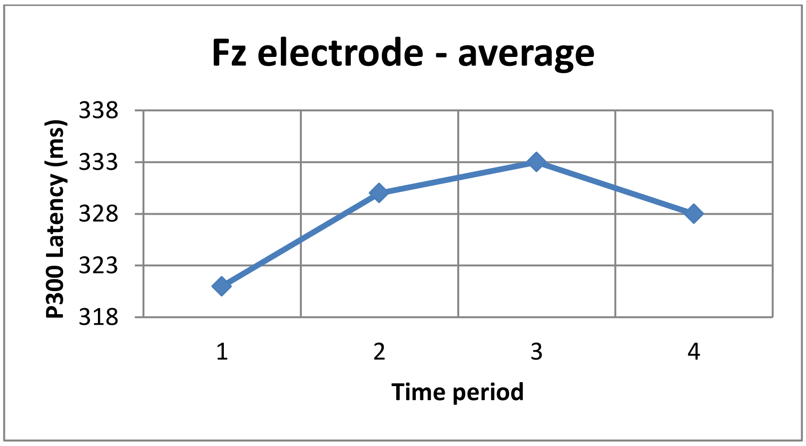


Figure 5:3 - Tendency of P300 latency in the periods

To compare the chart of averaged latencies with the chart of average respiratory rate Figure 5:4 was plotted. The chart does not show the frequency, but the absolute averaged number of breath-in per one period (410 seconds).

During the first and second period this number was similar, but after the second period decrease of frequency was observed. Then it stagnated again to end.

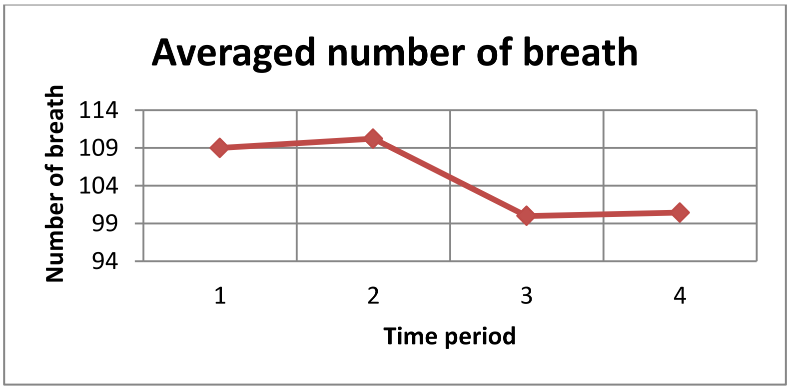


Figure 5:4 – Tendency of respiratory frequency in the periods

From the diagrams no clear correlation can be observed.

The next three figures show grand averages over all four analyzed intervals. Only the electrodes Fz, Cz, Pz are shown.

In the first figure (Figure 5:5) the averaged epoch from the Fz electrode is shown. The latency is equal to x-axis value, in this case the latency is 327 ms.

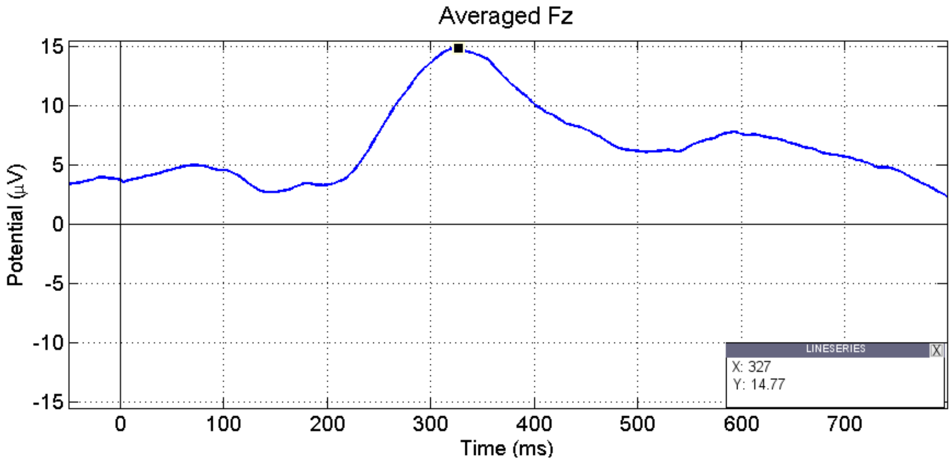


Figure 5:5 - Fz grand average

In the second figure (Figure 5:6) you can see the averaged Cz electrode signal. The latency of P300 is 321 ms.

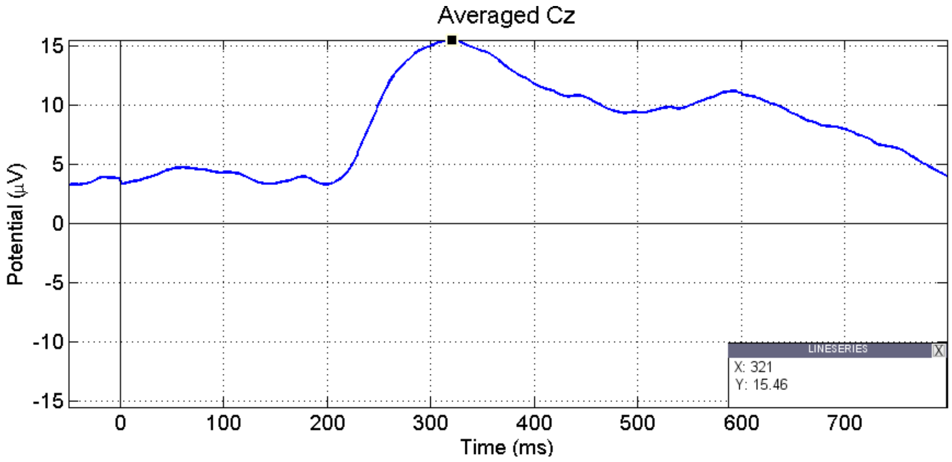


Figure 5:6 - Cz grand average

In the last figure (Figure 5:7) the averaged epoch from the Pz electrode is shown. The latency is equal to x-axis value. The latency is equal to 328 ms.

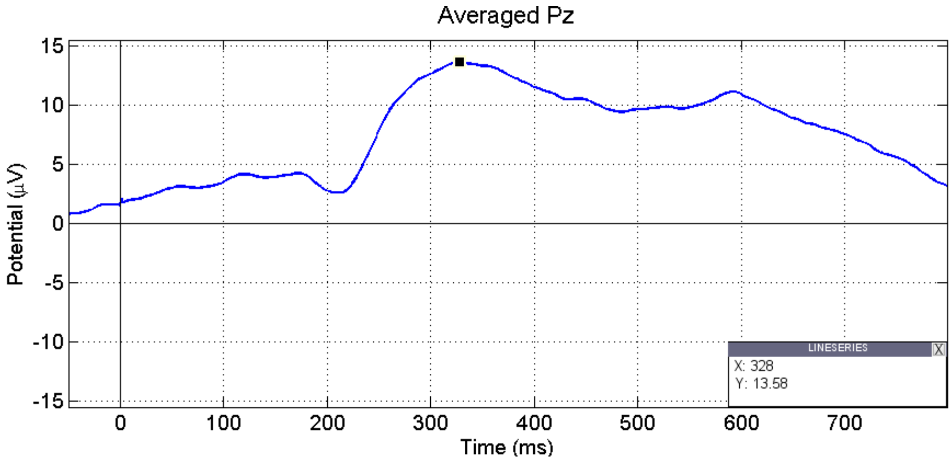


Figure 5:7 - Pz grand average

For clarity of the changes of the latency, the comparison of the ERP component in all of intervals (P1, P2, P3, P4) is attached and compared with the averaged component. The Fz electrode in each of four time period was compared. Each of period has different color: P1 – blue, P2 – red, P3 – green, P4 – yellow, average of period – black.

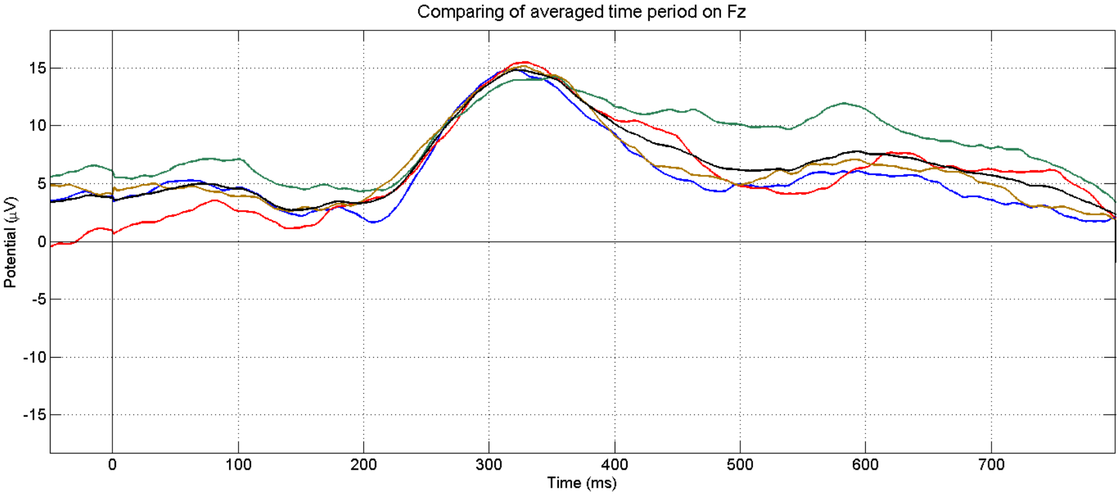


Figure 5:8 - Comparing of the averaged time intervals

6 EEG DATA STORING

6.1 State of the Art

INCF electrophysiology task force tries to specify a common file format for storing electrophysiological data and metadata.

Currently there are several initiatives to develop models and methods of storing neuroscience electrophysiology data. Their common layouts are using the same four types of data – time series data, experimental events, time series segments and neural events. Despite of using the same essential type, different methods of storing the data are used. It includes a different metadata model and its storing [30].

6.1.1 NDF

The Neurophysiology Data Translation Format [31] provides a means of sharing neurophysiology experimental data and derived data between services and tools developed within the CARMEN project.

The NDF specifies a format structure for sharing data. A dataset consists of a configuration file in XML format. This configuration file manages a variable number of host data files and it contains metadata too. The NDF specifies a set of commonly used data entities such as possibility to be extended to other data entities – third party format data files can be attached to and NDF dataset as an external format [31].

The NDF internal data types

Here is summary of NDF data types and file formats specification taken from [31] page 6:

- **Image data** – files are stored in the appropriate industry standard formats. Metadata of file are copied into the configuration file. Image data are therefore fully defined by the host data file.
- **TimeSeriesData** – is the basic data type for storage of multichannel signal data. Data can be of any numeric data types defined by the associate MAT file. This data type is fully defined by the configuration file together with the host data file.
- **SegmentData** – is the type used to store time-series data segments of one or more channels.
- **NeuralEventData** – data type used to store event data, where only time instances are of concern, such as spike time data.

- **ExperimentalEventData –** time stamp – annotation / event value pair sequences.
- **GenericMatrix –** provides a means of creating application specific data types.

The configuration file

The schema of the XML configuration files is shown here in Figure 6:1. It contains metadata associated with the host data files. The configuration file consists of one main element NdfDataId and sections GeneralInfo, DataSet and History.

- GeneralInfo – contains general information about the dataset.
- DataSet – core of the configuration file, metadata required for interpreting are in there.
- History –history of processing is stored.

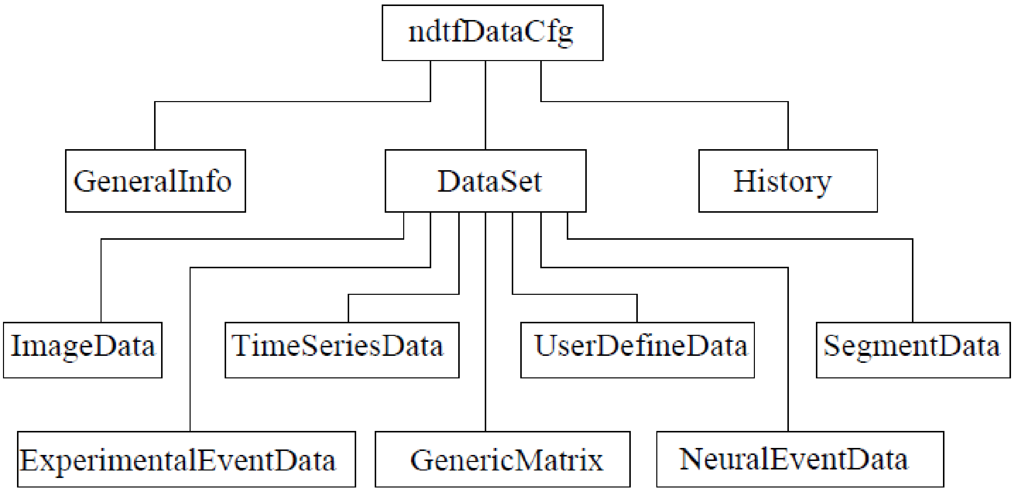


Figure 6:1 - Configuration file tree diagram [31]

6.1.2 Neuroshare

The purpose of the Neuroshare Project [32] is to create standardized methods for storing neurophysiological data from experiments in different data formats. In relation to standardized methods there is open-source software tool for managing these methods.

The Neuroshare Native Data File specification [33] describes a file format for storing common neurophysiology data from experiments. The structure of headers and data formats are based on the Neuroshare API Library specifications. A block of binary data values is referred by header information. This header information followed by binary data represents a data element.

The neuroshare native data format (.nsn) consists of several types data elements. The general structure for one data entity is a tag, followed by one or more informational header structures and then a block of binary data. The only information that is not tagged is the general file information structure placed at the beginning of the data file [33].

The purpose of Neuroshare API standard [34] is to define a common interface for accessing neurophysiology experiment data files. Using this interface (Figure 6:2) the data in a variety of proprietary file formats are accessible throw vendor-specific libraries.

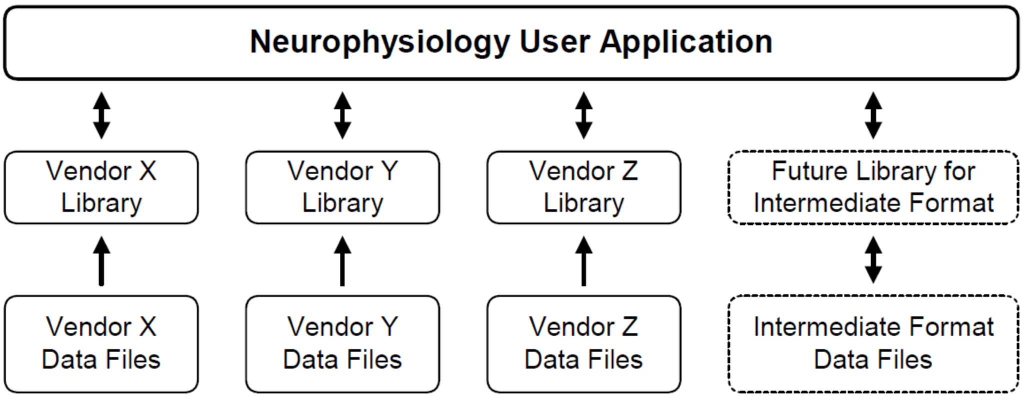


Figure 6:2 - Neuroshare schema [34]

Data types according to Neuroshare API are classified into four basic categories:

- **Event entities** – time-stamped data, represent data such as experimental events or digital input values.
- **Analog entities** – tepresentation of continuous digitized analog signals as electrode signals from EEG or EKG.
- **Segment entities**
- **Neural event entities**

6.1.3 NEO

The Neo is well established standard used by several projects (OpenElectrophy, G-Node, NeuroTools). It is a package for representing electrophysiology data in Python. It supports different file formats, such as Plexon, NeuroExplorer, Spike2. Furthermore, it supports non-proprietary formats including HDF5 [35].

Limited support of metadata and annotations can be considered as disadvantage.

A hierarchical data model is implemented and well adapted to EEG data including support of multi-electrodes. Neo objects represent neural data and collections of data. They are based on the NumPy arrays extended of metadata and unit conversion. The objects can be divided into categories: object, container and group.

- Data object – representation of data in arrays of numerical values, these are associated with metadata. There are main data objects: AnalogSignal, Spike, Event and EventArray, Epoch and EpochArray.
- Container object – used to make hierarchy. Contains: Segment and Block.
- Grouping object – are used to represent the relationships between data. The items are RecordingChannel, RecordingChannelGroup and Unit.

6.2 Data model requirements

A similar problem is solved in [30], [36], [37], [38] wherefrom some pieces of information are taken. From these articles request for using HDF5 arisen.

Storing of neuroscience data includes a lot of information related to electrophysiology datasets; there are not only the data themselves. The binary data need for an interpretation metadata. The metadata are in different types and are related to different binary data. The characteristics of the model should be clearly and exactly defined in detail. The implementation independence should be possible. Therefore the proposal (Chapter 7.2) of the data model is separated from its implementation (Chapter 7.3) in this thesis.

6.3 Hierarchical data format (HDF)

The Hierarchical Data Format [39] is a data model, file format and library for managing and storing data. The model (Figure 6:3) consists of several submodels, an abstract model and an abstract storage model. The library provides interface to implementation of the abstract models [40].

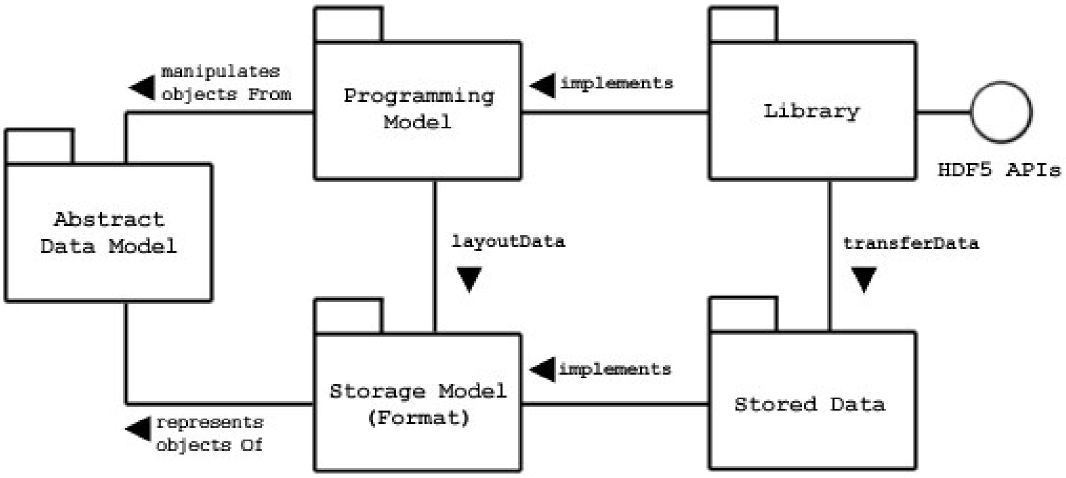


Figure 6:3 - HDF5 models and implementations [35]

The abstract data model

The abstract data model (ADM) is independent and conceptual model of data and theirs organization and types. It is not fixed to storage medium or programming environment. Many different kinds of data can be mapped to objects of the ADM, and therefore stored and retrieved using HDF5. Every data need to be mapped to the concepts of the Abstract Model.

File

Substance of HDF5 is organization into collections of objects (groups, datasets,...) and wrapping these collections into a file container. The objects are organized in the root structure; every file has the root group. The root group is only required object in the file. Each other object is a member or descendent of the root group. Root group is a special case of group, because may not be a member of any other group.

Group

A group can be seen as a common file system directory or folder. None or more objects can be members of the group. Every object must be a member of at least one group.

Dataset

The Dataset is a collection of data elements and metadata which describe raw data elements. The shape of a dataset is described by the dataspace object. The most common is one-dimensional or two-dimensional array of elements, but also more multi-dimensional arrays are allowed. The element of data can be a single unit like a number, character, an array of characters or numbers. Even heterogeneous records are available. The layout of bits representing a data element is described by the datatype.

Dataspace

It is one of the required components of the HDF5 dataset or the attribute definition. The size (size of dimension) and shape (number of dimensions) of the dataset or attribute raw data is defined in the dataspace. The definition of the dataspace is required when the dataset is created.

Datatype

This is the object describing the layout of a single data element. Data types are divided into several classes. The datatype class describes what the data mean and how these data are stored.

Attribute

Attributes are small data objects used to store metadata objects describing the main data object. As the main the group or the dataset object can be considered.

7 HDF5 MANAGER

The software tool Hdf5Manager is developed as an instrument for storing electroencephalography data measured at the University of West Bohemia. The data are recorded by Brain Vision Recorder software into its proprietary data format.

The application was developed at the initiative of the INCF group to try creating an open and portable data model in Hierarchical Data Format 5.

7.1 Used technology

The application uses HDF5 format as an output data format. HDF5 technology suite includes a software library that implements a high-level API with C, C++, Fortran 90 and Java interfaces. The developed application is written in C# under the .NET 4.0. That is the reason why the wrapper had to be used.

7.1.1 .NET Wrapper

HDF5DotNet wraps a subset of the HDF5 library API in a .NET assembly for consumption by .NET applications. The wrapper is written in C++/CLI and uses .NET P/Invoke mechanism to call a native code from a managed code which facilitates multi-language development in other .NET languages such as C#, VB.NET and IronPython [41].

The HDF5DotNet assembly is available only for .NET 4.0. To compile .NET application adding a reference to HDF5DotNet.dll is necessary.

7.1.2 WPF (MVVM)

The used pattern called Model/View/ViewModel (MVVM) is a variant of classic three-layer Model/View/Controller (MVC). Design of user interface is usually declared in XAML (or HTML) [42], [43].

- Model – data or business logic of the application is user interface independent and can be represented by data encoded in relation XML or tables.
- View – represents graphic elements. It is usually defined declaratively.
- ViewModel – contains commands in order to allow the View to interact with the Model.

Using WPF minimalizes the scope of C# code for graphic user interface (View layer). The user interface is specified in XAML code and managing of data transfer between GUI and business logic (ViewModel layer) is realized by data binding. Thus using WPF allows a strong separation between the business data and the user interface control. Business logic is independent from the controls used to render the data.

7.2 Data model

The realized data model is inspired by the proposes from [30], [36], [37], and contain some ideas from there. The scope of the proposed data model is to provide option to store and conversion recorded EEG signals from neurolaboratory at UWB. Against the original propose from INCF here are several limitations in this model.

In the epHDF [36] proposed for storing data in HDF5 the definition of the data types contains four main types: time series, time series segment, neural event and experimental event. As it can be seen in chapter (6.1) equivalents of these types are common for existing models and format compared in the chapter (6).

For the records evaluation two data types are sufficient:

- Time series –raw continuous EEG signal.
- Experimental events – position and type of stimuli presented during experiment.

In the data model these two types are expanded of further data types containing information about raw data. The specification of channels and events is linked to the raw data.

The proposed model is implemented in the HDF5 container format. HDF5 has three elements for definition of the structure: *Group, Dataset* and *Attribute*. Detailed description can be found in chapter 7.2.2 Metadata. There are used the following graphic elements (Figure 7:1) representing HDF elements in the diagram schema.

- Group – describes entities of the model and divides the model structure into logical parts. The groups are represented as the ovals.
- Attribute – contains metadata information. A collection of attributes is represented as a rectangle with a diagonal corner
- Dataset – contains main data such as a raw binary EEG signal. Datasets are visualized as a rhomboid.

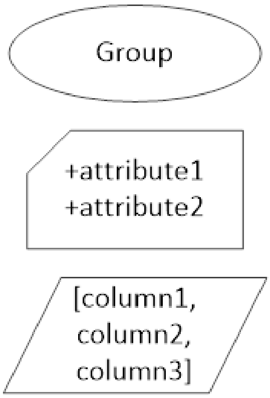


Figure 7:1 - Elements of schema

7.2.1 Schema

In Appendix C you can see a complete schema. It is described per partes in this chapter.

7.2.2 Metadata

Metadata section of data model contains information related to global conditions, properties of one experiment and information about a tested subject, scenario condition, settings, used hardware and software.

The group "/" is a root group of the HDF hierarchical structure. The groups serve only as wrapper entities for attributes representing metadata about these entities. Metadata part of model is shown in Figure 7:2.

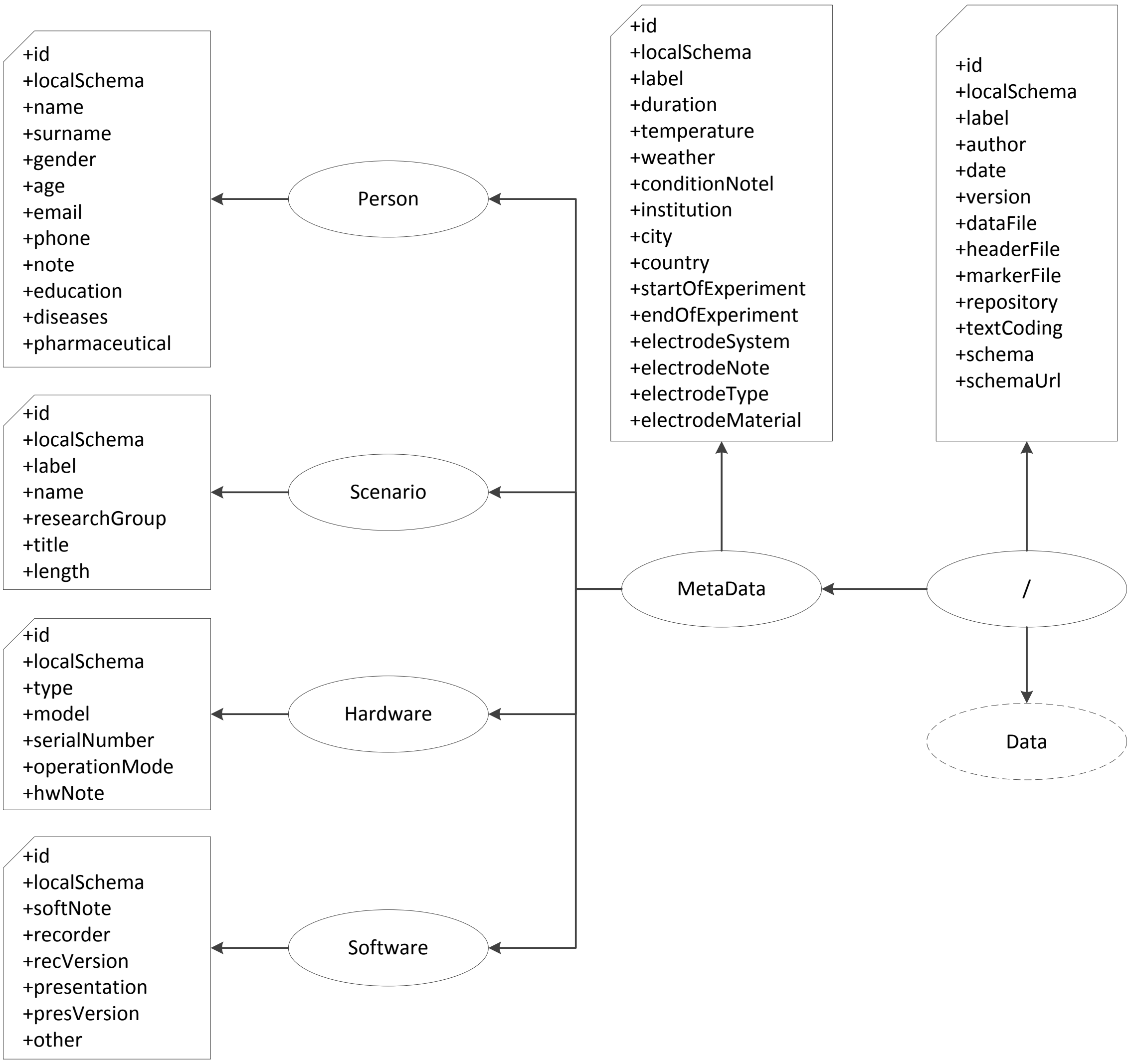


Figure 7:2 - Metadata part of data model

The hierarchy of group is root-oriented. The *Root* group contains common information, and references to *MetaData*. This group relates to other four groups: *Person, Scenario, Hardware, Software.*

All metadata entities have two common attributes: *id* and *localSchema.*

- **id:** represents a unique identification number of an object in the HDF hierarchy.
- **localSchema:** this string item contains a serialized structure of a whole entity. Serialization is done by JSON[7] and complies its format and structure.

In the most entities attribute called *label* is.

- **label:** text description of the saved data.

Root

The Root is the main and entrance entity of the data model (and the HDF file). This entity relates to the Metadata and Data nodes.

- **author:** name of person, who responses for the measurement and experiment.
- **date:** date of experiment.
- **version:** version of experiment. Some experiments has the same goal but the variables of the experiment could be changed.
- **dataFile:** path to an original .eeg file.
- **headerFile:** path to an original header .vhdr. file.
- **markerFile:** path to an original marker .vmrk file.
- **repository:** URL path to a repository, where original data can be found.
- **textCoding:** text coding used in the file strings.
- **schema:** contains a string with path to a document with description of this data model and its attributes.
- **schemaURL:** contains URL to a document with description of this data model and its attributes.

MetaData

The MetaData node contains general information about condition and settings during the experiment.

- **duration:** length of time in minutes. It is the time of the first task of experiment to the last task experiment. It is not the time interval from start stimulation to the end of stimulation.
- **temperatue:** temperature in the room during experiment.

---

[7] JSON – JavaScript Object Notation is a lightweight data-interchange format.

- **weather:** outside weather. A sunny day provides different than stormy day.
- **conditionNote:** note about anything important dealing with environmental properties.
- **institution:** name of institution where an experiment was done.
- **city:** name of the city where the institution is located.
- **country:** name of the country where the city is located.
- **startOfExperiment:** time and date when an experiment begun. It is not the start of recording the signal. The preparation phase can be included at this time.
- **endOfExperiment:** time and date when an experimenting ended.
- **electrodeSystem:** standard is 10-20 system.
- **electrodeNote:** application of actiCap can be noted here.
- **electrodeType:** type of used electrodes, e.g. EEG Cap.
- **electrodeMateriel:** material of used electrodes.

The MetaData node wraps other meta-data entities:

Person

The person entity contains the following attributes:

- **name, surname, gender, age, email, phone, note, education, diseases, pharmaceutical**

Scenario

Every experiment is realized according to some scenario. Its basic properties are:

- **name:** name of scenario.
- **researchGroup:** name of a group that realized the scenario.
- **title:** title of scenario description.
- **length:** time of testing.

Hardware

- **type:** type of a used amplifier.
- **model:** model name of a amplifier.
- **serialNumber:** unique number of hardware.
- **operationMode:** specific mode of interpreting and transforming data.
- **hwNote:** any other used hardware can be entered there, e.g. sensors.

Software

- **recorder:** name of software used to record EEG signal.
- **recVersion:** version of recorder software.
- **presentation:** name of software used for presentation of stimuli.
- **presVersion:** version of presentation software.
- **other:** any software used.

7.2.3 Data

The Data part of the data model (a schema is attached in Appendix C) relates to the recorded raw data and their context, opposed to the section about the metadata (7.2.2 Metadata), where user inputs are only considered. Most attributes in the data section are filled by parsed items from the source files. There are datasets containing data values.

Group and Datasets

- **TimeSeries:** primary data group contains dataset "Data". This dataset contains binary representation of raw data – byte array. The attributes relating to the dataset describe binary data.

  The group TimeSeries relates to two other groups Unit and Sample. These groups store information about physical units, like voltage.

- **Events:** group Events contains two datasets: **Position** and **Description.** The Position dataset contains numerical representation of the event (marker) position in the data file. The description dataset contains information about a specific marker.

- **Channels:** group contains two datasets: **Resolution** and **Description.** The Resolution dataset is the numerical representation of resolution of each channel. The Description dataset contains the string representation of channel information as a name or a used unit.

- **SoftwareFilters:** this group contains a dataset with items saved in string array. Each string contains parameters of software filtering during recording the signal (if filtering was active).

- **AmplifierSetup:** group contains a dataset with items saved in string array. Each string contains parameters of an amplifier. These attributes mirror settings from the Brain Vision Recorder workspace.

7.3 Implementation

7.3.1 Existing data format

The Brain Vision Recorder currently supports only the “BrainVision Data Exchange Format”. The complete description of the file structure is in [44]. Generally an EEG record consists of three different types of files:

- .eeg – recorded eeg signals in binary representation.
- .vhdr – header file in ASCII format; it consists of metadata for interpretation of .eeg file. Its format is based on the Windows INI format.
- .vmrk – markers file associated to .eeg data.

Vhdr file

The header file (example is shown in Figure7:3) is separated into several main sections, which contain keywords associated with values. The specification of the header file is described below:

```
Brain Vision Data Exchange Header File Version 1.0
; Data created by the Vision Recorder

[Common Infos]
Codepage=UTF-8
DataFile=RerichaJ000005.eeg
MarkerFile=RerichaJ000005.vmrk
DataFormat=BINARY
; Data orientation: MULTIPLEXED=ch1,pt1, ch2,pt1 ...
DataOrientation=MULTIPLEXED
NumberOfChannels=18
; Sampling interval in microseconds
SamplingInterval=1000

[Binary Infos]
BinaryFormat=IEEE_FLOAT_32

[Channel Infos]
; Each entry: Ch<Channel number>=<Name>,<Reference channel name>,
; <Resolution in "Unit">,<Unit>, Future extensions..
; Fields are delimited by commas, some fields might be omitted (empty).
; Commas in channel names are coded as "\1".
Ch1=Fp1,,0.0488281,µV
Ch2=FP2,,0.0488281,µV
```

Figure 7:3 – Example of .vhdr file content

Vmrk file

The marker file has a similar structure (Figure 7:4) to the header file; the keyword sections are followed by their values:

```
Brain Vision Data Exchange Marker File, Version 1.0

[Common Infos]
Codepage=UTF-8
DataFile=RerichaJ000005.eeg

[Marker Infos]
;  Each  entry:  Mk<Marker  number>=<Type>,<Description>,<Position  in  data
points>,
; <Size in data points>, <Channel number (0 = marker is related to all
channels)>
; Fields are delimited by commas, some fields might be omitted (empty).
; Commas in type or description text are coded as "\1".
Mk1=New Segment,,1,1,0,20130418164613701122
Mk2=Stimulus,S  1,21606,0,0
Mk3=Stimulus,S  1,22429,0,0
```

Figure 7:4 – Example of .vmrk file content

List of Keywords

The keywords from the .vhdr file and from the .vmrk file are enumerated in the Figure 7:5.

.vhdr file			.vmrk file
Common Infos	**Binary Infos**	**Channel Infos**	**Common Infos**
DataFile	BinaryFormat	Channel name	DataFile
MarkerFile	ChannelOffset	Reference channel name	**ASCII Infos**
DataFormat	DataOffset	Resolution	Type
DataOrientation	SegmentHeaderSize	Unit	Description
DataType	TrailerSize		Position
NumberOfChannels	UseBigEndianOrder		Channel number
SamplingInterval			Date
Averaged			
AveragedSegments			
SegmentDataPoints			
SegmentationType			
DataPoints			
CodePage			

Figure 7:5 – List of keywords in .vhdr and .vmrk files

7.3.2 Parsing and classification

The source data are in three files. Two text files are *.vhdr* and *.vmrk* files, the binary file is *.eeg* file. *TxtReader* class reads all lines from the text files into the string array; in the next step these are preprocessed in the classes *VhdrProcessing* and *VmrkProcessing*. Binary file is read in *EegReader class* into the byte array.

Preprocessing of the header and marker files mainly indexes every of section from the source files. The indexes are used as positions or border of sections; it helps with positioning in the parser.

The bytes array preprocessing solves the order of bytes in the output array and positions of vectors of each recorded channel. There are two possibilities of bytes channel vector positioning. First, the bytes of the first channel in the first position are followed by the bytes of the second channel in the first position, etc. Finally, after the first position the last channel, the second position of the first channel continues. The second option of bytes order is based on vectors. There is a vector of all positions of the first channel followed by the vector of all positions of the second vector, etc. Class diagram of data loading is in Figure 7:6.

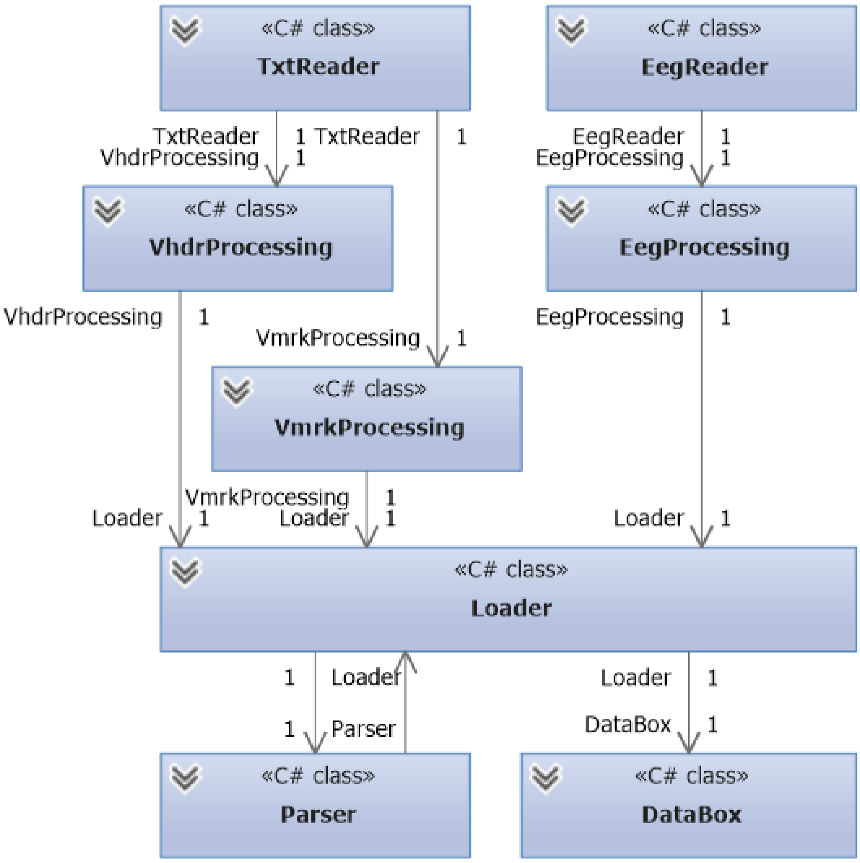


Figure 7:6 - The loader part of application

The *Loader* class uses the preprocessing classes and gets the string and byte array from these classes. Then the arrays are inserted into the *Parser* constructor, where the parsing of strings to the attributes is done.

Thanks to a relatively simple and defined structure of the text files the parser uses standard string processing functions and *switch*() or *if* () logical decisions.

The output of the parses is saved into the *DataBox* instance. The *DataBox* class (Figure 7.7) wraps all necessary data classes. Properties of wrapped classes serve for store the parsered attributes. The parser fills all read attributes to the properties of its DataBox instance and then returns this instance back to the Loader. After that the Loader returns these data into the MainViewModel.

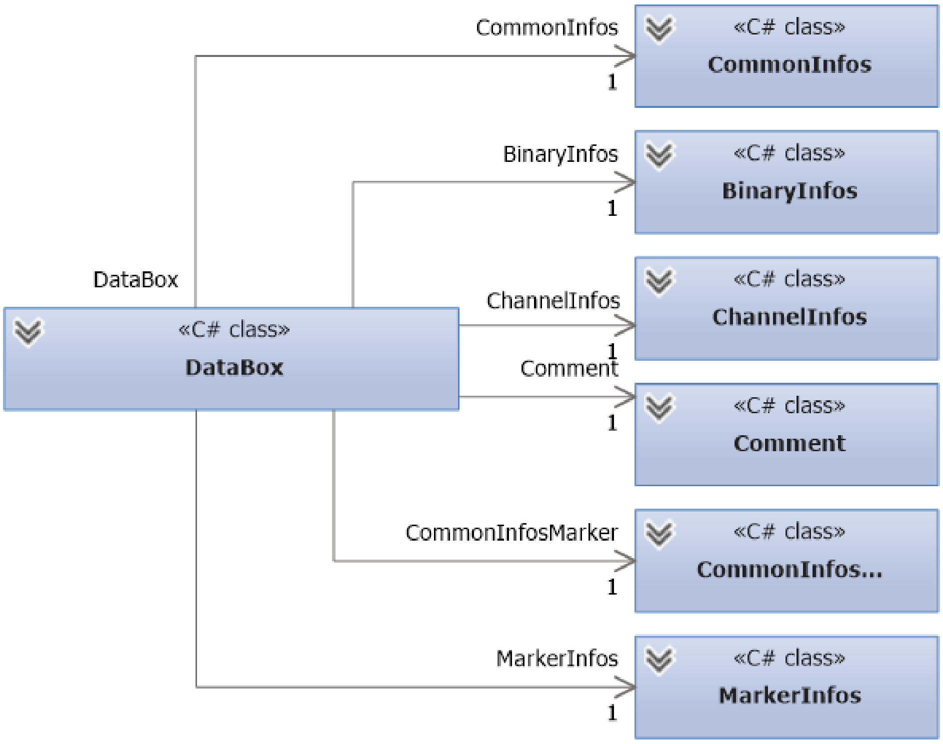


Figure 7:7 - DataBox class

7.3.3 Filling data model structure

When the parser filled the data class *DataBox* is returned to MainViewModel, where this class is put into the wraps class *DataNodeBox* as a parameter of a conversion method.

The *DataNodeBox* wraps all classes representing the entities from the data model and entities attributes are represented by the properties of the wrapped classes. The method *convertToNodes(DataBox db)* converts every filled property to a new structure. The properties of *DataNodeBox* without equivalent in the parser-filled *DataBox* are filled using data binding from GUI.

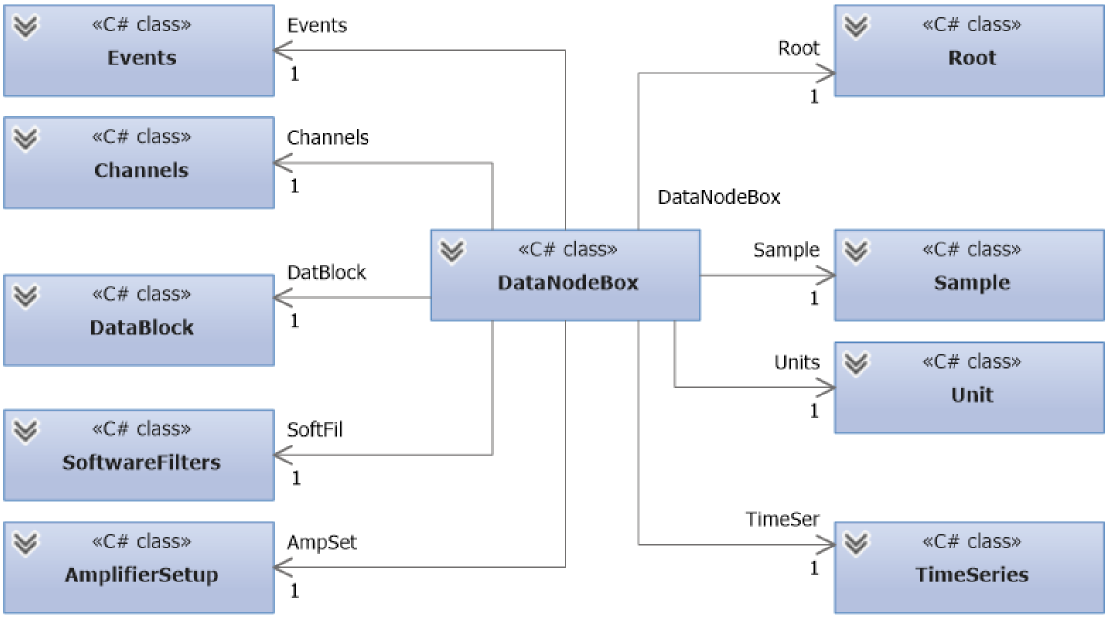


Figure 7:8 - DataNodeBox class diagram

Up to now the text above described handling the data obtained from the original recorded files (.vhdr, .vmrk., .eeg). The parser was used for loading them - data are loaded in classes corresponds to the original files layout.

The conversion of parsed data into the data-model structure is necessary. The data are converted into the new class structure (Figure 7:8). This structure contains data in datasets and metadata in attributes and correspondent the data-model layout.

The description how to obtain the user data inputs continues in the text below. As shown in the schema (Figure 7.9) all GUI inputs are realized by data binding and use the ViewModel layers.

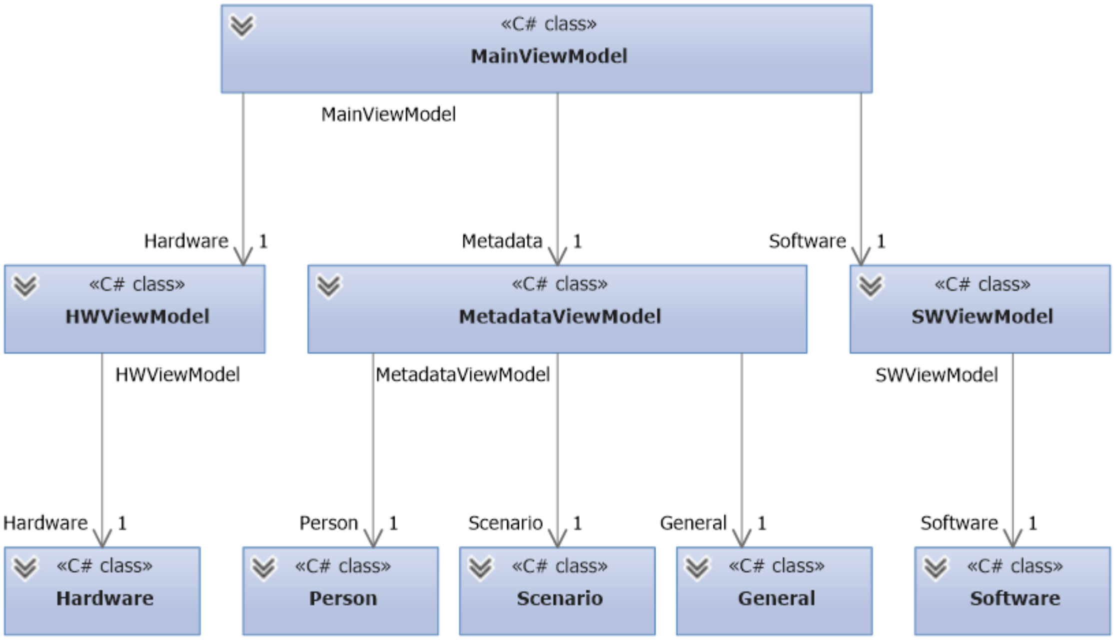
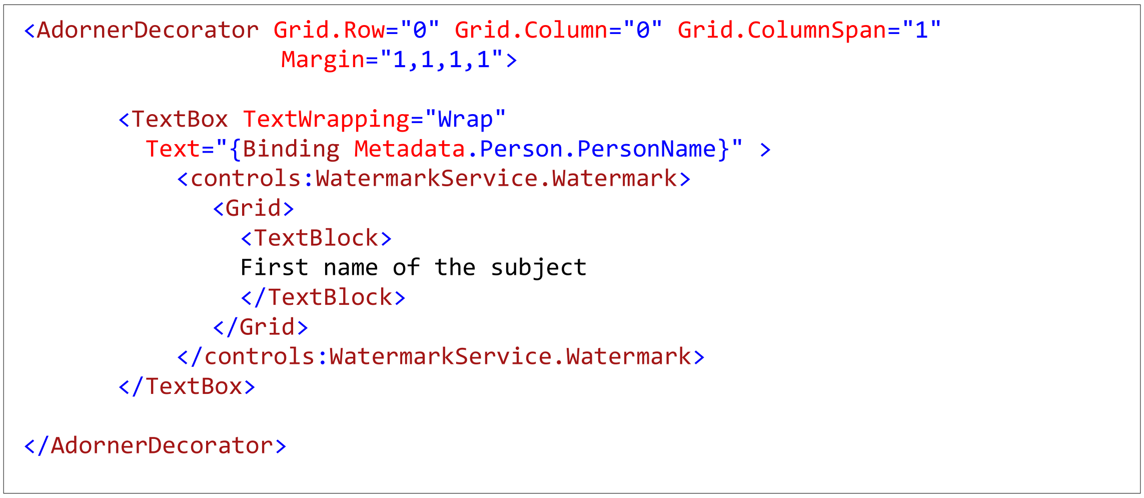


Figure 7:9 - ViewModels schema

The *MainViewModel* manages user interface commands. This main view model contains instances of next three view models: *MetadataViewModel*, *SWViewModel* and *HWViewModel*

All user inputs from the user interface are data bound directly to the Model that is exposed by the ViewModel. Example of textbox with data binding is shown in Figure 7:10.



```
<AdornerDecorator Grid.Row="0" Grid.Column="0" Grid.ColumnSpan="1"
                   Margin="1,1,1,1">

       <TextBox TextWrapping="Wrap"
         Text="{Binding Metadata.Person.PersonName}" >
           <controls:WatermarkService.Watermark>
              <Grid>
                <TextBlock>
                First name of the subject
                </TextBlock>
              </Grid>
           </controls:WatermarkService.Watermark>
       </TextBox>

</AdornerDecorator>
```

Figure 7:10 - Example of data binding

7.3.4 Exporting to HDF5

When the *DataNodeBox* container class is filled by the conversion method and the *MetaDataBox* viewmodel class is filled by the user interface, the creation of HDF5 file starts.

The *Creator* class uses the .NET Wrapper functions to access a subset of the HDF5 library API. The data used to filling the created HDF5 file are taken from the *DataNodeBox* and *MetaDataBox*.

To save the string attributes the *StringHelper* class was made, there are methods returning the string attributes. The returned attribute can be inserted into the File, Group or into the Dataset. The code for creating the string and its adding into the group is illustrated in Figure 7:11.

```
var charArray = StringToBytes(textValue);
var dims = new long[] { 1 };

H5DataSpaceId dspace = H5S.create_simple(1, dims); //definition of hdf dataspace
H5DataTypeId memtype = H5T.copy(H5T.H5Type.C_S1); //definition of datatyp
H5T.setSize(memtype, textValue.Length); //definition of size

H5AttributeId attr = H5A.create(groupId, attributeName, memtype, dspace);

var hdf5array = new H5Array<byte>(charArray);
H5A.write(attr, memtype, hdf5array);
```

Figure 7:11 - Illustration of method for creating a string attribute

The calling of adding attribute looks like (Figure 7:12):

```
StringHelper.SaveStringAttribut(gRoot, "Id", gRoot.Id.ToString());
StringHelper.SaveStringAttribut(gRoot, "Label", dnb.Root.Label);
```

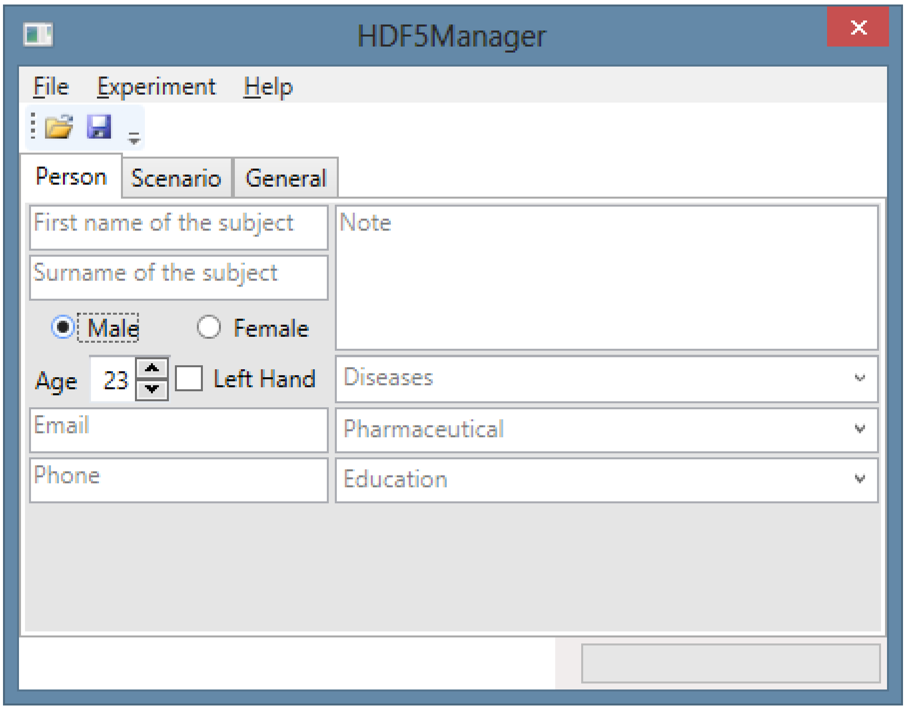
Figure 7:12 - Adding of the attribute to the HDF group

7.4 User interface

Graphic user interface (GUI) of the application is written in the WPF and uses data binding to transfer inputs to data classes. Data binding create the independent GUI and business logic is preserved.

GUI of the application consists of the main window (Figure 7:13) and several dialog windows are reachable from menu. The main window contains the menu, status bar and main user control – tabs control.

The tabs Person, Scenario and General are the main elements designed to insert metadata by the user. Every text input has watermark description inside to help user recognize required input.



● Male ○ Female

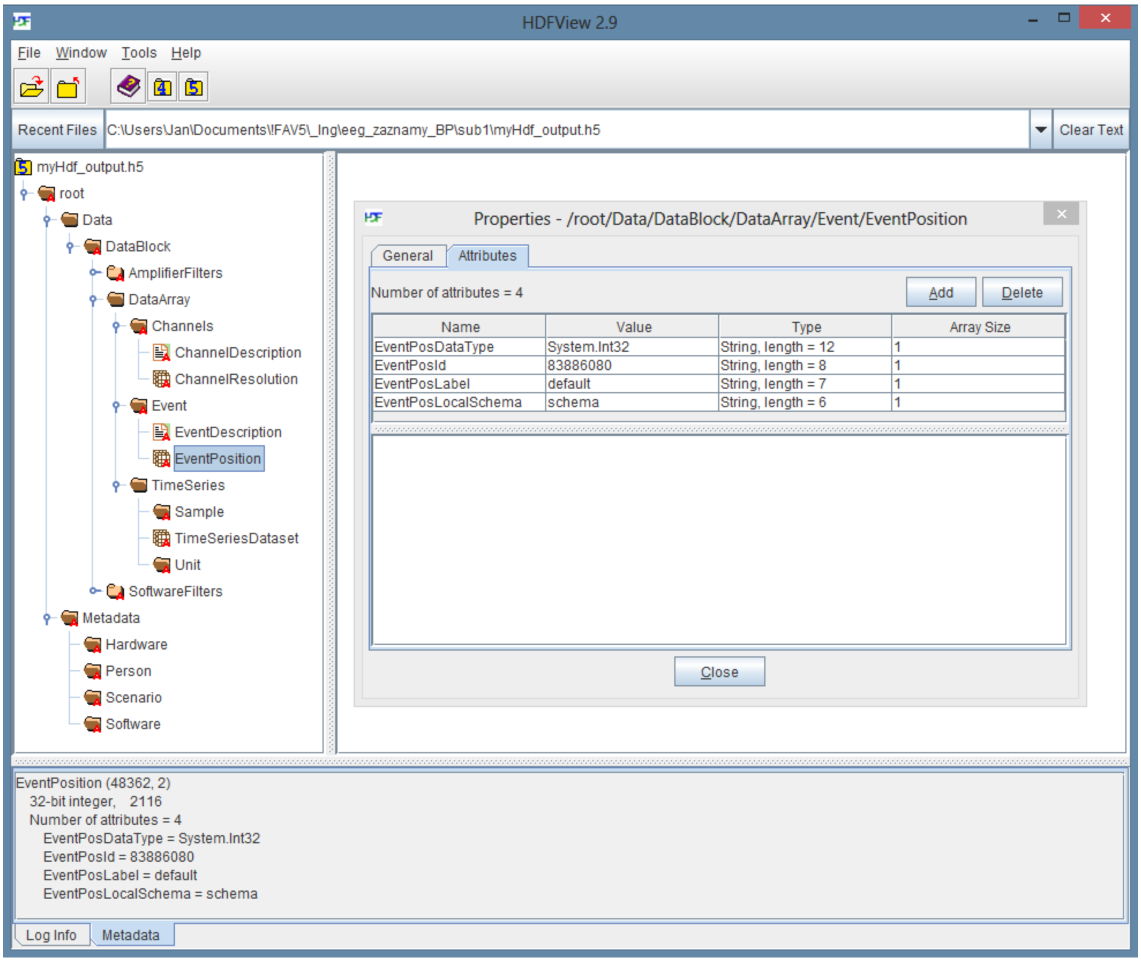
Age 23 ☐ Left Hand

Figure 7:13 - Main window of application

7.5 Testing

The application is designed to save data to HDF5, edit or read created data is not available. For the testing purpose the software HDF View [45] was used. It is a Java based application what allows users reading and editing of existing HDF5 (*.h5*) files.

The HDFView application (Figure 7:14) loads the *.h5* file and shows the structure of file. All groups and dataset are shown. The option to see the attributes inside the file, group and dataset allows checking correct content of entities.



Number of attributes = 4

Name	Value	Type	Array Size
EventPosDataType	System.Int32	String, length = 12	1
EventPosId	83886080	String, length = 8	1
EventPosLabel	default	String, length = 7	1
EventPosLocalSchema	schema	String, length = 6	1

```
EventPosition (48362, 2)
  32-bit integer,   2116
  Number of attributes = 4
    EventPosDataType = System.Int32
    EventPosId = 83886080
    EventPosLabel = default
    EventPosLocalSchema = schema
```

Figure 7:14 - HDFView application

Correctness of the binary *.eeg* file was tested by a method inside the HDF5Manager application. Creation of the binary file back from the dataset was implemented. When the dataset is filled, it is read again and data are written into the binary file with .eeg file extension. The new created binary file with original *.eeg* file by the byte is read for detecting compliance. Further the control file was used as an input to EEGLAB and an analysis was realized successfully.

8 CONCLUSION

The experiment for testing driver's attention was designed, and then 10 measurements were realized. After that the results were analyzed and interpreted. The data-model for storing experimental data was designed and implemented in the HDF5 Manager software.

During experiments the ERP component P300 was successfully localized. The position of the component was interpreted from the averaged epochs. After reading latency values from the epoch waveform, time distances were entered into the tables.

When comparing these values, a partial increase of latency was observed. The latency increased in the second time period compared to the first time period. The third period latency was longer than the second one, but from the third to the fourth period the latency decreased.

Respiratory frequency between the first time period and the second time period is almost the same. Between the second and third time period the frequency decreased, from the third to the fourth time period the frequency stagnated.

Evaluation of galvanic skin response could not be performed, because the recorded signal was distorted. During analysis correlation between the sample waveform morphology and real measured waveform shape was not found.

7 out of 10 measured subjects claimed that the scenario seemed to be tiring and boring for them. They confirmed emotional growth of fatigue after the experiment. From this perspective the experiment was well-designed. The respiratory frequency measurement could be affected by sitting in the car seat and leaning back to the backrest of the seat. However, the analysis of the respiratory signal did not show essential distortion of the signal.

The data model for storing neuroinformatics data was designed, described and implemented. Application was implementation in C# and HDF5 file container was chosen as the output file structure of application.

During the application testing, the correctness of inserted attributes through the HdfView application was tried. Attributes from the output .h5 files can be listed. The binary .eeg file was verified by comparing the original input file to the testing output file.

In the future work it would be fine to investigate and interpret other components following the P300 component. It would be also suitable to find the reason why skin response signal could not be interpreted in this case.

LIST OF ABBREVIATIONS

API – Application Programming Interface

EEG - Electroencephalography

ERP – Event-Related Potentials

GSR – Galvanic Skin Response

GUI - Graphical User Interface

HDF – Hierarchical Data Format

JSON – JavaScript Object Notation

MVVM – Model View ViewModel

PCL – Presentation Control Language

RIP – Respiratory Inductance Plethysmography

SDL – Scenario Description Language

UWB – University of West Bohemia

XAML – Extensible Application Markup Language

LIST OF FIGURES

LIST OF TABLES

BIBLIOGRAPHY

[1] MD-Health. Lobes of the brain. [Online]. [Cited: 2013-February-1]. <http://www.md-health.com/Lobes-Of-The-Brain.html>.

[2] MAUTNER, Pavel; MOUČEK, Roman. *Neuroinformatika - měření mozkové aktivity.* Pilsen : University of West Bohemia, Internal materials, 2010. 16 p.

[3] MD-Health. Parts of the brain and function. [Online]. [Cited: 2013-February-2]. <http://www.md-health.com/Parts-Of-The-Brain-And-Function.html>.

[4] HARRIS, John; *Rewiring neuroscience.* [Online]. [Cited: 2013-February-2]. <http://www.rewiring-neuroscience.com/retina-of-memory-we-caught-breeze-after/>.

[5] SANEI, Saeid; CHAMBERS, J. A. *EEG signal processing*. Chichester: John Wiley & Sons Ltd., 2007. ISBN-13 978-0-470-02581-9.

[6] Scientific American: *Function of the various brainwaves*. [Online]. [Cited: 2013-February-3]. <http://www.scientificamerican.com/article.cfm?id=what-is-the-function-of-t-1997-12-22>.

[7] The McGill Physiology Virtual Lab: *Biomedical Signals Acquisition*. Mc Gill University. [Online]. [Cited 2012-Februar-21]. <http://www.medicine.mcgill.ca/physio/vlab/biomed_signals/eeg_erp.htm>.

[8] LUCK, Steven J. *An introduction to the event-related potentail technique*. Cambridge: MIT Press, 2005. ISBN 97802626221960.

[9] LUCK, Steven J.; WOODMAN, Geoffrey F.; VOGEL, Edward K. *Event-related potential studies of attention*. Trends in Cognitive Sciences [Online] 2000. Vol. 11. [Cited: May 2, 2011]. p. 432 - 440. <http://www.sciencedirect.com>.

[10] MOUČEK, Roman; ŘEŘICHA, Jan. *Driver's Attention during Monotonous Driving.* International Conference on Biomedical Engineering and Informatics (BMEI). Conference article, 2012, p. 486 - 490. ISAN 10.1109/BMEI.2012.6513051. <http://ieeexplore.ieee.org>.

[11] SOLDATI, Nicola. *The Respiration Belt MR: a new device for parallel respiratory measurements*. Brain Products Press Release Volume. Brain Products Press, vol. 46 [Cited: 2013-March-2].

[12] LOPEZ-MEYER, P.; SAZONOV E. and CHEUNG B. *Comparative sensor analysis for an electronic wearable and non-invasive respiratory signal acquisition system*. International Conference on Sensing Technology (ICST). Conference article, 2012, p. 805 – 808. IEEE 978-1-4673-2248-5

[13] Bio-medical: GSR or Galvanic Skin Response. [Online] 2002. [Cited: 2013-April-8]. <http://bio-medical.com/news/2002/05/gsr-or-galvanic-skin-response>.

[14] TARVAINEN, M.P., et al. *Principal component analysis of galvanic skin responses*. Chicago: Annual EMBS International Conference, 2000, p. 3011 - 3014. 0-7803-6465.

[15] RANTA-AHO, P.O., et al. *On Correlation Between Single-Trial ERP and GSR Responses: a Principal Component Regression Approach.* New York City: EMBS Annual Internation Conference, 2006, p. 5499 - 5502. ISBN 1-4244-0033-3.

[16] LUCK, Steven. J. *Ten Simple Rules for Designing and Interpreting ERP Experiemtns.* 2003. University of Iowa.

[17] BENNINGTON, J. Y.; POLICH John. *Comparison of P300 from passive and active tasks for auditory and visual stimuli*. International Journal of Psychophysiology. Elsevier, 1999, vol. 34, p. 171 - 177

[18] CHUNHAU, P., et al. *Analysis of P3 in a continuous 40-min auditory oddball task*. International Congress Series. Elsevier, 2005, vol. 1278, p. 105 – 108

[19] DE ROSARIO, H., et al. *Controlled inducement and measurement of drowsiness in a driving simulator*. IET Intelligent Transport Systems. The Institution of Engineering and Technology, 2010, vol. 4, p. 280 - 288. ISMN ISSN 1751-956X.

[20] SYNETIC GmbH & Co. KG. Synetic the Funfactory. *World Racing 2.* [Online] 2005. [Cited 2012-November-29]. <http://www.synetic.de/flash_frameset.php>.

[21] Neuro Behavioral Systems. *Presentation.* [Online]. [Cited: 2013-January-9]. <http://www.neurobs.com/presentation/docs/index_html>.

[22] Brain Products. *BV Recorder technical specifications.* [Online] 2002. [Cited: 2013-January-8]. <http://www.brainproducts.com/productdetails.php?id=21&tab=2>.

[23] Swartz Center for Compurational Neuroscience. *EEGLAB*. [Online]. [Cited: 2013-January-13]. <http://sccn.ucsd.edu/eeglab>.

[24] ERP Toolbox Home: ERP info. *ERPLAB Toolbox* [Online]. [Cited: 2013-January-14]. <http://erpinfo.org/erplab>.

[25] EEG method. *ADInstruments* [Online]. [Cited: 2013-January-22]. <http://www.adinstruments.com/solutions/research/applications/eeg#method>.

[26] BRAIN PRODUCTS. *GSR-MR Module*. 2012. Operating Instructions. [Online]. [Cited: 2013-March-22]. <www.brainproducts.com>.

[27] BRAIN PRODUCTS. *Respiration Belt*. 2011. Technical specification. [Online]. [Cited: 2013-March-22]. <www.brainproducts.com>.

[28] ŘEŘICHA, Jan. *Driver's Attention and Auditory Stimulation (ERP Experiment)*. Pilsen: 2011. Bachelor Thesis. University of West Bohemia Faculty of Applied Sciences, Department of Computer Science and Engineering.

[29] RADA, Jan. *Pozornost řidiče při monotónní jízdě a sluchová stimulace (ERP experiment)*. Pilsen: 2012. Bachelor Thesis. University of West Bohemia Faculty of Applied Sciences, Department of Computer Science and Engineering.

[30] STOEWER, Aadrian; GREWE, Jan. *Data model/container for electrophysiological data*. 2012. Draft.

[31] LIANG, B., et al. *The Neurophysiology Data Translation Format (NDF)*. 2010. Specification v.1.2.1.

[32] Neuroshare. *Neuroshare.* [Online]. [Cited: 2012-November-6]. <http://neuroshare.sourceforge.net/index.shtml>.

[33] Neuroshare. Neuroshare Native File Format Specification. [Online]. 2003, 0.9.d [Cited: 2012-October-30]. <http://neuroshare.sourceforge.net/NSN-Documentation/NSNFileFormat-0-9d.pdf>.

[34] Neuroshare. Neuroshare API Specification. [Online]., 1.3 [Cited: 2012-November-2]. <http://neuroshare.sourceforge.net/API-Documentation/NeuroshareAPI-1-3.pdf>.

[35] Neo: *Neo.* [Online]. [Cited: 2012-November-20]. <http://pythonhosted.org/neo/>.

[36] INCF Eelectrophysiology HDF5 Working Group. *epHDF: A proposed standard for storing neuroscience electrophysiology data in HDF5*. 2013.

[37] INCF Eelectrophysiology HDF5 Working Group. *HDFds - Conventions to facilitate data sharing using HDF5*. 2013.

[38] PFEIFFER, A., I. BAUSCH-GALL a M. OTTER. *Proposal for a Standard Time Series File Format in HDF5*. Oberpfaffenhofen. BAUSCH-GALL, DLR Institue of System Dynamics and Control.

[39] HDF GROUP. *The HDF Group* [Online]. [Cited: 2012-November-8]. <http://www.hdfgroup.org/HDF5/>.

[40] HDF GROUP. *HDF5 User's Guide*. [Online]. 2012, Release 1.8.10 [Cited: 2012-November-8]. <http://www.hdfgroup.org/HDF5/doc/UG/UG_frame03DataModel.html>.

[41] HDF5DotNET. *HDF5DotNet wrapper.* [Online]. [Cited 2012-December-6]. <http://hdf5.net/default.aspx>.

[42] Dario Airoldi TechNotes: *MSDN Blogs* [Online]. [Cited: 2013-April-15]. <http://blogs.msdn.com/b/darioa/archive/2009/01/03/key-reasons-why-use-wpf-on-a-business-application.aspx>.

[43] Tales from the Smart Client: *MSDN Blogs* [Online]. [Cited: 2013-April-15]. <http://blogs.msdn.com/b/johngossman/archive/2005/10/08/478683.aspx>.

[44] Brain Products. *BrainVision Recorder*. 2010. User Manual.

[45] The HDF Group. *HDF View.* [Online]. [Cited: 2013-March-2]. <http://www.hdfgroup.org/hdf-java-html/hdfview/>.

[46] Emka Technologies. Respiratory belt. [Online]. [Cited: 2013-April-10]. <http://pulmonary.emka.fr/prod.php?prod=6>.

APPENDIXES

APPENDIX A

In next Figures averaged epochs of subject 5 are. Epochs were averaged on Fz, Cz, Pz electrodes.

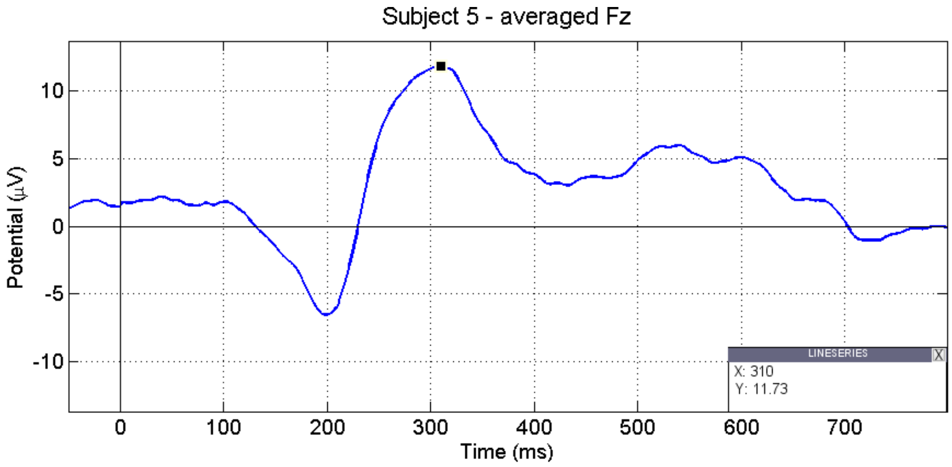


Figure A:1 - Averaged epoch of subject 5 on Fz

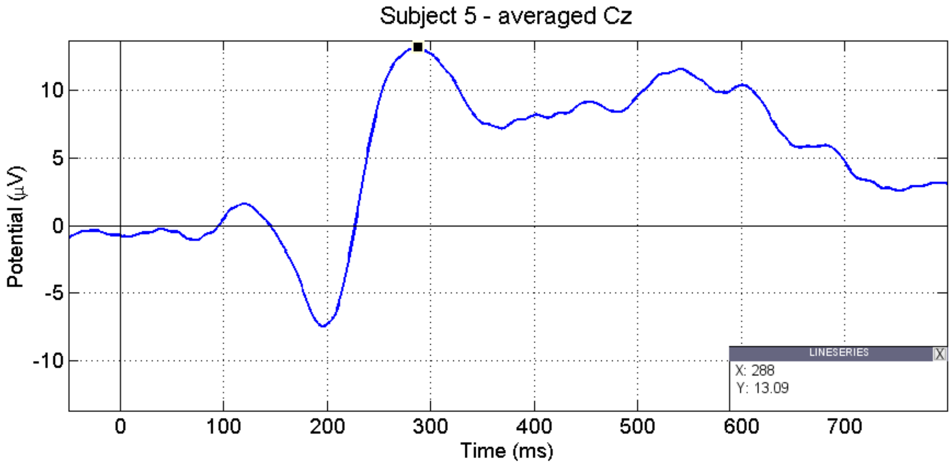


Figure A:2 - Averaged epoch of subject 5 on Cz

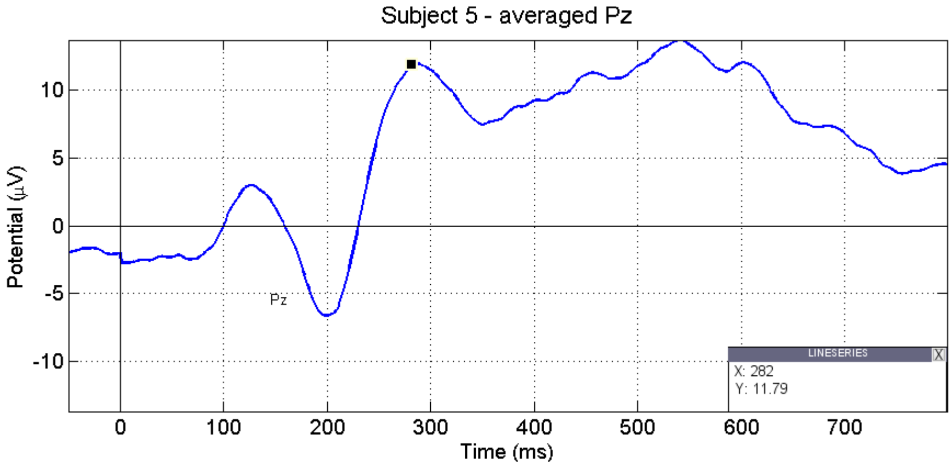


Figure A:3 - Averaged epoch of subject 5 on Pz

APPENDIX B

In next Figures epochs on Fz electrode of subject 5 are shown in every period, average and comparing.

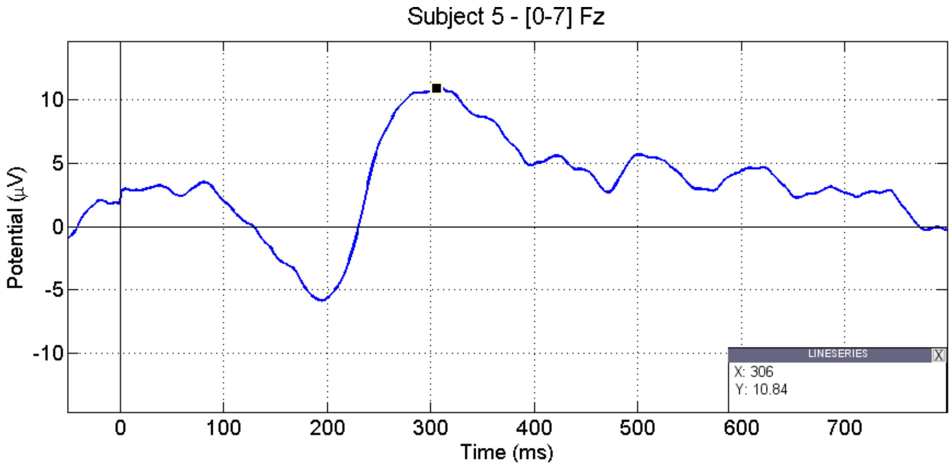


Figure B:1 - Subject 5 in Period 1

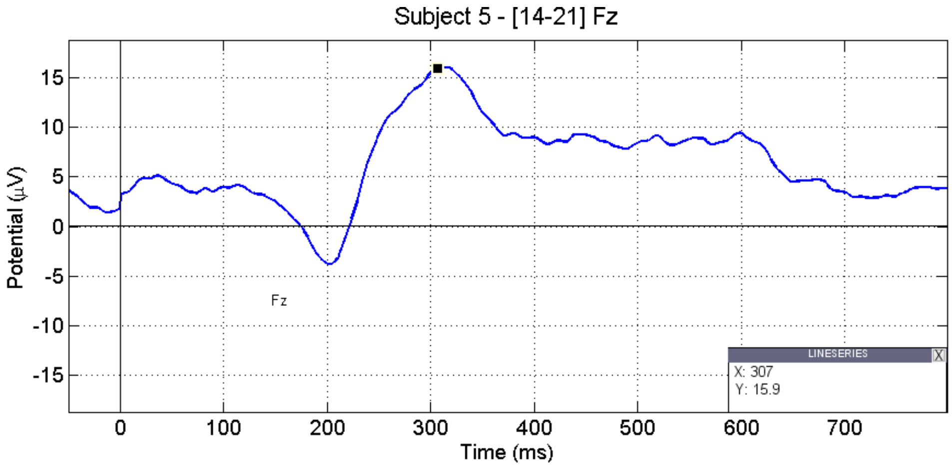


Figure B:2 - Subject 5 in Period 2

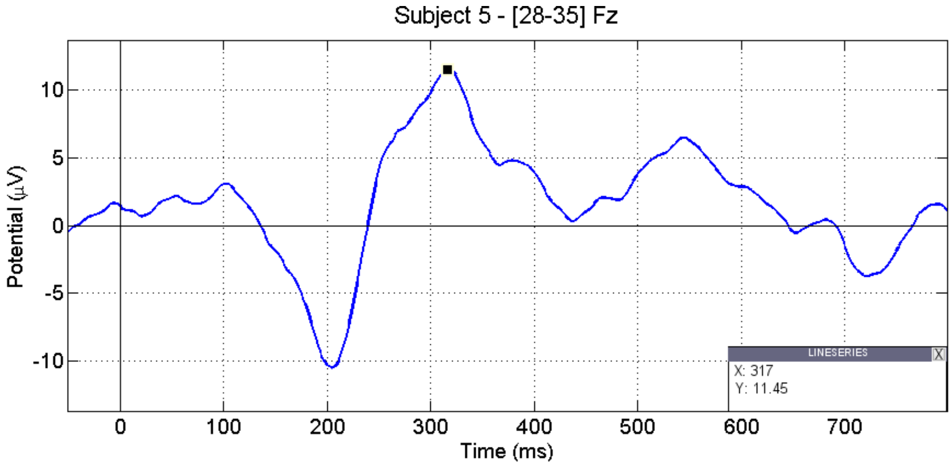


Figure B:3 - Subject 5 in Period 3

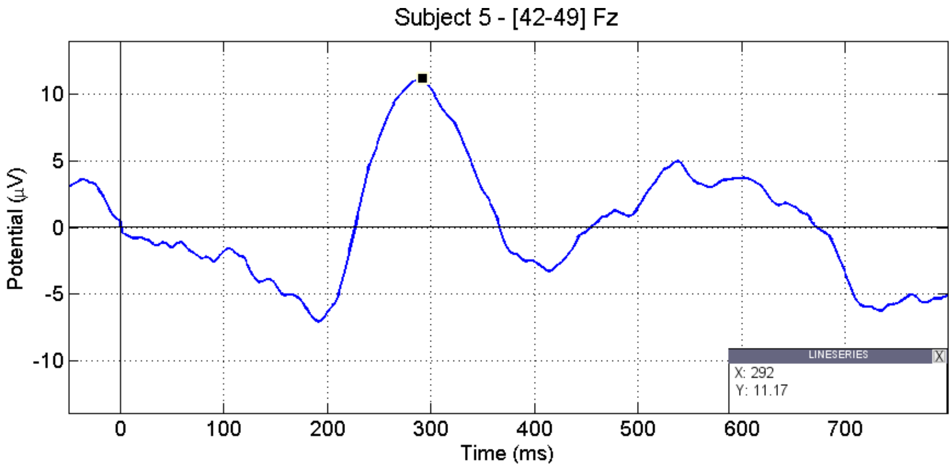


Figure B:4 - Subject 5 in Period 4

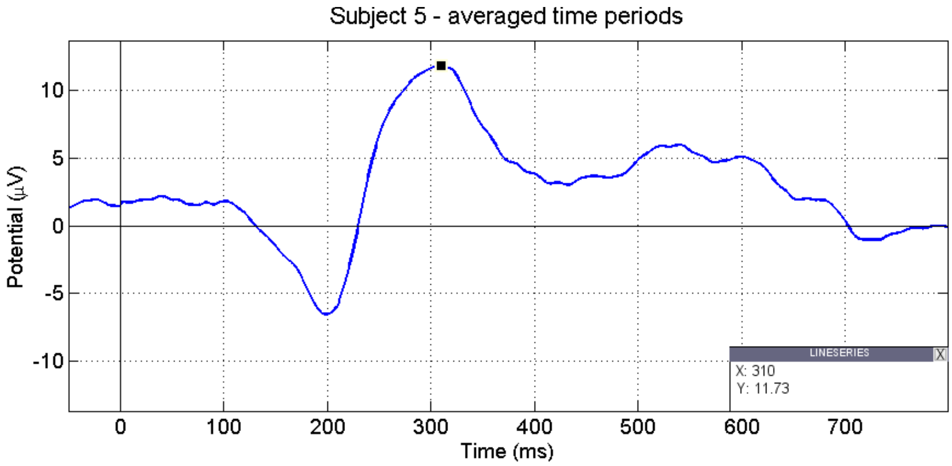


Figure B:5 - Subject 5 average of Periods

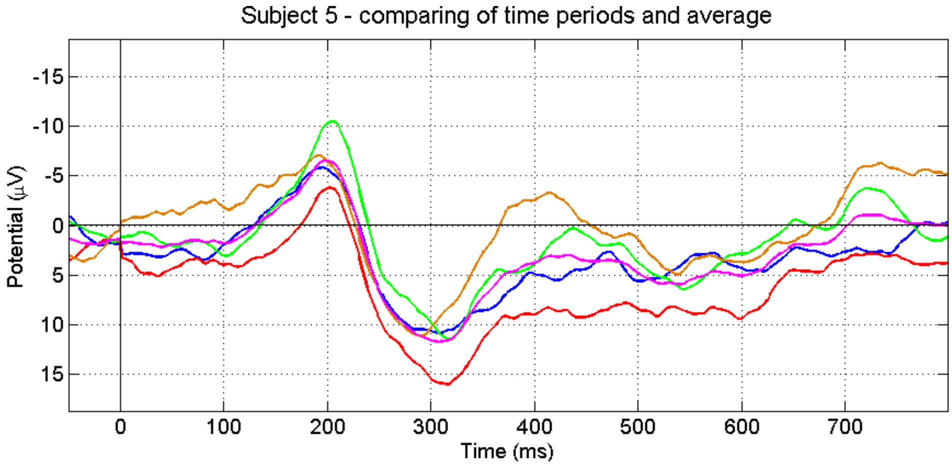


Figure B:6 - Comparison of Periods and average

APPENDIX C

Data part of Data Model

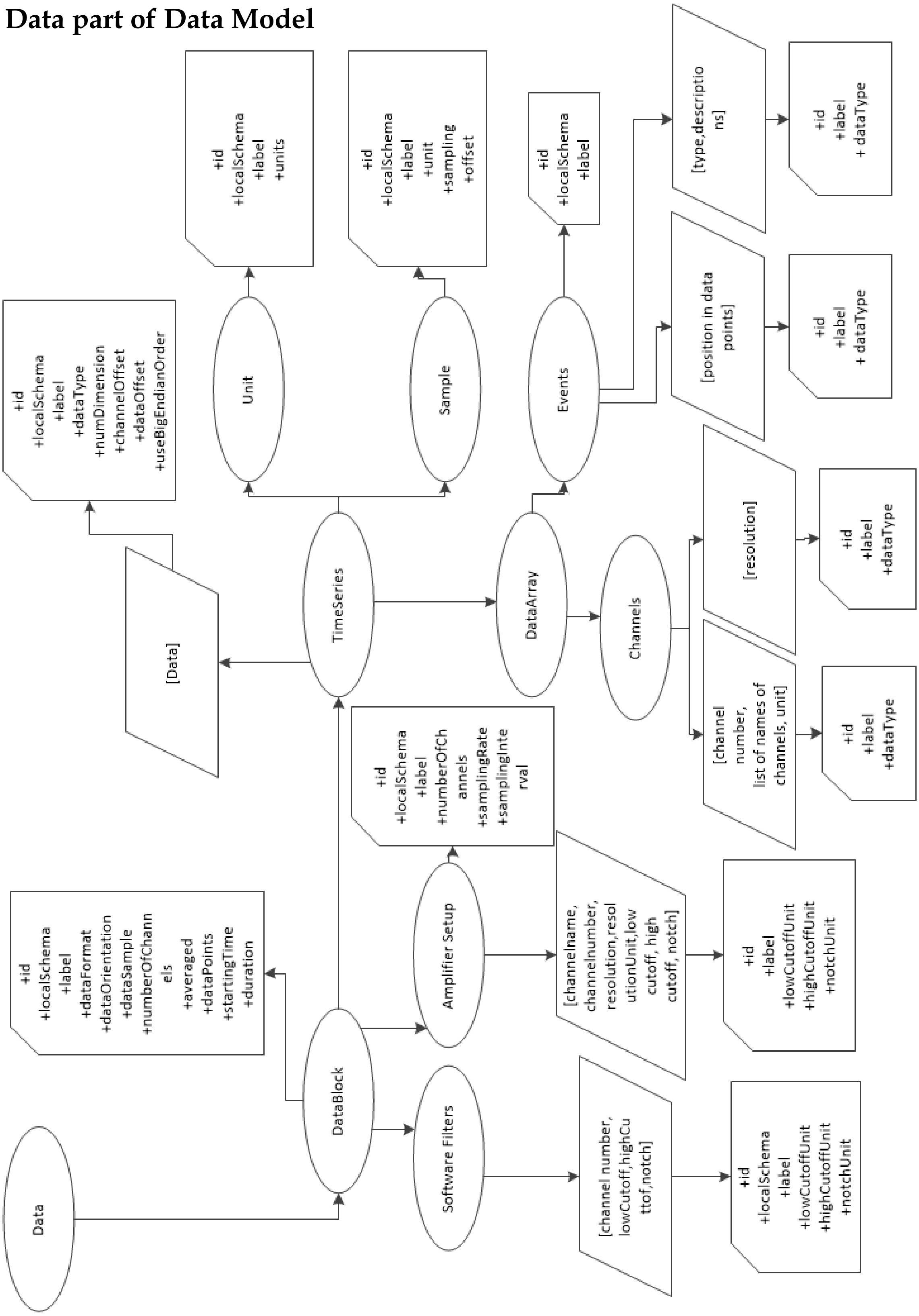


Figure C:1 - Data part of data model

Complete Data Model

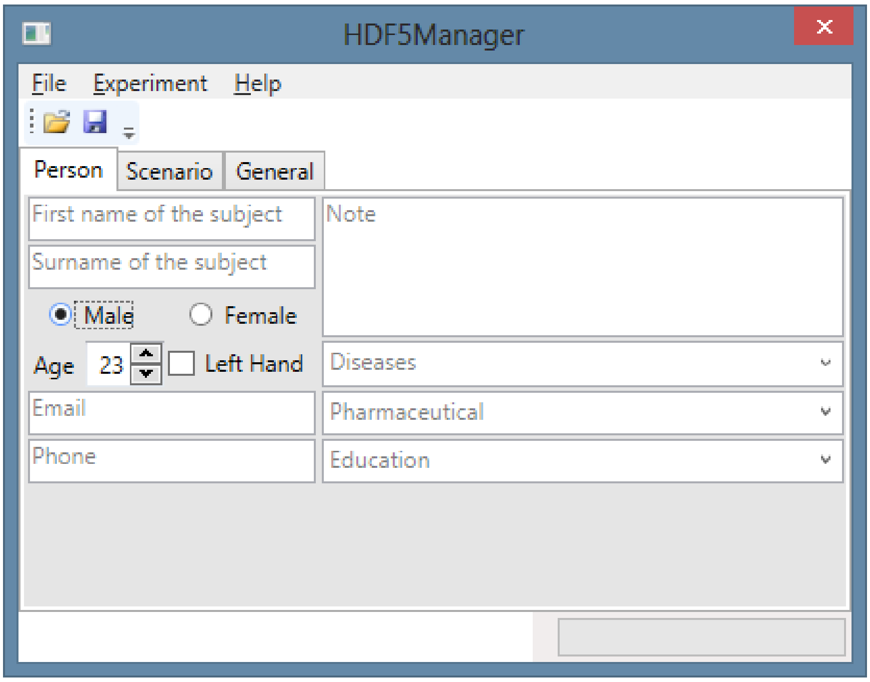
Figure C:2 - Complete schema of data model

APPENDIX D

User Manual

Application runs under Windows 8 and .NET 4.0 has to be installed.

The main page of the application contains three tabs for inserting basic metadata and menu with several options.

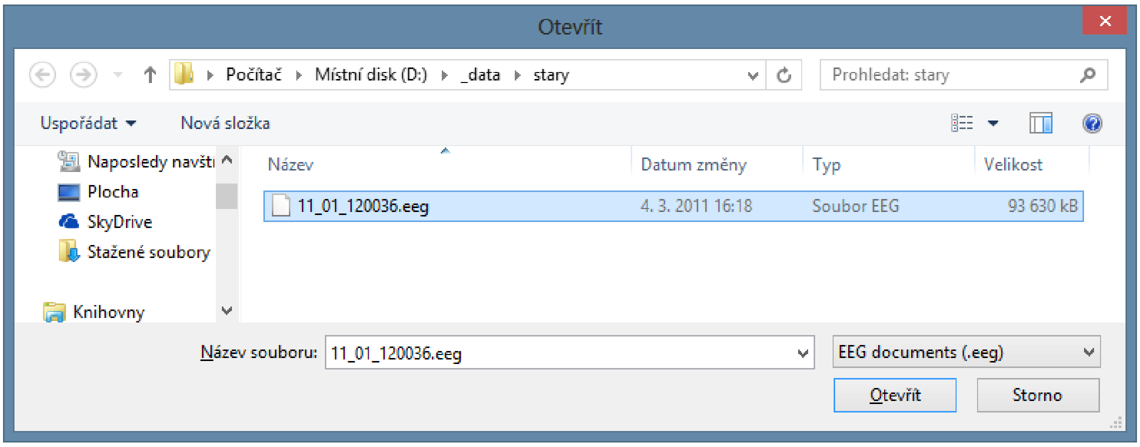


● Male ○ Female

Age 23 ☐ Left Hand

Figure D:1 - Main Window

The first step is to load the original files. The program supposed that .vhdr and .vmrk files are in the same folder as .eeg file. Therefore only the .eeg file will be selected, loading of next two files is done automatically.



Název	Datum změny	Typ	Velikost
11_01_120036.eeg	4. 3. 2011 16:18	Soubor EEG	93 630 kB

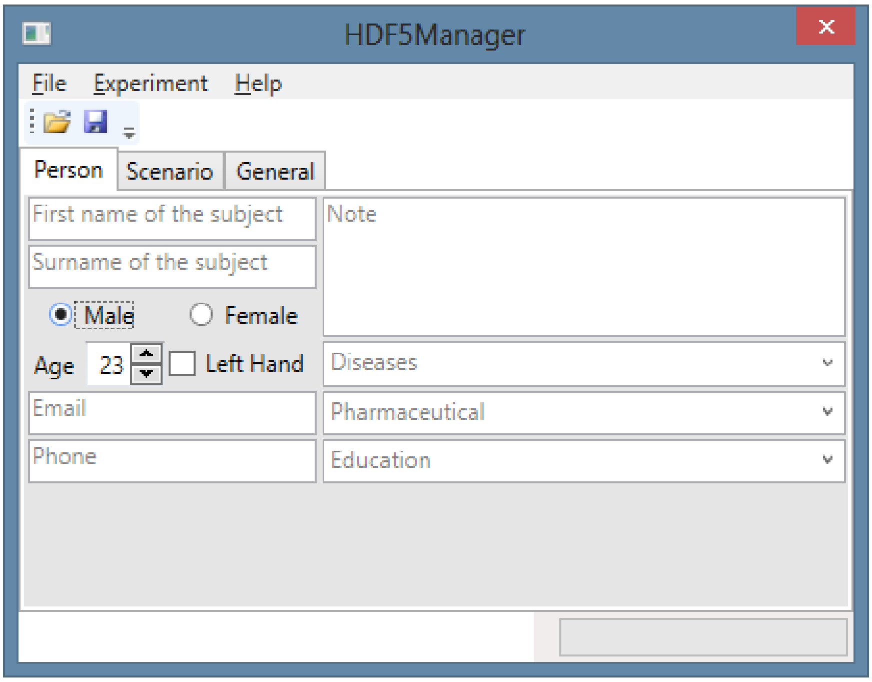
Název souboru: 11_01_120036.eeg

EEG documents (.eeg)

Figure D:2 - Load file dialog

After loading the original data, metadata should be filled before exporting the file into the hdf file.

When the application is started, the main window has set “Person” tab by default. Information about tested person should be inserted here.

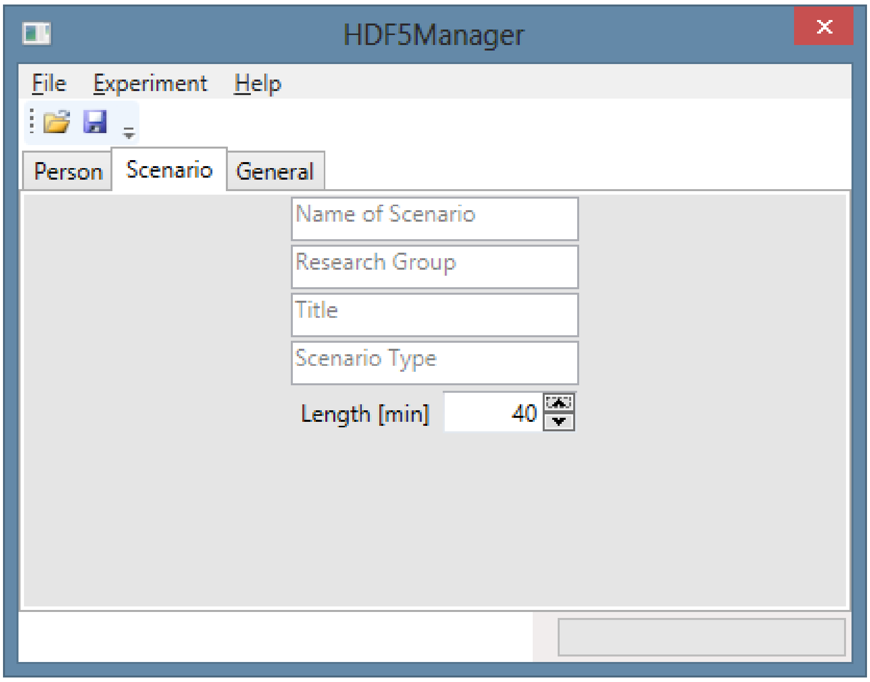


● Male ○ Female

Age 23 ☐ Left Hand

Figure D:3 - Person tab

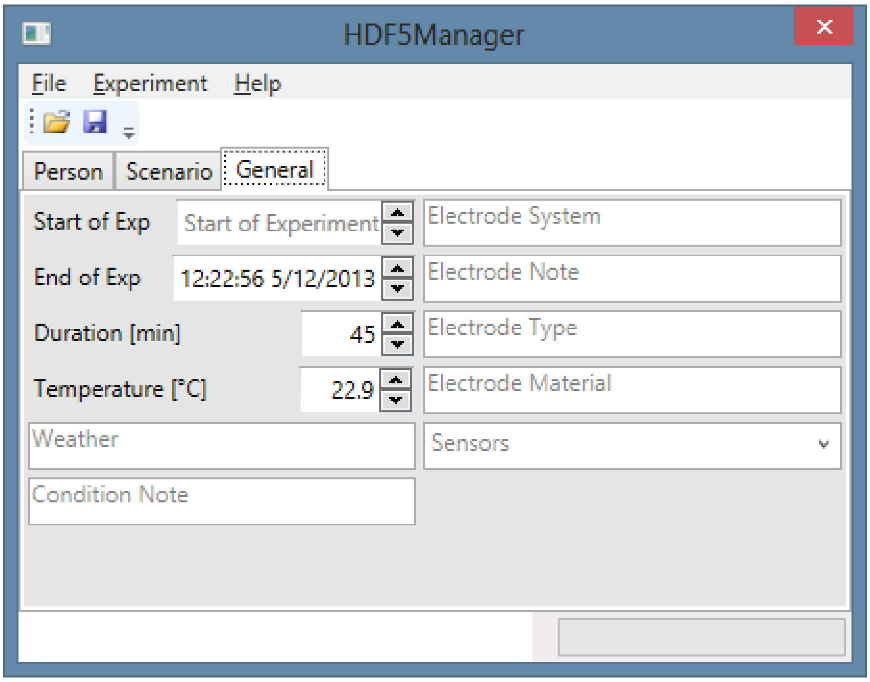
The second tab servers for inserting scenario information.



Length [min] 40

Figure D:4 - Scenario tab

The third tab General contains inputs for general information about experiment and environment.



End of Exp 12:22:56 5/12/2013

Duration [min] 45

Temperature [°C] 22.9

Figure D:5 - General tab

Other and advanced inputs are available in the menu. The information about hardware, software and other properties can be inserted.

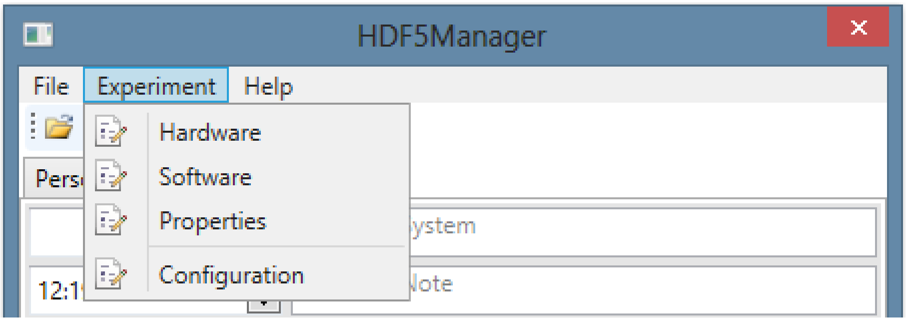


Figure D:6 - Experiment menu

Window for setting hardware properties.

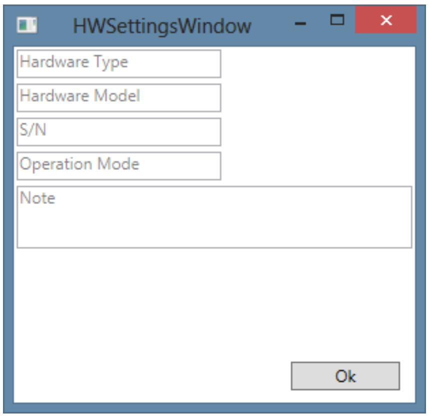


Figure D:7 - Hardware window

Window for information about used software.

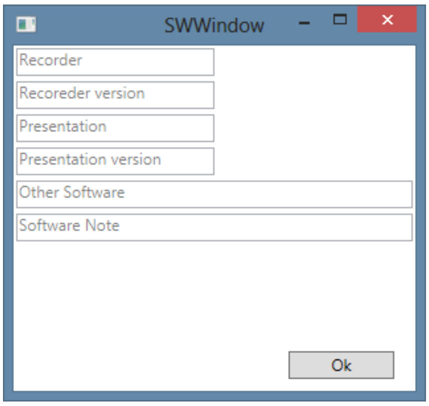


Figure D:8 - Software window

Other general information about experiment.

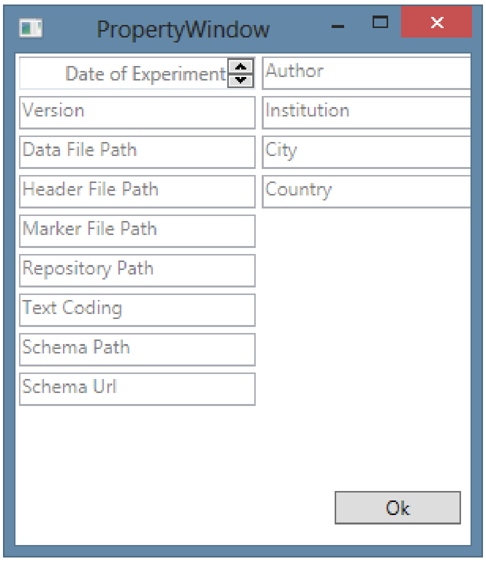
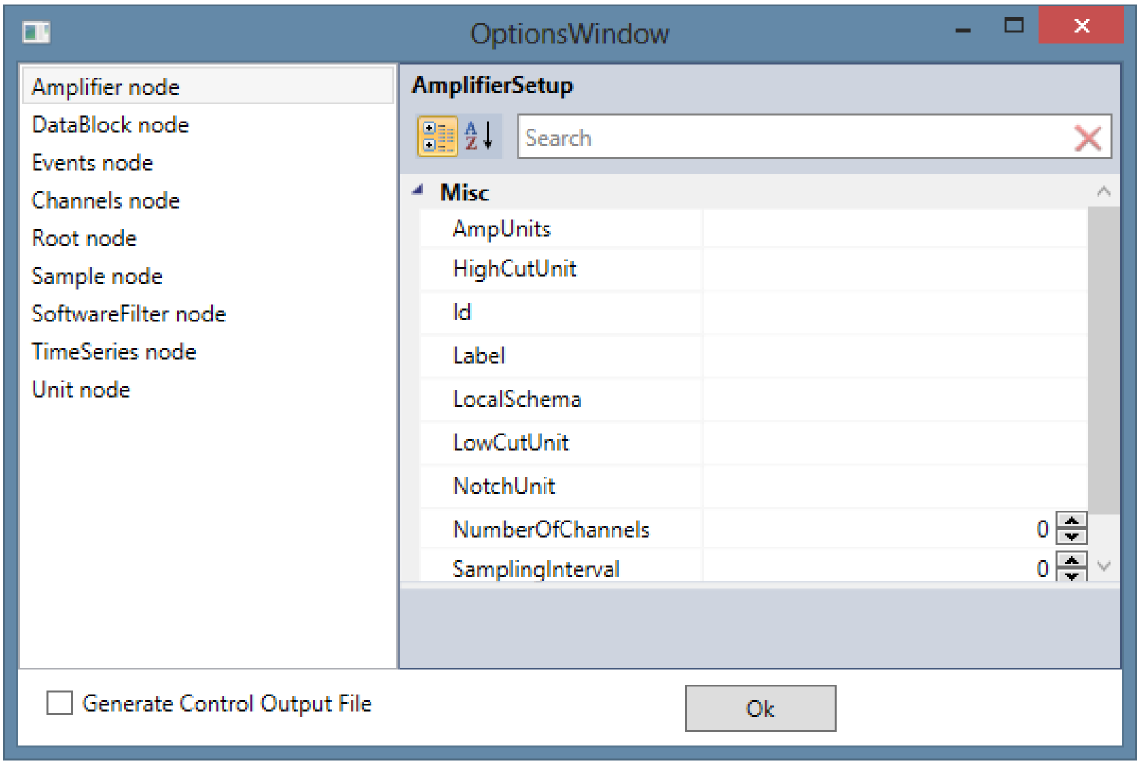


Figure D:9 - Property window

In the option window the grid connected with all properties is located. The setting of the properties is not a safe operation (the inputs are not validated), direct change of all properties is enabled.

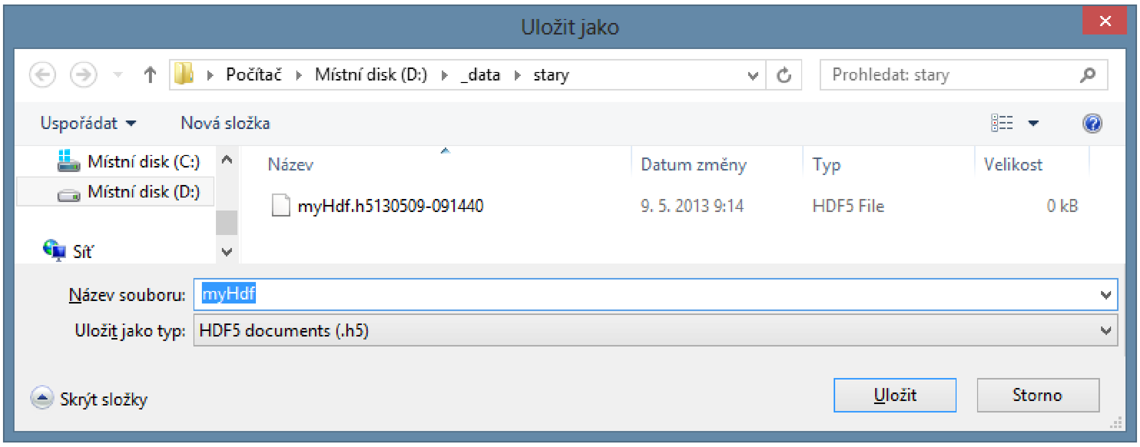


AmpUnits
HighCutUnit
Id
Label
LocalSchema
LowCutUnit
NotchUnit
NumberOfChannels 0
SamplingInterval 0

☐ Generate Control Output File

Figure D:10 - Options window

The last step is to export data / metadata to the HDF5 (.h5) file.



Název	Datum změny	Typ	Velikost
myHdf.h5130509-091440	9. 5. 2013 9:14	HDF5 File	0 kB

Název souboru: myHdf

Uložit jako typ: HDF5 documents (.h5)

Figure D:11 - Save file dialog

APPENDIX E

The personal questionnaire

Tested person information:

- Name
- Surname
- Gender
- Left/right handed
- Age
- Diseases
- Pharmaceutical
- Education
- Telephone number
- Email
- Subjective feelings before experiment
- Subjective feelings after experiment
- Note

The experimental questionnaire

Experimenter information:

- Name
- Surname
- Email
- Employ/ student

Experimental conditions:

- Date and time of experiment start
- Date and time of experiment end
- Duration of experiment
- Temperature
- Weather
- Used hardware
- Used sensors
- Used software
- Electrode system
- Electrode note
- Electrode type
- Electrode material
- Important notes

Scenario information:

- Name of scenario
- Research group
- Description
- Duration

APPENDIX F

The measuring process declaration

The steps of measurement are:

- A person is introduced into the measuring process and the description of the required task is presented to him.
- The EEG cap is put on and the conductive gel is applied
- Together with measuring EEG the physiological sensors can be used (Respiration belt, GSR sensor).
- The connection of EEG channels and sensors is realized.
- The person is asked to filling the questioner.
- The stimuli presentation is started.
- The recorded signals / data will be saved into the local database or will be uploaded to the internet portal. The database and internet portal are available only for authorized staff. The data will not be freely accessible.
- After measurement the basic hygiene tools are provided.

Conditions for participation in the experiment

- Participation is voluntary.
- The person is older than 18 years (or the guardian agrees).
- The person declares that he/ she do not know any disease or other complication affecting person's health.
- Person has to sign these conditions.

Declaration

I hereby that I was familiar with the terms, risks and experimental process.

I hereby that I understand all of the terms and consequences of the experiment.

I declare that I have provided correct and true information.

I agree with storing and analyzing recorded data.

I declare that I sign this document voluntarily.

Pilsen, …………… ……………………

Signature

APPENDIX G

Content of the attached DVD:

- Data – original records in the Brain Vision data format (.eeg, .vhdr, .vmrk)
- Datasets – processed data in the EEGLAB dataset format (.set, .fdt)
- Documentation – master thesis document (.docx, .pdf)
    - Figure
    - Tables
- Software – freeware software used for evaluation of data (EEGLAB, ERPLAB, GrandAverage plugin)
- Application:
    - bin – executable file (.exe)
    - src – source codes
    - VS2012 – Visual Studio 2012 project (.sln)