CONCURRENCY AND COMPUTATION: PRACTICE AND EXPERIENCE
Concurrency Computat.: Pract. Exper. 2010; **00**:1–20
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/cpe

Monitoring and Improving Performance in Human-Computer Interaction

Davide Carneiro[1], André Pimenta[1], Sérgio Gonçalves[2], José Neves[1] and Paulo Novais[1]*

[1] *Algoritmi Centre/Department of Informatics, University of Minho, Braga Portugal*
[2] *Informatics Department, University of Vigo, Ourense, Spain*

SUMMARY

Monitoring an individual's performance in a task, especially in the workplace context, is becoming an increasingly interesting and controversial topic in a time in which workers are expected to produce more, better and faster. The tension caused by this competitiveness, together with the pressure of monitoring, may not work in favour of the organization's objectives. In this paper we present an innovative approach on the problem of performance management. We build on the fact that computers are nowadays used as major work tools in many workplaces to devise a non-invasive method for distributed performance monitoring based on the observation of the worker's interaction with the computer. We then look at musical selection both as a pleasant and as an effective method for improving performance in the workplace. The proposed approach will allow team coordinators to assess and manage their co-workers' performance continuously and in real-time, using a distributed service-based architecture.

Received . . .

KEY WORDS: Performance Monitoring; Attention; Distributed Architecture; Music

1. INTRODUCTION

In an organizational context, performance management is a process by which coordinators and employees continuously work together to plan, monitor and review an employee's work objectives and overall contribution to the organization. The main aim of such processes is to promote and improve employee effectiveness. Traditionally, performance measurement includes indicators such as quantity, quality, timeliness, cost-effectiveness or absenteeism.

Until a few years ago, performance management consisted of regular (generally yearly) meetings, in which the performance of a worker was compared against the established goals, possibly resulting in new goals and other actions aimed at maintaining or improving the worker's productivity. In recent years, electronic performance monitoring emerged, allowing a continuous and real-time assessment of the worker's performance. From the point of view of management, it represents an efficient and effective way of carrying out this task. However, from the point of view of workers, knowing that their throughput is being constantly monitored may result stressful (especially if they believe productivity goals to be excessive) [1], is often seen as an invasion of privacy or as ethically reprehensible [2], and does not necessarily have a positive effect on productivity [1].

In this paper we propose an alternative approach for the problem of performance management in the workplace. Its most innovative aspect is that it is not based on the described traditional performance indicators, inherited from organizational sciences and tendentiously oriented to

*Correspondence to: Algoritmi Centre/Departamento de Informática, Universidade do Minho, Braga Portugal productivity. We propose a set of different indicators that quantify the performance of the worker's interaction with the computer which, as will be seen ahead, is a good indicator of task performance. We are more concerned with *how* a worker produces, rather than *how much*. Moreover, these indicators also predict cognitive processes such as stress [4], attention or mental fatigue [3], important when it comes to manage performance.

We also look at performance assessment and performance management as two distinct problems. Performance assessment concerns simply the quantifying of one's production. To this end, traditional indicators could suffice. Performance management, on the other hand, concerns not only performance assessment but also understanding the reasons underneath performance indicators and finding sustainable ways to improve them.

We detail a distributed architecture that allows for a team coordinator to acquire data, in real-time, about the proposed indicators. The architecture takes this data and builds high-level information regarding the workers' levels of stress, mental fatigue and attention. From the point of view of the team coordinator, this information is very important in the sense that it may allow him to contextualize and understand a given change in performance, rather than just observing it. Finally, the proposed architecture uses musical selection to act on each individual worker according to the coordinator's and/or the worker's aims.

In past work we have explored these issues preliminarily. Namely, we have devised an architecture to support the intended functionalities [20] and we have studied the potential effects of music on the individual's performance [19]. This preliminary research showed that it is possible to acquire the intended features in real-time, from a distributed system. Moreover, it also showed that music has indeed a measurable effect on the performance of our interaction patterns.

Having verified both these issues, in this paper we bring these two lines of research together to address the monitoring of performance in human-computer interaction, especially in what concerns the influence of attention, and on how music can be used to improve performance indicators. Especially, we detail the process of data acquisition and feature extraction and selection, which are fundamental for the whole process.

The rest of the paper is organized as follows. The following two subsections address, respectively, the issues of performance in human-computer interaction and the influence of music on the human being, especially in performance. Section 2 addresses the distributed architecture developed to implement the proposed approach, detailing the acquisition of data from multiple users and its transformation into useful features. Section 3 details how performance is monitored, exemplifies how the proposed approach can be used and depicts some results and its validation process. Similarly, section 4 addresses the issue of improving performance through music, detailing the results of the study carried out, the validation process and the developed music recommendation service. The paper ends with Section 5, which presents some concluding remarks.

1.1. Performance in Human-Computer Interaction

Personal computers are increasingly present in our lives. As they gradually take over the space of other objects such as televisions, radios or books, they start to become irreplaceable. In this process, the time we spend interacting with computers also increases significantly, often to the point of surpassing the time we spend interacting with co-workers, family or friends. This is especially true in the workplace. The importance of this interaction grew to the point of leading to the emergence of the field of Human-Computer Interaction (HCI), whose main aim is to improve the interaction between users and computers by making computer interaction mechanisms adapt to the users and the tasks instead of the other way around. In a more long-term perspective, HCI aims to design systems that minimize the barrier between the Human's cognitive model of what they want to accomplish and the computer's understanding of the user's task, as well as help in solving real world problems [5, 6].

The present work can be framed in this field of research in the sense that the performance of the computer user while using the mouse and keyboard can, itself, be used as an interaction means. In the long-term we envision software or hardware that can adapt or react to the performance of their users. At the moment, nonetheless, we aim only at the development of reporting tools that can support the team coordinator's decision making processes.

DOI: 10.1002/cpe

The central concept in these reporting tools is, as mentioned before, the performance of the user's interaction with the computer while using the mouse and keyboard. Performance is generally seen as the combination of the speed and success/accuracy of an individual when carrying out a task. Perhaps the most widely used indicator is the observation of the degradation of task performance over time or in response to changes in task demands. Welford distinguished four types of performance changes that may occur [7]: (1) impairment of sensory or perceptual functions; (2) slowing of sensory-motor performance; (3) irregularity of time; and (4) disruption of performance.

Paradoxically, similar effects on performance emerge when workload is significantly below the adequate level. In such scenarios the individual tends to get tired and bored, which will likely also result in decreased performance [8]. This is one of the challenges in using performance indicators: how to distinguish between boredom and excessive workload. This issue can be addressed with the use of contextual information about the user (e.g. time on task, measures of productivity), as will be seen ahead in this paper.

Another challenge is that a decrease in the performance is often not "seen" immediately, which may be explained by the *reverse capacity* phenomenon [9]. The theory of reverse capacity states that an individual rarely works at his maximum capacity, and can temporarily absorb the load of additional cognitive or physical work while still maintaining performance indicators, as long as their inverse capacity is not exceeded. For this reason, a performance drop due to fatigue, for example, may not be immediately visible [9]. The effects of fatigue on performance, in particular, may only be felt in the course of time, when it may be too late to avoid them. Relevant knowledge about the user's context, profile and workload are thus crucial for an accurate assessment of mental fatigue, especially when this assessment is based on performance indicators.

The indicators considered in this work for quantifying performance stem from the use of the keyboard and the mouse. Namely, we look at aspects such as the velocity of the mouse, how much it is moving in a straight line or the typing velocity or rhythm. These interaction features are described in detail in Section 2.2.

1.2. Music and its Influence

Music is one of the oldest forms of cultural expression, with rudimentary instruments such as flutes carved from animal bones as old as 40000 years. It is and it has been for millennia, a very important part of the way of life of many people in many different cultures. This is, very likely, due to the many different ways in which music influences us. Indeed, music can affect aspects as different as emotions, physiology, behaviour, cognitive functions, health, well-being and physical or cognitive performance.

In our research lab, as a prior validation of the hypothesis of the study carried out and of the distributed data collection architecture described further ahead, we had the opportunity to objectively measure such effects. Figure 1 shows a plot of the galvanic skin response, also known as skin conductance, of an individual listening to two different types of music. Until before minute 8, the individual listens to relatively calm music, after which some vigorous music starts playing. As Figure 1 shows, musical selection alone is enough to arouse the individual significantly, as evidenced by the change in the skin conductance. We have observed related effects in other indicators such as hearth and respiratory rate.

Given its potentially positive effects on the Human being at so many different levels, music has been studied in the last decades for many different purposes.

In commercial environments such as shopping malls or stores, the strategies for selecting music are often driven by the objective of activating people, inducing an increased shopping behaviour. Moreover, [10] concludes that specific types of music should be used in particular areas of the store to appeal to different types of consumers.

The effects of music on health and well-being are also well-known. Hatem et al. [11] show how music can improve recovery time, recovery quality and pain indicators of children following cardiac surgery. Music is also medically used to control emotions and reduce anxiety and stress. Miluk-Kolasa et al. examined the reduction of pre-surgical anxiety using an individually designed music

DOI: 10.1002/cpe

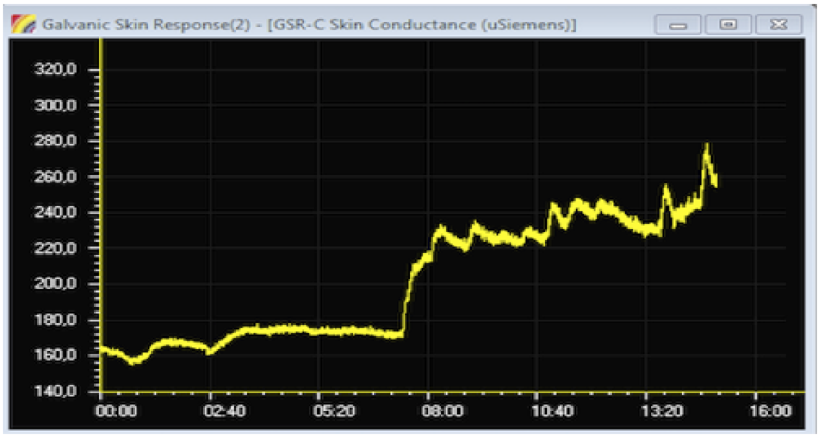


Figure 1. Example of the physiological effect of music on skin conductance. The individual becomes more aroused near minute 8, when he starts listening to more aggressive music.

program [12], with effects on systolic blood pressure, heart rate, cardiac volume, blood glucose and skin temperature.

Concerning physical performance in sports, Bacon et al. [18] determined, in a cycling study, that participants who cycled in time to the music required 7% less oxygen to do the same work as cyclists who did not synchronize their movements with background music.

This last research work is particularly interesting in the context of this paper as it shows how music can improve human performance. According to recent research [18], this happens as follows. During any activity, the human brain is constantly monitoring itself. When fatigue or boredom starts to set in, the human body suffers some transformations, such as the rise in the level of lactate after prolonged physical exercise. What music achieves is that it competes with this physiological feedback for the brain's conscious attention, decreasing the apparent importance of these indicators. Moreover, music can also have an effect on motivation or emotions, driving people to *want* to perform better.

In this paper we aim to explore these characteristics to manage the performance of a group of people working together, once the objectives of the environment are known and once we know how each individual is affected by a particular type of music.

2. ARCHITECTURE

Taking into account what has been stated in the introductory section, the main objective of this work can now be properly defined. We aim to develop an approach for assessing the performance of a group of users, in real time, and to manage it according to the objectives of the group coordinator. Performance will be assessed from indicators that quantify the performance of the interaction with the computer and will be managed through musical selection.

From a technological point of view this represents several challenges. First of all, it implies the need to monitor performance indicators of several different users in real-time. This information must then be made accessible to a decision-support system that, based on the environment's or the coordinator's objectives, will output decisions or guidelines. These may range from recommendations such as a given user making a pause, to playing, pausing or changing music playing on sound speakers or for individual users.

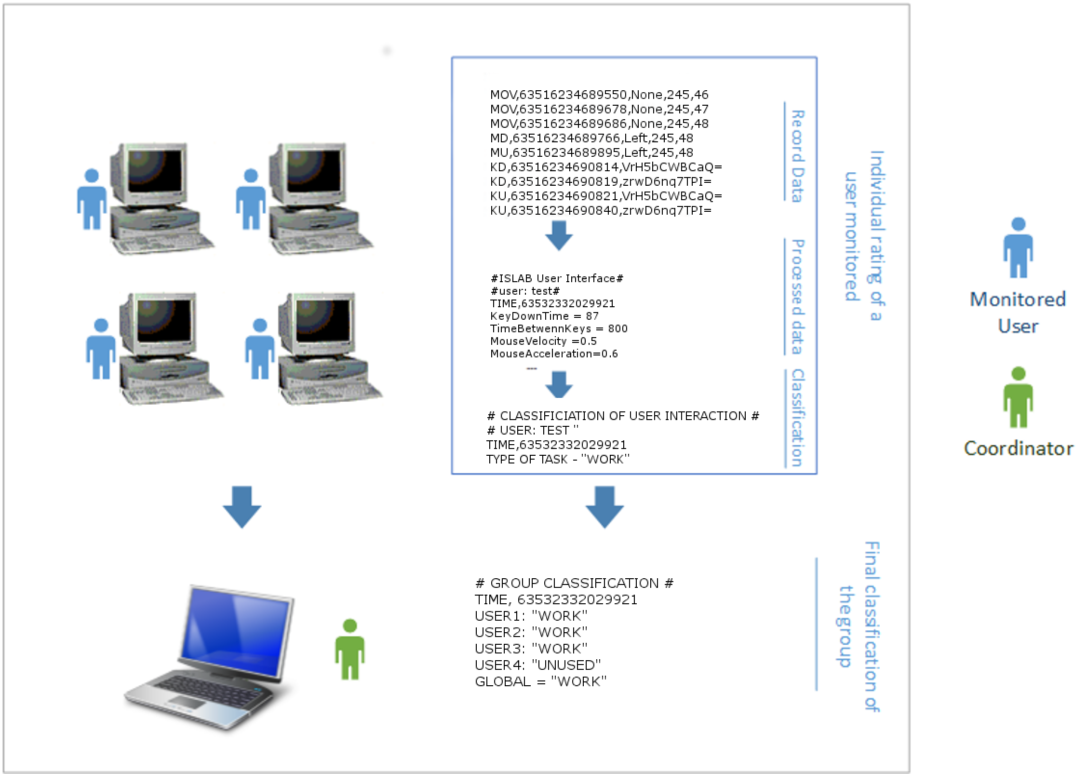
In order to facilitate its development and its deployment to other environments, this technological framework was developed as a Service-Oriented Architecture [13]. Specifically, we adopted a Service-Component Architecture (SCA): a group of OASIS specifications that has become an industry standard. It is intended for the development of applications based on SOA, which defines how computing entities interact to perform work for each other. Originally published in

November 2005, SCA is based on the notion that all the functions in an system should exist in the form of services that are combined into composites to address specific business requirements. In other words, it allows to build service-oriented applications as networks of service components. SCA is used for building service components, assemble components into applications, deploy to (distributed) runtime environments and reuse service components built from new or existing code using SOA principles.

From the several available implementations of SCA we have chosen JBoss SwitchYard since it is an open source solution in a relative mature state, and also enhances some of the SCA advantages. Specifically, Switchyard advocates transparency when running a service during its whole lifecycle. Important aspects such as connectivity, orchestration and routing do exist on SwitchYard in a modular format, which means one can deploy them in an independent way.

Services are used for all the distributed tasks, namely: the collection of data from multiple users simultaneously, the transmission of the collected data to the server, the persistence of the data in the database and the transformation of raw data into interaction features. Moreover, later access to the data is also ensured by web services. This makes the architecture highly modular and dynamic, allowing users to use all of its features or only reduced sets, as necessary.

In this architecture there are two main types of node. The so-called *Monitored User* represents any of the members of the group being monitored. The *Coordinator* represents the autonomous or human-guided decision-support system responsible for setting the objectives and managing the environment in terms of its user's performance. Figure 2 depicts the distribution of the roles and the flow of information from raw data to the classification of the state of the individuals.



```
MOV,63516234689550,None,245,46
MOV,63516234689678,None,245,47
MOV,63516234689686,None,245,48
MD,63516234689766,Left,245,48
MU,63516234689895,Left,245,48
KD,63516234690814,VrH5bCWBCaQ=
KD,63516234690819,zrwD6nq7TPI=
KU,63516234690821,VrH5bCWBCaQ=
KU,63516234690840,zrwD6nq7TPI=
```

```
#ISLAB User Interface#
#user: test#
TIME,63532332029921
KeyDownTime = 87
TimeBetwennKeys = 800
MouseVelocity =0.5
MouseAcceleration=0.6
...
```

```
# CLASSIFICIATION OF USER INTERACTION #
# USER: TEST "
TIME,63532332029921
TYPE OF TASK - "WORK"
```

```
# GROUP CLASSIFICATION #
TIME, 63532332029921
USER1: "WORK"
USER2: "WORK"
USER3: "WORK"
USER4: "UNUSED"
GLOBAL = "WORK"
```

Record Data
Processed data
Classification
Individual rating of a user monitored
Final classification of the group
Monitored User
Coordinator

Figure 2. Flow of information from raw data to the classification of the state of the individuals. The coordinator is depicted in green while users are depicted in blue.

Each monitored user provides raw data about their interaction patterns to the coordinator. This raw data essentially describes all the events related to the use of the mouse and keyboard, as described further ahead. It is processed by another web service, available at the coordinator, that transforms the raw data into processed data. Besides transforming data, outlier values are also filtered in this step. These outliers include, asides from the values pointed out statistically, repetition of key presses (e.g. continuously pressing the backspace key to delete blocks of text) and double clicks (which are inherently different from standard clicks, and are thus treated differently). Each two consecutive clicks that are less than 200 ms apart are considered a double click and are, consequently, seen as outliers.

includes the consecutive pressing of some keys , such as when the backspace key is pressed continuously to delete a set of characters.

This processed data constitute the features that quantify the performance of interaction of each user, which are also described further ahead. Moreover, this framework provides information about the current task being carried out by the individual and an estimate of attention. It is based on this data that the coordinator can take decisions regarding the management of the environment and of the individual's performance. All the generated information is also persisted permanently. This means that the framework gradually builds a profile of each user's behaviour. This allows not only to get to know each individual and the way they usually behave but also to detect significant behavioural changes that may be indicative of some issue.

The process described is depicted in Figure 3, which details the process of monitoring.

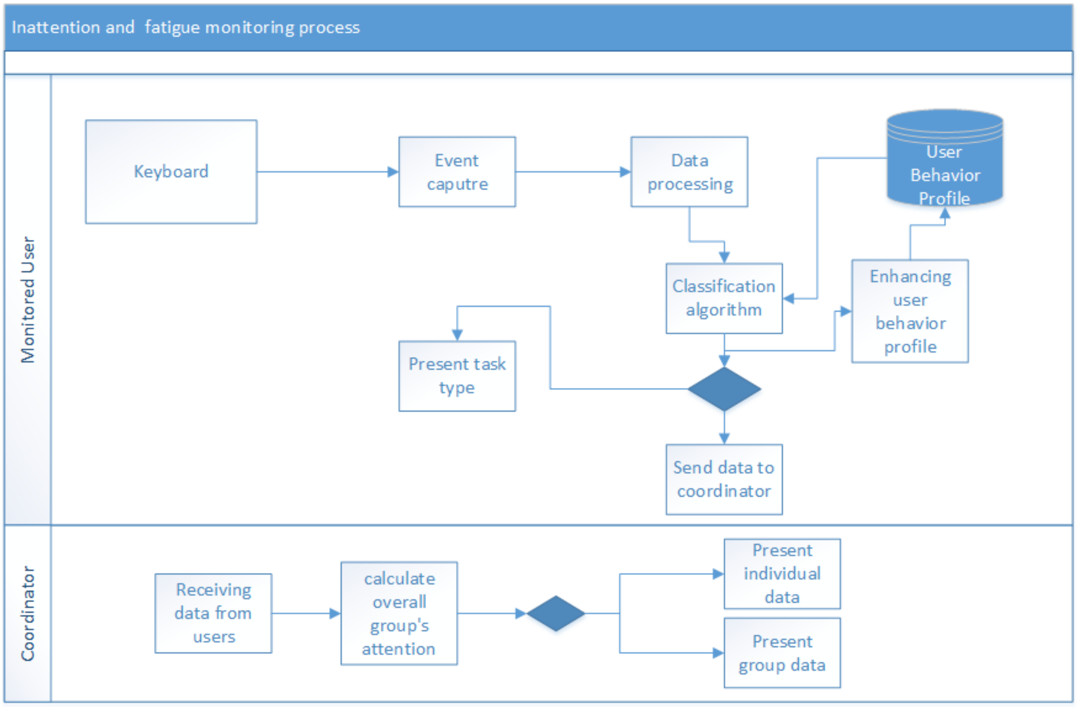


Figure 3. Workflow of information in the monitoring process.

2.1. *Data Acquisition*

As stated before, data is acquired from the monitored users that describe their interaction patterns with the computer while using the peripherals. This collection of data is completely transparent for the user, i.e., the application runs in the background and requires no explicit actions on the part of the user. The data acquisition module provides data in real-time about the following events:

- MOV, timestamp, posX, posY - an event describing the movement of the mouse, in a given time, to coordinates (posX, posY) in the screen;
- MOUSE_DOWN, timestamp, [Left|Right], posX, posY - this event describes the first half of a click (when the mouse button is pressed down), in a given time. It also describes which of the buttons was pressed (left or right) and the position of the mouse in that instant;

- MOUSE_UP, timestamp, [Left|Right], posX, posY - an event similar to the previous one but describing the second part of the click, when the mouse button is released;
- MOUSE_WHEEL, timestamp, dif - this event describes a mouse wheel scroll of amount dif, in a given time;
- KEY_DOWN, timestamp, key - identifies a given key from the keyboard being pressed down, at a given time;
- KEY_UP, timestamp, key - describes the release of a given key from the keyboard, in a given time;

Listing 1 depicts a brief log that starts with some mouse movement, contains a click (with a little drag) and ends with some more movement.

Listing 1: Sample of several events of interaction

```
MOV, 635296941683402953, 451,  195
MOV, 635296941684123025, 451,  197
MD,  635296941684443057, Left , 451, 199
MOV, 635296941685273140, 452,  200
MU,  635296941685283141, Left , 452, 200
MOV, 635296941685723185, 452,  203
MOV, 635296941685803193, 454,  205
```

Table I details the size of the dataset, for each feature and each type of task. Both the features considered and the types of task used are detailed further ahead in this paper.

Table I. Size of the dataset collected (number of events registered), for each feature and each type of task.

Feature	Chat	Work	Net	Others
Key Down Time	96131	14903	100070	34640
Time Between Keys	118047	17531	111990	50699
Mouse Velocity	643335	150425	1054447	362593
Mouse Acceleration	879561	204715	1431969	474461
Time Between Clicks	29253	3880	42630	8491
Writing Velocity	17909	3413	21700	7667
Double Click Duration	1332	84	2103	284
Average Excess Of Distance	21897	3594	32071	8189
Average Distance of the Mouse to the S. L.	21897	3594	32071	8189
Distance of the Mouse to the S.L.	21891	3591	32074	8183
Signed Sum of Angles	21892	3591	32070	8183
Absolute Sum of Angles	21897	3594	32071	8189
Distance Between Clicks	29290	3931	42692	8534

2.2. *Feature Extraction*

The six different types of event described in Section 2.1 constitute the lower level of data considered in this architecture. It is then transformed, by a Web Service, in a set of features that describe the performance of the interaction of the individual with the computer. The following features are extracted from the collected data:

- Key Down Time - the timespan between two consecutive KEY_DOWN and KEY_UP events, i.e., for how long was a given key pressed.
- Time Between Keys - the timespan between two consecutive KEY_UP and KEY_DOWN events, i.e., how long did the individual took to press another key.

- Mouse Velocity - The distance travelled by the mouse (in pixels) over the time (in milliseconds). The velocity is computed for each interval defined by two consecutive MOUSE_UP and MOUSE_DOWN events. Let us assume two consecutive MOUSE_UP and MOUSE_DOWN events, mup and mdo, respectively in the coordinates $(x1, y1)$ and $(x2, y2)$, that took place respectively in the instants $time_1$ and $time_2$. Let us also assume two vectors $posx$ and $posy$, of size n, holding the coordinates of the consecutive MOUSE_MOV events between mup and mdo. The velocity between the two clicks is given by $r\_dist/(time_2 - time_1)$, in which r_dist represents the distance travelled by the mouse and is given by equation 1.

$$r\_dist = \sum_{i=0}^{n-1} \sqrt{(posx_{i+1} - posx_i)^2 + (posy_{i+1} - posy_i)^2} \tag{1}$$

- Acceleration - The velocity of the mouse (in pixels/milliseconds) over the time (in milliseconds). A value of acceleration is computed for each interval defined by two consecutive MOUSE_UP and MOUSE_DOWN events, using the intervals and data computed for the Velocity.
- Time Between Clicks - the timespan between two consecutive MOUSE_UP and MOUSE_DOWN events, i.e., how long did it took the individual to perform another click.
- Writing Velocity - number of keys pressed per second.
- Double Click Duration - the timespan between two consecutive MOUSE_UP events, whenever this timespan is inferior to 200 milliseconds. Wider timespans are not considered double clicks.
- Average Excess of Distance - this feature measures the average excess of distance that the mouse travelled between each two consecutive MOUSE_UP and MOUSE_DOWN events. Let us assume two consecutive MOUSE_UP and MOUSE_DOWN events, mup and mdo, respectively in the coordinates $(x1, y1)$ and $(x2, y2)$. To compute this feature, first it is measured the distance in straight line between the coordinates of mup and mdo as $s\_dist = \sqrt{(x2 - x1)^2 + (y2 - y1)^2}$. Then, it is measured the distance actually travelled by the mouse by summing the distance between each two consecutive MOUSE_MV events. Let us assume two vectors $posx$ and $posy$, of size n, holding the coordinates of the consecutive MOUSE_MV events between mup and mdo. The distance actually travelled by the mouse, $real\_dist$ is given by equation 1. The average excess of distance between the two consecutive clicks is thus given by r_dist/s_dist.
- Average Distance of the Mouse to the Straight Line - in a few words, this feature measures the average distance of the mouse to the straight line defined between two consecutive clicks. Let us assume two consecutive MOUSE_UP and MOUSE_DOWN events, mup and mdo, respectively in the coordinates $(x1, y1)$ and $(x2, y2)$. Let us also assume two vectors $posx$ and $posy$, of size n, holding the coordinates of the consecutive MOUSE_MOV events between mup and mdo. The sum of the distances between each position and the straight line defined by the points $(x1, y1)$ and $(x2, y2)$ is given by 2, in which $ptLineDist$ returns the distance between the specified point and the closest point on the infinitely-extended line defined by $(x1, y1)$ and $(x2, y2)$. The average distance of the mouse to the straight line defined by two consecutive clicks is this given by s_dists/n.

$$s\_dists = \sum_{i=0}^{n-1} ptLineDist(posx_i, posy_i) \tag{2}$$

- Distance of the Mouse to the Straight Line - this feature is similar to the previous one in the sense that it will compute the s_dists between two consecutive MOUSE_UP and MOUSE_DOWN events, mup and mdo, according to equation 2. However, it returns this sum rather than the average value during the path.
- Signed Sum of Angles - with this feature the aim is to determine if the movement of the mouse tends to "turn" more to the right or to the left. Let us assume three

consecutive MOUSE_MOVE events, $mov1, mov2$ and $mov3$, respectively in the coordinates $(x1, y1)$, $(x2, y2)$ and $(x3, y3)$. The angle α between the first line (defined by $(x1, y1)$ and $(x2, y2)$) and the second line (defined by $(x2, y2)$ and $(x3, y3)$) is given by $degree(x1, y1, x2, y2, x3, y3) = \tan(y3 - y2, x3 - x2) - \tan(y2 - y1, x2 - x1)$. Let us now assume two consecutive MOUSE_UP and MOUSE_DOWN events, mup and mdo. Let us also assume two vectors $posx$ and $posy$, of size n, holding the coordinates of the consecutive MOUSE_MOV events between mup and mdo. The signed sum of angles between these two clicks is given by equation 3.

$$s\_angle = \sum_{i=0}^{n-2} degree(posx_i, posy_i, posx_{i+1}, posy_{i+1}, posx_{i+2}, posy_{i+2}) \tag{3}$$

- Absolute Sum of Angles - this feature is very similar to the previous one. However, it seeks to find only how much the mouse "turned", independently of the direction to which it turned. In that sense, the only difference is the use of the absolute of the value returned by function $degree(x1, y1, x2, y2, x3, y3)$, as depicted in equation 4.
  UNITS - degrees

$$s\_angle = \sum_{i=0}^{n-2} \mid degree(posx_i, posy_i, posx_{i+1}, posy_{i+1}, posx_{i+2}, posy_{i+2}) \tag{4}$$

- Distance between clicks - represents the total distance travelled by the mouse between two consecutive clicks, i.e., between each two consecutive MOUSE_UP and MOUSE_DOWN events. Let us assume two consecutive MOUSE_UP and MOUSE_DOWN events, mup and mdo, respectively in the coordinates $(x1, y1)$ and $(x2, y2)$. Let us also assume two vectors $posx$ and $posy$, of size n, holding the coordinates of the consecutive MOUSE_MOV events between mup and mdo. The total distance traveled by the mouse is given by equation 1.
  Although a total of 13 interaction features are supported by this architecture, not all of they may prove relevant. In fact, the relevance of each feature depends on the nature of the study to be carried out. For this reason, different feature selection processes may be carried out, according to each study's objectives. In this specific case, and as described in the following section, we built on results from previous studies to select the most appropriate features [14, 15].

2.3. *Feature Selection*

The set of features described in the previous section characterize, to a good extent, several aspects of the performance of the individual's interaction with the computer. However, this does not mean that they are all affected equally or that they are all affected at all when performance drops. To this end, the best features must be selected according to the objective of the environment and according to their significance.

To perform this selection we have been carrying out research studies, in real environments, to collect data that may allow us to determine how people are affected by the variables being studied. So far, we have studied the effect of stress [15] and fatigue [14] on the performance of interaction. From these studies we concluded that the most significantly affected performance features are the Keydown Time, Errors per Key Pressed, Time Between Keys, Mouse Velocity, Mouse Acceleration and Time Between Clicks. In all these features 90% or more of the participants show statistically significant differences when comparing the baseline with the state under the influence of the studied variable (e.g. stress or fatigue). For this reason, these are the features that are selected to monitor the individuals' performance. The remaining features are nonetheless detailed in this paper as they are part of the group that can be extracted from this architecture and they may prove interesting in the future for studies of other nature.

DOI: 10.1002/cpe

3. MONITORING PERFORMANCE

So far this paper has described the approach and the features used to acquire contextual information about the user and its performance. In this section we describe one of the ways in which this information can be used. Specifically, we show how it can be used to estimate the attention of the user to the task at hand, one of the most important determinants of task performance.

We look at how the user distributes the time devoted to each running application, and at which of these applications are related with the task at hand. We thus complement and improve the previously developed fatigue monitoring approach (task-independent, based solely on the performance of the interaction with the computer) with a measure of attention, which can be used as a reliable indicator of fatigue [16].

3.1. Methodology

One of the aims of this work is that performance can be monitored in a non-intrusive and non-invasive way, so that the monitored individuals can carry out their work without any interference. Having this in mind, and with the objective of studying the problem of performance monitoring, an experimental study was set up involving 20 men and 7 women, all students from the University of Minho. These participants were briefed about the installation and use of the application, but details of the nature of the study were not provided. They were also instructed to simply perform their work routines as usual as the application needed no special interaction. The study lasted approximately one month, a period during which data about the previously mentioned features were collected.

After the process of data collection, the events were organized into four categories, according to the application that the user was interacting with at the time of the event. These categories were defined according to the most frequently used applications and are defined as follows:

1. *Chat* - this category includes inputs from applications such as Skype, Hangouts or Facebook Mensseger;
2. *Browsing* - this category is populated by events generated from the use of web browsers such as Internet Explorer, Firefox or Google Chrome;
3. *Work* - this category is composed of events generated from the interaction with applications such as the Eclipse IDE, Microsoft Office Suite, TexMaker, Adobe Reader, Evernote or Netbeans;
4. *Games* - this final category includes events generated while playing games.

In a preliminary analysis of the data, we concluded that there are indeed different interaction patterns depending on the type of task being performed. To conclude this, we looked at the distributions of the data collected for each category of application and analysed the statistical significance of their differences. To this aim, the following approach was implemented.

To assess normality, data for each feature were divided according to the four existing categories. The Pearson's chi-squared test was used to determine that most of the distributions of the data were not normal, as detailed in Table II. This fact resulted in the subsequent use of the Kruskal-Wallis test for the analysis of the data carried out. This non-parametric statistical test is used to test if two samples are independent. If they are, we are in the presence of two distinct behaviours. The aim is thus to determine if there are indeed differences in the distributions of interaction data for the same features, when the data was collected while interacting with different applications.

This test thus calls for the definition of the so-called null hypothesis: H_0 : all samples have identical distribution functions, against the alternative hypothesis that at least two samples have different distribution functions.

The selected test returns, for each set of samples compared, a so-called p-value. A small p-value thus suggests that it is unlikely that H_0 is true. Thus, for every Kruskal-Wallis test whose $p - value < \alpha$, the difference is considered to be statistically significant. This means that the null hypothesis is rejected. A standard value for α generally accepted by research is 0.05. For this reason it is also used in this work.

Table II. Results of the Pearson's chi-squared test for each feature and each type of task.

Feature	Chat	Work	Net	Others
Key Down Time	< 2.2e-16	< 2.2e-16	< 2.2e-16	< 2.2e-16
Time Between Keys	< 2.2e-16	< 2.2e-16	< 2.2e-16	< 2.2e-16
Mouse Velocity	< 2.2e-16	< 2.2e-16	< 2.2e-16	< 2.2e-16
Mouse Acceleration	< 2.2e-16	< 2.2e-16	< 2.2e-16	< 2.2e-16
Time Between Clicks	< 2.2e-16	< 2.2e-16	< 2.2e-16	< 2.2e-16
Writing Velocity	5.032e-08	0.002768	2.794e-10	2.519e-07
Double Click Duration	1.18e-07	0.9624	8.662e-11	0.001776
Average Excess Of Distance	< 2.2e-16	< 2.2e-16	< 2.2e-16	< 2.2e-16
Average Distance of the Mouse to the S. L.	< 2.2e-16	< 2.2e-16	< 2.2e-16	< 2.2e-16
Distance of the Mouse to the S.L.	< 2.2e-16	< 2.2e-16	< 2.2e-16	< 2.2e-16
Signed Sum of Angles	2.023e-14	0.0004798	< 2.2e-16	1.766e-08
Absolute Sum of Angles	< 2.2e-16	7.414e-14	< 2.2e-16	< 2.2e-16
Distance Between Clicks	< 2.2e-16	2.989e-12	< 2.2e-16	< 2.2e-16

An example of the differences were are interested in finding is clearly depicted in Figure 4, for two different features: writing velocity and mouse velocity. The distributions of the data depicted in the Box-and-Whisker Plots show, for example, that the participants' writing speed is the fastest when interacting with chat applications, while the mouse speed reaches its maximum when participants are playing games.

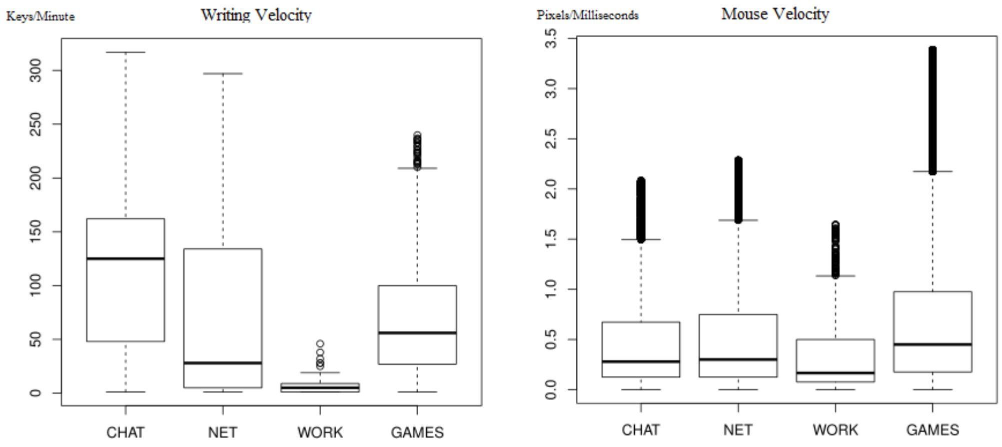


Figure 4. Differences in the use of mouse and keyboard between the different categories of applications.

The results of the significant test when comparing the distributions of each feature for each of the four types of task is detailed in Table III

3.2. Validation and Results

The methodology described in the previous section was used to show the existence of different behaviours in the use of the mouse and keyboard when interacting with different tasks. The existence of these statistically significant differences supported the training of a classifier able to determine the type of task being carried out by the user, taking as input his interaction patterns. To achieve this goal, the KNN (k-Nearest Neighbour) algorithm was used, which we have been using in the past with satisfying results for similar studies.

This algorithm is based on closest training examples in the feature space. The classifier was trained with the collected data. Several tests with different numbers of neighbours (K) and with

Table III. Results of the Kruskal-Wallis test for each feature, when comparing its distributions in each task.

Feature	***p*-value**
Key Down Time	2.339e-05
Time Between Keys	<2.2e-16
Mouse Velocity	<2.2e-16
Mouse Acceleration	<2.2e-16
Time Between Clicks	2.564e-05
Writing Velocity	7.046e-11
Double Click Duration	1.499e-15
Average Excess Of Distance	2.292e-14
Average Distance of the Mouse to the Straight Line	0.0003032
Distance of the Mouse to the Straight Line	0.05312
Signed Sum of Angles	1.634e-07
Absolute Sum of Angles	<2.2e-16
Distance Between Clicks	4.372e-12

different heuristics to the distance between neighbouring (rectangular, triangular, epanachicov, gaussian, rank, optimal) were performed. With a maximum of 50 K, the solution having a lower mean squared error (MSE), was found with K = 10 and using the rectangular kernel. This section describes the performance of the classifier and its validation, as well as the validation of the technological framework itself.

A case study was set up that included the previously mentioned roles of Monitored Users and Coordinator. Besides the test and validation of the developed system, we were also looking for effects of distraction and fatigue. The case study was thus implemented during a course on programming. The Coordinator is a teacher that is responsible for teaching a programming language. The Monitored Users are students, who interact with computers in the classroom during the duration of the class. They need to interact with the computers to perform the tasks they are assigned since the class is mostly practical. There are however no restrictions on the use of other applications, i.e., they are free to use the computers as they wish. From the observed behaviours, students frequently use the computers for tasks such as taking notes, visiting social networking sites, searching for additional information or examples and for instant messaging. The selected course had 18 students enrolled, 13 men and 5 women, aged 18 to 25 (average age is 19).

The course takes place weekly, during a semester. Each weekly session has a duration of three hours and is generally structured in two parts, a theoretical one and a practical one. At the beginning of the session the teacher introduces theoretical concepts and code examples. The rest of the session is spent solving practical exercises on the computer. For this end, the students use a specific IDE (Integrated Development Environment). Included in this practical part of the session are one or more exercises that the students must solve and submit for evaluation. These exercises aim to determine how well the topics of each session were learned by the students.

At the end of each session, besides handing in the solved exercises, the students also fill in a questionnaire. This questionnaire aims to perform a self-evaluation regarding their attention levels, i.e., students rate how distracted they were. To this end, they rate their level of attention with a value between 0 (highly distracted) and 5 (highly attentive). The aim of combining all these factors is to find a relationship between distraction, fatigue, interaction performance and academic performance. In this case, and given the domain of the case study, the classifier only classified interaction as being or not work-related.

Table IV depicts the results achieved during this phase. We concluded that the participants who were less attentive during the sessions also performed worst in the evaluation exercises, and vice versa. Participants 4, 6, 8, 14, 15 and 17 are examples of this. The answers in the questionnaires also point out that these students were aware of their lack of attention during the session.

When studying the relationship between the variables, the following positive correlations were found, measured with the Spearman's rank correlation coefficient: correlation between

DOI: 10.1002/cpe

Table IV. Overall results of all participants, depicting the values of the monitoring, evaluation and values of the subjective level of attention of each participant.

User	Minutes in work tasks	Minutes others tasks	Minutes unused	% Work	% others	% Subjective attention	% Final evaluation
1	96	36	48	73%	27%	5	90%
2	60	36	84	63%	38%	4	100%
3	72	12	96	86%	14%	4	100%
4	54	48	78	53%	47%	2	50%
5	90	0	90	100%	0%	3	50%
6	54	90	36	38%	62%	3	30%
7	90	12	78	88%	12%	4	100%
8	60	42	78	59%	41%	4	70%
9	96	42	42	70%	30%	4	100%
10	108	0	72	100%	0%	5	80%
11	84	24	72	78%	22%	4	90%
12	96	24	60	80%	20%	5	100%
13	84	24	72	78%	22%	3	70%
14	48	42	90	53%	47%	3	50%
15	54	36	90	60%	40%	3	0%
16	60	30	90	67%	33%	4	90%
17	48	54	78	47%	53%	3	20%
18	114	30	36	79%	21%	5	80%
Mean	76	32	71	71%	29%	4	70%

the percentage of time spent interacting with work-related applications and final score (0.54), correlation between the percentage of time spent interacting with work-related applications and subjective evaluation of attention (0.51) and correlation between subjective evaluation of attention and final score (0.68).

As previously stated, there was also an interest on validating the data collection tool and the classification algorithm. To this end, while the interaction with the computer took place, the logger application was on the one hand recording the application the student was interacting with. On the other hand, and at the same time, the classification algorithm was classifying the interaction patterns of each students as related to work or not related to work. Although the students were suggested to use the Dev-C++ IDE, some cases were found in which alternative editors were used, such as Microsoft Office, TextMaker or Evernote. We thus look at the application being used and the type of application provided as output by the classifier in each moment in order to validate its efficacy.

As Table V depicts, the accuracy of the classifier is very satisfying. Indeed, 96% of the students' interaction with work-related applications were also classified as work-related. On the other hand, 92% of the interactions that were not work-related were correctly classified. Supported by these results, we consider the approach validated.

Table V. Results of the classifications algorithm.

Task type	Instances	% Correctly classified	% Incorrectly classified
Work tasks	228	96%	4%
Others	97	92%	8%
Unused	215	100%	0%

The data collected as well as the information compiled by this distributed system is easily accessible to the group coordinator and the users, in real-time through different web interfaces. While a worker can only see his own information, a coordinator can see the information about all the workers. Figure 5 shows one of such interfaces, showing the computed fatigue level and the classification of the the tasks throughout time. Other information accessible through these interfaces

DOI: 10.1002/cpe

includes statistic measures and performance indicators. These interfaces also issue notifications when fatigue or performance indicators exceed some pre-determined value.

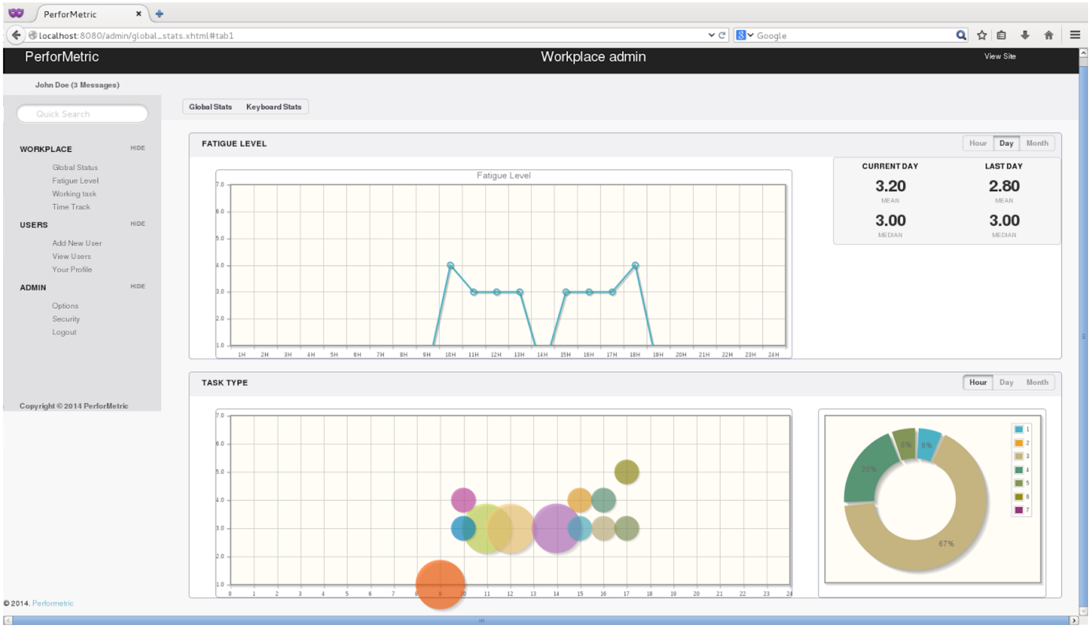


Figure 5. Interface of the Performance Monitoring Web Service.

4. IMPROVING PERFORMANCE

So far, this paper has addressed the analysis of a computer user's performance in real-time and how such knowledge can be used, namely to estimate attention or mental fatigue. In this section we explore the possibility of positively influencing this performance. Given all the positive effects of music in the human being, namely in terms of performance, we consider its use to improve performance in human-computer interaction. To appropriately study these effects, an experimental study was carried out. This study aimed to:

- Determine if musical selection (the independent variable) has an effect on the performance of the interaction patterns (the dependent variable) of the users with the computer;
- Determine if different types of music have different effects on the variable;
- Study and quantify the effects of different types of music on the variable;
- Determine if users are conscious of the effects measured or, at least, of some effect at some level;

The independent variable in this study was thus musical selection, while the dependant variable was the performance of the participants, measured in terms of the features detailed in section 2. Five different types of music were used, first classified and put forward by [17] in the form of five so-called mood clusters. Each cluster contained music classified as follows:

- Cluster 1: passionate, rousing, confident, boisterous, rowdy
- Cluster 2: rollicking, cheerful, fun, sweet, amiable/good natured
- Cluster 3: literate, poignant, wistful, bittersweet, autumnal, brooding
- Cluster 4: humorous, silly, campy, quirky, whimsical, witty, wry
- Cluster 5: aggressive, fiery, tense/anxious, intense, volatile, visceral

In the past we have studied how performance is negatively influenced by fatigue throughout the day. In the present study we aim to determine if music may have a positive effect on the performance of the individual by improving it or by delaying its decrease during the day or during particularly stressing periods. The verification of this possibility will support the development of a music recommendation service aimed at improving musical selection with particular objectives, such as improving individual or group performance, satisfaction with music selection or motivation to work. This is expected to consequently improve indicators such as work satisfaction, productivity and quality of the working environment.

4.1. Methodology

This experimental study took place in the Intelligent Systems Lab of the University of Minho. In this lab, numerous students and researchers spend their day working with a computer and are allowed to listen to music using headphones. Twelve participants were selected to take place in this study, aged between 20 and 28, with an average of 24.3. The recording of performance indicators was carried out in the background using the previously described technological framework.

The actual phase of data collection was preceded by the filling of a questionnaire in which participants provided information to characterize their profile, namely in terms of demographics and musical preferences. Among other tasks, in these questionnaires participants must rate some sample musics from the different clusters according to their perceived level of valence and activation.

The actual study was structured as follows. Each participant took part in five different moments of data collection, each in a different day. In each one, participants listened to music from one of the different clusters while carrying out their regular work-related tasks. Each daily moment of data collection had a minimum duration of three hours.

At the end of each data collection moment, each participant answered another questionnaire, aimed at determining the subjective feelings of the participant regarding the music and its perceived influence on work performance, i.e. is the participant aware of any effect due to music? This questionnaire also aimed to determine if music truly induced the expected mood in the participant, i.e., if the clusters are perceived according to the classification put forward by [17]. The data collected, from both the questionnaires and the performance monitoring software, was analysed using statistical software and the results are described in the following subsection.

Essentially, we aimed at both a subjective and objective analysis. The subjective analysis relied on the participant's feedback, namely concerning their performance self-evaluation during the study. The objective analysis consisted on the study of the interaction patterns of the participants during the study, using the previously mentioned framework.

4.2. Validation and Results

One of the objectives of this work was to determine if listening to the different clusters would be experienced by the participants as expected, i.e., if the clusters deemed to be calm would be experienced so by the participants. As Figure 6 shows, this does happen. Cluster 3, which contains music that can be described as autumnal, brooding or literate, is the one that relaxes participants the most. Cluster 5, on the other hand, containing music that can be classified as rather aggressive, fiery or tense, is the one that relaxes them the least.

There is, however, not necessarily a direct relationship between how relaxed one is and one's performance. Indeed, this work shows that this relationship is more complex than it may seem at first sight. In this study we found that our performance does not depend only on musical selection but also on the musical profile of the individual.

Figure 7 details this issue further. Figure 7 a) details the variation of performance as self-evaluated by the participants, for each cluster. Each participant provided, in the questionnaire, a self-evaluation of their performance at the beginning and at the end of the day, using a 9-point Likert scale (1 - not active at all; 9 - very active). From the analysis of the participant's feedback, Cluster 4 is the one that contributes more to the improvement of the felt performance of the participants.

Figure 7 b), on the other hand, details the improvement in performance as measured by the proposed framework. In this case, performance variation is quantified as the average number of

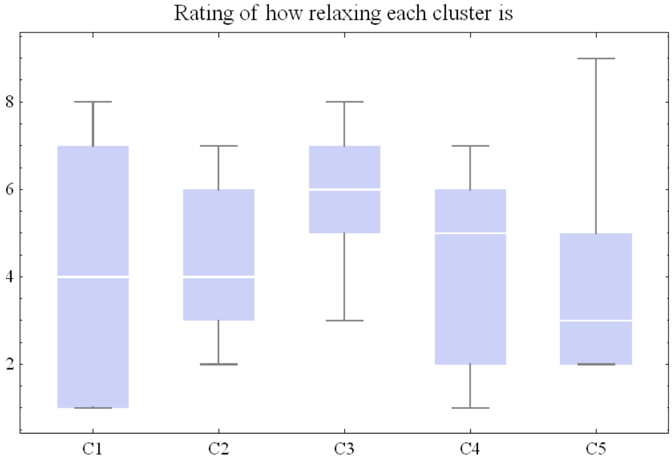


Figure 6. Distribution of the gradings, by the users, of how relaxing each Cluster is (1 - no relaxing at all, 9 - highly relaxing).

features in which there is a positive variation of performance, providing an overall view of the participant's performance. Once again, the cluster in which the positive variation of performance is more marked is Cluster 4.

Indeed, Cluster 4 contains music that can be classified as humorous and silly. It thus positively influences aspects such as mood and motivation. One of the most interesting results of this work is, indeed, that the effects of this tye of music stand out from the rest, when measured both objectively and subjectively. While Cluster 2 can be considered similar, it activates people less. Cluster 5, on the other hand, contains the heaviest music. This, although generally activating, when listened for long periods will wear individuals out, especially those who do not appreciate this kind of music. It thus ends having a negative effect.

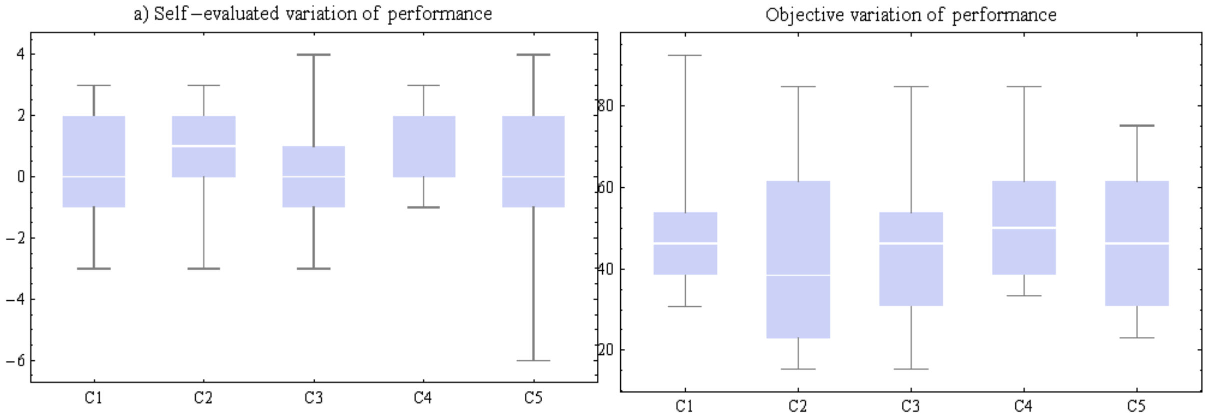


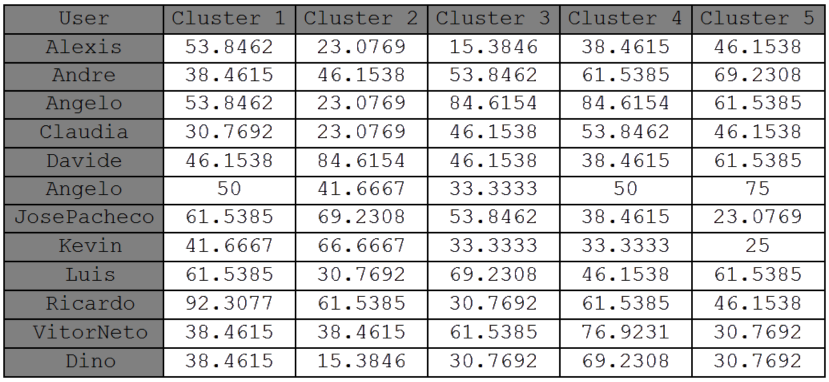
Figure 7. a) variation of the performance as felt by the participants. b) variation of the performance measured by the proposed framework.

These differences, although visible and significant when considering all the participants, can result still more interesting when considering individual participants and their specific musical profile. Indeed, we observe that people that are more prone to listen to heavy music on a daily basis are more positively affected and see their performance improved by a longer timespan when listening to clusters containing this kind of music. They are activated by this music in a positive manner and work more efficiently. These are also people that have a higher baseline activation, i.e., they are naturally more "stressed". People that are naturally calmer, on the other hand, find this music annoying and sometimes hurtful to hear and are unable to concentrate, which consequently

DOI: 10.1002/cpe

and evidently affects their performance. These individuals thus exhibit increased work performance with more calm music.

A good example of this are participants "Davide" and "Vitor Neto" (Figure 8). Davide, who can be described as someone who regularly listens to heavy music, achieves the best performance results with Clusters 2 and 5. Vitor, on the other hand, a calmer person by nature, demonstrates better performance in Clusters 3 and 4.



User	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
Alexis	53.8462	23.0769	15.3846	38.4615	46.1538
Andre	38.4615	46.1538	53.8462	61.5385	69.2308
Angelo	53.8462	23.0769	84.6154	84.6154	61.5385
Claudia	30.7692	23.0769	46.1538	53.8462	46.1538
Davide	46.1538	84.6154	46.1538	38.4615	61.5385
Angelo	50	41.6667	33.3333	50	75
JosePacheco	61.5385	69.2308	53.8462	38.4615	23.0769
Kevin	41.6667	66.6667	33.3333	33.3333	25
Luis	61.5385	30.7692	69.2308	46.1538	61.5385
Ricardo	92.3077	61.5385	30.7692	61.5385	46.1538
VitorNeto	38.4615	38.4615	61.5385	76.9231	30.7692
Dino	38.4615	15.3846	30.7692	69.2308	30.7692

Figure 8. Percentage of features that improved over the day, for each user and each cluster.

Indeed, the problem of determining the most appropriate style of music for an individual is a complex one and, as these results show, several variables must be taken into account. Namely, and besides the type of music, the musical profile of the individual. Moreover, the objective of the individual at the time (e.g. does he need to complete a task quickly? Does he prefer to work calmly?) as well as the timespan (e.g. we have the tendency to grow tired of a type of music if listen to it for prolonged periods of time) should also be included in future work.

4.3. Music Recommendation Service

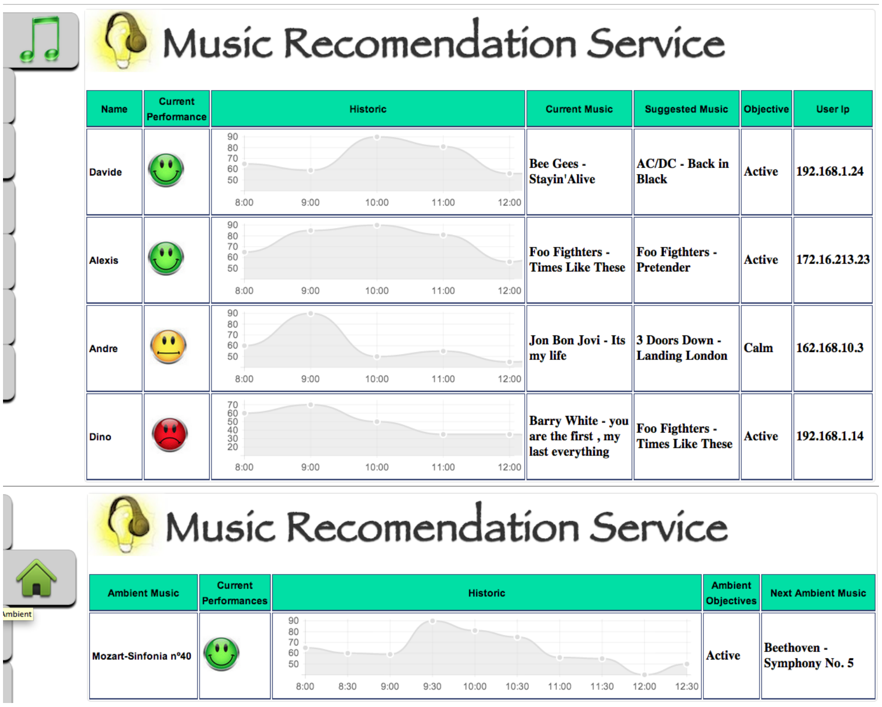
The results described in the previous section supported the development of a prototype for a music recommendation service (Figure 9). This prototype aims to select the most appropriate style of music at a given time, without intervention from the users, i.e., in an autonomous way. It can do so at two different levels: user-level (in which the prototype optimizes the musical selection for a given user) and group-level (in which it does so for a group of people).

From the profiles defined through the questionnaires, we know how each individual feels about each type of music: how much they like it, how relaxing/activating they find it or to which extent they preferred to have carried out their activities without listening to this particular type of music.

Users also provide the prototype with their current objective, using a minimalist interface. At any moment, the objective can be to relax (e.g. when the user is involved in a creative task), to activate (e.g. when the user needs to complete a given task quickly) or, aside from performance issues, to listen to their favourite musics. Besides assigning an objective, the user also assigns a weight to *Performance* and *Musical Preference*. That is, placing a more significant weight on *Performance* will result in the selection of music that contributes more to the user's activation, disregarding preferences. Otherwise, music that is more to the taste of the user will be selected, despite less effective results being expected in what concerns interaction performance.

These variables, as well as the weights assigned by the user, are used by optimization functions to attribute a score to each Cluster, at any moment, normalized in the interval [0,1]. To prevent people from getting tired of constantly listening to the same type of music, musics are then selected from all the five Clusters in a frequency that is proportional to these scores (e.g. if Cluster 1 has twice the score of Cluster 2, musics form Cluster 1 will be selected with twice the frequency).

A similar process is used for selecting ambient music. However, in this case, it is the coordinator that determines the objective of the environment. If there is a scheduled brainstorming session, the coordinator may decide to play activating music in order to stimulate participants. On the other hand, if the end of the day is nearing, the coordinator may decide to put a more relaxing music as individuals are already tired and activating music may have negative effects, as our results show.



Music Recomendation Service

Name	Current Performance	Historic	Current Music	Suggested Music	Objective	User Ip
Davide	🙂		Bee Gees - Stayin'Alive	AC/DC - Back in Black	Active	192.168.1.24
Alexis	🙂		Foo Figthters - Times Like These	Foo Figthters - Pretender	Active	172.16.213.23
Andre	😐		Jon Bon Jovi - Its my life	3 Doors Down - Landing London	Calm	162.168.10.3
Dino	☹		Barry White - you are the first , my last everything	Foo Figthters - Times Like These	Active	192.168.1.14

Music Recomendation Service

Ambient Music	Current Performances	Historic	Ambient Objectives	Next Ambient Music
Mozart-Sinfonia nº40	🙂		Active	Beethoven - Symphony No. 5

Figure 9. Detail of the interfaces of the Music Recommendation Service for individual users (upper image) and for the group of users (lower image).

5. CONCLUSIONS

Tracking performance in the workplace is nowadays a very interesting and challenging topic. It is interesting in the sense that employers want to track worker's performance so as to improve the organizations' performance. It is challenging in the sense that workers often oppose such initiatives arguing that they are invasive, time-consuming or constitute an invasion of their privacy.

In this paper we presented an innovative approach to the problem. Its main distinguish aspect is that it is not based on productivity indicators, as most existing approaches. It looks instead at performance indicators, specifically at the performance of the interaction with the computer. It does so in a non-intrusive and transparent way, meaning that workers do not need to carry out any specific task apart from the regular carrying out of their work.

The use of performance indicators has additional advantages. Namely, it allows to estimate task performance or attention, as described in this paper, and to estimate mental fatigue or stress, as already demonstrated in previous work. Finally, using the same indicators, the proposed approach also allows to determine the type of task the user is carrying out. It allows to do so in real-time, in an efficient way, as it uses models that are trained beforehand.

To implement this approach we developed a service-based distributed architecture that allows for data collection and actuation in real time. It allows for a number of arbitrary nodes to be connected at any given time. Coordinator nodes can be set up in specific environments, in the same physical location of the Monitored Users, or they can be elsewhere. This definitely facilitates the collection of data in real environments to conduct research studies, as we did. In a later stage, it will also simplify and facilitate the implementation of such approaches in real-life scenarios.

With this work we have demonstrated not only that it is possible to acquire performance indicators in real-time and in a non-intrusive way but also that these indicators can be used for a wide range of purposes. Moreover, we have also shown that worker's performance can be influenced, namely through the use of music. This opens new possibilities in terms of work environment management. It is possible to use the proposed architecture to perform individualized musical selection according to the particular state of each worker (e.g. a performance drop due to fatigue can be countered with more intense music, if the worker appreciates such type of music). On the other hand, music playing openly in the environment can also be selected according to specific objectives (e.g. playing activating music during a group brainstorming session).

In the overall, this kind of approaches will facilitate and improve performance monitoring initiatives, especially in the workplace. They will facilitate it by using non-invasive methods, that require no specific actions or inputs. They will improve it by providing real-time and continuous information about the state of the individuals and, above all, by allowing the environment coordinator to influence the performance of workers. From the point of view of management this will, undoubtedly, improve the quality of information available and significantly improve decision-making processes.

ACKNOWLEDGEMENT

This work is part-funded by ERDF - European Regional Development Fund through the COMPETE Programme (operational programme for competitiveness) and by National Funds through the FCT - Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) within projects FCOMP-01-0124-FEDER-028980 (PTDC/EEI-SII/1386/2012) and project PEst-OE/EEI/UI0752/2014.

REFERENCES

1. Aiello JR, Kolb KJ, Sauter SL (Ed), Murphy LR (Ed). 1995. Organizational risk factors for job stress. Washington, DC, US: American Psychological Association, xii, pp. 163-179.
2. Ball KS. 2001. Situating workplace surveillance: Ethics and computer based performance monitoring. Ethics and Information Technology, 3(3), 209-221.
3. Pimenta R, Carneiro D, Novais P, Neves J. 2014. Analysis of Human Performance as a Measure of Mental Fatigue. In Hybrid Artificial Intelligence Systems (pp. 389-401). Springer International Publishing.
4. Carneiro D, Castillo JC, Novais P, Fernández-Caballero A, Neves J. 2012. Multimodal behavioral analysis for non-invasive stress detection. Expert Syst. Appl. 39(18): 13376-13389.
5. Picard RW, Wexelblat A, Clifford IN. 2002. Future interfaces: social and emotional. In CHI'02 Extended Abstracts on Human Factors in Computing Systems (pp. 698-699). ACM.
6. Corchado E, Abraham A, Carvalho A. 2010. Hybrid intelligent algorithms and applications. Inf. Sci. 180(14): 2633-2634.
7. Welford AT. 1977. Motor performance. Handbook of the psychology of aging, 2(450), 95.
8. McClernon CK, Miller JC. 2011. Variance as a measure of performance in an aviation context. The International Journal of Aviation Psychology, 21(4), 397-412.
9. Kahneman D. 1970. Remarks on attention control. Acta Psychologica, 33, 118-131.
10. Yalch R, Spangenberg E. 1990. Effects of store music on shopping behaviour. Journal of Consumer Marketing, 7(2), 55-63.
11. Hatem TP, Lira PI, Mattos SS. 2006. The therapeutic effects of music in children following cardiac surgery. Jornal de Pediatria, 82(3), 186-192.
12. Miluk-Kolasa B, Matejek M, Stupnicki R. 1996. The effects of music listening on changes in selected physiological parameters in adult pre-surgical patients. Journal of Music Therapy, 33(3), 208-218.
13. Gomes M, Carneiro D, Pimenta A, Nunes M, Novais P, Neves J. 2014. Improving Modularity, Interoperability and Extensibility in Ambient Intelligence. In Ambient Intelligence-Software and Applications, 63-70. Springer International Publishing.
14. Pimenta A, Carneiro D, Neves J, Novais, P. 2014. A Non-invasive Approach to Detect and Monitor Acute Mental Fatigue. In Modern Advances in Applied Intelligence, pp. 338-347. Springer International Publishing.
15. Carneiro D, Gonçalves S, Novais P, Neves J. 2013. Studying stress on e-Learning users. In Progress in Artificial Intelligence, pp. 1-12. Springer Berlin Heidelberg.
16. Hwang KA, Yang CH. 2009. Automated Inattention and Fatigue Detection System in Distance Education for Elementary School Students. Educational Technology & Society, 12(2), 22-35.
17. Hu X, Downie JS, Laurier C, Bay M, Ehmann AF. 2008. The 2007 MIREX Audio Mood Classification Task: Lessons Learned. In ISMIR, pp. 462-467.
18. Bacon CJ, Myers TR, Karageorghis CI. 2012. Effect of music-movement synchrony on exercise oxygen consumption. J Sports Med Phys Fitness. Aug;52(4):359-65.
19. Costa M, Carneiro D, Dias M, Novais P. 2015. How Musical Selection Improves the Interaction with the Computer. Studies in Computational Intelligence Volume 570, pp. 19-28, Springer.

20. Pimenta A, Carneiro D, Novais P, Neves J. 2015. Detection of Distraction and Fatigue in Groups Through the Analysis of Interaction Patterns with Computers. Studies in Computational Intelligence Volume 570, pp. 29-39, Springer.