ISSN 1750-9823 (print)
International Journal of Sports Science and Engineering
Vol. 01 (2007) No. 03, pp. 183-188

Development of a Computing Utility to Measure Time-On-Task in Injury Research Studies

William J. Montelpare [1], Joseph Baker [2], Brent E. Faught [3], Paul Corey [4], Morag MacKay [5]
Norm Lavoie [1], and Murray Nystrom [3]

[1] Lakehead University, Thunder Bay, ON Canada
[2] York University, Toronto, ON Canada
[3] Brock University, St. Catharines, ON, Canada
[4] University of Toronto, Toronto, ON, Canada
[5] European Child Safety Alliance (ECSA)

(*Received June 3 2007, accepted August 26 2007*)

Abstract. Quantifying time-on-task is important in many areas, but is of considerable importance to researchers considering the relationships between exposure and injury risk. This paper describes the method used to develop a generic data collection, storage and retrieval program for measuring time-on-task. The program uses hypertext markup language, javaScript and Perl to provide an easily navigated computer interface for recording exposure variables on a tablet style computer (laptop). This method allows for data collection in the field and provides several benefits over traditional methods of data capture. This utility may have application in a range of research areas including education and skill acquisition, in addition to providing a precise measure of exposure to high-risk environments.

Keywords: exposure, time on task, injury surveillance, data capture.

1. Introduction

Time-on-task data can be used to quantify exposure to specific environments, activities or systems that are identified as being of 'high risk'. One of our goals has been to develop a computing application that is easy to use in a field setting, and can generate reliable and valid estimates of exposure. Although the application described here was designed for the specific purpose of monitoring injury risks, there is transferability in both the technical aspects of the exposure monitoring tool and the procedures for data collection in other settings. For example, during the development of the initial prototype, researchers used the tool to conduct an ergonomic assessment of lifting and turning tasks in a bottling plant.

The purpose of this paper is to describe the generic approach used to design the data collection, data storage, and data retrieval program for measuring time-on-task. The initial application of the time-on-task tool was to evaluate the suggested injury risks associated with participation in youth ice-hockey.

The internet is ubiquitous, and we considered the familiarity of the user community – primarily other researchers, technicians, graduate students – to internet based applications when developing the time-on-task computing utility. In creating this utility a guiding principle was to avoid proprietary software dependency, and use only open source tools so that we could create an application on a stand-alone personal computing system that looked like a standard web page and required a minimal learning curve. The time-on-task computing utility is based on an integration of a web-server application and the scripting features of the interpreted languages javaScript and Perl. The specific application described here provides interactivity on a tablet computer to deliver a dynamic environment for data collection, data storage and data retrieval.

The main reason to develop this time-on-task utility was to provide an electronic method by which we could measure the exposure of ice hockey players to injury. This specific measure was part of a larger investigation of risks associated with introducing the skill of body checking among minor league ice hockey players.[1]

The first step in determining a score for participant exposure was to develop a procedure and protocol by

which the amount of exposure (in seconds) and the number of body contacts between participants could be measured. It has generally been accepted in sports injury epidemiology that the amount of exposure among participants to the risks of an injury reflects the pre-scheduled duration of the entire playing time for a planned event.[2] For example, injury incidence is a function of athlete exposure, where the number of "new injuries" recorded within a sport during a specific time period is divided by the total number of individuals that are considered participants during that time period. Recording injury incidence in this way enables researchers to compare between studies within a given activity.[1] However, in most sports, a general assumption is made that the participant is at risk for the entire duration of the event. This assumption typically overestimates the actual exposure time for a participant, given that rarely would an individual be involved actively for the entire duration of any event, and that most participation is dis-proportionately intermittent between participants.

A second problem that emerges in studying exposure is whether or not the amount of exposure to participation contributes to a positive injury prevention characteristic. This notion, which may be labelled injury hardening, emerges from the concept of work hardening whereby an individual increases their tolerance, level of knowledge, and practice of safe behaviours to specific job-related stimuli.[3] Similarly, the injury hardening framework suggests that the more an individual is exposed to 'risk' within an event, the more likely they are to habituate to the stimuli of the event and develop accommodating strategies which will enable them to avoid injury. In particular, exposure to a variety of scenarios in a game setting may increase an individual's awareness to events that may be more likely to cause injury in the novice or unexposed individual. Higher rates of exposure may lead to more rapid learning of safe behaviours, which ensure that they are more likely to avoid being injured and less likely to cause injury to other participants.

2. Technical Specifics

2.1. Scripting language and approach

The time-on-task computing utility was built with standard and advanced hypertext markup language (HTML), combined with the client-side and server-side interpreted scripting commands of javaScript and perl. Through the integration of these languages, the application was designed to present information and collect data of various types (time, frequency, and text).

HTML is the "language of the Internet".[4] The versatility of HTML encourages author creativity in both document delivery and in connectivity between documents and computer applications. HTML functions across hardware specific platforms and across operating systems.

Interpreted computer languages, of which javaScript and perl are but two examples, are comprised of *scripts* or *commands* that are either embedded in HTML files or called by commands within HTML files. The scripts are passed directly to the user, or client, along with the HTML code when the client accesses the web page during either an internet or intranet session. The scripts are interpreted by the client's web-browser and processed accordingly.[4]

An alternative to interpreted languages are "compiled" computer languages (e.g. java, C++). These languages are comprised of code that is processed by a compiler residing on the author's computer. The compiled code is accessed by the end-user in the form of an executable program. Compiled programs are typically more complex to create, require specific compiler software, and run as stand-alone applications (executable programs) on platform specific operating systems. Most compiled programs are proprietary.[4]

The common gateway interface (CGI) is a computer communication standard developed to enable communication across inter/intranets. CGI provides interactivity between the user's communication tool (i.e. the client computer) and a central processor (i.e. the server). In most inter/intranet applications the CGI is used to direct client computer information to a server based location on the web author's electronic storage directory.

The application described here was written in Perl, javaScript and html to run on an Acer TravelMate C110 tablet computer. The application was designed as a self-contained intranet middleware program that invokes an open source code Apache web server as the processing engine to serve a pen-based (tablet style) data collection template.

The unique features of this application include the ability to capture the actual time from the client computer's internal clock, the ability to create an image object that allows users to know when they are recording the exposure times for specific individuals, the computation of a running total for exposure times, and the ability to add and subtract counts from the individual cells of the player-wise recording table.

The application, shown in Figure 1, provides estimates of total amount of playing time by each player monitored at an event (whether the event be the total game or the individual periods), the number of shifts taken by the player within the period, and the number of body contacts the player makes over the entire monitoring period. The time-on-task utility has been used successfully in the Province of Ontario to collect data on Atom age minor hockey players (age 9-10) since the 2003-2004 hockey season (September to March), and more recently on Bantam age minor hockey players (age 14) during the 2006-2007 season.[1]

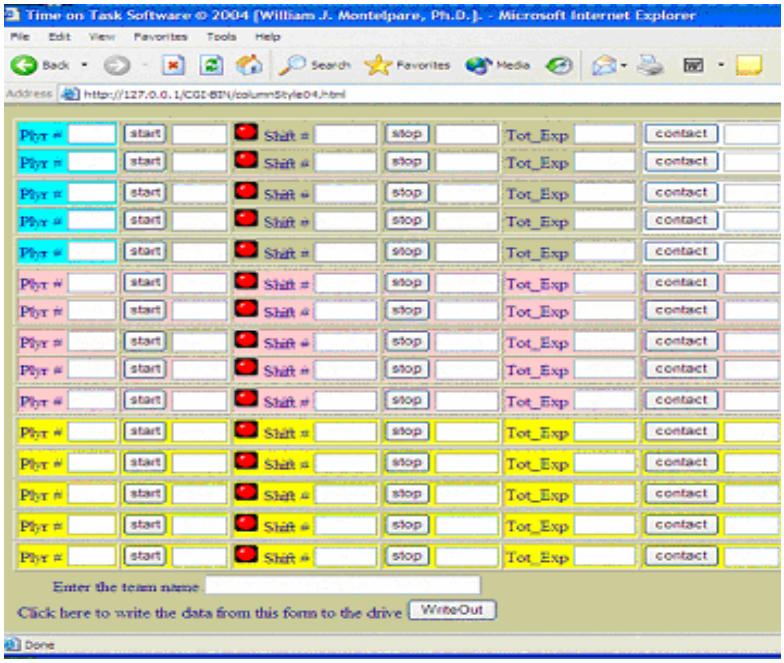

Figure 1. Screenshot of the pen-based data collection screen.

2.2. The workplan

The following plan was used to create the time-on-task computing utility running on a tablet (laptop) computer. The benefits of combining a scripting language (javaScript) with a common gateway interfacing software (perl) include increased speed of data recording as well as reduction in errors which are common in transcribing pencil-paper records to computer data sets. Through the development of a time-on-task measurement application researchers visited the regularly scheduled games and practice sessions to record the actual exposure time and number of body contacts through direct observation on-site (compared to watching video, e.g., [5]).

The underlying plan for the data collection, CGI processing, and data retrieval is presented in Figure 2. In step 1, the researchers worked within the realm of intranet development. Part of this work requires that the tablet computer be equipped with a recent version of open source web server software. For this, the researchers used the APACHE web server software Version 2.0, which can be downloaded from http://httpd.apache.org/download.cgi, or similar mirror sites. Once the web server is loaded and invoked on the user's tablet computer, the next step is to create a standard HTML form document that resides within the root directory of the "c" drive, in the program files sub-directory. Specifically, the path to the program used in this project was c://Program Files/apache group/apache/htdocs/index.html. The main page, called index.html, is a form document that includes a submit option, which invokes a perl based CGI program.

The perl based CGI program is considered to be the work horse of this monitoring system. The primary function of the CGI program is to capture the data from the html form document. That is, once the user has collected the data on the tablet, and clicked the submit button, the CGI program is invoked. The perl code interprets the input data from the html form document and converts the information to a format that is passed to a data storage file on the tablet computer. The CGI program and the HTML form document program reside in the cgi-bin directory, one level below the htdocs directory.

Data are read by the CGI program and are written to a standard data storage program as text. The text data output is stored in the cgi-bin directory as shown in Figure 2. The data can be accessed by the user via a shortcut icon from the tablet desktop. Providing shortcut access directly to the data set enables the user to transfer the raw data to other media for statistical analysis or interpretation. Once the data are written to the text storage file, the final task of the CGI program is to return the user to the data entry screen for continued monitoring.

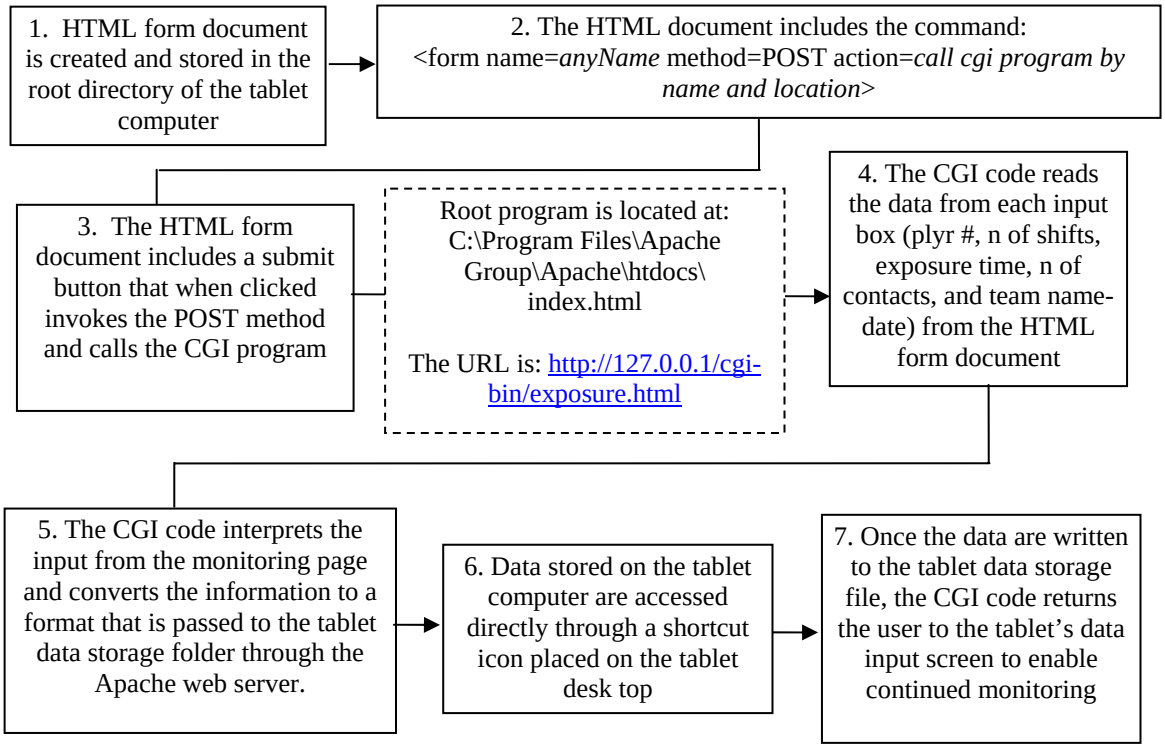

Figure 2. Flowchart of CGI processing and HTML procedures.

2.3. The data

A sample of the data collected using this method is shown below in Table 1. The preface statements: Plyr #=, shifts=, exposure time=, and num of contacts=, are text included by the CGI program each time the data are written to the storage file. This organizational strategy improves the readability of the raw data file. The preface statements can be easily removed from the raw data file, using any word processing program or text editor.

Player number is a nominal level variable and could be replaced by the player's name or other identifier. If the same team is monitored continuously, identifiers could be coded directly into the html form document. The number of shifts and exposure time are computed by the program based on the initial start and subsequent stop time buttons. The start and stop times are initiated by the user and exposure time is computed using the internal clock from the tablet. The number of contacts provides a cumulative count that is computed from the direct input of observations by the user. All data are reset when the user submits the form to the CGI program.

Table 1. Sample of data collected and written to the tablet computer.

Plyr #= 2	shifts=7	exposure time=218	num of contacts=6
Plyr #= 3	shifts=6	exposure time=198	num of contacts=6
Plyr #= 4	shifts=7	exposure time=348	num of contacts=10
Plyr #= 5	shifts=8	exposure time=212	num of contacts=1
Plyr #= 6	shifts=4	exposure time=187	num of contacts=6
Plyr #= 8	shifts=8	exposure time=207	num of contacts=2

Plyr #= 10	shifts=8	exposure time=244	num of contacts=3
Plyr #= 11	shifts=6	exposure time=289	num of contacts=5
Plyr #= 12	shifts=6	exposure time=249	num of contacts=10

3. Discussion

This paper describes a time-on-task computing utility that may have application in a range of research areas. The unique features of this application include the ability to capture the actual time from the internal clock of the tablet computer thereby allowing an accurate timestamp of the data, the ability to create an image object (e.g., a green light for 'on' and a red light for 'off') that clearly shows the user when they are recording exposure, the computation of a running total for exposure times and number of contact events, and the ability to add and subtract counts from the individual cells of the player-wise recording table in the event of a recording error. In addition, this application can provide a reliable method of collecting field data for time-on-task, which is expected to be faster than traditional methods, such as analysis of video footage.

Besides the injury epidemiology application noted earlier, the time-on-task application would be useful in several other areas. For instance, researchers examining the processes involved in skill development recognize the importance of time spent practicing; however, determining time-on-task, particularly in dynamic environments (e.g., team sports) is often difficult. This utility, with its multiple player inputs, could simplify this process appreciably. Likewise, time-on-task is a critical variable in studies of adapted physical education.[6-7]

Development of this time-on-task computing utility continues. With the advent of smaller computing devices with increasing storage capacity, similar systems may be possible for Personal Digital Assistant (PDA) devices and other wireless handheld devices, like wireless telephones. Researchers are encouraged to explore these options to assist in the collection of data pertaining to time-on-task and similar measures. Researchers that wish to obtain a copy of this software for non-commercial research may do so by contacting the author.

4. Acknowledgements

The authors wish to acknowledge the Ontario Hockey Federation and Hockey Canada for both financial support and access to ice hockey players, parents and coaches within the league membership.

5. References

[1] W.J. Montelpare, J. Baker, P. Corey, B. Faught, M. MacKay, N. Lavoie, M. Nystrom. *A multidisciplinary approach to measuring the effects of introducing body checking at the atom age level.* Ontario Hockey Federation and Hockey Canada, 2007.

[2] R. Pelletier, W.J. Montelpare, R. Stark. Intercollegiate ice hockey injuries: A case for uniform definitions and reports, *Am J Sports Med*. 1993, **21**: 78-82

[3] Guidelines: occupational health physical therapy: work conditioning and work hardening programs BOD G03-01-17-58 (Program 32) [Retitled: Occupational Health Guidelines: Work Conditioning and Work Hardening Programs, Amended BOD 03-00-25-62; BOD 03-99-16-49; BOD 11-94-33-109; Initial BOD 11-92-29-134]. American Physical Therapy Association. *Standards, Positions, Guidelines, Procedures: Professional and Societal* .

[4] W.J. Montelpare. McPherson M. Client-side processing on the InterNet: Computing the McNemar test of symmetry and the kappa statistic for paired response data. *The International Electronic Journal for Health Education.* 2000, **3**: 253-271.

[5] S. Horton, J. Baker, J. Deakin. Experts in action: A systematic observation of 5 national team coaches. *Int J Sport Psychol.* 2005, **36**: 299-319.

[6] B. Edwards-Duke, B. Boswell, S. McGhee, J. Decker. Creative educational dance and children with behavioral disorders: Encouraging a spirit of cooperation. *Journal of Dance Education*. 2002, **2**: 23-31.

[7] G. Owilia, R French, V. Ben-Ezra, L.M. Sillman. The influence of reinforcers on the time-on-task performance of adolescents who are profoundly mentally retarded. *Adapted Physical Activity Quarterly*. 1995, **12**: 275-288.