Challenges and Directions of Script-Based Tracking in Tutorial CALL

FENFANG HWU
University of Cincinnati

CHING-YEH TZSENG

ABSTRACT
Script-based tracking refers to using computer codes to log learners’ actions with computer-based activities. For decades, researchers have advocated using this type of data to advance CALL research. However, the amount of effort and time required to collect and analyze such data may have contributed to their limited use in tutorial CALL. This paper aims to describe the technical and methodological challenges researchers face and present corresponding solutions. The first step of the solutions includes defining a tracking taxonomy and a standard tracking interface. We will discuss a model in which these two sets of information can be used by the related parties in the CALL field, including researchers, software companies, and programmers, to create solutions for existing challenges and empower non-programmer researchers to conduct research using tracking data.

INTRODUCTION
Computer assisted language learning (CALL) researchers have a variety of computer-based devices at their disposal to gather tracking data, which can inform them of how learners use CALL materials. Script-based tracking is one such device. It uses computer code to log tracking data as learners use CALL software. Over several decades, researchers have documented the need for and various important pieces of information which may be drawn from using tracking data to advance CALL research (e.g., Chapelle, 2001; Chapelle & Mizuno, 1989; Collentine, 2000; Curtin, Avner & Provenzano, 1981; Fischer, 2007, 2008; Garrett, 1987, 1991, 1995, 1998; Hwu, 2003, 2007, 2012a; Jamieson & Chapelle, 1987; Liou, 1997, 2000). Further, three CALL professional organizations acknowledge the potential contribution of this type of data on the knowledge of second language acquisition (SLA) and the development of CALL theory (CALICO, EUROCALL, & IALLT, 1999). However, to date, “relatively few researchers have
actively pursued projects in this area” (Fischer, 2012, p. 15; see also Fischer, 2007). Some researchers attribute the limited use of tracking data in CALL research to the challenges in collecting and analyzing them (Fischer, 2007, 2012; Hwu, 2012a, 2012b). The specific challenges, however, remain unknown to many language instructors, CALL practitioners, and researchers.

This paper describes the challenges in obtaining script-based tracking data, hereafter, tracking data, as well as using and analyzing them. Additionally, taking the challenges into account, it suggests directions toward enabling CALL researchers to more easily obtain and use this type of data in their research.

**CHALLENGES: OBTAINING TRACKING DATA**

The first issue that the researchers who want to use tracking data face pertains to obtaining such data. This section describes various challenges in this area.

For example, some CALL authoring tools or applications do not provide tracking features. For example, in order to serve a large number of users who do not have a lot of technical knowledge or control over server technology, the authors of Hot Potatoes chose not to include tracking features in their tool suite (Arneil & Holmes, 1999). When a researcher wants to collect tracking data from an authoring tool or CALL application that does not provide tracking features, he or she needs to find a programmer to write computer code to collect and retrieve such data. Additionally, if the application is browser-delivered, a server needs to be acquired to store the data. However, funding or resources may not be available to accomplish these objectives.

Further, the tracking features of an authoring tool may have been conceived to aid instruction only rather than both instruction and research. As a result, they lack the control features required in a research design. For example, FLAn (Thibeault, 2012), a hypermedia authoring tool, tracks what glosses and program features (e.g., audio, video, translation, and web link) a learner accessed and at what point they were accessed. To obtain such data, the researcher has to ask the learners to remember to click a button to record their usage of the application and print such data before quitting the application. However, relying on learners to collect and provide tracking data is not an optimal or reliable way for a researcher to obtain data.

Another software tool that shares the same problem is Blackboard (9.1), a leading web-based course management system. The Review Status function, included in some of Blackboard’s course tools, allows the instructors to see whether and at what point (years, dates, hours, minutes, and seconds) an item or document was reviewed by the learner. However, such tracking data is not obtained automatically; the learner has to remember to click a designated button to indicate that an item has been reviewed. Even if a learner does remember to do so, a researcher can never be sure that an item has indeed been viewed.

Even if an authoring tool or template system is equipped with tracking features, the data needed for a research study still may not be collected because (a) the tracking features were not built to record certain usage data, (b) the tracking
features were not built to record the usage data of certain program components, or (c) the tool or application lacks the program components that can elicit certain usage data (Liou, 2000). We will explain each case in more detail below.

Blackboard (9.1) provides a good example of software not built to record the specific data a researcher might need. It provides instructors with the hit count of a learning tool (e.g., Course Documents, Assignments, Announcements, and Forums) or an area of the course (e.g., Content areas, overall summary), by date, hour of the day, and day of the week, through View Statistics Report and Course Reports. The hit counts may be useful to instructors but not to researchers. The reason is that the types of data that have been collected by researchers from applications providing similar program components, i.e., pages, go far beyond hit counts; they include the pages learners visited (Hwu, 2003, 2007, 2012b), the order in which the pages were visited (Desmarais, Laurier, & Renié, 1998; Hwu, 2007, 2012b), and the duration learners spent on a page or task (Collentine, 2000; Hwu, 2003, 2012b; Jamieson & Chapelle, 1987; Weinberg, 2005, 2007).

WebCT provides a good example of software lacking tracking of some, possibly the most, important components. The earlier versions of WebCT, a web-based course management system now owned by Blackboard Inc., tracked the content pages within its predefined content module; however, it did not track the customized links or pop-up windows within the content module (Hwu, 2003; Weinberg, 2005).

Blackboard also serves to exemplify the third case: it does not track some types of usage a researcher might want. Blackboard does not provide the program components (e.g., glosses and annotations) that can elicit the types of data collected by researchers investigating reading applications, such as the number of words readers clicked (e.g., Chun & Plass, 1996; Davis & Lyman-Hager, 1997; Knight, 1994), what annotation properties (e.g., text, picture, L1, L2) were associated with the words clicked (e.g., Laufer & Hill, 2000; Plass, Chun, Mayer, & Leutner, 1998), how much time readers spent on clicking annotations (De Ridder, 2000), and so on and so forth (Hwu, 2012a).

Even if an authoring tool or template system provides the tracking features that collect the data required by a research study, more fine-grained data may be needed, but it is sometimes difficult or impossible to obtain such types of data. For example, WebCT’s tracking records available for instructors only displayed hours and minutes; however, to differentiate the access times of the pages that shared the same hours and minutes, the page access times accurate to the second would be needed from the server records (Hwu, 2003; Weinberg, 2005). Unfortunately, WebCT did not provide instructors or researchers with an easy way to retrieve this type of data; instead, the server administrator had to give researchers access to the server to locate and retrieve the aforementioned data (Hwu, 2003). The tracking data of FLAn, a hypermedia-authoring tool, also do not show the access times accurate to the second. Since the consultation of a gloss or annotation usually lasts less than one minute, it is impossible to use its tracking data to accurately calculate the duration of a gloss consultation.
The authoring tools and template systems may not provide the tracking data in a format that allows the researchers to easily import such data into a data manipulation tool, such as an Excel spreadsheet or Access database. For example, while the tracking data provided by FLAn is in the plain text file format, WebCT only displayed the access records of the content module on the screen, and only ten records by a learner at a time (Hwu, 2003; Weinberg, 2005). As mentioned earlier, tracking data stored on a server may be more fine-grained (see Figure 1), but to efficiently import such data into a data manipulation tool, a program written to read such data is needed (Hwu, 2003).

**Figure 1 Sample tracking data retrieved from WebCT server**

```
LOC__WEBCT__homepage__LOC_TYPE__Home_page__TIME__1012581262
LOC__WEBCT__homepage__LOC_TYPE__Home_page__TIME__1012846986
LOC__no_quiso_decir_sergio_INTRO.html__LOC_TYPE__Notes__TIME__1012848268
LOC__no_quiso_decir_sergio_QS.html__LOC_TYPE__Notes__TIME__1012848284
LOC__no_quiso_decir_sergio_QS.html__LOC_TYPE__Notes__TIME__1012848300
LOC__no_quiso_decir_sergio_VIDEO.html__LOC_TYPE__Notes__TIME__1012848320
LOC__no_quiso_decir_sergioQUIZ.html__LOC_TYPE__Notes__TIME__1012848597
LOC__no_quiso_decir_sergio_EXP.html__LOC_TYPE__Notes__TIME__1012848709
LOC__no_quiso_decir_sergio_QS.html__LOC_TYPE__Notes__TIME__1012848752
LOC__no_quiso_decir_INTRO.html__LOC_TYPE__Notes__TIME__1012848757
```

**CHALLENGES: USING AND ANALYZING TRACKING DATA**

By now, it is likely that many instructors and researchers are aware that tracking data can tell them whether and how learners use software components, and consequently, they can use such data to guide them in program design and learner training (Fischer, 2007, 2012; Hwu, 2012a). Nevertheless, how exactly such data can facilitate the aforementioned and, in particular, other areas of research studies may not be as evident to them. In fact, throughout the years, CALL researchers have noted various challenges in this area. For example, Goodfellow and Laurillard (1994) noted that although it is clear that tracking data provides insights about language learning processes and there are suggestions on the areas where they would be useful, precisely what researchers should do with this information is unclear. In Chapelle’s (2001) view, the real challenge pertains to determining what kind of tracking data evidence is required to address a particular research question. Further, Hwu (2007) indicated that when the application type being studied was uncommon in CALL research, formulating research hypotheses *a priori* was impractical because exactly what behavior would be observed was unknown. What was done instead was visually sifting through behavioral data, large in size, after they were collected to identify patterns that could help formulate research
hypotheses *a posteriori*. Finally, Weinberg (2007) claimed that instructors often do not consider examining tracking data due to the difficulty of extracting useful and meaningful information from them (see also Fischer, 2012; Hwu, 2003).

With regard to tracking data interpretation, Hwu (2003) and Weinberg (2005, 2007) pointed out that because many factors can affect the amount of time learners spend on an activity, it is challenging to interpret such data. For example, learners can leave a pop-up window open while accessing a different page, print a page to read, leave the machine while being logged on, or engage in multitasking or continuous partial attention (Stone, 2012) while being on a page. Since a longer duration can mean a number of things — the activity being too difficult, the learner being distracted or, in contrast, engaged — careful considerations need to be made to find optimal ways to interpret time on task. Further, Weinberg (2007) stated that technical assistance was needed to manipulate and clean data. For example, automatic runtime routines had to be used to check for page-access overlaps and spurious data records, such as extremely short and unrealistically long durations.

The various challenges related to using tracking data in CALL research is summarized well by Fischer (2012):

Analysis of tracking data can be an arduous and time-consuming undertaking, requiring careful planning and preparation, anticipating and resolving technical problems in the software itself, storing student information in safe and secure sites, coding learner data appropriately, and finally trying to make sense of the potentially massive amounts of data.  
(p. 15)

**FUTURE DIRECTIONS**

We believe that to resolve the aforementioned challenges, CALL researchers first need to create a tracking taxonomy and define a standard tracking interface (STI). Creating a data-format standard or specification to facilitate information exchange and data sharing among members of a discipline is a common practice in both education and industry. For example, the Chemical Markup Language (CML) is used in chemistry to facilitate the exchange of molecular information. Further, the Interactive Financial eXchange (IFX) specification is used in the banking industry to aid financial transactions between companies. Finally, the Classification Markup Language (ClaML) standard is used in the medical field to assist information exchange on diagnoses and procedures.

The use case diagram in Figure 2 shows how the tracking taxonomy, the STI, and other actors should interact to empower researchers, usually not proficient in programming, to use tracking data in their research studies.
First, CALL researchers need to create a tracking taxonomy by examining past studies in various application types to identify the range of research questions answered by using tracking data and the behavioral data collected and used to answer these questions (e.g., Hwu, 2012a, 2012b), as well as technical and methodological needs arising during data collection and analyses (e.g., Hwu, 2003, 2007; Weinberg, 2005, 2007).

Second, using the taxonomy, a committee of CALL researchers will need to define a STI, a set of specifications outlining the needed tracking features in all CALL applications. This committee can be comprised of members from organizations such as CALICO, EUROCALL, IALLT, and IATEFL. It should also be supported or endorsed by these organizations. These specifications may suggest the range of data to be collected for the applications providing annotations, the recording of behavioral data to be automatic (vs. learner-dependent), and the customizable windows and links to be trackable. They may also recommend the degrees of accuracy of page access time (e.g., seconds, nanoseconds), the tracking
data formats (e.g., database schema, XML schema), the tracking data destination types (e.g., DB, XML), and the functions of a data manipulation or cleaning tool (e.g., to allow researchers to define spurious data and remove them).

Third, because the purpose of the specifications of the STI is to provide standards for tracking application programming interfaces (APIs) to ensure that different CALL applications follow the same standards, the availability of a STI may motivate software companies to provide their CALL applications with tracking APIs. Note that in addition to motivating software companies to develop tracking APIs, another option is to submit the STI to open-source learning management systems, such as Sakai and Moodle, so that applications supporting the STI can be developed and researchers using open-source learning management systems can immediately benefit from such development.

Fourth, when CALL applications are equipped with tracking APIs, it will simplify the tasks of programmers who write computer code to collect tracking data from CALL applications because they can use the same code in different applications; most importantly, it may motivate programmers or software companies to create data collection utilities that allow non-programmer researchers to collect tracking data and export them to a desired location (e.g., a database), making the work of collecting and analyzing tracking data less arduous. Because the data format and the output format of such tracking data will have already complied with the STI, such data can be easily shared among researchers through data repositories, thus allowing researchers to achieve many important goals in research (Hwu, 2012a, 2012b).

Note that an additional benefit of a tracking taxonomy is that it can help guide researchers in methodological choices during the stage of research design, such as what data evidence is required to address a particular research question. Another benefit is that as more research studies that demonstrate useful ways to interpret learners’ usage data in CALL applications are conducted, software companies, such as Blackboard, and the authors of CALL applications will become more informed about tracking needs and may begin to provide tracking features and program components useful to CALL research in their applications. Further, after the STI becomes widely accepted, university server administrators may become more willing to assist researchers in using university servers to collect learning data.

CONCLUSION

Understanding the ways in which learners use CALL materials is significant for the advancement of CALL. As tracking data can provide direct evidence of the learner’s use of CALL materials, they can help researchers develop and improve such understanding. However, before more researchers are motivated to undertake this type of research, it is necessary to identify the challenges encountered previously so that appropriate actions can be taken and solutions can be devised. The proposal made in this article is preliminary and high-level in nature. Much more work still needs to be done by CALL researchers. Only when they work
collectively, will the related parties of the field, such as software companies, university server administrators, and programmers, begin to take notice of and accommodate the needs of research and development in CALL. Then tracking data may be improved upon and used regularly to gain insights into student learning with CALL applications.

NOTES

1 Blackboard uses a variety of tools and functions to provide instructors with the information about learner activity within Blackboard, e.g., View Statistics Report, Course Reports, Performance Dashboard, and Last Course Access (Blackboard Inc., 2012).

2 Some template systems provide tracking data in a format that allows easy exportation of data. For example, Blackboard provides the hit count data in Excel format (in addition to PDF, HTML, and Word) and SANSSpace, a virtual language learning environment, provides tracking reports in CSV format (comma-separated values).

REFERENCES


