Percy – an HTML5 framework for media rich web experiments on mobile devices

Christoph Draxler

Institut für Phonetik und Sprachverarbeitung
LMU München, Munich, Germany
draxler@phonetik.uni-muenchen.de

Abstract

Percy is a small software framework for perception experiments via the WWW. It is implemented entirely in dynamic HTML and makes use of the new multimedia tags available in HTML5, eliminating the need for browser plug-ins or external players to display media content. With Percy, perception experiments can be run on any platform supporting HTML5, including tablet computers, smartphones or game consoles and thus access new participant populations.

Percy supports touch interfaces and measures reaction times. It stores its data in a relational database system on a server. This allows immediate access to the experiment data via standard database access APIs.

The system has been used for a number of experiments in German, Castilian Spanish and English.

Index Terms: online experiment, HTML5, reaction times, multilingual framework

1. Introduction

Online experiments have become a standard way of running experiments, e.g. in psychology, market research, linguistics, phonetics, and other fields. A number of frameworks that support both configuring and running web-based experiments have been developed, e.g. WebExp [1], DEWEX [2], and others. Furthermore, there now exist widely accepted guidelines on the design and performance aspects of web experiments [3].

HTML up to version 4 did not have specific tags for media content. As a workaround, a generic object tag had to be used. In order to playback audio or video, developers basically had three options: use a plug-in that runs within a browser, e.g. Adobe Flash, use an external player called by the browser, e.g. Windows MediaPlayer or QuickTime Player, or install an application and run the entire experiment outside the browser, e.g. using Java Web Start.

In general, these options are not available on mobile or home entertainment devices – as a consequence, most existing online experiment software with media-rich content cannot be used on these devices.

However, these devices are becoming increasingly popular:

1. music players, smartphones, tablet computers and game consoles have become now powerful enough to display web and media content, and

2. intuitive touch and multimodal user interfaces greatly facilitate the use of web services.

Thus, making web experiments available on mobile devices allows accessing participant populations that until now were difficult to reach, and in environments that until recently were inaccessible, e.g. schools, retirement homes, public places – even at home, using the TV!

2. Client-Server Architecture

The client-side software consists of an HTML5 (*hypertext markup language*) document describing the structure of the web site, a CSS (*cascading style sheet*) document defining the appearance of the web page elements, and a suite of JavaScript program scripts implementing the functionality of the system. These scripts make use of the open source framework jQuery [4] for accessing and manipulating web page elements.

Any modern browser supporting HTML5 can run web-experiments using Percy without any plug-ins, external media players or extra software.

In a typical experiment session, a user registers for the experiment (anonymous participation is supported). The experiment stimuli are downloaded to the client and arranged according to the experiment specification, i.e. either in sequential or random order. During the experiment session one stimulus item is presented after the other, and the system proceeds automatically until all items of the session have been processed. Then either a thank you message or a summary of the experiment is presented to the participant.

The server receives data from the client as attribute-value pairs according to the CGI standard using UTF-8 encoding via the post method, allowing for arbitrarily large amounts of data to be transferred. Data is sent to the client in XML-format and Unicode UTF-8 encoding.

The current version of Percy is implemented as a Java servlet accessing a relational database system. The use of a database system allows tools with database interfaces, e.g. spreadsheet programs such as Excel or statistics packages such as R, and human operators to access the experiment data using standard SQL queries.

3. Data model

The data model of Percy is divided into a configuration branch which defines the contents of an experiment, and a participation branch which holds the data generated during an experiment session (fig. 1).

In the configuration branch, a project consists of one or more experiments. Each experiment contains one or more scripts which in turn are organized in sections. Sections are groups of items; the order of sections is fixed, but within a section, items can be arranged sequentially or in random order. An item consists of a link to an audio stimulus, an instructions text, and an options and an assessment field.

In the participation branch, an experiment session links a participant with an experiment script and one or more user input items. An input item contains a stimulus item and the value of the user input, together with log data recorded during the presentation and processing of the stimulus item.

3.1. Relational database

The data model of Percy is implemented using a relational database with 8 tables normalized to third normal form (fig 1).
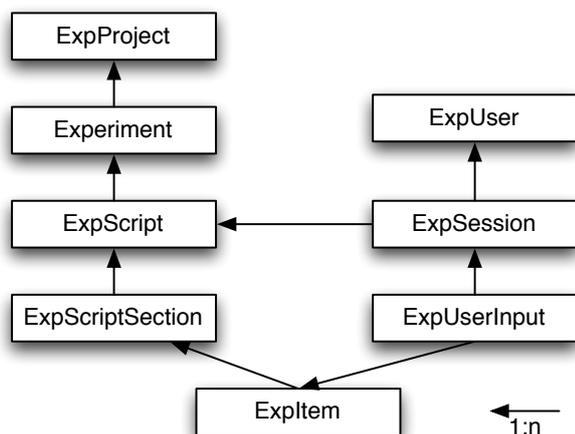

Figure 1: *Relational data model for the experiment database*

The values for the session and user table attributes are collected at the beginning of an experiment session. Wherever possible, fixed vocabularies are used to enter this data, e.g. a pull-down menu with the options *earphones, headphones, built-in speakers, external speakers* for the audio playback device.

3.2. User input data

The input data consist of both automatically collected data and data entered manually by the user as a response to the stimuli. The manual data is entered via interactive widgets, e.g. buttons.

The automatically collected data comprise the number of repeated playback of the stimuli plus four timestamps for the i) presentation of the stimulus, ii) start and iii) stop of stimulus playback, and iv) the user input time (see fig. 2).

The display of icons and input elements depends on the status: when the stimulus is ready for playback it becomes active. The input element only becomes active when the stimulus has played at least once. When the maximum number of playbacks has been reached, the stimulus icon is crossed out to prevent further playback (see fig. 2).

4. Experiment Examples

Percy has now been used in a number of experiments: regional variation of phonetic features in German, aspiration in Castilian Spanish, irony detection in US English, and others.

Fig. 3 shows the experiment view for the regional variation experiment – the user was asked to decide whether he or she heard a voiceless or a voiced initial alveolar fricative for the word *sieben* (/ˈz iː b ə n/ vs. /ˈs iː b ə n/, *seven*).
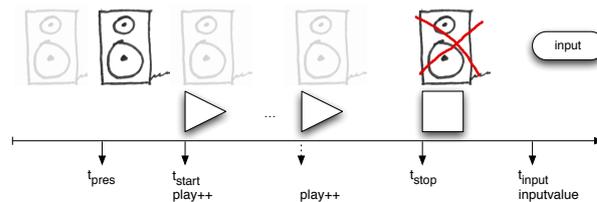

Figure 2: *Logging of user input and timestamps. The triangles denote begin of playback, the square stop of playback. A solid icon is active, i.e. can be clicked for playback, a grey icon is inactive, i.e. the stimulus is being loaded or playing.*
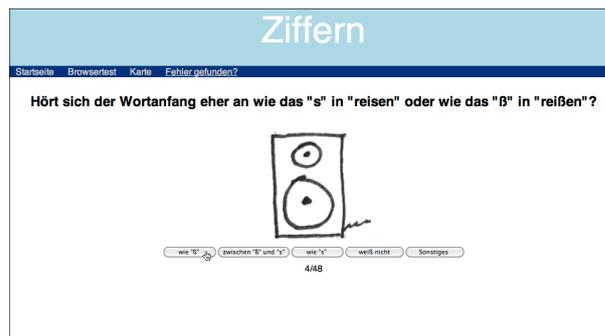

Figure 3: *View of a typical experiment view with the audio playback icon and buttons to capture user input.*

5. Conclusion and Outlook

Percy is now stable and mature. The major problems we encountered were compatibilty and audio playback problems. The most recent versions of Firefox, Opera, Chrome and Safari worked well on a large variety of devices, including tablets (iPad) and smartphones (Android and iPhone).

Currently, setting up an experiment requires access to the server's database and HTML knowledge. A web-based editor for the configuration of online experiments is under construction. Once the editor is done, the software will be made publicly available.

A set of web experiments using Percy is available at `webapp.phonetik.uni-muenchen.de/WebExperiment/index.html`.

The Bavarian Archive for Speech Signals offers to host web experiments as a service for the speech community. Contact the author for details.

6. References

[1] F. Keller, G. Subahshini, N. Mayo, and M. Corley, "Timing accuracy of web experiments: A case study using the webexp software package," *Behavior Research Methods, Instruments and Computers*, vol. 41, no. 1, pp. 1–12, 2009.

[2] A. Naumann, A. Brunstein, and J. K. Krems, "Dewex: A system for designing and conducting web-based experiments," *Behavior Research Methods*, vol. 39, no. 2, pp. 248–258, 2007.

[3] U. Reips, "Standards for internet-based experimenting," *Experimental Psychology*, vol. 49, no. 4, pp. 243–256, 2002.

[4] "www.jquery.org."