Tutorial: Introduction to PCIbex – An Open-Science Platform for Online Experiments: Design, Data-Collection and Code-Sharing

Florian Schwarz (florians@ling.upenn.edu)
Department of Linguistics & MindCORE, 3401-C Walnut St, Suite 300
Philadelphia, PA 19104 USA

Jérémy Zehr (jeremyz@ldc.upenn.edu)
Linguistic Data Consortium, 3600 Market St # 810
Philadelphia, PA 19104 USA

Keywords: online behavioral experiments; software; data collection platform; audio- and video-recording; eye tracking

Introduction

PCIbex is a free web platform for designing and running a wide variety of behavioral experiments online. It provides a simple and accessible, yet versatile code interface for designing experimental tasks, and furthermore makes it possible to share experiment code and resources with a single click, allowing for easy replication. PCIbex comes with an easy-to-learn mini-language, which builds on a standard JavaScript engine. Advanced features include audio- and video-recording as well as an integration of the webgazer eye-tracking API. The workshop will provide a hands-on introduction to the basic functionalities, as well as illustrations of advanced features and time for Q&A. While already in fairly wide use in the linguistics community, the tutorial aims to make this resource more widely accessible within the cognitive science community at large.

The PCIbex Farm

The PCIbex Farm, at `https://farm.pcibex.net/`, is a web platform serving two key functions: providing a simple coding interface for implementing experimental designs, and sharing the resulting experiments via web browser both for data collection and Open Science resource sharing. The platform builds on IBEX (Drummond, 2007), but expands its capabilities and makes a much wider range of tasks easily accessible through the `PennController.js` library; the PCIbex Farm also offers an overhauled design interface and numerous new functionalities.

Experiment Design Interface

PCIbex uses its own mini-language (requiring no JavaScript or other prior programming skills) to specify the structure of individual trials as well as the overall experiment. The design interface furthermore makes it easy to integrate resources, such as visual and auditory stimuli files, and also includes a trial preview window for easy testing during development.

Simple but Versatile Code Base The PCIbex mini-language is set up to be maximally accessible while allowing for full control over what happens when (and where) in a given trial. The central ingredients are *elements* (text, images, audio, video, timers, etc.) which can be subjected to a host of actions (presentation/playback etc.) with detailed control over timing and location on the screen. Beyond basics such as presenting text in various forms and asking for text/key-press or mouse-click input, it can incorporate a diverse range of features to accommodate both simple and complex experimental task paradigms with a wide range of dynamic and interactive features (e.g., visual stimuli, dynamically unfolding trial structure, response feedback, scripted/ timed events, playing audio and video), with the capacity of logging timing and inputs throughout. It also seamlessly integrates existing functionalities for common psycho-linguistic tasks from the original IBEX, including self-paced reading and rating studies.

Intuitive Design Interface The design interface on the new and improved farm launched in 2021 provides easy access to all main features on a single page. The code window uses the Ace Editor (ajax.org, 2010-21), offering many conveniences such as command completion drop-down menu while typing and automated color-coding. The preview window for easy testing during development, and a resource folder section where stimuli files (e.g., images, sound files, etc.) can be uploaded and managed via click & drag. A list of action links lets users manage sharing, access results, download a zip archive, and access settings. Users can create new studies from scratch, by uploading zip files of prior IBEX-based experiments (from any farm), or from the provided templates (as well as one-click sharing links; see below), and have any number of projects (up to the maximum storage limit).

Running and Sharing Experiments Online

Any experiment on the PCIbex farm can be instantly shared via two different URLs: one for the purposes of data collection for participants, and one for sharing code and resources with the scientific community in a single click.

Experiment Running Data collection capacity is directly built into the farm. All that is needed to point participants to take the experiment is to share a data-collection link once the experiment is set to 'Published.' This setup is compatible with a variety of recruitment platforms, e.g., SONA, Prolific, or Mechanical Turk, and user ID's and confirmation links can be automatically handled through PCIbex functions accessing URL parameters. Test runs during development use a

separate URL to keep apart actual data and test-run results.

PCIbex experiments are executed entirely locally after all necessary files and resources are downloaded, which keeps to a minimum the standard limitations of the hardware used by participants. Independence from connection quality during runtime leads to reliable and accurate timing control and information, both for stimulus presentation and responses, making it suitable for most time-sensitive tasks, including (many variants of) priming studies.

Open Science Code Sharing Experiments come with a (fully anonymous) 'demonstration' link to facilitate public access to the actual experiment experience, e.g., during review and for readers of papers. A first hand experience greatly enhances engagement with reports of experimental work and conveniently supplements methods sections. Furthermore, by default, the demonstration experiment page contains a link for cloning the experiment, with full access to underlying code and resources to promote Open Science practices (this can be turned off if necessary), allowing for one-click replication (no account sign-in required to see code)—illustrated for the basic tutorial experiment here: `https://farm.pcibex.net/r/QuFrkC/`.

Advanced Functionalities

PCIbex offers a number of advanced functionalities, which are of particular importance during current limitations on in-lab data collection. These include trial-by-trial recordings of audio and video via participants' microphone and webcam. Furthermore, it integrates the webgazer eyetracking API (Papoutsaki et al., 2016). While these currently require additional initial setup for storage of the resulting recordings and files on the experimenter's own server space, these capacities substantially lower the bar for deploying more sophisticated paradigms in online experimentation, including language production studies, video-based gaze tracking (with posthoc manual gaze coding), and eye tracking within the limits of using webgazer with webcam input. In addition, server space setup by individual researchers enhances IRB-compatibility, as more sensitive data, e.g., video recordings of child participants, can be stored on suitable server space approved by the researcher's institution.

The Tutorial

The tutorial aims to make the PCIbex platform and its code interface accessible to a wide audience, with no pre-requisites or prior technical skills required, but will also be of interest to students and researchers with more substantial background. The first part offers a hands-on introduction to its basic functionality, whereas the second part illustrates some more advanced features and allows for Q&A of both basic and advanced technical questions.

Part 1: Hands-on Introduction

The hands-on tutorial—a variant of which has been given in numerous contexts already—walks through the creation of a sample experiment combining display of unfolding text, images, and audio with different response input options, first for a single trial and then for a multi-trial experiment with basic randomization. As participants follow along creating their own version of this experiment, they learn the basic syntax of the PCIbex mini-language, as well as the logic of implementing basic trial and experiment structure, including the use of audio and visual resources. While necessarily confined to basic functionalities, this enables participants to independently explore working with the platform afterwards, and, with the help of the extensive online documentation and active support forum, should equip them to flesh out their first full experiment.

Part 2: Illustration of Advanced Features & Q&A

The advanced functionalities outlined above offer a wide range of more sophisticated experimental paradigms of general interest to the CogSci community. While we will not be able to work through all technical details within the tutorial, a basic illustration and general introduction to the additional setup needed should provide a solid starting point for participants to start putting together their own setup for utilizing these advanced capacities.

Towards the end of the tutorial, we will also leave room for Q&A to allow participants to raise any questions—whether basic or advanced—that they run into as they're learning about PCIbex.

Acknowledgments

We gratefully acknowledge support from various sources: Early versions of the PennController library at the core of PCIbex were developed during work within NSF grant BCS-1349009 to Florian Schwarz, and were more fully developed with a grant from Penn's University Research Foundation titled 'Building a Comprehensive Library for Online Experiments in IBEX.' The project has also received additional support in various forms from the Interdisciplinary Language Sciences & Technology (ILST) and MindCORE at the University of Pennsylvania. We'd like to thank everyone that has contributed to PCIbex, specifically Amila Mujkić and Angelica Pan for their work to upgrade the documentation. Finally, thanks are due to Alex Drummond, whose original open source IBEX platform was our essential starting point.

References

Ajax.org, B. (2010-2021). *Ace editor.* `https://ace.c9.io/`. (Accessed: 2021-02-02)

Drummond, A. (2007-2021). *Ibex: Internet-based experiments.* `https://ibex.spellout.net/`. (Accessed: 2021-02-03)

Papoutsaki, A., Sangkloy, P., Laskey, J., Daskalova, N., Huang, J., & Hays, J. (2016). Webgazer: Scalable webcam eye tracking using user interactions. In *Proceedings of the 25th international joint conference on artificial intelligence (ijcai)* (pp. 3839–3845).