BRAMS-OTP

Introduction.

JATOS training http://www.jatos.org/.
https://brams.org/category/online-testing-platform/

Alex Nieva
Outline

• Pre-requisites to use the BRAMS-OTP
• Introduction to the BRAMS-OTP.
• Preparing your experiment.
• Working in JATOS local installation.
  • Adapt experiment to work in JATOS.
  • Test experiment.
  • Export files.
• Working in JATOS server.
  • Get participant (worker) links.
  • Downloading data from JATOS.
• Questions
Pre-requisites to use the BRAMS-OTP

• An experiment coded in JavaScript and that is functional.
• Basic programming experience.
• JavaScript basic user experience.
• HTML is a plus!
• Check out and review examples and demos!
  • [https://brams.org/category/online-testing-platform/](https://brams.org/category/online-testing-platform/)
BRAMS OTP layer structure

Server computer → Linux → Web Server → JATOS → HTML/JavaScript → Chrome/Firefox/Other

Storage

Participant

Local JATOS installation

Experimenter

Code editor

International Laboratory for Brain, Music, and Sound Research
Working experiment in HTML-JavaScript

• Hello World experiment.
• Major-Minor chords experiment.
  • We are going to use the jspsych library (https://www.jspsych.org/) which is written in JavaScript
Integrating the experiment with JATOS

1. Install JATOS in your personal computer.
2. Import an example in JATOS.
3. Create experiment in JATOS (this will create a folder in study_assets_root).
4. Modify already existing HTML-JavaScript so that it works with JATOS.
5. Copy files into JATOS assets directory.
6. Create your first component.
Modify already existing HTML-JavaScript so that it works with JATOS.

How to turn your jsPsych experiment into a JATOS study

1. Include the *jatos.js* library in the `<head>`

   ```html
   <script src="/assets/javascripts/jatos.js"></script>
   ```

2. Wrap jsPsych's init call *jsPsych.init* in a *jatos.onload* call

   ```javascript
   jatos.onload(function() {
     jsPsych.init({
       // ...
     });
   });
   ```

That's all. If you additionally want to send your result data to JATOS read on.

Send jsPsych's result data back to JATOS

Here we use jsPsych's function *jsPsych.data.getData()* (jsPsych 5) or *jsPsych.data.get().json()* (jsPsych 8) to collect the data into a JSON-formatted string. Then we use JATOS' function *jatos.submitResultData* to send your result to JATOS and ask JATOS to move to the next component, if there is one.

Send results back to JATOS

Send jsPsych's result data back to JATOS

Here we use jsPsych's function `jsPsych.data.getJson()` (jsPsych 5) or `jsPsych.data.get().json()` (jsPsych 6) to collect the data into a JSON-formatted string. Then we use JATOS' function `jatos.submitResultData` to send your result to JATOS and ask JATOS to move to the next component, if there is one.

### jsPsych 6

```javascript
jatos.onLoad(function() {
    jsPsych.init({
        // ...
        on_finish: function() {
            var resultJson = jsPsych.data.get().json();
            jatos.submitResultData(resultJson, jatos.startNextComponent);
        }
    });
});
```

Integrating the experiment with JATOS

1. Install JATOS in your personal computer.
2. Import an example in JATOS.
3. Create experiment in JATOS (this will create a folder in study_assets_root).
4. Modify already existing HTML-JavaScript so that it works with JATOS.
5. Copy files into JATOS assets directory.
6. Create your first component.
What are components?

*components are the main procedures of the study. The study can be one big component, or can have modules consisting in several components.

Important: If you are starting from scratch or adding a new component, you have to link the html file with JATOS via Components->+New. Fill up the required information.

*components are the main procedures of the study. The study can be one big component, or can have modules consisting in several components.
Integrating the experiment with JATOS

1. Install JATOS in your personal computer.
2. Import an example in JATOS.
3. Create experiment in JATOS (this will create a folder in study_assets_root).
4. Modify already existing HTML-JavaScript so that it works with JATOS.
5. Copy files into JATOS assets directory.
6. Create your first component.
Testing the experiment in Local JATOS
What are workers?

- Workers are the participants of your study!

Worker Types

Summary: Details of different worker types

Table of Contents
- Overview
- Jatos Worker
- Personal Single Worker
- Personal Multiple Worker
- General Single Worker
- General Multiple Worker (since version 3.3.2)
- MTurk (Sandbox) Worker

http://www.jatos.org/Worker-Types.html
What type of workers can you have?

<table>
<thead>
<tr>
<th></th>
<th>Jatos</th>
<th>Personal Single</th>
<th>Personal Multiple</th>
<th>General Single</th>
<th>General Multiple (since v3.3.2)</th>
<th>MTurk (Sandbox)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Icon</strong></td>
<td><img src="image" alt="Icon" /></td>
<td><img src="image" alt="Icon" /></td>
<td><img src="image" alt="Icon" /></td>
<td><img src="image" alt="Icon" /></td>
<td><img src="image" alt="Icon" /></td>
<td><img src="image" alt="Icon" /></td>
</tr>
<tr>
<td><strong>Typical use</strong></td>
<td>During study development</td>
<td>Small targeted group, each one of them gets a link</td>
<td>Small targeted group of workers who pilot the study or need to do it multiple times</td>
<td>Bigger groups but with less control; link shared e.g. via social media</td>
<td>Bigger groups and where the workers need to do it multiple times</td>
<td>For Amazon Mechanical Turk</td>
</tr>
<tr>
<td><strong>Created when?</strong></td>
<td>Together with the JATOS user</td>
<td>When you create the link</td>
<td>When you create the link</td>
<td>On-the-fly whenever someone uses the link</td>
<td>On-the-fly whenever someone uses the link</td>
<td>On-the-fly after a MTurk worker clicked on the HIT link</td>
</tr>
<tr>
<td><strong>Repeat the same study with the same link</strong></td>
<td>(has no links)</td>
<td>(keeps the same worker)</td>
<td></td>
<td></td>
<td>(creates a new worker each time)</td>
<td>(with Sandbox)</td>
</tr>
<tr>
<td><strong>Run different studies with the same worker</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Using batches

A batch is a collection of workers together with some properties. Using different batches is useful to organize your study runs, separate their results and vary their setup. E.g. you could separate a pilot run from the “proper” experiment, or you could use different batches for different worker types.

http://www.jatos.org/Run-your-Study-with-Worker-and-Batch-Manager.html
Organize your workers (participants) in batches. Click on New Batch.

Click on Get Worker Links and fill out the required information.

More batches with more Workers?

Send links to your Workers ([http://www.jatos.org/Worker-Types.html](http://www.jatos.org/Worker-Types.html))

Start

End
Going online

Importing your study to JATOS in BRAMS-OTP

otp@brams.umontreal.ca
Import your study to the BRAMS-OTP

- Get a BRAMS-OTP account.
- Export your study from your local JATOS
- Import your study to the BRAMS-OTP JATOS server.
Final steps

• Test your experiment again!!
• Get worker links. This is important since these will have the correct address.
• Get your data once the participant(s) is/are done.
Click on top Results tab (blue) to view results from all components. Click on light blue Results to view each component separately.

Select one, several, or leave blank for all the results

Click on Export Results and Export Metadata to save your data locally.
Collecting your data

• Export results and Export Metadata.
  • Results have the responses you expect from the participant, Metadata has the admin information such as Participant ID.

• Both files are going to have json type data structure.

• Each line on the exported results correspond to each line on the exported metadata.
Resources

• https://brams.org/category/online-testing-platform/
• http://www.jatos.org/
• http://www.jatos.org/Example-Studies.html
• https://youtu.be/PkZNo7MNFNg JavaScript tutorial for beginners
• https://www.jspsych.org/