

## Appendix A: Externalization Framework

This section briefly discusses features of our externalization framework not mentioned in the main submission. To get a better idea of how users interact with our system, we recommend watching the video in our additional materials. Figure 1 shows the visual analytics system with the incorporated externalization framework we used in our pilot study. In Figure 2, the different modes and their settings as provided by the framework are displayed. Both were implemented with the help of Vue.js, with the externalization framework being imported as a component library.

### State Management

Our externalization framework relies on the host application, in this case the visual analytics system, to define and pass the state as a JavaScript object when it changes. This allows the host application to decide which data should serve as a state representation and when an update should be tracked. Whenever a change is registered, the state is hashed to determine whether its new, whether a layer with this state already exists and whether it is different to the active layer's state. Thereby, we can switch to an existing layer when the same application state was already tracked and show any annotations made for that state.

We also define an *annotation* mode that determines when new layers are created. Three different options are provided: *on state change*, *on annotation* and *manual*. The latter, as aptly indicated by the name, simply never creates a layer automatically, leaving that choice to the user. When *on state change* is selected, a new layer is created whenever the state changes. Although this might seem like a good choice, as it tracks the complete history of application states, it can often result in large numbers of useless layers, where only one minor interface parameter was changed. Our default mode, *on annotation*, tracks the state changes but only creates a new layer when the state has changed compared to the active state *and* the user creates an annotation. This choice of default also seems to align with participants' expectations of how the application should work, although one participant was confused that the new layer is only created "after" an annotation is made. We also implemented a feature to let the user save the current application state to a layer, replacing its previously associated state.

### Interaction History

In addition to recording application states we implemented a simple interaction history tracker. Any interaction logged by our framework or the host application, i.e. the visual analytics system, is then reversible and shown in a hidden side panel of the application. This allows users to easily undo strokes or other interactions.

### Exporting Findings

In real-world scenarios, findings often need to be reported, documented and shared. There are many options to implement this and we chose to let users export an archive for a single layer that contains several files we generate from the layer's data. First, we generate a PDF report that includes an image of the annotated visualizations and any comments attached to the layer. Then, we also

generate two JSON files, one containing only the application state and one containing the complete layer information. The latter can be imported into the application to recreate the layer and all its annotations. An example report that was generated from our pilot study is included at the end of this document.

## Appendix B: Pilot Study

This section includes some more details on the study procedure and observations. Before participants worked on the tasks, they were given a short questionnaire with five questions detailed below. In Table 1, we list participants' answers to those questions.

**Q1:** What is your age?

**Q2:** What is your gender?

**Q3:** What is your highest qualification?

**Q4:** How familiar are you with interactive visualization (1 = barely to 5 = very)?

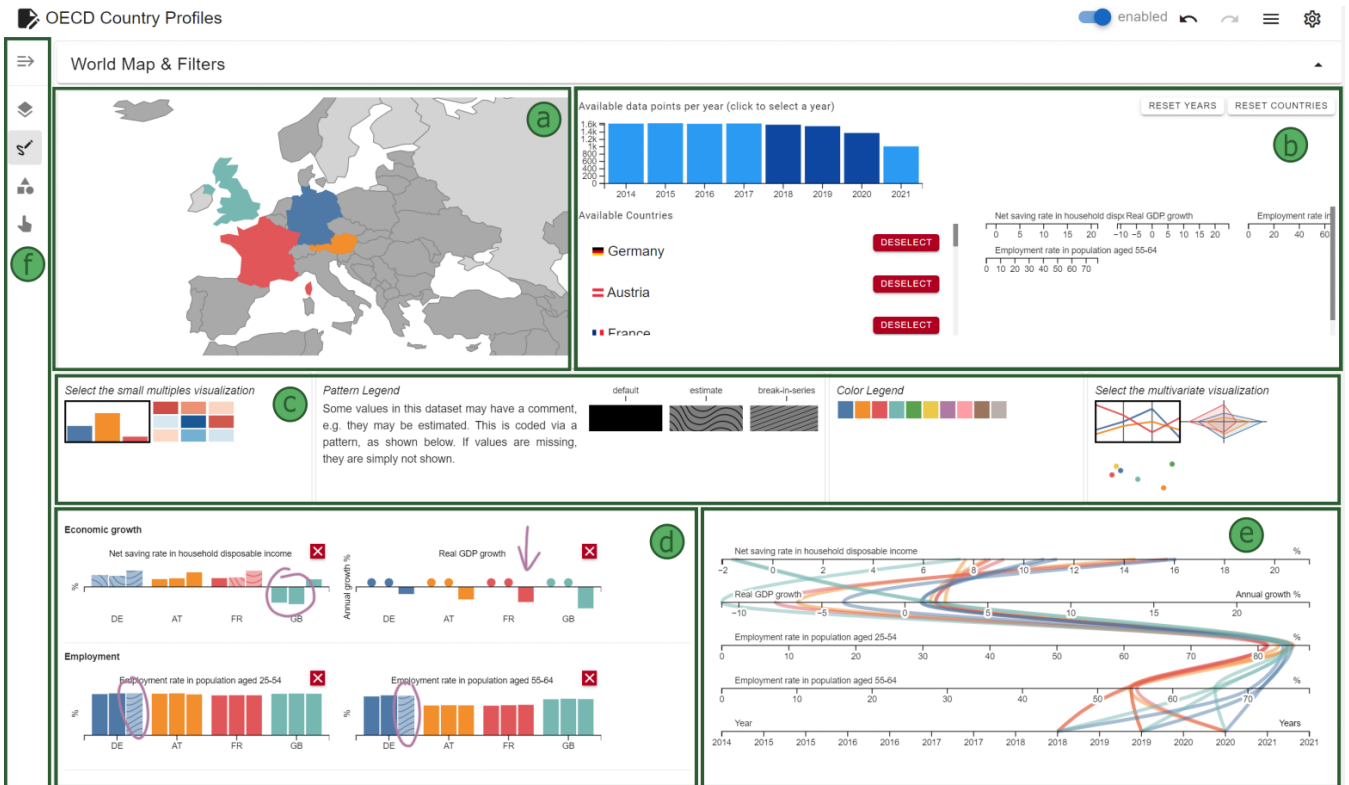
**Q5:** How often do you use touch and digital pen devices (1 = never to 5 = often)?

	$P_1$	$P_2$	$P_3$
<b>Q1</b>	42	36	30
<b>Q2</b>	male	female	male
<b>Q3</b>	graduate degree	graduate degree	graduate degree
<b>Q4</b>	5	5	4
<b>Q5</b>	4	5	2

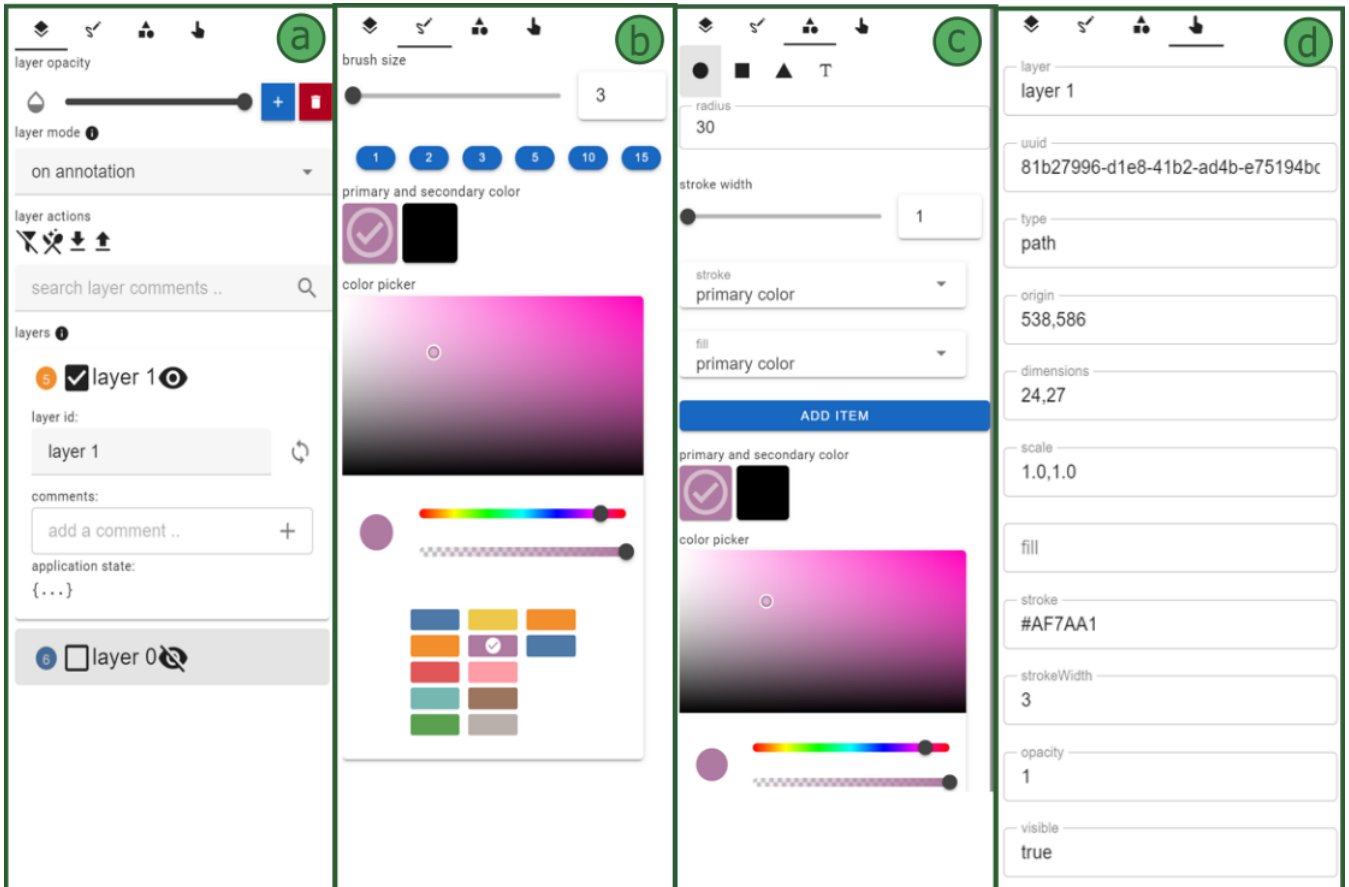
**Table 1:** Questionnaire answers from all participants.

### Observations

In this subsection we present more observations we made in our study. For  $P_1$ , we encountered the unfortunate but interesting situation that they accidentally lost the state for one layer by saving a new application state. While their verbal findings summary in the first task included most findings from the analysis phase, they only explored a handful of indicators. This changed for the second task, where they explored a larger number of indicators which they could then not remember in detail in their verbal summary, due to not having the expected annotations and visualizations available. Whilst trying to recreate his previous analysis,  $P_1$  narrated his actions and explicitly named the annotations he previously made. This might be an indicator that the actions and resulting annotations stood out more to him than the factual findings. For  $P_2$ , we observed several occasions where she forgot to switch between the layer and brush mode to scroll or annotate. In addition, she was the only one to mention that she was too lazy to switch to the text tool in the shape mode and instead preferred to just write a few letters as reminders instead. For  $P_3$ , who only uses pen and touch devices sometimes, we observed a different annotation behavior compared to the other participants. Although our brief introduction to the annotation functionalities mentioned making marks and highlighting interesting parts of the visualization,  $P_3$  instead made a bullet list of all his findings by writing all over the visualizations. He also showed some confusion that he could not keep this list even when the application state changed. While this could be solved by choosing a different annotation mode, some form of a state-independent notepad may work better.



**Figure 1:** Our visual analytics system with integrated externalization framework. a) displays a world map, where light gray countries can be (de-)selected and colored countries are part of the active selection b) shows the settings panel that also lets users (de-)select countries and additionally years and limit indicates values ranges via axis brushing c) contains several small panels that explain the interface and let the user switch between visualizations d) is the small multiples view that shows the country profile data for the selected indicators, countries and years with the specified visualization e) shows a chosen multivariate visualization of the same data f) is the collapsed settings panel from the externalization framework that can be expanded to for more extensive settings or used in the collapsed form to switch between modes.



**Figure 2:** The settings panels for each interaction mode. a) shows the layer mode, where layers can be selected, modified or searched via their comments b) shows the brush mode, which allows the user to pick the brush color and size c) shows the shape mode where shapes or text with varying size can be added with a button click d) shows the edit mode when an annotation object was selected.