Critique

Tracing a Paradigm for Externalization — Avatars and the GPII Nexus

Philip Tchernavskij

I find myself in violent agreement with Colin Clark and Antranig Basman. In this critique, I relate their paper to some Human-Computer Interaction (HCI) literature. HCI offers a tradition of studying computer-mediated human activity, and helps us build on the presented pattern. To illustrate this, I speculate on what sort of non-program artifacts avatars could beget. I also relate the concept of avatars to transclusion, as presented in our paper, *What can software learn from hypermedia?*. This paper is also instructive for HCI readers, in its ability to connect the means of programming with the ends of non-programmers. I conclude with some remarks on the research agenda that is implied therein.

The community perspective

A software pattern is a solution to a problem in a context. The avatar pattern is in part interesting because of the context and problem that it prioritizes. The authors find avatars in the wild, as one should when describing a proven solution. Their context is 'communities which need to produce artefacts that are shared as part of an open ecology of function'. The surveyed communities in the paper are musicians working with MIDI instruments, programmers debugging code, and people with disabilities that prevent them from effectively using unadapted interfaces.

Traditionally, object-oriented software patterns are evaluated for how costly it is to modify the resulting code, and how reliably the running program provides some specified service. Importantly, these are values for the producers of the software, not its consumers. The avatar pattern is argued for its ability to support living design artifacts that will occur in contexts not imagined by the original developers. It has been shown to be a solution to problems like:

- Letting musicians integrate a digital instrument in changing arrangements over decades
- Letting programmers create and use new debugging tools as they become computationally feasible
- Adapting interfaces to visually impaired users after they have been shipped

This perspective on technological artifacts as things that re-occur in *n+1* contexts has a history. Susan Leigh Star was a sociologist who was highly influential in HCI for her work on how information and technology is utilized in and across communities. In 1989, she argued in the Artifical Intelligence community that the ability to change and be appropriated is a better measurement of software intelligence than the ability to mimic a human sitting alone in a locked room. She hypothesized an alternative to the Turing Test, named after the socioligist Émile Durkheim:

> So the Durkheim test would be a real time design, acceptance, use and modification of a system by a community. Its intelligence would be a direct measure of usefulness applied to the work of the community: its ability to change and adapt, and to encompass multiple points of view while increasing communication across viewpoints or parts of an organization. Such a test also changes the position of metaphors with respect to design and use considerations. In an open, evolving system, the boundaries between design and use, between technology and user, between laboratory and workplace, necessarily blur.
>
> — , Susan Leigh Star, *The Structure of Ill-Structured Problems: Boundary Objects and Heterogeneous Distributed Problem Solving*, 1989

Studying real communities over time, as both Star and the authors of this paper have done, reveals what 'open ecologies of function' look like, and what kind of life cycles our tools should be built to support. Star notes, as do the authors of this paper, that building digital artifacts for such ecologies changes the relationship between designers and users.

With James Griesemer, Star developed the concept of *boundary objects*, objects that are simultaneously plastic enough to adapt to the local needs of one community of practice and robust enough to facilitate collaboration among several communities (1989). They described a taxonomy of boundary objects found in the ecology around Berkeley's Museum of Vertebrate Zoology in the early 20th century. This exemplar taxonomy included repositories, ideal forms, coincident boundaries, and standardized forms as types of artifacts that were used to mediate collaborative work across communities. Perhaps the next stage in tracing the paradigm for externalizaion is to develop a taxonomy of externalizations and related technical artifacts.

The tools perspective

The paper does not focus on the creation and manipulation of externalizations. It seems that avatars are assumed to be acted on through programming, albeit in some cases programming tools specialized for a particular community, such as musicians. Considering the ambition of the authors, to build adaptable user software in part by utilizing the externalization paradigm, we ought to investigate how the paradigm could be reflected in user interfaces. The user interface is a system's conditions for users to act towards objects and subjects (Bødker, 1987, p. 38). What conditions for instrumental and communicative actions should avatars provide?

Klokmose and Zander's 2010 paper about the design of digital laboratory notebooks provides an interesting scenario, developed in a design workshop with physicists. The physicists envisioned a system in which they had access to a shared digital notebook, as well as graphical control panels for every instrument in their lab. In the scenario (Figure 1), a group of scientists are performing experiments with a computer-controlled laser. A common task in experimental physics is to note down the configuration of the equipment, to aid experiment analysis and reconstruction. A particular setting of a group of knobs and sliders can be copied into the notebook trivially. Later, the scientists can reuse the configuration in the notebook, because it maintains its nature as a particular configuration of the laser. It is as if the software environment permits unscrewing the face of the instrument and storing it someplace else, as a piece of documentation of past actions, and as a saved image of a given configuration.

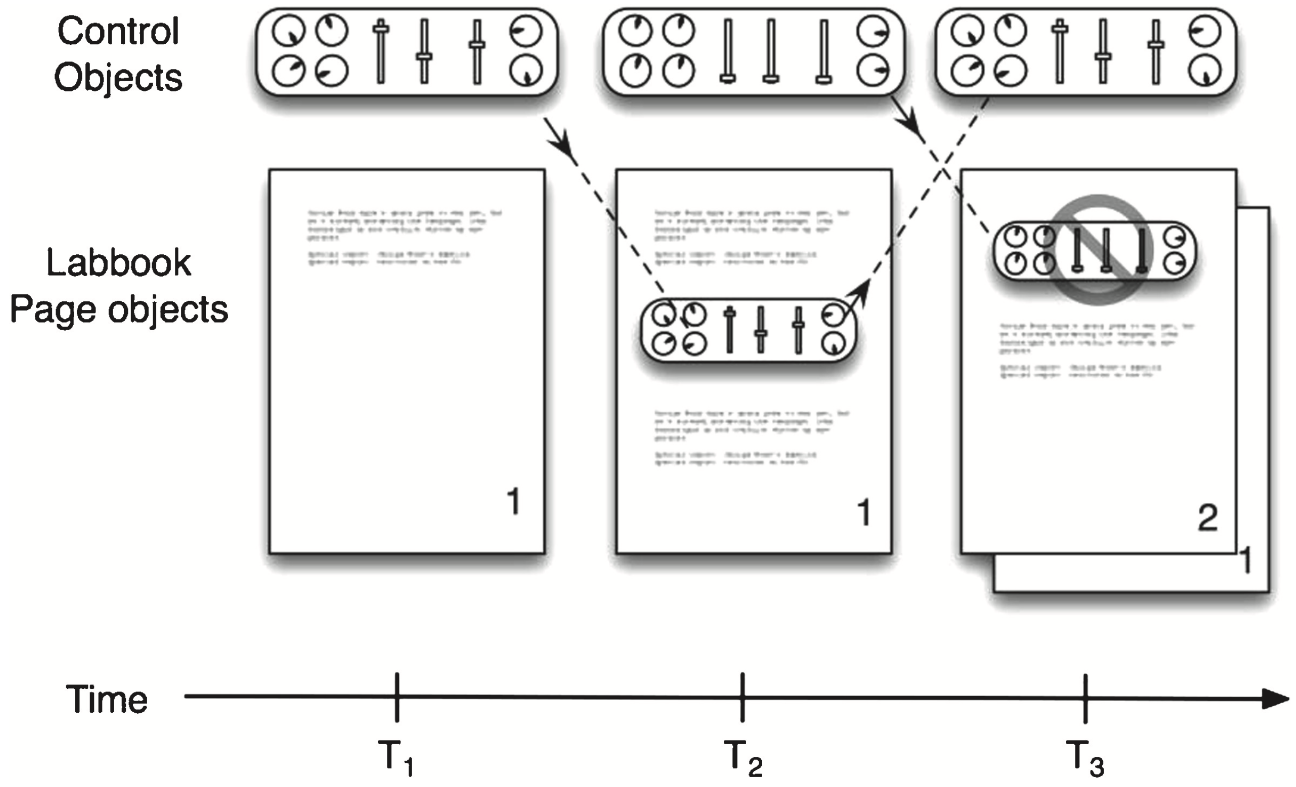

Figure 1. In the envisioned laboratory notebook there is no distinction between objects for documentation and objects for control. Documenting the position of a laser at a given time involves copying the object that acts as a control to a page in the notebook. Later this control can be reloaded to restore the setting. In the figure a copy of a control pane is stored on a page in the notebook. The controls are changed but reverted back to the original by using the copy stored in the book. The intermediate parameters are documented in the notebook as well and annotated as non-working. Reprinted with permission from (Klokmose & Zander, 2010).

In this case, the control panel can be seen as an externalization of the system that controls the laser. It is a digital artifact that oscillates between functioning as representation and tool. This echoes Clark and Basman's proposition that externalizations can 'bridge the divide between the living and the dead'.

What capabilities does this fiction imply for avatars?

In the scenario, avatars are stored, copied, reloaded, and annotated. Avatars are shown to be not just a means of control, but communicative artifacts to be read and written. Clark and Basman propose that interactive systems should advertise their internal structure via data and metadata. This advertisement should be designed for consumption by people as well as by other systems.

If I come across this control panel as I leaf through a laboratory notebook, it should be possible to understand what the knobs and sliders mean, and what machines the control panel is meant to attach to. The ability to attach human-readable annotations to the externalization can support this communicative task.

If a computer-controlled laser is given the control panel, it should be able to resolve the panel to its particular units and physical mechanisms. This binding task can be supported by metadata such as types, names, and operational histories attached to the control panel’s components.

This speculation implies further work to develop a readable, writeable, manipulable material for externalizations. Thus, considering and evaluating the material conditions in which a software architecture will occur motivates further technical research.

Avatars and transclusion

Finally, the avatar pattern bears some similarities to transclusion, as described in Clemens Klokmose, Michel Beaudouin-Lafon and my paper for this workshop. Ted Nelson defines transclusion as 'reuse with original context available' (1995) or 'the same thing knowably in more than one place' (2012). Avatars can be seen as transclusions utilized as a remote control mechanism. In *Webstrates*, deploying transclusion in this way is called *reification* (Klokmose et al., 2015). A reification is a webstrate that is bound to the state of some thing outside of the network of webstrates. For example, Klokmose et al. bidirectionally bound the state of a physical, programmable clock to a DOM representation through a small *node.js* server. Reifications are clearly a kind of avatar, though reification brings things *into* the *Webstrates* ecosystem, where avatars can be extracted from systems and live on externally.

In our paper, we ruminated on the limited ways in which apps could be composed. We argue that in the general case it is only possible to use apps sequentially, and then only when they have a shared file type. As our main perspective on transclusion is to see it as a software composition mechanism, perhaps avatars can be utilized that way as well. This would require avatars that can be combined, to create mashups of different systems, or translated, to bridge systems.

Webstrates is another illustration of the power of the externalization paradigm, and Clark and Basman’s analysis is a thoughtful reflection on how to build a robust mechanism for the kind of shared activity that we describe in our paper.

Concluding remarks

In this paper, Colin and Antranig persuasively argue for what software should be. If we uncritically follow engineering best practices, we take on the assumption that designers and developers should specify the full extent of acceptable software use. If we instead consider software as artifacts that live, grow, and intertwine within and across communities, we must find new best practices. The languages, protocols, frameworks, and libraries that support software systems are in dialogue with our idea of what people can and should do with software. Hence, pushing the dialectics of computer science, the stated mission of Salon des Refusés 2017, is part and parcel of creating more humane technology.

I am part of a five-year HCI research project, *ONE - Unified Principles of Interaction* (http://www.erc.one/). The goal of that project is to rethink interactive systems top-to-bottom to take advantage of the multiplicity of devices, interaction techniques, media, and collaborators that one is commonly surrounded by today. Part of that vision is to create building blocks for software that are adaptable by non-programmers for themselves and their community. The externalization paradigm sketched in this paper is a load-bearing step towards this goal. Work such as this is instructive to me personally, and generally to HCI researchers that want to push the means of programming towards human ends.

References

Susanne Bødker. „Through the Interface: A Human Activity Approach to User Interface Design“. DAIMI Report Series, [S.l.], n. 224, apr. 1987. ISSN 2245-9316. Available at: https://tidsskrift.dk/daimipb/article/view/7586.

Clemens N. Klokmose and Pär-Ola Zander. „Rethinking Laboratory Notebooks.“ In: Proceedings of COOP 2010 (2010), pp. 119– 139. doi: 10.1007/978-1-84996-211-7_8.

Clemens N. Klokmose, James R. Eagan, Siemen Baader, Wendy E. Mackay, and Michel Beaudouin-Lafon. „Webstrates: Shareable Dynamic Media.“ In: Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology - UIST ’15 (2015), pp. 280–290. doi: 10.1145/2807442.2807446.Star1989AI

Theodor H. Nelson. „The heart of connection: hypermedia unified by transclusion.“ In: Communications of the ACM 38.8 (1995), pp. 31–34.

Theodor H. Nelson. „Transclusion: Fixing electronic literature.“ 2012. url: https://www.youtube.com/watch?v=ohiKTVVtDJA (visited on 01/03/2016).

Susan L. Star. „The structure of ill-structured problems: boundary objects and heterogeneous distributed problem solving.“ In: Distributed Artificial Intelligence 5 (1989).

Susan L. Star and James R. Griesemer. „Institutional ecology, ‘Translations’ and boundary objects: Amateurs and professionals in Berkeley’s museum of vertebrate zoology, 1907-39.“ In: Social Studies of Science 19.3 (1989), pp. 387–420. issn: 0306-3127.doi: 10.1177/030631289019003001.