

Substrates: Thru the Looking Glass

Gilad Bracha
Technical Fellow, F5

In the near future, multi-modal AIs will interact with us via integrated visual, aural and haptic interfaces, for both input and output. The arena for this interaction is two or three dimensional space - be it a document, a virtual reality room, or (most likely) two or three dimensional overlays over the space around us - augmented reality.

It is imperative that humans should have complete control of these digital spaces. The alternative is to trap us without agency within ever more alluring and immersive spaces, which rather than augment our capabilities, will stultify, stupefy, spy on and manipulate us.

Hence, the spaces must be programmable in a deep way. Everything about the space should be programmable, and the response to our programming should be immediate. In other words, we need a live, self modifying system.

While in the past, we could aspire to programming via powerful yet friendly programming languages, the next phase is programming by interacting with AI. This means interaction thru natural language, but also thru body language, or thru music, or drawing, or sculpting or any modality humans can use to express themselves. Of course, at some point direct neural interfaces will plug into our consciousness directly, but this author is in no hurry to get there.

We will express our desires thru the high level, natural interfaces. The AI may interpret our requests and let us give feedback on the results, or engage in a socratic dialog with us, pointing out contradictions and ambiguities.

The use of AI puts the dream of the [Dynabook](#) [1, 2] in reach for the vast majority of people who cannot program in the traditional way. However, underneath the AI and its rich interfaces, there will be traditional programmed code, for reasons of both precision and efficiency.

The hardware for interacting with these spaces will almost certainly differ from the tablet form of the Dynabook. The UIs will resemble the vision of *Minority Report* [3, 4]. At this writing, the most promising candidate hardware are smart glasses; one goes thru the looking glass, as it were, to reach into digital space.

A *substrate* is a mechanism for creating and managing such spaces. Substrates need to be open for the reasons noted above. This is far from certain, as the pressures for proprietary control of the software (e.g., the AI models and inference engines, the OS for the underlying hardware, such as smart glasses) are very strong.

There are many more detailed requirements for modern substrates and the spaces they support. These include support for interoperability with other software, collaboration and the ability to work offline, persistence - all while maintaining security and privacy. No doubt there are other requirements as well.

In my own work, I am making modest progress toward this vision thru the [Ampleforth](#) [5, 6] live document system, which in turn relies on the [Newspeak](#) [7] programming environment. The system runs on the web on top of Wasm and HTML. This is a mixed blessing, providing access to the web and its vast resources and capabilities at the cost of a heavy burden of working within the limitations of mainstream software.

The latest iteration of that work builds upon [Croquet](#) [8] to support real-time, live collaboration without any special effort by the programmer. This, I believe, is a step toward a system that enables a seamless transition between online and offline work, as first outlined in [Objects as Software Services](#) [9] (which is similar, yet distinct from, [local-first](#) [10]).

The spaces we describe include both programs and state. Experience with Smalltalk and its image model [11, 12] teaches us that we must be able persist the program without the state, as well as snapshotting the complete world of program combined with state. The same issue applies to the spaces we define here.

There is an inherent tension between security and usability. This is all the more acute in a system that provides its users with great power. Newspeak's object-capability support is a promising way to resolve this.

Current experience with Ampleforth is based on two-dimensional documents. As we evolve from a 2D document system to a 3D space, hyperlinks become portals. I expect concepts like transclusion to translate to 3D but have no real evidence for that yet.

How do we integrate AI into this model? In particular, how do we keep AI aware of our spatial and temporal context as we navigate these spaces?

Building, maintaining and above all, regularly using a working system is crucial. Day-to-day usage brings up many needs and concomitant questions. Some examples:

1. How easily can one add mechanisms for commenting on existing documents? Ampleforth's approach to these issues relies heavily on transclusion of documents.
2. How do we manage the permissions on the documents and the comments - e.g., how can we ensure that different users can read but not edit the comments of others? We naturally expect to leverage object capabilities. The details of how transclusion interacts with capabilities are still being worked out.
3. Are spaces of fixed size, or logically unbounded (like [Kansas](#) [13] or [The Humane Interface](#) [14]).
4. Are the advantages of using HTML markup worth the associated awkwardness and complexity.

Obviously, Ampleforth is light years away from this vision. However, the components are coming together. Many have existed for many years; others are just starting to come into their own, but they will be here soon, and for the rest of our lives.

The challenge of realizing this vision with limited resources is daunting, but the alternative is to be stuck using what the mainstream produces. We owe it to ourselves to make the AI Dynabook substrate a reality.

References

1. Alan Kay and Adele Goldberg. *The Dynabook -Past, Present and Future*. In A History of Personal Workstations. ACM Press. 1988.
2. Alan Kay. *A Personal Computer for Children of all Ages*. August 1972.
3. John Underkoffler. *The I/O Bulb and the Luminous Room*. PhD Thesis, MIT, February 1999.
4. John Underkoffler. *Pointing to the Future of UI*. TED Talk. June 2010. <https://www.youtube.com/watch?v=b6YTQJVzwII>
5. Gilad Bracha. *Ampleforth: A Live Literate Editor*. Live 22 workshop. December 2022, Auckland, New Zealand.
6. Gilad Bracha. *DocuApps: Ampleforth Documents as Applications*. Video of presentation at Live 24 workshop. October 2024. Pasadena, California.
7. Gilad Bracha, Peter Ahe, Vassili Bykov, Yaron Kashai, William Maddox and Eliot Miranda. *Modules as Objects in Newspeak*. Proceedings of the 24th European Conference on Object-Oriented Programming. June 2010, Maribor, Slovenia.
8. Yoshiki Oshima, Aran Lunzer, Jenn Evans, Vanessa Freudenberg, Brian Upton and David A. Smith. *An Experiment in Live Collaborative Programming on the Croquet Shared Experience Platform*. <https://dl.acm.org/doi/10.1145/3532512.3535224>. In Programming 22, Companion Proceedings of the 6th International Conference on the Art, Science and Engineering of Programming. December 2022.
9. Gilad Bracha. *Objects as Software Services*. August 2006.
10. Martin Kleppmann, Adam Wiggins, Peter van Hardenberg, and Mark McGranaghan. *Local-first Software: You Own Your Data, in spite of the Cloud*. 2019 ACM SIGPLAN International Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software (Onward!), October 2019, pages 154–178. doi:10.1145/3359591.3359737
11. Adele Goldberg and David Robson. *Smalltalk-80: The Language and Its Implementation*. Addison-Wesley. 1983.
12. Gilad Bracha. *An Image Problem*. [Room 101 Blog](#) post. October 2009.
13. Randall B. Smith, David Ungar and Mario Wolczko. *From Kansas to Oz: Collaborative Debugging When a Shared World Breaks*. CACM, Vol. 40, No. 4. April 1997.
14. Jef Raskin. *The Humane Interface*. ACM Press/Addison-Wesley. March 2000.

AUTHORS: Gilad Bracha

TITLE: Substrates: Thru the Looking Glass

+++++++ REVIEW 2 (Tom Larkworthy) +++++++

The author is very future looking. I share many of their aspirations for AI VR enhanced experiences and I do think this is a possible direction for substrate research. I found the Ampleforth document system unpersuasive, my first interaction broke with "Doubleplus Ungoodness Detected." when clicking "increment". Where can the bugs come from in an isolated system (I assume this broke at some point since publishing from external forces)? The choice of a paid proprietary sync engine for games "croquet" also seems a strange choice. Perhaps the author is correct multiplayer is needed for AI, but I am not so sure and I fear it is an orthogonal and distracting concern.

The underlying programming language "newspeak" is very interesting. It is self-modifiable, I think that is a gem of a substrate. I would like to learn more about the design goals of that language and how it has met, or not met, its goals. I could not find many programs written in newspeak. Is there a useful post mortem there? What were the wins and losses of newspeak over the last 5 years?

I think mimics of existing software on substrates are difficult, such as document editing, as they are naturally over-shadowed by their more mature single purpose counterparts. I think new software needs to offer new value propositions, I think we, as a community, have greater chance of recognition by writing unique applications, like VR, so I hope the author pursues that direction.

I also share their belief that the ideal programming substrate should be hermetic and self bundling. I see many underlying similarities with lopecode, I hope we can find ways to help each other, maybe simply by reviewing each other's progress.

+++++++ REVIEW 3 (Clemens Nylandsted Klokmoose) +++++++

Gilad pushes further on thoughts I too have been having than I have been daring to do. I also believe that we need to break our work out of the 2D world of laptops and embrace the space around us. And we should domesticate AI so we can have it in our space without worrying about our safety, but put it to good use.

"I expect concepts like transclusion to translate to 3D but have no real evidence for that yet." I have been working on this! I am looking very much forward to discussing this more.

I take this position paper as an encouragement for us to discuss spatial substrates as the foundation for digitally augmented spaces to think in.