Appendix G: Why not head-mounted displays?

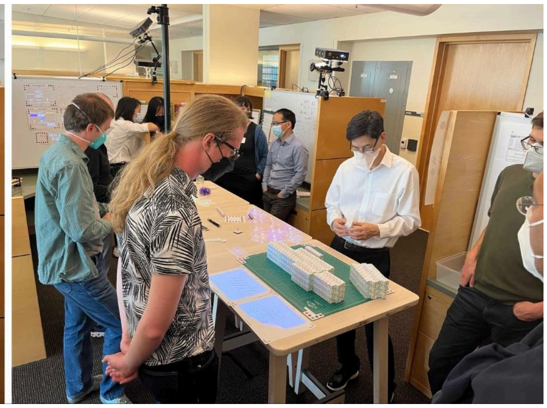
We are often asked why Realtalk doesn't use head-mounted displays like Augmented Reality (AR) or Virtual Reality (VR), or if Realtalk is in competition with these platforms.

The work we are doing today is *not possible* with today's head-mounted displays. And even in the future, head-mounted displays will never meet our requirements of visibility, agency, physicality, and in-person collaboration.

What we do today. We have regularly hosted a hundred or more people, in the same space, engaging with a roomful of Realtalk activities over the course of an evening or entire day.

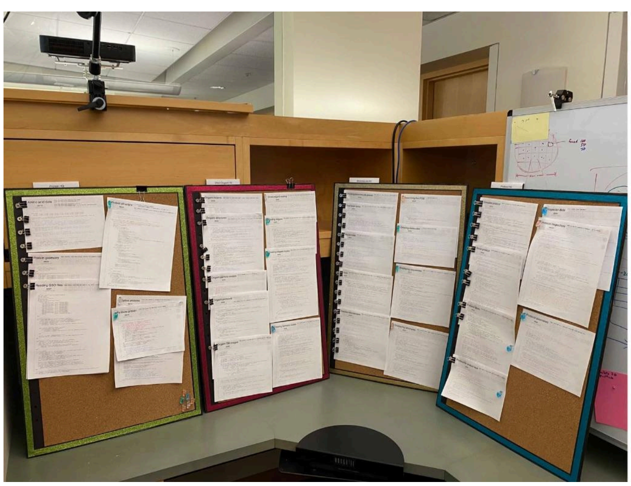
There *does not exist* a head-mounted display that can be worn comfortably and continuously for an entire day, and support a shared activity with a hundred people casually coming and going.

Our prototype environments for biomolecular design were situated in a busy university building, where colleagues and passersby would notice what we were doing and stop to check it out. They could immediately start engaging with our tools using their hands, which would lead into a productive discussion as we jointly explored computational models on the table. If we had been wearing goggles and pantomiming in the air, these potential collaborators would have been repelled, not drawn in.



AR proponents like to believe that “someday” the myriad technical problems will be solved, and all people will wear headsets as unobtrusively as they carry smartphones. We need to emphasize that even the most ideal, technically-perfect, universally-adopted head-mounted display *will not work* as a basis for what we are trying to achieve. This is best seen through our lenses of visibility, agency, physical reality, and in-person collaboration.

Visibility. The posterboards on the left hold the complete implementation of our biomolecular design tools. Those on the right hold the complete implementation of the Realtalk operating system — about 500 pages, the length of a typical nonfiction book.



The complete implementation of a screen-based operating system, if physically realized, would be over 1,000,000 pages, and would weigh 5 tons. And that's just to simulate a simple 2D virtual world with windows and mouse pointers. Simulating a high-fidelity 3D world, and tracking and overlaying that simulation over the real world, brings in far more complexity.

Systems this complex will never be *visible.* It will never be possible to observe the system in its entirety, and understand what all parts of the system are doing at all times. These systems can only ever be black boxes.

Realtalk systems have orders of magnitude less complexity, by radically minimizing the amount of computation required in the first place. Real-world physics don't require a physics engine. Real-world materials render themselves. Many tasks that would have to be explicitly programmed in a virtual world are accomplished in Realtalk with zero lines of code: by arranging objects by hand; physically modifying objects by folding, cutting, and gluing; and using physical tools such as pens, paperclips, binders, and shelves. The parts of a project that actually require computation are thus small and high-leverage, and instead of a “codebase” or “repository”, we find ourselves with a few pages of simple, readable programs.

Readers who haven't experienced this may find it hard to believe, and we don't fully understand the phenomenon ourselves. But it's not theoretical. For years, this has been our daily experience.

Agency. Perhaps our most surprising discovery has been: to maximize agency, *minimize* what the computer knows. Every time we make the computing system *less aware* of what's going on, every time we remove “user interfaces” in favor of noticing simple spatial relations between objects, every time we remove programmed rules in favor of socially-agreed-upon practices, our systems become more flexible and composable, and new dimensions open up for improvisational modification.

Virtual worlds, by contrast, must *maximize* awareness. In screen-based systems and VR, the entire world is simulated. Anything the computer is unaware of, simply doesn't exist. AR has the further challenge of inferring a high-fidelity model of the real world on which to overlay its virtual world. Anything the computer is unaware of will generate a clash between the real world and its virtual overlay.

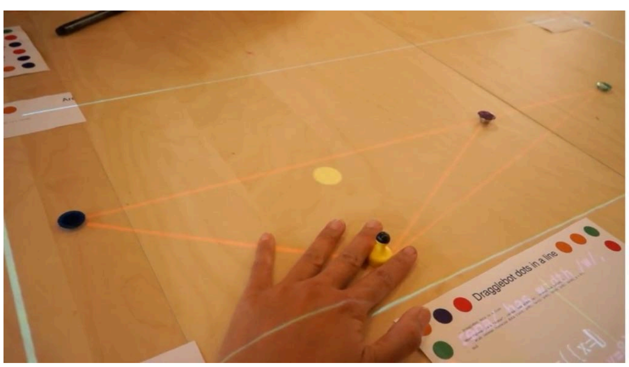
A person wearing a head-mounted display is a guest in someone else's house. The totalizing nature of the simulation, and its overwhelming computational complexity, prevent anything like the constant remixing, reworking, and improvisation that we regularly see in Realtalk.

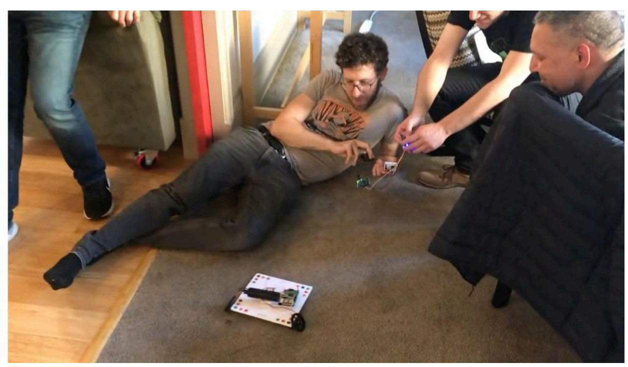
As scientists, our responsibility is to discover the unknown, and this requires the agency to freely explore without externally-imposed boundaries or assumptions, to try alternatives as quickly as possible, to constantly remix, rework, and improvise.

Physical reality. A common misconception is that Realtalk is a “projection-mapping system”, in comparison to an AR “head-tracking system”.

Realtalk is actually a “programming system”, for programming real-world materials. Projection-mapping, using cameras and projectors, is *one* technique for realizing programs on real-world materials. But a Realtalk environment incorporates *any* sensors for recognizing real-world objects, and any actuators for giving these objects physical capabilities.

For example, projectors enable our real-world objects to display text and images. But on our robotic table, real-world objects gain an additional capability — they can *move themselves around.* Our objects with integrated robotics can move around *anywhere.* These robotic capabilities fit seamlessly into Realtalk, because Realtalk is about programming things in the real world, not about overlaying a virtual world via projection-mapping.





In Realtalk, the real world is ground truth. In AR, the real world is a kind of staging area for a virtual simulation.

Physical reality, the real world, is the primary concern of science. As scientists, we want a computing environment that helps us work directly with cells and molecules, test tubes and gels, temperatures and concentrations and optical densities. Realtalk brings computation to the lab, whereas AR requires reconstructing the lab inside the computer.

In-person collaboration. Our goal is an environment that strengthens face-to-face interaction, shared hands-on work, tacit knowledge, mutual context, and generally being present in the same reality. The best rule of thumb that we have found is: *augment the environment, not the individual.*

Anyone can walk into a Realtalk environment and immediately join a computational activity. They don't need to buy a device. They don't need to download anything. They don't need to connect to a network or enter a password or give permissions or worry about the privacy of their personal data.

They learn by watching other people work, because working means moving around physical objects which can be plainly observed. They notice other people's focus, and what they're paying attention to. They can build a shared computational environment around themselves, because programs are real physical things that can be handed around, pinned up, and put in physical places. If a new capability is needed, it is added to the *room*, where it benefits everyone who comes into the space, and strengthens the entire group.

Private devices, such as head-mounted displays, destroy this common ground. A person can't be sure of what other people are even *seeing*. They might be scrolling through private notifications, or interacting with a virtual menu floating in space, or in another app entirely. People can't hand anything to each other, because nothing physically exists. Adding a feature to one's headset does not strengthen the group, but creates further fragmentation.

Our stance here is not due to inexperience. Bret Victor was a central figure in Apple's future-interfaces prototyping group, and directly evaluated and built on a wide range of experimental technologies, some of which were developed into recent Apple products. Luke Iannini spent four years as a full-time VR engineer, creating numerous prototypes for the earliest iterations of the Oculus Rift, and a multi-user live-programmable computing environment for Valve’s prototype room-scale VR systems. Shawn Douglas has evaluated a multitude of VR and AR systems for integrating into his laboratory. These first-hand experiences were essential to formulating the values of communal computing.