

The software stories we reproduce

Musings on one-world-storytelling in technology, and what could be

Addendum to *Re-futuring through fostering stewardship in public spaces*

Sarah Fjelsted Alrøe (201905161)

Throughout my work with this project, the same question has remained on my mind: *What decides what we question and what we don't?* Cautiously steering clear of the rich philosophical implications this claim carries with it, I argue that the answer can be found in our societal failure to recognise that we don't understand our world as classes of objects that follow observable laws, but as collectively retold stories of how it all works. My goal with this essay is not to argue for a certain ontological view, but simply to show the practical implications of views commonly held as universal truths within our field.

Fault

Stories saturate our existence. Stories about how the world works, how we should live and who we are. These stories are not manifestations of some underlying truth or preordained creation story. They are produced and reproduced by people living these stories, mediated through technology, society and other structures, themselves stories.

In certain areas of computer science, we wrongly think ourselves free from stories. Originating somewhere within and remaining closely tied to the natural sciences and mathematics, the story pervades that there is an objective truth out there either waiting for us to observe, postulate and validate or to assert through mathematical proof, blinding us to the existence of the story itself. While we must recognize a progression of the sciences, the ongoing replication crisis and the entire field of quantum mechanics highlight how this story clearly is not universal. Computer science (CS) as a field especially highlights this contradiction. CS produces results, not for the sake of uncovering or describing a preexisting reality, but because the results are meaningful to the society we live in. Yet we apply the same observation-hypothesis-validation or axiom-theorem-proof approaches, to arrive at research stories of a universal science of software postulates instead of the highly situated and interventional work that actually occurs.

In my opinion, the most terrifying issue with this discrepancy is the construing of locally accepted stories as fact. I find no clearer example of this than the oft misfortunate understanding and treatment of minority groups by CS research. In *The misgendering machines*, Os Keyes lays out how research into automatic gender recognition algorithms with few exceptions construes gender as a binary, immutable and physiological property, leading to algorithms that are ignorant of and in worst case actively harmful towards transgender and gender-nonconforming people. Because this research is presented as objective science, Keyes shows how in turn these algorithms are applied without question in HCI research, practically rendering gender minorities non-existent. [3]

In the information technology industry, the same thing happens. Through its scientific appearance, marketing becomes scientific fact. In cryptocurrency communities, the theoretical immutability of a blockchain has been turned into the slogan "Code is law", a shared story of absolute truth written in software, completely ignorant of the social structures that institute and revoke the validity of said blockchains. [9] In artificial intelligence, as Advait Sakar observes, the marketing push towards presenting AI products as actual intelligences with agency turns AI tool use into Human-AI collaboration, at the same time obscuring the gruelling human labor involved in producing the training data for these systems. [7] This reification of market tendencies in turn shapes our language. Because technology-as-product becomes the central *thing* we deal with, people in turn become consumers or the singular, universal *user*, at the same time individualized and central to the story, but entirely abstract and impersonal.

All these stories brought together create a software story of individualized users in their hero's journey meeting software actors to experience growth in their consumer behavior, in an otherwise immutable and deterministic world.

Trace

In his 2018 book [2], Arturo Escobar draws together several authors to describe neoliberal globalisation as a defuturing force, not just continuing colonialist resource extraction from the global south through economic inequality, but enforcing a eurocentric world view under the notion of *development*, centering beliefs in the individual, in an objective reality, in a science that validates this reality, and in the economy. What Escobar describes is an ontological colonialism. Not just an enforcement of certain culture and values, but of a singular understanding of reality. I argue that it is exactly the nature of the stories we currently predominantly tell in our field and industry. In our one-world tradition, we inadvertently (or worse, knowingly) create software that tells one-world stories, that in its structure is incapable of providing for anything besides *what is*. I would argue a recent trend in the industry is the clearest example of this yet: The metaverse. Instead of creating software to be carefully embedded in our existing way of being, the *next new thing* is to build comprehensive new virtual realities. The dream is to build a world so capable of filling out our every want and need, it would replace the realities we actually exist in. A world free from agonism and disagreement, where truth is singular and unquestionable. A world where code *is* law. Reality as defined by software. Utter dystopia.

Refactor

So what do we do with these software stories turned nightmares? We tell new stories. Problem is, it's not that simple. Blockchain, metaverse, AI, sustainability. The industry invents a new set of explicit stories every two to five years, yet they always seem the same. Why? When the fundamental story about infinitely increasing profit stays the same, it matters little whether the explicit story is about freedom through decentralisation, the realisation of snowcrash, AI as new forms of life, or for that matter *sustainable development*. What reproduces these tendencies over and over again is the one world story of neoliberal globalisation, a story so deeply embedded that challenging it directly is near meaningless. But what we can do is recognise these stories for what they are, and start decolonising our own storytelling. Luckily, we are not starting from scratch. For the two cases described in the first section, both Keyes and Sakar propose solid and workable alternative stories of gender and artificial intelligence, and as Escobar observes in indigenous movements, the theory has been there all along. We just need to start implementing it in our daily practice instead of relegating it to being *a novel perspective at alt.CHI*.

I will now spend the remainder of this essay opening issues to suggest new stories:

Reimagining user stories – In 1986, Brenda Laurel argued that we should understand and design the user interface through dramatic theory [5]. Far from unique, this use of explicit storytelling to structure our understanding of software and user, i.e. User stories, is a common tool in the modern UX designers toolset. Laurel's proposal betrays her background in theatre, establishing memesis as a fundamental element of interaction. These assumptions too exist in all other storytelling, albeit less explicit. If we recognize the narrative archetypes pervading our storytelling, we can start to challenge them. I would argue that much storytelling today casts the user in a hero's journey structure. Without questioning it, we are creating software heroes, users as the individualised ego-centric sole agents of the story that at the same time lose all agency in their heroic destinies. I propose we instead start telling stories with different narrative archetypes. As one alternative, I would propose the Japanese literary tendency of *mono no aware*, the empathy for the impermanence of things [6]. Through such an approach we could start imagining software that recognises its mutual impermanence and fickleness with the people and world around it. This would be a perspective that could discard individualisation in favor of a mutual recognition.

New and old stories of the web – Contradictory to the supposed innovation of web3, the base stories of the web have further calcified the last many years. What if we had web tools instead of

web applications and platforms? What if we did not have user profiles bound to singular people? What if we replaced individualized feeds with local notice boards? How could we challenge the structures imposed by the dominant web-frameworks of today? Many of these ideas are not new, and I would love to see lost aspects of the web revisited, like the web-mashups of Yahoo Pipes, or the proliferations of personal websites through services like GeoCities.

Reconsidering software modelling – Many forms of modern software development presupposes a rigid relational model of all the concepts that the software will process. We define databases and APIs that end up structuring the rest of the system and its interaction. This modelling occurs through abstractions of, in the best case explanations from people who will use the system, in the worst case boardroom fantasies. As described by Blackwell et al. this abstraction of people and requirements can lead to the actual people using the system being forced to conform to the abstracted, dehumanised and practically insufficient model of the user [1]. I propose we drop this need for abstraction in favor of malleable software that is not of a model but of a type. Approaches like designing for appropriation [8] may give a promising lead in this direction, though at the risk of forcing computational thinking upon others. For a concrete example, I would argue that the webstrates platform indeed provides an example of data and code co-existing without the need for abstraction [4].

Software beyond the product/service – As part of this decolonialisation of stories, I argue we have to reject the stories of competition, the idealisation of progress and the need to economically justify our work. We need to recognise that there are ways in which software can exist outside the market. Exactly because software is immaterial, and therefore practically free to reproduce, it is a tremendous candidate for existing under other ways of being. My suggestion is to look towards software existing in the relation between people. As an illustration of this idea, I would highlight the browser extension Shinigami Eyes [10]. The extension is a crowdsourced map of online hazards, used by a significant portion of the online transgender community to indicate profiles and places that mean them harm. The project survives and thrives, not on any economic model, but on the goodwill of, and desire for mutual protection within, the community.

Throughout my project, I have actively worked with these ideas. While I can't claim my work as a complete implementation of this thinking, I hope and believe that these ideas visibly permeate my work. The stories I have criticized are so fundamental to our practice and existence that addressing it in a single work is impossible. All we can do, is try again.

References

- [1] Alan F Blackwell, Luke Church, and Thomas Green. 2008. The Abstract is an Enemy: Alternative Perspectives to Computational Thinking. (2008), 10.
- [2] Arturo Escobar. 2018. *Designs for the pluriverse: radical interdependence, autonomy, and the making of worlds*. Duke University Press, Durham.
- [3] Os Keyes. 2018. The Misgendering Machines: Trans/HCI Implications of Automatic Gender Recognition. *Proc. ACM Hum.-Comput. Interact.* 2, CSCW (November 2018), 1–22. DOI:<https://doi.org/10.1145/3274357>
- [4] Clemens N. Klokmoose, James R. Eagan, Siemen Baader, Wendy Mackay, and Michel Beaudouin-Lafon. 2015. Webstrates: Shareable Dynamic Media. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology* (UIST '15). DOI:<https://doi.org/10.1145/2807442.2807446>
- [5] Brenda K. Laurel. 1986. Interface as Mimesis. In *User Centered System Design* (0 ed.), Donald A. Norman and Stephen W. Draper (eds.). CRC Press, 67–86. DOI:<https://doi.org/10.1201/b15703-4>
- [6] Graham Parkes and Adam Loughnane. 2018. Japanese Aesthetics. In *The Stanford Encyclopedia of Philosophy* (Winter 2018), Edward N. Zalta (ed.). Metaphysics Research Lab, Stanford University. Retrieved June 5, 2023 from <https://plato.stanford.edu/archives/win2018/entries/japanese-aesthetics/>
- [7] Advait Sarkar. 2023. Enough With “Human-AI Collaboration.” In *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems*, ACM, Hamburg Germany, 1–8. DOI:<https://doi.org/10.1145/3544549.3582735>
- [8] Pierre Tchounikine. 2017. Designing for Appropriation: A Theoretical Account. *Human-Computer Interaction* 32, 4 (July 2017), 155–195. DOI:<https://doi.org/10.1080/07370024.2016.1203263>
- [9] 2022. *Line Goes Up – The Problem With NFTs*. Retrieved June 4, 2023 from https://www.youtube.com/watch?v=YQ_xWvX1n9g
- [10] Shinigami Eyes: An extension that highlights trans-friendly and anti-trans social network pages. Retrieved June 5, 2023 from <https://shinigami-eyes.github.io/>