Agile software production in computational infrastructures

Donald Jay Bertulfo and Seda Gürses
Delft University of Technology, the Netherlands

In recent years, several publications have highlighted how the tech industry -- particularly Big Tech companies -- is increasingly shaping the priorities and trajectories of computer science research, pushing research topics that benefit their business. We posit that beyond funding and collaborations, this also has something to do with how software production is organized today in computational infrastructures concentrated in the hands of a few companies. To further tease out the interdependent relationship between computer science research and industrial, we propose examining why *software production* looks the way it does today. This talk presents cases from an ongoing research project that uses *genealogy* to examine incremental shifts that prefigured today's software – produced using agile methods in the computational infrastructures controlled by large tech companies. The cases, taken together, reassert computer science, in particular software engineering, as a far-from-apolitical discipline that is continuously shaped and reshaped by broader political and economic forces.

Context

Undone computer science urges scholars to examine questions left unexamined by the discipline. In recent years, several publications have highlighted how the tech industry -- particularly Big Tech companies -- is increasingly shaping the priorities and trajectories of computer science research, pushing research topics that benefit their business (Giziński et al., 2024; Mendonça et al., 2025). The influence of funding in this shaping is well-understood. Beyond funding, we posit that this also has something to do with the computational infrastructures (cloud and mobile/end devices) pushed as software production environments, globally. The influence of contemporary computational infrastructures in shaping computer science research is manifest in, among others, the field's current emphasis on data-centric solutions (e.g., machine learning, generative AI), assumptions about continuous updates as a default for ensuring the security of applications, and privacy enhancing technologies increasingly developed for ensuring confidentiality in cloud environments or on end-devices that report to major service providers –instead of peer to peer solutions.

To more deeply understand how computer research is influenced by these technical developments, we propose examining why *software production* looks the way it does today. Commercial software production today has been heavily shaped by various agile methods, using service architectures and production of services increasingly taking place in computational infrastructures – that is, clouds concentrated in the hands of a few companies, and devices increasingly treated as accessories of those clouds (Gurses & Hoboken, 2018). This agile production of software sits at the heart of current-day datafication and platformization processes that reorganize not only the operations of public institutions and businesses but also value production in today's economy (Gurses & Hoboken, 2018; Thylstrup et al., 2024).

We contend that, far from given, agile software production in computational infrastructures has been prefigured by past attempts to reshape how software is made to serve broader political and economic goals. In this sense, though 'software production' is not often used in computer science parlance, it combines computer science and economic registers to capture not only how professionals 'make' software in a technical sense, but also how their work contributes to the production of (economic) value.

This talk features three cases – agile manufacturing, agile software production and software-as-a-service – that demonstrate how practitioners' ongoing attempts at negotiating the terms of value production vis-à-vis changing political and economic conditions gave rise to visions and practices that prefigured today's agile software production, and with it, the research priorities of computer scientists. It contributes to existing research that reasserts and positions computer science, and software engineering, as far from an apolitical discipline.

The cases

- **Agile manufacturing:** In 1991, the anticipation of post-Cold War military defunding intersected with anxieties over a perceived long-standing decline in U.S. manufacturing competitiveness. This prompted a group of corporate, academic and military actors to come together in crafting a national industrial strategy centered on a projected 'agile manufacturing' future. This multi-actor initiative, facilitated by academics then-fascinated with developments in computing and engineering, gave rise to the document now widely-regarded as the origin of the term 'agile manufacturing' (Nagel & Dove, 1991) – a document that directly prefigured the broader concept of agility at the enterprise level (Goldman et al., 1995), and 'agile' as a specific practice of iterative and incremental software production in close collaboration with the customer. The background conditions that led to the making of the agile manufacturing report, together with the discourse embedded in it, presents a rich case for how 'software' – and, by extension, 'programmability' – was envisioned to revitalize U.S. manufacturing at a time when the industrial use of computers were limited at best. This vision was not only built on technical fantasies, but importantly on the intersecting realities of imperialist military ambitions, technological sovereignty and national economic prosperity at the end of the Cold War.

  Agile software development: In 2001, a group of software practitioners who branded themselves as 'organizational anarchists' wrote the now-famed Agile Manifesto (Beck et al., 2001). The Manifesto succinctly articulated in four values and twelve principles an approach to software production that had been gaining popularity among commercial enterprises during the 1990s. Positioned in contrast with the largely military-backed top-down and sequential software production approach called 'waterfall', agile software development promoted iterative software production with frequent feedback loops with the customer – a

practice that has since been widely adopted in commercial software. Genealogically, agile software development links interestingly with the earlier concept of 'agile manufacturing': unbeknownst to most authors of the Agile Manifesto, they adopted the term 'agile' from the earlier concept of 'agile manufacturing'. However, beyond a mere transmutation of terms, agile software development exemplified the key production tenets (i.e., customer-centricity and change-responsiveness) promoted by its manufacturing predecessor. Our analysis of the background conditions that led to the creation of the Agile Manifesto reveals its genealogy as one that is rooted in varied – and heterogeneous – attempts by *commercial* software developers and software project managers to deliver on software's promise to produce 'value' for business software users, and for the enterprise as a whole during the turbulent business environment of the 1990s. In this sense, the making of the Agile Manifesto represents a fertile case for how broader social, economic and technological forces reshaped the technical practices of software production.

- **Software-as-a-service:** Today, agile software development seems inseparable from software delivered as services, yet there was a time when software was not produced as a service using agile methods. This case examines the rise of software-as-a-service (SaaS) not only as a new technical proposition for software delivery, but also as a software business model premised on producing and deploying software from what is today called 'the cloud'. It does so using Salesforce as an empirical case to reveal the incremental steps a business went through to make software residing in third-party computational infrastructures legible and acceptable to enterprise customers accustomed with purchasing on-premise, packaged software. The early beginnings of SaaS represents a stark contrast today, when most enterprise software is delivered as cloud-based and cloud-produced services.

Conclusion

Our work foregrounds agile software production in computational infrastructures as a current configuration forged through decades of technology development that is fundamentally inseparable from broader political and economic forces. We hope that this would shed light on how computer science research is defined not only through funding and collaborations, but also by a business of computing that vies to establish technical environments that concentrate 'software production' in the hands of a few companies, for the benefit of a few. We further hope our research helps raise critical questions about what forces need to be mobilized to ensure computer science research goes beyond serving narrow interests of the dominant industry players and extends to develop CS for systems that serve broader publics in times of political, economic and environmental challenges we are facing today.

Our research helps bring this pressing issue in solidarity with interdisciplinary legal and social science scholarship that attempts to grapple with power in the digital economy (Kapczynski, 2020; Nieborg & Poell, 2025; van der Vlist & Helmond, 2021). Our particular focus on 'software production'

complements, but not repeats, extant studies linking concepts and ideas found in traditional political economy, neoclassical economics or allied fields to technical objects familiar to computer scientists – such as data, algorithms, platforms and clouds (Amoore, 2018; Narayan, 2022; Poell et al., 2019; Rikap, 2020; Sadowski, 2019; Srnicek, 2017).

Finally, the cases explored in this project have been deliberately chosen and considered parts of a broad, and partial genealogy, of agile software production. In this sense, the project – in its entirety – can be interpreted as extending the rich stock of historical scholarship on software production (Campbell-Kelly, 2004; Ensmenger, 2010; Haigh, 2002; Kelty, 2008; Yost, 2017) which, to our knowledge, has not yet sufficiently explored the practices that prefigured agile software production. The specific political-economic sensibility of this project also responds to the emerging fascination of some computing historians with so-called 'computing capitalisms' (Kennedy & Con Diaz, 2020).

References

Amoore, L. (2018). Cloud geographies: Computing, data, sovereignty. *Progress in Human Geography*, *42*(1), 4–24. https://doi.org/10.1177/0309132516662147

Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., & Jeffries, R. (2001). *Manifesto for agile software development*.

Campbell-Kelly, M. (2004). *From Airline Reservations to Sonic the Hedgehog: A History of the Software Industry*. MIT Press.

Ensmenger, N. L. (2010). *The Computer Boys Take Over: Computers, Programmers, and the Politics of Technical Expertise*. MIT Press.

Giziński, S., Kaczyńska, P., Ruczyński, H., Wiśnios, E., Pieliński, B., Biecek, P., & Sienkiewicz, J. (2024). Big Tech influence over AI research revisited: Memetic analysis of attribution of ideas to affiliation. *Journal of Informetrics*, *18*(4), 101572. https://doi.org/10.1016/j.joi.2024.101572

Goldman, S. L., Nagel, R. N., & Preiss, K. (1995). *Agile Competitors and Virtual Organizations: Strategies for Enriching the Customer*. Van Nostrand Reinhold.

Gurses, S., & Hoboken, J. (2018). *Privacy After the Agile Turn*.

Haigh, T. (2002). Software in the 1960s as concept, service, and product. *IEEE Annals of the History of Computing*, *24*(1), 5–13. https://doi.org/10.1109/85.988574

Kapczynski, A. (2020, March 30). *The Law of Informational Capitalism*. Yale Law Journal. https://yalelawjournal.org/review/the-law-of-informational-capitalism

Kelty, C. M. (2008). *Two Bits: The Cultural Significance of Free Software*. Duke University Press.

Kennedy, D., & Con Diaz, G. (2020). Introduction to Computing Capitalisms. *IEEE Annals of the History of Computing*, *42*(3), 5–10. https://doi.org/10.1109/MAHC.2020.3011884

Mendonça, S., Silva, E., & Damásio, B. (2025). The knowledge base of Big Tech: Research as a source of informational leadership by the dominant US digital platforms. *Telecommunications Policy*, *49*(7), 102908. https://doi.org/10.1016/j.telpol.2025.102908

Nagel, R., & Dove, R. (1991). *21st Century Manufacturing Enterprise Strategy: An Industry-Led View*. Lehigh University. https://books.google.nl/books?id=dSjsn_ECSSsC

Narayan, D. (2022). Platform capitalism and cloud infrastructure: Theorizing a hyper-scalable computing regime. *Environment and Planning A: Economy and Space*, *54*(5), 911–929. https://doi.org/10.1177/0308518X221094028

Nieborg, D. B., & Poell, T. (2025). Analyzing institutional platform power: Evolving relations of dependence in the mobile digital advertising ecosystem. *New Media & Society*, *27*. https://doi.org/10.1177/14614448251314405

Poell, T., Nieborg, D., & van Dijck, J. (2019). Platformisation. *Internet Policy Review*, *8*(4). https://doi.org/10.14763/2019.4.1425

Rikap, C. (2020). Amazon: A story of accumulation through intellectual rentiership and predation. *Competition & Change*, 102452942093241. https://doi.org/10.1177/1024529420932418

Sadowski, J. (2019). When data is capital: Datafication, accumulation, and extraction. *Big Data & Society*, *6*(1), 2053951718820549. https://doi.org/10.1177/2053951718820549

Srnicek, N. (2017). *Platform Capitalism*. Polity Press. https://www.wiley.com/en-us/Platform+Capitalism-p-9781509504862

Thylstrup, N. B., Archer, M., & Steiner, H. (2024). Desiloization and its discontents: The politics of data storage in the age of platformization. *Information, Communication & Society*, *27*(13), 2419–2437. https://doi.org/10.1080/1369118X.2024.2371803

van der Vlist, F. N., & Helmond, A. (2021). How partners mediate platform power: Mapping business and data partnerships in the social media ecosystem. *Big Data & Society*, *8*(1), 20539517211025061. https://doi.org/10.1177/20539517211025061

Yost, J. R. (2017). *Making IT Work: A History of the Computer Services Industry*. https://doi.org/10.7551/mitpress/9375.001.0001