Simple Problems

Evan Hepler-Smith
Princeton University

Editor: Nathan Ensmenger

> "Now, this is real simple."
> —Barack Obama, 26 September 2013

Five days before the launch of the online health insurance marketplace mandated by the 2010 Affordable Care Act on the website Healthcare.gov, US President Barack Obama described to an audience of students and legislators at a Maryland community college how the system would work. It would simplify individuals' relationships with insurers—"Now you're part of a big group plan," he told the students. It would also simplify evaluation and comparison of the products on the market: "It will say clearly what each plan covers, what each plan costs. The price will be right there. It will be fully transparent." The key to this simplicity, the President explained, was Healthcare.gov itself, "a website where you can compare and purchase affordable health insurance plans, side-by-side, the same way you shop for a plane ticket on Kayak, same way you shop for a TV on Amazon."[1]

The terms in which the Healthcare.gov rollout has been described and critiqued illustrate a deep entanglement of ideas about the simplicity and complexity of software and of government in contemporary American public discourse.[2] Politicians, technologists, activists, and academics increasingly invoke software as both a tool for simplifying the relationship between the individual and the state and as a metaphor for the trouble caused by the complexity of legislation and administration.

In 1999, Lawrence Lessig countered visions of an anarchic online utopia with the dictum "Code is Law"; a decade and a half later, to many, law is a matter of code.[3] The history of the multivalent notion of simplicity that underwrites digital metaphors and "solutions" for government is part of the history of computing and political history alike, and it would make an important tool for understanding, deploying, and critiquing simplifying technologies and the rhetoric of simplification today. It is a history that we ought to tell.

The Politics of Complexity

Online systems that cut across the complexity of legislation and bureaucracy involved in filing taxes, renewing a driver's license, or petitioning a public official have become a principal means by which the individual interacts with the state. Code for America, a nonprofit that supports public-private collaborations in this emerging field of "civic tech,"[4] has described its mission as "help[ing] cities create 21st century interfaces to government that are simple, beautiful and easy to use."[5] These have included such digital tools as Adopt-a-Hydrant, which "allows citizens to claim responsibility for shoveling out fire hydrants after heavy snowfall," and Textizen, "a civic dialogue platform that allows residents to connect with a powerful mix of offline outreach and online engagement."[6] Like these systems, Healthcare.gov was to provide a simple digital bridge across a head-spinning mass of regulations, public agencies, and private firms to link an individual with a government-guaranteed service.

Plagued by crashes, errors, and delays, the first few months of operation for the online marketplace were anything but simple. A *Washington Post* feature highlighted the ill effects of attempts to paper over the system's complexity, reporting that a successful dry run on "a simplified demonstration application" prevented the President and his staff from discovering the website's woes in early September. The *Post* also reported that White House officials had forbidden staff working on the federal insurance exchange from sharing planning documents with their counterparts at the state level, fearing that "because the diagrams were complex" they might be used by political opponents to disparage the system.[7] The President, too, sought to frame the website's troubles in terms of technical complexity. At a mid-November press conference, a reporter asked whether the social dynamics of his administration were involved in "how this mess came to be." The President responded by pointing instead to the "very complicated" architecture of the online system.[8]

Johns Hopkins political scientist Steven Teles has proposed this sort of complexity—the complexity of software—as a model for the complexity of the American state. In a widely cited 2012 essay, Teles identifies complexity as the signal "ailment" of contemporary American government, christening the phenomenon "kludgeocracy."[9] Teles borrows the term "kludge" from computer programming, where it indicates a cumbersome patch that temporarily solves a specific problem in a computer program without addressing the structural features of the system that brought it about. He argues that the entrenchment of existing law, coupled with the American public's desire for a state that looks small but acts big, has led to overlapping jurisdictions, pervasive coordination problems, and in general a government that behaves like "a very complicated program, one that is hard to understand and subject to crashes. In other words, Windows."[9] As with poorly designed software, Teles suggests, when layers of

continued on p. 85

Published by the IEEE Computer Society

continued from p. 88

legislation and bureaucracy accumulate, they begin to create more systemic problems than they solve.

This is a compelling metaphor. Like the turn to digital tools for simplifying government, its appeal rests on several assumptions about simplicity that are well worth historicizing:

- That simplicity is a virtue (and complexity a defect) in both software and governments.
- That simplicity is commensurable across these realms.
- That well-built computer systems are a privileged means and model for achieving simplicity.

We might start by asking what it has meant for a digital system to be "simple." Over the second half of the 20th century, computer scientists and software engineers developed formal models of the complexity of computational problems and solutions, focused on mathematical representations of algorithms.[10] In these models, and in the cybernetics-influenced study of complex systems in general, simplicity is figured as a modest degree of complexity or as a property of individual entities and operations that make up the system, in contrast with the emergent complexity of the system as a whole.[11] However, as Nathan Ensmenger has written, software itself was and is heterogeneous, inherently messy, and replete with competing agendas and unintended consequences—in other words, political.[12]

How have the political aspects of software been incorporated into or written out of technical metrics of its complexity? What are the consequences for the mobilization of software as a tool and metaphor in explicitly political settings? The social sciences, too, have formal models of complexity with ties to the study of complex systems.[13] Does a metaphor like kludgeocracy disrupt or reinforce assumptions about simplicity and complexity within political science?

Furthermore, how has simplicity come to be seen as a virtue? There is a saying circulating among software developers: "Junior Engineer—Creates complex solutions to simple problems. Engineer—Creates simple solutions to simple problems. Senior Engineer—Creates simple solutions to complex problems."[14] How has an ability to simplify come to be the sign of talent and experience? How have computer scientists and software engineers balanced or blended simplicity with other computational virtues, such as flexibility or power? How has the technical virtue of simplicity competed with other conceptions of the value of simplicity—Jacksonian democracy, for example—in political discourse?

We might start by asking what it has meant for a digital system to be "simple."

Toward a History of Simplicity

Such questions present a methodological challenge: how do we write the history of simplicity? There is a rich array of models in the history of science and technology for dealing with complex concepts, events, and artifacts and for illuminating the contextual complexity of those that appear simple to a Whiggish eye.[15] Science Studies, too, offers categories for analyzing complex imbrications of the social and the technical—Bruno Latour's hybrids and Donna Haraway's cyborgs, among others.[16] Yet the criteria that have made it possible to perceive one problem or solution as simple are surely as historically specific and in need of explanation as those that make another appear complex.

Historian of science Ted Porter has recently called for more attention to an analogous problem in the history of the natural and social sciences, which he calls "thin description." Porter observes that standardized evidence susceptible to algorithmic analysis and general scientific methods like the randomized clinical trial are themselves the products of particular, richly contextual processes.[17] A similar approach could be useful in uncovering the complex conditions in which certain problems and solutions in computing came to be defined as simple.

"Why isn't the Obamacare system simpler? Politics, mostly," concluded economist Paul Krugman, citing Teles, in an October 2013 column entitled "The Big Kludge."[18] In taking up the metaphor of kludgeocracy, Krugman cast the political phenomena that he criticized—antigovernment ideology and

Historians might investigate how computer systems themselves came to be seen as simple or complex and how this came to be a judgment of virtue.

the structural power of entrenched in terests—as sources of excessive and deleterious technical complexity akin to those manifested in poorly designed computer systems, complexity that then found its way into the actual software and hardware of the health insurance exchange. If only the American state could be rid of "the policy complexity that their [the public's] own ideological incoherence helps to create," as Teles puts it, the simplified operation of the program of government would eliminate the bugs of perversity, corruption, and injustice that vex Americans of all ideological stripes.[19]

The kludgeocracy model suggests that much of the ineffectiveness of government and the challenge of navigating laws and bureaucracy is technical in kind and would be susceptible to a technical solution, if only entrenched structures and interests could be swept away like outdated hardware or legacy code. This is a central assumption of discourse that presents the complexity of the state through digital metaphors and proposes its simplification through the use of digital tools. The assumption is not right or wrong; it must undoubtedly hold in some cases and not in others. By investigating how computer systems themselves came to be seen as simple or complex, and how this came to be a judgment of virtue, historians can shed light on the fitness of such digital tools and metaphors and the problems of public policy that are likely to defy solution or description in this manner.

They will not be alone in doing so. "A lot of focus has been on the website and the technology," the President noted in a November press conference. "But even if we get the hardware and software working exactly the way it's supposed to with relatively minor glitches, what we're discovering is that insurance is complicated to buy."[20]

Acknowledgments

Thanks to Michael Barany, Stephanie Dick, and Nathan Ensmenger for their comments and suggestions.

References and Notes

1. B. Obama, "Remarks by the President on the Affordable Care Act," White House Briefing Room, 26 Sept. 2013; www.whitehouse.gov/the-press-office/2013/09/26/remarks-president-affordable-care-act.
2. The same could be said for other national settings; Estonia's experiments with "e-democracy," in particular, have attracted significant attention.
3. L. Lessig, *Code,* 2nd ed., Lawrence Lessig, 2006, p. 5. Extending Lessig's analogy, Rebecca MacKinnon has argued that users of large online and telecommunications networks ought to hold the firms that administer them to the same standards as they do democratic governments; *Consent of the Networked: The World-Wide Struggle for Internet Freedom,* Basic Books, 2012.
4. The scope of "civic tech" is rather nebulous. One recent survey of the field includes applications that support peer-to-peer sharing, which in some cases allows consumers and service providers to sidestep regulations, fees, and taxes entirely; "Trends in Civic Tech," Knight Foundation, 4 Dec 2013; www.knightfoundation.org/features/civictech.
5. As described on the Code for America webpage, www.codeforamerica.org/about, accessed 11 Dec. 2013.
6. See the "Our Apps and Projects" webpage on the Code for America site, www.codeforamerica.org/apps.
7. A. Goldstein and J. Eilperin, "HealthCare.gov: How Political Fear Was Pitted against Technical Needs," *The Washington Post,* 2 Nov. 2013; www.washingtonpost.com/politics/challenges-have-dogged-obamas-health-plan-since-2010/2013/11/02/453fba42-426b-11e3-a624-41d661b0bb78_story.html. For a complementary account, see E. Austen, I. Lipton, and S. Lafraniere, "Tension and Flaws Before Health Website Crash," *New York Times,* 22 Nov. 2013; www.nytimes.com/2013/11/23/us/politics/tension-and-woes-before-health-website-crash.html.

8. B. Obama, “Statement by the President on the Affordable Care Act,” White House Briefing Room, 14 Nov. 2013, www.whitehouse.gov/the-press-office/2013/11/14/statement-president-affordable-care-act.
9. S. Teles, “Kludgeocracy: The American Way of Policy,” New America Foundation, 10 Dec. 2012, http://newamerica.net/publications/policy/kludgeocracy_the_american_way_of_policy, quotes on p. 1. On the history of mechanical metaphors for government and their relationship to the history of computing, see J. Agar, *The Government Machine: A Revolutionary History of the Computer,* MIT Press, 2003.
10. For a technical overview, see M. Sipser, *Introduction to the Theory of Computation,* 3rd ed., Cengage Learning, 2013.
11. M. Mitchell, *Complexity: A Guided Tour,* Oxford Univ. Press, 2009.
12. N. Ensmenger, *The Computer Boys Take Over: Computers, Programmers, and the Politics of Technical Expertise,* MIT Press, 2010, pp. 7–11.
13. P. Erickson, et al., *How Reason Almost Lost Its Mind: The Strange Career of Cold War Rationality,* Univ. of Chicago Press, 2013.
14. See for example S. Hanselman, “The Myth of the Rockstar Programmer,” 29 Sept. 2013, www.hanselman.com/blog/TheMythOfTheRockstarProgrammer.aspx.
15. See for example T. Hughes, *Networks of Power: Electrification in Western Society, 1880–1930,* Johns Hopkins Univ. Press, 1983, and D. Mackenzie, *Inventing Accuracy: A Historical Sociology of Nuclear Missile Guidance,* MIT Press, 1990, respectively.
16. B. Latour *We Have Never Been Modern,* trans. C. Porter, Harvard Univ. Press, 1993; D. Haraway, “A Cyborg Manifesto: Science, Technology, and Socialist-Feminism in the Late Twentieth Century,” *Simians, Cyborgs, and Women: The Reinvention of Nature,* Routledge, 1991, pp. 149–181.
17. T. Porter, “Thin Description: Surface and Depth in Science and Science Studies,” *Osiris,* vol. 27, no. 1, 2012, pp. 209–226.
18. P. Krugman, “The Big Kludge,” *New York Times,* 27 Oct. 2013, www.nytimes.com/2013/10/28/opinion/krugman-the-big-kludge.html.
19. Teles, “Kludgeocracy,”p. 9.
20. Obama, “Statement by the President on the Affordable Care Act.”

Evan Hepler-Smith *is a PhD student at Princeton University. Contact him at ehepler@princeton.edu.*

cn *Selected CS articles and columns are also available for free at http://ComputingNow.computer.org.*