The Means of Production

David Alan Grier, George Washington University

Technical leadership remains one of the great problems of software production.

In the study of software development, Ralph Flanders is a minor historical figure who’s connected to agile methods only through a long chain with many rusted and incomplete links. Generally known as a business leader and politician, his moment of glory came in 1954 when he introduced the motion into the US Senate that would censure his colleague Senator Joe McCarthy. Yet, early in his career, after an apprenticeship as an industrial engineer, he identified several key problems that hindered industrial production, which are the same problems that hinder software development.

The connection between industrial engineering and software engineering is strong. Early software developers, including John von Neumann’s colleague Herman Goldstine, shaped their early notions of software with concepts from industrial production. The word “program” came from the term that described a plan for factory operation. Flow charts came from the diagrams that traced the movement of material through a plant. Sequencing originally referred to the steps of factory operation. The terms “program quality” and the “software life cycle” were borrowed from the work to develop production plans that would create uniform goods.

Flanders did most of his engineering work in the 1920s, when Henry Ford was the dominant industrial engineer. Ford argued for production methods that placed workers under strict production control: each worker had one place on the production line, did one job repeatedly, and learned how to do that job efficiently. Flanders argued that this approach was ill-suited to the production of many products. He claimed that the most challenging problems of manufacturing were not inefficient workers but instead manufacturing overhead. Flanders wrote that many plants saw only “a moderate increase in output” when they switched to Ford’s methods, and yet these plants saw a great increase in overhead, particularly in “the number of foremen and in the amount of clerical labor they were called upon to do.”

Flanders made recommendations that are familiar to anyone who has followed recent trends in lean and agile methodologies. He urged manufacturers to organize production by product, not by process, and to design their product to be as simple as possible. He suggested that factories use small teams and encourage team members to develop a wide and comprehensive set of skills. Finally, he recommended that groups work as independently as possible so that the group leader could supervise the work rather than attend to meetings, memos, and reports. “The foreman is bounded only by his production orders and his schedule,” Flanders stated; within “these limits he is king of his territory.”

Flanders appears to have had limited influence as an engineer, though his ideas might have been thwarted first by the reduced production of the Great Depression and second by the overwhelming demands of World War II. However, his concern for technical leadership echoed throughout the engineering literature: it was picked up by management consultant Peter Drucker in the 1950s, Russian engineer S.P. Mitrafanov in the 1960s, and software engineer Frederick P. Brooks in the 1970s. A “major part of the cost [of software] is communication and correcting the ill effects of miscommunication,” Brooks wrote in his seminal work on software production. Hence, he concluded, computer software needs to be built by “as few minds as possible.”

Technical leadership remains one of the great problems of software production, for it requires a different set of skills for conventional mid-level management, skills that handle the tasks of planning, allocating, and coordinating assets. Technical leaders need to understand the details of their product, teach key skills to their workers, and assess the output of their team. Excessive communication, coordination, and reporting reduce the efficiency of these leaders—often fatally so.

We should care about Flanders because he cared about technical leadership long before it was fashionable to have such concerns. Like many who followed him, he recognized that such leadership was difficult and that we could make it easier only by finding better ways of organizing the means of production. 

David Alan Grier is an associate professor at George Washington University. Contact him at grier@gwu.edu.