Above the Line,

People working above the line of representation continuously build and refresh their models of what lies below the line. That activity is critical to the resilience of Internet-facing systems and the principal source of adaptive capacity.

Below the Line

RICHARD I. COOK, M.D.

THE RESILIENCE OF INTERNET-FACING SYSTEMS RELIES ON WHAT IS ABOVE THE LINE OF REPRESENTATION.

Imagine that all the people involved in keeping your web-based enterprise up and running suddenly stopped working. How long would that system continue to function as intended? Almost everyone recognizes that the "care and feeding" of enterprise software systems requires more or less constant attention. Problems that require intervention crop up regularly—several times a week for many enterprises; for others, several times a day.

Publicly, companies usually describe these events as sporadic and minor—systemically equivalent to a cold or flu that is easily treated at home or with a doctor's office visit. Even a cursory look inside, however, shows a situation more like an intensive care unit: continuous monitoring, elaborate struggles to manage related resources, and many interventions by teams of around-the-clock experts working in shifts. Far from being hale and hearty, these

are brittle and often quite fragile assemblies that totter along only because they are surrounded by people who understand how they work, how they fail, what can happen, and what to do about it.

WHAT'S GOING ON?

The intimate, ongoing relationship between tech software/hardware components and the people who make, modify, and repair them is at once remarkable and frustrating. The exceptional reach and capacity of Internet-based enterprises results from indissolubly linking humans and machines into a continuously changing, nondeterministic, fully distributed system.

Their general and specific knowledge of how and why those bits are assembled as they are gives these humans the capacity to build, maintain, and extend enterprise technology. Those bits continuously change, creating an absolute requirement to adjust and refresh knowledge, expectations, and plans. Keeping pace with this change is a daunting task, but it is possible—just—for several reasons:

1. The people who intervene in failure are often the same people who built the stuff in the first place. The diagnosticians and repairers are frequently the same people who designed, wrote, debugged, and installed the very software and hardware that are now failing. They participated in the intricacies, dependencies, and assumptions that produced and arranged these artifacts. Even when they did not, they often have worked and interacted with others and have learned along the way who contributed and who is expert in those areas. This sets

Past performance is no guarantee of future returns.

the community apart from other operator communities (e.g., pilots, nurses).

2. The people who intervene have unprecedented access to the internals of the assemblies. The fixers can look at source code, interrogate processes, and view statistical summaries of activities in near realtime. In no other domain is there so much detail available for troubleshooting problems. Admittedly, the huge volume of accessible material is a challenge as well: It can be difficult to find a meaningful thread of cause and effect and to trace that thread through to its sources. Here again the collective is often the critical resource—someone knows where and how things are connected and dependent so that the work of addressing an incident in progress often includes the work of figuring out what is germane and whose expertise matches the pattern.

3. The continuing failures constantly redirect attention to the places where their understandings are incomplete or inaccurate. There is an ongoing stream of anomalies that demand attention. The resulting engagement produces insight into the fragility, limitations, and perversities that matter at the moment. Anomalies are pointers to those areas where problems manifest, what Beth Long calls "the explody bits." These are also areas where further exploration is likely to be rewarding. This is valuable information, especially because continuous change moves the locus of failure. This is also why longitudinal collections of incidents so rarely prove useful: Past performance is no guarantee of future returns.

4. There is a distinct community of practice with its own ethos. The people who do this work form what Jean Lave

and Etienne Wenger call a community of practice.[3] This is a tangled network characterized by communications and processes that simultaneously share knowledge, distribute responsibility, and provoke actions. The network has some remarkable features. New people are joining all the time, and their induction into the community leads its members to revisit old ground. Because so much learning takes place on the job and in real rather than simulated settings, it has qualities of a guild. Because the people involved change jobs frequently, the guild extends over time and across corporate boundaries. This produces diffusion of expertise across the industry and simultaneously creates a relationship mesh that bridges corporate boundaries.

The barriers to entry into this network are low. There is not yet a formal process of training nor certification of authority found in other domains (e.g., medicine). This has promoted rapid growth of the community while also creating uncertainty that manifests in hiring practices (e.g., code-writing exercises).

This community of practice appears to have a distinct ethos that puts great emphasis on keeping the system working and defending it against failures, damage, or disruption. The community values both technical expertise and the capacity to function under stress; membership in the community depends on having successfully weathered difficult and demanding situations. Similarly, the collective nature of work during threatening events encourages both cooperation and support. As Lave and Wenger observed for other communities of practice, mastery here is gained via "legitimate peripheral participation."

5. The work is demanding and has remained challenging

over time. Keeping the enterprise going and growing requires the expertise and dedication that this group provides. Although technology enthusiasts have predicted a diminishing role for people in the system, there is no sign of this happening. The intervals between breakdowns are so short and the measures required to remedy faults so varied that only a concerted and energetic effort to replenish the network and refresh its knowledge has any chance of success.

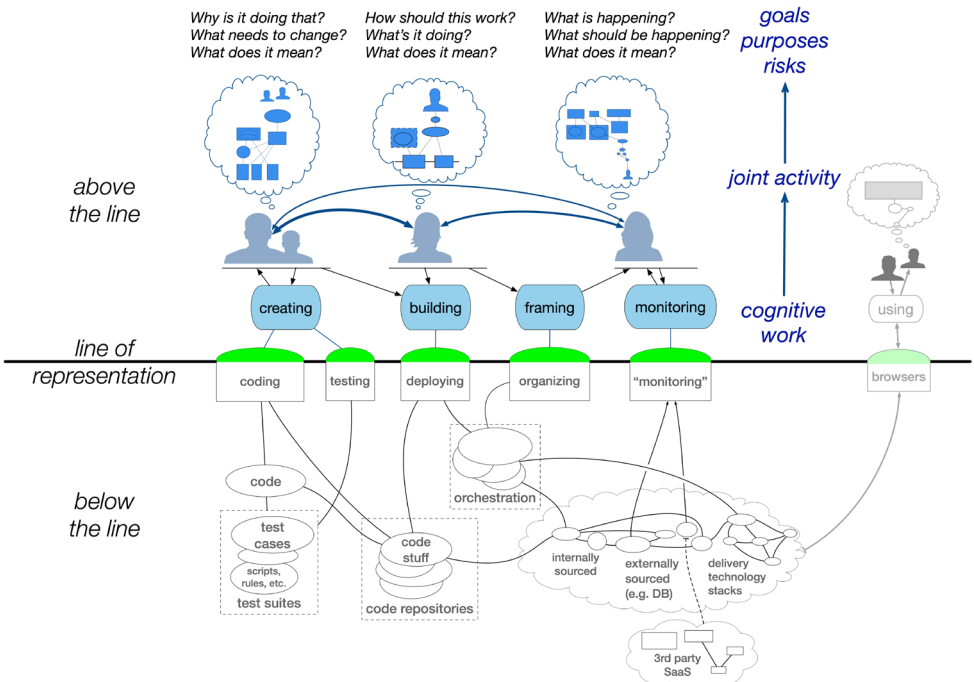
THE LINE OF REPRESENTATION

All these features are simultaneously products of the environment and enablers of it. They have emerged in large part because the technical artifacts are evolving quickly, but moreso because the artifacts cannot be observed or manipulated directly. Computing is detectable only via representations synthesized to show its passing. Similarly, it can be manipulated only via representations.

Figure 1 shows an Internet-facing system. The horizontal line comprises all the representations available to people working above that line, including all the displays, screens and other output devices, and keyboards and other input devices. Below this line lie the technical artifacts: code libraries, IDEs, test suites, compilers, CI/CD (continuous integration/continuous delivery) pipeline components, and the computational capacity itself including technology stacks and services. Above the line of representation are the people, organizations, and processes that shape, direct, and restore the technical artifacts that lie below that line.

People who work above the line routinely describe what is below the line using concrete, realistic language.

FIGURE 1: **ABOVE AND BELOW THE LINE OF REPRESENTATION**



Yet, remarkably, *nothing* below the line can be seen or acted upon directly. The displays, keyboards, and mice that constitute the line of representation are the *only* tangible evidence that anything at all lies below the line.

All understandings of what lies below the line are *constructed* in the sense proposed by Bruno Latour and Steve Woolgar.[2] What we "know"—what we *can* know—about what lies below the line depends on inferences made from representations that appear on the screens and

displays. These inferences draw on our mental models—those that have been developed and refined over years, then modified, updated, refined, and focused by recent events. Our understandings of how things work, what will happen, what *can* happen, what avenues are open, and where hazards lie are contained in these models.

IMPLICATIONS

It will be immediately apparent that no individual mental model can ever be comprehensive. The scope and rate of change assure that any complete model will be stale and that any fresh model will be incomplete. David Woods said this clearly in what is known as Woods' theorem:[4]

As the complexity of a system increases, the accuracy of any single agent's own model of that system decreases rapidly.

1. This is a complex system; it is always changing. The composition and arrangement of the components are such that the system's behavior is nondeterministic. Continuous and often substantial change is going on both above and below the line. There is no way to capture its state nor to reproduce a given state. All models of the system are approximations. It is impossible to anticipate all the ways that it might break down or defend against all eventualities.

The level of complexity below and above the line is similar. As the complexity below the line has increased, so too has the complexity above the line.

2. Collaboration is necessary; collaboration is routine. Many episodic activities—especially troubleshooting and repair beyond handling of minor anomalies—cannot be accomplished by a single person and require collaboration.

Although occasionally the demands of an event may be well matched to the knowledge and capacity of the first person who encounters it, work on most incidents is likely to require joint action by several (or many) people. These events test the capacity to combine, test, and revise mental models. This can be seen to play out in the incident dialog and the after-incident review.

3. Coordinating collaborative efforts is challenging. The job of bringing expertise to bear and coordinating the application of that expertise is nontrivial and often undertaken under severe time and consequence pressure. A burgeoning field of interest is the application of various methods to identify and engage people in problem solving, to generate productive, parallel threads of action, to bring these threads back together, and to evaluate and make decisions. Many organizations are developing support tools, managerial processes, and training to address this need. In particular, controlling the costs of coordination of these parallel and joint activities is a continuing challenge.

For some events troubleshooting and repair are highly localized below and above the line. When there is a one-to-one mapping from a below-the-line component to an above-the-line individual or team, the work of coordination can be small. For other events the troubleshooting and repair can be arduous because the manifestations of the anomaly are far from its sources—so far, in fact, that it is unclear whose knowledge could be useful.

These events are often quite different from those in domains where roles and functions are relatively well defined and task assignment is a primary concern. Coordinating collaborative problem solving in critical

digital services is the subject of intense investigation and the target of many methods and tools, yet it remains a knotty problem.

4. Similar faults and failures can occur above and below the line. The reverberation across the line of representation tends to shape the structure below the line (particularly the functional boundaries) to be like that above the line, and vice versa. Because structure and function above the line parallel structure and function below, parallels can be expected in the forms of dysfunction that can occur. Both are distributed systems. This suggests that specific below the line failure forms (e.g., susceptibility to partition, CAP [consistency, availability, partition tolerance], or even the potential for saturation or cascading failure) will also be found in some form above the line.

5. It's one system, not two. The line of representation appears to be a convenient boundary separating two "systems," a technical one below the line and a human one above it. Reciprocal cause and effect above and below make that view untenable. People are constantly interacting with technologies below the line; they build, modify, direct, and respond to them. But these technologies affect those people in myriad ways, and experience with the technologies produces changes above the line. These interactions weld what is above the line to what is below it. There are not two systems separated by a representational barrier; there is only one system.

A similar argument developed around human-computer interaction in the 1970s. Efforts to treat the computer and the human operator as separate and independent entities broke down and were replaced by a description of human and

computer as a "system." Large-scale distributed computing and the similarly distributed approaches to programming and operations are replicating this experience on a larger scale.

INCIDENTS OCCUR *ABOVE* THE LINE

Incidents are a "set of activities, bounded in time, that are related to an undesirable system behavior."[1] The decision to describe some set of activities as an incident is a judgment made by people above the line. Thus, an incident begins when someone says that it has begun and ends when someone says it has ended. Like the understanding of what lies below the line, incidents are *constructed*.

CONCLUSION

Knowledge and understanding of below-the-line structure and function are continuously in flux. Near-constant effort is required to calibrate and refresh the understanding of the workings, dependencies, limitations, and capabilities of what is present there. In this dynamic situation no individual or group can ever know the system state. Instead, individuals and groups must

Related articles

Continuous Delivery Sounds Great, but Will It Work Here?
It's not magic, it just requires continuous, daily improvement at all levels.
Jez Humble
https://queue.acm.org/detail.cfm?id=3190610

A Decade of OS Access-control Extensibility
Open source security foundations for mobile and embedded devices
Robert N. M. Watson
https://queue.acm.org/detail.cfm?id=2430732

The Network's NEW Role
Application-oriented networks can help bridge the gap between enterprises.
Taf Anthias and Krishna Sankar
https://queue.acm.org/detail.cfm?id=1142069

be content with partial, fragmented mental models that require more or less constant updating and adjustment if they are to be useful.

References

1. Allspaw, J., Cook, R.I. 2018. SRE cognitive work. In *Seeking SRE: Conversations About Running Production Systems at Scale*, ed. D. Blank-Edelman. O'Reilly Media, 441-465.
2. Latour, B., Woolgar, S. 1979. *Laboratory Life: The Construction of Scientific Facts.* Beverly Hills: Sage Publications.
3. Lave, J., Wenger, E. 1991. *Situated Learning: Legitimate Peripheral Participation.* Cambridge: Cambridge University Press.
4. Woods, D.D. 2017. *Stella: Report from the SNAFUcatchers Workshop on Coping with Complexity.* The Ohio State University; https://snafucatchers.github.io/.

Richard I. Cook, M.D. *is an internationally recognized expert in safety, accidents, and resilience in complex systems. His 30 years of experience includes work in medicine, transportation, manufacturing, and information technology. His most often-cited publications are "How Complex Systems Fail" and "Going Solid: A Model of System Dynamics and Consequences for Patient Safety." He holds appointments at The Ohio State University and is a principal in Adaptive Capacity Labs. He lives in Chicago.*

CONTENTS