A gray-haired scientist stands in front of the screen at the head of the room. Projected behind him is a colorful spreadsheet, the boxes in its rows and columns luminescent in shades of yellow, light green, orange, red, and purple. Forty of his colleagues are in the room, busily asking questions, referring to individual boxes, requesting changes. He dutifully edits the document as requested in real time so all can see. Their glances mutually entrained on the projected Excel spreadsheet, this group is hard at work trying to decide where their spacecraft should travel for the next seven years.

STS scholars have explored the politics and infrastructural assumptions of bespoke digital tools in the sciences, such as specialist databases and convergent cyberinfrastructures (Ribes and Bowker 2008; Lee et al. 2006). But STS studies of the off-the-shelf software suites are rare, despite the near ubiquity of smartphone apps, productivity suites, and social media platforms like Facebook or LinkedIn in the laboratories and communities we study (Vertesi 2014; Gillespie 2010). Technologies of this type are highly blackboxed for users, pre-prepared for particular conditions of use by the corporations that produce and update them. Yet they also facilitate a variety of usage patterns in order to secure the broadest possible user base. Hence the analysis of these commercial and widely distributed tools in specialist use contexts demands a vocabulary that makes sense of the technology’s varied local contexts of meaning making. Such an analytical vocabulary should take into account existing STS tools for understanding commercial/user relationships, such as relevant social groups (Pinch and Bijker 1987) or market lock-in (Schwartz Cowan 1985), as well as frameworks for understanding varieties of usage contexts such as interpretive flexibility (Pinch and Bijker 1987) or de-scription (Akrich 1992). It must also note that, in mass commercial software, radical shifts in the understanding, design, or functioning of the artifact are much more limited than the framings of actor-network theory or the social construction of technology might imply, even if the social and networked constraints upon immediate technical use are relatively flexible.

Commercial software suites like Microsoft Office are an excellent site for this type of analysis given their extensive use across a variety of sectors. Conceived
with a range of use cases and scenarios in mind, such systems are relatively open to situated interpretation. Yet they are used by such a broad cross-section of the population that this delimits the ability for specific relevant social groups to demand sweeping changes in product design. As such, one might suppose that there are nearly as many ways of using Excel or PowerPoint as there are users: but this is not always the case.

In this essay, I show how the interactional norms that guide usage of such suites are not due to individual perception, widespread usage patterns, or particular “affordances” available for use. Instead, interactional norms are enrolled in the production and reproduction of the local organization and its membership, in practices that make available different opportunities for software interaction. Since the methods of working that I observed empirically in my field site required using the software suites in specific ways—not necessarily the primary use envisioned by their designers—I offer a way to address how the specific properties of the commercial software suite come to matter for its organizational use. I do so by inverting the question, asking instead how organizational usage settings come to matter for making software properties visible and accountable. This shifts our attention from taxonomizing built-in material properties or built-in “affordances” toward articulating software accomplishments and their deployment in the field.

**Digital Properties and Affordance Theory**

At stake in my account is how STS might approach the study of software suites in conversation with the concerns of organizational studies on the one hand, and communications and media technologies on the other. The former community has developed a notion of “sociomateriality” that draws on STS theories of hybridity (i.e., Latour 1991; Haraway 1991; Barad 2003) to describe the material and semiotic aspects of software in context (Orlikowski 2010; Leonardi 2011). In the latter community, the language of “affordances” aims to bridge the gap between social and technical determinisms. New media scholars frequently deploy this terminology, imported from Gibsonian psychology through human-computer interaction (Gibson 1979; Norman 2002), to delineate how material objects and digital systems alike both enable and constrain particular forms of social action.

In the classic sense, affordances are assumed to be relational: that is, only visible or perceptible to particular users in contexts of use. Norman’s use of the term relies on the psychological qualities of the encounter between a user and an object with an eye toward design: how to make the usage properties of an object immediately perceptible to a novice user. This relational perspective offers a useful middle ground between object and subject, but it steers clear of familiar STS analytical perspectives on object hybridity and situated encounters. From an STS perspective, this runs the risk of neglecting the networked, political, and ontological work-in-the-world that makes such capabilities available for action in the first place. Even concepts such as “scripts” (Akrich 1992) do not taxonomize baked-in properties of technical objects, but instead seek to reveal the networked conditions that make particular use cases possible. Still, deployments of “affordances” in contemporary studies of digital media frequently elide those elements of a technology that are “found” in the object and guide its usage, and those elements that are explicitly designed in. In such work, material constraints are frequently considered to be “in” the object and available to perception, application, or subversion in various
use contexts, whether these were designed in at the level of code or the result of the natural properties of digital materials (boyd 2010; Faraj and Azad 2012; Ilten 2015; Majchrzak et al. 2013; Neff et al. 2012).

Affordance theory has clear merits in the design sphere, where purposefully including certain capacities for action—and restricting others—can be an empowering way to approach wicked technical problems. But when it comes to analyzing digital artifacts, platforms, and tools in their sites of use, affordance theory renders users passive with respect to receiving, deploying, or reacting to elements that are simply “baked in” to the technology, ready-made for the perceiving. Further, it makes an error in post hoc propter hoc reasoning by assuming that properties located by actors in contexts of use must somehow have been there prior to their use case, thereby making them perspicuous for action in the first place. Finally, it assumes that actors’ categories that describe material conditions are in fact analytical categories: real, hard constraints in the world. In naturalizing these emic categories, affordance theory imbues objects with political qualities that reproduce instead of question the social relations at hand.

Assuming that “affordances” are the explanandum, not the explanans, I seek to reframe the conversation in terms more familiar to the STS scholar. I turn to microsociological approaches (along with Sharrock and Coulter 1998) to show how any such material “constraints” or properties are emergent, constructed, and made perspicuous at different times as members deploy quotidian software suites to participate in organizational activity. I argue, therefore, for shifting our analytical language from one of software affordances to software accomplishments (drawing on Garfinkel 1967). I mean accomplishments in two ways. Here, I address the user’s situated accomplishment in producing or making perspicuous qualities of use relevant to their problem at hand. There is, of course, a parallel, no less situated accomplishment on behalf of the designer(s) that make such qualities appear to be natural, psychological, or decontextualized, and otherwise devoid of politics: this is beyond the scope of this essay. Instead, my aim for this chapter is to chart an analytical course for STS scholars that eschews post hoc taxonomies and assumptions about the material or political conditions of software usage, while at the same time remaining attuned to the creative, flexible, and sometimes unexpected things that actors do with such technologies on a regular basis.

To do so, I describe the use of popular Microsoft Office Suite software as it is deployed on two NASA spacecraft teams, each with different ways of collaborating and organizing work among their members. I first describe how the practices of such software use are both unusual with respect to the software’s typical patterns of use, yet at the same time highly organizationally standardized. I then turn to the challenges that such practices pose to existing theories of software materialities. Finally, I demonstrate an alternative approach that emphasizes users’ situated accomplishments to offer novel insights into the constructed-yet-obdurate nature of such software tools in practice.

Workplace Software in Space

The two missions arrived at their respective planets in the early years of the 21st century, carrying with them a suite of instruments to make observations in the different planetary environments. One is an orbiter, sweeping through a planetary system on a charted course over many years. As the spacecraft passes by moons
and other features in the planetary environment it captures rich streams of data through 12 different instruments about the planet’s physical environment, its moons’ weathering and geological activity, and any observed changes over time. The other robots are landed, roving vehicles. They conduct scientific investigations on a planet by driving to new locations, observing rocks, dust, and atmospheric conditions, and protecting the robot overnight from the cold. At the time of my fieldwork there were almost three times as many mission personnel working on the orbiter as there were working on the ground-based team.

Although the missions enroll many of the same scientific and technical personnel and the same institutions, they are organized and operated quite differently. The orbiter mission coordinates its collaboration deploying a matrix organization. This features a bureaucratic hierarchy (Weber 1968) composed of 12 distinct instrument teams. A group of coinvestigators is selected along with team leaders as core team members, with affiliated researchers and students following in descending order. Additionally, scientists from across these teams gather in thematic groups responsible for planning segments of spacecraft activity depending on its location in the planetary system and when it flies past targets of interest such as various moons. The leaders of each instrument team and each thematic group meet as an executive committee to make critical decisions about the mission. The ground-based robot collaboration, by contrast, is structured laterally under a single principal investigator as a charismatic collective (Shils 1965; Weber 1968). Members describe themselves as a single team of scientists with many different skills and backgrounds, each of whom may request and collectively use observations from any robotic instrument and all of whom make decisions via consensus. While scientists have different official designations (PI, team leaders, co-investigators, graduate students), in practice there is no distinction between these roles in the daily planning practices of the collaboration.

Planning cycles for the matrix team’s orbiter take several weeks and many rounds of interaction between scientists, science planners, and representatives of various engineering specialties before uploading to the spacecraft. Planning for the collective’s ground-based mission takes place within a single day for the next day’s events, requiring the construction of agreement on a rapid and highly compressed timescale. To manage these planning activities and their temporal rhythms, each group prefers different software with different visual metaphors and infrastructural capacities for capturing, representing, and collectively enacting various aspects of their work (Mazmanian et al. 2014). Certainly each mission has bespoke software systems that they use for the purposes of entering and managing observational requests (see Cohn’s contribution to this volume). I focus here instead on the process of “mutual entrainment” (Collins 2004) in the teams’ daily or weekly planning meetings. These center around shared PowerPoint presentations on the ground mission and shared Excel spreadsheets on the orbiter.

Attention to screens at these meetings is paramount. Members of each collaboration keep their software application of choice open on their desktop continuously, sometimes with several files loaded at once. They then use these spreadsheets or slide decks as the focus of their collective attention and the primary site of their decision-making work. As the postsocial milieu in which organizational interactions take place (Knorr Cetina and Bruegger 2002), these digital presentations serve to ground the conversation, guide the planning process, and produce spacecraft activity. In this way, plans are “enacted in practice” (Ziewitz 2011) through a continuous achievement of group-and-spacecraft coordination, producing activi-
ties to be implemented in space and documenting team agreement. Yet even as such software grounds the enactment of the planning process, each group departs from the software’s built-in assumptions and templates, preferring to use their software in unusual ways that are consistent with the organizational orientation of each group. As such, they raise analytical possibilities for STS practitioners studying software in organizational and scientific contexts.

Roving with PowerPoint

Many organizational scholars have examined how PowerPoint presentations are both reflective of and constitutive of organizational cultures. Joanne Yates, Wanda Orlikowski, and Huburt Knobloch recommend analyzing PowerPoint presentations as a “genre” of business communication that plays a structurational role at the firm (Yates and Orlikowski 2007; Knoblauch 2013), while Sarah Kaplan focuses on the presentation in ritual meetings also play a structurational and material-discursive role by means of which team members establish knowledge production priorities and (re)produce their lateral social order. But they do so by departing from the standard, built-in templates for the PowerPoint presentation used across the business world with a long history in business communication (see Robles-Anderson and Svensson 2016; Yates and Orlikowski 2007).

Even visualization expert Edward Tufte (n.d.) might be surprised by the PowerPoint slides on the collectivist team. Avoiding the templates that feature a title with bullet points, collaboration members import large images into their slide decks, frequently taking up the entire screen. These images are typically taken by the robot itself and marked up by a team member with annotations to indicate important features around the robot. This requires laboriously maneuvering text boxes, open circles, and arrows (not template bullet points) on screen until they sit over top of the image in just the right spot to direct attention to a feature on the planet’s terrain. In this way, team members have their own slide templates that they adhere to day after day, frequently copying the prior day’s presentation and updating its content for today’s meeting.

I observed this style of PowerPoint usage at every mission team meeting over my two years of ethnographic immersion with the team. At each planning meeting, a daily ritual, the scientist in charge of the meeting used the transition between slides to pace conversation and get everyone on the same page, as well as to impart information. The opening slide always featured a full-screen image taken from the robot’s current location with the date (both on the planet and on Earth) and the primary meeting-related roles for the day. The following slides set the scene for the immediate planning context with a series of images taken from the ground or from orbit, annotated with elements of the robot’s current position and complemented with a list of activities to accomplish at or near this location. The rhythm of the meeting is established with, “Next slide . . . next slide” (figure 1).

Annotations on the slides such as colored dots, lines, and text might indicate prospective plans for investigation or identify areas around the robot; they might also cross-correlate elements visible from different vantage points or propose how to plan a drive (figures 2 and 3). Such slides provide an evolving snapshot of the team’s planning process at a particular point in time. They also frequently carry
forward from one meeting to the next, grounding collaboration members in prior decisions yet without constraining them too tightly to one course of action over another.

Annotation techniques are not impossible in PowerPoint, but the tools are clumsy. The lines, arrows, and built-in shapes were built for creating flowcharts for business and engineering presentations. Scientists struggled with the auto-place feature, resorted to default system fonts like Comic Sans and Arial to overlay text on the image, and frequently lost visuals in their presentations due to PC/Mac version incompatibly. One might wonder why they used PowerPoint in this way at all. However, this method of annotation aligns with a technique with a long heritage in planetary geology. Planetary scientists are taught, based on the principles of geomorphology, to draw their way through their work. They practice drawing geological contact points and identifying features on images taken from orbit or from the air. Annotation also served an organizational purpose. As I have argued elsewhere, getting everyone on the same page toward consensus requires bringing far-flung collaboration members together into a singular position (Vertesi 2015). The PowerPoint slides on this team enable the process of arriving at this consensus by establishing place and shared conversational ground (Schegloff 1972), at the same time as they enable all team members to view the same elements on the terrain around the robot as the same type of thing (“seeing as”; Vertesi 2015). This encourages the conversational flow throughout the meeting as one that moves from a shared position at the meeting’s outset to a consensus moment at its end.
Alongside this repetition of visual imagery, then, the mutual entrainment of the PowerPoint presentation with its recognizable local style was an essential part of the meeting’s ritual practice (Collins 2004). The slide deck’s relatively stable format and rhythmic presentation contributed to a sense of a ritual quality in each individual meeting, no matter the specific issues of the day. As such, the team’s use of PowerPoint to enable a unified, shared perspective helped to reproduce the collaboration’s organizational commitments toward collectivism as they planned a robot’s actions on another planet.

How to Excel in Orbit

Just as PowerPoint is ubiquitous on the collectivist team, work with spreadsheets is a central part of life on the matrix collaboration. On the latter mission, an entire subgroup typically works on a spreadsheet at once during periods of copresence, projected onto a shared screen in the room and circulated by email for remote meeting participants to follow along during the conversation. As a site of mutual entrainment through which interactions are ordered, the spreadsheet coordinates work within the collaboration. It is also the site of recording those same interactions, as individuals input text into the cells to record the group’s agreements. As such, work with spreadsheets on the orbiter mission is group work that also produces collective entrainment and paces the ritual elements of their meetings.

Much like PowerPoint is based on business presentation formats, the Excel program in the Microsoft Office productivity suite is based upon both digital and analog accounting and budgeting platforms, including Lotus software (see Dourish 2017). The software presents a grid organized in columns and rows, and is equipped with a variety of number-crunching tools to enable the quick and routine tabulation of money, percentages, and graphs. As part of producing these tables and graphs, Excel comes equipped with text functions (largely for titles and labels), and the ability to color spreadsheet cells or text to improve legibility in large tables of numbers. But while Excel spreadsheets are optimized for computation and tabulation of numbers, orbiter members rarely use its actuarial functions, numerical expressions, or graphic capabilities. Instead, they enter text and colors in rows and columns that suggest and produce an order to their conversation. The orbiter’s scientists and engineers use these attributes of the Excel software to articulate their work and archive its achievement.

There is no material element of the planets or exploration context that automatically recommends Excel to an orbital mission and PowerPoint to a land-based one. But the choice of software is consistent with each team’s organizational context. Excel on the matrix mission provides an environment for text entry in a tabular, gridded environment to reflect the team’s internal organization and to catalog priority assignments among what are often incommensurate observational requests from different subgroups. In a matrix organization where individuals may cluster by both research interest and instrument team membership, it is essential to locate the correct spokesperson for each subgroup as well as to visualize and seek coverage of all of the collaboration’s priorities. The presence of the spreadsheet cell at the intersection between row and column speaks to the position of individual members of the collaboration: members responsible for atmospheric observations fill in the atmosphere boxes, while individuals on the infrared instrument team are responsible for those cells related to infrared observations.
This continues until all perspectives are accounted for and made visible across subgroup boundaries. In this way, the record of conversation, group convergence, and accountability is carried forward and made actionable as scientific priorities. These are ultimately transformed into spacecraft commands.

The spreadsheet therefore not only captures knowledge from different sides of the organization but also participates in the work of delineating and recording both ownership and prioritization. Like the collectivist team and PowerPoint, most scientists and engineers on the matrix collaboration had a number of colorful spreadsheets open on their desktops and splayed across communal screen. Through discussion at their meetings they either modified the spreadsheet or created a new one based on the values that came up through the discussion. That spreadsheet was then carried forward to the next task, where the process repeats. Like the many generations of PowerPoint slides described above, orbiter teammates typically use the spreadsheet as a guide for discussion, focusing attention, filling in or changing the spreadsheet subsequent to discussion, and then archiving the change in the spreadsheet. Thus these working spreadsheets also have a heritage and a trajectory, generating new spreadsheets or nested amongst other spreadsheets.

For example, a meeting I attended in April 2009 was organized around and through spreadsheet work and annotation. The goal of the meeting was for the group to divide up over 100 flybys of a moon of interest in the planetary system, assigning observational time to different instrument teams. At the start of the three-day meeting, held at an engineering laboratory on the East Coast of the United States, a spreadsheet was projected at the screen at the front of the room and a hard copy was circulated in the room as well. Each row indicated a prospective flyby of the moon, assigned a number in sequence. Also listed were relevant details that located a particular pass in a particular year or period in the future such as start time, end time, operational modes, and any instruments that might observe as an additional “ride along” opportunity. Each flyby also had a tentative assignment of an observation requested by one or more instruments in advance. Some of these were noted to be in conflict or otherwise contested by more than one instrument: these were highlighted in yellow on the spreadsheet (figure 2).

At the outset of the meeting, the coordinator announced the process: “What we thought we would do is sort of go around the room and annotate the spreadsheet to give an idea of what observations have been allocated.” She then asked the group, “Is there any discussion we wanna have before we just jump into the spreadsheet?” There was some discussion over one pass, which was contested between two instruments and required intervention from a senior member of the collaboration to determine whether the pass should go to one instrument or another: he suggested the latter. As a result, a meeting coordinator expressed, “here’s the plan”: “[The senior scientist] is going to do a little more work on his spreadsheet, and then he’s going to present what we have come to as a consensus . . . and then we thought what we’d do is present part of the answer as a suggestion . . . and see if that helps.” Note that the spreadsheet was the location for working out a conflict between two of the matrix organization’s subgroups. The team gravitated toward this spreadsheet row and its indicated conflict as a problem worth getting out of the way before they began the rest of their process. They then stepped away from the spreadsheet to resolve what should be placed in that cell. When the conflict was resolved, the spreadsheet was updated to represent the group’s decision. The senior scientist would then confirm the agreed-upon solution to the conflict, indicated in the contested Excel cell. Following this, the group went through the spreadsheet line by
Organizational Consistencies, Technical Fluidities

These two examples establish a way for STS analysts to consider mundane software tools used in the context of scientific and technical work. Both presentation formats bring collaboration members together in a moment of collective focus, but each does so in a way that represents the organizational orientation of the collaboration. The collectivist team’s PowerPoint slides bring individuals together into the same subject position from which there is a single, shared point of view, to address the challenges of decision making via consensus. The matrix team’s Excel rows and columns make clear the responsibilities of different individuals for exerting ownership over different tasks and elements of exploration, with colorful boxes visually representing the plurality of perspectives that must be factored into any decision. Rows and columns account for the competing priorities across the matrix organization, while the annotated slides circulating in the collectivist mission serve as a site for alignment around a unified stance.

Thus both collaborations’ members conscript these two ubiquitous software tools from the Microsoft Office Suite into the coordination of their collaborative
activities, the collective entrainment of collaboration meetings, and the ultimate ritual reproduction of the organizational orientation on each mission. Both mission teams use software to structure their teammates’ interactions in their ritual meetings to formulate plans for the spacecraft’s activities. The experience of working with the same spreadsheet or observing the same presentation slides brings members of each collaboration into a moment of collective focus and mutual entrainment. In this interaction ritual (Collins 2004), each organization reproduces itself through its interactions, and representational forms. And the productivity software is an essential part of this interactional moment, with implications for the scientific data that are ultimately collected on each team. Each of these software conscriptions therefore represents a considerable accomplishment on behalf of the individuals using standardized tools for their specific, situated ends.

At first glance, this recalls early work in organizational studies of technology. Like Steve Barley’s observations of CT scanners in two different radiology departments, we might read these contrasting use cases as examples of “structuration” and the development of local “scripts” that underlie or undermine organizational hierarchies and work processes (Barley 1996). More recently, organizational scholars such as Joanne Yates, Wanda Orlikowski, Paul Leonardi, and others have explored how social norms and cues combine with material properties to shape and constrain activity in an organizational milieu. The material properties of these technologies acquire meaning by virtue of the context of organizational practices that imbue them with local significance: putting the “socio-” in “sociomateriality.” Thus in a parallel study of a spacecraft collaboration, Mazmanian et al. (2014) argue that software is “figured and reconfigured” in contexts of practice that produce meaning and legibility on earth of matters in space. In another take on sociomateriality, Leonardi argues that material and social properties are “imbri-cated” in an organizational setting, stacked amid each other much like tiles on a rooftop (Leonardi 2012).

Yet the question remains as to which properties and which elements come to matter in practice, both in actors’ and in analysts’ views. What do actors (and analysts) assume is “written in” to the software? What do they think is the “material” with which “the social context” must contend? Science and technology studies has largely shied away from describing patterns of technology uptake in organizational context, so it is unclear how our present theoretical tools would help us to describe creative repurposing of commercial software. Current framings might instead reveal the commercial-governmental axis along which such software suites appear to seamlessly flow (Edwards 1997), presenting a lock-in situation where such tools are ready to hand even if they are imperfect or inefficient for the task at hand (Schwartz Cowan 1985) and perhaps suggesting an approach to the politics of platforms (Gillespie 2010). But this would not allow us to assess which properties come to matter in an organizational milieu, and why. They might suggest a form of interpretive flexibility deployed by a relevant social group (in this case, a scientific collaboration) with a problem that the technology does not quite solve, or a form of local tinkering to address a problem at hand (Kline and Pinch 1996). However, the properties of commercial, widely available software suites appear to be simply there for the taking: users cannot open the hood or the black box in order to alter inner workings (Gillespie 2006) or initiate interactions that disrupt patterns of usage (Woolgar 1990; Latzko-Toth et al., this volume). And while looking at how contexts of use influence design iterations is important for the SCOT framework (Kline and Pinch 1996; Oudshoorn and Pinch 2003), in the case of mundane
and ubiquitous technical tools not every set of users can be considered a “relevant social group” whose use may reshape the technology or influence design. Ubiquitous corporate software products typically ascribe limited agency to local users, making individual users ineffective writ large when it comes to shaping the attributes of the systems that they deploy.

This is also not a clear case of de-scripting a network. The networked conditions of use are very much present and still in place, from governmental-industrial partnerships to the institutional relations that make this software appear to be “ubiquitous” in the first place (Latour 2005; Kling 1991). Nor are these users “non-users” (Wyatt 2003) with a particular politics. And while their representational work of other planets may produce hybrid objects inscribed with earth-bound analytical work (Haraway 1991, 1997; Barad 2003; Myers 2014; Vertesi 2015), it is not clear if these mild diversions from expected use require meeting the material properties of the software halfway.

Enter the Gibsonian “affordance.”

Affordance Theory and Its Discontents

The language of “affordances” might seem to safely occupy this middle ground between the social construction of technical tools and local patterns of use, between intentionally designed-in aspects of software and their potential creative repurposing in social context. It offers a lightweight material determinism that points to which properties of a technology people are working with when they do their (social) work. This notion of “affordances” is prominent in the sociotechnical analysis of digital systems, especially in the communications literature, to point to the role of the underlying architecture of systems in influencing yet not entirely determining digital interactions. In these accounts, technologies possess different capacities for action: “affordances” that are baked into hardware or software and can be uncovered, used, or resisted by users as they encounter or otherwise perceive them in the wild.

STS scholars moving into the study of software will encounter the phrase in widespread use. Social media sites are said to “have” the affordances of “visibility” and “persistence,” affecting how these platforms are used in a range of contexts, from corporations to high schools to protest movements (boyd 2010; Treem and Leonardi 2013; Tufecki 2017). In a similar vein, the many affordances of paper over digital systems make the idea of a paperless office an obvious “myth” (Sellen and Harper 2001). These highly cited ethnographic accounts of user practices arise from taking the relational view of affordances, articulating how social media’s designed-in qualities of persistence and visibility are managed by coworkers, by teens, or by activists, or how paper is preferable for office workers. But as their analysis travels, these affordances harden into essentialist, taxonomized properties of the system at hand: properties that analysts assume are then either taken up or resisted elsewhere in ways consistent with different cultural milieus (e.g., Vitak and Kim 2014; Raja-Yusof et al. 2016; Vaast and Kaganer 2013, etc.; see also critiques in Evans et al. 2017 and Costa 2018).

In line with prior work in this vein, the story of the present chapter could be one in which new affordances of Excel and PowerPoint are surfaced to add to a list of Microsoft Office properties, demonstrating how actors deploy these affordances and resist others to achieve their local tasks consistent with their cultures. But this
would produce certain blind spots that trouble the STS scholar. After all, object properties are neither static or neutral. There was a time in the 1990s when the Internet represented ephemerality and anonymity, not persistence and visibility, a viewpoint that persists among librarians if not among commercial data-gathering platform engineers. Push a little further and each of these “affordances” appears to be baked into not the digital or technical product, but rather the eye of the beholding usage community. Again, although “affordances” initially arose from precisely this relational view, as affordance taxonomies travel they blackbox the social construction of the (software) system and focus too readily on the context of use as meaning determinative.

This sets up a peculiar logical fallacy. As a thought experiment, consider what it would take to list all the affordances of Excel as a software program up front. Not only would this be a very long list, but it could be populated only by particular attributes of the technology that are visible in particular times and places of situated practice. For instance, it is only in observing the use of Excel by the matrix team members that one might think to notice the use of color and text alongside formula management as an “affordance” of Excel in the first place. As such, any interest in an exhaustive list of technical affordances is a Sisyphean exercise, reminiscent of the Borgesian map that expands to represent a territory in its entirety. Because “affordances” are visible only in practice, they cannot necessarily be thoroughly cataloged in advance. Due to this mistake of post hoc propter hoc, their resulting explanatory capabilities are thin.

This visibility in practice leads to a frequent confusion between actors’ categories and analytical categories. That is, it is only in local contexts of use, in situated accomplishments, that particular attributes of a technology become perspicuous as affordances at all—and then only to particular communities of users. This proposed vocabulary recalls the ethnomethodological concept of accomplishment, with its focus on how the intelligibility of social life and interaction cannot be taken for granted a priori, but must be achieved through practical, observable members’ work (Garfinkel 1967). Casting work with software as a question of members’ accomplishments therefore suggests a different orientation toward the problem of the material constraints of software and the relevance of its designed-in features in daily life. Software accomplishments remind the analyst that it is members’ work that makes software systems legible in practice, and that casts certain features of a system as relevant or perspicuous to sense making or navigating local social worlds.5

A key element of orienting toward software use as situated accomplishment is observing members’ own material-semiotic work in the world: the local dualities of nature-culture, human-machine, or object-agent that individuals routinely draw to make sense of software-facilitated action (cf. Suchman 2006). This is also part of members’ work-in-the-world. But it also means that any post hoc account of the demarcation between “the material” from “the social” that assumes such categories arise from the properties of objects themselves is logically problematic. Because any attempted taxonomy of a technology’s available affordances must, by definition, derive from a singular user group’s perspective on a technology at hand—based on that group’s local, material-semiotic, situated, practical accomplishments—elevating this emic definition to an etic one commits a second logical fallacy. Indeed, all “affordances” can only ever be what Nagy and Neff (2015) call “imagined affordances”: whether imagined by users and projected into the context of design, imagined by designers and projected into context of use, or imagined by analysts and projected into the contexts of both use and design.
Because articulating “affordances” (and their discontents) requires us to take on a particular user (or designer) group’s material-semiotic work as primary and to view alternatives as disruptive, attempts at defining a technology’s affordances are a power-laden venture that writes certain users in and leaves others at the margins. To adopt a single group’s local definition of a technology’s “affordances” as canonical flies in the face of the “partial perspectives” (Haraway 1997) that are so essential to understanding the complex relationships between individuals and technologies, and contradicts many of the core principles that STS scholars espouse. We need instead some way to state the importance of the wide variety and modes of technology use to the production of group membership and identity, without assuming a priori categories that describe certain techniques as othering practices as opposed to practices of belonging.

Another Way Forward

One solution to this problem is to argue for a return to the original, relational concept of the affordance. But “affordances” even in this view are too often an *explanandum* masquerading as *explanans*. Instead of conceptualizing technical constraints as baked in and passively there for the using, or viewing material elements as available a priori and then selected for their virtues in structuring an organizational field site, I argue that an STS-centric approach to digital software systems in practice should instead focus on *software accomplishments*: how different opportunities for acting-with a technology are enacted or brought into focus by each group. It is not simply that a material property (an “affordance”) becomes visible and can be catalogued through the relational processes of acting-with (or acting-against). Rather we must eschew the taxonomic and passive view suggested by contemporary deployments of “affordances” in exchange for a view of *how actors work with software in the world*: how individuals deploy various attributes, write novel scripts, or otherwise enroll software-enacted techniques as resources for producing and maintaining local social order. After all, it is only by means of situated practice that certain elements available to software users, such as coloring or annotation tools, come to be seen as perspicuous properties to begin with. The language of software achievements captures this local, situated, organizational, and ultimately material-semiotic perspective.

Further, moving away from a passive, taxonomizing language of “affordances” allows us to better observe where actors themselves draw the line between software properties (“the material”) and organizational expectations for software use (“the social”). Such a perspective shifts the sociomaterial view of constraint and enablement within technological systems. Rather than being etic categories—properties of an object in the world—locally achieved software properties and distinctions between the technical and the social should be analyzed as *emic* categories: local achievements of sense making and membership work. These local assumptions are ones that the analyst must endeavor to understand and ground in context of use.

We must therefore go a step further to identify what local actors consider to be “constraining” and what they consider to be “enabling” in the technologies they use. This assists the analyst in uncovering the politics at play in the local site (on a related example on “constraint,” see Vertesi 2015, chap. 7). From this point of view, the technique of coloring in Excel cells or of importing large image files into PowerPoint does not become meaningful, perspicuous, or identifiable as an important
attribute of the software until actors locate and engage that element of the software tool and narrate it as such. Should we take this work for granted, we would obscure the politics of the production of “the material” and “the social” in the software suite, casting these as something external to locally meaningful actors’ work in the world.

This approach offers to shift present discussions of digital technologies in social context. For example, we might more readily frame “persistence” or “searchability” online not as natural qualities of the Internet or as designed-in properties of social media sites, but as the heterogeneous accomplishments of economic actors like Facebook, Twitter, and Google: companies that configure and stabilize activity online as an engine of capital, as opposed to alternative configurations of the online sphere as a place for anonymity (as in, for instance, Turner 2008 or Coleman 2014). This avoids elevating certain groups’ understandings of digital interactions, made perspicuous in their practices and concerns at a particular time and place, as the matters of concern and objective realities of a digital tool itself, silencing other voices.\(^7\) It also avoids assuming retrospectively that non-searchability was a pressing problem that needed solving (a Whig-historical approach), while revealing the invisible labor of those who work hard to stabilize such a view of the Internet as natural or possessing such objective properties. The goal of reframing software taxonomies as a question of accomplishments instead of as technological “affordances” is to surface these STS-relevant topics of concern for focused scholarly investigation.

**Identifying Material/Social Properties as Members’ Work**

With this in mind, let us return to the spacecraft case for an example of what this perspective makes perspicuous to the STS scholar. I here draw attention to those moments when the stability of the software artifact and the social milieu are under negotiation: they must be accomplished in context. While certainly a process of stabilization would take place over the early days of a collaboration, even several years into the collaboration there are moments when such qualities are made visible. This is especially clear in jokes that question or draw attention to the standardized practices at hand, questioning their naturalness and wondering how it could be otherwise.

Above I described how the ground-based team’s PowerPoint slides are said to “feed forward” into new presentations. It is tempting, perhaps, to label “feeding forward” or some archival property involved as an “affordance” of the file format or software. However, this is a property made perspicuous only in this particular context of use (consider how many slides are quickly forgotten elsewhere). This was evident when members of the team themselves made jokes or comments about this property of the slides, legible only in their social context. For instance, slide annotations were so critical to the collaboration’s work that scientists often took each other to task if they felt the annotations did not align with their own observations, expectations, or discussion. When a senior team member was challenged over the interpretation they had written onto a slide, one of his former students explained to me that it was important that the image showed the right thing: “If you label the slide, it just snowballs . . . [team members] reuse the slide, and it becomes part of the lexicon” (personal conversation, August 25, 2011). Thus the “feeding forward” property of the slides was problematized not necessarily as a
natural feature of the software, but as an element embedded in social practices associated with legitimacy and knowledge work.

This does not mean that “accuracy” was a taken-for-granted social norm either. In another case, one of the collaboration members on the mission used Photoshop to doctor an image that the spacecraft took to commemorate a distance milestone, embedding a fictional four-digit odometer at 9,997 on the robot’s body, close to rolling over to 0,000. The image was imported into the PowerPoint deck as a joke and remained there for up to a week, generating chuckles from collaboration members each time it was displayed. This Y2K-style visual humor played on the archival sensibility that slides had acquired in this context, drawing attention to its import while at the same time subverting the importance of accuracy in the archival record as a social practice.

Like consensus team members’ jokes about their slides, Excel is frequently the subject of humor on the matrix mission. In one mission meeting, scientists were debating the relative merits of a task they faced together when one of them in the room offered bluntly, “It could be a colorful spreadsheet, so we have to do it.” This good-humored jab at the work of spreadsheet filling highlighted the activity’s centrality to the collaboration’s practices and processes. It also drew attention to how the embeddedness of spreadsheets—their heritage and trajectory within a wide variety of spreadsheets—was itself not a property of the spreadsheet itself, but an element of organizational practice that had become entangled with the spreadsheet as an obvious (to members) matter of fact.

Another example of this was the coordination work executed across instrument teams in the orbiting spacecraft’s path selection process, the final stages of which I observed in 2009. The group had to rate seven possible pathway options according to how well these achieved local scientific goals, reporting back to the collaboration with a master spreadsheet. Each pathway had its own column, the scientific priorities were in rows, and the team had to fill in single color—red, yellow, or green—filling in the corresponding box to indicate the tour’s satisfactory rating. Then, “everything” would be put into “one combined spreadsheet” (January 27, 2009). As one subgroup’s meeting kicked off, the team’s science planner described the purpose of the process, embedded in a history of spreadsheet work that had come before:

The [group planning] approach was to take the prioritized spreadsheet that had been worked on very hard . . . priority that is well articulated, determine the seasonal changes in the methane—hydrocarbon. . . . Okay who contributes to this objective? They got their own row in this massive spreadsheet I’ve been talking about. . . . What is their piece of the pie, so to speak, in terms of how they’re going to achieve this objective? Over to the right there are 20 rows that are, “what are the [instrument] priorities?” . . . Now we’re at the point where each instrument is going to assess how well this part of the [path] [gestures to columns] achieves this objective [gestures to row]. (January 26, 2009)

The spreadsheets that the group faced on that day were themselves the latest in a chain of inscriptions (Latour 1995) that subsumed a long list of priorities and suggestions. As this science planner described its heritage, “The story so far is that we took the objectives that were drafted and the tour designers asked each group to draft the objectives . . . we have hundreds of pieces of input.” Note how in this quote and above the spacecraft’s path and the group’s objectives were inextricably
entangled and enacted with columns and rows, reflecting the wicked organizational problem of giving each group “their piece of the [observational] pie” through “their own row.” This generated “hundreds of pieces of input” from across the heterogeneous organization.

As was typical in this team’s collaboration, the spreadsheet became the site of their collective attention. It also provided the order for the conversation in the room as the group went through, coloring in one cell at a time. The spreadsheet was projected on a screen at the front of the room, “just so everyone’s getting used to looking at this spreadsheet,” as well as emailed out to members of the group (figure 3). The meeting proceeded by looking at the tours and priorities and assigning colors to the spreadsheet boxes. “For tour one, priority one . . . looking at that for this group, [tour number] seven is green, or yellow, or red?” The science planner called upon different subgroups to voice their opinions of the tour, filling in the spreadsheet accordingly. For example, one team said, “With our simulation . . . [that] our colleagues . . . shows that number six is the worst.” The planner then responded, “Is there any objections to turning this [box] to red?” The scientist confirmed, “It can turn into red.”

It soon became clear that the three colors did not adequately express the range of opinions in the room. Scientists hedged their accounts by suggesting that a particular observation should be ranked below a green-ranked one, but not as far down as the yellow range. The science planner started suggesting intermediary colors, like “lemon-lime” and “orange” and the spreadsheet lit up in a rainbow of colors, with scientists eventually commenting along the lines of this request: “I want to change the color of that box. It was an orange but it should be a pink.”

Excel provides built-in functions that add up, multiply, or otherwise compute the values listed in the cells. But text and color cannot be automatically tabulated. When the group reconvened later that afternoon, then, they were faced with the problem of aggregation. How to compute all these instrumental needs, represented on a rainbow scale, into a single recommendation for or against a tour option in one of only three colors? The scientist who led the group suggested assigning numbers to the boxes, which could then be computed as an average, giving a score that could produce a red, yellow, or green recommendation. He randomly values on a scale of 1 to 100 to correspond to each color, and began inputting them in the cells. This provoked consternation: How were these numbers chosen? What did it mean to give 85 to lemon-limes? Should all lemon-limes have 85s, or did the color represent a range of values? One scientist suggested using a continuous color scale. His colleague corrected, “we should use numbers, which are a continuous scale”—to which another piped up, to much laughter, “Can we use irrational numbers?” The question then arose as to whether or not the computed average should be weighted or not, and once the averages were revealed and a score of 83.6 was computed for one of the columns, one of the scientists asked puckishly if they shouldn’t differentiate in the third significant figure. Someone else suggested consulting a different document, saying, “I think I have a spreadsheet somewhere. . . .” A senior scientist in the room jokingly described this process of assigning numbers to colors to his colleagues as an instance of “saving the hypothesis,” and compared it to the epicycles in “the universe of Ptolemy.” Indeed, the numbers seemed to serve more of a gut check in the final analysis, as I noted the group started changing color assignments to end up with different averages for each column.

Certainly the team’s humor pointed to the central role that Excel had come to play in their organizational life and decision-making process. Moving from colors
to numbers and back again offered creative solutions to the local problem of deciding what the spacecraft should do by considering a plurality of inputs. However, this activity also represented a software accomplishment, at the same time opening up to analytical view the very processes of meaning-making assigned to the spreadsheet, its properties, and its suggested organizational outcomes. For a team well practiced in using color to depict preferences, this breakdown in their ability to use color to suggest an equitable and “objective” outcome of the practice laid bare their assumptions about spreadsheet work, especially those of its properties that were typically used to generate fair decisions. The dissatisfaction with numbers and the ultimate color scale that the team thought best preserved their preferences troubled the very question of what Excel is good at doing. Neither the spreadsheet’s toolbar of potential options nor its organizational qualities of maintaining distinctions between groups were natural or obvious in this circumstance. Indeed, the team had to work to distinguish which decision-making elements were “of the spreadsheet” and which properties were “of the group”—the latter making the enterprise a house of cards “like the universe of Ptolemy.”

Discussion

Because actors engage aspects of a mundane software tool as resources for structuring and ordering their local social world, calling attention to how a particular group uses software packages is a question of surfacing how software is enrolled
in group membership and organizational practices. *Software is thus a ready-to-hand element in the situated achievement of work.* Attunement to the organizational context of this work can reveal patterned ways in which users deploy different attributes of software to different ends in ways that are nonconforming, yet not idiosyncratic: after all, these local techniques are ultimately both organizationally contingent and reproducing. Even when user groups identify certain practices as problematic or against the grain and others as mainstream, then (as in the “mis-users” in Latzko-Toth et al., this volume), we would be remiss to elevate any single group’s emic categorization to the level of an etic taxonomy, or anything other than an expression of group membership, participation, and local practices. When actors perform sense making and decision making with software, this is more frequently a question of ordering phenomena in the world than of sense making about software itself (on an ethnomethodological approach to that topic, see Singh et al., this volume).

The “here and now”—ness of these software forms—what Garfinkel would call their *quiddity* or *haeccicity* (Garfinkel 1967)—enables them to become more than simply tools or a genre of organizational expression. While there are certainly elements of their local usage that beg the “genre” framing (Yates and Orlikowski 2007), they might also analytically be identified as the stuff of actors’ work with which the social orders of each collaboration are performed, expressed, made, and remade. Hence examining how groups work with these digital tools does not require identifying passive attributes that can be taken up in social context; it is rather a question of revealing which situated *software accomplishments* groups turn to in order to simultaneously perform and reify their organizational and social milieu.

*Software accomplishments* are also the ways in which members of these collaborations perform membership and individual fluency with local norms and practices. Members’ own ethnomethods (Garfinkel 1967) include facility with particular modes of software interaction and display, through which local sense making and management of team members with their varied interests can be accomplished. Membership on each team overlaps slightly, but members of both teams do not mix modes or presentational expectations. Thus the story of these tools and representational forms is not simply a story of how roving and orbiting team members have adjusted to or taken up these classic productivity software suites to local organizational ends. It is a story of how both collaborations have produced the local accomplishment of incorporating these tools, ways of visualizing, and projecting spacecraft planning into the very heart of their planning process, in ways consistent with their local organizational norms and structures. Ultimately, it is also a story about just how malleable and visible these competing elements are within the context of actors’ work.

There are a few contrasts here to organizational studies of software at work. In a move popular in information systems study and design at the end of the 20th century, Kling and Scacchi (1982) proposed that isomorphism between organizational structures and software infrastructures is essential to tool uptake. This isomorphic tendency guided early computer-supported cooperative work and groupware project development. However, PowerPoint and Excel do not emerge from the (black-)box pre-prepared to align with all organizational structures. If we were to read these systems in such a way, the template architecture built into PowerPoint suggests a rather more bureaucratic form than the ground-based
team enacts using that same software. Instead, these examples of technical practices demonstrate that contexts of use are more important than any structural tendencies embedded within the software itself.

This is equally the case in sociomaterial scholarship, which attends deeply to practice and to bottom-up theorizing as opposed to top-down categorization or structuration. The case of these two spacecraft teams reveals a paradox in which an even more micro perspective is necessary. “The social” and “the material” take shape through members’ interaction, much as “affordances” were initially intended to arise from relational encounters. But it only through attenuation to practice, in the bottom-up work of collaborative planning and interaction, that individual software elements become analytically perspicuous, resolve into “social” or “material” categories, or are naturalized analytically as “sociomaterial” understandings. Arriving at the observational moment with categories assigned a priori—social and technical, normal and deviant, built in and uncovered—can too easily lead us to read such categories backward into the field, causing us to lose sight of the political or interinstitutional processes that shape what we, as analysts, might consider to be a digital property or an organizational imperative in the first place.

Conclusion

Thinking about mundane software tools in technoscientific context reveals several possibilities for STS analysis. Much existing work in STS has shown how categories and assumptions are built into information systems, and the work of wrestling with messy phenomena to produce data with no “residual categories” (Woolgar 1990; Ribes and Bowker 2008; Millerand et al. 2013; Bowker and Star 1999; Ribes and Jackson 2013). Studying software in scientific and technical organizations requires adopting an approach to representation and documentation in which the electronic documents in question are not only the result of work, but the sites of work as well (Ziewitz 2011; Vertesi 2014). As the center of collective attention, they are the place where work is done. It is therefore important to see them as evolving and live representational forms that both serve to structure a meeting and can be manipulated and changed as part of the meeting process. They are not passive sites for inscription, but an active part of meetings at which individuals are copresent virtually and physically (Beaulieu 2010). Whether projected on a shared screen or emailed out in advance, the documents participate in coordinating and focusing collaborative work. Even if the documents come to us “cold” from many years in an archive, analysts must work to see them as evolving documents with both a history and a trajectory, as well as a just-here and just-now-ness with which individuals actually do their work. The language of software accomplishments encompasses this active, ongoing, meaning-making process.

The ubiquity and general availability of these software tools speak to a broad circulation across laboratory environments and other contexts (for example, managerial contexts), perhaps on a scale not otherwise visible for other types of knowledge-making equipment. At the same time, it is clear that such one-size-fits-all solutions like office productivity software are used differently across local environments to suit quite different sorts of ends. We cannot truly call this a process of interpretive flexibility, stabilization, and closure, as such local groups cannot be said to be “relevant” in any way to the continuing development of the software.
Further, the software’s use in these different contexts does not descript any networks or radically upend our understanding of how software should be used. Still, a standard software package’s use in these different contexts suggests valuable ways of approaching software—like other tools—in and across technoscientific organizational milieus.

The first is a commitment to the organizational situatedness of knowledge work. In this case, the software is a representational tool that not only represents the organization to itself but is part of the active work of producing and reproducing that organization. Exactly how these software systems are put to use locally—not necessarily adapted, modified, or even appropriated per se, but quite simply and daily, used or acted-with—reveals organizational work of ordering the natural, social, and software worlds at the same time. In this sense, even the most mundane, ubiquitous, cookie-cutter tools can serve to produce and reproduce different organizations in different contexts. Further, neither the organization nor the tool come “preloaded” with organizationally enhancing ways of acting or interacting: both the tool as site of interaction and the organization as collaborative framework are enacted through the local use of the software.

Second, it is important to present an alternative way of examining the diverse ways in which technologies are put to local use without resorting to the language of affordances as a question of embedded material properties that permit certain patterns of use over others. The notion that technologies might in and of themselves suggest, prompt, or require different ways of using them from human bodies or interlocutors neglects the richness and complexity that occurs when different groups take up technological tools to achieve local ends: a richness and complexity that STS has always aimed to surface. It also risks reifying one social group’s notion of patterns of use, due perhaps to their power in controlling the narrative, at the expense of another’s, othering particular patterns of use without understanding local contingencies and expectations of group action. Instead of leaving “affordances” as an analytic black box, STS scholars should use it as a starting point for analysis: when something is glossed as an “affordance,” whether by users, designers, or analysts, this is where our analytic process must begin. Continued attention to observable practice can help us to address how such software is put to use in different kinds of contexts as a tool for achieving local ends and making local distinctions, without resorting to taxonomies that unintentionally delimit particular groups of users.

Finally, there is value in studying and surfacing different forms of software as the invisible tools of knowledge work. There is much work in STS and computer-supported cooperative work that has examined the development of bespoke tools, drawing largely on the infrastructure or cyberinfrastructure perspective (Ribes and Bowker 2008; Lee et al. 2006). Microsoft products are perhaps less sexy than large-scale, government-funded e-science projects with their conflicting ontologies and interdisciplinary groups of computer scientists and domain experts. Yet without PowerPoint or Excel the work of these spacecraft collaborations would look quite different; without including these tools in the story of these scientists’ work, any account of their practices would be severely limited. Returning to the shop floor to witness the everyday work of scientific practice can surface the most surprising of technologies and techniques that are essential to the conduct of scientific practice today.
Acknowledgments

The author thanks the spacecraft teams for permitting her observation of their work, the National Science Foundation for support of the empirical work in this project, and the Spaceteams research group Marisa Cohn, Matt Bietz, David Reinecke, Melissa Mazmanian, and especially Paul Dourish, in dialog with whom many of the early ideas in this piece developed. Helpful comments from danah boyd, Ingrid Erickson, Steve Saywer, Oliver Marsh, Michael Lynch, David Nemer, Malte Ziewitz, Carla Ilten, and the anonymous reviewers shaped and dramatically improved this piece. I am also thankful to Erika Robles-Anderson for circulating her award-winning paper about PowerPoint before its publication.

Notes

1. This chapter focuses on how users interact with software systems, black-boxing decisions concerning software design that take place in a corporate context. This limitation may inform my preference for local accomplishments rather than identifying designed-in properties. While it is certainly true that design decisions can purposefully attempt to circumscribe use of a platform or system, it is important to avoid mistakenly identifying limitations encountered by users as intentionally designed-in “affordances,” even when such components benefit the manufacturing corporation. A study of design and use is the only way to truly get at the designer/user dialectic; when the corporate context of design is closed to the analyst, attention to accomplishments is one way to be circumspect about which attributes of a system we associate with designerly intent. It is possible, further, that the organizational process and practices of designing features into software may also be analyzed as a form of software accomplishment, without assuming user acceptance of these features.

2. The social scientists and computer scientists who studied the mission in its early days noted that science team members were more likely to print photos out and write all over them using colored pens than to use immersive, 3-D digital environments or even a purpose-built digital whiteboard to interpret the terrain and make decisions about robotic action. Eventually the social scientists therefore suggested bringing in large tables and printers so that scientists could pore over printed images together. This was more popular with the science team, who retired the digital whiteboards as “digital clocks.” I am grateful to Roxana Wales, Bill Clancey, Wendy Ju, Jeff Norris, Alonso Rivera, and Mark Powell for sharing these recollections.

3. This analysis of the spreadsheet resonates with that of Dourish (2017); I am grateful for our many conversations about this phenomenon and its analysis.

4. These observations were acquired April 17–19, 2009.

5. See Reeves et al. (2017) for an application of the ethnomethodological “accomplishment” lens to software, in this case gaming systems.

6. This is not incompatible with the relational view of affordances that focuses on how actors confront technological limitations whether through daily practice, evasion, or creative repurposing. However, this view refrains from identifying as material properties of the system those elements that users encounter as limitations; and from assuming those properties to be the result of designerly or unintentional circumscription of user activity.

7. A synergistic argument is offered in Costa (2018), released as this essay went to press, critiquing communications scholars for elevating Anglo-American practices as stable, intrinsic social media “affordances” and arguing also for a renewed emphasis on practice.

8. For more on the Ptolemaic worldview, the epicycles as “saving the hypothesis,” and the transition to Copernicanism, see Kuhn (1958).

Works Cited


FROM AFFORDANCES TO ACCOMPLISHMENTS 391


