Discourse strategies in technical-support interactions

by

Vincent David Robles

A dissertation submitted to the graduate faculty

in partial fulfillment for the degree of

DOCTOR OF PHILOSOPHY

Major: Rhetoric and Professional Communication

Program of Study Committee:
Jo Mackiewicz, Major Professor
Volker Hegelheimer
Pol Herrmann
Dave Roberts
Geoff Sauer

Iowa State University

Ames, Iowa

2017

DEDICATION

To Sareena
To my kids
To my parents

Non nisi te, Domine.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

I would like to thank my committee chair Jo Mackiewicz for pushing me to become a better, more disciplined scholar. Thanks, Jo, for being such a bulwark during this study.

Thank you to my RPC committee members Dave Roberts and Geoff Sauer. Coach, you've had my back since you first introduced me to ISU. Geoff, you've made me a better technical communicator.

Thank you also to Volker Hegelheimer and Pol Herrmann for your support and for your pointed and thoughtful perspectives on this study.

Thank you to Charlie Kostelnick, Tina Coffelt, Stacy Tye-Williams, Abby Dubisar, and Barb Blakely for their advice in forming my professional identity as both a teacher and a scholar.

Thank you to my friends and colleagues for your support and friendship at ISU. Here's to many more years as we meet again at conferences and for research projects.

Thank you to Michael Li, Keegan Kwirant, and Jamie Carroll for your help as coders and transcribers. I had no chance of getting this project finished without you.

Thank you to the Association for Business Communication for funding this study. You made getting research participants and assistants more possible.

Thank you to my research participants—both technical-support providers and users.

Lastly, and most importantly, thank you to my family. This is for you.

ABSTRACT

This study responds to the lack of research on the discourse of technical-support interactions. Researchers have explored successful practices for recruiting technical-support personnel, constructs for determining user satisfaction in survey instruments, and discourse moments of miscommunication and empathy. Even though organizations must maintain users by successfully training technical-support providers to interact with users, no empirical study exists that explores the discourse strategies these workers can employ in the process of defining and resolving technical problems during that interaction. In this study, I examine 20 helpdesk interactions between users and technical-support providers in an educational technology helpdesk at a four-year university in the Midwest United States. Using discourse analysis to explore speech acts at both macro- and microlevels, I provide a comprehensive overview of the discourse in these interactions, helping organizations to maintain users by providing concrete, data-driven discourse strategies that technical-support providers can use in their service to users as they define and resolve technical problems during this important service interaction.

CHAPTER 1: INTRODUCTION

Despite attempts of technology design and development teams to anticipate users' experiences with and to meet users' expectations for a technology, users may still have difficulty with it. To mitigate this difficulty, technical communicators use written, oral, and visual communication to broker the relationships among users, technologies, and the designers and developers. As Dobrin (2004) argued, technical communication exists to "accommodate technology to the user" (118). Technical communicators accommodate the technology to the user by creating printed or online documentation, such as instructional videos or wikis, in which users can generate their own documentation, or discussion forums in which users can converse with other users (Selber 2010). More recently, Kimball (2016) argued that "it would be more accurate to say that technical writing accommodates users to technology" (7). Documentation and help does not necessarily change the machine, but rather "it changes user behavior to make interactions with the machine more successful in terms of the machine" (7). In relation to potential design flaws, "[a]ll technical communication can do is to patch over the rough spots of user-machine interaction by directing the user's fulfillment of the complex process; it typically can't make the process less complex" (8). This reasoning highlights the effect of technical help on users; it directs them through various media to fulfill their technological intentions (at least as far as possible).

Needing help typifies a common user experience with a technology. They may also not feel certain they are doing something correctly. Others may feel opposed to technology in general. Users come to technology and to technology

help with different dispositions—toward technology or toward help or both. To understand the dynamics at work when a user seeks help and a technical communicator provides it, consider the following interaction. In this case, the user sought help with a learning management system.

After settling into their respective positions—the technical-support provider (TS) at the computer and the user (U) sitting to the left, TS began the interaction: "O.K. So please remind me of your last name."

U stated the name, and TS typed it into the learning management system. "And this is for [*course number*]?" TS asked, identifying the user in relationship to the learning management system. U gave the name of the course, and TS was able to find the correct course website.

Having understood that U wanted to delete a student from being enrolled in the course website because the student dropped the course, TS showed the process while U listened:

TS: So the first thing that I would do here is I would go to the administration block and click on users and enrolled users and this is [g- gonna give me the-

U: [O.K.

TS: entirety of your class list.

TS then moved from showing the procedure of finding enrolled students back to inquiring to understand what U needed further, "Um, do we know what student we want to unenroll?"

U began to tell TS where to find the name, "Um, if you scroll down-," but before U could finish, TS specified the original inquiry to better understand what exactly U needed, "Uh, what's their last name?"

U stated the name, and in response, TS began showing U how to find the student: "O.K. great. [*1 second*] So all I did there was hit control find and then entered [*student's first name*], and it took me straight to their name."

U reciprocated, "O.K."

TS paused by checking that U was understanding, "So what we would want to do here then is you see this gear icon [*hovering cursor over icon and pointing with finger*] that's kind of on the right here?"

After U confirmed comprehension, TS began showing the procedure again, using the mouse to highlight and complete operations. Soon into this demonstration, TS paused from showing the operations so that he could explain the benefits of *suspending* rather than *deleting* a student from a course website. U reciprocated agreement as TS explained further that U should suspend, not delete, students who dropped the course. Then TS suspended the student from the course.

U then declared the problem solved: "O.K., that's easy. Um, and then I had another question about the gradebook." And then the movement from problem to resolution began again until U was satisfied.

This interaction between this technical-support provider and this user illustrates the way in which a technical communicator accommodates a user to a technology—what it can and cannot do and what the user should do to more successfully interact with it. Still, some users simply do not like these limitations. While this example showcased a user who calmly participated in the interaction with a technical-support provider and felt the interaction was satisfactory, another user in my study did not feel a technical-support interaction went as

well, to the extent that this user wrote a note on the post-session survey that she "hate[s] technology."

To better understand the interactions between technical communicators and users, researchers in technical communication have traditionally discussed various asynchronous interactions (documentation, forums, wikis), but as the excerpt exemplifies, technical communicators also help users through synchronous person-to-person interactions.

This study explores such a spoken interaction between users and technical-support providers who work at a help desk. In these interactions, technical-support providers "accommodate the user to the technology" by using interpersonal communication strategies to build rapport and communicate empathy, to understand the users' problems, and to resolve the users' problems (Clark, Murfett, Rogers, and Ang 2012; Xu, Wang, Forey, and Li 2010). Such interpersonal, oral technical communication remains relatively unexplored in technical communication research compared to that of written technical communication, despite the benefits such interactions give to both users and organizations. Lam and Hannah (2016), who explored technical-support on Twitter, summarized this dearth of research in this way:

> Ultimately, the relative lack of recent, specific technical communication scholarship about help desk interactions is unsurprising. This is because much of the existing work examines the documentation of technical issues rather than on the live, dialectical problem-solving or troubleshooting relationship between a user and the company. An explanation for this lack of attention in technical communication scholarship could be that customer service and customer satisfaction generally have not been

> understood as technical communication concerns. Rather, they have been understood as a business communication concern, particularly in the area of customer relationship management, or CRM […]. However, […] the distinction between business communication and technical communication is being blurred as customers become users (or vice versa) and have the ability to participate in the development and delivery of technical support […]. Therefore, the time is ripe to consider more fully how technical communicators can and ought to design for and deliver customer service as part of the technical support work they do. (39)

Echoing this growing relationship between technical and business communication, early technical communication research by Redish (1995) called the field to articulate the return on investment of technical communication. More recent research views technical communication as the way that customers engage with business products through communications between users and other users and through communications between users and businesses (Kimball 2016; Lam and Hannah 2016). A study on technical-support interactions thus explores in part this relationship between business and technical communication, showcasing the way that users are customers requiring service and that technical communicators are employees providing that service.

As customers, users value person-to-person help facilities such as helpdesks or helplines because the information they receive in these interactions addresses their specific goals and concerns (Steehouder 2003; van Velsen, Steehouder, and de Jong 2007). Person-to-person help facilities provide "the total user support package" because they complement existing technical communication infrastructure such as documentation or support forums (van

Velsen, Steehouder, and de Jong 2007, 228). Also, users value help facilities because they expect that technical-support providers will resolve their technical problems quickly and that technical-support providers will express concern and investment in the specific problems the users face (Callaghan and Thompson 2002). Documentation may not fulfill these expectations for quick and empathetic help. This expression of empathy, or identification with and understanding of the users' problems, requires technical-support providers to listen attentively and to provide the necessary response to meet the needs efficiently (Clark, Murfett, Rogers, and Ang 2012, 125). This carefully crafted response to the needs of users characterizes the work of technical communicators who must understand not only technology and how it works but also users and what they need.

Because such communication work attends to users and their individual needs, it maintains user loyalty and moderates users' technology acceptance. This user loyalty and acceptance brings value to organizations. When organizations provide technical support for the technologies they produce for customers or for the technologies they require their employees to use, organizations maintain trust with these customers or employees, and they also enable these users to develop more confidence and trust in the technologies themselves (Bell, Hall, and Smalley 2005; Hall, Verghis, Stockton, and Goh 2014; Lee, Hsieh, Ma 2011; Nguyen, Groth, Walsh, and Henneg-Thurau 2014). Thus, promoting user satisfaction with these interactions not only serves the goals and concerns of users but also the goals and concerns of organizations.

However, few studies have explored the technical communication within these interactions despite these known benefits. Clark, Murfett, Rogers, and Ang

(2012) analyzed call-center conversations to demonstrate the empathetic communication strategies customer-support providers use, such as showing concern, reciprocating, listening attentively, and others. These strategies characterize generally positive communication customer-support providers employ with customers seeking help. Xu, Wang, Forey, and Li (2010) conducted a genre analysis of call-center conversations in order to demonstrate the genre structure of these interactions, particularly in an intercultural setting. Kelly (2014) and Beldad and Steehouder (2015), using quantitative content analysis and conversation analysis respectively, focused on miscommunication moments in technical-support interactions across cultures. These researchers focus on specific linguistic facets of the technical-support interactions (empathetic communication strategies and miscommunication moments). Also, all of this research focuses on call-center contexts, which is not the only way this type of interaction happens. They also occur in face-to-face interactions. Also, in early technical-support interaction research, Steehouder and Hartman (2003) and Steehouder (2007) provided an overview of their interpretations of a given helpdesk interaction transcript, showcasing specific outcomes that may help "helpdesk agents" to instruct users. These interpretive analyses were not explicit about the source or reliability of their interpretations. Drawing heavily from Baker, Emmison, and Firth (2005) analysis of helpdesk interactions, Steehouder and Hartman (2003) and Steehouder (2007) identified the phases of the interactions and then reported an interpretation of the interaction for each phase. More reliable and comprehensive discourse research could build on these early studies and strengthen this research area.

Other fields such as organizational behavior (Barely 1996; Das 2002; Pentland 2002), management studies and management information science (Armestad, Keily, Hole, and Prescott 2002; Burgers, de Ruyter, Keen, and Streukens 2000; Callaghan and Thompson 2002), and marketing (Bell, Auh, Smiley 2005; Hall, Verghis, Stockton, and Goh 2014; Nguyen, Groth, Walsh, and Henneg-Thurau 2014), have for some time explored customer and worker satisfaction with both face-to-face and voice-to-voice help interactions. These fields examine managerial decisions and psychological factors that contribute to satisfactory interactions, but they do not analyze the communication that contributes to user satisfaction within these interactions, despite agreeing that "service quality distinguishes an organization from its rivals" and agreeing that service quality links to many business metrics such as customer loyalty, word-of-mouth referral, price insensitivity, sales growth, and market share (Bell, Auh, and Smalley 2005, 169). After interviewing managers about what they expect when hiring new technical-support providers, Callaghan and Thompson (2002) found that managers value "the quality of communication" (234). Hall, Verghis, Stockton, and Goh (2014) concluded that "[i]mproving the quality of communication in customer service can benefit companies, customer service employees of those companies, and customers" (505). Technical communication research into the actual interaction between users and technical-support providers can yield a clearer understanding of what this communication specifically looks like within them. With this knowledge, this study contributes to the research these others fields have done about managerial goals and decisions with technical-support and about how technical-support providers can better serve users (Armestad, Keily, Hole, and Prescott 2002; Hall, Verghis,

Stockton, and Goh 2014). It demonstrates the moment-by-moment communicative actions these workers take as they engage in high-level knowledge work when helping users (Barley 1996, Das 2003). In demonstrating these actions, this study also elevates perceptions of "support work" and "technical-support work" by demonstrating the intricate communicative work these workers employ.

Technical Support as Discourse Work: A Theoretical Framework

Technical-support provider work is a part of a cultural and professional class of work often called "support work." Support work faces assumptions that relegate it to mere craft that primarily focuses on "creating or maintaining the technical infrastructures that enable other people to do their work" (Barley 1996, 422), other people such as medical doctors, engineers, and scientists (Barley 1996, 411). Yet ethnographic observation of technical-support work and of the work of those who do similar support work reveals that these workers demonstrate "more abstract, more symbolic" work requiring them to think at a level "increasingly distanced from [technologies'] physical and sensory referents" (Barley 1996, 411). That is, workers often interpret technical knowledge by adapting it "to the contextually specific needs of users, clients, or customers" (Barely 1996, 423). Technical communicators have long understood that their work requires high-level thinking processes such as these that enable them to adapt technical knowledge for users and to think quickly and systematically as they do so (Johnson-Eilola 1996; Mehlenbacher 2013), and this understanding has allowed the technical communication discipline to argue that it has expertise beyond

mere support and commodity work. Similarly, technical-support providers also have expertise beyond merely enabling others to do their work.

Johnson-Eilola (1996, 2005), using a term from Reich, calls this high-level work "symbolic-analytic work," and this kind of work elevates the value of technical communication beyond the routine production of texts or mere in-person service. Johnson-Eilola describes "symbolic-analytic work" as a process of experimentation, collaboration, abstraction, and systems thinking. Technical-support providers engage in experimentation when they isolate operations and functionality issues, testing the technology as they "talk-out" the breakdown the user experience. They engage in collaboration with other technical-support providers when they consult together about breakdowns or when they consult with managers or technical experts from other disciplinary knowledge bases. They engage in abstraction when they organize and structure information for users to understand. They also engage in system thinking when in order to address a breakdown they consider larger cultural and technical system issues beyond just discrete operations or functionalities. Furthermore, technical-support providers often must think through the users' social and organizational contexts, the users' goals, and the users' values. This analysis allows technical-support providers to better understand the problems users describe and to better make technical knowledge accessible to users when they resolve those problems. Like technical communicators who compose help documentation, technical-support providers produce texts and solve problems through analysis and the manipulation of language.

For this reason, this symbolic-analytic work that technical-support providers engage in is discourse work. It involves using language in a specific

context to say and to do (Gee 2011, 2). Technical-support providers do discourse work by using ideational discourse to "talk-out" users' experiences with technology and by using textual discourse to sequence, organize, and control interactions within institutional contexts.

Technical Support and Ideational Discourse

Throughout technical-support interactions, technical-support providers engage in discourse work by translating technology for users. The technical-support providers translate the technical object through a highly symbolic and abstract process by serving as "brokers" who take the designers' script of the technical object and interpret it for the users (Barley 1996, 423). As Akrich (1992) describes this idea, when designers create a technology in anticipation of its users, they "attempt to predetermine the settings that users are asked to imagine for a particular piece of technology and the pre-scriptions [*sic*]…that accompany it" (208). When users encounter that "pre-scribed" technology and their expectations for that interaction conflict with their experience, a breakdown occurs. According to Das (2003), this breakdown can take two forms: missing functionality and erroneous operations (417). When users expect the technology to do something that it does not do or does not do well enough for them, the breakdown prompts them to seek technical support to address missing functionality. When users encounter the technology failing to meet the designers' specifications (e.g., an unresponsive operation), the breakdown prompts users to seek technical support to address an erroneous operation.

In the process of repairing that breakdown, technical-support providers broker the relationships among the designers, the technologies, and the users.

This brokering process elevates the material world (the technology) from more than routine production into the discursive world (the conversation), a world in which language facilitates, reflects, and articulates the experience of the material realm. The discursive interplay between users and technical-support providers facilitates both the technical-support providers' and the users' understanding of the designers' script until they both find a common understanding that gives order to the users' negative experience with the technologies and that provides the starting point from which technical-support providers can help users resolve their experiences positively.

This help can enable users to envision what the designers intended and can unveil usability problems or user functionality requests for designers and developers to resolve. With these systemic problems or requests unveiled, the interaction expands its discourse from one of "translating" the technology to one of "articulating" the technology, a process explored by Slack, Miller, and Doak (1993). This expansion makes technical-support interactions "an event—with material and social consequences" (Herndl and Licona 2007, 134). The interaction can give the users and the technical-support providers a voice, "an opening, however brief," to have authority over the technology and its design (Herndl and Licona 2007, 150). In this way, technical-support providers, like technical communicators, bridge a user group and a technical community, a process that organizational studies has identified as the high-level knowledge work technical-support providers do (Barley 1996, 423–24).

When technical-support providers and users come to a shared understanding about the users' experiences and the technologies themselves, they exemplify the ideational metafunction of language, which is language's

capacity to reflect human experience or human logic (Halliday and Matthiesen 2014, 30). Through this metafunction of language, the technical-support providers and the users interact to reflect an experience with the technology's functionality and its operations, and the language reflects the technical-support providers' and users' converging understandings of how the technology can or should work. As they converse, they each "impos[e] linguistic order on [the] experience of the endless variation and flow of events" (Halliday and Matthiesen 2014, 30): namely, the technical-support providers and the users give linguistic order to the breakdown the users experience and the events that led to it. Thus to study technical-support interactions successfully, I needed to carefully consider how technical-support providers and users employed language to define and resolve users' experiences with breakdowns. Specifically, this study investigated the language of users seeking help with technical problems and technical-support providers attempting to resolve and also succeeding to resolve technical problems. The types of communication that both technical-support providers and users employ impose order on "a variety of unknowns that become visible as tasks are understood and solutions are tried" (Swarts 2014, 168). This study thus codifies these linguistic actions so that technical-support providers can identify them and use them appropriately.

Technical Support and Textual Discourse

Besides thinking through the dynamics of representing experience and logic (ideational discourse), technical-support providers also demonstrate awareness of "the construction of text" (Halliday and Matthiessen 2014, 30). Technical-support providers build up a sequence and order to the interaction, organize its

flow, and create continuity and cohesion as the interaction moves along. This textual metafunction of language enables technical-support providers to keep the conversation focused. Because technical-support interactions involve the interaction between a member of an institution and a client who encounters the institution, the entire interaction exemplifies the textual metafunction of institutional discourse (Agar 1985, 147). Agar proposes three stages for characterizing the textual metafunction of institutional discourse: diagnosis, directives, and reports.

Diagnosis is the thematic moment of institutional interactions in which the institutional representatives (in this case, technical-support providers) communicate their understanding of the institutions' needs for the interactions, what Agar calls the "institutional frame" (1985, 149). The clients have their own perception of what they need—the "client frame" (Agar 1985, 149). The diagnosis stage is thus "that part of the discourse where the institutional representative[s] fits the client[s'] ways of talking…to ways that fit the institution" (Agar 1985, 149). This stage tends to require a question and answer component that allows for this alignment to occur, and according to Agar (1985), the institutional representatives do the majority of the asking, while the clients do the majority of the answering (150). This theory of problem diagnosis coheres with the problem-solving nature of technical support in which technical-support providers must understand the problems (missing functionality or erroneous operation) as the users describe them.

The directives stage involves the institutional representative directing the client to take actions or directing the organization to take actions for the client (Agar 1985, 149). This discursive stage typically stands at the center of the client's

concerns and can determine the level of the client's satisfaction or not (Agar 1985, 155). Moreover, the directive stage exemplifies the work of institutions: "That is what institutions are supposed to provide—efficient ways to handle routine problems" (Agar 1985, 156). This stage makes sense for technical-support interactions because the users will expect to have their problems resolved.

The report stage of institutional discourse involves the summary of the interaction, usually produced by the institutional representative (Agar 1985, 149). This stage may or may not involve the client and may be written or oral. More frequently the report stage involves some report to other members of the institution and may serve as "the goal against which the progress of the diagnosis is measured" (Agar 1985, 155). The report may serve as the basis for showing how accurate the diagnosis was or for investigating the history of a problem and its solution that the institution has addressed or continually addresses. Generally, this stage involves a representation of the diagnosis and directives into terms that match the discourse of the institution. For technical-support contexts, the report stage may occur as an oral summary of the way the interaction went in cases in which the report stage occurs with the user present, and this oral summary may benefit the user by adding coherence to the interaction. Also, the report stage may occur when the technical-support providers record the interaction in a written report for organizational archiving or record-keeping, which benefits the organization (Schryer 1993). To generate this report, technical-support providers may record the problems and their resolutions from specific interactions. This record allows the technical-support provider team to identify recurrent problems and their recurrent solutions, a process that may help increase efficiency when technical-support providers see

these problems again or that may provide evidence and examples for reporting to a design and development team so that they may improve the technology.

These three stages together make up the larger theoretical framework of the "technical-support interaction genre," which current technical support studies have well established (Clark, Murfett, Rogers, and Ang 2012; Xu, Wang, Forey, and Li 2010). Nevertheless, this study reports on the stages in these interactions not only to reconfirm existing research but also to empirically locate microlevel discourse strategies (smaller discourse units) that serve to diagnose or resolve problems. By locating these microlevel strategies across macrolevel stages, I richly describe the constitutive parts of each stage. This description enables technical-support providers to maintain coherence in the interaction and also to maintain cohesion between stages. Such logical flow orders the users' and the technical-support providers' understanding of how the interaction went, what solutions were reached (or not reached), and how the users' accommodation to the technology itself has reached a new and potentially positive outcome.

The Focus of this Study

This study described the discourse strategies of technical-support providers in face-to-face technical-support interaction with users, or what van Velsen, Steehouder, and de Jong (2007) call "helpdesks" (219). They define helpdesk interactions as sessions in which a customer (or user) personally consults an agent (or technical-support provider) face-to-face about a problem or question they have (219). They contrast these interactions with what they call "helpline" interactions in which customers call a paid or toll-free telephone number for a

voice-to-voice conversation about their help issues. While Halbe (2012) suggests that the communication behavior in face-to-face and voice-to-voice interpersonal interactions may differ, other research supports my claim that the benefits of exploring helpdesk sessions and helpline sessions are similar. Exploring helpdesk interactions provides implications for helpline interactions and vice versa because researchers can understand customer satisfaction the same way in both interactions. Specifically, van Velsen, Steehouder, and de Jong (2007) explored customer satisfaction with both kinds of help facilities using a set of customer service constructs identified by Parasuraman, Zeithamal, and Berry (1988). In business and marketing research, this set of constructs is known as the SERVQUAL approach, and business and marketing researchers often use this approach to inform their survey designs when they want to measure customer satisfaction. The survey van Velsen, Steehouder, and de Jong (2007) used to explore customer satisfaction in both kinds of help facilities—helpdesks and helplines—used the SERVQUAL approach constructs. These constructs are tangibles (the physical space), reliability (competency and knowledge), responsiveness (willingness and promptness), assurance (trust and confidence), and empathy (individualized attention). The overview of items on their research instrument demonstrates that the only difference between customer satisfaction in helpdesks and customer satisfaction in helplines is that of the tangibles construct (the physical space). This study focused on the conversation (the communication between participants in the interaction) because the discourse-based research in technical-support literature is sparse and highly needed (Hall, Verghis, Stockton, and Goh 2014, 504; van Velsen, Steehouder, and de Jong 2007, 228). Because this study of helpdesks largely omitted attention to the physical

environment, this study mirrored those conducted by Xu, Wang, Forey, and Li (2010), Clark, Murfett, Rogers, and Ang. (2012), Kelly (2014), and Beldad and Steehouder (2015), who all studied conversations in helplines.

To contribute to this research about improving user satisfaction with technical-support interactions, I report and discuss results from a systematic discourse analysis of both successful and unsuccessful helpdesk interactions and, based on this analysis, I recommend specific discourse strategies that can enable successful interactions. This study focused on the discourse strategies technical-support providers used for problem diagnosis and for problem resolution. Technical communication research has not focused on these discourse strategies; however, researchers report that communication facilitates technical-support providers reaching both problem diagnosis and problem resolution at certain stages in these interactions (Clark, Murfett, Rogers, and Ang 2012; Xu, Wang, Forey, and Li 2010). This study revealed how that communication works exactly, and it codified these strategies. Having explicit understanding of what these strategies look like and how they work, the findings from this study can help technical-support providers promote user satisfaction.

I analyzed these discourse constructs in 20 helpdesk interactions located within a higher-education context in which technical-support providers help faculty and graduate students use and succeed with teaching technologies, namely, a learning management system, an electronic portfolio system, and a content management system that enables users to contribute various teaching strategies and activities for sharing with other instructors.

This exploration, though located in a higher education setting, provides implications for businesses and industries like those explored by Xu, Wang,

Forey, and Li. (2010), Clark, Murfett, Rogers, and Ang (2012), Kelly (2014), and Beldad and Steehouder (2015). Helpdesk interactions in both higher education and business and industry settings require the same key service quality constructs described by foundational studies such as that conducted by Parasurman, Zaithamal, and Berry (1988), including those constructs' most recent iteration conducted by Burgers et al. (2000). These same service constructs, observed in business firms, guided my determination of satisfactory versus unsatisfactory interactions in my post-session survey, just as others have done with their surveys (van Velsen, Steehouder, and de Jong 2007). As mentioned previously, van Velsen, Steehouder, and de Jong (2007), who applied these constructs to helpdesk interactions, argue that person-to-person help facilities such as helpdesks or helplines are increasingly important for users because the information they impart is tailored to specific users and their specific needs. Specifically, Govindarajulu (2002) shows that users, including those who work in educational contexts, tend to prefer informal sources of help such as friends or colleagues rather than documentation. This finding agrees with other research that finds that users' value quick, empathetic, and customized help.

Asynchronous technical-support spaces such as those in web-based discussion forums provides yet another avenue for research of person-to-person help conversations, but as Swarts (2014) noted, documentation and discussion forums exist when "hands-on troubleshooting" (helpdesks or helplines) are "impracticable" (168). My study explored a context in which synchronous, person-to-person technical support does exist along side documentation. Furthermore, when these help facilities are not available (and perhaps only documentation is available), user satisfaction decreases. Islam (2014) found, for

instance, that instructors in a university expressed dissatisfaction with an educational technology system when they felt a lack of such technical support that would otherwise help them to do their jobs (255). Lee, Hsieh, and Ma (2011) corroborate this finding in their study of employees at three Taiwanese industry firms in which those employees' acceptance of an e-learning system depended on key technology acceptance constructs: perceived usefulness and intention to use. These constructs, they found, depended on organizational and managerial support, specifically support for using the technology. Their study suggests that even though businesses will implement technological systems to streamline employee productivity and enhance employee satisfaction, user (or employee) technology acceptance hinges on key organizational support factors such as a technical-support infrastructure. Echoing Lee, Hsieih, and Ma's (2011) finding, but in a university setting, Nawaz and Kahn (2012) stated that investing in the training and infrastructure of a help facility strengthens higher education goals because such investments serve and help the main users of the systems: teachers and students (42). Therefore, administrators of businesses concerned for their employees and their needs would agree with university administrators that by devoting attention to the support that employees receive in using a key organizational technology, they can further the goals of their organizations.

The Goal and Contribution of this Study

While many experienced or intuitively competent technical-support providers have developed an innate sense of how best to talk with users in helpdesk and helpline interactions, empirical research such as that conducted for this study can systematically codify that intuitive sense for the benefit of technical-support

centers such as the one in this study. This study contributes to the sparse technical-support research in technical communication by describing how technical-support providers not only adhere to a generic, or macrolevel, discourse structure (Xu, Wang, Forey and Li, 2010), but it also describes how technical-support providers use microlevel discourse strategies to achieve successful interactions. My study explored successful interactions and unsuccessful interactions for the absence and presence of these strategies, and it explored other microlevel strategies not already explored as empathetic discourse strategies have been (Clark, Murfett, Rogers, and Ang 2012). This study also contributed to the extensive conversation about technical support in other fields by lending a technical communication research angle to it. Lastly, by analyzing technical-support providers' discourse strategies, I demonstrate specifically how these workers engage in high-level knowledge work because of their extensive process of manipulating symbols and abstractions, that is, language (Das 2002; Pentland 2002). Demonstrating how technical support requires high-level knowledge processes can contribute to organizational communication studies that aim to counter claims suggesting that "support work," such as technical support, is somehow less important or less valuable than other kinds of work as Barely (1996) has shown.

Furthermore, my study showed how technical-support providers can use specific discourse strategies that will help them contribute to the organization's goals described by other researchers (Armestad, Keily, Hole, and Prescott 2002; Burgers, de Ruyter, Keen, and Streukens 2000; Callaghan and Thompson 2002; Lee, Hsieih, and Ma 2011). Uncovering these strategies can also help technical-support providers better serve their organizations' customer-users or employee-

users (Hall, Verghis, Stockton, and Goh 2014; Nawaz and Kahn 2012). Lastly, this study contributed back to my research site by enabling its technical-support providers (and its future technical-support providers) to communicate effectively with its users based on empirically grounded recommendations. From this perspective, this study existed in the tradition of technical communication research that contributes to the specific workplace practices of the research sites that the researchers explore (Bernhardt 2003; Palmer and Killingsworth 2002).

Conclusion and Overview of the Study

Informed by theories of discourse and by the literature I describe later, my research questions were the following:

- **RQ1:** In helpdesk interactions, to what extent do the interactions follow the established macrolevel structure?
- **RQ2:** In helpdesk interactions, how do technical-support providers and users communicate to diagnose problems?
- **RQ3:** In what stage(s) of the technical-support interactions do they diagnose them?
- **RQ4:** In helpdesk interactions, how do technical-support providers communicate to resolve problems?
- **RQ5:** In what stage(s) of the technical-support interactions do they resolve them?

To answer these questions, I conducted a discourse analysis of conversations in helpdesk interactions in order to describe the discourse strategies technical-support providers use. I defined satisfactory interactions as those in which both the technical-support providers and the users reported

satisfaction in a post-session survey. Based on my analysis, I recommend specific discourse strategies that can enable successful interactions. For my analysis, I use a discourse analysis coding scheme informed by my theoretical framework of technical-support interactions, informed by various, related streams of research literature, informed by my pilot study, and informed by rounds of interrater reliability coding as I developed the coding scheme used in this study.

This chapter has introduced the motivation, the theoretical framework, the focus and goals, and the research questions of the study. Chapter 2 synthesizes the literature on technical support and various related and relevant sets of literature. Chapter 3 describes in detail the research context, the data collection and analysis procedures, including reporting on the development of the coding scheme for the discourse analysis.

Chapter 4 reports and discusses the results answering research question 1, which relates to the textual function of discourse and the macrolevel analysis of the interactions. Chapter 5 reports and discusses the results for answering research questions 2 and 3, which relates to the ideational function of discourse. Chapter 6 reports and discusses the results from research question 4 and 5.

Lastly, chapter 7 concludes by qualifying my claims based on the limitations of this study, by recommending specific discourse strategies for technical-support providers to use in their helpdesk interactions, and by describing implications my study has for technical communication research. It also discusses future research possibilities and my plans for expanding on my study. In the next chapter, I introduce the relevant research in technical support, the sparse research on technical-support interactions, and the relevant research from one-to-one tutoring interactions.

CHAPTER 2: LITERATURE REVIEW

In this chapter I focus on research relevant to understanding the discourse strategies technical-support providers use in technical-support interactions. I review research about the variables that guide satisfactory technical- and customer-support interactions, as informed by technical- and customer-support literature; then I review research about how technical-support interactions operate at macrolevels of discourse; and lastly, I review research on both technical-support and tutoring interactions, which together shed more light on how technical-support interactions operate at microlevels of discourse.

Satisfactory Technical- and Customer-Support Interactions

Technical- and customer-support providers must meet specific measures of performance to demonstrate satisfactory work (Armestead, Keily, Hole, and Prescott 2002; Burgers, de Ruyter, Keen, and Streukens 2000). Among other issues, Armestead, Keily, Hole, and Prescott (2002) found from their case study of two support centers that managers evaluate support interactions based on two key performance indicators: efficiency and customer service, each of which entails both "hard" measures, such as number of sessions over time or customer survey data, and "soft" measures such as how a manager interprets the "interpersonal performance" of support providers (250). Researchers have found that measuring support efficiency is difficult because of the need to differentiate between types of support centers and a variety of human resources practices (Kinnie, Hutchinson, and Purcell 2000), such as what efficiency means in different organizations. While technical-support success does depend on managerial measures of efficiency, this study focuses on the "interpersonal

performance" described by Armistead et al. (2002), specifically the discourse strategies associated with that interpersonal performance.

Accordingly, satisfactory support interactions hinge on the client's perception of a support provider's attitudes and behaviors. These perceptions "build an image" of the organization (Burgers, de Ruyter, Keen, and Streukens 2000, 143; Bettancourt and Brown 1997). In an early study in customer service interactions, Bearden, Malhotra, and Uscategui (1998) proposed that three employee characteristics moderated the relationship between interpersonal contact and customer satisfaction—self-efficacy (the employee's belief that he or she can execute a work activity), empathy (the employee's ability to provide caring, individualized attention to a customer), and adaptability (the employee's ability to respond to the customer throughout the interpersonal interaction). This seminal proposition for how to explore customer service prompted other researchers to investigate the relationship between employee characteristics and customer satisfaction, usually through meta-analyses of personality test data that human resource managers administered in selecting new personnel. Such meta-analyses found that technical- or customer-support providers must retain personality characteristics coherent with a service orientation, an orientation involving politeness, responsiveness, friendliness, conscientiousness, agreeableness, and emotional stability—all positively related to high quality job performance in one-to-one interpersonal interactions (Frei and McDaniel 1998; Mount, Barrick, and Stewart 1998). These early studies use a psychological approach and rely on personality tests as the main source of understanding satisfactory support interactions. Specifically, this research measures personality constructs and discusses the validity of measuring such constructs, and in doing

so, it aims to inform managers about the kinds of personality characteristics that they should value when hiring new workers. This research focus does not, however, provide specific strategies that can help newly hired workers to operationalize their service orientation and facilitate a successful support interaction.

As an alternative to this applied psychology research, qualitative research studies have provided further insight into the kinds of attributes associated with successful support interactions. Building on meta-analyses like that conducted by Frei and McDaniel (1998), Callaghan and Thompson (2002) argued that meta-analyses of personality testing data rely on a "spread of data samples [that] covers a wide and often heterogeneous number of service work situations," many of which do not match the kinds of service interactions typical of technical- and customer-support (235). Indeed, studies like Frei and McDaniel (1998) pull data from many kinds of service organizations and do not provide specific insight into the technical- and customer-support interaction, even if they do provide general insight about the qualities of effective service providers. As an alternative approach to understanding the kind of characteristics support providers must maintain, Callaghan and Thompson (2002) tracked the recruitment, selection, and training processes of one support center, looking specifically for the kinds of employee attributes recognized and valued by management and the mechanisms by which managers selected and shaped those attributes. Using multiple interviews and ethnographic observations, they found that employees had to have a "positive attitude," "a sense of humour," and positive personality traits communicated through "verbal pitch, fluency, and energy and enthusiasm" (Callaghan and Thompson 2002, 240–41). Additionally,

successful characteristics included patience, tolerance, level-headedness, listening skills, and flexibility (Callaghan and Thompson 2002, 242). Such personality qualities, they found, differentiated effective support providers from those who only had product expertise, which managers saw as a "pre-requisite for the job" and not necessarily as exceptional (Callaghan and Thompson 2002, 241). While helpful in triangulating the personality testing data analyzed by other researchers, this case study focused on the attributes that employers value and shape in their employees, and it did not provide any specific strategies that workers can employ to operationalize those valued attributes in their communication. As such, the technical-and customer-support literature appears underdeveloped, reporting only on what managers say about how they train and recruit support providers and not on what occurs within the communication of these interactions.

Discourse Strategies in Technical- and Customer-Support Interactions

Few studies exist that actually look at the interaction for the kinds of discursive qualities that generate successful interactions. Research focusing on this discourse could provide insight into the kinds of discourse strategies that may be useful in training new technical-support and customer-support providers. Hall, Verghis, Stockton, and Goh (2014) asked proxy user to rate recordings and transcripts of the first two minutes of support interactions based on perceptions of the providers' courtesy and professionalism and also overall impressions of the providers' performances. They found that the raters successfully predicted how the original customers rated the providers. This research suggests that the first two minutes of a session are an important predictor of how the session

ultimately goes. This ability to predict satisfaction after only the first two minutes proves important in a support context in which calls must be handled efficiently because it communicates how satisfaction can be achieved soon in an interaction.

Nevertheless, this study ultimately repeats customer satisfaction surveys in retrospectively assessing an interaction to gain an overall impression of how it went. To counterbalance this focus on impressions gained after the interaction, the researchers also conducted a sentiment analysis using Linguistic Inquiry and Word Count (LIWC), a computer-aided text analysis software, on the transcripts of the interactions to quantify word usage associated with "negation, affect, positive emotion, negative emotion, sadness, tentativeness, certainty, assent, and nonfluency" (Hall, Verghis, Stockton, and Goh 2014, 504). They found that the providers' use of the first-person pronoun "I" positively correlated with satisfactory interactions (504–5). They speculated that using this personal pronoun meant that the technical-support providers were effectively functioning through their language as single agents of change in handling the customers' technical problems because every time they began sentences with "I," the providers spoke of what they could do, would do, or did, which these researchers argued likely pleased customers (506). These researchers found also that word usage associated with assent such as "uh-huh," "okay," and "yes" were negatively correlated with satisfactory sessions. They concluded that assent words are more often associated with interactions that are not going well, particularly with difficult or irate customers to whom assent may have been the only recourse the providers could take. However, studying language use was not the primary focus of this study.

While their microlevel analysis of particular language allowed them to draw conclusions about pronoun usage or assent words and phrases, the microlevel focus did not track specific types of microlevel language that demonstrates how specifically the support providers were helping customers. Moreover, it did not track the more global, macrolevel strategies, such as organization and coherence, associated with successful interactions, and the researchers concluded their study by challenging other researchers to consider how "linguistic usage may be a fruitful avenue" for customer-service research (506). Focusing on both macro- and microlevels of discourse could provide a fuller and more comprehensive study of "linguistic usage" in customer-service interactions, specifically in technical-support interactions, allowing researchers to identify what language features promote user satisfaction.

Macrolevel Discourse in Technical- and Customer-Support Interactions

The research in technical- and customer-support interactions tends to agree on the general macrolevel discourse of these interactions. Though not reporting within the technical- and customer-support literature, Agar (1985) presented a helpful macrolevel framework that provides a foundational description of what technical-support interactions look like. He describes a context of conversations he calls "institutional discourse" in which one person is an expert or person of authority affiliated with an institution and the other is not, the person he calls the client. Agar exemplifies this dynamic in courtroom or medical settings, but Mackiewicz and Thompson (2015) argued that this framework represents other interactions such as those in writing centers, which involve a tutor (an institutional representative) and a student (a client) (14). Agar's institutional

discourse concept applies well to a technical-support context, too, because a technical-support interaction involves a technical-support provider (institutional representative) and a user (client).

The three stages of institutional discourse conversations, Agar argued, are diagnosis, directive, and report. In the diagnosis stage, "question and answer play a central analytic role" (Agar 1985, 150). In this stage, the institutional representative usually engages in questioning the client, and the client answers—all with the purpose of fitting the "client's problem to the institution's frame" (Agar 1985, 150). That is, the representative brings the client into the institution's discourse, trying to fit the client's ways of thinking about the problem into that of the organization. If a client expresses experiencing a certain problem, the representative questions the client to learn more about the context and events that led the client to find the issue, gradually determining "which [institutional] frames are relevant" to the client's frames of understanding. This understanding will later inform the directive stage (Agar 1985, 150). The directive stage grounds institutional discourse contexts in which the client goes to the representative to figure out what to do about the problem that was diagnosed; directives may not occur in other contexts such as courtrooms (Agar 1985, 156). In healthcare contexts and others in which clients seek directives, a "lack of directives might be a client complaint" (155). In the reporting stage, the institutional representative summarizes the diagnosis and directives stages in order to archive the interaction for the institution (Agar 1985, 154). His three stages, while broad, do sketch the macrolevel structure of technical-support interactions and can influence a macrolevel analysis of a technical-support interaction. This

framework provides the initial stages that others examining institutional discourse contexts can use for their discourse analyses.

Moving beyond Agar's (1985) framework, studies of technical- and customer-support interactions have demonstrated a relatively consistent structure to them. Baker, Emmison, and Firth (2005) identified seven main phases through their analysis of 120 hours of technical-support interactions for a software firm: opening phase, problem analysis phase, diagnosis phase, solution phase, instruction phase, evaluation phase, closing phase. Steehouder and Hartman (2003) and Steehouder (2007) reappropriated this phase structure in their close analyses of a transcript. Clark, Murfett, Rogers, and Ang (2012) analyzed 289 stressful support calls and identified the macrolevel structure for the sessions they analyzed. They defined "stressful" calls as those "in which the agent was under social stress as a result of caller aggression or ambiguity" (127). They identified six "phases" in these calls: greeting, identifying, defining, negotiating, resolving, and closing (128). Similarly, Xu, Wang, Forey, and Li (2010) investigated 100 support calls using a quantitative genre analysis and found five "moves" in their corpus: greeting, purpose, information, service, and farewell (458–59). The macrolevel components identified by Agar (1985), Clark, Murfett, Rogers, and Ang (2012), and Xu, Wang, Forey, and Li (2010) are visually compared in Table 2.1.

Table 2.1 Comparison of Macrolevel Discourse Units

Agar (1985) Institutional Discourse Stages		Baker, Emmison, Firth (2003); Steehouder (2007)		Clark, Murfett, Rogers, and Ang (2012)		Xu, Wang, Forey, and Li (2010)	
		Opening	Provider greets and identifies the customer	**Greeting**	Provider states a standard greeting	**Greeting**	Provider states a standard greeting and invites the customer to relate the problem
				Identifying	Provider identifies customers based on ID numbers		
Diagnosis	Representative asks questions to understand the problem	**Problem Analysis**	Provider invites customer to describe a narrative about the problem, including questioning the customer	**Defining**	Provider invites customers to relate the problem	**Purpose**	Provider relates the problem and worker asks for clarification if needed
		Diagnosis	Provider announces what the problem is	**Negotiating**	Provider responds with questions to understand the problem	**Information**	Provider checks the customer's information and asks questions

Table 2.1 continued

Directives	Representative gives solutions	**Solution**	Provider announces the solution	**Resolving**	Provider resolves the issue	**Service**	Provider gives information and solutions
		Instruction	Provider gives instructions to the customer				
		Evaluation	Provider invites client to announce the problem has been resolved				
		Closing	Provider and customer exchange goodbyes and thank you's	**Closing**	Provider confirms the issue and any other issues are resolved and thanks the customer	**Farewell**	Provider confirms the issue and any other issues are resolved and thanks the customer
Reporting	Representative documents the diagnosis and its directive for archiving						

Within the opening (blue in table 2.1) macrolevel discourse, the only difference (apart from the naming) between the components identified by Baker, Emmison, and Firth (2005), Clark, Murfett, Rogers, and Ang (2012) and Xu, Wang, Forey, and Li (2010) is that Clark, Murfett, Rogers, and Ang provided an additional "identifying phase," which Xu, Wang, Forey, and Li included in their "information move" and which Baker, Emmison, and Firth included in the "opening phase." Agar (1985) did not discuss greetings or opening discourse.

Within the defining (orange in table 2.1) macrolevel discourse, the only difference between the researchers (apart from naming) is the way they divide up the discourse. Baker, Emmison, and Firth (2005) outlined a problem analysis phase followed by an announcement of the problem (diagnosis). Clark, Murfett, Rogers, and Ang (2012) had two phases (defining and negotiating) that could be collapsed into Baker, Emmison, and Firth's problem analysis phase. Furthermore, Clark, Murfett, Rogers, and Ang's phases imply an announcement of what the problem is. Lastly, Xu, Wang, Forey, and Li (2010) collapsed all of this discourse into one move—purpose. Agar's diagnosis stage spaned the phases and moves of the other researchers.

Agar's directives stage spans the purpose and information moves of Xu, Wang, Forey, and Li and the solution, instructions, and evaluation phases of Baker, Emmison, and Firth. Agar's directives stage matches the resolving phase of Clark, Murfett, Rogers, and Ang, and it also matches the service phase of Xu, Wang, Forey, and Li. Both Xu, Wang, Forey, and Li and Clark, Murfett, Rogers, and Ang discussed one discourse moment that Baker, Emmison, and Firth split into three distinct moments (solution, instruction, and evaluation).

Lastly, all researchers, except for Agar, who does not discuss closings or farewells, have a common closing moment to their analysis. Agar alone discusses a reporting stage. Thus, the combination of Agar's (1985) foundational institutional discourse model and the three more-detailed support interaction models provide ample groundwork for how to classify the macrolevel discourse moves of technical-support interactions.

This overview of the macrolevel discourse research for customer- and technical-support interactions reveals that the macrolevel discourse is relatively well-established. Xu, Wang, Forey, and Li (2010) did note, however, that their moves vary across interactions because of variables such as products or services involved in the interaction, customers' purposes in calling, interlocutors' language proficiencies and communicative competencies, interactional dynamics, cultural differences, and social factors (465). Variations in the macrolevel structure include optionality (some moves are omitted) and iteration (some moves are repeated) (465). The extent to which variations of this macrolevel structure exist across sessions remains unknown. Furthermore, Clark, Murfett, Rogers, and Ang (2012) and Xu, Wang, Forey, and Li (2010) concentrated on customer-support interactions more broadly and not on technical-support interactions specifically (interactions involving a provider helping a user with a technical issue). Baker, Emmison, and Firth (2005), who did explore technical-support interactions, do not differ from the other researchers, except for having an instruction phase. This phase may be the contextual difference between technical- and customer-support interactions. A study of technical-support interactions should account for and discuss variations in

generic structure based on the technical context of the session. Thus the first research question of this study was as follows:

> **RQ1:** In helpdesk interactions, to what extent do the interactions follow the established macrolevel structure?

A study of technical-support interactions should also examine microlevels of discourse in order to provide a fuller range of insight and a more comprehensive study of linguistic usage, especially because the literature on microlevel discourse in technical-support interaction agrees that strategies within this level of discourse can facilitate satisfactory interactions. Thus, the next section describes what the existing technical support literature states about microlevel discourse in technical-support interactions.

Microlevel Discourse in Technical- and Customer-Support Interactions

Few studies look closely at the microlevel discourse in technical-support interactions, but those that do look at this level agree that specific language strategies can facilitate satisfactory interactions. Pentland (1992) proposed that a "move" provides a viable unit of analysis for situated, face-to-face interactions such as technical-support interactions because a move provides a discursive marker of one situation transforming into a new situation. Thus, moves provide more insight about the discourse than sentences do (530). Pentland, however, does not necessarily mean the "move" of move-step analyses in which global, macrolevel units of discourse are identified in order to reveal specific conventions of a given genre, or type, of discourse (Swales 1990). Instead, Pentland (1992), invoking Goffman's (1981) "replies and responses" framework, argues that moves are meaningful linguistic and nonlinguistic actions, like those

in a game, that demonstrate willful agency in response to a situation (530). Thus, Pentland demonstrated how technical-support providers "move" within structures that control and inhibit the possibilities and potentialities of those moves, ultimately equating those inhibiting structures to the workers' organizational structure (533). This combination of Goffman's "move" and Giddens' "structures" (1984) provides Pentland his theoretical framework for analyzing the ways technical-support providers respond to their users successfully within the structures of the workers' organization. Thus, Pentland calls these discourse markers "organizational moves," which technical-support providers enact when they "draw on […] resources necessary to respond" to users, namely, specific material or social resources available to them within their organization (528). Even when providers lacked the technical knowledge necessary to help customers, for example, they would still get help themselves or refer the users to some other resource within the organization. This "knowledgeable performance" demonstrated providers' ability to solve problems and to find answers, a performance that serves to help the providers to "organize work" (Pentland 1992, 528). This framework draws from the organizational communication tradition because it highlights the specific organizational context in which these workers discursively constitute their work.

According to Pentland, organizational moves consist of two major types: "get help" moves and "give away" moves. Get help moves mean the technical-support providers asked a technical expert such as an engineer or a designer to explore the problem, either through "quick question" or "take a look" moves. In "quick question" moves, the providers asked another provider or expert to confirm an idea or to clarify it. Pentland called it a quick question move because

it served as one moment of interruption in the interaction in which the support providers needed to fill his or her own information gaps by seeking out others in the organization. The relative speed of quick question moves was contrasted by longer "take a look" moves, which demand a deeper commitment from others in the organization. With this move, the providers asked another provider or expert to look at or review an issue or even to serve as a third member in the interaction.

"Giving away" moves demonstrated the providers' ability to find answers within their organizational structure through processes of "assigning," "referring," "transferring," or "escalating." These moves typically ended the interaction because the entire interaction was given away to another person. The apparent synonymy of Pentland's (1992) subcategories of giving away moves reflects Pentland's specific organizational and call-center case study (e.g., transferring means the user was transferred to another department while assigning means the user was assigned, based on a call-screening procedure, to another department or worker even before the interaction with a technical-support provider began). Another give away move was the technical escalation, where the providers referred the users to a manager because the providers lack the technical knowledge or technical authority (e.g., permissions and access to a system). Lastly, Pentland described the political escalation move in which the providers referred the users to a manager because the users were irate or had a high organizational position that made their problem more politically acceptable for the manager to handle.

Certainly these processes of "getting help" or "giving away" reflect the technical-support providers' organizational knowledge through the specific

strategies they employ, but this organizational behavior focus did not include the kinds of personality constructs and performances associated with satisfactory support interactions as found in the support and customer service literature, such as friendliness, empathy, or self-efficacy. Furthermore, Pentland, while stating that he was doing discourse analysis, seemed more to focus on specific "organizational performances" the providers used in the interactions and did not seem to focus on discursive performances, or pragmatics (speech acts), the technical-support providers used to accomplish goals in the interaction such as diagnosing the problem or resolving the problem. Thus, Pentland's study, while built on similar theoretical principles as those underpinning discourse analysis, does not quite function as a discourse analysis in itself and provides minimal help to language and communication researchers for exploring the microlevel discourse at work in these interactions. It does provide insights about the contextual work practices of technical-support providers, and these insights can help researchers interpret the discourse in light of those work practices.

Current research largely uses some variation of conversation analysis, which focuses on speaker-to-speaker turn-taking, to explore specific discourse constructions such as technical competence or miscommunication (Baker, Emmison, and Firth 2005; Beldad and Steehouder 2015; Kelly 2014). This focus on specific discourse constructs lends helpful insight into the nature of these interactions, but it largely avoids a discussion of the key moments of the interactions (problem diagnosis and problem resolution), providing little insight into how technical-support providers and users diagnosis problems or resolve technical problems. Only two studies focus on the speech acts within the interaction. Clark, Murfett, Rogers, and Ang (2012) analyzed 289 stressful

support calls for empathetic communication strategies. For each communication strategy, they also described what they called "inhibitors," which worked against the intentions of the empathetic communication strategy.

Attentive empathy means the customer-support providers engaged in "active listening, such as acknowledging, repeating, paraphrasing, elaborating on the customers' ideas, summarizing, and asking questions" (Clark, Murfett, Rogers, and Ang 2012, 134). This process faced inhibitors such as surface listening or feigned attention. Affective empathy means the customer-support providers identified with the customers' emotions and responded with the same emotion through apologies or by referring to the experiences of other customers (Clark, Murfett, Rogers, and Ang 2012, 137). This strategy faced inhibitors such as customers' doubting the providers' sincerity or the organization or customers' viewing such empathy as inappropriate for the context. Lastly, the researchers identified cognitive empathy, which counterbalanced empathy's emotional component with intellectual categories such as providing specialized terminology when the customers did not have it, proposing potential options the customers can take, stating what other customers have done in similar situations, or anticipating what customers may really mean when they ask questions (Clark, Murfett, Rogers, and Ang 2012, 138). Because of this anticipatory (perhaps officious) disposition, misdiagnosing customers' needs was this strategy's biggest inhibitor.

In determining when it was most appropriate to use these strategies, Clark, Murfett, Rogers, and Ang (2012) identified eight actions indicative of customers' needs for empathy: invites empathy, disinvites empathy, expresses discontent, demonstrates misunderstanding, repeats the customers' concerns,

asks the providers to repeat, requests affirmation, and criticizes service (141–42). These researchers found that affective empathy was the least used in support calls, and when it was used, the customer hastened the conversation to the core problem motivating the conversation to begin with—either because the customers were uncomfortable or because the customers simply preferred to keep the conversation on point (Clark, Murfett, Rogers, and Ang 2012, 143–44). As a corollary, their finding about affective empathy in support interactions means that the affective empathy strategy may work against resolving the problem, evidenced by its potential to turn the interaction off-topic, and also it may work against customer satisfaction as a result. The researchers thus recommended that providers use this discourse strategy with caution, especially if their job performance depends just as much on efficiency as it does on customer satisfaction. Clark, Murfett, Rogers, and Ang (2012) ultimately concluded that attentive and cognitive empathy strategies have greater affect on customer satisfaction.

Their study provides a rich groundwork on which to build future studies of the discourse strategies associated with satisfactory support interactions, but their study seems heavily reliant on empathy as a key construct. One reason for this focus on empathy may be that they focused on "stressful calls," meaning that they specifically excluded more routine calls in their data collection, assuming that customer aggression or ambiguity were prerequisites for expressions of empathy to appear (126–27). This approach invites further research on more routine calls just as technical-support interactions may be instigated by routine technical problems, and technical problems may mean user aggression, though not necessarily. While these researchers rightly position their study in the

research conversation about technical- and customer-support, stating that empathy is a key personality marker of successful support providers (Bordoloi 2004; Burgers, Ruyter, Keen, and Streukens 2000; D'Cruz and Noronha 2008; Dorman and Zijlstra 2003; Pontes and Kelly 2000), they do not seem to account for providers' abilities to instruct customers about what to do. Specifically, instances in which customers are unsure about what they need may come from unique circumstances associated with users' expertise levels with the technology, meaning that they may not know what they do not know and thus require instruction. In response, the support providers must employ specific instructional discourse strategies to help the users to arrive at a shared vocabulary and understanding, including giving recommendations that make sense to the users. Clark, Murfett, Rogers, and Ang (2012) agree with this technical knowledge dynamic to the interaction:

> But the customer's role in getting service is not demand free. Customers need sufficient understanding to communicate their reason for calling and to interpret the agent's response, which may require some technical knowledge about the products or services at issue. A customer who does not understand may experience a loss of face. (131)

This technical knowledge dynamic demonstrates that users need instruction, and this instructional discourse is one component of effective technical-support interactions. Clark, Murfett, Rogers, and Ang (2012) did not quite uproot the discourse strategies support providers would use to help users learn and apply technical knowledge, but they did invite other researchers to explore this instructional component, which is where this present study fit.

Another study explored the microlevel discourse strategies in technical-support interactions. Steehouder (2007) builds off Baker, Emmison, and Firth's (2005) study of the phases in the interaction and the way that technical-support providers and users discussed one another's technical competence. To do so, Steehouder (2007) explores the discourse in each phase and overviewed his interpretations of the microlevel discourse in each phase. Namely, he reported on one interaction and structured his report by giving an excerpt of a phase followed by his interpretation of the discourse inside of that phase. This approach does not fully describe the specific instructional strategies that these technical-providers use. However, he did conclude his report with an overview of key features of the instruction: chunking the instruction into steps, using modal verbs with imperatives ("you may click…), using metaphors for human-computer interaction ("go to…" or "choose…"), and listening for referential installments ("Do you see a tab?"). These overviews are helpful indications of the kind of discourse technical-support providers may use when instructing users, but the presentation in this research is too brief and more importantly lacks a systematic approach for how these discourse moments were identified and how reliably they may be used across helpdesk interactions in other contexts (Steehouder reports only on one interaction and discusses not coding scheme development process). Therefore, further study is needed on the ways technical-support providers instruct users.

In summary, technical-support research that examines specific microlevel discourse strategies in an interaction focuses on either organizational knowledge and specific discourse constructs within the turn-taking for each speaker. Speech act analyses have identified convincingly the empathetic communication

strategies that customer-support providers have used. Because customer-support in general differs in context from technical-support (specifically the lack of technical instruction), further study could identify these strategies associated with technical problem diagnosis and resolution. The only existing examination of this instruction did not present a full or systematic analysis. Thus, borrowing from the research in tutoring interactions could reveal more about how this instruction may happen in a technical-support context. The next section describes the specifics of this research in order to lend more insight about the potential microlevel discourse strategies that could facilitate successful technical-support interactions.

Microlevel Discourse in Tutoring Interactions

The research in writing tutoring agrees that the questions that tutors ask of writers can help facilitate the session positively. Thompson and Mackiewicz (2014), modifying a framework from applied psychology (Graesser, Person, and Huber 1992), explored the questioning types that experienced writing center tutors use when helping students with their writing. They argued that questions serve as "prompts for learning and for maintaining students' engagement" (2014, 38). One fundamental difference between writing center tutoring and technical support is that writing center tutors often engaged in "cognitive scaffolding," in which "tutors give students opportunities to figure out what to do on their own" (Mackiewicz and Thompson 2014, 56). In other words, writing center tutors often employed questioning "to help students formulate explanations for themselves" because writing center theory asserts that students can better apply their learning on their own if they arrive at explanations for themselves (Mackiewicz and

Thompson 2014, 61). Technical-support providers may not employ "leading-scaffolding" or "known-information" questions designed to get the users thinking in this way (Mackiewicz and Thompson 2014, 61; Thompson and Mackiewicz 2014, 43). The customer just wants to get the problem solved (Callaghan and Thompson 2002, 245). As such, Mackiewicz' and Thompson's (2014) "leading-scaffolding" question types do not seem to apply to the purposes of technical-support interactions.

Besides the "leading-scaffolding" question type, however, Thompson and Mackiewicz (2014) described "knowledge-deficit" questions in which tutors asked questions that helped the tutor to gain accurate knowledge of the students' writing needs, such as specific details in the student's work about which the tutor would not know without asking (42). They also noted that tutors asked questions to establish common ground with students, such as to better understand what the student already knew about writing, about the writing topic, or about their assignment. Tutors also asked such common ground questions to help themselves better understand the writing assignment, the students' goals for the conference, and the students' understanding throughout the conference session (42). Thompson and Mackiewicz (2014) also found that tutors asked questions to coordinate the social dimension of the interaction, including asking questions to request students to read their writing aloud and to provide indirect advice in question form (43). Lastly, they found that tutors asked questions to control the conversation by starting and closing the session, by shifting the topic, and by employing rhetorical questions meant to instruct the students (43). This discourse research in tutoring interactions lends the kind of specific microlevel strategies that may occur in technical-support interactions,

especially at the diagnosis stage described by Agar (1985), or the similar stages found by Xu, Wang, Forey, and Li (2010) and Clark, Murfett, Rogers, and Ang (2012). At these stages, these researchers consistently agree that the institutional representatives (the technical-support providers) question the clients to better understand what they need. Nevertheless, questioning may not be exclusively tied to the diagnosis stage, even as Clark, Murfett, Rogers, and Ang (2012) claimed that the support provider may engage in questioning in the closing stage, and Xu, Wang, Forey, and Li (2010) claimed that questioning may occur at the farewell stage. These questioning types presented from tutoring research thus provide a helpful starting point for analyzing the questioning technical-support providers may employ when facilitating an interaction. However, questioning occurs with the intention of diagnosing a problem, and the current empirical research in technical- and customer-support remains unclear if questioning is the only strategy the technical-support providers use with the intent to diagnose a problem. As such, the second and third research questions of this study remained both open to other kinds of problem diagnosis strategies and attentive to the questioning types identified by Thompson and Mackiewicz (2014). The second and third research questions were the following:

> **RQ2:** In helpdesk interactions, how do technical-support providers and users communicate to diagnose problems?
>
> **RQ3:** In what stage(s) of the technical-support interaction do they diagnose them?

Writing tutoring research also notes the positive correlation between tutors giving directives and student experiencing satisfaction with the interaction (Kiedaisch and Denitz 1993; Thompson, Whyte, Shannon, Muse, Miller,

Chappell, and Whigham 2009; Mackiewicz 2004; Murphy 2006). Agar (1985) also noted that a lack of directives can lead to client dissatisfaction (155). Thus, writing tutoring research has explored the language by which tutors give those directives. Technical-support research has not done as much work on this feature of the interaction. Clark, Murfett, Rogers, and Ang (2012) described specific strategies workers might use to portray empathy with customers and describe a kind of empathy they call cognitive empathy, which as Clark, Murfett, Rogers, and Ang (2012) defined it, is an instruction-like empathy more associated with the intellect. They define cognitive empathy as "intellectually assuming the other person's perspective while retaining sufficient judgment to helpfully intervene" (137). They state that they saw this form of empathy expressed when workers provided language that the customer needed but could not find, when workers proposed options for eventualities that the customer might face, and when workers stated what other customers had done in similar situations. While these strategies seem useful as a form of empathy focused on the customer's intellectual categories and how workers can best meet them, the term "cognitive empathy" itself seems slippery and difficult to grasp. Also, the intellectual needs of customers seem relegated to three categories that are not comprehensive enough for discussing how the support providers help the customers understand technical concepts. For example, this scheme does not address the specific strategies the support providers employ when talking to customers so that customers understand what they are told. Thus, this scheme needs clarification from the literature in one-to-one instructional interactions, specifically writing tutoring research.

Accordingly, the research in writing tutoring agrees that specific kinds of instructional discourse facilitate successful tutoring interactions. Mackiewicz and Thompson (2014) argued that instructional strategies typically include directives about what to do with more or less uses of mitigation. They argue that when telling students what to do, tutors must ensure their directives still allow the student to "save face," that is, that they avoid making the students feel challenges to their autonomy and competence as writers or students, a construct Mackiewicz and Thompson (2014, 59) adopt from politeness theory (Brown and Levinson 1987). Thus, instructional strategies may be what they call "telling," which is a directive with little to no mitigation, or "suggesting," which is a directive with more mitigation (Mackiewicz and Thompson 2014, 60). By mitigation, they meant the kinds of microlevel linguistic strategies that create a longer inferential path between what is meant and what is said (Blum–Kulka 1987). The longer the path to the intended meaning, the politer (more mitigated) the language is. Politeness strategies that mitigate the directive in this way may use downgraders such as subjectivers ("I think…"), hedges ("maybe"), downtoners ("possibly"), appealers ("…, O.K.?"), cajolers ("You know...") and understaters ("You should just…")—all strategies meant to ensure that the students' autonomy or competence does not feel threatened by the advice (Mackiewicz and Riley 2003, 85–6). Another example of mitigation is when speakers use "conventionally indirect" strategies by making the listener's obligation less clear by employing low-value modal verbs such as "can" or "could," giving the listeners a sense that they may (but need not) heed the speakers' recommendations and should use their own discretion (Mackiewicz and Riley 2003, 89–90). Other examples of suggesting could be

"nonconventionally indirect," which Mackiewicz and Riley (2003) equate with hints (90). They argue that strong hints state an observation of a writer's work without specifically recommending any change to it, while weak hints give general rules of writing without indicating that the writer has or needs to follow that rule (91). Hints are particularly difficult to identify in discourse data because "[h]ints are vague. . .[and] allow for several potential meanings" (Mackiewicz and Thompson 2015, 56). Hints consist of words (locutions) that do not match their underlying speech acts (illocutions). In both writing tutoring and technical-support interactions, the listeners "may miss [hints] altogether," and experienced tutors (and perhaps by extension experienced technical-support providers) may eschew hinting in favor of other strategies entirely that listeners can more easily identify (Mackiewicz and Thompson 2015, 117; Mackiewicz 2005). Indeed, Mackiewicz and Thompson (2015) found hinting occurred rarely.

The writing tutoring research also remarks that tutors may employ longer form instructional discourse than just telling or suggesting. For example, Mackiewicz and Thompson (2014) also described an "explains and exemplifies" instructional strategy, in which "[t]utors offer reasons for and illustrate their advice" (Mackiewicz and Thompson 2014, 61). Elsewhere, this advice-giving strategy has been described as using "payoff statements" that offer justifications for the advice, which gives the writers an "explicit benefit of complying" to advice and thus explains what that advice means to the writers specifically (Mackiewicz and Riley 2003, 88). This discourse research in tutoring interactions imply what specific microlevel strategies may occur in technical-support interactions, especially at the directives stage described by Agar (1985), or the similar stages found by Xu, Wang, Forey, and Li (2010) and Clark, Murfett,

Rogers, and Ang (2012). At these stages, these researchers consistently agree that the institutional representative (the technical-support provider) gives suggestions and instruction based on the diagnosis. The advice-giving strategies found in tutoring interactions may be apparent in technical-support interactions as well, and analyzing these kinds of strategies can reveal more about instructional discourse in technical-support interactions. However, because directives serve the purpose of resolving the problem, other discourse types (not just directives) may appear that may work toward the same purpose. For this reason, my fourth and fifth research question existed in anticipation that directives may occur toward this purpose (as informed by the writing tutoring research), but it did remain open to finding other kinds of problem resolution strategies. Accordingly, the final research questions for this study were the following:

RQ4: In successful helpdesk interactions, how do technical-support providers communicate to resolve problems?

RQ5: In what stage(s) of the technical-support interaction do they resolve them?

Conclusion: Toward a Study of Technical-Support Interactions

In summary, this literature review has shown that technical-support research has gradually grown interested in studying the discourse present in the interactions. This trend marks a growing interest in the discourse strategies providers can employ to facilitate user or customer satisfaction. While the overall macrolevel discourse of technical-support interactions appears well-established, how and why variations of this macrolevel structure exist across interactions and contexts

remains unknown. Studies on the microlevel discourse of technical-support interactions, however, are sparse and also tightly focused on strategies such as organizational knowledge performances ("organizational moves"), technical competence, miscommunication, and empathetic communication. My research builds on these studies by exploring specific microlevel discourse strategies technical-support providers can use to facilitate satisfactory interactions. Thus, the one-to-one instruction research, specifically in writing tutoring research, provides additional microlevel discourse insights such as questioning and instructional directives, and these discourse types were useful starting points for studying the microlevel discourse of technical-support interactions, especially because of the similarities between technical-support interactions and one-to-one tutoring interactions. As a result of this literature and the current gaps in the research, this study of technical-support interactions needed to answer the following research questions:

- **RQ1:** In helpdesk interaction, to what extent do the interactions follow the established macrolevel structure?
- **RQ2:** In helpdesk interactions, how do technical-support providers and users communicate to diagnose problems
- **RQ3:** In what stage(s) of the technical-support interactions do they diagnose them?
- **RQ4:** In helpdesk encounters, how do technical-support providers communicate to resolve problems
- **RQ5:** In what stage(s) of the technical-support interaction do they resolve them?

In this next chapter, I describe my methodology and specific procedures for answering my research questions, including a description of how I have iterated a coding scheme from the foundational discourse constructs introduced in this chapter into a more reliable and relevant coding scheme for analyzing the discourse in technical-support interactions.

CHAPTER 3: METHODS

To examine the macrolevel and microlevel discourse in technical-support interactions, I employed a discourse analysis of 20 technical-support interactions. In this chapter, I describe how I analyzed these interactions, and this analysis allowed me to relate specific discourse strategies to user satisfaction. With this method, I help fill the overarching gap in the literature, which does not comprehensively explore this relationship.

Prior research has revealed the macrolevel structure of these interactions, but the extent to which that specific macrolevel structure is followed across contexts remains unclear (Xu, Wang, Forey, and Li 2010). To help further explore the influence of certain contexts on this macrolevel structure, I explored an additional specific context—a higher education context in which users seek help with educational technologies that help them administer the courses they teach. This study explored, in particular, the macrolevel structure of technical-support interactions between instructors and a team of providers that supported these instructors. By determining the macrolevel structure of these interactions, I explored the ways in which the interactions adhered or differed from the established structure previously found in studies of customer-support interactions more broadly (Xu, Wang, Forey and Li 2010; Clark, Murfett, Rogers, and Ang 2012). As Xu et al. (2010) state, exploring "institutional or domain differences" is essential for analyzing professional discourse (466). As they explain, "[t]he technical or professional expertise required and the relevant information needed vary significantly for different products or services, and so do the talk exchanges" (Xu, Wan, Forey, and Li 2010, 466–67). These institutional

and domain differences influence the macrolevel structure for technical-support interactions (465). The domain differences they described include the products or services involved, the customers' purposes for seeking help, both participant's language proficiencies and communication skills, interactional dynamics, cultural differences, and social factors. My study explored the macrolevel discourse of these interactions in a different context than the customer-support call centers they explored, providing unique results to the current research.

Whereas prior research provided a relatively established knowledge of these interactions' macrolevel structure, past studies did not address the microlevel discourse strategies at work within satisfactory interactions' macrolevel structure. Understanding these microlevel discourse strategies not only provides a comprehensive knowledge of technical-support interactions but also provides technical-support providers and their managers language practices they can use to maintain customer satisfaction.

The following sections describe the selected research site, the research procedures and instruments employed in the study, and the data-analysis procedures, including the development of the discourse coding scheme.

Research Site and Participants

The setting for this study was the English department at a large, Midwestern university in the United States. In this department, three main technical systems facilitated the instruction for English courses: a learning management system, an electronic portfolio system, and a teaching strategies depository system—all three of which were built and designed on the open-source platforms Moodle and WordPress. The systems were highly customized for the department because

open-source platforms allow for malleable design to fit specific contexts. Though communication courses in professional writing such as business and technical writing used only one of these systems (the teaching strategies depository), instructors of courses such as composition, literature, and English as a second language could take advantage of all three systems, based on instructors' individual preferences.

Technical-Support Provider Participants and Recruitment

Five technical-support providers supported the instructors who used these systems, and they each had some role in building and designing the systems. The technical-support providers came from diverse technical backgrounds and experiences with these technologies, and all five members had teaching experience in the courses that used the systems. Furthermore, these providers were all graduate students, who used the technical-support provider position as a graduate assistantship. The support team's supervisor, affiliated with the larger university online-learning initiatives, agreed to the study, and after receiving approval for this study from the Institutional Review Board (IRB) at my university (see appendix A), I recruited each technical-support provider individually.

All five technical-support members agreed to participate. After the pilot study, three participants left the team. My study's IRB permissions allowed me to recruit new members for participation, so I recruited the three additional hires in their place for the main study. In total, this study used the strategies from 6 of these providers because 2 of the providers' strategies were only explored in the

pilot study, which occurred before they left the team. I discuss more information about those who participated in a later section.

User Participants and Recruitment

Users of these technical systems were English course instructors from various classifications: full-time and part-time faculty, lecturers, and graduate students on a teaching assistantship. These instructors specialized in technical and professional communication, composition, applied linguistics, teaching English as a second language, creative writing, English education, or literature. User enrollment entailed users' assent to the data collection procedures I describe later in this chapter. I told users that they could use their own or the support team computers to facilitate their support sessions and that they should only visit the support office for genuine technical problems they faced. Namely, I tried to convey that they should not feel obligated to visit the support office simply because of their enrollment in the study. Nevertheless, many users during enrollment indicated that they had intention to visit the support office at least once. Users who did not enroll specifically said that they had no intention to visit. I discuss more information about those who participated in a later section.

Technical Systems at the Research Site

As mentioned previously, the department uses an open-source learning management system (Moodle) to administer course content for many of its courses. Users could visit the helpdesk for this system to address issues related to course and instructional design, system features, and system procedures. The department also used an open-source electronic portfolio system (ePortfolio) built on WordPress. The system allowed students to build online portfolios of

their work. During the study, the support team also developed a secondary use for the ePortfolio system, which they called "eProfiles," which allowed instructors and students to create professional profile websites for professional development purposes. The technical-support providers also supported this system. Because instructors had to help students use the electronic portfolio and profile systems, instructors could seek help in effectively employing it in their courses or for their own professional development. Lastly, the department used a depository system to which instructors could submit or retrieve teaching activities specific to English courses. Instructors could seek help for how to navigate the system to retrieve teaching activities or for how to submit their own activities.

The systems were all built on open-source software. Open-source software allowed the designers and developers to highly customize the design for the specific instructional and technological needs of the contexts. Certain technical-support providers led the design and development of these systems, and others continued the design and development from support providers before them who left the team. The design and development these providers accomplished fluctuated based on their individual experiences with the technologies and with the process of web design and development. Also, while the English department used the systems, university servers hosted the systems, and the university information technology staff administered them. The technical-support team thus led the design, development, and support for the systems but did not lead the server or security administration. An administrator supervised the team, and this administrator affiliated with the university-wide online learning initiative and not the English department. Ultimately, because all systems helped the

department's English instruction to function, discourse in technical-support interactions could relate to all three systems, and the interaction could vary in purpose and goals.

Research Methodology

To answer my research questions, I employed a discourse analysis of the conversations between these users and technical-support providers. Discourse analysis is "an approach to the analysis of language that looks for patterns across texts as well as the social and cultural patterns in which the texts occur" (Paltridge 2012, 1). These "social and cultural patterns" describe the context in which participants share an understanding about the expectations and definitions of an interaction. Discourse analysis studies the language in use throughout a meaningful exchange in which participants accomplish communicative goals. Two main overarching approaches to discourse analysis are the "textually-oriented approach" that looks closely at language structure and purposes and the "social-theoretical approach" that uses critical and social theory to interpret language (Paltridge 2012, 6). Discourse analysts do not see these approaches as mutually exclusive or incompatible, especially because they agree that language cannot be interpreted apart from its social and cultural situations. However, researchers employing discourse analysis usually focus on one approach as they attempt to answer the questions they have about a given discursive exchange. Scholars such as Fairclough, Van Djik, and Gee, for example, describe a "social-theoretical approach" to discourse analysis that explores how participants use language to create identity, perform cultural functions, and inscribe or challenge the social constraints that form and shape

their exchanges. Scholars in technical and professional communication who employ this approach to discourse analysis typically want to explore how participants form identity or exercise or respond to social control through discursive interactions in technical workplaces (e.g., Barton 2002; Faber 2007). Discourse scholars such as Sacks, Yule, Swales, Levinson, Halliday, Searle, and many more, use the "textually-oriented approach" that explores the structure and goals of language through varying levels of close textual analysis. Scholars in technical and professional communication who appropriate discourse analysis that is textually-oriented often explore how technical and professional communicators accomplish goals with their language or cooperate in conversational expectations (Beldad and Steehouder 2015; Mackiewicz 2010). Adopting this textually-oriented approach to discourse analysis, I employed speech-act analysis of illocutionary acts to explore the actions that participants accomplish when they cooperate in an exchange (Brown and Yule 1983; Searle 1976), because my research questions directed me to employ an analytical tool that explores how language was used to accomplish goals or actions. By employing discourse analysis and this specific approach to discourse analysis, I could identify specific linguistic actions technical-support providers and users took in their interaction and could relate those linguistic actions to specific outcomes (satisfactory and unsatisfactory interactions).

Research Procedures and Instruments

After participants consented to enroll in the study, they completed a seven-question screening survey so that I could determine that they were not minors and so that I could determine their years' experience and technical proficiency

with the technical systems. When user participants visited the support office, technical-support providers asked them to confirm their participation in the study and checked a confidential list of all participants in the study. Once confirmed, the technical-support providers began recording the interaction. I initially used Silverback 2.0, a type of usability testing software, to record the software because it easily recorded audio in the room, computer screen activity, and facial expressions. The department's information technology (IT) team later requested that I use a university-supported tool instead, Panopto. The university's IRB approved the new technology after I had collected data for four interactions, and I then used the new technology for recording interactions with the remainder of the pilot study and almost all of the main study. Like Silverback, Panopto recorded audio, the screen, and facial expressions. During the main study, one technical-support provider would occasionally use Screencast-O-Matic if Panopto would not work. Two interactions were captured using this software; the other eighteen were recorded with Panopto.

The audio-recording feature captured the conversation for discourse analysis, while the screen-recording feature and facial recordings captured what Brown and Yule (1983) call "paralinguistic cues" (4), which can be used by speakers to "reinforce the meaning" of their spoken discourse. Paralinguistic cues, such as leaning forward, laughing, smiling, breathing, screen activity, and others provide cues for interpreting speakers' meaning. Though users did not have to use the support computers when visiting the office, all users in the pilot and main studies opted to use the support computers. As a result, all interactions captured audio, screen, and facial expressions. However, in many cases for both the pilot and main study, likely because of spatial and technological constraints,

the technical-support providers' faces, as opposed to the users', were primarily visible in these recordings.

After beginning the recording, the technical-support interactions continued as it usually would without a study going on. After the session was over, the users and the technical-support providers each completed a post-session survey designed for each of their roles. Paper surveys kept participation confined to the help interaction, so that both users and technical-support providers did not have to remember to take an electronic survey sent to them via email. After completing the survey, each person folded the paper survey and placed it in a locked submission box kept in the technical-support office. Only I, as the investigator, had the key to open the box. After the session was finished, the technical-support providers exported the video recording and uploaded it to a private cloud-based file management system so that I could retrieve the video for analysis. This uploading procedure was only necessary with Silverback 2.0 and Screencast-O-Matic. Panopto, the new required system, automatically uploaded the videos to a private server to which only I had access. Later, I told the technical-support providers to delete any videos still on the support computers in order to maintain data confidentiality and security.

After I collected pilot study data, I interviewed technical-support provider participants about their experiences with supporting users and the kinds of discourse strategies they tried to use for different purposes (appendix B). I interviewed technical-support providers after the main study as well—two technical-support providers in June 2016 and three technical-support providers in January 2017. For the main study, however, I selected excerpts from two interactions for each technical-support provider, except for one provider who

only had one interaction in the entire study. After coding my data, I found moments in my coding where I needed help interpreting the technical-support providers' intentions. I played back the video and audio recordings of these moments for the providers and asked them what they were trying to accomplish with their language. Their answers helped me to interpret more difficult discourse data, and combined with answers about less difficult discourse data, these interviews lent credibility to my interpretations of the discourse.

In summer and fall 2016, two undergraduate students and I transcribed the interactions from the pilot and main studies using Microsoft Word with numbered lines and double-spaced, 12-point size text. We accounted for the following discourse features in our orthographic transcription:

- Overlaps (e.g.,"That's the same thing [happening with others."
  ["Oh., O.K.")
- Pauses in seconds rounded up to the nearest whole number. (e.g., [*4 seconds*])
- Paralinguistic cues (e.g., [*laughing*], [*breathing*], [*singing*], [*hovering cursor over "edit settings"*])
- Questioning intonation. (e.g., "That works in student view?")
- Reading aloud in quotation marks (e.g., "Continue")
- Confidential information with ID numbers (e.g., "My name is [*U5*].")
- Punctuation rules based on Chicago Manual of Style, 16th edition
- Capitalization rules based on Chicago Manual of Style, 16th edition
- O.K. as the consistent spelling (rather than okay, ok, or OK)
- Uh-huh, Mm-hmm as lexical agreement indicators

- Mm-mm or uh-uh as lexical disagreement indicators
- Eye dialect spellings (e.g., “kinda,” “gonna,” “cuz”)

For each interaction, I checked three times the transcription that my two undergraduate students completed before I began data coding, lending validity to the data because two people were involved with each interaction’s transcription. Additionally, transcribers checked one another’s transcriptions for 5 of the interactions as a way to train them in the process before they continued with the remaining 15 interactions on their own.

After discussing the pilot study with my participants and seeking their feedback on the procedures, we agreed that these procedures did not need to change for the main study because no logistical problems arose during the pilot study. They agreed that the procedures were understandable, agreeable, and nonintrusive. I thus used the same data-collection and -security procedures for the main study. I collected data for the main study from April to October 2016.

Participants and Screening Survey

This section details background information about the participants in the main study, specifically those who not only enrolled in the study but also engaged in a technical-support interaction during it. The main study had more female users than male users (10 and 1, respectively). Because I encouraged my enrolled participants to visit the support team only if they had genuine technical problems, I could not control the gender equity in my participants. Of 41 participants who enrolled in my study as users, only 9 users attended sessions, and 2 of them attended more than 1 time. The technical-support provider participants had an equal number of male participants than female (three and

three), however. Table 3.1 gives the identifying code and biological sex for each participant.

Table 3.1 Participant Profiles

Participant ID	Gender
U2	Female
U5	Female
U11	Female
U14	Female
U19	Male
U20	Female
U23	Female
U32	Female
U35	Female
U40	Female
U41	Female
TS2	Male
TS3	Male
TS4	Female
TS6	Female
TS7	Female
TS8	Male

As I mentioned previously, to learn more about my participants I employed a screening survey when participants enrolled in my study. I used this seven-question survey to confirm that my participants were not minors and to determine participants' levels of technical proficiency and years' experience with each technical system in the research site. Questions about technical proficiency employed a four-point (thus forced-choice) Likert scale asking participants to report their agreement or disagreement with statements about their proficiency with each technology, for example, "I am proficient with the Moodle Learning Management System" (see appendix C). After using this screening survey in the pilot study and because my study did not focus on the relationship between discourse strategies and technical experience or proficiency, I determined it

provided enough data for the kinds of questions I needed to ask for the main study. Table 3.2 gives the distribution for the main study participants' individual years' experience with each technology. Table 3.3 describes these data by category of years' experience.

Table 3.2 Participants' Technical Experiences (By Individual)

	Moodle	Portfolio	DRAW
U2	3	1	1
U5	5	1	2
U11	5	2	1
U14	1	1	1
U19	1	1	1
U20	5	1	5
U23	5	1	1
U32	1	1	1
U35	1	1	1
U40	5	4	1
U41	5	1	3
TS2	2	2	2
TS3	1	1	1
TS4	2	1	2
TS6	1	2	2
TS7	2	2	1
TS8	3	2	3

Note: (1) Less than 1 year experience (2) 1–2 years' experience (3) 2–3 years' experience (4) 3–4 years' experience (5) 5 or more years' experience

Table 3.3 Participants' Technical Experiences (By Category)

	Moodle		Portfolio		Depository	
	TS	U	TS	U	TS	U
Less than 1 year experience	2	4	2	9	2	8
1–2 years' experience	3	0	4	1	3	1
2–3 years' experience	1	1	0	0	1	1
3–4 years' experience	0	0	0	1	0	0
4–5 years' experience	0	0	0	0	0	0
More than 5 years' experience	0	6	0	0	0	1

Figure 3.1 illustrates these tabular data, revealing that generally the users had more experience with Moodle than the technical-support providers did but

that technical-support providers had more experience with the portfolio and profile system and the teaching depository. For Moodle, 6 users had more than 5 years' experience, and 1 user had 2 to 3 years' experience. None of the technical-support providers had this level of experience. The most experienced technical-support provider had 2 to 3 years' experience. Of the 11 participating users, 9 had less than 1 year of experience with the ePortfolio and eProfile system. By contrast, 4 of 6 technical-support providers had 1 to 2 years' experience with the ePortfolio and eProfile system. The technical-support providers also had more experience with the teaching depository system. But generally, the technical-support providers and users had little experience with this system. All had less than 2 to 3 years' experience. Only one participant, a user, had over 5 years' experience with it.

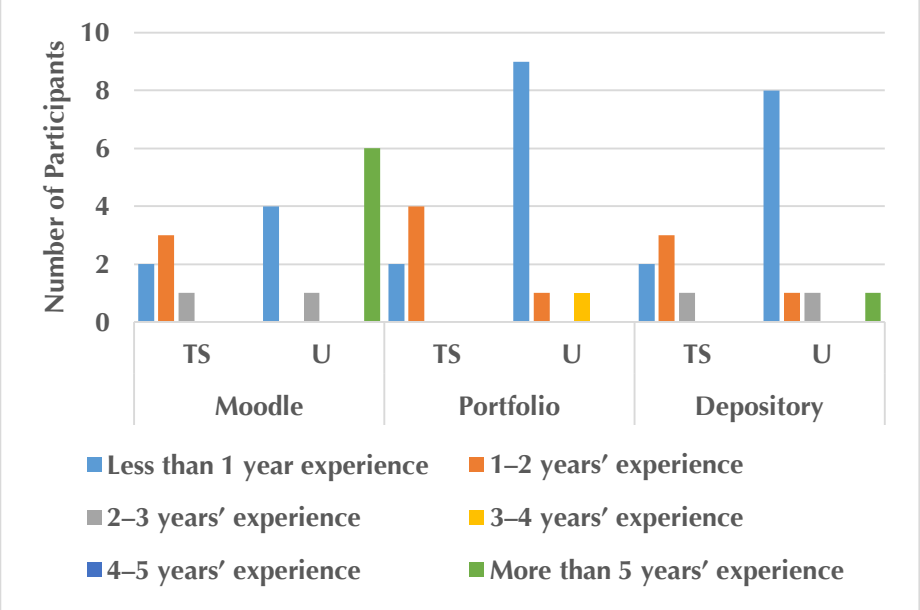

Figure 3.1 Users have more years' experience with Moodle and fewer years' experience with the other two systems.

These data make sense because the Moodle system had been around longer than the other two systems (since approximately 2005). Users affiliated

with the department longer than the technical-support providers would have more experience with it, especially because the users may have been faculty or lecturers while the technical-support providers were graduate students, with a shorter tenure at the university. The ePortfolio and eProfile system and depository system had many more users with less than 1 year of experience, and generally, technical-support providers had more years' experience with these technologies. Table 3.4 gives the descriptive statistics for these main study participants' perception of proficiency with each technology. The lower the number, the more the participants perceived they were proficient with the technology.

Table 3.4 Participants' Perceived Proficiency

	Moodle			Portfolio			Depository		
	Mean	Median	Mode	Mean	Median	Mode	Mean	Median	Mode
Users (n = 11)	2.1	2.0	2.0	4.0	4.0	4.0	3.6	4.0	4.0
Technical-Support Providers (n = 6)	1.5	1.5	2.0	1.5	1.0	1.0	2.2	2.0	1.0

Note: 1 = Strongly Agree; 2 = Agree; 3 = Disagree; 4 = Strongly Disagree; the higher the number, the more the participant feels he or she lacks proficiency

Figure 3.2 illustrates that technical-support providers felt more proficient than users for each of the technologies. For the Moodle system, both sets of participants felt more closely proficient (M = 2.1; M = 1.5, users and technical-support providers respectively). The difference was larger for the other two systems, with technical-support providers feeling more proficient in both technologies.

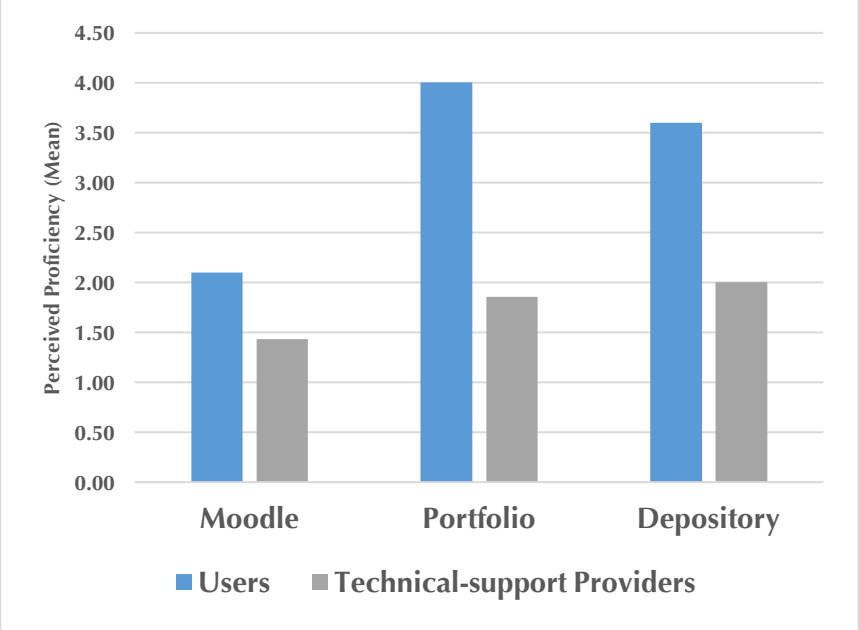

Figure 3.2 Technical-support providers feel more proficient than users.

As expected, those who support a system feel more proficient than the system users because they should and often do have technical proficiency with the systems they support (Callaghan and Thompson 2002). Likely both users and technical-support providers felt less proficient with the other two systems because the department has not used the two systems as long as they had used the Moodle system. Because users had used Moodle longer than technical-support providers, they likely felt almost as proficient with it as technical-support providers.

Post-session Satisfaction Survey

To determine a satisfactory or unsatisfactory outcome from an interaction, I used post-session surveys. I designed one post-session survey for users and one for the technical-support providers. These two surveys allowed me to determine if both members agreed the outcome from the interaction was satisfactory or not. The post-session survey designed for the pilot study is found in Appendix C. The

post-session survey for the pilot study measured key constructs confirmed by the technical-support research. It also used a four-point, forced-choice Likert scale. According to Bearden, Malhotra, and Uscategui (1998), three employee characteristics moderate the relationship between interpersonal contact and customer satisfaction—self-efficacy (the employee's belief that he or she can execute a work activity), empathy (the employee's ability to provide caring, individualized attention to a customer), and adaptability (the ability to respond to the customer throughout the interpersonal interaction) (803–4). I measured each construct from both the technical-support provider's perspective and from the user's perspective by creating items related to each. Table 3.5 gives the relationship between the constructs identified by Bearden, Malhotra, and Uscategui (1998) and the items in the six-item survey.

Table 3.5 Constructs Measured in Pilot Post-Session Survey

Construct	Definition	TS's Statement	U's Statement
Efficacy	ability to execute a work activity	I successfully addressed the problem the user had for this session.	My problem was successfully addressed during this session.
		The user felt the problem was addressed.	The technical support person felt the problem was addressed.
Empathy	ability to provide caring, individualized attention to a customer	I listened carefully to the user during this session.	The technical support person appeared to listen to me during this session.
		I was polite during this session.	The technical support person was polite during this session.
Adaptability	ability to respond to the customer throughout the interpersonal interaction	I used language the user could understand when discussing the problem.	I understood what the technical support person was telling me.
		The user appeared to listen during this session.	I listened carefully to the technical support person during this session.

I used this six-item survey on a four-point Likert scale as the basis for determining which sessions were satisfactory and which sessions were unsatisfactory in my pilot study. During the pilot study, I modified the original

survey based on further research, and I used a new survey in the main study. I chose to develop a new survey for the main study because the pilot study survey was self-developed and did not go through the necessary validity and reliability tests. Furthermore, developing a user satisfaction survey was not the focus of my study. By slightly adapting a pre-existing (and tested) user satisfaction survey to serve as post-session satisfaction survey, I could more reliably and validly claim that some interactions were satisfactory while others were not. To modify this survey, I conducted further research about the constructs identified by Bearden, Malhotra, and Uscategui (1998), which served as the basis for the items in the six-item survey in the pilot study. Burgers, de Ruyter, Keen, and Streukens (2004) developed and tested a scale that determined customer's expectations with support conversations, which they based on the previous customer satisfaction by Bearden et al. (1998). They modified the survey for a technical helpline context because the Bearden et al. study explored customer service interactions more broadly.

Table 3.6 describes how each of the thirteen items was based on the four constructs these researchers identified and tested; it also demonstrates how the two sets of participant statements mirrored one another in order to help identify agreement between technical-support providers and users about satisfactory interactions.

Table 3.6 Constructs Measured in Post-session Satisfaction Survey

Construct	Definition	TS's Statement	U's Statement
Adaptiveness	ability to help and interpret different user problems and questions	I answered different question(s) or complaint(s) the user had with little difficulty.	The technical support member answered different question(s) or complaint(s) I had with little difficulty.
		I adapted to every situation that occurred during the session.	The technical support person adapted to every situation that occurred during the session.
		I took the user's knowledge into account when helping solve the problem(s)	The technical support person took my knowledge into account when when helping solve the problem(s).
		I remained calm and friendly no matter what feelings I was interpreting from the user.	The technical support person remained calm and friendly no matter how I was feeling.
		I helped define specifically the problem(s) and source of the problem(s).	The technical support person helped define specifically the problem(s) and source of the problem(s).
		I was able to help with each and every problem in a timely way.	The technical support person was able to help with each and every problem in a timely way.
Assurance	ability to clearly and thoroughly explain the steps and procedures of solving the customer's question, which reduces customer uncertainty	I clearly and thoroughly explained each and every step I took when solving the problem(s).	The technical support person clearly and thoroughly explained each and every step he or she took when solving the problem(s).
		I clearly and thoroughly explained solutions or recommendations.	The technical support person clearly and thoroughly explained solutions or recommendations.
Empathy	ability to give the customer the feeling that the customer and his or her problem is important	I was able to imagine what the user was going through with his or her problem(s).	The technical support person was able to imagine what I was going through with my problem(s).
		I treated the user uniquely from other users.	The technical support person treated me uniquely from other users.
		I treated the user's problem(s) as important.	The technical support person treated my problem(s) as important.
Authority	the technical or political ability to execute necessary steps to help the customer	I had the necessary authority to solve the user's problem.	The technical support person had the necessary authority to solve my problem.
		I will have to follow-up with the user to help his or her with the problem(s) because I need to ask somebody for permission or help.	The technical support person will have to follow-up with me to help me with my problem(s) because he or she needs to ask somebody for permission or help.

Two other researchers overviewed the survey, providing feedback on it for clarity and ease of use. This process checked face validity, which is the informed examination of an instrument to determine if the instrument appears to measure what the researcher intends (Wrench, Thomas-Maddox, Richmond, and McCrosky 2013, 234). Furthermore, I changed the survey from a four-point (forced choice) Likert scale to a six-point (forced choice) Likert scale. According to Lozano, García-Cueto, and Muñiz (2008), survey instrument reliability decreases the fewer choices are possible, and they suggest a range of four to seven alternatives (78). By moving the choices from four to six, I increased the reliability of the survey. After making these modifications, I used this survey to measure a satisfactory or unsatisfactory outcome in the main study. This survey is found in Appendix E.

To determine whether or not an interaction was successful, I recoded the six-point scale so that "Strongly Agree" and "Agree" counted as 2, "Agree Somewhat" and "Disagree Somewhat" counted as 1, and "Disagree" and "Strongly Disagree" counted as 0. Question 13 was reverse coded because a generally positive response ("Strongly Agree" or "Agree") meant that the technical-support provider lacked authority or ability to resolve the problem, which was not a satisfactory outcome for technical-support interactions. Therefore, for this question, "Strongly Agree" and "Agree" counted as 0, "Agree Somewhat" and "Disagree Somewhat" counted as 1, and "Disagree" and "Strongly Disagree" counted as 2. I then summed the scores from each item on the scale for each participant in an interaction to have one total for the user and one total for the technical-support provider. With the recoding, the highest possible score for one person was 26. The higher the score, the more satisfied the

user or technical-support provider felt about the interaction. I then calculated the difference between the user's score and the technical-support provider's score to determine if the user and technical-support provider disagreed and to what extent. I then summed the technical-support provider's score with that of the user's score. With this summation, the highest possible score was 52. Total scores ranging 0–9 were very unsatisfactory, scores ranging 10–18 are unsatisfactory, scores ranging 19–27 were somewhat unsatisfactory, scores ranging 28–36 were somewhat satisfactory, scores ranging 37–45 were satisfactory, and scores ranging 46–52 were very satisfactory. These categories corresponded to common six-point Likert scales meant to measure satisfaction. Also, the ranges of 9 (the roundest divisor of 52) allowed for a greater possibility of seeing variation in the satisfaction levels. The more obvious and easier divisor (13) limits the number of categories and does not allow for variation in the satisfaction levels (only four items on the scale), especially when participants politely avoided more negative responses in their survey answers. Furthermore, I maintained consistency with my six, rather than four, alternatives in order to maintain a more reliable instrument.

Table 3.7 gives the satisfaction results from the main study interaction. These results give a baseline measurement describing the level of satisfaction for each interaction. While no interactions were "Unsatisfactory," "Very Unsatisfactory," or "Somewhat Unsatisfactory," the slight variation in satisfaction could provide some early insights into the kinds of discursive or contextual qualities of any interactions that were less than "very satisfactory." Of the 20 interactions, only 11 were "very satisfactory," while 8 were "satisfactory" and 1 was "somewhat satisfactory."

Table 3.7 Main Study Interaction Satisfaction

Interaction	Participants	TS	U	Difference	Total	Satisfaction
1	TS2 and U11	23	24	1	47	Very Satisfactory
2	TS2 and U11	17	24	7	41	Satisfactory
3	TS4 and U2	24	24	0	48	Very Satisfactory
4	TS4 and U20	21	24	3	45	Satisfactory
5	TS2 and U14	15	22	7	37	Satisfactory
6	TS7 and U2	22	21	1	43	Satisfactory
7	TS7 and U19	26	24	2	50	Very Satisfactory
8	TS7 and U2	20	22	2	42	Satisfactory
9	TS8 and U2	24	24	0	48	Very Satisfactory
10	TS2 and U23	12	24	12	36	Somewhat Satisfactory
11	TS6 and U2	22	25	3	47	Very Satisfactory
12	TS2 and U5	12	25	13	37	Satisfactory
13	TS2 and U40	25	26	1	51	Very Satisfactory
14	TS8 and U2	23	25	2	48	Very Satisfactory
15	TS3 and U2	24	24	0	48	Very Satisfactory
16	TS7 and U40	25	26	1	51	Very Satisfactory
17	TS3 and U35	17	21	4	38	Satisfactory
18	TS2 and U32	19	24	5	43	Satisfactory
19	TS7 and U41	26	26	0	52	Very Satisfactory
20	TS2 and U40	24	26	2	50	Very Satisfactory

When examining the differences in perception between technical-support providers and users, I found only two interactions in which users had lower totals than technical-support providers. All other differences may be partially explained by technical-support providers evaluating themselves harshly while users remained polite. In interaction 12, TS2 may not have felt he answered the questions accurately or that he could not do so accurately, while U5 may have felt satisfied nevertheless. In other cases, the use of the survey instrument influenced the totals. For example, in interaction 10, TS2 did not complete one question on the survey, which garnered a value of zero and thus lessened the total. Despite these differences, these measurements provided a way to categorize levels of satisfaction for the interactions and thus validly explore the relationship between different discourse strategies and satisfactory outcomes in technical-support interactions. Table 3.8 gives a comprehensive overview of each interaction, including the topic and time duration.

Table 3.8 Interaction Overview

Interaction	Participants	Time	Agenda	Tech	Topic	Date	Outcome	Success
1	TS2-U11	6:11	Function	Moodle	Getting a course website duplicated	Mar. 2016	Very Satisfactory	Resolved
2	TS2-U11	10:16	Function	Moodle	Getting lessons to show students the correct answer to activity questions	Mar. 2016	Satisfactory	Unresolved
3	TS4-U2	1:56	Operation	Moodle	Missing button	Apr. 2016	Very Satisfactory	Resolved
4	TS4-U20	31:02	Function Operation	Moodle	Getting student unenrolled from course website; getting alternate color choices in editor; font size in editor changes to a large size; creating a label; Student is not able to submit an assignment; moving files on course website	Apr. 2016	Satisfactory	Both
5	TS2-U14	8:55	Operation	Moodle	Students cannot see the grade for an assignment	Mar. 2016	Satisfactory	Unresolved
6	TS7-U2	11:47	Operation	Moodle	Students cannot submit assignments.	Aug. 2016	Satisfactory	Resolved
7	TS7-U19	7:33	Operation	Moodle	Students cannot enroll in the course or into Moodle	Sep. 2016	Very Satisfactory	Resolved
8	TS7-U2	8:58	Function	Moodle	Making the attendance module extra credit in the gradebook	Sep. 2016	Satisfactory	Unresolved
9	TS8-U2	6:27	Operation	Moodle	News announcements are not going to students' email inboxes	Sep. 2016	Very Satisfactory	Unresolved
10	TS2-U23	10:31	Function Operation	Moodle	Making unit images uniform on course website; quizzes are not showing the questions that U23 created; using wikis	Sep. 2016	Somewhat Satisfactory	Unresolved
11	TS6-U2	11:12	Operation	Moodle	Students' see different weight percentages for the same assignment; news announcements are not going to student e-mail inboxes; assignment submission modules reset due date to the current time upon saving the settings	Sep. 2016	Very Satisfactory	Both
12	TS2-U5	1:57	Function	Moodle	Accepting more than one answer on a quiz	Sep. 2016	Satisfactory	Resolved
13	TS2-U40	17:36	Function	Moodle	Explaining rubrics to students; using clickable rubrics; ensuring students can upload documents with images	Sep. 2016	Very Satisfactory	Resolved

Table 3.8 continued

14	TS8-U2	7:39	Function	Moodle	Setting week views to prevent students from being unable to view grades	Oct. 2016	Very Satisfactory	Resolved
15	TS3-U2	1:49	Function	Moodle	Viewing edit screen to leave students feedback files	Sep. 2016	Very Satisfactory	Resolved
16	TS7-U40	33:12	Function	ePortfolio Moodle	Getting correctly sized images to upload to portfolio; using clickable rubrics in Moodle; getting password to work in Adobe product; separating course sections while grading	Oct. 2016	Very Satisfactory	Resolved
17	TS3-U35	12:52	Operation	Moodle	Student course grades don't appear to calculate correctly.	Oct. 2016	Satisfactory	Resolved
18	TS2-U32	5:47	Function	ePortfolio	Making websites private from students; managing websites for different purposes	Oct. 2016	Satisfactory	Resolved
19	TS7-U41	27:42	Function	ePortfolio	Preparing for a ePortfolio workshop; using ePortfolios for portfolio pedagogy; grading ePortfolios	Oct. 2016	Very Satisfactory	Resolved
20	TS2-U40	15:24	Function	Moodle	Setting extra credit for an assignment; Selecting a rubric for an assignment	Oct. 2016	Very Satisfactory	Resolved

The average interaction time was 10:45, and the interactions ranged between 1:45 and 33:12. Of the 20 interactions, 13 interactions had resolved problems, 5 interactions left the problems unresolved, and 2 had both resolved and unresolved problems. Of the 20 interactions, 17 interactions discussed Moodle, 2 interactions discussed the ePortfolio and eProfile system, and 1 interaction discussed both Moodle and the ePortfolio and eProfile system. Also, 12 interactions had functionality as the agenda (how to do something with the technology), 7 interactions had operationality as the agenda (something is not working), and 2 interactions had both types for the agenda. Lastly, U2 attended the most interactions (7), followed by U40 (3) and U11 (2). TS2 administered the most interactions (8), followed by TS7 (5) and TS3, TS4, and TS8 (2 each); TS6 only administered 1 interaction.

Coding Scheme Development

The pilot study served as the basis for starting to develop my coding scheme for this study. I based the first-cycle coding scheme on codes I derived from a combination of studies. In accordance with process coding, I maintained a gerund for each code (Saldaña 2013, 96). According to Saldaña (2013), process coding "uses gerunds ('-ing' words) exclusively to connote action in the data…and more general conceptual action" (96). Speakers conduct these actions, or speech acts, with the purpose of reaching goals or handling problems. Coherent with speech act discourse analysis's foundational concern for what speakers and hearers are doing with language in a specific context (Brown and Yule 1983; Searle 1976), these process codes, using gerunds, nominalize the

actions employed by the speakers' utterances and best exemplify what speakers are doing with their discourse.

The following sections describe the iterations of these codes for both macrolevel and microlevel discourse analysis. For both macrolevel and microlevel coding scheme development, I had three independent coders help me develop my codes to increase their reliability. The coders coded both macrolevel and microlevel codes in the same rounds, but I report the coding scheme development in the following sections by first discussing the macrolevel coding scheme development and then the microlevel coding scheme development.

Macrolevel Coding Scheme Development

I used codes from Agar (1985), Xu, Wang, Forey, and Li (2010), and Clark, Murfett, Rogers, and Ang (2012) to develop the macrolevel first-cycle coding scheme in table 3.9 (see table 2.1 for a comparison of their macrolevel codes).

These macrolevel stages may fluctuate or repeat based on the speakers' and situation's needs, which Xu et al. (2010) calls the "genre structure potential" (459–460). The interaction can maintain its full structure in linear sequence or have certain components out of order or repeated. As Xu, Wang, Forey, and Li (2010) stated it, the number and ordering of the stages can fluctuate according to the purposes of the customer, the content of the session, and the worker's knowledge and skill (461). As an example of fluctuation, if the issue is difficult for the worker to resolve, the defining stage, with its question and answers, may need to be repeated several times (461). Also, the sequencings of defining and resolving may repeat if the customers decide they have another problem to solve.

Pilot study coding. To test this macrolevel coding scheme, I coded two interactions in order to better observe variation and more fully test the codes (i.e., increase the chances that more instances of discourse codes will repeat).

To develop the coding materials for an independent coder, I unitized the transcript from each interaction to mark the beginning and ends points of where I saw changes in the macrolevel stages. I provided the coder the transcript with open and closing horizontal lines before and after certain points in the transcripts. I determined these units by identifying when the discourse changed topics. As Mackiewicz and Thompson (2015) state, these topics are identified by "monologic or dialogic strings of conversation that coherently address one subject" (16). I found singular or strings of turns that cohered around one subject and also looked for discourse markers that signaled a change ("Now" and "I also had another question."). I fractured three interactions (transcripts) in this way. The coders also received a copy of the codes, which served as the codebook they used during the coding session.

I trained one coder by reviewing the macrolevel codes and by going through one interaction (transcript) with him. This process served to norm the coder so that he understood what the codes meant and how they looked in the discourse. Furthermore, this process helped the coder learn how the coding materials worked.

After training the independent coder, I gave the coder the data from the two remaining interactions. The coder then coded these two interactions for both macrolevel and microlevel discourse using the codebook as a guide. The coder took one week to code the two interactions. Using Cohen's Kappa to determine

agreement between the coder and I, I found that our agreement was .90, which is considered satisfactory (Wrench, Thomas-Maddox, Richmond, and McCroskey 2013, 284). Because I had reached a sufficiently reliable level of agreement, this coding session ended the pilot study phase of my macrolevel code development.

Main study coding. Despite a reliable level of agreement in the pilot study, I responded to the disagreements in the coding between the independent coder and I in the pilot study by revising the codes for the macrolevel discourse by adding more clarifying and specifying details to the code descriptions.

I followed the same procedure as I did in the pilot study coding for the main study coding. I fractured three different interactions from my main study data. I used the first interaction to norm with a new coder because the first coder could not continue with the study. The next day, the coder took a few hours to code two other interactions. Using Cohen's Kappa, I found our agreement was lower than experienced in the pilot study (.69). Through discussion with this second coder, we determined we should update the codebook to include an additional stage, different than that in the current research literature and which I had not determined necessary with my first coder in the pilot study. Specifically, a new code helped to identify stages in which the problem was not resolved. After a third round of coding following the same procedures as before (two days of coding—one for norming on one transcript and one for independent coding of two transcripts), the macrolevel reached a Cohen's Kappa of .80.

Thus, the reliable macrolevel coding scheme for this study is the following (table 3.9):

Table 3.9 Macrolevel Codes

Code ID	Definition
Identifying	Identifying U as part of the technical system such as Moodle, including obtaining U's name and any other pertinent identifying information about U, such as course section.
Defining	Outlining, summarizing, and/or indicating that there is a problem or question. Often prompted by U but could also be prompted by TS.
Attempting	Working through possible solutions to the problem or possible answers to the question. The problem does not get resolved fully or the question answered fully in that session. TS or U may not be satisfied with a proposed resolution or answer. Or U and/or TS move on to a new problem without a resolution or answer.
Resolving	Providing information, instruction, and/or solutions for a problem and confirming a specific problem is resolved. TS and U are satisfied with resolutions or answers. The problem has to be resolved or the question answered in that session. Making plans to solve the problem at another time (e.g., following-up through e-mail or another meeting, or trying something later at home) does not mean the problem or question was resolved or answered.
Closing	Confirming that U is satisfied, that U has no more problems to talk about, and saying good bye and/or setting-up a follow-up meeting or email conversation; includes taking the post-session survey if recorded

Microlevel Coding Scheme Development

In addition to macrolevel discourse analysis, I also analyzed the discourse for two kinds of microlevel discourse: problem diagnosing strategies and problem resolving strategies, which researchers have acknowledged exist in these interactions but which have not been explored in any detail. To develop the problem diagnosis codes, I followed Thompson and Mackiewicz's (2014b) coding scheme, which modified a framework from applied psychology (Graesser, Person, and Huber 1992). Thompson and Mackiewicz explored the questioning types that experienced writing center tutors use and also the questioning types that students use. One fundamental difference between writing tutoring and technical support, however, is that writing tutors often use "cognitive

scaffolding," in which "tutors give students opportunities to figure out what to do on their own" (Mackiewicz and Thompson 2014c, 56). In other words, writing tutors employ questioning "to help students form explanations for themselves" because writing center theory assumes that students can better apply their learning on their own (later without a tutor) if they arrive at explanations for themselves (Mackiewicz and Thompson 2014c, 61). Technical-support providers may not employ such "leading-scaffolding" or "known-information" questioning types designed to get the user thinking (Mackiewicz and Thompson 2014b, 43; Mackiewicz and Thompson 2014c, 61). The user just wants to get the problem solved (Callaghan and Thompson 2002, 245). As such, I modified Mackiewicz and Thompson's (2014b) questioning types for the purposes of technical-support interactions by omitting such questions and also by tracing their question types to the study on which they built them. By tracing their coding scheme's origin, I found them in their earliest forms, forms that address one-to-one tutoring interactions in general and not writing center tutoring specifically (Graesser, Person, and Huber 1992). This process allowed me to better assess how Mackiewicz and Thompson modified the original scheme for the writing tutoring context and allowed me a larger range of options for my own coding scheme. These questioning types provide a helpful framework for analyzing the problem diagnosis types in technical-support interactions.

Also, questioning may have interrogative syntax, but it may also have declarative syntax with the illocution (the intended meaning) of questioning. Thompson and Mackiewicz (2014b) confirmed that questioning that uses declarative syntax can be called "inquiring": "although the illocutionary act of questioning is typically manifested in interrogative syntax, questions can also

manifest themselves in noninterrogative syntax" (41). As such, I included in my coding scheme "inquiring" as my way of describing questioning discourse.

I also developed codes for the problem resolving strategies that the technical-support providers used to help users resolve their technical problems. Mackiewicz and Thompson (2014b) described "instructional strategies" that writing tutors employ when directing students. They identified three directive types: telling, suggesting, and explaining and exemplifying. The telling strategy uses unmitigated discourse and serves as a bald-on-record statement about what to do (Mackiewicz and Thompson 2014b, 60; Brown and Levinson 1978). Specifically, the inferential path, or the length between what is meant and what is said, in a telling strategy is a short path. When speakers use this strategy, they use little-to-no measure to save the hearer's negative face; that is, they overtly impose on their listeners' wills, or autonomy ("Upload your profile picture using drag and drop"). However, the speaker may use a politeness strategy to downgrade the force of the directive. Specifically, speakers use downgraders such as subjectivers ("I think…"), hedges ("maybe"), downtoners ("possibly"), appealers ("…, OK?"), cajolers ("You know,…") and understaters ("You should just…"), ensuring that the listeners' autonomy or competence feels less threatened by the advice (Mackiewicz and Thompson 2014b, 60; Mackiewicz and Riley 2003, 85–6). These mitigating devices do not shorten the inferential path, but they do mitigate the force of the directive.

The suggesting strategy, by contrast, specifically lengthens the inferential path by giving the directive indirectly. However, indirect strategies (suggestions) can be conventional or nonconventional. Suggesting may be "conventionally indirect" by making the listener's obligation less clear by employing low-value

modal verbs such as "can" or "could," giving the listener a sense that they may (but need not) heed the speaker's recommendation and should use his or her own discretion (Mackiewicz and Riley 2003, 89–90). Suggesting could be "nonconventionally indirect," which Mackiewicz and Riley (2003) equate with hints (90). They argue that strong hints state an observation without specifically recommending any change to it, while weak hints will state general rules, guideline, principles, or practices without indicating the listener needs to heed those rules, guidelines, principles, or practices (91).

The last directing strategy Mackiewicz and Thompson identified (2014b) is the explaining and exemplifying strategy, in which speakers "offer reasons for and illustrate their advice" (61). Mackiewicz and Riley (2003) describe another version of this advice-giving strategy as using "payoff statements," which allow the speaker to justify the advice by giving an "explicit benefit of complying" to the advice (88). Together the coding schemes identified and employed in tutoring and editing research provided the groundwork for the first cycle coding scheme.

Pilot Study Coding. Just as I explained for the macrolevel coding scheme development rounds, I unitized three interactions. I determined my unit of analysis for this process as the illocutionary unit. The illocutionary unit was my unit of analysis because it spanned sentence boundaries and was manageable for determining beginnings and endings. According to Rourke, Anderson, Garrison, and Archer (2000), an illocutionary unit "should be viewed with the following question in mind: What is the purpose of a particular utterance? A change in purpose sets the parameters for the unit" (10–11). I highlighted each unit in alternating colors, and indeed, these units did span sentence and even speaker

boundaries. I specifically omitted backchannels ("Uh-huh," "Mm-hmm," "Yeah," "O.K.") and echo phrases (instances in which the speaker echoed what the other speaker just said in reciprocation). I carefully included what may seem to be a backchannel if it was an obvious answer to a question. For example, "Mm-hmm" or "Yeah" could be the answer to "Does Moodle allow you to embed video?" I also ignored joking, overt references to the fact that participants were being recorded or in a study ("Is it recording?"). I also omitted confidential information, such as talk from those not involved in the study or the names of speakers or members of the organization, and I ignored off-topic talk. For example, a user and a technical-support provider spoke about when a paper was due for a course they were both taking.

When receiving the highlighted transcript, the coder needed only to identify which kind of strategy the interlocutors used based on the codebook I provided them. I used the first interaction to norm with the coder and to explain the codes to my coder. The coder then took one week to code the two remaining interactions. After conducting a Cohen's Kappa calculation, I found that my coder and I had unsatisfactory agreement ($k = .51$). I had expected this result for this first-cycle coding scheme because I based the codes on a similar though different discursive context.

I rethought my microlevel coding scheme. As a result of these complications and the difficulty of knowing for sure if a speaker meant to inquire or not, I determined to include questioning but to create additional codes reflecting other speech acts. At this point, my discourse analysis moved from a deductive approach, based on previously identified discourse codes, to a more inductive approach that responded to the data to form new codes.

I made into separate codes the following instances:

- Technical-support providers or users made an observation.
- Technical-support providers or users made a speculation about what may be the source of the problem
- Technical-support providers or users indicated what they were doing at certain moments.
- Technical-support providers or users gave background information.
- Technical-support providers or users gave an opinion about the technology (positive or negative).
- Technical-support providers or users stated what they needed.

I also clarified what "inquiring to gain knowledge" meant by specifying what kind of knowledge a speaker sought to understand: to understand the technology, to understand needs, to understand background.

For the second-cycle problem resolving strategies, I responded to my observations of users actually directing technical-support providers to do things, typically during the defining stage. For example, a user would tell a technical-support provider to "try" different actions or "go to" different places in order to show the technical-support provider more about the user's problem. Also, both technical-support providers and users would confirm or deny statements from the other person in order to direct one another toward diagnosing or resolving a problem. Lastly, I decided to completely drop indirect directives (both conventional and nonconventional) from my codebook. I determined that describing degrees of politeness was beyond the scope of the study because my

focus was on functional rather than social categories of discourse. A future study could build off this study to determine levels of politeness and explore more the social element of the interactions.

I also split "explaining and exemplifying" by clarifying and specifying the code and the description based on what I was seeing in the data. I created one code for showing, in which a technical-support providers showed the user how to do something or how something works by using the technology to demonstrate it, which prompted me to rely heavily on paralinguistic cues accounting for screen action. This reliance tightened the focus of transcription to carefully locate screen actions in the video where they were available. In the cases in which the technical-support providers only explained the process or procedure, but did not use the technology, I created another code. Lastly, I also isolated for its own code instances in which technical-support providers or users actually "tell," or give each other a directive to do something in the interaction. After unitizing three other interactions into units for my code, I followed the same procedures in which one coder and I normed on one interaction and coded two more interactions independently. After calculating Cohen's Kappa for these sessions, my coder and I came to unsatisfactory agreement for two additional rounds. Again, a round involved norming on one interaction and coding two interactions independently. On the fourth round, we finally reached satisfactory agreement (k = .75). The pilot study was then over for developing a reliable coding scheme.

Main Study Coding. I then took 3 interactions from my main study data, which accounted for 150 coding decisions. With a total of 1921 units in all 20

interactions, 150 was 7% of my data set. Throughout the pilot study, I determined that my coders could maintain accuracy and energy at around 150–160 coding decisions (about two hours of coding); careful to avoid coder fatigue I decided to avoid the additional 40–50 coding decisions for a given coding session, which would have made for a true 10% of the data. However, the norming session did add 60 additional decisions for discussion. This total of 210 coding decisions is more than 10% of the data. Furthermore, 2 transcripts were 10% of the 20 transcripts available in the study. After unitizing the interactions, my coders and I normed on one interaction, and we independently coded the remaining two interactions. After 8 rounds of unsatisfactory agreement (4 with one coder and 4 with another), I determined that the coders were fatigued by the number of codes about which they needed to make decisions. Our Cohen's Kappa numbers ranged from .64 to .68. I thus collapsed many codes (from 21 codes to 16 codes). Specifically, "inquiring to understand needs" and "inquiring to understand background information" were collapsed into one code. Also, expectations for the technology or expectations for the session were collapsed into one code "stating needs." I also completely removed "inquiring to confirm" as a code because my coders could not determine consistently when the topic had been brought up before in the conversation and thus when the speaker was confirming something what was said. After two more rounds with this simpler coding scheme, my second coder and I finally reached satisfactory agreement (k = .78) for the coding scheme. This reliable coding scheme is presented in table 3.10 and represents a useful coding scheme for identifying the microlevel discourse technical-support providers and users employ in these interactions.

Table 3.10 Microlevel Coding Scheme

Code ID	Description for TS and U
inquiring to understand needs or background information	inquiring to understand or confirm listener's needs or background information
inquiring to learn about the technology	inquiring to learn about the technology, its settings or features, and/or how to use them
inquiring to check comprehension	inquiring to check if listener comprehends what speaker said, did, or saw/sees
inquiring to gain permission	inquiring to gain permission to do something at that moment during the interaction
stating needs	stating needs for the technology's settings/features or for the session's procedures
giving background information	giving background information about the problem or question brought up in that macrolevel unit or session
confirming or denying	confirming or denying what listener or speaker said, did, or asked with a yes- or no-type answer, an I-don't-know-type answer, or a noncommittal answer
declaring the problem or problems as solved	declaring a problem as solved or a question answered
judging the technology	judging the technology and/or its features
observing	describing what speaker sees, hears, or notices while using or observing the technology at that moment during the session
speculating	speculating about a problem or question brought up in that macrolevel unit or session
signaling	signaling what speaker is doing at that moment or what speaker will do next
planning	planning what to do either within or after the session
showing how the technology works or how to do something with it	showing listener how the technology works or how to do something with it by using the technology itself
explaining how the technology works or how to do something with it	explaining to listener how the technology works or how to do something with it without using the technology itself
telling	telling listener what to do at that moment in the session

Conclusion

In this study, I employed a discourse analysis of 20 technical-support interactions between technical-support providers and users. Using screening surveys, post-session satisfaction surveys, and a reliable macro- and microlevel coding scheme, this study identified discourse strategies at helpdesks in order to more reliably describe the discourse in satisfactory interactions.

In the next chapter, I report and discuss macrolevel discourse using word volubility and frequencies per interaction. Then, I report the frequencies with which microlevel discourse strategies appear in each macrolevel stage. Lastly, I report on the genre structure potential for these interactions and give an overview of the genre structure for each interaction. These results and discussions provide a rich description of the larger discourse moments in these interactions, allowing organizations, managers, and technical-support providers to maintain coherence in these interactions by knowing what moments should characterize their conversations. Furthermore, that discussion builds on current research in customer-support interactions by contextualizing that macrolevel discourse research for technical-support interactions in particular.

CHAPTER 4: MACROLEVEL DISCOURSE IN TECHNICAL-SUPPORT INTERACTIONS

In this chapter, I describe the identifying, diagnosing, attempting, resolving, and closing stages of technical-support interactions. I first describe each stage one-by-one in terms of volubility counts (word counts) and then give the frequencies and types of microlevel discourse that occurs within each macrolevel stage. Lastly, I describe the extent to which the macrolevel discourse follows the established macrolevel structure that other researchers described by presenting for each interaction the order and repetition for each stage and by giving the generic structure potential of technical-support interactions.

The Identifying Stage of Technical-Support Interactions

In the identifying stage, technical-support providers and users discuss where the technical-support providers can identify the users by gaining identifying information that helps the technical-support providers relate the technology to the users. To help describe the extent to which each speaker contributed to this stage, I quantified the word volubility of both the technical-support providers and the users in each of the 20 interactions. Of the 20 interactions, only half of them contained the identifying stage. The reason the other 10 did not have this stage is most likely because the identifying stage occurred before the video and audio recording began. It was unlikely I could capture all introductory comments between the participants before they began the recording, and in the interactions without identifying stages (2, 4, 8, 9 10, 12, 13, 16, 19, and 20), the discourse began with the speakers already where they needed to be to discuss the problem, suggesting that an exchange, likely an identifying exchange, occurred before the recording began. The presence of the identifying stage in the

other 10 interactions, however, still provides a helpful description of the discourse in that stage. Table 4.1 shows the results of the volubility in the 10 interactions in which the identifying stage was captured.

In the identifying stage, volubility ranges from a high of 106 words (interaction 18) to a low of 5 words (interaction 15). In these 10 interactions, the users' volubility exceeds that of the technical-support providers in all but 2 interactions (interaction 5 and interaction 18). However, in every interaction except 18, the difference in volubility between the two speakers is minimal. Ultimately, the speakers contribute relatively equally to the stage. More revealing is that the volubility for this stage is low overall—the 10 interactions only have 270 words devoted to identifying (1.04% of words for all 20 interactions and 3.19% words for the 10 interactions with the identifying stages captured). This low volubility count suggests that identifying contributes minimally to the entire interaction.

Table 4.1 Identifying-Stage Volubility

Interaction	Participants	TS	U	Total
18	TS2-U2	78	28	106
1	TS2-U11	22	25	47
17	TS3-U36	12	16	28
3	TS4-U2	5	14	19
5	TS2-U14	10	8	18
6	TS7-U2	7	10	17
7	TS7-U19	3	10	13
14	TS8-U2	4	5	9
11	TS6-U2	2	6	8
15	TS3-U2	2	3	5
Total (%)		**145 (53.7)**	**125 (46.3)**	**270**

In examining the stages more closely, I found that the stages contained particular microlevel discourse relevant to identifying. Table 4.2 describes the types of macrolevel discourse that appears in the stage and who spoke that discourse.

Table 4.2 Microlevel Discourse in the Identifying Stage

Microlevel Discourse and Speaker Type	TS	U	Total
Inquiring to understand needs or background information	18	2	**20**
Giving background information	0	12	**12**
Confirming or denying	3	9	**12**
Inquiring to gain permission	0	1	**1**
Signaling	5	0	**5**
Telling	0	2	**2**
Speculating	1	0	**1**
Total	**27**	**26**	**53**

The most frequent discourse type in the identifying stage is "inquiring to understand needs or background information," which occurred 20 times (37.7% of the identifying discourse). The second-most frequent discourse types were "giving background information" and "confirming or denying," which both occurred 12 times (22.6%). These types likely occurred in this stage because the identifying stage is the opportunity for the technical-support providers to ask the users where to find the problem ("Um, which, uh section?" or "[*course number*]?") and the users' response may be the information itself ("It's the [course number] class"). Also, the user may provide a confirmation or denial to the technical-support provider's question. ("Yeah, that's the one.") Notably, technical-support providers inquired more frequently than users (18 times), and users gave information more frequently (21 times). Furthermore, the users told the technical-support providers what to do in this stage twice by informing the technical-support providers where to find the users' websites. When the technical-support providers employed "speculating," the technical-support

providers simply wondered aloud if the users' website could not be found because the technical-support providers had input the users' ID incorrectly. Lastly, when the user asked for permission, the user simply wanted to ask if she could sit closer to the screen.

This stage, however low its volubility, is a crucial stage in the technical-support interaction because it enables the interaction to successfully launch into diagnosing and resolving the technical problem. Without the antecedent knowledge of who the user is and what account or website to engage, the participants cannot diagnose or resolve the problems. As a form of institutional discourse, a technical-support interaction must entail aligning the clients' problems to the institutions' frames for understanding and accessing those problems (Agar 1985, 150). My study ratifies Agar's claim in that for technical-support interaction to succeed, they seem to require an exchange to gain institutional identifying information.

The Defining Stage of Technical-Support Interactions

During the defining stage, the technical-support providers' main goal is to ensure that the users' problems are understood, even if the users do not understand the problem fully. As Agar (1985) noted, institutional discourse requires the institutional representative to "[fit] the client's ways of talking about the encounter to ways that fit the institution's" (149). The defining stage allows the two speakers to understand one another in this way. To help describe the extent to which each speaker contributed to this stage, I quantified the volubility of both the technical-support providers and the users in each of the 20 interactions. All 20 interactions contained this stage, and the volubility reveals its

importance to the interactions. Table 4.3 gives the volubility for this stage. The volubility counts range from 22 (interaction 17) to 1340 (interaction 4). Interaction length only partially explains this variation, however. For example, interaction 15, the shortest interaction (1:56) only had 91 words for the defining stage, but interaction 17 was over six times as long as interaction 15 (12:52) and contained fewer words in the defining stage (22). Likely, this stage's volubility varies based on how quickly the participants understand the problem, or even based on how many problems the users want to discuss. Of the total number of words in the data set (26023), defining required 28.04% of the discourse.

Table 4.3 Defining-Stage Volubility

Interaction	Participants	TS	U	Total
4	TS4-U20	286	1064	1350
11	TS7-U2	238	571	809
6	TS7-U2	240	463	703
2	TS2-U11	156	517	673
16	TS7-U40	152	469	621
10	TS2-U23	85	524	609
7	TS7-U19	45	234	279
1	TS2-U11	59	217	276
8	TS7-U2	72	201	273
19	TS7-U41	29	236	265
9	TS8-U2	110	148	258
18	TS2-U32	49	150	199
14	TS8-U2	33	160	193
13	TS2-U40	14	171	185
5	TS2-U14	34	96	130
3	TS4-U2	9	128	137
12	TS2-U5	0	125	125
20	TS2-U40	24	75	99
15	TS3-U2	0	91	91
17	TS3-U35	5	17	22
Total (%)		**1640 (22.5)**	**5657 (77.5)**	**7297**

More revealing than the volubility for each interaction is the more prominent speaker in this stage. In the interactions, the users spoke more frequently in this stage, suggesting the users spent the stage giving information relevant to diagnosing the problem. In two interactions, the technical-support providers said nothing. Both Agar (1995) and Steehouder (2007) affirmed that the clients provide answers or narratives in response to the institutional representatives' questions, suggesting that the clients tend to keep the floor and gives the majority of the substance to the stage. These volubility results confirm that research.

In examining the stage more closely, I found that the defining stage contained certain types of microlevel discourse relevant to defining the problem. Table 4.4 gives the types of microlevel discourse in this stage and who spoke the discourse. The most frequent discourse in the defining stage was "giving background information" (136 times) followed by "inquiring to understand needs or background information" (91 times). It makes sense that this pair of questioning and answering would occur in this stage because the defining stage exists to help the two speakers understand the problems so that they can resolve them. Further confirming the existing research, the technical-support providers inquired about needs or background information (80 times) and the users provided such information (127 times), either in response or on their own. Two of the common ways users responded to technical-support providers, apart from "giving background information," is "inquiring to understand the technology" (40 times) and "stating needs" (49 times). "Confirming and denying" was the third-most frequent microlevel discourse type (84 times), but again, users provided this discourse more frequently (52 times), suggesting that the users

would provide "yes-or-no" type responses to the technical-support providers' questions. Other frequent discourse included "telling" (37 times) and "signaling" (36 times), with users and technical-support providers serving as the most prominent speakers of those discourse types respectively. These results suggest that users tell technical-support providers what to do in this stage in order to help define the problems ("So keep scrolling down"), while technical-support providers signal what they are doing or about to do when defining the problem. ("Let me just log in as a student really quickly- and then we'll see what the student sees.")

Table 4.4 Microlevel Discourse in the Defining Stage

Microlevel Code	TS	U	Total
Giving background information	9	127	**136**
Inquiring to understand needs or background information	80	11	**91**
Confirming or denying	32	52	**84**
Inquiring to understand technology	4	40	**44**
Stating needs	2	39	**41**
Telling	4	32	**37**
Signaling	29	7	**36**
Observing	8	8	**16**
Explaining how the technology works or how to do something	10	4	**14**
Inquiring to check comprehension	2	7	**9**
Inquiring to gain permission	1	6	**7**
Speculating	1	6	**7**
Judging the technology	1	4	**5**
Showing how the technology works or how to do something	3	2	**5**
Planning	2	2	**4**
Total	**188**	**347**	**536**

Ultimately, these results confirm that the defining stage is about the technical-support providers seeking information and the users providing it. However, Agar (1985) assumed that the "territory within which the client can talk are in the hands of the institutional representative" (152). These results counter Agar's

assertion because they reveal that the client controls the conversation, providing the information necessary for the institutional discourse to function appropriately. As a form of institutional discourse, certainly the technical-support providers try to understand the client's frame of reference, but as a service interaction, the interactions must allow the clients to control the flow of the conversation, directing the agenda and pointing the institutional representatives to appropriate information that helps them better solve the clients' problems. In chapter 5, I describe in more detail these microlevel exchanges to showcase what each strategy looks like as users and technical-support providers define the problems together. For now, these results reveal the relative size of the stage, the most frequent speakers, and the microlevel discourse types that appear in it.

The Attempting Stage in Technical-Support Interactions

During the attempting stage, the technical-support providers try to solve the defined problem; however, the problem does not get solved in the interaction, either because the technical-support providers have to follow-up with the users in some way or because the technical-support providers give the users instruction that the users should attempt on their own at a later time, leaving the problem unresolved at that moment. The attempting stage by its nature is not a resolved problem, even if both participants hope that it can or will be resolved later. To help describe the extent to which each speaker contributed to this stage, I quantified the volubility of both the technical-support providers and the users in each of the 20 interactions. Half of the 20 interactions contained this stage. In these interactions, a problem or more than one problem was left unresolved.

Table 4.5 gives the volubility for this stage. The volubility counts range from 28 in interaction 1 to 856 in interaction 4. Of the total number of words in the data set (26023), attempting required 14.17% of the discourse. This relatively low number can be explained in two ways. First, only half of the interactions required attempting. Secondly, the technical-support providers did not take very long to realize that they did not understand how to solve the problem. As TS2 told me in a post-interaction stimulated-recall interview, TS2 did not think it served the users well for users to wait while the technical-support providers try to figure out an answer. Instead, it made more sense to follow-up with the users later when the problem was solved. For this reason, attempting was relatively short because technical-support providers would rather try to solve the problem on their own rather than have the users wait on them.

Table 4.5 Attempting-Stage Volubility

Interaction	Participants	TS	U	Total
4	TS4-U20	698	158	856
9	TS8-U2	406	271	677
5	TS2-U14	392	224	616
10	TS2-U23	468	34	502
8	TS7-U2	232	157	389
2	TS2-U11	189	87	276
7	TS7-U19	95	62	157
11	TS6-U2	91	29	120
16	TS7-U40	66	0	66
1	TS2-U11	0	28	28
	Total (%)	**2637 (71.5)**	**1050 (28.5)**	**3687**

More revealing perhaps than the general volubility of this stage is that the technical-support providers spoke the most in the stage in every interaction but the first interaction. In the first interaction, the stage was very small (28 words), taking 4.74% of the interaction's total discourse. In that interaction, U14 quickly

stated that U14 planned to send TS2 an email with additional information, without which TS2 could not resolve the problem. In all other cases, the technical-support providers were the most frequent speakers, and overall, the technical-support providers spoke the most (71.5%) in the attempting stage. This finding makes sense after considering the microlevel discourse that occurred.

To better understand this nature of the attempting stage, I quantified the types of microlevel discourse that appeared in it. I found that the microlevel discourse corresponded logically with the kinds of discourse that would occur while technical-support providers attempted to solve a problem. Table 4.6 describes the type of microlevel discourse that appears in the attempting stage and also which participants spoke the discourse.

Table 4.6 Microlevel Discourse in the Attempting Stage

Macrolevel Code and Speaker Type	TS	U	Total
Signaling	91	2	**93**
Confirming or denying	25	16	**41**
Giving background information	11	28	**39**
Observing	32	5	**37**
Speculating	21	7	**28**
Planning	14	12	**26**
Explaining how the technology works or how to do something	25	1	**26**
Inquiring to understand needs or background information	16	3	**19**
Inquiring to understand technology	1	11	**12**
Telling	2	5	**7**
Judging the technology	5	1	**6**
Showing how the technology works or how to do something	1	1	**2**
Stating needs	0	1	**1**
Declaring the problem or problems as solved	0	1	**1**
Total	**244**	**94**	**338**

The most frequent discourse type was “signaling” (occurred 93 times or 28.5% of the time). Signaling is a discourse type in which a speaker tells the

listener what he or she is doing or is about to do. The technical-support providers signaled the most (91 times). It makes sense that the technical-support providers would speak aloud what they were doing at certain points while attempting to solve the problems. All the technical-support providers reported in stimulated-recall interviews that they would speak aloud what they were doing as they tried to solve problems. They gave three main reasons for this: (1) to break the awkward silence; (2) to help themselves think while they tried to solve the problem; and (3) to potentially teach the users how to problem solve if the users happened to be listening. This finding suggests a unique discourse component to one-to-one, interpersonal technical communication, something not noted elsewhere in the research on these interactions. Technical-support providers coordinated social dynamics (breaking the awkward silence), worked as problem-solvers, and instructed users how to find answers to their own problems. This high-level of discourse confirms that technical-support providers engage in multiple frames of discourse (social and instructional) as they do their work (Barley 1996). Technical-support providers are skilled discourse workers.

The second-most frequent discourse type was "confirming or denying" (41 times), which was spoken almost equally by users and technical-support providers (25 and 16 times respectively). The most frequent "confirming or denying" statement for technical-support providers in this stage was a variation on "I don't know" (10 times). Other statements confirmed that the technical-support providers would follow-up with the users about the problems. ("Yeah, sure, sure, sure. I will do that.") Or confirmations were yes-or-no type questions to users asking about what may be the solution. ("No, that couldn't be it actually.") Users responded yes-or-no type answers to technical-support

providers when they inquired about background information that may have helped the technical-support providers to solve the problem. When users did not give a closed "yes-or-no" type answer, they would employ "giving background information" discourse (39 times). The users spoke this discourse the most often to tell the technical-support providers information that may help them solve the problem. ("So, I'm not actually grading it on Moodle or online. I'm grading it by hand and then typing it in there.")

Following these discourse types in frequency were "observing" (37 times), "speculating" (28 times), "planning" (26 times), and "explaining how the technology works or how to do something" (26 times). The occurrence of these discourse types in the attempting stage corresponds to the nature of the stage. The technical-support providers would note something they would see during their attempt at solving the problem ("observing"). For example, TS2 described what he saw while searching for an assignment in the gradebook: "O.K., so we have assignment three here. Um, and I do see it's grayed out." Also, the technical-support providers, and 7 times the user, would state the potential solution to the problem or problems in the attempting stage ("speculating"). TS2 stated once, "Um, I wonder if it's calculating correctly." U11 wondered allowed if one option would fix the problem: "'Allow student review'! Maybe that's it." To solve the problem eventually, the speakers would communicate about how they could solve it in some other way (they engaged in "planning" almost equally). For example, TS4 needed to consult the Moodle documentation: "I can get back to you as soon as possible. We do have the documentation." Also, U20 needed to ask a student to learn more about what the student was experiencing with the problem: "I will talk to her." It also makes sense that the technical-support

providers would explain how the technology works or how to do something if the users offered potential solutions that may not technically work. For example, U20 wanted different colors for the text editor, but TS4 explained the limitations of the Moodle system for providing color options. The problem was not resolved, but TS4 did provide information about how the technology works. In one instance, U2 told TS8 how to get to a certain part of the website as TS8 attempted to solve U2's problem. ("If you click on it, it pops up the window.")

One finding that runs counter to expectations for microlevel discourse occurring in the attempting stage was the single occurrence of "declaring the problem or problems as solved." The attempting stage by its nature does not have a solved problem. In interaction 9, U2 stated "[t]hat explains the problem for me then" in response to TS8's explanation of how the announcement feature works in Moodle. U2 intended to take TS8's advice, but according to the parameters of my coding scheme, I could not call the macrolevel discourse "resolving" because U2 never got her announcement feature to work during the interaction. For U2's case, it appeared U2 was satisfied with an explanation about how the announcement feature works; I could not speculate, however, that U2 ever got the problem solved and so kept to the parameters of my coding scheme by calling the macrolevel discourse attempting. For this reason, this appearance was a strange instance of microlevel discourse that closely resembled a code in my coding scheme but that did not fit it exactly.

It seems logical that one technical-support provider would ask a question about how the technology works in an attempt to solve the problem ("Would it be something to do with the [*inaudible*] block?"), and in frustration with not being able to solve the problem, technical-support providers might judge the

technology. (“So, it'd be nice if there was an option in here that you can say ‘always-’ ‘Always mail.’) Technical-support providers judged the technology 5 times, more often than users did (1 time). U2, for example, stated, “I have philosophical- philosophical differences with the Moodle- main Moodle administrator.”

Overall, the attempting stage’s microlevel content maintains its character as a stage in which problems remain unresolved. Technical-support providers signaled their way through the attempt. Technical-support providers declared that they didn’t know the answer. Both speakers speculated about the solutions, and both speakers made plans to solve them eventually in the future. These findings contribute to the existing research on macrolevel discourse in technical-support interactions because they reveal how technical-support providers work through situations in which they cannot resolve the problem. As Pentland (1995) reveals in his organizational communication study, technical-support providers find answers to problems, even when they themselves do not know the answers. The discourse in this stage demonstrates that these workers can contribute to organizations in other ways beyond mere technical know-how. This study confirmed that technical-support providers can demonstrate communicative competence in situations in which they do not have answers for users. They quickly found appropriate solutions that could bring the interaction or a certain problem within an interaction to a close by communicating their need for information, by communicating their lack of knowledge, and by communicating their intention to find solutions for users as soon as possible. I do not describe further the attempting stage because the focus of this study is on resolved problems. However, these results that I have explained in this chapter reveal the

relative size of the stage, the most frequent speakers, and the most frequent microlevel discourse types.

The Resolving Stage of Technical-Support Interactions

The resolving stage intends to solve the users' problems through technical instruction. By its nature, the presence of a resolving stage reveals a success for technical-support providers and effective interactions overall. To help describe the extent to which each speaker contributed to this stage, I quantified the word volubility of both the technical-support providers and the users in each of the 20 interactions. Of the 20 interactions, 17 contained this stage. In these interactions, a problem or more than one problem was resolved. Table 4.7 gives the volubility for this stage. The volubility counts range from 40 in interaction 3 and 3516 in interaction 19. The variation likely corresponds to the length of the interaction. Interaction 3, for example, was only 1:56 while interaction 19 was 27:42, one of the longest interactions. The longest interaction, interaction 16 (33:12), had the second highest volubility count (2075) for resolving. Of the total number of words in the data set (26023), resolving required 53.48% of the discourse, suggesting that resolving a problem requires the majority of the time and talk exchange in technical-support interactions.

Table 4.7 Resolving-Stage Volubility

Interaction	Participants	TS	U	Total
19	TS7(TS8)-U41	2690 (119)	707	3516
16	TS7(TS8)-U40	1401 (147)	527	2075
13	TS2-U40	1318	450	1768
20	TS2-U40	1239	510	1749
17	TS3-U35	534	175	709
7	TS7-U19	514	137	651
4	TS4-U20	344	282	626
11	TS7-U2	385	236	621
14	TS8-U2	458	156	614
18	TS2-U32	384	145	529
6	TS7-U2	209	186	395
1	TS2-U11	152	88	240
12	TS2-U5	90	68	158
15	TS3-U2	33	48	81
8	TS7-U2	58	22	80
2	TS2-U11	19	45	64
3	TS4-U2	9	31	40
Total (%)		**10103 (72.6)**	**3813 (27.4)**	**13916**

More revealing perhaps than the fact that resolving requires the most volubility is that technical-support providers were the most prominent speakers of the discourse in this stage. Except for interactions 3 and 15, the technical-support providers spoke more frequently. These two interactions were much shorter, however (the two shortest interactions at 1:56 and 1:49 respectively), suggesting that given the opportunity, the technical-support provider would speak the most during this stage. This finding makes sense given the fact that technical-support providers were the main agents of change in the discourse who facilitate the problems toward resolution. Without the technical-support providers' knowledge and communication of that knowledge, the problems could not get resolved. As another example of the power of technical-support

providers' speaking presence, in two interactions (interactions 16 and 19), TS8 stepped in, without which the problem would not have come to resolution.

In examining the stage more closely, I found that the resolving stage contained certain types of microlevel discourse relevant to resolving a problem. Table 4.8 gives the types of microlevel discourse in this stage and who spoke the discourse. The most frequent discourse type in this stage was "signaling" (151 times), with the technical-support providers using this strategy the most (139 times). This finding suggests that like attempting, the technical-support providers would signal what they were doing or about to do while they resolved the problems. The second-most frequent discourse type was "explaining how the technology works or how to do something," which corresponds naturally to the nature of resolving a technical problem. The user never explained information like this during the resolving stage. In fact, the most frequent discourse type spoken by the users during this stage was "inquiring to understand the technology," suggesting that the users continued to inquire to gain insight about the technology, even as the problem was on its way toward resolution. "Confirming or denying" appeared as the third-most frequent discourse type, with technical-support providers speaking this discourse more than users. As I discuss further in chapter 6, technical-support providers often used yes-or-no type answers in helping users to understand how the technology works or how to do something with it. For now, I note how the simple response ("Mm-hmm," "Yeah," "No") served to help technical-support providers to resolve technical problems. Lastly, a frequent discourse type in this stage was the users' inquiring to understand the technology (100 times). These inquiries came from users who needed confirmation following the initial technical instruction from technical-

support providers. ("So if somebody just drags and drops an image to their desktop and they click it, it will go to 'Preview?'") Of course, the answer to these questions were "explaining how the technology works or how to do something," but it was also a simple confirmation or denial ("Yes").

Table 4.8 Microlevel Discourse in the Resolving Stage

Microlevel Discourse and Speaker Type	TS	U	Total
Signaling	139	12	**151**
Explaining how the technology works or how to do something	141	0	**141**
Confirming or denying	78	37	**115**
Inquiring to understand technology	2	100	**102**
Giving background information	34	63	**97**
Observing	54	17	**71**
Telling	42	15	**57**
Inquiring to understand needs or background information	49	5	**54**
Showing how the technology works or how to do something	49	1	**50**
Stating needs	1	23	**24**
Speculating	18	5	**23**
Planning	8	15	**23**
Declaring the problem or problems as solved	2	15	**17**
Inquiring to check comprehension	4	4	**8**
Judging the technology	4	2	**6**
Inquiring to gain permission	2	3	**5**
Total	**627**	**317**	**944**

The microlevel discourse in the resolving stage corresponds naturally to the nature of the stage. Technical-support providers signaled to users what they would do or were doing as they resolved the problem; they explained how the technology works or how to do something with it; and they confirmed or denied inquiries from the users seeking to understand the technology. These results correspond to the procedural instruction that Steehouder (2007) argues occurs in technical-support interactions; they also show how "signaling" and "confirming and denying" function in these one-to-one technical communication interactions

as a uniquely interpersonal convention of technical help—a convention that written instructions do not share. For example, written instructions cannot supply yes-or-no type answers as a meaningful form of instruction. In "Frequently Asked Question" documents, moreover, the discourse emulates a one-to-one interaction with a user, for which a yes-or-no type answer serves as a viable form of instruction. In chapter 6, I describe these microlevel exchanges to showcase what each strategy looks like as users and technical-support providers succeed in resolving the problems. The results presented in this chapter reveal the relative size of the stage, the most frequent speakers, and the most frequent microlevel discourse types.

The Closing Stage of Technical-Support Interactions

The closing stage intends to conclude the technical-support interaction by ensuring that the problem or problems have been resolved or at least addressed (in the case of unresolved technical problems). To help describe the extent to which each speaker contributed to this stage, I quantified the word volubility of both the technical-support providers and the users in each of the 20 interactions. Of the 20 interactions, 15 contained this stage. For those 5 interactions for which there was no closing, the recording ended rather abruptly after the problem was resolved. As with the identifying stage, I may have lost some of these exchanges in the interactions because the audio and video recording ended before all closings could occur. This low number also corresponds to the kinds of data that were relevant for this study; I did not code or analyze affective discourse such as joking and small-talk, which occasionally occurred in the closings but was not coded for this analysis. Speakers exchanged good-byes or other social

conventions, but I do not report these data here because they did not relate to solving the technical problem at hand. That said, the data I report here from the 15 interactions do in fact reveal the kind of discourse that occurs during this stage, specifically discourse that relates to the technical-help problems that generate the interactions in the first place. Table 4.9 gives the volubility for the 15 interactions that had this stage. The volubility counts range from 9 in interaction 9 to 225 in interaction 6. The variation is minimal for these interactions, and overall the closing stage is rather short. Of the total number of words in the data set (26,023), closing required 3.28% of the discourse, suggesting that, similar to identifying, closing requires minimal time and talk in technical-support interactions, specifically as far as the discourse relates to the technical problem. Overall, the speakers spoke a relatively equal number of words, with the technical-support providers speaking only a little more (52.2%). In 6 interactions, the technical-support provider spoke no discourse related to the problem, but in all of these 6 interactions, the closings were relatively small overall. In longer closings (interactions 5 and 6), the technical-support providers spoke much more equally with the users.

Table 4.9 Closing-Stage Volubility

Interaction	Participants	TS	U	Total
6	TS7-U2	121	104	225
5	TS2-U14	100	40	140
8	TS7-U2	79	55	134
7	TS7-U19	85	34	119
10	TS2-U23	31	0	31
12	TS2-U5	12	18	30
20	TS2-U40	0	28	28
2	TS2-U11	0	27	27
11	TS7-U2	4	22	26
15	TS3-U2	0	25	25
3	TS4-U2	0	22	22
16	TS7-U40	0	14	14
19	TS7-U41	2	10	12
4	TS4-U20	11	0	11
9	TS8-U2	0	9	9
Total (%)		**445 (52.2)**	**408 (47.8)**	**853**

In examining the stage in more detail, I found that the closing stage contained certain types of microlevel discourse relevant to closing a technical-support interaction. Table 4.10 gives the types of microlevel discourse in this stage and who spoke the discourse. The most frequent discourse type, "planning," reveals that technical-support providers and users engaged in the last moments of the interaction making plans related to solving the problem. ("I'll look into it.") The second- and third-most frequent discourse types were "giving background information" (9 times) and "declaring the problem or problems as solved" (8 times). In the cases in which technical-support providers and users would give background information, they would reflect on or restate an experience they had with the problem. ("That parent category just wasn't popping up for me.") Users would also declare the problem or problems as solved. ("Well, I think that

answers all my questions.") Technical-support providers never made this claim in this stage, which further reveals that the users largely controlled the flow of the conversation and could end it more appropriately than the technical-support providers could. Also worth noting is the "speculating" discourse type that occurs as both users and technical-support providers reflected backward on the technical problem to speculate what may have caused it. ("So it must have reloaded or something.")

Table 4.10 Microlevel Discourse in the Closing Stage

Microlevel Discourse and Speaker Type	TS	U	Total
Planning	12	6	**18**
Giving background information	2	7	**9**
Declaring the problem or problems as solved	0	8	**8**
Confirming or denying	2	3	**5**
Speculating	2	2	**4**
Inquiring to understand needs and background information	2	1	**3**
Judging the technology	0	2	**2**
Explaining how the technology works or how to do something	1	0	**1**
Total	**21**	**29**	**50**

As appropriate for a closing stage to a technical-support interaction, the microlevel discourse in this stage corresponded to wrapping-up a one-to-one interaction revolving around a technical problem. Participants would plan what they would do next after the interaction, they would relate experiences they had with the problems, or they would declare the problems addressed.

Technical-Support Interaction Macrolevel Discourse Chains

Research question 1 for this study asked whether the technical-support interactions followed the established macrolevel discourse reported in other research (Clark, Murfett, Rogers, and Ang 2012; Steehouder 2007; Xu, Wang, Forey, and Li 2010). This study found that technical-support interactions do

follow the established macrolevel structure, except for one component—the attempting stage, which characterizes the exchange when the technical-support providers cannot resolve the technical problems. Apart from this difference, however, the established discourse structure from the research does characterize technical-support interactions.

This study reveals a few changes in the ordering and frequency of certain macrolevel discourse elements, however. Figure 4.2 describes the macrolevel discourse chains for each session. It reveals that defining (D) is followed either by identifying (I), attempting (A), or resolving (R) in every instance. Furthermore, these chains reveal that identifying, if available, almost always occurs after the users and technical-support providers diagnose the problem, at least partially. In post-session stimulated-recall interviews, TS7 and TS8 confirmed that it was almost always the case that users would begin by describing their problem before the technical-support provider could identify their name with the system. This finding highlights a unique order to the discourse that the other research has not revealed. Research into other technical-support interaction contexts may reveal if defining coming first is unique to technical-support interactions in general or if this finding is unique only to this specific case study. The discourse chains also reveal that closing, when available, always comes last. Lastly, the discourse exchange reveals that iterations, or repetitions, are possible, depending on the number of problems the user brings to the interaction. A new defining stage signals that a new problem was introduced into the interaction. In the longest interaction, interaction 4, TS4 and U20 defined problems 14 times. Also, this study showed that attempting and resolving are optional, either one or the other will come after defining. If an established structure could be demonstrated

from these 20 interactions, these findings reveal that the following generic structure potential describes technical-support interactions (figure 4.1):

{I→D}→{(A) → (R)} →C

Figure 4.1 Generic Structure Potential for Technical-Support Interactions

The arrow symbol (→) indicates sequence, the curly bracket ({) shows that the degree of iteration for enclosed elements is equal, and the parenthesis () show that the enclosed elements are optional. I=identifying, D=defining, A=attempting, R=resolving, C=closing

Interactions 3 and 15 reveal an ideal and successful structure—I→D→R→C—while interaction 4 reveals a technically complete but still unsuccessful interaction—I→D→A→C.

Conclusion

This chapter has discussed the macrolevel discourse in 20 technical-support interactions. I presented the volubility (word counts) and microlevel discourse types for each macrolevel discourse type—identifying, defining, attempting, resolving, and closing. I then presented the discourse chains for each interaction and what can be considered the generic structure potential for technical-support interactions. Through this discussion, I describe the unique role each stage plays in the technical-support interaction and how this research confirms, and in some cases challenges, the existing research on the established macrolevel discourse for customer-support interactions. In the next chapter, I discuss examples in more depth to reveal technical-support providers' and users' microlevel discourse in the defining stage of these interactions. I present the contexts for and larger excerpts from specific exchanges in order to provide a rich description of the microlevel discourse types for each speaker as they define the technical problem.

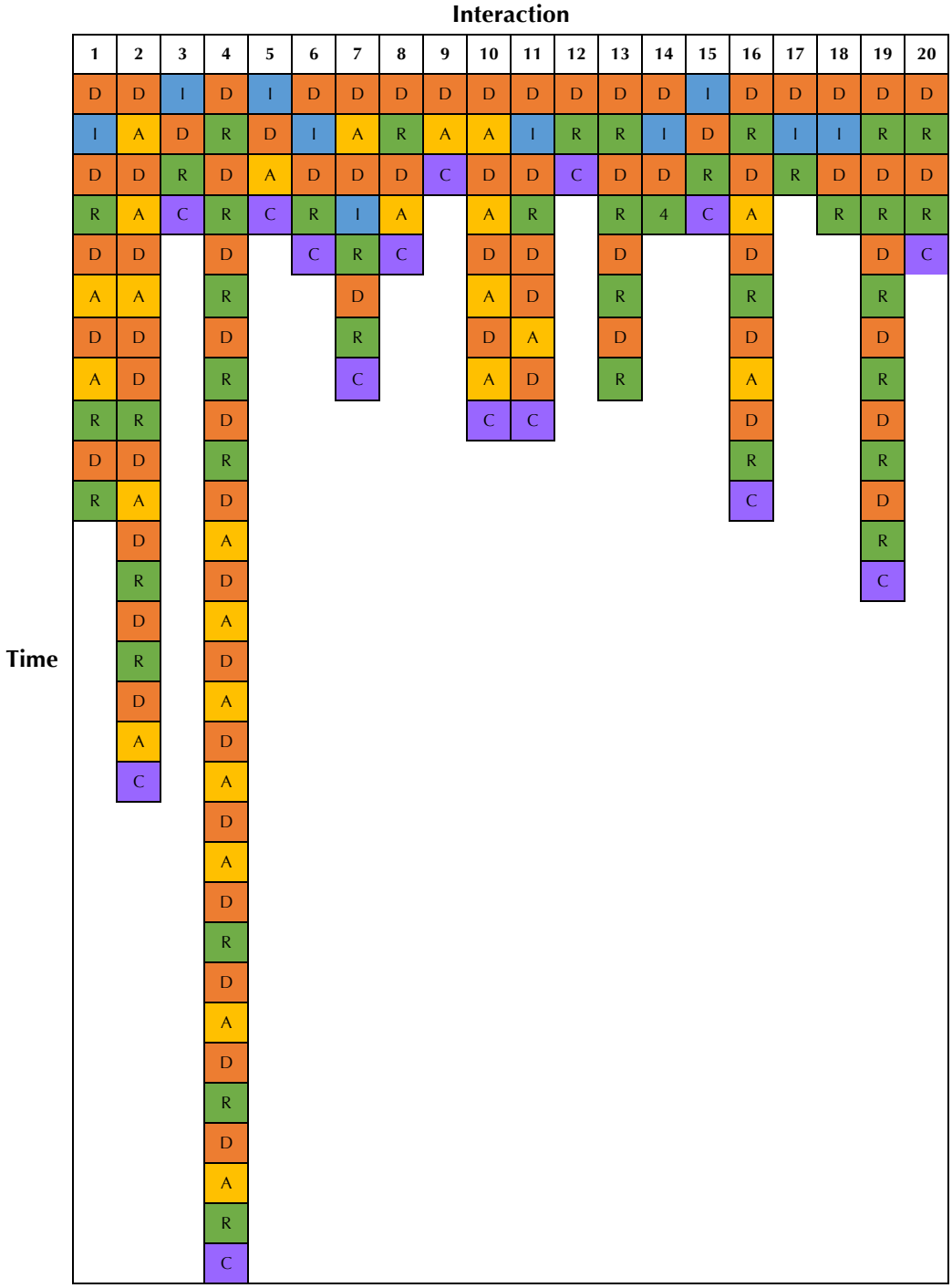

Figure 4.2 Macrolevel Discourse Chains in 20 Technical-Support Interactions

Each of the 20 interactions (x-axis) had a certain number of macrolevel discourse stages over time (y-axis). I=identifying, D=defining, A=attempting, R=resolving, C=closing

CHAPTER 5: MICROLEVEL DISCOURSE STRATEGIES FOR DEFINING TECHNICAL PROBLEMS

In this chapter, I describe the defining stage in greater detail by presenting excerpts of both technical-support providers' and users' discourse as they define the technical problems. I first describe the discourse strategies that the technical-support providers employed. Then I describe the strategies that the users employed. I end by discussing common exchange patterns between technical-support providers and users, noting how some microlevel discourse strategies came in typical pairs.

Table 5.1 gives the overview of the types of microlevel discourse spoken in the defining stage and the participants that spoke the discourse.

Table 5.1 Microlevel Discourse in the Defining Stage

Microlevel Code	TS	U	Total
Giving background information	9	127	**136**
Inquiring to understand needs or background information	80	11	**91**
Confirming or denying	32	52	**84**
Inquiring to understand technology	4	40	**44**
Stating needs	2	39	**41**
Telling	4	32	**37**
Signaling	29	7	**36**
Observing	8	8	**16**
Explaining how the technology works or how to do something	10	4	**14**
Inquiring to check comprehension	2	7	**9**
Inquiring to gain permission	1	6	**7**
Speculating	1	6	**7**
Judging the technology	1	4	**5**
Showing how the technology works or how to do something	3	2	**5**
Planning	2	2	**4**
Total	**188**	**347**	**536**

The defining stage contained microlevel discourse relevant to defining the technical problem. The most frequent discourse in the defining stage was "giving

background information" (136 times) followed by "inquiring to understand needs or background information" (91 times). It makes sense that this pair of questioning and answering would occur in this stage because the defining stage exists to help the two speakers understand the problem so that they can resolve it. Also, the technical-support providers inquired about needs or background information (80 times), and the users provided such information (127 times), either in the response to technical-support providers or on their own initiative. Two of the common ways users responded to technical-support providers, apart from "giving background information," was "inquiring to understand the technology" (40 times) and "stating needs" (39 times). "Confirming and denying" was the third-most frequent microlevel discourse type (84 times), but again, users provided this discourse more frequently (52 times), suggesting that the users would provide "yes-or-no" type responses to the technical-support providers' questions. The following sections present these results more closely and in greater detail.

Technical-Support Provider Discourse Strategies in the Defining Stage

The technical-support providers, as discussed in chapter 4, spoke less in this stage than users did, likely because the users spent time giving information while the technical-support providers spent time seeking it. In this section, I describe the technical-support providers' discourse strategies.

Inquiring to Understand Needs or Background Information

The most common microlevel discourse for technical-support providers in this stage was "inquiring to understand needs or background information" (80 times). As implied in the discourse code's name, this discourse used

interrogative syntax (structured as questions with question marks) or declarative syntax (structured as statements with periods). As mentioned in chapter 2, however, I collapse these distinctions into one name "inquiries" because the underlying illocution was to gain information, despite the syntactic structure the participants used.

Also, implied in the discourse name is that speakers sought two types of information: needs or background information. In early stages of the microlevel discourse code development, I separated these two types into separate codes. However, to increase reliability in my coding scheme and because my independent coders had difficulty identifying the two types, I collapsed them into one code. Despite this situation, I discuss this discourse type by distinguishing the types of information about which the speakers inquired. This distinction helps me to describe this discourse more richly and helps technical-support providers and others to see how it functions in an exchange.

Technical-support providers most often inquired to gain background information (59 times). This emphasis makes sense because technical-support providers required contextual information about the users' needs to adequately define the problem. Table 5.2 describes the types of background information that I identified technical-support providers inquiring about.

Table 5.2 Inquiring to Understand Background Information

Background Information Types	Examples
Inquiring to understand where the problem is located	This one for participation? Could you let me know the student who dropped your course? So which assignment did [she actually submit? Which is it? Do you know a name of one of the students that's having trouble? "Grades?" Is it this one? What your course is?
Inquiring to understand experiences with the technology	So, when you go to "add new topic" [*1 second; clicking on "Add new topic"*], and you click the "mail now" button [*1 second; clicking "Mail now" radio button*]- it didn't go out until what, like [*1 second*] [sometime- So you said that this entire editor box gets bigger? They're not showing up clearly? So were they getting in or were they stuck on the log-in? just using this right? [*cursor pointing to "Announcements" module*] It resets to the current time?
Inquiring to understand previous actions	And did you duplicate the quiz when you created the new one or did you just create a new one from scratch? And they both turned in the same things? Do you have the assignment set up? So they dropped your course- classes? They dropped it?

Technical-support providers inquired about where the problem might be so that they could have access to the problem. In one example, TS4 asked U20, "Could you let me know the student who dropped your course?" TS4 did this

while searching a roster of students, clearly looking to find that student so that she could suspend that student from the course website. In another instance, TS7 asked U2, "'Grades'?" while navigating to the place where U2 stated she was having a problem making course attendance as extra credit. TS7 appeared to ask if that was the place where she should check. It makes sense that technical-support providers would seek this kind of background information because they need to know where users encountered the problem. Such background information helped the technical-support providers to define the problem specifically in relationship to the technology.

Technical-support providers also inquired about users' experiences with the technology. In interaction 4, TS4 asked U20 to confirm her experience with the textbox in the Moodle grading system: "So you said that this entire editor box gets bigger?" This information likely helped TS4 to ensure that she and U20 had similar frames of understanding (Agar 1985, 153). TS4 likely better understood U20's experiences by confirming that experience through words that resonated with U20. Likewise, in interaction 9, TS8 confirmed that U2's message never went to her students' email inboxes when she used the announcement module in Moodle: "It didn't go out at all? At all?" These kinds of inquires helped the technical-support providers to ensure that they understood the problems by giving the problems more definition and by providing users the opportunity to help define it further through their responses to the inquiries.

Technical-support providers also inquired about users' and others' (such as students') previous actions. This information, TS2 told me in a post-session stimulated-recall interview, helped him to walk through potential causes of the problem, whether users or the system caused the problem. For example, TS2

asked U23 how she created a quiz that was not behaving the way she wanted: "And did you duplicate the quiz when you created the new one or did you just create a new one from scratch?" This inquiry seemed to help TS2 to determine if U23's previous actions caused the problem or not. In another case, TS4 wanted to ensure that a student had dropped a course before removing that student from the course website: "So they dropped your course- classes? They dropped it?" This information helped TS4, she told me, to ensure that she followed protocol when handing a common request from instructors (to remove students from the course website when they drop the course). According to TS4, technical-support providers had to ensure they confirmed this information so that instructors and students did not encounter a deeper problem when needing to recover a deleted profile, should a student who did not officially drop the course return from a long absence. This inquiry demonstrates this technical-support providers' awareness of organizational power (protocols they must follow) and of previous experiences with similar problems. This awareness confirms Pentland (1995) when he explored the organizational awareness and problem-solving capacity of technical-support providers.

Some types of background information were not as clearly distinguished as these examples were, which is one reason these background types are not discourse codes in my coding scheme. In interaction 9, TS8 walked through an operation to understand U2's previous actions before inquiring about U2's experience with the technology—all in the same sentence and with seemingly two purposes in mind for the utterance:

TS8: So, when you go to "add new topic" [*1 second; clicking on "Add new topic"*], and you click the "mail now" button [*1 second; clicking "Mail*

now" radio button]- it didn't go out until what, like [*1 second*]

[sometime-

Using this strategy TS8 could understand not only what U2 did in attempting to resolve the problem but also U2's experience with how the technology behaved when U2 attempted to resolve it. TS8 told me in the post-session, stimulated-recall interview that he purposefully wanted to identify if U2 did not know that clicking "Mail Now" was required for the message to go to students' email inboxes. Without selecting the option, the announcement simply went to the announcement archive on the course website. Indeed, one of U2's next statements was, "I'm- not sure that I actually clicked the 'mail now' checkbox," indicating that TS8's strategy yielded the intended result, to determine and ultimately define the problem.

As mentioned previously, this microlevel discourse "inquiring to understand needs or background information" implies two types of information that technical-support providers sought: background information and needs. Technical-support providers inquired to understand needs less frequently (21 times). Table 5.3 gives categories and examples of technical-support providers inquiring to understand needs.

Table 5.3 Inquiring to Understand Needs

Need Types	Example
Inquiring to understand specifications	Like [*semester year*] maybe? Or section [*number*]?
	So you want it to display the question- again- but have it be checked or-?
	So you them to keep doing it until they get it right?
	You need it for your classes- You need it for [*course number*]? And for [*course number*]?
	O.K., so you want to look at the gradebook [*1 second*] and if we can make something extra credit, right?
Inquiring to understand general needs	And who's next?
	So what- can I officially- help you with now?
	Was that the main issue you were concerned about, the shadow or? Or?
	What can I help you with?
	So what is your question?

Technical-support providers tended to inquire about the specifications of users' needs. In one example, TS2 helped U11 create a lesson module for U11's course website. As TS2 set it up for U11, TS2 inquired about the specifications U11 wanted for the lesson's behavior. For example, "So you want them to keep doing it until they get it right?" Here TS2 asked if U11 wanted her students to have the ability to retry tests until the students got the correct answer. Also, TS2 asked, "So you want it to display the question- again- but have it be check or-?" In this example, TS2 asked a question to understand exactly how U11 wanted the lesson to appear as students went through the test.

In another example, TS7 asked U2 precisely how U2 wanted to treat an assignment in the Moodle gradebook: "O.K., so you want to look at the

gradebook [*1 second*] and if we can make something extra credit, right?" These examples demonstrate technical-support providers' careful tendency to understand what current situation that users were facing and what they wanted the technology to look like or do for them.

In these interactions, technical-support providers carefully considered that they were meeting users' needs. In other cases, technical-support providers wanted to understand needs in general. As may be expected, a conventional question was "what can I help with" or a variation on it such as "so what is your question?" In a slightly more customized variation, TS2 confirmed that he was meeting U23's specific needs with an image on U23's website: "Was that the main issue you were concerned about, the shadow, or? Or?" As I discussed in chapter 4, defining often came first in the interaction, before identifying. It seems that this general needs inquiry ("What can I help you with?") perhaps prompted this sequence in the macrolevel discourse, especially if it was the first question technical-support providers employed. Users' responses likely would not be identifying institutional information corresponding to the identifying stage but instead some explanation of the problem. To investigate this possibility, I found which defining stages began with this conventional inquiry. I found 8 defining stages that began with this kind of inquiry, all of which were interactions that began with a defining stage rather than an identifying stage (interactions 2, 4, 10, 14, 17, and 19). Of the 8 defining stages that began with this conventional question, 2 of the stages appeared in the same interaction. Other defining stages began with inquiries about specific needs. This investigation reveals that the inquiry to understand general needs does not appear to affect the order of the stages. Future studies, especially studies that do not ask participants to manage

the audio and video recording but that could catch any and all beginning dialogue, could investigate whether the defining stage begins these interactions and what microlevel discourse may cause that stage to occur first.

This discussion of the most common discourse strategy that technical-support providers use indicates that technical-support providers were aware of the power and purposes behind their inquiries, using them strategically to better define the problem. This research confirms Agar's (1985) claim that the "diagnosis" stage, as he calls it, involves a series of questions from the institutional representative (152).

Confirming or Denying

The second-most frequent microlevel discourse that technical-support providers employed in the defining stage was "confirming or denying." Because this microlevel discourse is a short "yes-or-no" type response, it required contextual information, including the discourse that surrounds it. Specifically, my data set, filtered to show all the instances of this code revealed a series of statements such as "Yeah," "Mm-hmm," "Uh-huh," "No," or "Right." Such discourse requires further exploration to fully understand how it functions in each interaction. I investigated the 32 instances in which the technical-support providers employed "confirming or denying" and examined the discourse to which the technical-support provider responded. I found that technical-support providers responded with confirming or denying most frequently to "to inquiring to understand the technology" (13 times).

In these instances, a confirmation served as a means for instruction. For example:

U20: O.K. So the first one is that I have some student names on here-

TS4: Mm.

U20: -which I want to take off. Is there a way of doing that?

TS4: Uh, O.K. **yeah**, I [think-

In this interaction, TS4 started by confirming that Moodle could do what U20 wanted before TS4 began to inquire more about which students U20 wanted to remove from the website. In another interaction, TS2 directly confirmed what Moodle could do for U11:

U11: Yes, that's the one, and I just want it an exact duplicate.

TS2: O.K.

U11: That's pretty easy, right?

TS2: **Yes, it is very easy.**

In some cases, however, the confirmations and denials were less direct and more uncertain, serving as a springboard for explanations about the limitations of the technology or for further inquiry. In interaction 1, for example, TS2 responded less directly to U11's question about the settings in lesson modules that would enable the module to automatically respond to students' answers. It provided a less direct denial in answer to U11's inquiry:

U11: O.K. Um, all right so we don't have [*1 second]* control over the behavior of each [*2 seconds*] lesson question response?

[*2 seconds*]

TS2: **Well you would.**

TS2's no-type response was less direct, using the mitigating word "well"; however, the utterance did precede an explanation about how the technology

works and how it could not be controlled, as far as TS2 knew, in the way that U11 would have liked. TS2 ended his explanation with another denying statement:

> TS2: You would be able to enter a manual response but what you want is for it to actually show [*1 second*] like check boxes and things like that and I'm not- off the- anyway off the top of my head. **There might be a way to do it- but I'd have to do some digging.**

After explaining what Moodle can do, TS2 denied U11's question by stating that it might not be able to do what U11 wanted—TS2 didn't know. At the end of the interaction, TS2 directly denied U11's insistent inquiry about the same issue: "Well I- I don't know how to do it."

In another interaction, TS2 hedged in answering U23's question about the unwanted behavior of a quiz module in Moodle:

> U23: I can't remember it now. [*3 seconds*] Would that make a difference? If it's duplicated?
>
> TS2: **Possibly.**

TS2 then proceeded to inquire more about U23's experience with the problem, careful not to imply that the problem had been defined without gaining further information. These instances demonstrate how technical-support providers employed short, yes-no-type responses to either directly or indirectly help users to understand the technology or to communicate that the technical-support providers did not know the answers. The short responses tended to set up further explanation or inquiry, without which the defining of the problem may not have been possible. These findings do reveal that technical-support providers stated their lack of knowledge soon after hearing questions about the problem,

especially if users insistently asked about the problem. However, within the defining stage, technical-support providers continued to investigate the problem further until it was clear to them that they could or could not define it.

The second most frequent microlevel discourse to which technical-support providers responded with confirming and denying was "inquiring to understand needs or background information" (8 times). As I discuss later in this chapter, users employed inquiries of this type during the defining stage because they wanted to see what the technical-support providers needed either to define or to resolve the problems. Technical-support providers' responses to these inquiries were short, yes-no-type responses. In interaction 1, for example, U11 wanted to confirm if TS2 needed to gain permission to do what U11 required:

U11: And, um, I don't know if you need to talk to [*manager*] about it and he needs to talk to [*manager*] or whatever? But uh-

TS2: **No, I don't think so.**

Here, U11 wanted to know what TS2 needed to define the problem adequately. In the same interaction, U11 asked if TS2 required specific information before TS2 could do what U11 required, information that U11 would need to retrieve later if he did require it:

U11: O.K. Uh- Do you need- a name and designator before you start the duplication?

TS2: **No.** We can always change the name later.

Similarly, in another interaction, U20 confirmed that TS4 had the information she needed to resolve the problem, in this case, the information about the course:

U20: [*Course number*], [*Course name*]. [*1 second*] Is that what you need?

TS4: **Uh-huh**

In another instance, U2 hoped that TS4 was ready for the final question in a long series of questions:

U20: You ready for my last question? [*laughing*]

TS4: **Yeah sure. [*laughing*]**

Indeed, interaction 4 was the second longest interaction (31:02) with 14 different problems that U20 wanted to define and resolve. U20's question, and their collective laughter, reflects their mutual awareness of how long the interaction had gone.

These responses indicate that technical-support providers experienced reciprocation from users about not only the users' needs but also the technical-support providers' needs. The presence of these inquiries demonstrates that users were aware that technical support providers required specific information or general readiness to diagnose and to resolve problems.

The third-most common inquiry to which technical-support providers responded with a short, yes-or-no type response was "inquiring to check comprehension" (6 times). Most of these cases appeared in interaction 4 (5 times), indicating that one of U20's interactional patterns was to check for comprehension, specifically checking that TS4 could "see" something in U20's experience with the technology. The following examples reveal a consistent use of "see" and TS4's confirming response to U20's inquiries.

In one example, U20 gave background information about U20's practice to mark student work with multiple colors:

U20: See how I use color- a whole lot?

TS4: **Mm-hmm.**

U20's checking that TS4 could see the use of color set up another exchange of this sort:

U20: See like this one that I converted from a previous time and they use purple?

TS4: **Yeah.**

U20: See?

Again, U20 ensured that TS4 could see the problem as U20 saw it: that in previous courses, Moodle let her use a darker purple to mark student work. However, at that moment Moodle did not have dark colors except for red, which U20 did not want to use. She then checked to see if TS4 could see the problem when she used one of the other color options:

U20: See how light that is? With that light blue? See I-

TS4: **Yeah.**

After moving on to another problem, U20 noticed the same problem again and brought it up again:

U20: See how it doesn't show up very much?

TS4: **Yeah.**

It's possible that TS4's responses could simply be a backchannel. Indeed, I specifically avoided yes-no-type discourse that served as backchannels and polite reciprocations. However, I coded these instances because they demonstrated that a minimal response ("Mm-hmm" and "Yeah") could serve as a strategy to help users and technical-support providers understand one another. In these instances, TS4's responses were not merely polite signs of following along but instead engaged and precise indications that TS4 saw what U20 saw. In

fact, this kind of exchange appeared in interaction 6 when U2 asked a similar question as U20 did, and TS7 provided a confirming response:

U2: O.K. See how it says "submissions closed"?

TS7: **Uh-huh.**

The implication of this type of exchange is that technical-support interactions, in particular helpdesks in which technical-support providers and users share the same space, may involve a reciprocating exchange that the user and technical-support provider can "see" the problem the same way, which confirms research by Agar (1985) and Steehouder (2007) when they emphasize that problem definition requires aligning the clients' and institutional representatives' understanding. In this case, the participants came to that understanding through a shared vision.

Signaling

The third-most frequent microlevel discourse that technical-support providers employed in the defining stage was "signaling" (29 times). Because this microlevel discourse appeared relatively obscure when seen alone, it required contextual information to identify it, including the discourse that surrounded it. I noted two main types of "signaling" in the defining stage: "let" phrases and thinking aloud.

Technical-support providers used "let" phrases to signal what they were about to do. For example, TS4 signaled that she would be silent while she investigated U20's first problem: "Let me see." At another point TS4 signaled that she would look at her course website to help define the problem: "Let me see

that- let's go to my site." Similarly, TS2 signaled in another interaction what he was about to do:

> TS2: Let me just log in as a student really quickly- and then we'll see what the student sees. It's going to help me kind of see what's going on [*logging-in as student*]

TS7 used this strategy as well, signaling that what she would try to do to investigate the problem further:

> TS7: Let's uh- [*2 seconds*] check "edit settings." [*1 second*] And see if there's some weird permissions thing that's going on.

In another example, TS6 employed similar signaling discourse: "All right, so let's open Moodle." These signals, all technical-support providers told me, helped not only to ensure that users knew what was going on ("so I don't look like I'm just clicking around," TS6 said) but also to instruct the users about how to use the technology. TS6 told me that this strategy, if the users were paying attention ("sometimes they're on their phones or whatever," she said), could help users become familiar with the website and with ways of using it.

Some signals of this nature, however, described actions just completed. In one case, TS7 appeared to begin a signal about what she would do, but she paused to signal what she just did:

> TS7: Let's try it now that this is- So I disabled the due date and the what-cha-ma-call-it. The cut-off.

Other signals did not use the "let" structure but used some variation with the same signaling purpose: "So have a go with this one," "We're going to look for it together," "I'm going to go into the courses," and "I'm not going to even to try to guess."

All these instances demonstrate that technical-support providers were aware of their users' needs to understand what was happening during moments in the interaction, providing users context for the actions technical-support providers would take with the technology and in some ways helping users to learn how to use the technology themselves.

Technical-support providers also used signaling to "think aloud." All technical-support providers said they thought aloud by reading the screen (the names of buttons or text that they see or select). This process helped them to think as they worked to define the problem, and they also noted that it helped them to demonstrate how the technology works and how to navigate it. The following are examples of the think aloud signals found in various interactions as the technical-support providers worked to define the problems:

- "Users" [*reading a button*]
- "Lesson plans" [*reading a heading on the website*]
- "Rationale for Unit" [*reading the name of an assignment module*]
- "Restrictions" [*reading a button*]
- "None" [*reading a radio button option*]
- "Rhetorical Analysis" [*reading the name of an assignment module*]
- "Honors" [*reading the title of a course*]

In other instances, the TS7 read aloud the names of a roster as she scrolled down the roster looking for a particular student. These instances reveal again, as TS6 shared, that technical-support providers also wanted to instruct users in part about how to use the website.

Not all thinking aloud, however, was reading. In other instances, the technical-support providers would think aloud questions they were having at that moment. They appeared to do this to tell the users that they were searching for something. In one instance, TS4, signaled that she was in the process of looking for a version of the text editor that might explain why U20 seemed to have more color options previously. After spending moments searching, she said aloud "Where is it?" While this discourse may be considered as "inquiring to understand the technology," I coded it as signaling because of the context in which it occurred. In this instance, TS4 appeared to use the question as a way of explaining what she was doing, not fully expecting U20 to provide an answer (how would U20 know?) but instead accounting for the wait-time as she searched for the different text editor option. In another instance, TS4 stated "activities, maybe?" as she searched the website for the correct location of U20's lesson plan assignment. Here again, TS4 seemingly accounted for the wait-time by thinking aloud to signal that she was searching for something.

Through "let"-type phrases and thinking aloud, technical-support providers signaled to users what they were doing or what they did. Furthermore, they coordinated the social dimension of these interactions by accounting for silence or wait-time, making the user aware of where they were in the interaction. Lastly, technical-support providers meant to instruct users through the signals, hoping that they could show various operations as they moved through the technology interfaces. This purposeful discourse confirms Agar (1985) and Barley (1996) who argued that institutional representatives, and specifically technical-support providers as Barley argued, manage multiple

frames of discourse in their interactions with their clients, showing that they are discourse workers who manage information for the sake of users.

User Discourse Strategies in the Defining Stage

The users spoke more overall than technical-support providers in the defining stage. The most common microlevel discourse they employed was "giving background information" (127 times). This finding corresponds to the most common microlevel discourse technical-support providers employed ("inquiring to understand needs or background information"). The second-most common microlevel discourse users employed was "confirming and denying," or short, minimal responses (52 times). Lastly, the third-most common microlevel discourse was "inquiring to understand technology" (40 times). In this section, I describe how users employed these strategies in more detail.

Giving Background Information

The most frequent microlevel discourse that users employed in the defining stage was "giving background information." Corresponding perhaps to the inquiries from technical-support providers, users, who spoke more overall in this stage, gave either short-form or long-form information about the technical problem or problems they were facing. The maximum volubility for one of these units was 131 words, and the minimum was 1 word. The length varied based on the potential purposes behind the information users gave. I deciphered three basic categories related to this microlevel discourse: stating previous actions taken, describing experience with the problem, and stating location on the technology. The most common form was stating previous actions taken (58 times). Because these descriptions focused on actions taken, they often sounded

like stories or narratives. For example, U32 began the defining stage in interaction 18 with a long narrative about previous actions she and others had taken:

> U32: when I had, um, someone from [*team name*], it was [*TS8*]- He came to my classroom and showed my students how to set up their e-portfolios, and he used mine as an example. It was the one that I had set up during a training session for [*course name*] students. So I had like some of my own things, and then [*TS8*] was like "you don't want this visible to your students, you need to have your e-portfolio for [*course name*] separate from like whatever it was that I had it." And he said he could help me if I came here, which I then completely ignored until now because I've actually got an assignment due, of course this week. So of course now I'm coming in to ask for help.

In this instance, U32 gave background information behind the problem by describing what she and in this case TS8 did in the past. The information served to spring board the conversation into the defining stage. In another example, U20 described her problem with a student's assignment submission:

> U20: I had a student who came in last week, and I told her to come in because she wasn't submitting things on Moodle, and I said, "I'm missing this, and this assignment." And she said, "well I submitted them." And um, so then I said, "well do it again and I'll see if it shows up on here," and she did it on her computer and I- She showed me where it says "submitted," and it never ever showed up for me.

U20 and TS4 then exchanged more information about the events in order to define the problem further. In another, shorter example, U5 narrated a quicker story:

> U5: I am teaching this [*course number*] course online. Gave a quiz. Now I'm looking back, and I think one of the answers is a little ambiguous.

She then asked if it was possible to have the system automatically award students the points for the problematic quiz question.

In these examples, users narrated actions they or others had taken in relationship to the larger context around the problem. In other instances, users shared previous attempts they had taken in solving the problem. When trying to explain more about a problem with an assignment module's cut-off settings, U2 stated what actions she had taken to solve the problem:

> U2: So there, the "allow submission from" and "due date" and "cut-off date." So I just disabled those because they were causing problems. But if you enabled them and set them- um- and then click "save and return to course," then it resets probably two thirds of the time, for whatever date time it is right now

Here U2 discussed her actions of disabling and enabling the cut-off date settings. She also accounted for her attempt in solving it, specifically what buttons she was using:

> U2: So the only thing that I've been playing with is under this "availability," um, so but you know the cut-off date isn't until sixteen hundred tonight.

In another instance, U2 hoped she could collapse the weeks on the course website so that it was easier for students to navigate the website. However, when she did so, Moodle would hide the assignments (and their corresponding grades) in the gradebook if those assignments' modules were in the hidden weeks. Because of this missing information, students could not see their grades for those assignments. U2 stated her actions: "And so my solution was just to open everything up." These examples reveal that users would describe the actions they took or that others took in relationship to the problem. Users would explain those actions whether to shed more context on the technical problem or to state what specific attempts they made to solve the problem.

Users also gave background information to describe their experience with the technology (56 times). For example, U20 shared her experience typing feedback to students:

> U20: I'll be typing and all of a sudden it will get bigger. And I don't know how to make it- like this part [*pointing to U20's personal computer*] this whole thing will get bigger-

In one instance U19 shared the experiences of his students attempting to enroll into the course website:

> U19: And I had a few students approach me with difficulties getting out of Moodle, so some students have said that their name just like wasn't- or that the course just wasn't showing up when they went to [the]- website. And other students said that the course was showing up, but the password wasn't working.

He continued to share other experiences students were having:

> U19: I mean they said that like they went to, you know, [*Moodle website URL*] and so on, but they didn't see [*course name*] there.

U40 shared also her students' experiences, in this case when they uploaded images to the ePortfolio system:

> U40: So, my students are creating a new page. And they want to add an image. The problem that we've had is that message "too large won't accept" comes up. So, I've been trying to figure out.

These descriptions of the problem provided appropriate information about the users' experiences with the technology. In some cases, they were at the beginning of the macrolevel stage when the problem was first introduced, or in other cases, they were follow-up descriptions in response to questions from the technical-support providers. For example, U19's initial description about his students' inability to access the course website began the defining stage, but his second description was in a response to a follow-up question from TS7 asking for more information about what U19's students saw.

Whether in response or not, this microlevel discourse strategy functions at the heart of the defining stage. By its nature, defining requires contextual information that gives the problems their appropriate level of distinction so that the technical-support providers could move toward resolving them. Users communicated this contextual information during the defining stage, as Steehouder (2003) points out, through narratives or sequences of events, or what Steehouder calls "historical reports" (3). In this way, this study confirms the tendency of users to share information in this way during the problem formulation part of a technical-support interaction.

Confirming or Denying

The second-most frequent microlevel discourse that users employed in the defining stage was "confirming or denying." Because this microlevel discourse is a short "yes-or-no" type response, a minimal response, it required me to investigate the contextual information, including the discourse that surrounded it. I investigated the 52 instances in which the users employed "confirming or denying" and examined the discourse to which the users responded. I found that users responded with confirming or denying most frequently to "inquiring to understand needs or background information" (42 times). The other 10 instances were in response to "stating needs" (2 times), "confirming and denying" (1 time), "telling" (1 time), "signaling" (1 time), "giving background information" (1 time), "inquiring to understand the technology (2 times), "inquiring to gain permission" (1 time), and "inquiring to check comprehension (1 time). Clearly, the main purpose users employed confirming or denying was in response to technical-support providers inquiries about needs or background information.

As I discussed previously, technical-support providers would inquire about specific needs or about general needs. Also, technical-support providers would inquire about the location of the problem, the experience with the problem, or the previous actions users took. Users never responded to an inquiry about general needs with confirming or denying. It makes sense that a response to "How can I help you?" would not be a yes-no minimal response. However, the other types of inquiries to which users responded with minimal responses were relatively equal in frequency. Users employed minimal responses to inquiries about specific needs (10 times), previous actions (11 times), experiences

with the problem (9 times), and the location of the problem (10 times). Further study could reveal more about users' minimal responses to technical-support providers' inquiries; specifically, a larger data set might reveal more differences in frequency. In this study, however, I give examples of users' minimal responses to all technical-support providers' inquiries about needs or background information.

First, users gave minimal responses to inquiries about specific needs. In interaction 1, TS2 was helping U11 to duplicate a course. To do so, TS2 wanted to ensure he was inputting the courses' specifications correctly by asking U11 what name she needed in the course title:

TS2: So it will actually be [*semester year*]?

U11: **Yeah.**

In interaction 3, TS4 wanted to ensure that U20 wanted her students' names to appear by last name in a roster table in Moodle:

TS4: Last name only?

U20: **Yeah.**

In interaction 8, TS7 opened the defining stage, presumably right after the recording began with an inquiry about a specific need:

TS7: O.K., so you want to look at the gradebook [*1 second*] and if we can make something extra credit, right?

U20: **Yep.**

These exchanges describe how users employed these minimal responses to clarify their needs; other responses of this type were slightly different. For example, one user's minimal response was non-verbal. In interaction 2, TS2 asked if U11 wanted her students to be able to retake a lesson module test, to

which U11 nodded "yes." Another minimal response was more uncertain. In interaction 2, TS2 met with U11 to help her get a lesson module to respond a certain way to student answers on a test. For this problem, U11 wanted an indication of whether the answer was right or wrong with a green check (as in a Moodle quiz); however, the Moodle system for lesson tests did not indicate this visual information to students. As TS2 reviewed the settings, U11 changed the investigation suddenly and asked about the option for allowing students to select answers until they got the lesson test questions correct. TS2 checked on U11's needs for the test to respond in this way:

TS2: So you want them to keep doing it until they get it right?

U11: **It's just an option.**

Here U11 does not give a certain but rather an uncertain minimal response indicating that she was not sure whether to go with the option of allowing students to select answers until they finally got it correct. After a few moments of thought, however, U11 indicated a concern about this option. To this statement, TS2 inquired for and received a more certain response:

TS2: So do we not want that?

U11: **Yeah, let's not do that.**

Later in the same interaction, U11 again hoped that the questions would give a visual indicator of correct and incorrect answers to show, as TS2 finished her statement:

TS2: Which was wrong and which was right?

U11: **Yeah.**

In these exchanges, users provided minimal responses to inquiries about specific needs, helping the technical-support providers to understand their needs

more specifically. In other exchanges, users gave minimal responses to inquiries about previous actions. In many of these instances, the assumption underlying the inquiry was about an action that the users may have done in setting up their courses. For example, in interaction 3, U2 wanted to make an assignment to be extra credit like another assignment she already put together. To this, TS4 asked:

TS2: Oh, this is extra credit?

U2: **Yeah.**

In this exchange, TS2 inquired about an action that U2 had completed (making the other assignment extra credit). The underlying assumption behind the question appears to be "You made this extra credit?" positioning the question as an inquiry about previous actions. In another example, U2 shared about two students experiencing dissimilar gradebook appearances for a certain assignment. TS6 asked for further information about what the students did:

TS6: And they both turned in the same things?

U2: **Right.**

In a follow-up question about the way U2 had her gradebook set up, TS6 wondered about something she saw:

TS6: So attendance is a separate thing than "daily work"?

U2: **Yeah,** I don't have that added into anything yet.

In these two instances, TS6 wondered about particular actions taken prior to the encounter with the problem. Similarly, in interaction 14, TS8 asked what U2 did when she faced her problem with the missing gradebook items when U2 would hide the weeks from students.

TS8: And you hid it?

U2: **Yeah.**

These exchanges reveal the way that technical-support providers and users exchange information about previous actions. Technical-support providers also asked closed-ended questions about actions others had taken, and users' responses helped to define the problem further.

In other instances, users provided minimal responses to inquiries about experiences with the problem. For example, U20 shared about her experience typing into a textbox when the text would suddenly become larger. TS4 responded quickly:

TS4: Really?

U20: **Yeah.**

After this short exchange, U20 tried to show TS4 but suddenly commented on her inability to change the font after she copied and pasted text into the textbox. Before TS4 could address this concern, U20 shared again how the text would suddenly get larger when she typed:

TS4: Really?

U20: **Yeah.**

In another exchange on the topic, TS4 wanted to confirm what U20 experienced with the editor box:

TS4: So you said this entire editor box gets bigger?

U20: **That's correct.**

Similarly, TS2 wanted to confirm what U14 shared about students' not seeing an assignment on the course website:

TS2: They're not showing up clearly.

U14: **Yeah.**

Likewise, TS8 wanted to confirm that U2's announcement did not go to students' email inboxes.

TS8: It didn't go out at all? At all?

U2: **Yeah.**

Based on these examples, it seems that minimal responses from users appeared after technical-support providers inquired to confirm something that the users already shared about their experiences. Indeed, to phrase these inquiries technical-support providers needed prior knowledge about that experience in order to make the question close-ended and thus amenable to a yes or no response. In a previous version of my coding scheme, I did have a separate inquiry type: inquiring to confirm. This code presented difficulties for my coders because they could not ensure that the information had been shared previously in the conversation. If the coders could not feel certainty that the topic was brought up before, they could not call it an inquiry meant to confirm a previous idea. Despite this challenge in my code development, it makes sense that inquiries to confirm are met with confirmations. These results indicate that technical-support providers were careful to understand users' background information by asking close-ended follow-up questions meant to confirm what users shared about their experiences. In waiting to hear a yes or no answer in response, the technical-support providers used such exchanges to ensure that the two participants would come to a mutual understanding of the problem.

Lastly, users employed minimal responses to inquiries to understand where the problem was located relative to the technology. In interaction 3, U2 wanted to designate one assignment extra credit the way she had done with

another assignment. In the exchange, TS4 wanted to confirm which assignment U2 wanted to make extra credit:

TS4: This one for participation?

U2: **Yeah.**

Similarly, when helping U20 to remove a student from her course website, TS4 scrolled through the roster to find the student that U20 wanted to remove. After U20 gave the name, TS4 asked:

TS4: This one?

U20: **Mm-hmm.**

In interaction 8, U2 wanted to make the attendance module extra credit. When trying to understand where U2 went on the website in her attempt to make this happen, TS7 asked where she went by using a short inquiry:

TS7: "Grades"?

U2: **Yeah.**

In a few instances, the kind of discourse that would go into the identifying stage appeared, specifically when the technical-support providers needed to know again the users' identifying information. After hearing U23's description of her problem, TS2 asked again what course was in question by using U23's name, which helped identify the course in the system:

TS2: [*U23's name*]?

U23: **Yeah, [*U23's name*].**

In a similar instance in the same interaction, TS2 asked again which course was in question:

TS2: Let's see, so it's [*course name*]?

U23: **Yeah.**

TS2 sought identifying information in the defining stage as he looked to define the problem. U23 had already shared the course information; however, TS2 had not navigated there. Furthermore, the purpose of the macrolevel stage was not to exchange identifying information but to define the problem. These inquiries helped, considering this macrolevel purpose, to define the problem.

These examples demonstrate how users employed these minimal responses to inform technical-support providers when users were asked follow-up questions. Confirming or denying did not serve, as when technical-support providers often employed it, to give instruction in response to inquiries about the technology. Instead, users employed it to confirm or deny when technical-support providers wanted to ensure that they understood users' needs or background information. This finding demonstrates again that the defining stage is about the users and technical-support providers trying understand the technical problem. Users' short yes-no responses were a means for them to help technical-support providers to understand.

Inquiring to Understand the Technology

The third-most frequent microlevel discourse that users employed in the defining stage was "inquiring to understand the technology." It makes sense that this inquiry would appear because a misunderstanding about the technology is entirely what prompts technical-support interactions. However, users more often gave information than asked for it during the defining stage. The defining stage requires, as Agar (1985) explains, that the institutional representative ask questions to which the client responds (152). This study confirms that theory by

showing how clients' inquiries were not nearly as frequent in this stage. Still, they did occasionally ask about the technology.

As described in this study, users most often gave background information to describe their problem. Less commonly did they open these interactions by asking a question about how to do something or how the technology works. When they did ask such questions, they did so while the two speakers defined the problem.

In interaction 1, for example, U11 asked several such questions while TS2 helped her to duplicate a course. The entire interaction was about U11's request for this duplication process (involving U11 giving background information and stating needs), but during the defining stage, U11 inserted a few questions about the process of duplicating it. For example, U11 hoped that the site could exist independently of the semester in which it was taught because it was an on-going course module:

> U11: So what about template sites. Or what about like- do all those have to have a- an associated semester?

In another moment in the process as she gave TS2 the specifications of what the duplicate course should entail, U11 wondered about the short designator for the course title:

> U11: I don't know what your short course designator needs to be. What are they usually?

All of these questions related to the same problem, but they appeared at moments as the technical-support provider and the user defined the problem. In another example of this commonality, U19 described his experience with

students' inability to enroll in his course website. While TS7 defined this problem, he asked a few questions to learn more about the technology:

U19: So first you have to sign in and-?

Also, he asked:

U19: And then the enrollment key comes after that?

In one last example, TS2 tried to investigate U5's quiz to understand what she wanted to do with it considering her doubts that she shared about one of the questions on the quiz. As he explored this problem, U5 asked:

U5: So, if I would like to accept two answers, is the best thing going to be just to add the points to their- like override their score?

In these examples, users did not seem to start their technical-support interaction with how-to questions meant to inquire about the technology and how it works. More often, they either described background information related to the problem or they stated a need. During the defining process, they appeared to think of such inquiries as the technical-support providers explored the problem.

A few inquiries to understand the technology did begin the interactions, however. For example, U23 also wanted to learn how to use wikis, and U40 wanted to understand the differences between the electronic rubrics' wording on the course website and the paper rubrics' wording that came with the assignment sheets. These instances, however, were the only two in the data. More often, inquiries to understand technology came during the defining process, not at the beginning of the process and not as the agenda for the stage. Future study with a larger data set could provide more information about how frequently users ask how-to questions to understand the technology as their main agenda item.

These findings reveal, however, that the technical-support interactions were primarily about users' facing an undesirable experience with a technology and wanting to find a way to resolve it. In a few instances, they would ask to learn about something as their sole reason for the interaction, but more often, they wanted to learn about the technology as the technical-support interaction defined (and resolved) the original negative experience the users faced. Also, users more often gave background information through narratives or sequences of events. In this way, this study affirms Steehouder's (2007) assertion that users simply do not have the vocabulary to formulate their problem into a question and must instead share their experience through "historical reports" (3). This finding suggests that technical-support providers must listen for underlying inquiries when users describe experiences because users are less prone to ask well-formulated questions with correct and precise technical vocabulary.

Technical-Support Provider and User Exchanges in the Defining Stage

To further understand the microlevel discourse within the defining stage, I identified the exchange sequences in the defining stage for all of the interactions. Table 5.4 shows the five most frequent exchange sequences and the responding speakers in that discourse.

Table 5.4 Five Most Common Exchange Patterns in the Defining Stage

Exchange Pattern	TS response	U response	Total
Inquiring to understand needs or background information→Confirming or denying	8	42	50
Giving background information→Inquiring to understand needs or background information	26	5	31
Giving background information→Giving background information	4	25	29
Confirming or Denying→Giving background information	0	24	24
Inquiring to understand needs or background information→Giving background information	1	23	24

The most common exchange pattern was "inquiring to understand needs or background information" followed by "confirming or denying," a pattern most often started when technical-support providers inquired and when users provided the responses. I have already described this exchange pattern in the previous sections. Of more interest in this section are the remaining four-most frequent exchange patterns.

The second-most frequent exchange pattern was "giving background information" followed by "inquiring to understand needs or background information," a pattern most often started when users gave information and when technical-support providers inquired about it. To better understand this exchange pattern, I investigated the patterns more closely. In 29 instances, the discourse revealed a pattern in which users provided background information to which technical-support providers inquired further about that information. In the other instances, users provided background information and technical-support providers inquired further about a need (2).

In interaction 3, U2 wanted to make an assignment extra credit but could not find the option to do so. In her explanation, she noted that she had already made one assignment extra credit. To this information, TS4 responded with a question asking for background information:

U2: I already had an extra credit assignment there.

TS4: Oh, this is extra credit?

Naturally, as discussed previously for this example, U2's response was "confirming or denying" ("Yeah").

In interaction 4, U20 shared background information about a student dropping the course, to which TS4 responded with a question seeking further background information:

U20: For instance, I had one kid who enrolled and then dropped in the first week- and then I have one kid who, um-it's listed twice for her. [And so-

TS4: [Mm-hmm.

U2 so now I'm just ignoring them. [But it'd be nice to take them off.

TS4: [so they drop your course. Classes. They dropped it?

As with the previous exchange, TS4 responded to U20's giving background information by inquiring about that background information. As discussed previously, this exchange demonstrates how the technical-support providers carefully confirmed that they understood the context of the problem.

In interaction 5, TS2 likewise ensured that he understood a user's background information. U14 shared her experience with students' inability to see assignment modules:

U11: Yeah, so they can see assignment one and two, but they can't see assignment three. And I've [entered in those-

TS2: [Mm-hmm.

U11: grades and um they were hidden and then I unhid them, but they aren't [showing up.

TS2: [They're not showing up clearly?

As in the previous examples, the natural response to TS2's inquiry was a confirmation ("Yeah"), showcasing how these exchanges reveal that users and

technical-support providers followed a typical pattern (giving background information→inquiring to understand needs or background information→confirming or denying). As described in Table 5.4, 26 instances followed the giving background information and then inquiring to understand needs or background information pattern in which users shared the information and technical-support providers responded with an inquiry. Of these 26 instances, 15 followed the pattern of users providing background information, technical-support providers asking about that background information, and users responding with a confirmation or denial. Of the 26 instances, 11 exchanges involved the user responding to technical-support providers' questions, not with minimal responses, but with additional background information, which explains why one of the fifth most common exchange patterns was "inquiring to understand needs or background information" followed by "giving background information." Table 5.5 describes these common exchanges when users and technical-support providers discuss background information.

Table 5.5 Exchange Patterns When Giving and Inquiring about Background Information

Exchange Pattern	Total
U giving background information→TS inquiring to understand needs or background information→U confirming or denying	15
U giving background information→TS inquiring to understand needs or background information→U giving background information	11
Total	26

This analysis reveals that the exchange patterns in the defining stage often involved the users providing information and the technical-support providers inquiring about that information and the user providing confirmation or further information about the problem. This pattern confirms the very nature of the defining stage as it is a discourse move requiring the users and technical-support

providers to exchange discourse until they share an understanding about the problem that users have faced.

The third-most common exchange pattern involved "giving background information" followed by "giving background information" when the user spoke and then added additional information to what they just shared. In these instances, the users gave a description of a specific topic, experience, or action. They then followed up with additional information on a different but related topic. For example, in interaction 16, U40 shared different topics of background information:

U40: I asked somebody before and then I fooled around with it, and I haven't been able to come up with uh, the easy way to do it.

TS7: Mm-hmm.

U40: So, my students are creating a new page. And they want to add an image. The problem that we've had is that message "too large won't accept" comes up. So, I've been trying to figure out.

TS7: Mm-hmm.

U40: Is [*TS8*]- [*TS8*], is that his name?

TS7: Yeah.

U40: Yeah, he showed me how to do it, but then when I tried to do it, it didn't workout.

In this exchange, U40 follows three topics of background information: her experience seeking help with the problem before, her experience with the problem, and then her experience seeking help with the problem again. I unitized these data into separate units because each utterance has a different purpose, even if ultimately, once I coded the data, they each had the larger

purpose of giving background information. So even though these units received the same code, they were three separate units and thus allowed three back-to-back instances of "giving background information." In another similar instance, U2 told TS8 what part of the schedule her course was on so that TS8 could find the corresponding location on the course website. After stating this background information, she shifted topics to give information about actions she had taken previously when arranging the content on her course website.

> U2: So where the blue turns to white is where we're at now. So I did have everything above the white closed down so they didn't have to scroll through all that.

In these two sentences, the overarching intent is to give background information (just as I coded them); however, the purposes are different, which is why they were unitized into two units. One unit intends to describe the location of the problem. ("So where the blue turns to white is where we're at now.") The other unit describes previous actions taken. ("So I did have everything above the white closed down so they didn't have to scroll through all that.") As a result of these decisions, two instances of "giving background information" appeared back-to-back from the same speaker.

More revealing, however, is that the reason that "giving background information" appears after "background information" so often in the data is that users often shared background information on different topics. This finding implies that technical-support providers should cultivate listening skills, as Callaghan and Thompson (2002) found when they interviewed talent recruiters, so that they can succeed at their jobs. Users did not stay focused in their overview of their experiences, and so technical-support providers should

carefully discern when users shift topics, even in utterances that all similarly intend to share background information.

One of the fifth-most common discourse exchanges in the defining stage is "confirming or denying" followed by "giving background information" when the users provided the response. In these 24 exchanges, 14 of them involved the user expanding (not exactly responding) on their own confirmation or denial by giving additional background information, and 10 of them involved the users responding to the technical-support providers' confirming or denying.

When users expanded on their own confirmations or denials, they went beyond the closed-ended inquiry posed to them. For example, in interaction 20, TS7 posed a close-ended question seeking background information to which U41 responded with the appropriate close-ended response. U41 then expanded on that response:

TS7: Uh, we're working on your portfolio? Is that right?

U41: **Yeah.** Because I got an email- that said that my student profile is ready to be set up.

In interaction 10, TS8 and U23 shared a similar exchange as U23 described her problem with a duplicated quiz module:

TS8: So, all the questions in here?

U23: **Yeah**, so these are questions from the next topic-

TS8: O.K.

U23: All I did was create the new quiz.

Here, TS8 inquired about which part of the quiz U23 was experiencing difficulty. To this inquiry, U23 confirmed ("Yeah") but then added additional background

information beyond the initial confirmation ("so these are questions from the next topic- All I did was create the new quiz.")

In interaction 16, TS7 and U40 followed a similar pattern, but in this exchange U40 denied TS7's action (not her inquiry):

TS7: So the summary is the one that they're working on right now?

U40: **No, no it's my honor's section.** They turned in what's called an "extended definition"

In this case, U40 stopped TS7 from going to the wrong course website, having apparently noticed that TS7 was navigating to a different section. Despite this correction, U40 expanded on her problem by noting what area of that course website she was concerned with (the "Extended definition" assignment module), for which she wanted to better understand the electronic rubric that went with it.

These exchange patterns reveal that users first confirmed or denied technical-support providers' inquiries about background information, but they often moved beyond their initial responses to provide additional information. This finding implies that technical-support providers should be wary of asking close-ended questions to users in the event that users do not provide any additional information beyond the initial response. In these instances described, users readily expanded on their responses, but they were not prompted to do so. Technical-support providers run the risk of miscommunication if they continually use close-ended questions to understand users' experiences.

Of the 24 exchanges that had a sequence of confirming or denying followed by giving background information, 14 involved users providing background information after they themselves confirmed or denied an inquiry from technical-support providers. Of these 24 exchanges, the other 10 exchanges

involved users providing background information in response to technical-support providers' confirmations or denials. The most common exchange pattern of the 10 instances involved U20 and TS4 (4 times), specifically when U20 would ask TS4 if she could see the problem she was facing (inquiring to check comprehension) followed by TS4's confirmation followed by U20 providing additional background information. The following is an example of 1 of the 4 exchanges that followed this pattern:

U20: See how I use color- a whole lot?

TS4: **Mm-hmm.**

U20: But sometimes when I use it from previous times, it goes back to black and white to black. Um, and then I have to redo them.

Of the 10 instances, users would also inquire to understand the technology, receive a confirmation or denial from technical-support providers, and then provide background information in response (3 times). The following is one such example:

U23: If I create items for quizzes, I can always go back and use items from whatever they're called, quiz item bank whatnot and create new quizzes with the same items, right?

TS2: **Yeah.**

U23: I taught a course last spring [*course name*], and I taught some of the topics there that I'd like to use some quiz questions from before. So I don't know what happened last, uh, spring, but one of my quizzes got overridden by another one.

In this example, U23 sought to understand the technology and asked a close-ended question. TS2 provided an appropriate minimal response, and U23 added

additional background information. These exchanges reveal that users asked close-ended questions seeking to understand the technology, and upon receiving their answer, provided background information about the inquiry or about a related problem. These findings imply that technical-support providers have a large listening role in the defining stage as users move from question to narrative in short periods. In these instances, the technical-support providers' short responses provide a minimal but powerful bit of information to which users quickly reacted, showcasing the extent to which technical-support providers must embrace and develop their skills in this listening role they play in given exchange patterns.

Conclusion

Ultimately, this study has revealed that the defining stage plays an important role in the technical-support interaction because it helps the users and technical-support providers to come to a mutual understanding about the users' needs or experiences. These results highlight the most common discourse strategies that both technical-support providers and users employed in this stage. It reveals also how the microlevel discourse displays the users taking the speaking role as they provided information more than seeking it. This study also shows that technical-support providers played a listening role as they asked for information more than giving it. These results affirm Agar's (1985) theory that in this stage institutional representatives seek information while clients give it. This study also confirms Steehouder (2007) when he speaks of users providing "historical reports" rather than clearly articulated questions. In the next chapter, I describe a similar analysis of the resolving stage of technical-support interactions.

CHAPTER 6: MICROLEVEL DISCOURSE STRATEGIES FOR RESOLVING TECHNICAL PROBLEMS

In this chapter, I describe the resolving stage in greater detail by presenting excerpts of both technical-support providers' and users' discourse as they resolve the technical problem at hand. I first describe the discourse strategies that the technical-support providers employed. Then I describe the strategies that the users employed. I end by discussing common exchange patterns between technical-support providers and users, noting how some microlevel discourse strategies come in typical pairs.

Table 6.1 gives the overview of the types of microlevel discourse spoken in the resolving stage and the participants that spoke the discourse.

Table 6.1 Microlevel Discourse in the Resolving Stage

Microlevel Discourse and Speaker Type	TS	U	Total
Signaling	139	12	**151**
Explaining how the technology works or how to do something	141	0	**141**
Confirming or denying	78	37	**115**
Inquiring to understand technology	2	100	**102**
Giving background information	34	63	**97**
Observing	54	17	**71**
Telling	42	15	**57**
Inquiring to understand needs or background information	49	5	**54**
Showing how the technology works or how to do something	49	1	**50**
Stating needs	1	23	**24**
Speculating	18	5	**23**
Planning	8	15	**23**
Declaring the problem or problems as solved	2	15	**17**
Inquiring to check comprehension	4	4	**8**
Judging the technology	4	2	**6**
Inquiring to gain permission	2	3	**5**
Total	**627**	**317**	**944**

The resolving stage contained microlevel discourse relevant to resolving the technical problem. The most frequent discourse in the resolving stage was "signaling" (151 times) followed by "explaining how the technology works or how to do something" (141 times). It makes sense that explaining would occur so often in a macrolevel stage devoted to resolving users' technical problems. Signaling describes how technical-support providers resolved the problems by working through the problem and communicating what they were doing or about to do. Also, technical-support providers spoke both of these discourse strategies more often than users did. Users never once explained how the technology works, and users signaled 12 times compared to the 139 times technical-support providers employed it. "Confirming and denying" was the third-most frequent microlevel discourse type (115 times), and again, technical-support providers provided this discourse more frequently (78 times), suggesting that the technical-support providers would provide "yes-or-no" type responses to the users' questions. Users did employ this strategy as well (37 times), the third-most frequent strategy they employed. The following sections present these results more closely and in greater detail for both speakers.

Technical-Support Provider Discourse Strategies in the Resolving Stage

The technical-support providers, as discussed in chapter 4, spoke more often in this stage than users did, likely because the technical-support providers were the main agents of change in this stage because they had the expertise to resolve the problems, and they spent time giving information while the users spent time seeking it. In the next section, I describe the technical-support providers' discourse strategies.

Explaining How the Technology Works or How to Do Something

The most common microlevel discourse for technical-support providers in this stage was "explaining how the technology works or how to do something" (141 times). As implied in the discourse code's name, the speakers who employ it explained two types of information: how the technology works or how to do something with the technology.

Identifying which type of information the technical-support providers explained presented challenges in my code development. For example, in one instance, TS3 presented the following information for U35:

> TS3 I think that is just to override the whole- to add additional points under, in calculation- we have numerator and denominator twice and then that's extra credit also the numbers without putting the- I mean some numbers under the denominator. When we give extra points or extra assignments, we put numeric- some numbers to the numerator and the denominator as well. Both of them, but I'm thinking in the extra credit it only adds the numerator so-

In this instance, I wondered if TS3 meant to explain how the technology calculated the points (how it works) or if he intended to instruct U35 how to do the calculation (how to do something). To analyze TS3's possible intent, I considered the rules of my unitizing process in which I considered the question: "what is the purpose of [this] particular utterance?" (Rourke, Anderson, Garrison, and Archer 2000, 11). Thinking through the context, I discerned in this instance and in others like it that the speakers may have intended to instruct how the technology works (rather than instruct the user how to do something)

because in this case, for example, TS4 was explaining why the technology makes calculating extra credit difficult for instructors. His explanation of the calculation was an attempt to explain why the technology requires certain specifications to work appropriately. In other cases, the technical-support providers gave information about how the technology works, but the underlying illocution was to instruct the users how to do something. For example, TS8, intervening to help TS7 in interaction 19, described how U41 could grade the ePortfolios:

> TS8: Just go through it, like think-aloud protocol. What they do in usability testing where you just start, um, go through the site for the first time and say "O.K. I'm going to walk through your site" and comment on navigation and design. See if you've got all your content. Basically just like talking to yourself through the site.

In relation to the turn excerpted here, I questioned whether TS8 meant to explain how think-aloud protocol works or how to grade the portfolios. Considering the context, TS8 likely meant to instruct U41 in a technique for grading the portfolios. These instances suggest that future studies on technical-support interactions carefully distinguish not only types of utterances but also the content (or "aboutness") of those utterances. Mackiewicz (2017) demonstrates in her study that a corpus-driven linguistic analysis can reveal the manifest content (observable, surface content) of one-to-one interactions in ways that complement an analysis of latent content (interpretive, subjective content) such as that employed in discourse analysis (5), the approach used in this study. In her study, she demonstrates how such a mixed-methods approach yields greater insight into what speakers in writing center conferences actually discuss; as the corpus-driven linguistics analysis reveals what the conversation is actually about, the

discourse analysis helps to "identify how speakers co-construct their interaction on a moment-to-moment basis" (3). Future studies could foreground what "how something works" and "how to" instructional discourse is about, that is, what content makes instructional discourse in technical-support interactions unique from other kinds of instructional interactions. This study, however, did not set to describe the interactions in that way. Nevertheless, acknowledging the limitations of classifying the discourse strategies into these two categories, I describe them using this categorization so that I can give a richer description of the strategies and also help technical-support providers and others to see how they function in an exchange.

Table 6.2 describes the types of instructional language that I identified technical-support providers using.

Table 6.2 Explaining How Something Works or How to Do Something

Explanation Types	Examples
Explaining how to do something	I would recommend the bottom one. [*6 seconds; scrolling through rubrics*] Because that's the most recent one that we created- The way the system works is, yeah, the most recent one we created is going to be the one on the bottom. O.K., and then you'll just adjust the size down to, uh, whatever will fit onto our thing. So we have 300 KB I believe- so it shows you how much your- it shows you the width and the height and how much, um, um, how big the file size of- the file size that is. So usually what I do is I just play around with the numbers up there until- it goes down. They can drag and drop from the desktop or they can hit 'Select Files' and upload it.
Explaining how something works	Quizzes allow you a little more flexibility that way. It's probably because it's attendance and not an assignment- that makes it extra credit. I think that they assume that if you're putting a grade in, in terms of attendance that [it's just part of the grade. [T]he gradebook should be- once the gradebook's set up, then no matter if you grade all the papers at once or in separate groups, it's all going to go to the gradebook separate.

Technical-support providers more often explained how to do something with the technology (74 times) than they did how the technology works (67), though the two types appeared relatively equally. Some explanations were shorter (the shortest was 2 words) and some explanations were longer (the longest was 340 words). Most explanations were more moderate in length (averaging 32 words). In interaction 4, TS4 described for U20 how to implement text on the course website through what Moodle calls labels: "So we can't actually put the uh- what is it- the bullet points but instead I do as a instructor was that I added label." Likewise, in interaction 6, TS7 explained how to fix an issue related to U2's students' inability to submit assignments:

> TS7: So it looks like you may need to go into all of your dropboxes and do "Save and Display" rather than- Because I usually do "Save and Return to Course." I don't know why that's not working- because they should both work.

The longest explanation had TS7 explaining to U41 how to grade ePortfolio's using screen-casting software (340 words). The shortest explanation was TS4's short response to a question from U20 about where to go on the website ("I go down to label?"). To this question, TS4 confirmed ("Yes. Yeah.") and added with the short explanation, "And 'add,'" indicating the next step in the sequence. Here the instruction was less of a directive ("[click] 'add.'") and more of a process description of where U20 would go next if she wanted to add her label to the website.

Technical-support providers also described how the technology works. The shortest description was 4 words and the longest was 95 words, indicating

that technical-support providers spent fewer words to show how the technology works than they did when they explained how to do something with it. Likely, the technical-support providers main function in this stage was not to showcase their knowledge about how the technology works but rather how the users can use the technology to solve their problems. In interaction 15, TS2 explained briefly that ePortfolios have a 5MB limit for any media users want to upload. "That's for all media," he explained after U40 asked if that limitation applied to all media types. In another instance, TS2 explained the privacy settings for the ePortfolio, depending how U32 wanted to use it:

> TS2: Just in case you would like that. Because eProfiles- the whole idea there is you know- create a site where you'll market yourself. So you're less concerned with security as we've set it up because the idea is to make it public, available to everyone.

In another, longer explanation, TS7 explained how students usually miss the correct course when they are searching for it:

> TS7: And then the other thing that's happening is some students will go to- So they'll search for like [*course number*] [*section ID*]. [*2 seconds; entering text into search bar*] Or they won't even get that far. They'll only search [*course number*], and let's say they're in section [*section ID*]. They'll, um, like, or [*section ID*]. Let's say they're in section [*section ID*]. They'll scan it real fast [*hovering cursor over search results*], and they don't realize that there's different sections under one course so they'll skip over this one- and they're just looking for [*section ID*] instead of the combined courses.

These examples demonstrate how technical-support providers explained to users how something works or how to do something. They explained how to do something with the technology only 7 times more than they explained how something works in the technology. A larger data set might reveal which type of information technical-support providers explain more. Nevertheless, clearly in the resolving stage technical-support providers employ their agency by helping users through their explanations. That this was the most frequent discourse type gives researchers little surprise given the nature of resolving a problem in a technical-support interaction, but it does confirm the assumptions of Agar (1985) and Steehouder (2007) when they claim that this moment within the interaction entails instructional discourse. My study does challenge Steehouder's (2007, 3) findings wherein he describes technical-support providers giving users step-by-step sets of directives (imperative mood discourse with various degrees of directness). My findings show that "telling," as I call it, happens relatively fewer times (42 times; or 6.7% of all of the strategies the technical-support providers employed). These findings suggest that instructional discourse in such technical-support interactions works differently than it does in technical documentation, wherein a series of sequential, numbered, imperative mood steps assist users in completing their tasks. Instead, these one-to-one interactions more often entailed descriptive discourse (using indicative mood and with declarative syntax). With this discourse strategy, technical-support providers served less as "coaches," as Steehouder calls them (3), and more as lecturers, who describe and define the functions of the technology so that users understand it better.

To showcase this observation further, I investigated all 42 instances of technical-support providers employing "telling" or what the technical

communication discipline typically characterizes as typical and expected discourse for instructions or documentation. Interestingly, this microlevel discourse happened in the resolving stage of only four interactions (4, 16, 19, and 20). TS7 used it 22 times, TS4 used it 19 times, and TS2 used it once. TS7 used it 19 times in interaction 16 and three times in interaction 19. In all instances, TS7 employed the strategy in the manner typical of documentation discourse:

- "Exit out of that real fast."
- "And then up where it says 'Preview.'"
- "Slide over to 'File.'"
- "Um, 'Open.'"
- "And then 'Photos library.'"
- "Slide down a little bit."
- "So if you pick that photo and 'Open.'"
- "And then, um, when you're in here you'll go to 'Tools.'"
- "And then down to 'Adjust size.'"

TS7 employed these directives as U40 requested to know how to modify an image's size in order to ensure she could upload it to her ePortfolio, given the system's size limitations. Similarly, in interaction 19, U41 wanted to know how to change the default title of the ePortfolio, and TS7 gave her the steps necessary to do so as U41 followed along on her computer:

> TS7: So that's the, um, that's just the, like, gen- standard title that you get. **basically if you go to the customize menu up here.** Um, **for that you will go to- down to "site identity." All the way down [*clicking "site identity"*].** And then- you can just- like change it to,

oops, whatever. You can delete the title line or change it to something else, or whatever you need.

U41: I kind of want to put like some info about myself.

TS7: **Hit "Save and Publish," so it saves it.**

These few instance of telling may reveal two features about the resolving stage. Users rarely asked for step-by-step walkthroughs of a technical process so that they could follow along and do it along with the technical-support providers. Steehouder (2007) noted from his study that when technical-support providers began to give step-by-step walkthroughs "[technical-support providers] assume[d] that their clients [were] sitting in front of the computer, that the computer [was] on, and that the program [was] active" (3), because they began their instructional discourse without checking to see if users were ready for it. About his findings, Steehouder notes that (due to the helpline context) agents at times realized too late (after five minutes in one case) that users were not ready for the information because they could not see if users were using computers during the call. Users in these instances may not have called (or in my study visited) technical-support service with the readiness to follow steps. Instead, they seemed ready to get advice or receive solutions they could implement on their own time or that the technical-support providers could do for them right then in the interaction. These findings reveal that users may be prone to seek advice or solutions rather than step-by-step instructional discourse. Even in cases in which they received information about how to do so something, the context indicated that both users and technical-support providers envision not only resolving the immediate problem but also that the users would take away knowledge with them about how to use the technology. Technical-support interactions it seems

differ in this way from documentation wherein users may follow steps to complete tasks. More research with a larger data set and in another context could reveal how frequently users seek step-by-step, real-time instruction rather than descriptive explanations as they did in this study.

Signaling

The second-most frequent microlevel discourse that technical-support providers employed in the resolving stage was "signaling." Because this microlevel discourse is relatively obscure in isolation, I required contextual information when coding it, including the discourse that surrounded it. I investigated the 139 instances in which the technical-support providers employed signaling. I found that technical-support providers employed signaling in two types: think-aloud signals and announcing signals. The most frequent signal type was think-aloud signals (81 times). Just as in the defining stage, this discourse type often involved the technical-support providers reading buttons or text on the screen as they used the technology. The following are examples of when the technical-support providers employed this strategy:

- "Continue"
- So "Continue"
- "Create"
- "User report"
- "Save and Display"
- "Five p.m."
- "[*course semester*]"
- "Administration"

In other instances, technical-support providers thought aloud as they encountered challenges with the technology. For example, as TS7 helped U40 to modify an image's size so that the image could be uploaded to the ePortfolio, she thought aloud about a challenge she faced when attempting to demonstrate how to do it. As she attempted to upload the adjusted image, she encountered an error messages from the ePortfolio system demonstrating that the image was still too large to upload. In response, TS7 thought-aloud about the file she must have attempted to upload:

TS7: Did I pick the right one? I probably didn't pick the right one. [*Mumbling*] [*13 seconds; resaving image to another file name and location*]

Whether reading buttons or text on the screen or thinking aloud as they faced challenges, technical-support providers spoke aloud their thought process in order to account for the wait time as they resolved the problem, specifically when the wait time came from small challenges they faced with the technology. TS7 specifically mentioned in a post-session stimulated recall interview that she tried to think aloud during these moments in order to account for the wait time ("to make it less awkward") and to make sure the user knew what was happening during the wait time. Just as discussed previously in chapter 5, technical-support providers employed this strategy to break the silence, to help themselves think, and to potentially teach the users how to use the website should the users be paying attention at those moments in the interaction.

Technical-support providers also signaled by giving announcements about what they did or were about to do. Often this announcement was a "let's" phrase such as "Let's see what [*U name*]'s course is like," or "let's double-check."

In other instances, the technical-support providers would announce what they were about to do, as when TS7 announced, "All right, I'm going to go do the exact same thing and log in as [*student name*] again." In another example, TS6 announced what she would do to check on the team's discussion of a problem that U2 experienced with announcements: "I can just double check my email." These examples reveal that technical-support providers carefully coordinated their resolving efforts with discourse that signaled that they were thinking about the users' presence and the users' needs to stay aware about what was happening as the technical-providers resolved their problems. As discussed in chapter 5 when technical-support providers employed signaling in the defining stage, they showcased how they managed multiple frames of discourse, including social and instructional frames, by carefully considering their audiences' presence and the best way to instruct them.

Confirming or Denying

The third-most frequent microlevel discourse that technical-support providers employed in the defining stage was confirming or denying (78 times), a minimal response that serves as substantive and viable participation in the interaction as users employed questions or statements. Because this microlevel discourse appeared relatively obscure when seen alone ("Yeah," "Yup," "Mm-hmm."), I required contextual information to better identify it, including the discourse that surrounded it and the discourse to which it responded. The most frequent discourse to which technical-support providers responded with confirming or denying was inquiring to understand technology (55 times), each time in response to the user. Of the 23 remaining instances of confirming or denying, 6

were in response to users' inquiries to understand needs or background information, 4 were in response to users giving background information, and 3 were in response to users stating needs. Overwhelmingly then, technical-support providers used this minimal response when users inquired to understand the technology. This section will provide examples of those instances.

In interaction 4, U20 wanted to rearrange the gradebook table so that it displayed the records by last name. After TS4 told her to click on "last name" in the column header, U20 asked to ensure she understood how to use the table:

U20: O.K. So I just click on "last name"?

TS4: **Yes.**

In a similar instance in which U20 required a confirmation for an instruction she just received, she wanted to confirm where she could find the option to add a label to her website:

U20: Is it, um, under "resources"?

TS4: **Yeah, there you go.**

This pattern of users confirming instruction they just received continued throughout the interactions. In interaction 7, U19 wanted to confirm how the procedure works for students when they enroll in a course website:

U19: So if they've got to the enrollment key phase, then they've successfully entered their username and password?

TS7: **Right. Mm-hmm.**

In interaction 15, TS3 confirmed the question from U2 about leaving feedback files for students after TS3 helped her to find the way to do so:

U2: And then I could leave my feedback files?

TS3: **Right.**

At some points the confirmations or denials were not in response to confirmation questions. At one point U2 simply wanted to confirm what it was that was on the screen:

U2: Is that a label?

TS8: **That is not a label.**

One instance found U41 asking for instruction about the electronic rubrics in Moodle:

U41: Isn't there one that's a little bit different for the visual analysis, or not?

TS2: **No.**

Ultimately these findings reveal that, just as with the defining stage, technical-support providers give substantive instruction through minimal responses. Interestingly, at no point were the confirmations or denials uncertain or noncommittal; that is, they were clear and certain variations of "yes" or "no." This observation makes sense because by its nature resolving indicates certainty and resolution instead of the unstable uncertainty that may characterize the process of defining problems. Here the technical-support providers resolved the technical problems, and the minimal responses they provided illustrate this certainty and resolution.

User Discourse Strategies in the Resolving Stage

The users spoke less than technical-support providers in the resolving stage. The most common microlevel discourse they employed was "inquiring to understand the technology" (100 times). This finding corresponds to the most common microlevel discourse technical-support providers employed ("explaining how

the technology works or how to do something"). The second-most common microlevel discourse users employed was "giving background" (63 times). Lastly, the third-most common microlevel discourse was the short, minimal response "confirming or denying" (37 times). In this section, I describe how users employed these strategies in more detail.

Inquiring to Understand Technology

The most frequent microlevel discourse that users employed in the resolving stage was "inquiring to understand technology." As described in chapter 5, users rarely entered technical-support interactions with the intent of "inquiring to understand technology" but rather with a narrative of events ("giving background information"). I noted instead that users inquired about the technology only as or after the problem was resolved, showcasing that users did not come with pre-prepared inquiries; rather they came with prepared descriptions of their experiences. That users inquired about the technology so many times in the resolving stage corresponds to this finding from the defining stage. Users seemingly thought of questions about the technology as or after the technical-support providers resolved their problems.

One example of users thinking of questions as technical-support providers resolved the problems was when users needed to confirm the instruction they received. For example, in interaction 4, U20 followed up after instruction she received from TS4 about how to reorganize her gradebook table by last name.

U20: O.K. so I just need to click on "last name"?

Later, she followed-up in a similar way:

U20: Is it, um, under "resources"?

In this interaction, the longest interaction (31:02), U20 asked several such follow-up questions after TS4 gave her instruction (17 additional times). This discursive pattern was not unique to U20, however.

In interaction 6, U19 wanted to confirm a conclusion he deduced based on TS7's instruction about what students should be able to do as they attempted to enroll in U19's course website. He asked:

U19: So if they've got to the enrollment key phase, then they've successfully entered their username and password?

Similarly, U19 deduced another idea based on TS7's instruction about students enrolling in the website:

U19: So once they put in their username and password- then they have to search for [*course name*]?

The illative coordinating conjunction "so" indeed signals an inference on U19's part, showcasing how he reasoned from the instruction he was receiving. For him, a follow-up question helped to resolve his understanding.

In another example (interaction 13), U40 again deduced from the instructions she received from TS2 about which electronic rubric would work better for a given assignment:

U40: So for the rhetorical analysis I would just use "general"?

Again, the illative coordinating conjunction "so" shows the discursive movement toward resolution and certainty. In one last example of users' reasoning, U2 received instruction about how to reformat her course website so that she could make it easier for students to navigate. Upon receiving the location of the settings options, she asked:

TS2: So it's under "course format"?

Again, the illative coordinating conjunction "so" reveals her inferences from the instruction.

In interaction 12, U5 asked a similar question in response to TS2's explanation that Moodle quizzes do not allow users to retroactively award points for questions. TS2 speculated:

TS2: Because you might be able to just assign extra credit. But honestly overriding grades might be easier.

Completing TS2's explanation, she inquired:

U5: Just to add- just to add like the point onto their grade?

TS2's confirmation ("Mm-hmm") reveals that U5 was confirming the instruction she received. Here, however, the conversation reveals U5 simply wanting to confirm she was understanding TS2.

In a similar question, U2 responded to her instruction from TS3 about where to find the option to leave feedback files. After TS3 explained, she followed-up:

U2: And then I could leave my feedback files?

TS3's confirmation ("Right.") reveals that U5 was confirming the instruction she received from TS3. Here again I provide that exchange:

TS3: You can click this one [*clicking on grade icon*]. And scroll, and then- [*"feedback files" area appearing*]. Could you do that? Right. Mm-hmm.

U2: And then I could leave my feedback files?

TS3: Right.

These examples reveal how users responded to instruction either with questions highlighting their reasoning or through questions highlighting their need to confirm.

Some instances of new questions with new content did appear, however. In these cases, the questions were not meant to confirm but rather to learn about something that users had just considered as the technical-support providers worked to resolve the problems. In interaction 13, as TS2 helped U40 to understand the electronic rubrics, U40 asked a sudden question about the rubrics:

U40: What's the difference- What's the difference between those two?

In another instance, U40 intervened during an explanation with a question about the rubrics:

U40: Now for visual analysis do I use that same one?

After receiving instruction for how to ensure that she could see her sections one by one when grading the assignments, U40 asked about the procedure she was just taught:

U40: And then you changed it for this one assignment- will it continue to be like that for my assignments, or do I have to go in there for every assignment on that "group" thing?

In another instance (interaction 19), as she was receiving instruction about using ePortfolios in her class, U41 had a question about grading them and how students would interact with her commentary:

U41: How do they see the comments? I guess, because I know at the end of the semester students usually like pick up their portfolio. I'm just trying to like mentally figure out how this is going to work.

In these instances, the focus was less on drawing inferences from the instruction or confirming knowledge received during instruction and more about learning something that the users had just then considered. Notably, these questions from

U40 and U41 occurred when the users had an agenda in which they wanted to learn how to do something (i.e., the agenda for that given problem was a how-to issue). Specifically, in interaction 13, U40 wanted to learn how to use the electronic rubrics, and in interaction 19, U41 wanted to learn how to deploy the ePortfolios in her classroom. It makes sense that in cases where users set the agenda about learning about the technology, new questions of this sort materialize. In interactions in which users shared experiences with the technology that they wanted to fix, such new questions with new content did not appear. Instead, in those instances, users asked follow-up questions that showcased their inferences or their need to confirm something they just learned.

These examples reveal that the resolving stage in technical-support interactions is less about technical-support providers giving step-by-step procedures but instead about technical-support providers giving responsive explanations to users as users think of new questions. These findings imply that technical-support providers should listen carefully to users and even invite questions from users so that users can confirm their knowledge or ask new questions if they think of any.

Giving Background Information

The second-most frequent microlevel discourse that users employed in the resolving stage was "giving background information" (63 times). As with "inquiring to understand technology," it would seem that this code would appear only in the defining stage because the defining stage is where the user describes their problem. However, as technical-support providers resolved the

problem, users offered additional insight into their experiences, insight that might have enabled technical-support providers to resolve the problems.

As discussed in chapter 5, giving background information broke down into three major categories: describing where the problem is located, describing experiences, or describing previous actions. In the resolving stages these types appeared again. The most frequent type was describing previous actions (37 times), and the second most frequent type was describing previous experiences (19 times). Describing locations only appeared 4 times, and an additional category I discerned from the data was giving teaching context, which appeared 3 times. As an example of describing a location, in interaction 6, U2 shared 2 times a student's name to help TS7 find the location of the problem. And in interaction 20, U41 shared 2 times when a certain assignment would be due to help U2 find the assignment on the course website: "It's due on that next week." These instances of sharing a location were infrequent, implying that the resolving stage was late enough in the interaction that technical-support providers and users did not find it as pertinent to find problems' locations because likely they understood the location of the problem back in the defining stage.

When users shared teaching context (3 times), they seemed to provide seemingly unnecessary information that did not seem to help resolve the problem directly. In interaction 6, U2 shared twice that a certain feature would be used in class that day perhaps to emphasize the urgency of getting the problem resolved: "We're going to use the drop boxes again in class today." In interaction 19, U41 stated that she thought that students would find an overview of the ePortfolio confusing: "And they're going to be like 'what, what's going on' because they'll probably be focusing on their documented essay."

Overall, then, users communicated background information more often to describe previous actions (37 times) and to describe previous experiences (19 times), demonstrating that the main purpose for users giving background information in the resolving stage was to help technical-support providers resolve the problems more quickly.

When users shared background information about previous actions, they emphasized the problem. In interaction 7, U2 shared additional actions that her students took when attempting to upload files, despite the module not allowing them to do so:

> U2: Another thing that one student figured out that they were able to add submissions if they changed it to PDF, but not if they tried to upload as a Word document. Even though the other students in the classroom were able to upload Word documents at the same time. They were all uploading at the same time. So it's a little buggy.

As described previously, U2 shared twice that the student would use the same modules that day to emphasize the urgency she felt in getting the problem resolved. Here the additional comment "So it's a little buggy" showcases her frustration and determination to get the problem resolved. Her additional information about what students attempted to do demonstrates her intention to show that she tried to think through various solutions and that she ultimately required TS7's efforts.

In interaction 13, while receiving instruction about how to use rubrics, U40 shared the additional insight about what she had done to grade the assignment previously:

> U40: I do, what I have been doing is, um- I'm not sure if this is what you're referring to, but, uh- where you can do review and then comment. I've been doing that in the paper and this.

In interaction 19, while receiving instruction about how to use the ePortfolios, U41 shared the additional insight about how she usually grades the assignment:

> U41: Usually when I used to grade a portfolio, I go in, and I don't like go into depth with my comments. And I only do go into depth when they request the portfolio back. Because I don't want to like make so many comments when they're not going to see it.

Again in interaction 20, U40 shared her propensity when grading with rubrics as TS7 shared how they worked:

> U40: I almost never give people a one hundred on their papers anyway. If it's an A, I give a 95. If it's an A plus, I give a 98 or something like that.

These instances reveal that when giving background information about previous actions, users more often share additional insight about what they have tried or like to do with the technology. The nature of this discourse demonstrates perhaps that in the resolving stage additional background information is less pertinent to the conversation because the technical-support providers have already entered the instructional phase of the interaction and thus do not require as much background information about what the users or their students had done with the technology. This finding implies that technical-support providers

should give their instructions carefully and patiently, ready to hear from users about their actions or the actions of others prior to the problem, even if that information does not necessarily help to resolve the problem.

In other instances, users shared background information related to their experiences with the technology. As when they shared background information related to previous actions, the users' discourse when sharing background information related to previous experiences that did not necessarily relate to or help with resolving the problems. For example, in interaction 6, U2 shared an additional insight about her students' experiences with the problem:

> U2: The other thing is that for some of the students they've never had an error.

Similarly, U19 shared in interaction 7 his students' experience with the problem:

> U19: Most students haven't had any problems, because it's almost like 70 students total, and it's been maybe 4 who had problems.

Later, U19 shared a similar idea:

> U19: No, it's not like sporadic. Most students aren't having problems.

In these instances, U2 and U19 appeared to share the scope or severity of the problems by noting how infrequently and how few users actually experienced the problem.

In other cases, users appeared to share that they were surprised by how quickly the technical-support providers solved the problems, and they did so by giving background information about their experiences.

In interaction 11, U2 shared her surprise at how 3 people (her and her 2 students) could not solve the problem, even though TS6 just did:

U2: That's so weird because they were sitting side by side, and they still couldn't figure it out [*laughing*].

In a similar instance, U41 insisted that the problem appeared worse or more difficult in her experience than what it might seem to U7 and U8 after they resolved the problem.

U40: Oh, it wasn't like that before.

And then a few moments later:

U40: This wasn't up there before.

And then a few moments later:

U40: But just a minute ago it wasn't in there, so I just want to make sure.

In these examples from U2 and U40, users shared previous experiences to showcase their surprise at how easily the technical-support providers resolved their problems.

In one last sub-category of users sharing background information related to previous experiences, U2 appeared to share her experience with the technical-support team:

U2: Well, a lot of my questions they don't know how to fix.

And then moments later, when asking about a recurrent issue with Moodle announcements not going to students' email inboxes, she shared:

U2: Because I had heard that it was fixed, and then it wasn't fixed, because they thought it was fixed and then went back and forth.

These instances, whether to share previous actions or previous experiences, demonstrate users sharing information at a late stage in the interaction as or after

technical-support providers resolve their technical problems. The examples reveal the peripheral nature of the topic users bring up relative to the problem at hand, but they also reveal that users appeared ready to share their opinions about the technology through the background information, even when the technical-support providers resolved their problems. These instances imply that as with the background information that users share in the defining stage, technical-support providers should carefully focus on strategies for keeping their discourse focused on resolving the technical problems and less on the additional information that users may share during the resolving stage. At times, it seems, users shared additional information in this stage out of exasperation or to save face when the problem appeared easy for the technical-support providers to solve and perhaps they should have been able to solve themselves.

Confirming or Denying

The third-most frequent microlevel discourse that users employed in the resolving stage was confirming or denying (37 times). Because this microlevel discourse is a short "yes-or-no" type response, a minimal response, it required me to investigate the contextual information, including the discourse that surrounded it and the discourse to which it responded. I investigated the 37 instances in which the users employed "confirming or denying" and examined the discourse to which the users responded. I found that users responded with confirming or denying most frequently to "inquiring to understand needs or background information" (23 times). Another 11 instances were in response to "inquiring to check comprehension" (3 times), "confirming or denying" (2 times), "observing" (2 times), "showing how the technology works or how to do

something" (2 times), and "explaining how the technology works or how to do something" (2 times). The additional 3 cases of this discourse were isolated responses to "inquiring to understand technology" (1 time), "inquiring to gain permission" (1 time), and "stating needs" (1 time). Clearly, the main purpose users employed "confirming or denying" was in response to technical-support providers' inquiries about needs or background information.

In the three cases in which users responded with confirmations or denials to inquiries to check for comprehension, the exchange follows a typical pattern to yes-no questions. In interaction 13, TS2 recommended which rubric would help U40 with her needs:

TS2: For the rhetorical analysis and visual analysis, I just use the last option for the general- The third to last. Yeah. The last of the general ones [*laughing*]. There's a presentation one and there's an e-portfolio one. It mostly makes sense?

U40: **Yes, it does.**

In interaction 18, TS2 worked with U32 to help her manage her ePortfolio sites. He explained how password-protection works:

TS2: So that's one way to make it, you know, inaccessible to everyone- that you don't want seeing it. Say you wanted all your [*course number*] materials private, like you didn't want anyone seeing it. You could set each of your pages to password-protected. And then you could share that password with [*name*] or whoever- is going to be reading it. Does that make sense?

U32: **Yes. That makes a lot of sense.**

In the same interaction, TS2 explained how menus work with the ePortfolio website and wanted to check if U32 understood his terminology, perhaps implying that he would explain further if she needed him to:

TS2: Just one thing you might try doing is maybe create like a "[*site name*]" tab and then "[*course number*]" tab- or something and then you could create sub menu items. Do you know how to create sub menus?

U32: **Yes.**

These exchanges reveal one technical-support provider (TS2) carefully considering the needs of users by checking for comprehension to ensure that users understood the instruction. More often than these instances, however, users gave minimal responses to technical-support providers' inquiries to understand needs or background information.

As discussed in chapter 5, when inquiring about or stating needs, users and technical-support providers talked about two types of needs: general needs or specifications. In the resolving stage, users responded with confirming or denying to specifications, never for general needs (e.g., "How can I help you?") because general needs questions were not closed-ended. In interaction 1, U11 confirmed her needs for a course that she wanted duplicated:

TS2: So I imagine [*course name*] would be the name of the course?

U11: **Uh-huh.**

In the same interaction, U11 confirmed another need as TS2 made the duplication:

TS2: Call it [*year number*] I guess?

U11: **O.K. Yeah.**

In interaction 4, TS4 showed what Moodle labels looked like by showing U20 what labels looked like on TS4's website. She asked about U20's needs:

TS4: Do you want this type of thing, [right?

U20: **[Yeah!**

In other cases, users responded with confirming or denying to technical-support providers' inquiries about background information. In some cases, they responded to inquiries about location. In interaction 4, TS4 wanted to confirm that she had deleted the appropriate student profile by referring to the user they wanted to keep, the one that had recently logged-in (not 90 days ago but) 20 minutes ago:

TS4: Just- twenty minutes ago?

U20: **Yep, yep. That's the good one. O.K.**

Similarly, in interaction 7, TS7 wanted to ensure she was searching for the correct website so that she could determine why U19's students couldn't access the website to enroll in it:

TS7: Is it- what did you say- [*course name*]?

U19: **Yeah.**

In other cases, users responded with confirming or denying by responding to inquiries about previous actions that users took. In interaction 4, TS4 asked how she had proceeded in the past with delivering files to students for an assignment:

TS4: So basically you're uploading different, uh, files, right?

U20: **Uh-huh.**

In interaction 14, TS8 wanted to confirm how U2 had her course website's appearance set-up, having seemingly forgotten as he spent time investigating the different formatting options:

TS8: Because yours was set to "weekly," right?

U2: **Yeah.**

In other cases, users responded with confirming or denying by responding to inquiries about previous experiences that users or their students had experienced. In interaction 7, TS7 wanted to understand the scope of the problem with students not being able to get into U19's website:

TS7: It's only a few students who aren't seeing it?

U19: **Yeah.**

In interaction 11, U2 described how two students appeared to see two different appearances in the gradebook with different numbers for the percentage weight for the assignment, which raised concerns about the grading:

TS6: She had a fifty here and a thirty-three here?

U2: Yeah.

These examples reveal users responding to technical-support providers' questions about their needs or their previous actions or experiences. Even at a late stage in the interaction, this information helped the technical-support providers to resolve the problems. These results do challenge again the findings from Steehouder (2007) who seems to suggest that the "instructional phrase" (3), as he calls it, is primarily about the technical-support providers giving step-by-step instructional discourse. As I discussed previously, technical-support providers gave explanations, and they also asked questions. These findings also challenge Agar's (1985) theory that the "directives" stage is only about giving

suggestions and advice. While my findings certainly affirm that this happens in this stage, my findings also show technical-support providers continuing to ask questions as they worked to ensure users had their problems resolved. These findings imply that technical-support providers, even when they are relatively close to solving the problem, still need to ensure that they understand fully the experiences and needs of their users. Furthermore, checking for comprehension after instructions and descriptions can potentially keep users engaged in the instruction and ensure that users are understanding the instructions.

Technical-Support Provider and User Exchanges in the Resolving Stage

To further understand the microlevel discourse within the resolving stage, I identified the exchange sequences in the resolving stage for all of the interactions. Table 6.3 shows the five most frequent exchange sequences and the responding speakers in that discourse.

Table 6.3. Five Most Common Exchange Patterns in the Resolving Stage

Exchange Pattern	TS Response	U Response	Total
Signaling→Signaling	55	4	59
Inquiring to understand the technology→Confirming or denying	55	1	56
Confirming or Denying→ Explaining how something works or how to do something	30	0	30
Inquiring needs or background information→Confirming or denying	6	23	29
Signaling→Observing	23	5	28

The most common exchange pattern was signaling followed by signaling, a pattern started when technical-support providers signaled twice in sequence (54 times) and once in response to a user signaling. Users also signaled twice in sequence (3 times) and once in response to a technical-support provider signaling. These exchanges make sense considering the nature of the resolving

stage. Technical-support providers most often signaled in this stage as they resolved the technical problem, and these results indicate that they signaled many times in sequence. For example, in interaction 4, TS4 signaled twice as she was about to un-enroll a student from the course:

TS4: Let's double check. So let me un-enroll.

As she attempted to see why a student could not submit an assignment, she signaled three times in a row as she selected buttons on the screen:

TS4: "Create." "User report." Let's see.

This pattern was not solely TS4's, however; TS7 signaled four times in a row as she attempted to understand why U2's students couldn't submit their assignments. She noted when she selected buttons and then announced twice:

TS4: "Save and Display." "Five p.m." Alright I'm going to go do the exact same thing and log in as [*student name*] again. Let's see what we can see.

Similarly, TS6 announced what she was about to do and then thought aloud a button she pressed (a student's name):

TS6: So we need to log out of [*student name*]. [*5 seconds; logging-out of student name's view*] And it logs us all the way out. It's [*student name*].

Also, TS8 thought aloud a sequence of buttons he pressed as he worked on U2's course layout problem:

TS8: "Administration." "Edit settings."

Later, he did the same sequence as he worked through the same problem:

TS8: "Weekly format." "Collapse topics."

TS3 signaled four times in a row, giving a think-aloud signal, two announcement signals, and one more think aloud signal:

> TS3: "No Credits." This is going to be zero points. Going to put this one-under "No Credits." "Save Changes."

Lastly, TS2 signaled four times in a row, with two think-aloud signals and two announcement signals:

> TS2: "Introduction." "ePortfolio." So I want this one [*Selecting rubric option*]. Let me double-check it really fast [*reading softly to self*].

Clearly then, technical-support providers, as they worked to resolve technical problems employed signaling. As I discussed, signaling was their most frequent discourse strategy, and as this discussion shows, they employed this strategy in sequences. This finding makes sense because technical-support providers would likely employ such sequences as they worked through operations on a website. The combination of announcement signals and think-aloud signals reveals also that technical-support providers dynamically employed this strategy to ensure that users are aware of what is happening in an interaction.

The second-most frequent exchange pattern was "inquiring to understand the technology" followed by "confirming or denying," a pattern most often started when technical-support providers gave a minimal response after users inquired about the technology (55 times). I have already shown how this exchange pattern worked when I discussed how technical-support providers employed the minimal response when users asked them about the technology, and such minimal responses served as a means of instruction.

The third-most frequent exchange pattern was "confirming or denying" followed by "explaining how something works or how to do something" (30 times). In 17 instances, the discourse revealed a pattern in which technical-support providers responded to users with this sequence. A user would ask a question, the technical-support provider would confirm or deny, and then the technical-support provider would follow-up with an explanation. For example, an exchange in interaction 2 followed this pattern (U inquiring about technology→TS confirming or denying→TS explaining how something works or how to do something). The technical-support provider's confirmation or denial appears in bold in the excerpts that follow:

U11: And then it is that something that's a feature in quizzes or are quizzes structured just like this?

TS2: **No, quizzes are different**. Quizzes allow you a little more flexibility that way.

A moment in interaction 4 followed this same pattern:

U20: I go down to label?

TS4: **Yes. Yeah.** And "add."

In interaction 7, TS7 and U19 followed the exact same pattern:

U19: So once they put in their username and password, then they have to search for [*course name*]?

TS7: **Right. Exactly.** And then the other thing that's happening is some students will go to- So they'll search for like [*course number*] [*section ID*]. [*2 seconds; entering text into search bar*] Or they won't even get that far. They'll only search [*course number*], and let's say they're in section [*section ID*]. They'll, um, like, or [*section ID*]. Let's say

they're in section [*section ID*]. They'll scan it real fast [*hovering cursor over search results*], and they don't realize that there's different sections under one course so they'll skip over this one- and they're just looking for [*section ID*] instead of the combined courses.

In interaction 13, TS2 and U40 followed the same pattern:

U40: Is there one for rhetorical analysis and visual analysis?

TS2: **No.** We don't have them that specific. There is one; however, it doesn't have the points assigned. And that's basically because we just haven't gone through and organized them very well.

In interaction 16, TS7 and U40 followed the same pattern as well:

U40: Or did you save it?

TS7: **Yeah, I just saved it to the desktop.** And it doesn't matter where they save it as long as they know [*4 seconds*] where it is.

In interaction 17, U35 and TS3 followed the same pattern:

U35: Because I could just go in and distribute their points on different assignments instead, right?

TS3: **Yeah, that's possible.** Or you can just give- just make one assignment, that blank assignment, and give them some extra credits. Without using that extra credit.

These instances exemplify a common exchange pattern in which technical-support providers elaborated on their minimal responses to users to explain the confirmation or denial in more detail. As might be expected, technical-support providers must move beyond minimal responses in cases in which the minimal response does not provide enough information to help users. Technical-support

providers need to discern when minimal responses could leave users uncertain still, or they may need to always move beyond minimal responses with additional detail as their default strategy because it increases the chances that users will understand.

In 6 additional instances, technical-support providers responded to users' confirmations or denials with explanations. For example, in interaction 4, TS4 asked for background information, received a confirmation, and then explained how the technology works:

TS4: So basically you're uploading different, uh, files, right?

U20: Uh-huh.

TS4: So we can't actually put the uh- what is it- the bullet points but instead I do as a instructor was that I added label.

Later, TS4 and U20 followed the same pattern:

TS4: Do you want this type of thing, right?

U20: Yeah!

TS4: This is a label.

In interaction 18, U32 and TS2 followed the same pattern but asked for a need:

TS2: Sound good?

U20: Sounds great.

TS2: Then you can continue to use this e-portfolio [*stating web URL*] for if you ever want to demonstrate anything to your students you can use this. You know if you want to create an example portfolio or whatever. You have that space for that.

In one last example, U40 and TS7 followed the same pattern by asking for a need:

TS7: So, you set up like an assignment?

U41: Mm-hmm. So an assignment.

TS7: O.K. Um, that works, but you would have to get them- tell them when you're answering this make sure you take a screenshot of it.

The other 8 instances that had explanations following confirmations or denials followed no discernable commonality. One discernable pattern was in 2 instances in which the sequence contained the following codes: inquiring to check comprehension→confirming or denying→explaining how something works or how to do something. For example, in interaction 19, U41 and TS7 followed this pattern as TS7 explained how to grade the ePortfolios:

U41: Does that make sense?

TS7: **Yeah.** Well you can do the same thing here I mean you can go into as much depth as you want because you can just scroll through and say "O.K. I see you changed these few things." You can do basically end comments.

In interaction 18, U32 and TS2 followed this pattern as TS2 explained how to arrange content on the ePortfolios:

TS2: Do you know how to create sub menus?

U32: **Yes.**

TS2: So that would be one option and the other would just be, you know, make it very clear to [*name*] these are my materials over on these tabs. Whatever.

The other 5 instances were less informative for how the exchanges work in the resolving stage because they were singular instances of exchanges. Ultimately,

these more common patterns show that technical-support providers should employ minimal responses but do so carefully, knowing when it will be necessary to provide additional information beyond that response. In other instances, they should build off users' minimal responses by explaining further if necessary. Technical-support providers may need to be ready to speak more liberally and not hold back their explanations out of reticence to speak more often than users. After all, the resolving stage is about technical-support providers sharing their expertise, which implies they have facility to speak more often than their interlocutors.

The fourth-most common exchange pattern was "inquiring to understand needs or background information" followed by "confirming or denying." I have already explained how this exchange pattern works, particularly as users gave minimal responses often when responding to technical-support providers' inquiries.

The fifth-most common exchange pattern was "signaling" followed by "observing" (28 times). This pattern, like repeated sequences of signaling, makes sense in the resolving stage because of the nature of resolving technical problems. Of the 28 instances, 23 of them involved the technical-support providers responding to their own signals with observations. For example, in interaction 1, TS2 employed this strategy (I bold the observation):

TS2: I'm just looking through a bunch of them. **There's always some indication of like when it started.**

Similarly, TS7 employed it while exploring U2's problem with an assignment module:

TS7: Let's see what we can see. **Time stamped correct. We have an "add submission."**

Likewise, TS8 employed this sequence as he worked on U2's problem with her course website's layout:

TS8: "Edit settings." **No, that's not it.**

In one last example, TS3 employed this sequence as he checked on U35's gradebook calculation:

TS3: Let me check one by one. **So this is correct.**

The remaining five instances were users responding to a signal. For example, in interaction 4, U20 followed this pattern after TS4's signal, which occurred after U20 noticed that the student who has two profiles in Moodle used the same email for both of them:

TS4: So let me un-enroll.

U20: **O.K. but she has the same email for both of them.**

In interaction 14, U2 employed observing after TS8 signaled by stating a button on the screen, determining based off TS8's actions the way the technology works:

TS8: "Collapse topics."

U2: **So it's just an option.**

Later, as TS8 committed to pressing the button, U2 signaled by stating the button as TS8 pressed it. After he pressed it, she observed immediately that it solved her problem:

U2: "Collapse topics." **There we go. O.K.**

In interaction 16, U41 worked on her own computer, attempting to follow along with TS7's explanation. As she did so, she must have tried something and then found it didn't work the way she wanted. I wasn't able to see the screen on her computer:

U41: But let's see. **Well that doesn't work.**

In the same interaction, U41 stated what she did on her computer, and then what she observed after she did it:

U41: We'll see what it opens in. **So I just clicked that and "Preview" opened up.**

These sequences reveal the problem-solving nature of the resolving stage, whether the users or technical-support providers responded to signals with an observation. The resolving stage by its nature implies that the two speakers are working together to move the problem toward resolution. As they did so, they signaled what they were doing and made occasional observations about what they saw occur as they did it. In one notable case, the observation was that the problem was solved as when U2 stated, "[t]here we go. O.K." These instances reveal again that the resolving stage requires often exchange patterns that orient the speakers to problem resolution. Technical-support providers should signal what they are doing or about to do and note interesting observations. They should also listen for users' observations. In making and listening for observations, the discourse can help the participants move more quickly to problem resolution.

Conclusion

Ultimately, the resolving stage shows a lack of consistency not apparent in the previous studies on technical-support interactions. While it would seem that technical-support providers would provide step-by-step instructions and the user would follow along, the resolving stage contained inquiries, additional background information, searching, signaling, major or minor observations, and even frustration. Nevertheless, the speakers worked together to solve the problem, and the discourse enabled them to do that. While my study affirms the previous research on resolving technical problem, it does present some challenges to assumptions in those studies. My study revealed that technical-support providers were less "coaches," as Steehouder (2007, 3) calls them, and more lecturers, questioners, and problem-solvers. Indeed, the most common discourse strategy was signaling, which challenges the presumption that explaining or showing how the technology works or how to do something would be the most prominent discourse strategy in the resolving stage. This study revealed that the "directive" stage, as Agar (1985) calls it, involved more than giving directives, instructions, and solutions. The stage involved problem solving and inquiries more akin to the defining stage. Unlike the defining stage, however, the discourse seemed intended to resolve the problems (not define them), and users and technical-support providers sought and gave information to that end. In the next and final chapter, I discuss these discourse strategies' relationships to satisfaction and, from this analysis, introduce hypotheses that could launch future research. Then, I summarize the answers to my research questions, present limitations, and discuss implications for research and practice.

CHAPTER 7: CONCLUSION

In this chapter, I discuss the findings in relationship to each of the 5 research questions for this study. I do so by first discussing the macrolevel discourse of technical-support interactions. Then I discuss the microlevel discourse for both defining and resolving technical problems. For each discussion of the discourse, I present the implications this study has for technical communication research and practice. For each discussion, I also present hypotheses for future research. I end by presenting limitations to this study.

Macrolevel Discourse in Technical-support Interactions

The first research question related to the macrolevel discourse of technical-support interactions, specifically in face-to-face technical-support interactions, called helpdesks (van Velsen and Steehouder 2003). This research question was the following:

> **RQ1:** In helpdesk interactions, to what extent do the interactions follow the established macrolevel structure?

The helpdesk interactions studied here followed the established macrolevel structure described by Clark, Murfett, Rogers, and Ang (2012), Steehouder (2007), Xu, Wang, Forey, and Li (2010), and their the macrolevel structures mirrored the broader macrolevel structure of institutional discourse (Ang 1985). See table 2.1 for a visual comparison of these studies and appendix F for the final macrolevel discourse codes for this study.

These interactions included an identifying stage in which the technical-support providers greeted and identified the users in relationship to the organization and technology. For example, technical-support providers often

asked for information about which course website was at issue or which ePortfolio website to visit so that the problem definition and resolution could begin. I found that some interactions did not contain such a stage, but likely, the data collection procedure that required the participants to manage the recording process may have excluded some introductory discussion that had some elements of the identifying stage. Second, the interactions in this study included a defining stage that involved a discussion of the problems and the users' reasons for seeking technical help. In this stage, the speakers worked together to bring clarity and definition to the users' experiences and needs. Third, I identified a closing stage by which the technical-support providers and users ended the conversation by ensuring that all problems had been resolved or at least acknowledged. Again, as with the identifying stage, the nature of the recording process may have excluded some concluding exchanges.

The main difference between the findings from earlier research and those of this study, however, was the optionality of either resolving or attempting stages. In other words, after the defining stage, the structure could continue into a resolving stage in cases in which the speakers resolved the problems, or the structure could continue into an attempting stage in which the speakers could not resolve the problems. The genre structure allows potential for either the resolving or attempting stage. The previous research did not include a discussion of unsuccessful problem resolution. Indeed, all the proposed macrolevel structures in the previous studies appear to assume resolved problems. In the present study, the macrolevel discourse of helpdesk interactions followed the established pattern with this one exception. (See Figure 4.1 for a visualization of the genre structure potential and Figure 4.2 for an overview of the macrolevel

structure of the 20 interactions in this study.) Ultimately, this study has confirmed the findings of prior research on the macrolevel structure of these interactions but also has challenged that structure by identifying one potential discourse element: attempting. In identifying this stage, this study can help technical communication researchers identify and describe features of technical-support interactions that involve the technical-support providers' inability to resolve a problem or problems. These features can yield insights about the attempting discourse process and the particular ways technical-support providers speak in cases in which they do not know how to resolve a problem. I did not report on the attempting stage because I focused only on successful interactions. However, I will report such discourse from this study in future work.

In addition to this new finding, this study of technical-support macrolevel discourse presents the first empirical study to identify the generic structure of technical-support interactions. Clark, Murfett, Rogers, and Ang (2012) and Xu, Wang, Forey, and Li (2010) reported on empirical research on customer-support interactions, which are interactions with similar but still different contexts and purposes because there are no technical problems and resolutions involved in customer-support interaction. I modified the frameworks from research on customer-support interactions to build my coding scheme because of how similar these contexts are; that is, they are both examples of institutional discourse and involve representatives from a customer base and a business. Those studies that did explore technical support examined few interactions (from one to three) and proposed their coding framework for future researchers to employ (Baker, Emmison, and Firth 2003, 61; Steehouder 2007, 8). As van Velsen, Steehouder,

and de Jong (2015) put it in their exploration of users' satisfaction with technical-support interactions, research on these interactions requires, as this study argues, more than an understanding of user satisfaction but also "detailed qualitative research into the very specific experiences of customers consulting a helpdesk or helpline" such as "conversation analysis" (228). While this study did not use the analytical procedures of conversation analysis, it did provide a detailed qualitative discourse analysis to understand, as these researchers put it, "the influence of conversational styles on experience" (228). As such, this study presents the first large-scale empirical exploration of the macrolevel content in this technical context by drawing from previous discourse research on both customer- and technical-support interactions.

Future research on the macrolevel discourse of technical-support interactions should explore the effect of particular macrolevel discourse on user satisfaction. Interaction 10 had the lowest satisfaction level of the 20 interactions (somewhat satisfactory). It had no resolving stage but instead four attempting stages. Of the remaining 19 interactions, 8 interactions had the next level of satisfaction (satisfactory). All 8 of these interactions had a combination of resolving and attempting stages (some problems were resolved, others weren't). All remaining 11 interactions had the highest level of satisfaction (very satisfactory), and only 1 of these 11 interactions had an attempting stage (interaction 6). From these 11 very satisfactory interactions, the resolved problems may have influenced the satisfactory outcome. Indeed, the presence of attempting stages means that one or more problems were left unresolved and may have negatively affected the satisfaction levels for the other 9 interactions. I propose the following hypothesis for a study on a larger data set that can more

fully establish a relationship between the attempting stage and satisfactory outcomes:

> **H1:** The presence of an attempting stage has a relationship with the level of satisfaction in technical-support interactions.

To test this hypothesis, in future research, I plan to employ a discourse analysis of randomly sampled technical-support interactions, ensuring that the sample reaches an appropriate level of confidence for a company's, or different companies', annual number of technical-support interactions. With such a sample, I could explore the effect of the attempting stage on varying levels of satisfaction. Exploring the presence of the attempting stage on user satisfaction coincides with the current research's claim that users expect to get their problem solved (Callaghan and Thompson 2002). This hypothesis also follows this reasonable assumption that users would expect technical-support providers to help them resolve their problems or issues.

Apart from having implications for research, this study of macrolevel discourse in these interactions also provides implications for technical communication practice, specifically the practice of training and working as technical-support providers. This macrolevel analysis demonstrates that these interactions do have a generic structure, lending it the identity of a text, or a "passage of discourse which is coherent…with respect to the context of situation, and…with itself" (Halliday and Hasan 1976, 23). In other words, a text has coherence when it uses appropriate register and when it has cohesive features (23). To have appropriate register, a text must employ linguistic features that have coherence with the purpose and situation of the text and also the roles of the speakers (22). Because the speakers have a shared understanding of what

they are trying to accomplish and in what context (to resolve technical problems with a given technology or set of technologies) and what roles they have (user and technical-support provider), the speakers already bring to their interaction an awareness that can lead to a coherent interaction and thus meet this first criterion for a coherent interaction that Halliday and Hasan discuss. To further ensure that they meet this criterion, the technical-support providers should remember that in their roles as experts and institutional representatives, they must also provide a sense of leadership to the conversation, leading its movement from one moment to another.

They can employ this sense of leadership precisely by ensuring the conversation meets the other criterion that Halliday and Hasan (1976) say characterizes coherent discourse. They must ensure the text of the interaction "is coherent with respect to itself, and therefore cohesive" (23). Clear pronoun references, clear elliptical references, and also lexical repetition can help create this cohesion. The current study did not explore the word as the unit of analysis, but other research that takes a conversation analysis approach indeed found that unclear references were causes of miscommunication in helpdesk interactions (Beldad and Steehouder 2015, 184), implying that the interactions in that study lacked this cohesion. However, these microlevel discourse strategies speak to only one aspect of coherence. Carrell (1982) proposes that merely bonding sentences together with such devices does not necessarily create coherence because readers and listeners bring with them to a given message a priori knowledge about the meaning of words and phrases in various contexts, and this knowledge helps them to interpret content apart from cohesion devices (486). Further, Faris and Smeltzer (1997) found that cohesive devices had little to no

effect on readers' comprehension of business messages but that readers' background information about the business situation did have an effect on readers' comprehension. This research implies that writers and readers and speakers and listeners bring with them an intuited understanding of the context and purpose of a given interaction and interpret microlevel content considering that understanding of context and purpose. The understanding they bring with them to a given message is the macrolevel structure of that content. Freed and Roberts (1989) argue, for example, that cognition involves scripts that organize everyday occurrences into memory to suggest appropriate behavior for a specific occurrence, and they argue that cognition involves schema that represent generic concepts individuals have in memory for a given message (333). Drawing from this cognitive process theory, these researchers show how proposals carry with them a generic structure that allow readers to comprehend them as coherent discourse *because* readers make meaning from a priori expectations and purposes for the discourse from the script and schema of proposing. Similarly, users and technical-support providers can interpret technical-support interactions because they make meaning from a priori expectations and purposes for the discourse from the script and schema of giving and receiving technical support, or technical supporting.

Therefore, the macrolevel structure of technical-support interactions must mirror these expectations and purposes in order to help speakers view the interaction as coherent, and any microlevel discourse within the interaction gains meaning because of where it falls within these larger global moments in the interaction. Van Dijk (1980) supports this claim when he explains that speakers have intuitions about macrolevel discourse, which he calls "macrostructures,"

precisely because of the purposes they have when they enter into an interaction: "[U]tterances [should be seen] not only as manifestations of discourse but also as manifestations of social actions" (6). Macrolevel discourse, then, provides a way to interpret microlevel discourse because "[w]ithout [macrostructures] we would only be able to have a large number of links between information units at the local level" (11). Therefore, each microlevel speech act only has coherence when it works within a larger macrolevel speech act (11), whether, as in technical-support interactions, that macrolevel speech act is identifying, defining, attempting, resolving, or closing.

This theory of macrolevel discourse implies that technical-support providers should ensure that the text is "coherent with respect to itself" so that the interaction helps users view the interaction's many microlevel units as coherent with users' expectations and purposes for the interaction (Halliday and Hasan 1976, 23). To help both speakers interpret the microlevel discourse considering the macrolevel discourse, the technical-support providers should effectively employ their leadership role by purposely and explicitly employing the microlevel discourse strategy I called "signaling," which should communicate what stage that the speakers are entering and leaving, serving as a form of metadiscourse, which serves as "an intervention to refer to the discourse organization" (Hyland 2015, 1). For example, the technical-support providers can signal the beginning of the identifying stage: "First, I'd like to identify who you are with respect to the technology. What's your name and what [*course website/portfolio/profile*] will we work with today?" Many microlevel discourse exchanges may make up this identifying stage, but the speakers, especially the

users, can interpret all of that microlevel discourse in light of the macrolevel discourse and thus process the discourse as internally and contextually coherent.

Later in the defining stage, the technical-support providers might transition: "O.K., so now that we have identified what technology we are working with, let's learn more about your question or experience with it." Next, the technical-support providers might say, "O.K., so now that we have defined your problem or question, I'm going to try to resolve it." At a key moment, the technical-support providers can claim or confirm that the problems were resolved (and thus a resolving stage) or not (and thus an attempting stage). They might ask, "so does that resolve that issue?" or "so do you agree that we haven't resolved this issue right now?" They can then move to any additional problems, if the users have them: "O.K., do you have any other problems or issues you'd like to address?" If so, they can re-enter the genre structure.

If not, they can enter the closing stage. The technical-support providers can ensure all questions or issues are addressed and make any other plans that are necessary for coordinating the user's or institution's needs, including scheduling a new appointment or administering a satisfaction survey. For example, "O.K., so let's close this session with a few questions. Have we addressed all your concerns? Would you like to schedule a new appointment or receive a follow-up email? Will you please complete this short survey about your experience?"

By employing these explicit signaling discourse strategies, the technical-support providers can lend coherence to the interaction by denoting explicitly the stages of the interaction. In so doing, they would also employ an appropriate register by exercising their institutional and leading role. Together, this process

lends the interaction coherence, as Halliday and Hasan (1976) and van Dijk (1980) define it, and it may help promote satisfaction because of how the interaction harkens users' cognitive scripts and schemas for technical-support interactions. One hypothesis from this implication would be the following:

> **H2:** Technical-support providers' use of signaling to transition between stages of the macrolevel discourse has a relationship with the level of satisfaction in technical-support interactions.

I plan to explore this hypothesis by exploring in a larger data set the presence of this signaling feature as it denotes movement across the macrolevel structure.

Microlevel Discourse Strategies for Defining Technical Problems

The second and third research questions related to the microlevel discourse of helpdesk interactions. These research questions were the following:

> **RQ2:** In helpdesk interactions, how do technical-support providers and users communicate to diagnose problems?
>
> **RQ3:** In what stage(s) of the technical-support interactions do they diagnose them?

This study found that technical-support providers and users primarily diagnosed the problems in the defining stage. However, this study found that follow-up questions and commentary related to diagnosing did appear in the resolving stage as well; though as demonstrated, many of these questions were unrelated to the primary macrolevel speech act of resolving the problem.

Within the defining stage, users spoke more often than technical-support providers did, and users spent time giving information as the technical-support providers sought that information. Specifically, technical-support providers

inquired to understand needs and background information the most to understand the users' problems or issues. Also, the users most often gave background information. The users and technical-support providers both employed minimal, yes-no type responses, as their second-most common way to diagnose problems. However, users gave more such responses than technical-support providers did as they responded to the technical-support providers' inquiries to understand needs or background information. Lastly, the technical-support providers employed signaling as they worked to define the problem, and the users inquired to understand the technology, which was their third-most common discourse strategy. (See table 5.1 for an overview of these results along with the frequency counts for each discourse type and speaker.)

This study of technical-support microlevel discourse presents the first empirical study to identify the microlevel discourse in technical-support interactions. Clark, Murfett, Rogers, and Ang (2012) provided another microlevel discourse analysis but examined customer-support interactions, and they also focused on empathetic discourse types rather than the cognitive and instructional discourse types in this study. Indeed, in recent studies on technical-support interactions, Beldad and Steehouder (2015) claim that "[p]ublished studies using conversation analysis to investigate understanding problems and the strategies used to deal with them in the context of helpdesk or call center encounters, however, are still non-existent" (184). Even more recently, Lam and Hannah (2016) argued that "existing work examines the documentation of technical issues rather than on the live, dialectical problem-solving or troubleshooting relationship between a user and the company" (39). This study was the first such study, and apart from revealing how the speakers resolve the

problems, it revealed the moment-to-moment interactions between users and technical-support providers as they diagnosed the problems.

In so doing, the study confirmed Agar's (1985) claim that the diagnosing stage (as he calls the macrolevel discourse) manifests the institutional representative seeking information through questioning and the client giving that information (151). Steehouder (2007) and Ang (1985) both agree that the process of problem definition requires aligning the clients' and institutional representatives' understanding, and this study demonstrated how this happens, particularly through users inquiring about background information or technology and also through varying types of responses from both speakers. This study showed that users often used narratives of background information, what Steehouder (2007) calls "historical reports" (3), and this study revealed how they also provided "seemingly irrelevant details" as they communicated their problems, unsure "what might be relevant and what is not" for explaining their problem (7). Further, this study showed how both speakers employ minimal, yes-no responses in their interaction to confirm answers about experiences or to instruct how the technology does or does not work.

Through this study's examination of how technical-support providers employed signaling, I showed how technical-support providers manage relationships with users when they acknowledged users' needs to understand what the technical-support providers were doing or were about to do to define the problem, confirming Pentland's (1995) finding that technical-support providers are problem-solvers with a social awareness of the needs of their interlocutors. Hyland (2015) argues that such metadiscourse, as signaling may be called, reveals speakers' awareness of "the hearer/reader in the text, or at least

how the text creator sees him or her" (2). He shows that such language "guides the receiver's perception of a text" (2). When technical-support providers thought aloud as they did different actions with the technology, specifically when defining the problem, they "present[ed] themselves as credible speakers" (Hyland 2015, 10), because they showed how they were actively engaged in defining the technology. When technical-support providers announced what they were about to do to define the problem, they framed the defining process into a sequence that moved from conversing with users about their experiences to using the technology to define it ("Let's see if…") and then back again to conversing.

For the microlevel discourse these speakers employ to define the problem, future research should test the effect of microlevel discourse on satisfaction levels. Specifically, researchers should explore the relationship between microlevel discourse strategies and satisfaction. This study revealed frequencies for specific microlevel discourse that identified different types of inquiring and answering. My findings and the theories posited in previous research on technical-support and institutional discourse demonstrate that clients should have a larger role in the defining stage and institutional representatives should have a listening and inquiring role. Thus, I propose the following hypotheses:

> **H3:** Users having a more prominent speaking role in the defining stage has a relationship with the level of satisfaction in technical-support interactions.
>
> **H4:** Technical-support providers employing inquiries in the defining stage has a relationship with the level of satisfaction in technical-support interactions.

Lastly, my findings demonstrated that technical-support providers employed signaling to help keep users aware of what is happening during the stage. Previous research on technical-support providers' organizational knowledge suggests that such discourse demonstrates the social awareness technical-support providers have for their clients, and theories of metadiscourse suggest it signals a respect for and awareness of listeners. I thus propose the following hypothesis:

> **H5:** Technical-support providers employing signaling in the defining stage has a relationship with the level of satisfaction in technical-support interactions.

To test the effect of these microlevel discourse strategies on satisfaction, proxy users could rate defining stage transcripts or recordings, as Hall, Verghis, Stockton, and Goh (2014) did with the first two minutes of technical-support interactions. This procedure would help establish defining stages with varying levels of satisfaction. Having established defining stages with varying levels of satisfaction, researchers can relate these microlevel discourse strategies with those satisfaction levels and thus test these hypotheses. Such statistically significant results that relate microlevel discourse strategies and satisfaction could further support the use of discourse strategies that technical-support providers should employ and receive training in for the process of defining technical problems.

Apart from having implications for research, this study also presents implications for technical communication practice, specifically the practice of training and working as technical-support providers. My study revealed that

technical-support providers must inquire to understand users' needs or to seek background information. I have presented types of such inquiries:

- Inquiring to understand general needs
- Inquiring to understand specific needs
- Inquiring to understand problem location
- Inquiring to understand previous actions
- Inquiring to understand previous experiences

While my coding scheme broadly accounted for the discourse type "inquiring to understand needs and background information" and only reached reliability for this general category, these types did appear inductively during the study as I examined the larger discourse type more closely. I detail these subtypes in chapter 5. Future research should reach appropriate reliability for these subtypes. For now, however, technical-support providers should carefully consider the types of questions they ask during the defining stage and how these inquiry types can help them formulate their discourse as they define problems. They should also listen carefully for the corresponding answers to these questions. My study revealed that users often do not focus their topics when sharing background information, shifting from describing locations, to previous actions, to previous experiences rather quickly. As Steehouder (2007) put it, when users describe their problems, "what they do is 'firing a shower of shots,' hoping that one of them will hit the target" (7). As such, technical-support providers require listening skills that can help them differentiate between the purposes users have as they share these scattered narratives and how components of those narratives correspond to the inquiries that the technical-support providers employ.

Also, this study revealed that users may ask to learn how to do something, or they may ask how something works. Both inquiries relate to the technology but imply different goals. Technical-support providers should carefully listen for the kinds of learning that users are pursuing as they listen to these inquiries, whether to learn how to do something or to ask how something works. Such varying user motivations have been explored by scholars such as Redish (1989), who describes users' motivations when encountering documentation, such as reading to do and reading to learn and reading to learn to do (289).

This study also revealed the usefulness of signaling, either to announce or to think-aloud, to coordinate the social dimension during what may otherwise be long periods of silence as technical-support providers navigate the technology to define the problems. This strategy also helps, as a few technical-support providers in this study noted, to instruct users by keeping them learning about how the technology works and how to navigate it as the technical-support providers moved through parts of the websites.

Lastly, this study revealed that minimal, yes-no responses play a large part in the interactions during the defining stage. Technical-support providers employed this strategy often as an instructional tool to a yes-no question from users about the technology, and users often employed it as a minimal response to inquiries about needs or background information. This finding implies that technical-support providers should be wary of asking questions to users that promote such minimal responses (close-ended questions) because users may not provide any additional information beyond the initial response. Examples described in this study demonstrate that users readily expanded on their minimal responses, but they were not prompted to do so. Technical-support

providers run the risk of miscommunication if they continually use yes-no questions to understand users' experiences. Nevertheless, if used judiciously, yes-no questions can yield the information that technical-support providers need to diagnose a problem.

Microlevel Discourse Strategies for Resolving Technical Problems

The fourth and fifth research questions related to the microlevel discourse of helpdesk interactions also. These research questions were the following:

> **RQ4:** In helpdesk interactions, how do technical-support providers communicate to resolve problems?
>
> **RQ5:** In what stage(s) of the technical-support interactions do they resolve them?

This study found that technical-support providers and users resolved the problems primarily in the resolving stage. However, this study found that summaries and commentary related to resolving the problems did appear in the closing stage as well. For example, the technical-support providers reiterated how the problem or problems were solved. These explanations occurred late in the interaction and within the macrolevel purpose of closing the interaction; therefore, resolving occurs primarily in the resolving stage.

Within this resolving stage, technical-support providers spoke more often than users did, and technical-support providers spent time giving information as the users sought that information. Specifically, technical-support providers most often explained how the technology works or how to do something. The users most often inquired to understand the technology. The technical-support providers employed signaling as their second-most frequent discourse strategy

to announce what they were doing or about to do and to think aloud as they resolved the problems. The users often gave additional background information during this process as well. Lastly, both users and technical-support providers employed minimal, yes-no type responses as their third-most common way to resolve problems. Though, technical-support providers gave more such responses than users did as they responded to the users' inquiries to understand the technology. (See table 6.1 for an overview of these results along with the frequency counts for each discourse type and speaker).

The study confirms Agar's (1985) claim that the directive stage (as he calls the macrolevel discourse) involves the institutional representative giving information through advice and solutions (156). Steehouder (2007) notes that the process of problem resolution requires instruction (3), and this study revealed how this happens, particularly through explanations of how the technology works or how to do something with it. This instruction also occurs through minimal responses. My findings challenge Steehouder's (2007) claim that the technical-support providers are coaches who provide step-by-step procedures: "agents instruct clients to perform a number of actions in order to solve the problem" (3). Furthermore, he claims the discourse is comparable to the "'streamlined step' instructions typical for written procedural discourse" (4). He describes instances of a technical-support provider chunking the steps into related groups of steps and using imperative mood, comparable, as he says, to written instructions. This study's findings revealed that such step-by-step discourse happened relatively infrequently and that technical-support providers spent less time coaching and more time lecturing about the way that the technology worked or how to do something with it. Furthermore, this study

showed that users also inserted additional background information about their experience with the problem, even though the problem was already defined. These instances appeared to showcase users' exasperation about the problem at this later stage in the interaction. These instances, it appeared, did not help the technical-support providers' goal to resolve the problem because the background information involved content related to teaching practices, to what the user was planning to do with the technology when it was fixed, or to what the users preferred to do with the technology when they taught with it. While technical-support providers may consider ignoring such discourse as they work to resolve the technical problem, Steehouder (2007) notes that such seemingly irrelevant information may be useful: "there may be possible causes of the problem, and some of these may be unexpected or unique" (7). Still, the present study demonstrated that these instances of users giving background information at this late in the interaction did not appear to relate to the task at hand. Furthermore, because the macrolevel discourse had the pragmatic property of resolving (not defining) the problem, these microlevel discourse strategies appeared to me as incoherent. Nevertheless, previous research has not demonstrated that such seemingly unrelated content appears during this later stage. Steehouder (2007) noted that such unrelated content appears, rather, in the defining stage. Future research should explore further the presence of content unrelated to resolving the problem in the resolving stage to identify how technical-support providers coopt the information to help them resolve the technical problems or how technical-support providers brush past it.

This study also showed technical-support providers employing signaling as they resolved the problem, demonstrating how they manage relationships

with users by acknowledging users' needs to understand what they were doing or were about to do to resolve the problems. As in this defining stage, technical-support providers used this strategy frequently. Again, such strategies showed technical-support providers' awareness of their listeners as they moved from conversation to using the technology to resolve the problem. Lastly, this study revealed the effectiveness of yes-no, minimal responses to inquiries about how the technology works. Ultimately, these findings reveal what previous research has not demonstrated: resolving the technical problem involves more than step-by-step procedures. It involves continued question and answer, substantive though minimal responses, wait times as technical-support providers used the technology, and at times, unrelated background information.

For the microlevel discourse these speakers employ to resolve the problems, future research should test the effect of this microlevel discourse on satisfaction levels. This study revealed frequencies for specific microlevel discourse types that identified different types of inquiries and answers during the resolving process. These types may have a relationship to satisfactory technical-support interactions. The findings here demonstrated that the resolving stage entails technical-support providers having the larger speaking role and users having the listening and occasionally inquiring role. Thus, I propose the following hypothesis:

> **H6:** Technical-support having a more prominent speaking role in the resolving stage has a relationship with the level of satisfaction in technical-support interactions.

Lastly, my findings demonstrate that technical-support providers employ signaling to help keep users aware of what is happening during the stage.

Previous research on technical-support providers' organizational knowledge and theories about metadiscourse suggest that such discourse demonstrates the social awareness technical-support providers have for their clients during the interaction. I thus propose the following hypothesis:

> **H7:** Technical-support providers employing signaling in the resolving stage has a relationship with the level of satisfaction in technical-support interactions.

To test the effect of microlevel discourse on satisfaction, proxy users might rate numerous resolving stage transcripts or recordings, as Hall, Verghis, Stockton, and Goh (2014) did, to establish resolving stages with varying levels of satisfaction. Having established resolving stages with varying levels of satisfaction, researchers can relate these microlevel discourse strategies with satisfaction and thus test these hypotheses. Statistically significant results that relate microlevel discourse strategies and satisfaction could further support the use of discourse strategies that technical-support providers should employ and receive training in for the process of resolving technical problems.

Also in this study, the technical-support providers more often used explanations rather than step-by-step directives. This finding challenges current research, and future research using a larger data set could explore whether this study presents unique (rather than common) technical-support instructional practices. This future work would ground claims that view technical-support providers as either coaches, as Steehouder (2007) views them, or lecturers, as this study showed them.

Apart from having implications for research, this study also presents implications for technical communication practice, specifically the practice of

training and working as technical-support providers. This study revealed that technical-support providers explained often how to do something with the technology or how it worked. It did not frequently find technical-support providers *showing* how the technology works or how to do something with it, though it was a discourse code and did appear at times. The discourse code allows for an additional visual component that may complement the explanation and assist users in their understanding. "Showing" involves the same components of "explaining" but with the additional use of the technology to either highlight text on the screen, hover the cursor over key parts of the screen, or demonstrate operations rather than speak of them abstractly. In other words, showing employs the "tangibles" of helpdesks, which van Velsen, Steehoulder, and de Jong (2007) defined as "the physical aspects of the organization that promise a certain level of service" (220). In this study, the users and technical-support providers employed a desktop computer between them. This tangible gave technical-support providers the means to supplement their instruction. Furthermore, if they have opportunity to employ "telling," which involves the imperative mood and sequenced instruction that Steehouder (2007) discusses, they can complement this instruction with the tangibles as well. In many ways, showing uses the features of instructional videos because showing and instructional videos both provide "procedural information in multiple simultaneous channels (text, moving image, sound), creating complementary repetition that can help users isolate instructional messages" (Swarts 2015, 197). With these similarities in mind, technical-support providers, particularly those in helpdesks or those who in helplines who use remote controlled access to users' computers, should avail themselves of best practices in instructional video

design. For example, Swarts (2015) argues that the instructor in instructional videos should announce the step before the instructor shows the step, with a pause "long enough to get [the user] mentally 'set' for an action" (200). Such insights from this and other research in instructional videos might enhance the instructional process in the resolving stage, whether the instruction is step-by-step sequences or long-form explanation.

This study also revealed that users provided background information about their experience late in the discussion, even as far into it as the resolving stage, suggesting that technical-support providers should carefully consider when users are providing information that helps them to define or resolve the problem (the main goal for obtaining background information) and when the information is not as pertinent and seems more to help the users express their frustration. At the least, technical-support providers should be aware of this tendency from users, and at the most, technical-support providers should redirect the conversation back to the task at hand, resolving the problem. Also, taking Steehouder's (2007) advice in consideration that the information may prove insightful for understanding the problem more deeply, the technical-support providers should not dismiss this information outright.

Limitations

This study provided insight about the discourse in technical-support interactions that no previous study has done as completely. However, this study did have limitations. More data would reveal more useful information about the discourse in these interactions. The data-collection procedures took months and relied on users who were enrolled in the study to incidentally encounter a technical

problem and visit the technical-support helpdesk. These chance encounters took time, and for this reason, time constraints required the data-collection period to end. Though analyzing more interactions would provide an opportunity to explore further variations and additional examples of discourse, the 20 interactions in this study provided a rich data set from which to understand how helpdesk interactions work and still yielded insight for answering the research questions in this study and for projecting into future research.

Also, the discourse codes should be refined further and potentially split to form more precise discourse codes. For example, inquiring to understand needs and background information implies a logical split between two sorts of inquiries that with more time could be finessed into two reliable codes. Such precision in the coding scheme would have strengthened the findings and analysis. However, the 16-item coding scheme did reveal useful findings for understanding these interactions and for setting-up future research.

The data-collection procedures involved a level of obtrusiveness that may have altered the findings in small ways. The participants had to administer their own recording process, and the users and technical-support providers had to enroll in the study before they could be recorded. Were the interactions already recorded and part of an archive of organizational recordings, the organization (not each participant) would have provided consent for me to collect and analyze these interactions, which would have provided more natural results, yielding potentially different and richer findings for certain parts of the interaction, such as the identifying or closing stages. Despite this limitation, by the time the participants in this study got to the defining stage, the recording process appeared largely forgotten and the interaction more natural. For this reason,

these limitations did not hinder the focus of this study on the discourse involved when diagnosing and resolving technical problems because the participants acted more naturally by these points in the interactions.

Further, to enrich the triangulation process, post-session stimulated-recall interviews with users may have helped refine my interpretation of users' intentions. However, because the study primarily focused on technical-support providers, I focused instead on how what users said and did could influence the technical-support providers. Furthermore, this study did not have a goal of instructing users how to interact during their visits. For this reason, I reached the goal of the study by ensuring the discussion and implications focused on technical-support providers (not users), with whom I did complete post-session stimulated-recall interviews. I could use these interviews to triangulate my findings, ensuring that my interpretations mirrored the insights the technical-support providers shared about their intentions during moments in the interactions.

Conclusion

This study has revealed what the discourse at the macro- and microlevels looks like in technical-support interactions. It did so by analyzing 20 helpdesk interactions to determine how users and technical-support providers work together to diagnose and resolve technical problems. I calculated frequencies of discourse codes, word volubility, and presented examples of microlevel discourse. I determined that in many ways this study confirms what other research and general assumptions might assume about technical-support interaction, such as the genre structure, who speaks most often and when, and

when inquiries or answers might appear more often. I did describe, however, how the research challenges some of the ideas in previous research. In particular, this study demonstrated the messiness in these interactions such as the following:

- unresolved problems
- off-topic background information
- background information later in the interaction
- unfocused problem narratives
- unexpected inquiries about how the technology works later in the interaction
- fewer instances of step-by-step instructions
- many instances of short and long explanations

Ultimately, this study has provided the first empirical discourse analysis to inform the research and practice of technical-support interactions. This study also provides empirically driven advice to technical-support providers for how to communicate when diagnosing and resolving technical-support problems. It also shows what kind of discourse technical-support providers can expect from users at various parts of the interaction and how to react to it. In describing these instances and in giving advice to these workers, this study reached its goals to help organizations to maintain users, to help managers to train technical-support providers, to help technical-support providers to succeed at their work, and perhaps most central to the technical communication enterprise: to help users to get what they need.

REFERENCES

Aggarwal, Praveen, Stephen B. Castleberry, Rick Ridnour, and C. David Shepherd. 2005. "Salesperson Empathy and Listening: Impact on Relationship Outcomes." *Journal of Marketing Theory and Practice* 13 (3): 16–31.

Akrich, Madeliene. 1992. "The De-scription of Technical Objects." In *Shaping Technology/Building Society: Studies in Sociotechnical Change*. Edited by Wiebe E. Bijker and John Law, 205–224. Cambridge: MIT Press.

Armistead, Colin, Julia Kiely, Linda Hole, and Jean Prescott. 2002. "An Exploration of Managerial Issues in Call Centres." *Managing Service Quality: An International Journal* 12 (4): 246–56.

Agar, Michael. 1985. "Institutional Discourse." *Text: Interdisciplinary Journal for the Study of Discourse* 5 (3): 147–68.

Baker, Carolyn, Michael Emmison, and Alan Firth. 2005. "Calibrating for Confidence in Calls to Technical Support." In *Calling for Help: Language and Social Interaction in Telephone Helplines*. Edited by Carolyn Baker, Michael Emmison, and Alan Firth, 39–62. Amsterdam: John Benjamin Publishing Company.

Barley, Stephen R. 1996. "Technicians in the Workplace: Ethnographic Evidence for Bringing Work into Organizational Studies." *Administrative Science Quarterly* 41 (3): 404–41.

Barton, Ellen. 2004. "Discourse Methods and Critical Practice in Professional Communication: The Front-Stage and Back-Stage Discourse of Prognosis

in Medicine." *Journal of Business and Technical Communication* 18 (1): 67–111.

Bearden, William O., Manoj K. Malhotra, and Kelly H. Uscategui. 1998. "Customer Contact and the Evaluation of Service Experiences: Propositions and Implications for the Design of Research." *Psychology & Marketing* 15 (8): 793–809.

Beldad, Ardion, and Michael Steehouder. 2015. "Not that Button but the Other: Misunderstanding and Non-understanding in Helpdesk Encounters Involving Nonnative English Speakers." *Technical Communication* 59 (3): 179–94.

Bell, Simon J., Seigyoung Auh, and Karen Smalley. 2005. "Customer Relationship Dynamics: Service Quality and Customer Loyalty in the Context of Varying Levels of Customer Expertise and Switching Costs." *Journal of the Academy of Marketing Science* 33 (2): 169–83.

Bettencourt, Lance A., and Stephen W. Brown. 1997. "Contact Employees: Relationships Among Workplace Fairness, Job Satisfaction and Prosocial Service Behaviors." *Journal of Retailing* 73 (1): 39–61.

Bernhardt, Stephen A. 2003. "Improving Document Review Practices in Pharmaceutical Companies." *Journal of Business and Technical Communication* 17 (4): 439–73.

Blum-Kulka, Shoshana. 1987. "Indirectness and Politeness in Requests: Same or Different?" *Journal of Pragmatics* 11 (2): 131–46.

Bordoloi, Sanjeev. K. 2004. "Agent Recruitment Planning in Knowledge-intensive Call Centers." *Journal of Service Research* 6 (4): 309–23.

Brown, Penelope, and Stephen C. Levinson. 1987. *Politeness: Some Universals in Language Usage*. Cambridge: Cambridge University Press.

Brown, Gillian, and George Yule. 1983. *Discourse Analysis*. Cambridge: Cambridge University Press.

Burgers, Arjan, Ko de Ruyters, Cherie Keen, and Sandra Streukens. 2000. "Customer Expectation Dimensions of Voice-to-Voice Service Encounters: A Scale-Development Study." *International Journal of Service Industry Management* 11 (2): 142–61.

Callaghan, George, and Paul Thompson. 2002. "'We Recruit Attitude:' The Selection and Shaping of Routine Call Centre Labour." *Journal of Management Studies* 39 (2): 233–54.

Carrell, Patricia L. 1982. "Cohesion is Not Coherence." *TESOL Quarterly* 16 (4): 479–88.

Clark, Colin McKinnon, Ulrike Marianne Murfett, Priscilla S. Rogers, and Soon Ang. 2012. "Is Empathy Effective for Customer Service? Evidence from Call Center Interactions." *Journal of Business and Technical Communication* 27 (2): 123–53.

D'Cruz, Premilla, and Ernesto Noronha. 2008. "Doing Emotional Labour: The Experiences of Indian Call Centre Agents." *Global Business Review* 9 (1): 131–47.

Dobrin, David. 2004. "What's Technical about Technical Writing?" In *Central Works in Technical Communication*. Edited by Stuart A. Selber and Johndan Johnson-Eilola, 107–23. New York: Oxford University Press.

Dorman, Christian and Dieter Zapf. 2004. "Customer-related Social Stressors and Burnout." *Journal of Occupational Health Psychology* 9 (1): 61–82.

Faber, Brenton. 2007. *Discourse, Technology, and Change*. New York, NY: Bloomsbury.

Faris, Kay A., and Larry R. Smeltzer. 1997. "Schema Theory Compared to Text-Centered Theory as an Explanation for the Readers' Understanding of a Business Message." *The Journal of Business Communication* 34 (1): 7–26.

Freed, Richard C., and David D. Roberts. 1988. "The Nature, Classification, and Generic Structure of Proposals." *Journal of Technical Writing and Communication* 19 (4): 317–51.

Frei , Richard. L., and Michael A. McDaniel. 1998. "Validity of Customer-service Measures in Personnel Selection: A Review of Criterion and Construct Evidence." *Human Performance* 11 (1): 1–27.

Gee, James Paul. 2011. *Discourse Analysis: Theory and Method*. 3rd edition. New York, NY: Routledge.

Giddens, Anthony. 1984. *The Constitution of Society*. Berkley: University of California Press.

Goffman, Erving. 1981. "Replies and Responses." In *Forms of Talk*, 5–77. Philadelphia: University of Pennsylvania Press.

Govindarajulu, Chittibabu. 2002. "The Status of Helpdesk Support." *Communications of the ACM* 45 (1): 97–100.

Graesser, Arthur C., Natalie K. Person, & J. Huber. 1992. "Mechanisms That Generate Questions." In *Questions and Information Systems*, 167–87. Edited by Thomas Wesley Lauer, Eileen Peacock, and Arthur C. Graesser. Hillsdale, NJ: Lawrence Erlbaum.

Halbe, Dorathea. 2012. "'Who's There?' Differences in the Features of Telephone and Face-to-Face Conferences." *Journal of Business Communication* 49 (1): 48–73.

Halliday, Michael A.K., and Christian M.I.M. Mathiessen. 2014. *Introduction to Functional Grammar*. 3rd edition. New York, NY: 2014.

Halliday, Michael A.K., and Raqaiya Hasan. 1976. *Cohesion in English*. New York: Routledge.

Hall, Judith A., Phil Verghis, William Stockton, and Jin X. Goh. 2014. "It Takes Just 120 Seconds: Predicting Satisfaction in Technical Support Calls." *Psychology & Marketing* 31 (7): 501–8.

Herndl, Carl G., and Adela C. Licona. 2007. "Shifting Agency: Agency, Kairos, and the Possibilities of Social Action." In *Communicative Practices in Workplaces and the Professions: Cultural Perspectives on the Regulation of Discourse and Organizations*. Edited by Mark Zachry and Charlotte Thralls, 133–54. Amityville, NY: Baywood Publishing Company.

Hyland, Ken. 2015. "Metadiscourse." In *The International Encyclopedia of Language and Social Interaction*. Edited by Karen Tracy, 1–11. Hoboken, NJ: John Wiley & Sons.

Islam, A.K.M Najmul. 2014. "Sources of Satisfaction and Dissatisfaction with a Learning Management System in a Post-Adoption Stage: A Critical Incident Technique Approach." *Computers in Human Behavior* 30:249–61.

Johnson-Eilola, Johndan. 1996. "Relocating the Value of Work: Technical Communication in a Post-industrial Age." *Technical Communication Quarterly* 5 (3): 245–70.

———. 2005. *Datacloud: Toward a New Theory of Online Work*. New York: Hampton Press.

Kelly, Kendall. 2014. "Coding Miscommunication: A Method for Capturing the Vagaries of Language." *IEEE Professional Communication Conference Proceedings 2014*.

Kiedaisch, Jean, and Sue Dinitiz. 1993. "Look Back and Say 'So What?': The Limitations of the Generalist Tutor." *The Writing Center Journal* 14 (1): 63–74.

Kimball, Miles. 2016. "The Golden Age of Technical Communication." *Journal of Technical Writing and Communication*. Retrieved from: http://jtw.sagepub.com. /content/early/2016/04/22/0047281616641927.full.pdf+html

Kinnie, Nick, Sue Hutchinson, and John Purcell. 2000. "'Fun and Surveillance:' The Paradox of High Commitment Management in Call Centres." *The International Journal of Human Resource Management* 11 (5): 967–85.

Lam, Chris, and Mark A. Hannah. 2016. "The Social Help Desk: Examining How Twitter is Used as a Technical Support Tool." *Communication Design Quarterly* 4 (2): 37–51.

Lozano, Luis M., Eduardo García-Cueto, and José Muñiz. 2008. "Effect of the Number of Response Categories on the Reliability and Validity of Rating Scales." *Methodology* 4 (2): 73–79.

Lee, Yi-Hsuan, Yi-Chuan Hseih, and Chun-Yuan Ma. 2011. "A Model of Organizational Employees' E-learning Systems Acceptance." *Knowledge-Based Systems* 24: 355–66.

Mackiewicz, Jo. 2004. "The Effects of Tutor Expertise in Engineering Writing: A Linguistic Analysis of Writing Tutors' Comments." *IEEE Transactions on Professional Communication* 47 (4): 316–28.

———. 2005. "Hinting at What They Mean: Indirect Suggestions in Writing Tutors' Interactions with Engineering Students." *IEEE Transactions on Professional Communication* 48: 365–76.

———. 2017. *The Aboutness of Writing Center Talk: A Corpus-Driven and Discourse Analysis*. New York: Routledge.

Mackiewicz, Jo, and Isabelle Kramer Thompson. 2015. *Talk about Writing: The Tutoring Strategies of Experience Writing Center Tutors*. New York: Routledge.

———. 2014. "Instruction, Cognitive Scaffolding, and Motivational Scaffolding in Writing Center Tutoring." *Composition Studies* 42 (1): 54–78.

Mackiewicz, Jo, and Kathryn Riley. 2003. "The Technical Editor as Diplomat: Linguistic Strategies for Balancing Clarity and Politeness." *Technical Communication* 50 (1): 83–94.

Mehlenbacher, Brad. 2013. "What Is the Future of Technical Communication?" In *Solving Problems in Technical Communication*. Edited by Johndan Johnson-Eilola and Stuart A. Selber, 187–209. Chicago: University of Chicago Press.

Mount, Michael K., Murray R. Barrick, and Greg L. Stewart. 1998. "Five-factor Model of Personality and Performance in Jobs involving Interpersonal Interactions." *Human Performance* 11 (2/3): 145–65.

Murphy, Susan Wolff. 2006. "'Just Chuck It: I Mean Don't Get Fixed on It': Self-Presentation in Writing Center Discourse." *The Writing Center Journal* 26 (1): 62–82.

Nwaz, Allah, and Muhammad Zubair Khan. 2012. "Issues of Technical Support for E-Learning Systems in Higher Education." *International Journal of Modern Education and Computer Science* 2:38–44.

Nguyen, Markus Groth, Gianfranco Walsh, Thorstein Hennig-Thurau. 2014. "The Impact of Service Scripts on Customer Citizenship Behavior and the Moderating Role of Employee Customer Orientation." *Psychology and Marketing* 31(12): 1196–109.

Palmer, Jaqueline S., and M. Jimmie Killingsworth. "Research and Consulting in Technical Communication." *Technical Communication Quarterly* 11 (4): 389–409.

Paltridge, Brian. 2012. *Discourse Analysis: An Introduction*. 2nd edition. New York, NY: Bloomsbury.

Parasuraman, A., Valerie A. Zeithaml, and Leonard L. Berry. 1988. "SERVQUAL: A Multi-Item Scale For Measuring Consumer Perceptions of Service Quality." *Journal of Retailing* 64 (1): 12–40.

Pentland, Brian T. 1992. "Organizing Moves in Software Support Hot Lines." *Administrative Science Quarterly* 37 (4): 527–48.

Pontes, Manuel C. F., and Colleen O'Brien Kelly. 2000. "The Identification of Inbound Call Center Agents' Competencies that are Related to Callers' Repurchase Intentions. *Journal of Interactive Marketing* 14 (3): 41–49.

Redish, Janice. 1989. "Reading to Learn to Do." *IEEE Transactions on Professional Communication* 32 (4): 289–93.

———. 2003. "Adding Value as a Professional Technical Communicator." *Technical Communication* 50 (4): 505–518.

Rourke, Liam, Terry Anderson, D. R. Garrison, and Walter Archer. 2000. "Methodological Issues in the Content Analysis of Computer Conference Transcripts." *International Journal of Artificial Intelligence in Education* 11:1–16.

Saldaña, Johnny. 2013. *The Coding Manual for Qualitative Researchers*. 2nd edition. Thousand Oaks: Sage Publication.

Schryer, Catherine F. 1993. "Records as Genre." *Written Communication* 10 (2): 200–34.

Searle, John. 1976. "The Classification of Illocutionary Acts." *Language in Society* 5:1–24.

Selber, Stuart A. 2010. "A Rhetoric of Instruction Sets." *Technical Communication Quarterly* 19 (1): 95–117.

Slack, Jennifer Daryl, David James Miller, and Jeffrey Doak. 1993. "The Technical Communicator as Author: Meaning, Power, Authority." *Journal of Business and Technical Communication* 7 (1): 12–36.

Steehouder, Michael, and Daniel Hartman. 2003."How Can I Help You? User Instruction in Telephone Calls." *IEEE Professional Communication Conference Proceedings 2003*: 158–65.

Steehouder, Michael. 2007. "How Helpdesk Agents Help Clients" *IEEE Professional Communication Conference Proceedings 2007*.

Swales, John. 1990. *Genre analysis: English in Academic and Research Settings*. Cambridge, UK: Cambridge University Press.

Swarts, Jason. 2014. "Help is in the Helping: An Evaluation of Help Documentation in a Networked Age." *Technical Communication Quarterly* 24 (2): 164–87.

———. 2015. "New Modes of Help: Best Practices for Instructional Video." *Technical Communication* 59 (3): 195–206.

Thompson, Isabelle, and Jo Mackiewicz. 2014. "Questioning in Writing Center Conferences." *The Writing Center Journal* 33 (2): 37–70.

Thompson, Isabelle, Alyson Whyte, David Shannon, Amanda Muse, Kristen Miller, Milla Chappell, and Abby Whigham. 2009. "Examining Our Lore: A Survey of Students' and Tutors' Satisfaction in Writing Center Conferences." *The Writing Center Journal* 29 (1): 78–105.

van Velsen, Lex S., Michael F. Steehouder, Menno D.T. de Jong. 2007. "Evaluation of User Support: Factors that Affect User Satisfaction with Helpdesks and Helplines." *IEEE Transactions on Professional Communication* 50 (3): 219–31.

van Dijk, Teun A. 1980. *Macrostructures: An Interdisciplinary Study of Global Structures in Discourse, Interaction, and Cognition*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Wrench, Jason S., Candice Thomas-Maddox, Virginia Peck Richmond, and James C. McCroskey. 2013. *Quantitative Research Methods for Communication: A Hands-on Approach*, 2nd edition. New York: Oxford University Press.

Xu, Xunfeng, Yan Wang, Gail Forey, and Lan Li. 2010. "Analyzing the Genre Structure of Chinese Call-Center Communication." *Journal of Business and Technical Communication* 24 (4): 445–75.

APPENDIX A: IRB APPROVAL PAGE

IOWA STATE UNIVERSITY
OF SCIENCE AND TECHNOLOGY

Institutional Review Board
Office for Responsible Research
Vice President for Research
1138 Pearson Hall
Ames, Iowa 50011-2207
515 294-4566
FAX 515 294-4267

Date: 11/11/2015

To: Vincent D Robles
447 Ross Hall

CC: Dr. Jo Mackiewicz
413 Ross Hall
Dr. David Roberts
241 Ross Hall

From: Office for Responsible Research

Title: Technical Support Interactions with Users

IRB ID: 15-622

Approval Date: 11/9/2015 **Date for Continuing Review:** 11/2/2017

Submission Type: New **Review Type:** Full Committee

The project referenced above has received approval from the Institutional Review Board (IRB) at Iowa State University according to the dates shown above. Please refer to the IRB ID number shown above in all correspondence regarding this study.

To ensure compliance with federal regulations (45 CFR 46 & 21 CFR 56), please be sure to:

- **Use only the approved study materials** in your research, **including the recruitment materials and informed consent documents that have the IRB approval stamp**.
- **Retain signed informed consent documents for 3 years after the close of the study**, when documented consent is required.
- **Obtain IRB approval prior to implementing any changes** to the study by submitting a Modification Form for Non-Exempt Research or Amendment for Personnel Changes form, as necessary.
- **Immediately inform the IRB of** (1) **all serious and/or unexpected adverse experiences** involving risks to subjects or others; and (2) **any other unanticipated problems involving risks** to subjects or others.
- **Stop all research activity if IRB approval lapses**, unless continuation is necessary to prevent harm to research participants. Research activity can resume once IRB approval is reestablished.
- **Complete a new continuing review form** at least three to four weeks prior to the **date for continuing review** as noted above to provide sufficient time for the IRB to review and approve continuation of the study. We will send a courtesy reminder as this date approaches.

Please be aware that IRB approval means that you have met the requirements of federal regulations and ISU policies governing human subjects research. **Approval from other entities may also be needed.** For example, access to data from private records (e.g. student, medical, or employment records, etc.) that are protected by FERPA, HIPAA, or other confidentiality policies requires permission from the holders of those records. Similarly, for research conducted in institutions other than ISU (e.g., schools, other colleges or universities, medical facilities, companies, etc.), investigators must obtain permission from the institution(s) as required by their policies. **IRB approval in no way implies or guarantees that permission from these other entities will be granted.**

Upon completion of the project, please submit a Project Closure Form to the Office for Responsible Research, 1138 Pearson Hall, to officially close the project.

Please don't hesitate to contact us if you have questions or concerns at 515-294-4566 or IRB@iastate.edu.

APPENDIX B: PILOT STUDY INTERVIEW PROTOCOL

This protocol is semi-structured, with these general questions to prime the conversation.

1. How do you try to portray to users that you are listening to them? What's an example of your trying to give users the impression that you're listening to them?
2. How do you try to portray to users that you understand their problem? What's an example of your portraying your understanding of the problem?
3. How do you ensure you understand what the user is asking you? What's an example of your ensuring you understand what the user is asking you?
4. How do you try to build rapport with users in your sessions? What's an example of your trying to build rapport with users?
5. When solving a technical problem, how do you try to help users understand what you're telling them? What's an example of your trying to help users understand what you're telling them?

APPENDIX C: SCREENING SURVEY

1. Are you 18 years of age or older?
    a. Yes
    b. No
2. Describe your experience with the Moodle Learning Management System.
    a. Less than 1 year experience
    b. 1–2 years' experience
    c. 2–3 years' experience
    d. 3–4 years' experience
    e. 5 or more years' experience
3. I am proficient with Moodle Learning Management System.
    a. Strongly Agree
    b. Agree
    c. Disagree
    d. Strongly Disagree
4. Describe your experience with the ePortfolio system.
    a. Less than 1 year experience
    b. 1–2 years' experience
    c. 2–3 years' experience
    d. 3–4 years' experience
    e. 5 or more years' experience
5. I am proficient with the ePortfolio system.
    a. Strongly Agree
    b. Agree
    c. Disagree
    d. Strongly Disagree
6. Describe your experience with the teaching strategies depository system.
    a. Less than 1 year experience
    b. 1–2 years' experience
    c. 2–3 years' experience
    d. 3–4 years' experience
    e. 5 or more years' experience
7. I am proficient with the teaching strategies depository system.
    a. Strongly Agree
    b. Agree
    c. Disagree
    d. Strongly Disagree

I acknowledge by printing, signing, and dating my name that my answers to this questionnaire are true to the best of my knowledge.

Print Name:_____

Signature:_____

Date:_____

APPENDIX D: PILOT STUDY POST-SESSION SURVEY

TECHNICAL SUPPORT INTERACTIONS WITH USERS: TECHNICAL SUPPORT MEMBER

Post-Session Survey

Name:_____ Date:_____ Time:_____

Please answer the following questions about the support session you just had.

1. I listened carefully to the user during this session.
    a. Strongly Agree
    b. Agree
    c. Disagree
    d. Strongly Disagree
2. I was polite during this session.
    a. Strongly Agree
    b. Agree
    c. Disagree
    d. Strongly Disagree
3. I successfully addressed the problem the user had for this session.
    a. Strongly Agree
    b. Agree
    c. Disagree
    d. Strongly Disagree
4. I used language the user could understand when discussing the problem.
    a. Strongly Agree
    b. Agree
    c. Disagree
    d. Strongly Disagree
5. The user felt the problem was addressed.
    a. Strongly Agree
    b. Agree
    c. Disagree
    d. Strongly Disagree
6. The user appeared to listen during this session.
    a. Strongly Agree
    b. Agree
    c. Disagree

TECHNICAL SUPPORT INTERACTIONS WITH USERS: USER

Post-Session Survey

Name:_____ Date:_____ Time:_____

Please answer the following questions about the support session you just had.

1. I listened carefully to the technical support person during this session.
    a. Strongly Agree
    b. Agree
    c. Disagree
    d. Strongly Disagree
2. My problem was successfully addressed during this session.
    a. Strongly Agree
    b. Agree
    c. Disagree
    d. Strongly Disagree
3. I understood what the technical support person was telling me.
    a. Strongly Agree
    b. Agree
    c. Disagree
    d. Strongly Disagree
4. The technical support person felt the problem was addressed.
    a. Strongly Agree
    b. Agree
    c. Disagree
    d. Strongly Disagree
5. The technical support person appeared to listen to me during this session.
    a. Strongly Agree
    b. Agree
    c. Disagree
    d. Strongly Disagree
6. The technical support person was polite during this session.
    a. Strongly Agree
    b. Agree
    c. Disagree
    d. Strongly Disagree

APPENDIX E: MAIN STUDY POST-SESSION SURVEY

TECHNICAL SUPPORT MEMBER Post-Session Survey
Name:_____ Date:_____
Please indicate how much you agree with the following statements about the support session you just had.

1. I answered different question(s) or complaint(s) the user had with little difficulty.
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree
2. I adapted to every situation that occurred during the session.
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree
3. I took the user's knowledge into account when helping solve the problem(s).
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree
4. I remained calm and friendly no matter what feelings I was interpreting from the user.
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree
5. I helped define specifically the problem(s).
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree
6. I was able to help with each and every problem in a timely way.
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree
7. I clearly and thoroughly explained each and every step I took when solving the problem(s).
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree
8. I clearly and thoroughly explained solutions or recommendations.
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree
9. I was able to imagine what the user was going through with his or her problem(s).
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree
10. I treated the user uniquely from other users.
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree
11. I treated the user's problem(s) as important.
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree
12. I had the necessary authority to solve the user's problem(s).
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree
13. I will have to follow up with the user to help him or her with the problem(s) because I need to seek permission or help.
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree

USER Post-Session Survey
Name:_____ Date:_____
Please indicate how much you agree with the following statements about the support session you just had.

1. The technical support person answered different question(s) or complaint(s) I had with little difficulty.
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree
2. The technical support person adapted to every situation that occurred during the session.
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree
3. The technical support person took my knowledge into account when helping solve the problem(s).
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree
4. The technical support person remained calm and friendly no matter how I was feeling.
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree
5. The technical support person helped define specifically the problem(s).
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree
6. The technical support person was able to help with each and every problem in a timely way.
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree
7. The technical support person clearly and thoroughly explained each and every step he or she took when solving the problem(s).
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree
8. The technical support person clearly and thoroughly explained solutions or recommendations.
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree
9. The technical support person was able to imagine what I was going through with my problem(s).
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree
10. The technical support person treated me uniquely from other users.
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree
11. The technical support person treated my problem(s) as important.
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree
12. The technical support person had the necessary authority to solve my problem (s).
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagree
13. The technical support person will have to follow up with me to help me with the problem(s) because he or she needs to seek permission or help.
    1. Strongly Agree
    2. Agree
    3. Agree Somewhat
    4. Disagree Somewhat
    5. Disagree
    6. Strongly Disagre

APPENDIX F: FINAL CODING SCHEMES

Final Macrolevel Discourse Coding Scheme

After three rounds of coding and reaching a satisfactory agreement level with coders, I determined the following codebook acceptable for coding the macrolevel discourse in my main study of technical-support interactions.

Macrolevel Codebook

Code ID	Definition
Identifying	Identifying U as part of the technical system such as Moodle, including obtaining U's name and any other pertinent identifying information about U, such as course section.
Defining	Outlining, summarizing, and/or indicating that there is a problem or question. Often prompted by U but could also be prompted by TS.
Attempting	Working through possible solutions to the problem or possible answers to the question. The problem does not get resolved fully or the question answered fully in that session. TS or U may not be satisfied with a proposed resolution or answer. Or U and/or TS move on to a new problem without a resolution or answer.
Resolving	Providing information, instruction, and/or solutions for a problem and confirming a specific problem is resolved. TS and U are satisfied with resolutions or answers. The problem has to be resolved or the question answered in that session. Making plans to solve the problem at another time (e.g., following-up through e-mail or another meeting, or trying something later at home) does not mean the problem or question was resolved or answered.
Closing	Confirming that U is satisfied, that U has no more problems to talk about, and saying good bye and/or setting-up a follow-up meeting or email conversation; includes taking the post-session survey if recorded

Final Microlevel Discourse Coding Scheme

After nine rounds of coding and reaching a satisfactory agreement level with coders, I determined the following codebook acceptable for coding the microlevel discourse in my main study of technical-support interactions.

Microlevel Codebook

Code ID	Description for TS and U
inquiring to understand needs or background information	inquiring to understand or confirm listener's needs or background information
inquiring to learn about the technology	inquiring to learn about the technology, its settings or features, and/or how to use them
inquiring to check comprehension	inquiring to check if listener comprehends what speaker said, did, or saw/sees
inquiring to gain permission	inquiring to gain permission to do something at that moment during the interaction
stating needs	stating needs for the technology's settings/features or for the session's procedures
giving background information	giving background information about the problem or question brought up in that macrolevel unit or session
confirming or denying	confirming or denying what listener or speaker said, did, or asked with a yes- or no-type answer, an I-don't-know-type answer, or a noncommittal answer
declaring the problem or problems as solved	declaring a problem as solved or a question answered
judging the technology	judging the technology and/or its features
observing	describing what speaker sees, hears, or notices while using or observing the technology at that moment during the session
speculating	speculating about a problem or question brought up in that macrolevel unit or session
signaling	signaling what speaker is doing at that moment or what speaker will do next
planning	planning what to do either within or after the session
showing how the technology works or how to do something with it	showing listener how the technology works or how to do something with it by using the technology itself
explaining how the technology works or how to do something with it	explaining to listener how the technology works or how to do something with it without using the technology itself
telling	telling listener what to do at that moment in the session