The development of a facet analysis system to code and analyze data in a mixed-method study is discussed. The research goal was to identify the dimensions of interaction that contribute to student satisfaction in online Web-supported courses. The study was conducted between 2000 and 2002 at the Florida State University School of Information Studies. The researchers developed a facet analysis system that meets S. R. Ranganathan’s (1967) requirements for articulation on three planes (idea, verbal, and notational). This system includes a codebook (verbal), coding procedures, and formulae (notational) for quantitative analysis of logs of chat sessions and postings to discussion boards for eight master’s level courses taught online during the fall 2000 semester. Focus group interviews were subsequently held with student participants to confirm that results of the facet analysis reflected their experiences with the courses. The system was developed through a process of emergent coding. The researchers have been unable to identify any prior use of facet analysis for the analysis of research data as in this study. Identifying the facet analysis system was a major breakthrough in the research process, which, in turn, provided the researchers with a lens through which to analyze and interpret the data. In addition, identification of the faceted nature of the system opens up new possibilities for automation of the coding process.

In this article, we discuss the development of a facet analysis system used to code and analyze data in a mixed-method study. A review of the literature indicates that this represents the first use of facet analysis to analyze research data. The goal of the study was to identify the dimensions of interaction that contribute to student satisfaction in online Web-supported courses. Student success was not employed as a measure in this study for reasons discussed below.

The study was conducted between 2000 and 2002 at the Florida State School of Information Studies in Tallahassee, FL (now the College of Information). The researchers developed a facet analysis system that meets the requirements of S. R. Ranganathan, known for his highly influential work in library classification, for articulation on three planes (idea, verbal, and notational) as described in his *Prolegomena to Library Classification* (Ranganathan, 1967). This system includes facets derived from the abstraction of the content and participants of chat sessions via time-stamped log files (idea), a codebook (verbal), coding procedures, and formulae (notational) for quantitative analysis of logs of chat sessions and postings to discussion boards for eight master’s level courses taught online during the fall 2000 semester. Focus group interviews were subsequently held with student participants to confirm that results of the facet analysis reflected their experiences with the courses.

**Purpose and Significance of the Study**

The purpose of the research was to increase understanding and awareness of the dimensions of interaction that contribute
positively to student satisfaction in online, Web-supported master’s courses in library and information studies. Student satisfaction, for which course evaluation and focus group data were available to the researchers, was used in this study rather than student success. Identification of these dimensions may help Library and Information Science (LIS) teachers and other educators to design, develop, and implement online courses that are more effective in their support of learning and more satisfying to students and instructors alike. Library and Information Science programs may find the facet analysis system that is described in this article useful in the assessment of interaction in their online courses.

Background

Interaction in Online Learning Prior to 2000

The research literature across disciplines at the time that data collection for this study began in 2000 included little discussion of interaction in online learning. One conference paper (Baehr, 1995) and an article (Gilbert & Moore, 1998) discussed development and use of technology to facilitate interaction between teachers and learners. In 1995, Davies discussed techniques to coordinate interaction in an online course on computer networking (Davies, 1995). Garcia and Jacobs (1998) conducted a pilot study of computer-mediated quasi-synchronous classroom discussions. They video-taped three undergraduate students using computers in a classroom setting, and utilized ethnomethodological conversation analysis to discover how participants coordinated their actions and imported conventional procedures from oral conversation. Several distance education critics, most notably Postman (Postman, 1993, 1995), argued that successful learning could not take place online because mediation by a computer degraded the quality of the interaction between an instructor and a student. In research with goals similar to those discussed in this article, Sherry, Fulford and Zhang (1998) used a quantitative measure to assess interaction and a qualitative formative evaluation approach in which a facilitator gathered data about instruction from students using an instrument called the Small Group Instructional Diagnosis (SGID) to assess student satisfaction. This approach was not adopted in the research described in this article because of its limitations in representing relationships between content, actions and actors, as well as its lack of recognition of the multidimensionality of interaction in online learning.

Interest in interaction in online learning grew rapidly after 2000 as evidenced in the literature. Some researchers focused on the development of tools and interfaces to support synchronous interaction online (Donath & Viegas, 2002; Smith, Cadiz, & Burkhalter, 2000); while others discussed the importance of interaction to pedagogy (Aggarwal & Bento, 2000; Hentea, Shea, & Pennington, 2003; Wright, Sunal, & Wilson, 2006; Yanes, Peña, & Curtis, 2006). Im and Lee (2003/2004) conducted research involving the use of online discussion forums. They analyzed the interaction patterns of 40 preservice teachers, and found that the synchronous discussion forums were used more heavily to establish social bonds, whereas asynchronous discussions forums were used mostly for task-oriented activities. Herron and Wright (2006) discussed the importance of including interaction in assessment of learning. Yet another group (to which our research belongs) began to develop methods and frameworks for conducting research. Jeong (2003) developed and tested a software tool to support sequential analysis (a form of content analysis) of student postings to threaded discussion boards, and later proposed methods and a framework for evaluating, modeling, and predicting group interactions in computer-mediated communication (Jeong, 2005). A review of the literature revealed no instances of the application of facet analysis to research on interaction in online learning.

Facet Analysis

S. R. Ranganathan is generally credited with conceiving facet analysis in the 1930s, although Henry Bliss, Paul Otlet, and Julius Kaiser developed similar analytico-synthetic approaches to subject classification and indexing (Broughton, 2002). Ranganathan proposed that facet analysis be conducted on three planes: the idea plane, the verbal plane, and the notational plane, and developed canons, postulates, and principles to support the implementation of this analysis for the better expression of compound subjects (Spiteri, 1998). “Modern facet analytical theory contrasts with earlier views of knowledge as an integral whole (which is broken down into smaller and smaller units) in that it deals with individual terms or concepts which are clustered into categories to create a ‘bottom-up’ map of knowledge” (Broughton, 2002). The current study extends the application of facet analysis from its predominant contemporary use (to support the physical arrangement of documents) to develop a specialized content analysis system capable of representing the multidimensionality of relationships in online interaction.

Spiteri’s (1998) simplified model for facet analysis was used to guide development of the system, which was built from individual terms that were analyzed into groups within facets and ordered by the application of the system syntax through formulae. The formulae, which operationalized the relationships between the facets as dimensions of interaction, were developed to combine the terms and concepts and generate a three-dimensional structure, or FIT (frequency, intensity, and topicality), which is representative of the complexity of interaction in online learning.

In the next section, we provide an overview of the research design to facilitate understanding of the context in which the facet analysis system was developed and implemented.

Overview of Research Design

The study was conducted in three phases—including two data collection phases and one data analysis and interpretation
Faculty Office (FO) or the Technical Office (TO) were available on a 24-hour basis; students could post questions at any time of the day. Instructors or teaching assistants answered FO questions as time permitted; the TO was a cross-course discussion board that was centrally managed by the technical staff at the School of Information Studies. Synchronous components consisted mainly of the interactive chat (I-chat) sessions, which took place at weekly 2-hour real-time sessions, unless otherwise specified by the instructor. A logfile of each session was automatically recorded. Every action within each session was identified by the participant’s name and a time-stamp consisting of date and time. The researchers obtained electronic files for all asynchronous and synchronous components of each course and developed a replicable facet analysis system through the process of emergent coding. Manual coding of all the interactions that appeared in the logfiles and discussion board postings for eight courses offered in fall 2000 was performed, and formulae were developed and applied for analysis.

**Documentary Evidence**

Online Web-supported courses at the School of Information Studies had at the time of this study both synchronous (real time) and asynchronous components. Asynchronous components consisted of discussion boards, both for the exclusive use of students for learner–learner interaction and for the use of students and instructors for learner–instructor interaction. Asynchronous discussion boards such as the Faculty Office (FO) or the Technical Office (TO) were available on a 24-hour basis; students could post questions at any time of the day. Instructors or teaching assistants answered FO questions as time permitted; the TO was a cross-course discussion board that was centrally managed by the technical staff at the School of Information Studies. Synchronous components consisted mainly of the interactive chat (I-chat) sessions, which took place at weekly 2-hour real-time sessions, unless otherwise specified by the instructor. A logfile of each session was automatically recorded. Every action within each session was identified by the participant’s name and a time-stamp consisting of date and time. The researchers obtained electronic files for all asynchronous and synchronous components of each course and developed a replicable facet analysis system through the process of emergent coding. Manual coding of all the interactions that appeared in the logfiles and discussion board postings for eight courses offered in fall 2000 was performed, and formulae were developed and applied for analysis.

![Research design diagram](image-url)
Focus Group Interviews

To develop the focus group interview questions, the research team analyzed the gaps and anomalies that had emerged during the examination of the documentary evidence from both the main study and the pilot study. Two focus group sessions were held, one face-to-face (F2F) session and one online chat session. The F2F session was tape-recorded and transcriptions prepared. Logfiles were obtained for the online chat session. The focus group interviews were then analyzed using the established content analysis technique of meaning condensation.

Summary of Research Results

Three dimensions of interaction were identified and explored: frequency, intensity, and topicality (FIT). In this study, frequency was defined as the mean number of interactions per student enrolled occurring each weekday during an academic term. Intensity was defined as the total number of statements generated by students as compared to the total number of statements generated by the instructor(s) or teaching assistant(s). Topicality was defined as the total number of statements on or related to the topic or activity content of the course during the week the log and discussion postings represented, as compared to the total number of statements on or about activity processes, and the total number of ancillary statements, including nonrelevant statements and nonverbal statements. One course received a FIT ranking of high, three received a FIT ranking of low, and the remaining four received a FIT ranking of medium. A course was ranked high if it was above the mean in at least two dimensions, and was ranked low if it was below the mean in at least two dimensions. A course ranked medium if it was at or near the mean in at least two dimensions, or if it received one low, one medium, and one high ranking. (A more detailed account of the results of this study will appear in a 2007 special issue of the Journal of Library and Information Science Education.)

Development of the Facet Analysis System

Prior to identifying facet analysis as a candidate for this project, the researchers performed a needs analysis of existing software applications and content analysis methods to determine whether any existed with the capacity to both capture the character and intention of the interactions and relate the statements to participant categories. Satisfied that none were available, emergent coding was applied to explore development of a system that would meet these requirements.

Idea Plane: Identifying Roles and Functions

First, the coding development team read chat logfiles of courses taught during the summer 2000 from start to finish to discover the essential components of each logfile and to identify elements common to all the logfiles. Summer 2000 data were used to pilot the study and to ensure that the analysis system was developed with data similar to the study data. In this “from the bottom up” process, a simple abstract was composed for each logfile (see Figure 2). The abstracts summarized the content of the interactions, the actions they entailed or referred to, and included notes about the participant roles as related to the statements. Each abstract also included a list of key words. The team then discussed the abstracts and developed the core components of the analysis system over the course of several meetings.

At an early stage in this emergent coding process, the team identified two categories of terms within each logfile: general terms and course-specific terms. For example, in the course on information organization the terms information bearing entity, representation, and surrogates were identified as course-specific. Terms such as calendar, assignments, readings, URL, I-chat (or chat), and questions were identified as general terms. Identifying synonyms was also a concern and was dealt with on a case-by-case basis.

Next, the team identified concepts and actions shared across logfiles and compared these to those found in the discussion board postings for the same week. Concern was expressed that the inability to represent the content at a high-level of specificity would be a major drawback in the analysis. It was at this point that the development team first began to think about a facet type of analysis, although they did not identify it as such. The decision was made to code not for the content itself, but for how that content functioned in the context of the interactions. For example, rather than coding the specific topic of an interaction, an interaction focused on the course content for the week as listed in the syllabus would be coded.
“topic.” An interaction about the process of a course activity would be coded “activity process,” whereas an interaction about the content of a course activity would be coded “activity content.” Focusing on categorical classification by function rather than subject analysis of the content gave rise to the identification of two facets: content and activity.

It was also necessary to consider how the researchers would use the coded logfiles and discussion board postings to measure the concepts that the team identified. The researchers had expressed an interest in understanding the role that communication formats played in interaction, so the development team examined the data by format. Commonalities were found in the content across the synchronous and asynchronous files. For example, both logfiles and public board postings included discussion of articles, class topics, and activities, such as short exercises, homework, or particular assignments. There were also other differences: instructor-led lectures were found in the chat logfiles, whereas student-initiated Q&As was more common in the discussion board postings. Differences in the tone of interaction were also apparent: chit-chat, peer-support, and hall-talk (i.e., talking before and after a chat session formally began) were frequent in the chat logfiles, whereas a more formal, structured, and on-topic tone prevailed in the discussion board postings. This confirmed the need for a format facet.

Once the format facet was established, the team considered how to relate the interactions by format to the participants. They tried preparing a list of names found in each format, followed by the number of interaction statements for each name. In addition, each interaction was identified as facilitator–student, student–student, mixed, etc., and the number of occurrences of each type was noted. Although the specifics of this strategy were ultimately abandoned as cumbersome and nonproductive, examining the data in this way led to the development of the last two facets: action and actor.

In summary, a bottom-up process of emergent coding was used with pilot data to identify the structure of the analysis system. A system consisting of five facets was developed. The facets included format, actor, action, content, and activity. The next task for the development team was to develop the codebook, which is discussed in the following section.

Verbal Plane: Defining Facets

Ranganathan (1967) talks of the “capacity to develop an articulate language as medium for communication” when he describes his notion of the verbal plane (p. 327). Similarly, Spiteri (1998) describes it as consisting of the process of choosing appropriate terminology to express the component parts of the analysis system. Once the development team had decided which facets would make up the analysis system, they began to work on the terminology that would express its component parts, and to compile that terminology in a codebook. The codebook therefore contains a record of the terminology and definitions that were applied for each facet. It also includes codes and instructions for implementing those codes (the notational system), which is discussed in the section on the notational plane. In this section, we will discuss each facet and define the terms that were developed to represent that facet.

Terminology by facet. The format facet was the simplest in its structure, and the easiest for the coders to identify. It is a simple binary: asynchronous/synchronous. Any interaction that occurred in the log of an interactive chat (I-chat) session was coded as synchronous. Any posting to the faculty office discussion board was coded as asynchronous.

The actor facet was used to represent the participants in an interaction. Participants were coded by role in the course, and included: instructor, student, teaching assistants, help-staff, guests, and mentors. Class rolls were used to verify participant roles for instructors, students, and teaching assistants. Help staff were identified by their login, which contained “help” in its label. Guests and mentors were identified by consulting the instructors or teaching assistants assigned to the course.

The action facet was developed to represent the type and intent of each interaction statement. The development team initially identified two statement intentions: begin and respond. Statements that represented the intent to begin an interaction included questions, directives to start an interaction, and redirection statements. Statements, which represented the intent to respond, included responding to a previous statement, and verbal and nonverbal statements. A third intention was added in the second version of the code after the initial test coding demonstrated that this type of statement often signaled a change in the direction of the interaction that might affect the FIT of what followed, especially during intense lectures and Q&A sessions. Transition statements most often included two types of interactions in a single statement (e.g., instructor confirms a correct answer, then asks another question), or otherwise signaled a transition from one topic or activity to another. As coding proceeded, the team also realized they had no way to code nonverbal statements. These statements resulted from the particular chat software used. All text messages were identified as “public message” by the software. However, participants had the option of replying with an “action” by using certain command symbols (specifically, a colon before the text) that would create a “public message” to an “action.” For example, instead of Jane Doe typing “Yes, I understand” in response to a query she could create her own action message, such as: “Jane Doe gives the thumbs up.” The action messages usually represented body language or emotional states and appeared in pink text, allowing for quick identification. Because coding was already underway and redefinition of a facet would have meant reordering the notational system and recoding those logs and posts already coded, the team decided that it would be expedient to attach the nonverbal code to the end of the existing notational system. Properly, however, nonverbal statements should be considered part of the action facet.
The content and activity facets presented the greatest challenges in development of the code. In a faceted classification system the classes (facets), should not share concepts or entities. Ranganathan (1967) defined this as mutual exclusivity in his Canon of Exclusiveness. However it was difficult to maintain exclusivity between the two facets due to conceptual overlaps and because the development team found it necessary to alter the topic facet after the coding of the study data was already underway. As with the format facet, the content facet was initially conceived of and implemented as a simple binary: on-topic/not on-topic. A statement was considered on-topic if it discussed the content represented in the course syllabus for that week. A statement was considered not on-topic if the content was not related to the topic of the course discussion for that week as listed in the syllabus, and/or if the content of the statement related to an activity such as an assignment. At the same time that the development team noticed the inability to code nonverbal statements, they realized that it might be useful to represent a specific type of not on-topic statements separately. Nonrelevant statements consisted of social chit-chat that would normally be confined to the hallway in a traditional university brick-and-mortar setting. To distinguish these not on-topic statements—which rarely occurred except at the beginning and end of a chat session—from others that tended to be sprinkled throughout the session, the term nonrelevant was added. Although these side conversations were interesting from a social perspective they did not occur frequently enough to be implemented as a part of the facet in the study. Instead, nonrelevant statements were coded as miscellaneous elements and tacked on to the end of the notational system as were nonverbal statements.

The activity facet was a binary: activity-content/activity-process. A statement was coded as activity-content if it dealt with the content of an upcoming assignment or other course-related activity; it was coded as activity-process if it dealt with the process of an upcoming assignment or course-related activity (e.g., date due, length, etc.). If a statement was coded on-topic (content facet), the activity facet was not applied.

Notational Plane: Portraying Facets

Spiteri (1998) describes the notational plane as the part of the work that expresses the component parts developed in the verbal plan by means of a notational device. Ranganathan (1967) wrote that in the notational plane “words are often replaced by symbols pregnant with precise meaning” (p. 327). In essence, the use of symbols, most often ordinal numbers (as used in classification systems such as the Dewey Decimal Classification) clears away imprecision brought about by the inconsistency of language (e.g., synonyms and homonyms). A codebook was used in this study to record the three aspects of the notational plane for our research. The codebook has three sections: (a) the notations (or codes) used to represent the terminology for each facet, (b) the rules for applying and ordering those codes, or the notational system, and (c) the

<table>
<thead>
<tr>
<th>Facet</th>
<th>Terms</th>
<th>Codes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format</td>
<td>Synchronous</td>
<td>S</td>
</tr>
<tr>
<td></td>
<td>Asynchronous</td>
<td>A</td>
</tr>
<tr>
<td>Actor</td>
<td>Instructor</td>
<td>In</td>
</tr>
<tr>
<td></td>
<td>Student</td>
<td>St</td>
</tr>
<tr>
<td></td>
<td>Teaching Assistant</td>
<td>Ta</td>
</tr>
<tr>
<td></td>
<td>Help-staff</td>
<td>He</td>
</tr>
<tr>
<td></td>
<td>Guest</td>
<td>Gu</td>
</tr>
<tr>
<td></td>
<td>Mentor</td>
<td>Mt</td>
</tr>
<tr>
<td>Action</td>
<td>Begin</td>
<td>B</td>
</tr>
<tr>
<td></td>
<td>Respond</td>
<td>R</td>
</tr>
<tr>
<td></td>
<td>Transition</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>Nonverbal</td>
<td>Nv</td>
</tr>
<tr>
<td>Topic</td>
<td>On-topic</td>
<td>T</td>
</tr>
<tr>
<td></td>
<td>Not on-topic</td>
<td>[no code = not on-topic]</td>
</tr>
<tr>
<td></td>
<td>Nonrelevant</td>
<td>Nr</td>
</tr>
<tr>
<td>Activity</td>
<td>Activity-content</td>
<td>C</td>
</tr>
<tr>
<td></td>
<td>Activity-process</td>
<td>P</td>
</tr>
</tbody>
</table>

The code. Once the terminology had been settled on, the development team began working on the code, or notations, which would be used to implement the system. To reduce complexity and increase reliability, it was decided to make the codes short (one and two characters) and mnemonic rather than numeric. Table 1 displays the facets, terms, and corresponding codes.

The notational system. The notational system was designed to represent the data contained in synchronous and asynchronous formatted documents with equal complexity and accuracy, including over 100 viable permutations. To aid in analysis and interpretation of the coded data, as well as to improve reliability of the coding itself, an order of facets was established and strictly enforced. First, each coder was instructed to identify and note the format, actor, action, topic, and then the activity. As noted earlier, in this particular implementation, if either of the two miscellaneous codes (Nr or Nv) were applicable, they were appended at the end of a notational string. For expediency, all viable permutations were constructed and defined in advance and given to the coders in simple Word document tables. These tables constituted the codebook at this stage of the research. Coders were expected to code each facet independently, create the notational string, then check that notational string against the appropriate line in the codebook to ascertain that the order of facets had been implemented correctly, the codes well-formed, and that the interaction statement matched that of the definition accompanying the string in the codebook. A sample section from the codebook appears in Table 2.

Figure 3 shows the coded logfile of 3 minutes of a 2-hour chat session held on June 14, 2000. Chat sessions were not open-forum, were typically focused on a topic, and the
conversation was guided by the instructor. In these 3 minutes several questions and answers pass between the instructor (DM) and the students. In this example logfile, the notational string appears at the left, followed by the date-time stamp, actor’s initials (KS, BC, DM, etc.), message type (public or action), and the text of the message. This was considered the line and each instance was coded separately. The first line in Figure 3, is coded SStBT (see Table 1 for code facets and semantics). The first letter signifies a synchronous (real time) activity (S). The coder then identified the actor as a student (St) who began a new interaction (B) on the topic of the session (T), which in this particular session is about database field definition. This line and the three lines following it are students asking questions about the topic. The fifth line, SInRT, is the response of instructor to those questions, i.e., in a synchronous format (S), the instructor (In) responds (R) to the questions about the lesson topic (T). Again, the actual topic of the lesson is not coded; only the activity of discussing the topic is noted. In this sense, it is not content analysis so much as it is an interactivity analysis.

**Coding procedures and problems.** The actual coding proved to be tedious and time consuming. Initially, logfiles and discussion board postings were printed out and each line coded by hand. A typical log file could run anywhere from 20 to 30 pages and the coding, tallying of those codes, and then transferring those tallies to a spreadsheet was excessively time consuming. Accordingly, the coding team looked for ways to speed up and simplify the process. The most expedient solution, and the one adopted at the time, was to

**TABLE 2. Codebook sample.**

<table>
<thead>
<tr>
<th>Code</th>
<th>Semantic</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASBP</td>
<td>Asynchronous student beginning an activity process</td>
</tr>
<tr>
<td>ASRP</td>
<td>Asynchronous student responding to a statement about activity process</td>
</tr>
<tr>
<td>ASBT</td>
<td>Asynchronous student beginning an interaction about a topic</td>
</tr>
<tr>
<td>ASRT</td>
<td>Asynchronous student responding to a statement about a topic</td>
</tr>
<tr>
<td>ASBC</td>
<td>Asynchronous student beginning activity content</td>
</tr>
<tr>
<td>ASRC</td>
<td>Asynchronous student responding to a statement about activity content</td>
</tr>
<tr>
<td>SSBP</td>
<td>Synchronous student beginning an activity process</td>
</tr>
<tr>
<td>SSRP</td>
<td>Synchronous student responding to a statement about an activity process</td>
</tr>
<tr>
<td>SSBT</td>
<td>Synchronous student beginning an interaction about a topic</td>
</tr>
<tr>
<td>SSRT</td>
<td>Synchronous student responding to a statement about a topic</td>
</tr>
<tr>
<td>SSBC</td>
<td>Synchronous student beginning activity content</td>
</tr>
<tr>
<td>SSRC</td>
<td>Synchronous student responding to a statement about activity content</td>
</tr>
</tbody>
</table>

---

**FIG. 3.** Sample coded logfile (unedited text of a class chat session).
open the logfiles and discussion board postings in an HTML design software program and treat the files as Web pages. The notational strings could then be inserted at the beginning of each line using a red font. When coding was complete, each line of code was counted using the appropriate Find or Word Count function included in the software. This was made possible by having all code combinations created prior to coding. Although the code assignment itself remained a tedious process, it was no longer necessary to manually copy the codes from the printout to a spreadsheet, which both saved time and reduced the opportunity for error. Automating the counting and tallying process further decreased the time required to generate a file for analysis. When the coding for all logfiles and discussion board postings for a course was completed, the results were combined and tallied using a spreadsheet, to which the coded files were linked. The summary spreadsheets became the primary material for analysis, while the linked files provided detail and were used to verify the data when necessary.

Intercoder reliability was verified through the comparison of logfiles coded by several coders. Differences in coding due to different interpretations of content, or interactions, were judged to be minimal and did not affect the validity of the results. In addition, the coding team met often to identify and resolve any such occurrences. This was an important component of the project that aided in fine-tuning the code itself. A second form of validation was serendipitously discovered when electronically tallying the codes. As described above, the Find function in the HTML design software program was used to search for the predetermined permutations of the code. Comparing the total number of lines of text to the total number of groups of codes used for that logfile assured accuracy of coding; a discrepancy in the two totals indicated the presence of a nonvalid code that could then be located and corrected.

**FIT formulae.** The third section of the codebook consisted of the formulae used to calculate the FIT, or frequency, intensity, and topicality dimensions of interaction for the eight courses included in the study. Three formulae were developed:

- **Frequency:** \( F = \frac{\text{total # of statements}}{\text{75 weekdays} \times \text{# of students enrolled}} \)
- **Intensity:** \( I = \frac{\text{total # of St statements}}{\text{total # of In + Ta statements}} \)
- **Topicality:** \( T = \frac{\text{sum of T + C statements}}{\text{sum of P and Nr and Nv statements}} \)

The initial formulae were developed prior to the coding of the data, but underwent changes during its implementation, much the same as the code. The first change was made to the frequency formula at the time of the data analysis. As the researchers graphed the results of the three dimensions, they discovered that the original formula \((F = \text{total # of statements})\) generated numbers that were unaligned with those generated by the intensity and topicality formulae, resulting in display issues. As the researchers worked through this problem, they realized that the total number of statements formula did not take into account two important variables: the number of active course days and the number of students enrolled in the course. Introducing these two variables into the formula made for better representation of the concept of frequency, at the same time resolving the graphing problem. The topicality formula also underwent some change during the analysis process. The original formula was developed prior to the addition of the Nr and Nv codes, which were used to code not on-topic interactions. The formula was therefore adjusted to account for interactions coded Nr or Nv (Figure 4).

The revised formulae were applied to each of the eight courses, and means and medians were calculated. The means and medians were used in the analysis to rate courses as low, medium, or high in each dimension (frequency, intensity, and topicality), and to provide a combined FIT rating.

**Conclusions**

In this article, we discuss the development and implementation of a facet analysis system as applied to the analysis of the dimensions of interaction in online Web-supported distance learning. This represents the first use of facet analysis for the analysis of research data. The faceted system was developed through a process of emergent coding, which did not reach its conclusion until after the data collection and analysis had been completed. Therefore, some aspects of the system, such as the coding of nonrelevant and nonverbal statements within the proper facets, were not fully implemented during the research process. In fact, the analysis system was only distinctly identified as a facet analysis system during the interpretation and writing process that took place 5 years after the data were collected and analyzed. It is unlikely that it would ever have been identified as such except that two of our team members have expertise in information organization. Nonetheless, identifying it as a facet analysis system provided a major breakthrough in the research process, which, in turn, provided us with a lens through which
to analyze and interpret the data. In addition, identification of the faceted nature of the system opens up new possibilities for automation of the coding process.

Further exploration of existing software needs to be conducted to determine what might be available to automate the coding and quantifying processes, to address the time-related issues mentioned above. If this cannot be done with existing software, it should be relatively easy to design and program an application to handle these processes, although differences in format and structure of chat and discussion board software will need to be accounted for in the design. Such an application would make possible the analysis of a wider range of courses across institutions. Ongoing data analysis of online courses conducted by the authors indicates that such an approach can help researchers understand and predict the effect of interaction on student success and satisfaction with online learning (Burnett, Bonnici, Miksa, & Kim, in press). New facets could be developed to account for variables such as pedagogical style, student demographics, technological preparedness, and institutional support to name a few.

References


