The Computer Science Debate: It's a Matter of Perspective

Tracy L. Lewis
Department of Information Technology
Radford University
Radford, Virginia USA
tlewis32@radford.edu

Wanda J. Smith
Pamplin College of Business
Virginia Tech
Blacksburg, Virginia USA
wjsmith@vt.edu

Abstract

This paper examines some of the ongoing identity challenges for the computer science community: definition, purpose, pedagogy, and curriculum drivers. This research contends that there are three primary social theories driving one's perspective of computer science: resource based theory, resource dependency theory, and social exchange theory. A conceptual framework is presented that examines the likely definition, purpose, pedagogy, and curriculum drivers for each perspective. The framework was evaluated in a qualitative study using the e-mails threads from a discussion on the SIGCSE listserv. The results of this study and the implications of future of computer science are discussed.

Keywords: qualitative analysis, resource based theory, resource dependency theory, social exchange theory

1. Introduction

For more than 50 years, computer science (CS) scholars have examined the question: "*what is computer science?*". Unfortunately, the literature has yielded no consensus. More importantly, there appears to be no end to the disparate conceptualizations of computer science as other stakeholders (i.e., CS administrators, professionals, and students considering careers in computing fields) join the debate.

When examining the CS literature, the parable about the blind men and the elephant seems to be a fitting metaphor for the contrasting views of CS. In the parable about the blind men and the elephant, each man, concentrating on the single feature of the animal he touched first, confidently maintains his opinion on the nature of the huge beast. The blind man who touches the elephant's leg is sure it is a tree. The one who grabs the trunk claims with certainty it is a snake. And so it goes: the ear is a fan, and the tail is a rope with a brush on the end. The blind men each studied the elephant but were each limited by their perspectives.

The moral of this parable within the context of CS is that the CS community must look beyond its current perspectives and experiences to conceptualize the bigger picture. To facilitate this shift in perception, theoretically grounded lenses are needed. The intent of this article is to propose three social theories that will serve as lenses through which the CS community may view the nature of CS. In the long run, establishing theoretical frameworks will facilitate the acceptance and development of generally accepted CS definitions, purposes, and curricula. This framework may also help in the design of the ideal pedagogy to be used in CS programs.

2. Theoretical Foundations

By building on the basic premises of social theories (i.e., resource based theory, resource dependency theory and social exchange theory), this paper develops a conceptual framework that identifies three schools of thought within CS. These schools of thought will cover (1) how CS is defined, (2) what is its primary function, (3) how CS is taught (pedagogy), and (3) how the CS curriculum is influenced. This article will also integrate these theories to explore future directions of CS.

We begin our discussion by briefly describing each theory and its application to the context of CS education. Next, we present a conceptual model and a set of propositions that predict how different perspectives are likely to result in different outcomes (i.e., definitions, purpose, pedagogy, and curriculum). We then test our propositions with data from a qualitative study of a recent listserv discussion. In the last section of the article, the implications of the results are discussed together with potential avenues for the future of CS education.

2.1 Resource-Based Theory

The premise of the resource-based theory (RBT) is that a unit's internal strengths or capabilities are its sources of

competitive advantage and, ultimately, its survival. The unit can sustain its competitive advantage by enabling and sustaining product or service "differentiation" in important and durable ways [11].

When applying resource-based theory to the CS debate, we then have a perceptual approach to understanding and predicting one group's view of CS and its strategies for survival. For example, CS product differentiation is the uniqueness of CS students and curriculum. Focusing on training students for graduate school rather than industry builds and maintains a unique body of CS knowledge. This "exclusive but self-sustaining" strategy is reflected in the "many are called, but the chosen are few…" attitude found in some CS camps.

Nevertheless, looking through this theoretical lens turns a blind eye to the emerging complexities and interconnectedness of other computing fields. An alternative lens, the resource-dependency theory, recognizes that all organizations (as well as university departments) are dependent (to some extent) upon the external environment.

2.2 Resource Dependence Theory

Resource dependency theory (RDT) proposes that organizational units are not self-directed or self-dependent [10]. They need resources for their survival. These resources include money, materials, personnel, information and technology. This interdependence may take the form of direct dependence or mutual dependence [10].

In the context of CS education, this theory reveals the underlying assumptions of professionals and scholars who embrace the "service" role of CS within the university community. Persons looking through this lens recognize the direct interdependence of other computing fields and majors. They also, to some extent, acknowledge the indirect influence of industry when developing CS curriculums. Finally, they recognize the importance of creating reciprocal dependencies with applied computing departments that need foundational (non-substitutable) CS knowledge. Yet, the limitation of the resource dependency model is its limited view of the role of perceived social contracts.

2.3 Social Exchange Theory

Similar to resource dependency theory, social exchange theory (SET) [7], assumes organizational units are forums for transactions. The distinction between the two theories is that, according to the social exchange theory, individuals form psychological contracts with these transactions. When individuals or units perceive that their psychological contracts have been breached, they are likely to look for ways to try to restore the benefits they were expecting to receive. They also attempt to protect themselves against future breaches.

When examining the CS literature through this lens, there is growing evidence of a perceived contractual breach, especially from industry and other computing disciplines. For example, the growth of competing computing disciplines occurred, in part, in response to perceived social exchange gaps and violations. CS scholars and practitioners who embrace a SET lens recognize the importance of communication (the informal sharing of meaningful and timely information), and opportunistic behavior (maximizing self-interest with guile). This school of thought would also promote non-opportunistic partnerships among critical CS and non-CS stakeholders. The added distinction of this theory is it seeks to fulfill industry's psychological contact (expectations).

3. Conceptual Framework

A conceptual framework was developed using the social theories described above. As noted in Table 1, each theory shapes (1) the definition of CS, (2) the purpose it serves in the university and society, as well as (3) which stakeholders should be involved in the development of the CS curriculum. A brief summary is described below.

Table 1
Conceptual Framework of SIGCSE Conversation on "What is CS?"

Social Theories	Competitive Advantage Assumptions	Define & Conceptualize CS as…	Primary Function of CS	Drivers of Curriculum Development
Resource Based Theory (RBT)	Product differentiation	… a science grounded in theory: (algorithms & problem solving techniques)	Train CS students to maintain CS pipeline	Internal CS Stakeholders
Resource Dependency Theory (RDT)	Strategic inter-dependence	… a science & an art (i.e., application of algorithms & problem solving	Service the Computing Pipeline	Internal Computing Stakeholders
Social Exchange Theory (SET)	Expand networks and maintain psychological contracts	… an all inclusive field of computing & beyond	Service the University Community & beyond	Internal and External Stakeholders

Resource-based theory focuses on student and curriculum differentiation. Individuals with this lens would likely promote strategy of distinction and segregation. According to this theory, CS departments will primarily focus on training students to maintain the CS community. In contrast, resource dependence theory considers the interconnectedness of the computing pipeline. Advocates of this theory would define CS more broadly and seek to service CS majors as well as majors within the applied fields (software engineering). Lastly, when using the social exchange theory, CS departments would try to assist service disciplines (math, chemistry, & physics) and beyond (art, sociology, history, etc.) Finally, these theories can also be used to predict who the dominant drivers would

be in the development of CS curriculum. Consistent with each theory, key decision makers will range from a closed shop to a process open to input from industry and other stakeholders.

This proposed framework is used to explore recent SIGCSE listserv conversations responding to the question: *what is computer science*? The method used to accomplish this is described below.

4. Methods

4.1 Population and Time Frame

This research was motivated by the plethora of e-mails generated on the SIGCSE listserv after the release of the draft of the Computing Curriculum 2004. The topics related to "*what is computer science*?" were discussed for a period of three weeks during January, 2005, resulting in 117 relevant e-mails and 46 respondents. In order to assure that all e-mails were obtained, the researchers consulted the ACM's online public archive of listserv postings.

4.2 Data Collection

Shortly after the e-mail conversations surrounding this topic ceased, the researchers posted an e-mail to the listserv reporting the intentions of this study.

In order to maintain the anonymity of individuals involved in the e-mail dialogue, many of the comments are paraphrased. This is consistent with the research of Sharf [12]. Sharf suggests that when qualitative data analysis is based on the public-accessible information, the researchers should use limited direct quotations. While we paraphrased comments, efforts were made to maintain content validity of all comments.

4.3 Data Analysis

The grounded theory approach to qualitative analysis was used within this study. The use of the grounded theory approach involved working with data, organizing it, breaking it into manageable units, synthesizing it, searching for patterns, discovering what was important (extraction) and what was to be learned (interpretation), and deciding what to report to others [13].

The first step in this process was to perform open coding of the data. During open coding, the goal was to create descriptive, multi-dimensional categories which formed a preliminary framework for analysis. Words, phrases, or events that appeared to be similar were grouped into the same category. These categories were gradually modified or replaced during the subsequent stages of analysis that follows.

The data used in this study was retrieved from open-ended dialogues, so all qualitative analysis was conducted by hand (by the study researchers). The researchers went through several iterations of careful data classifications and reclassifications before settling on the final placements.

Finally, to position the emerging themes found in the data within the larger trends in the CS and computing literature, the researchers consulted numerous sources of CS historical and contextual material [1,2,3,4,5,6,8,9].

5. Qualitative Results

While there were nearly 60 individuals who contributed to the *"what is computer science?"* threads on the SIGCSE listserv, only comments from 46 people were used in the current study. Comments and individuals were excluded from analysis for several reasons: (1) individuals posed or replied to comments not related to the subject of the e-mail, (2) the view(s) of the individual was significantly divergent from other respondents and was not corroborated or questioned by others, or (3) individuals simply asked a question as a point of clarification. Further, of the 46 listserv participants used in the study, only 41 individuals were used in the actual data analysis; the remaining 5 individuals provided valuable insight on the history and future of CS.

5.1 Descriptive Statistics

Consistent with the three social theories (resource based, resource dependent, and social exchange), the terms segregationist, integrationist, and synergists are respectively used as descriptors to convey how society commonly views these profiles.

Using thematic coding for the segregationist, integrationist, and synergist perspectives, it was found that 32% (n = 13) of the participants defined CS according to the segregationist perspective, 24% (n = 10) according to the integrationist perspective, and 32% (n = 13) according to the synergist perspective.

It is worth noting that because of the unstructured format of the data, several individuals expressed a clear perspective in one category, while not addressing other categories. For example, three individuals counted in the synergistic pedagogical preference (Instruction & Construction), did not explicitly express their views on other issues (i.e., the definition of CS, the function of CS, or the drivers of the curriculum).

In addition to the definition, purpose and curriculum drivers, each perspective of CS is hypothesized to approach pedagogy in one of three ways: instruction, construction, and a combination approach. With traditional instruction (preferred by segregationist), the process of learning is didactic and centered on the teacher. One to opposite end of the continuum is the construction pedagogical approach. The construction pedagogical approach (preferred by integrationist) is centered on the student and is a collaborative learning process. And then there is the instruction and construction combined approach (preferred by synergist); which requires multiple methods of instruction to meet the individual learning styles of the students.

The following sections will provide a brief overview of the results of this study. Table 2 summarizes the extent to which the data supported the proposed theories. For more detailed information, please contact the researchers at tlewis32@radford.edu.

5.2 The Segregationist Perspective

The data clearly supported all categories of the hypothesized perception of the segregationist. For example, individuals who expressed segregationist views define CS in terms of algorithmic analysis and development. They also reported the need to focus on CS theory building, an *instruction* teaching methodology, and limiting curriculum development to internal stakeholders (CS faculty and administrators).

Some of the paraphrased supporting comments for this perspective were: *…so why are we letting [insert industry/discipline here] define our field* and *computer science is not a job training discipline…; in essence, computer science is a science of algorithms – designed to advance knowledge.* Others in this category declared *…computer science is not an art, it is problem solving, algorithmic analysis, and theory building;* and, *only a small percentage of all people think algorithmically*.

5.3 The Integrationist Perspective

Comments supporting the integrationist perspective were: *computer science is driven by what we do for others (computing fields)*, *courses in computer science should reflect the need of the computing disciplines more than industry desires*, and *teach what is needed by the computing masses.* Given these comments, the premise that integrationists would invite other disciplines to the table, when developing curriculum, was supported. Surprisingly, the anticipated integrationist's perspective of the service function of CS and "construction" pedagogical approach of CS were not supported by the current data.

5.4 The Synergist Perspective

Interesting comments were found to support the synergist perspective: *computer science is all the above and more… avoid the strict separation models*, *computer science is defined by its ability to be flexible to the demands of all*, and *computer science is defined by its diversity and constant innovation*. While a significant number of participants (13 out of 46) held views classified according to the synergist, all inclusive philosophy, many of these individuals did not provide explicit comments regarding CS pedagogy, curriculum, and purpose.

5.5 Summary

The e-mail conversations used within this study provided a wealth of historical facts, quotes, and varying opinions. During iterations of the qualitative analysis, it became clear that the segregationist voice was the loudest in the conversation. Individuals classified as preferring the segregationist perspective supported their opinion on the definition of CS by providing additional details related to the purpose, pedagogical approach, and primary drivers of the CS curriculum. Table 2 summarizes the results of this study.

While the segregationist perspective may not be the overall view of the CS education community, it is certainly clear that this view dominated the "*what is computer science?*" SIGCSE listserv conversations. The implications of these findings are discussed in the remaining sections of this paper.

Table 2 Summary of data support of the conceptual framework

Perspectives (CS Defined)	Function (Purpose)	Pedagogy	Drivers of Curriculum Development
Segregationist (Theory) n = 13	Theory Building **(supported)**	Instruction **(supported)**	Internal CS Stakeholders **(supported)**
Integrationist Distinct (Theory & Application) n = 10	Service Computing Pipeline (not supported)	Construction (not supported)	Internal Computing Stakeholders **(supported)**
Synergists All Inclusive (Theory & Application) n = 13	Service University Community (not supported)	Instruction & Construction (not supported)	Internal and External Stakeholders (not supported)

6. Discussion

This study is a first step in understanding the differing perspectives of CS influence critical decisions. As noted above, one third of our conservationists held segregationist views (i.e., a resource based perspective). According to the resource based theory, a department's survival is determined by its value (strategic importance) and uniqueness. Unfortunately, CS departments are no longer the only source of computing education. Since the emergence of information systems, information technology, and software engineering, students, parents, and industry often do not see the uniqueness of a CS education. Additionally, the CS competitive advantage is further weakened by industry's perception of the declining value of a CS education.

Holding on to segregationist (resource-based) beliefs such as *"only a small percentage of all people think algorithmically"* and *"there is something psychologically special about CS as compared to all other design and production disciplines"* is a path of impending doom for CS departments, if their customers do not see its relative value. While segregationist reported being "*bonded by a love of a certain kind of problem solving",* they can no longer afford to be *blinded* by that love.

Similarly, integrationist (a resource dependency view) may, in the long term, be disillusioned. This theory suggests that a department's survival is dependent on its

ability to develop strategically important interdependencies with other computing disciplines. These educators embrace the notion that "*computer science provides the foundation and theoretical training of a more specialized computing skill*". Yet, current CS enrollments do not reflect these beliefs. In many cases, these departments have developed "specialized" (according to some…watered down) CS fundamental courses. CS departments who have designed courses that "*teach what is needed by the computing masses*", may survive the increasing competition, but this will not ensure their long-term survival.

According to our framework, the future of CS lies in its willingness to embrace a synergistic (social exchange) perspective. Unfortunately, our results suggest that the conversationalists were not ready to operationalize this view. While 13 individuals expressed synergistic views (i.e., CS is defined by its ability to be flexible to the demands of all), few discussed how CS could partner with its many stakeholders and service all majors. And, only one respondent charged CS educators with the task of "*working with industry*" and "*preparing students for practical jobs*". Nevertheless, the synergist "*sign of the times*" paradigm must be embraced and operationalized by CS educators. Industry and other stakeholders are developing dynamic and demanding psychological contracts (expectations). Creating customer-oriented relationships with all of our stakeholders (i.e., students, parents, graduate schools, industry, and other university departments) and responding to their needs will ensure that CS will not only survive, but thrive.

7. Future Work

This framework, then, is useful not only for increasing awareness of the various perspectives of CS, but it can assist educators with attempts to influence the future of CS as well. For example, it is useful for identifying opportunities for collaborative and cooperative activities, such as joint curriculum development, partnering with industry, in addition to identifying and monitoring threats to CS existence. This framework can also be used to view competing disciplines and to track potential competitors' growth over time. Finally, this framework may provide a forewarning of impending doom if CS academics continue to embrace the segregationist perspective.

We suggest further exploration along these lines using a variety of qualitative and quantitative methods for several reasons. First, generalizability may be an issue because we cannot be certain that the listserv participants represent the perspectives of the general computing population. Many propositions were commented on by less than half of the conversationalists. Because this reduces the sample size even more, the conclusions we have drawn about these propositions may be invalid. Further, having only two researchers conduct the qualitative analysis also calls into question the interpretation of the data. Two more researchers should be asked to code the interview data to assure multiple perspectives on the interpretation of the data. These issues will be addressed prior to trying to conducting further research.

Acknowledgements

A special thank you to the ACM SIGCSE listserv participants for allowing us to use your e-mail conversations in this study.

References

[1] Abelson H., Sussman, G., Sussman, J. *Structure and Interpretation of Computer Programs*. MIT Press, 1996.
[2] Denning, P. J. et al. (1989) Computing as a discipline. *Communications of the ACM*, 32, 1 (Jan.), 9-23. Computer 22, 2 (Feb.), 63-70.
[3] Denning, P. J. Great Principles of Computing, *Communications of the ACM,* Vol.46, No.11 (Nov. 2003) pp. 15-20.
[4] Dijkstra, E. W. The End of Computing Science?, *Communications of the ACM*, March 2001. p. 92.
[5] Domik, G. Glimspes into the Future of Computer Science Education. *Journal of Universal Computer Science*, vol. 7, no. 5, 2001.
[6] Hamming, R.W. One Man's View of Computer Science, *Journal of the ACM (JACM)*, v.16 n.1, p.3-12, Jan. 1969.
[7] Kelley, H. H. *Personal relationships. Their structures and processes*. Hillsdale, NJ: Erlbaum, 1979.
[8] McGuffee, James W. Defining Computer Science, *SIGCSE Bulletin -inroads*, Volume 32, Number 2, June 2000.
[9] Nahapetian, A. Computer science (subjectively) defined, *ACM SIGCSE Bulletin*, v.11 n.4, p.11-14, December 1979.
[10] Pugh, D. S. and D.J. Hickson. *Writers on Organization* (5th ed.). Thousand Oaks: SAGE Publications, 1997.
[11] Rynes, S L, J M Bartunek, R L Daft. Across the great divide: knowledge creation and transfer between practitioners and academics, *Academy of Management Journal*, 44, 2 (2001), 340-355.
[12] Sharf, B. Beyond netiquette: The ethics of doing naturalistic research on the Internet. In S. Jones (ed.) *Doing Internet Research.* Thousand Oaks, CA, and London: Sage, 1999.
[13] Strauss, A.L., & Corbin, J. Grounded Theory Methodology: An overview. In N.K. Denzin & Y.S. Lincoln (Eds.), *Handbook of qualitative research*. Thousand Oaks, CA: Sage, 1994.