EDUCATION ON THE GIS FRONTIER: CYBERGIS AND ITS COMPONENTS

A Dissertation

by

FORREST JOSEPH BOWLICK

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	Sarah W. Bednarz
Committee Members,	Daniel W. Goldberg
	Robert S. Bednarz
	Bruce E. Herbert
Head of Department,	David M. Cairns

May 2016

Major Subject: Geography

ABSTRACT

Geographic information systems (GIS) are a fundamental information technology. Coupled with advancing developments in spatial analysis through geographic information science (GISci), the capabilities and applications of GIS and GISci continue to rapidly expand. This expansion requires practitioners to have new skills and competencies, especially in computer science and programming. One developing framework for GIS' future is that of Cyber Geographic Information Systems (CyberGIS), which fuses the technical capabilities of advanced cyber-infrastructure, like cloud and server computing, with the spatial analysis capabilities of GIS. This structure of GIS requires further computer science and programming abilities, but how GIS practitioners use and value the variant components within CyberGIS is unknown. This gap makes teaching and preparing students on the CyberGIS frontier difficult. The GIS skillset is in an ever-present state of re-imagination, but with the growing prominence of CyberGIS, which seeks to capitalize on advanced computing to benefit analysis in GIS, the need for an understanding of educational implications continues to grow.

This dissertation uses a mixed-methods approach to explore how CyberGIS functions academically. First, I explore how university geography departments in the U.S. integrate computer science and programming skills in their undergraduate geography and GIS degree programs by reviewing degree requirements in highly-ranked departments. Few departments require computer science or programming courses for undergraduate degrees. Then, I explore the nature of knowledge and skills in CyberGIS using machine reading and q-methodology to explore viewpoints of how key CyberGIS skills function. The three viewpoints I identify reveal highly conflicting mindsets of how GIS functions. Finally, I use syllabi from different GIS programming and computer science courses to identify common topics, course structures, and instructional materials across a broad sample of courses. Three major topic foci emerged, including GIS scripting with Python, web-enabling GIS with JavaScript and HTML, and geodatabase manipulation with SQL. Some common instructional materials exist, but syllabi show little consistency in their curriculum focus and instructional design within or across topics relating GIS programming and computer science.

There is little consistency or emphasis in current educational efforts concerning computer science and programming and how they function in building competencies required in CyberGIS. While CyberGIS promises advanced computing capabilities using complex systems, the fractured and uneven nature of basic computer science and programming instruction in GIS indicates that to achieve a Cyber-enabled GIS future, a much larger chasm between GIS and computer science must be bridged.

DEDICATION

For my Great-Grandmother Marguerite, who once commented on my undergraduate thesis by saying, “Well, I don’t understand most of it, but it sure is written well”. One of my greatest compliments.

As always, for my mom.

ACKNOWLEDGEMENTS

Typing four pages worth of acknowledgements seems strangely overwhelming after completing a dissertation, but here we are.

I will break with the usual tradition of these things and thank my wife, Katie, first and foremost for her support throughout this PhD adventure. Without her support, I would not be as successful, as grounded, or as capable as a student and researcher. When I am out of strength, she provides me the will to carry on. I would not be myself without her love and care. This work belongs to her.

My academic work has been structured, supported, improved and pushed most prominently and importantly by my advisor, Dr. Sarah Bednarz. I consider myself incredibly fortunate to have worked with her, learning from her extensive experience. She has provided me incredible opportunities to travel, meet an amazing network of researchers and future colleagues, and has supported me through successes and challenges (and repetitive edits). Her support will shape my career forever.

Just as I was fortunate to connect with Sarah at the end of her career, I had the advantage of meeting Dr. Dan Goldberg at the beginning of his. Dan's drive, energy, and extensive network of GIS resources have enabled my research to extend to areas I hadn't considered from my academic experiences. I know I'll be seeing him at many conferences and

workshops in years to come, and I am grateful for his support and tendencies to turn every project into a research opportunity.

The entirety of my committee deserves such praise. I thank Dr. Robert Bednarz for his guidance in my first semester, as I desperately tried to find something to connect with. I'm glad he kept lending me books and recommending people to chat with as I eventually found my way. I thank Dr. Bruce Herbert for his thoughtful and direct questions at each stage of this process. Finally, I thank Dr. Phil Klein for his willingness to discuss my work and share ideas on breaks as I was traveling. When I was an undergraduate, Phil taught one of my first geography courses, and I am very happy to have his support as I finish the long culmination of my status as a student.

There are extensive thanks due outside of my committee: to Dr. Oliver Frauenfeld, for fostering my success as an instructor and encouraging me to develop new ideas for coursework; to Dr. Steven Quiring, for the opportunities to write and revise the Planet Earth lab manual and for trusting me to do so in a professional and capable way; to Dr. Ra'Sheedah Richardson, for her ideas and imagination and review of my research and teaching. I need to thank Dr. Ellen Kohl for her perspectives and support as I slog to the finish, and Dr. Jamie Shinn for her great assistance trying to lock down a job. A knowing thanks to all of the graduate students who I have suffered alongside. Our community gives us strength, and I have learned from and been assisted by all of you.

Thanks to my family, especially my dad and my brother, who I don't talk to enough but am always thankful that I can. To my grandmothers and aunts and uncles and cousins, whose check-ins gives me perspective on what I am doing and achieving. To my wife's family as well, who make me feel as though I have so many additional homes to feel safe and rest in.

I have been supported by so many here at A&M, but as the hour grows dark on my time on campus, I know I would not have made it this far without the help of Student Counseling Services. Even now, as I am finishing this document and planning for the future, I seek their help as I try to close my time as a student. For anyone who comes across this work, I encourage you to seek similar help as well.

My friends sustain me in countless ways. I shouldn't try to list everyone because I will regret those I forget, and I will forget people. But to Brianna, Shelby, Luci, Jamie, thank you for the community; to Heather, Ellen, and Audrey, thank you for the games; to Andy, Sarah T., and Phil, thank you for the fun times; to Jarom, thank you for the perspective; to Iliyana, thank you for so many big and little things; to Parveen, Patrick, and Yolanda, thank you for sharing the cohort experiences; to all of the Geoservices minions, thank you for letting me wander through. To Dan V., thank you for the positivity; to Ryan D. thank you for the excellent conversations. To Trey and Tianna (and every 213 instructor over the years), thank you for being wonderful instructors (and for your inspiring successes); to Carl and Cesar, for being excellent TA's (and fantastic troubleshooters). To Courtney, Jesse, Jim, and Huck, who share the same struggle but outside these Aggie walls, thank you for your support, conversations, and comparisons. To Ryan H., Leigh, Chris, Nick, Donner, Regina, Lauren, Merida, Kelsey, Elia, Caitlin, Corbin, and everyone who I have forgotten, thank you for being.

To close I suppose I should acknowledge my own work. I rarely do.

You did it, Forrest. Now go get the next mountain.

NOMENCLATURE

API	Application Program Interface
BA	Bachelor of Arts
BoK	Body of Knowledge
BS	Bachelor of Science
CSS	Cascading Style Sheets
CyberGIS	Cyber Geographic Information Systems
GISys/GISystems	Geographic Information Systems
GISci/GIScience	Geographic Information Science
HPC	High Performance Computing
HTML	Hyper Text Markup Language
IT	Information Technology
SQL	Structured Query Language
TauDEM	Terrain Analysis Using Digital Elevation Models

TABLE OF CONTENTS

LIST OF FIGURES

Page

LIST OF TABLES

CHAPTER I

INTRODUCTION AND LITERATURE REVIEW

The fundamental skills and practices necessary to be proficient in Geographic Information Systems and Science (GIS and GISci) are in flux. GIS and GISci are fundamental tools for spatial analysis and essential information technologies. Through the investigations enabled by GIS and GISci, researchers explore the complexities of the world and derive meaningful comprehension of spatial processes. More than ever, items, objects, and ideas are spatially enabled, allowing for the discovery of patterns and relationships within the complex landscape of spatial interactions. This spatial enabling fuels the continuing growth of academic, industrial, and civic use of GIS, resulting in consistent demand for a GIS-enabled workforce. Consequently, the skills and techniques required by that workforce are fluid as GIS capabilities and utilities grow.

GIS has always been a field of rapid technological change. In the current era, GIS is web-enabled, capitalizing on cloud and server infrastructures to deliver spatial capabilities across platforms to a wide user-base. Simultaneously, advances in connecting spatial analysis with high performance and supercomputing capabilities allow fast processing of massive spatial datasets. These components are consolidated into a framework known as Cyber Geographic Information Systems, or CyberGIS. CyberGIS leverages traditional spatial analysis capabilities of GIS with advanced computer and web-infrastructures capable of high-performance computation, use of large datasets, and integration of distributed server architecture (Wang 2010). Computer science and programming are essential to the function of CyberGIS.

These growing interconnections between domain knowledge in geography, GIS, and computer science are changing the expectations of GIS users. This expanded topical core influences the actions and applications in using GIS. At the same time, the composition of the GIS skill set and the capabilities and awareness of a GIS user are also changing. The development of new technologies, applications for GIS analysis and means of collecting data have added complex avenues and topical familiarities necessary for GIS competence. Few investigations explore how web-enabled mapping, big data, and the prevalence of personalized GIS extensions (like mobile apps and GIS toolboxes) function in an educational context. Little research-based evidence exists contextualizing composition, nature, and requirements of GIS degrees, what knowledge or skills GIS experts value, or how key GIS components are delivered.

This dissertation investigates the knowledge, skills, and practices relevant to the shifting landscape of GIS, taking into particular account the changes precipitated by CyberGIS. CyberGIS is a cutting-edge, emerging form of GIS that relies on heavy integration of concepts, skills, and abilities from computer science and programming. To determine how this GIS frontier functions educationally, I research three distinct areas. I determine first the topics featured in existing CyberGIS and GIS programming courses; second, how experts value and rank these components; and third, the degree to which underlying computer science and programming knowledge is required in undergraduate degrees in geography and

GIS. Through these investigations, I address how CyberGIS components fit within the current landscape of GIS, and propose models for the structure and integration of these components. Within this work, I explore the educational implications and strategies suggested to meet these expectations.

Problem Statement and Importance

As more computer science skills become vital to making GIS function, how do courses, programs, and experts consider knowledge in this domain? CyberGIS is an amalgamation of GIS, computer science, and programming, which each have active domain-focused education research. However, there is little research on CyberGIS education, particularly on the functional fusion of its components. The interplay of these three distinct entities shapes education practice in GIS, and this dissertation explores those interactions.

The capabilities and applications of GIS continue to expand rapidly, requiring practitioners to have new skills and competencies. Advances in computing, software, and hardware have made it easier to access and use GIS technology to perform spatial analysis (Brovelli 2015). Engaging with the frontier of GIS requires additional capabilities, especially in computer science. GIS is no longer locked to a desktop computer or confined to expensive computer labs (Kong et al. 2015), but instead omnipresent on the web and mobile devices (Jazayeri et al. 2015). Fundamentals in computer science and programming are expanding and innovating what GIS can do, and have made these areas essential knowledge components for proficient GIS practitioners (Johnson 2010, Dramowicz et al. 1993, Liu et al. 2012). These innovations, like the ability to manipulate, analyze, and manage large data sets, and to design and implement systems to automate such actions are now core GIS competencies (Schulze et al. 2013). However, there is little research about how best to prepare the next generation of GIS users with such computer science and programming capabilities.

Formalizing the knowledge, skills, and practices necessary for students to capitalize on these affordances remains under-discussed. Despite underpinning the fundamental systems and enabling advanced research in GIS and GISci, how computer science and programming instruction are acquired in degree programs is unclear. Without a precise and detailed understanding of how experts in GIS use and value the variant components of the CyberGIS framework, developing curriculum and instructional strategies to prepare students in CyberGIS and computer science is difficult. CyberGIS is implemented in a range of applications (Agrawal 2015). Further, as computer science and programming skills integral to the use of CyberGIS and advanced spatial analysis become more essentially the skills of the GIS practitioner, we need guidance to direct and assist instruction in this realm.

The constant reimagining of the GIS exists with little empirical understanding of the function of GIS coursework. Broad consistencies have been identified as affecting how GIS courses function (Wikle and Fagin 2014). Components of CyberGIS and other advanced computing constructs, like developing software, automation, coding, and scripting, are quickly becoming fundamental GIS skills (Bearman et al. 2015). Indeed, growing industry evidence suggests that the computer science and programming components of professional GIS positions are increasing in importance (Hong 2015). Unfortunately, programming and computer science skills are the least developed essential GIS skills in academic coursework

(Şeremet and Chalkley 2014). Computer science and programming continue to transform the utility and ubiquity of GIS (Harvey 2013).

Three issues arise from this lack of knowledge. First, there is the issue of not having a research based understanding of the scope and sequence of computer science and programming coursework in geography. Second, there is no common understanding of the core components of CyberGIS and how GIS users value them. Third, there is no consensus on the content or learning outcomes of GIS programming courses, including clear frameworks for course topics, especially regarding the presentation of fundamental computer science knowledge in GIS. Together, these problems impair our ability to structure GIS courses, degrees, and cross-disciplinary fusions. I seek to address these concerns.

Research Questions and Chapter Outline

This dissertation consists of five chapters, front matter, and appendices. This first chapter presents an overview of the problem, defines key terms, outlines relevant literature and describes the methodologies used in the research to address the research questions. Chapters Two, Three, and Four are independent, standalone research articles describing the investigations conducted to answer my proposed research questions. I explore the following research questions to understand the educational situation of computer science and programming within GIS and CyberGIS. These questions address the problems and issues outlined previously. Finally, Chapter Five serves as a summary and synthesis of the related chapters, connecting the components within towards a cohesive direction for future work.

Chapter Two focuses on the role and relevance of computer science and programming courses in undergraduate geography degrees in select geography departments in the United States. In this chapter, I address questions concerning how computer science and programming courses are integrated into geography degree programs, using evidence from highly ranked geography degree-granting departments in the United States. I explore five research questions here, namely:

- *Are students acquiring computer science and programming knowledge and skills through courses within their primary degree programs in geography or outside of them?*
- *What factors influence the number of computer science or programming courses required for geography and GIS degrees? Especially of interest:*
    - *Do Bachelor of Science (BS) degrees require more computer science and programming instruction than Bachelor of Arts (BA) degrees?*
    - *Do GIS degrees or GIS-specific degree tracks or options require more computer science and programming courses than non-GIS degrees or tracks in the same department?*
- *Where is computer science and programming coursework typically positioned in the curriculum?*

These questions all focus on understanding how GIS, computer science, and programming are integrated in the formal structure of degree programs. Chapter Three outlines the key components of CyberGIS based on a review of published literature, exploring how GIS experts value these components. These data are used to construct conceptions of CyberGIS to relate to active educational practices in courses and degrees. This second article addresses three research questions:

- *What components of CyberGIS are most prominent in the published literature?*
- *How do experts value these components based on their experience?*
- *How do experts conceptualize CyberGIS?*

Answers to the questions in this article build an understanding of how experts conceive GIS and their attitudes towards components of CyberGIS. The final research article, presented in Chapter Four, analyzes course syllabi in GIS courses with programming and computer science components to identify common topics and approaches in curriculum and instruction in this topic. In this article, I investigate three research questions:

- *What is the nature of computer science and programming topics in GIS programming courses?*
- *How does the nature of these courses support the development of key GIS competencies given growing use of computer science and programming in GIS?*
- *What is the sequence of learning in these courses, and is there any curricular coherence in how these courses are presented?*

Through answering these questions, I describe GIS programming curricula and instructional strategies. Answers to these research questions develop a multi-faceted description of current education practices in GIS programming.

Each of these chapters has been or will be submitted to journals as individual articles. These three articles function in concert to describe the interconnections of computer science, programming, and CyberGIS in GIS education practice. To conclude, Chapter Five summarizes and synthesizes the findings of the previous research articles, discusses the studies' limitations, and makes recommendations based on the research. This linkage bridges the studies and establishes a context for a forward looking research agenda, outlining future work investigating the synthesis of geography, GIS, and computer science education.

Definitions

This research crosses disciplinary boundaries in three areas: GIS and GISci, programming and computer science, and expertise. Here, I define and clarify the core concepts in each and explain how these concepts interact in GIS education. While these sections scaffold a distinct organization of terms, the overlap and interactions of research in these areas facilitates a rich understanding of how the components interact to structure GIS education.

GIS and GISci

Geographic Information Systems (GIS) and Geographic Information Science (GISci) are often used interchangeably but have different meanings and associations (Wikle and Finchum 2003). Generally, GIS or GISystems refers to the tools, technologies and software that enable computer-based spatial analysis, and GISci refers to the theory behind such tools and technologies (Goodchild 1998). GIS, developed in the 1960's, is the original term for the software developed first by Roger Tomlinson to enable spatial analysis and the manipulation of spatial data (Goodchild 2010). The idea that geographic information has scientific components and fundamental truths embedded within the spatial nature of information arose much later, in the early 1990's (Goodchild 1992). A debate developed whether GIS was simply a tool for data manipulation or a science to discover inherit spatial relationships (Pickles 1997, Wright et al. 1997). This led to divisive frameworks for conceptualizing the nature of the growing field. Both the science of GIS and the technical tool and systems aspects continue to evolve; considerable advances in the tools and technology used with spatial data have been achieved. To balance the acknowledgement of this duality, especially given that the 'S' in CyberGIS represents systems instead of science, in this dissertation I use GIS when discussing the context relevant to both the systems and science aspects of the field, and notate GISci and GISystems when appropriate or necessary.

Computer Science and Programming

Computer science and programing are independent but complementary fields. CyberGIS requires general knowledge of computer science concepts and specific programming languages to develop technologies and accomplish advanced tasks. Computer science focuses on how computers function; courses therein teach fundamental skills and practices to develop perspectives on and understanding of those functions (Kay et al. 2000). Programming, in contrast, is concerned with the understanding of specific computer languages. Like learning a spoken language, learning to program involves developing skill in the semantics and syntax of a programming language to produce code and generate programs that direct computer functions (Van Merrienboer and Krammer 1987).

The primary enabling factor in facilitating the computationally intense nature of CyberGIS is high-performance computing (HPC) (Wang 2010). HPC uses grid, parallel, and cloud frameworks to improve aspects of GIS to enable features such as modeling and simulation (Kim and Tsou 2013). The first dedicated GIS supercomputer, ROGER (Resourcing Open Geospatial Education and Research) uses these frameworks to provide advanced computational capability in GIS applications (NCSA 2015). Grid computing integrates distributed HPCs for analytical optimization, beneficial in GIS simulations (Foster et al. 2001). Parallel processing allows complex actions to run on multiple processors to facilitate more rapid and complex problem solving (Foster 1995). Cloud computing uses internet services to store or process data, especially 'big data' which are the massive amounts of information generated by modern technology, tools and sensors of many types (Goldberg et al. 2014). The cloud-based aspects of HPC are distributed and scalable across systems. Distributed computing relies on networked computers to share processing tasks for greater efficiency, and open, distributed systems are adept at processing spatial and geo-data (Jhummarwala et al. 2014), while scalable computing efficiently processes growing amounts of work as queries are run (Zhou et al. 2015).

Additionally, while programming in GIS encompasses a number of languages, three are most relevant to this work. First, Python is a high-level, multi-purpose coding language used commonly to extend GIS capabilities (Zandbergen 2013). A customized Python package, ArcPy, is used extensively in GIS scripting (ESRI 2015). Second, as GIS moves to web enabled platforms, JavaScript is growing in prominence. It is a dynamic language for web development which, along with the Hyper Text Markup Language (HTML) and Cascading

Style Sheets (CSS), are essential to maintain web pages with GIS components (Wang and Dong 2014). Third, Structured Query Language (SQL), serves the special purpose of defining, managing, and manipulating databases, the structural foundation of storing spatial data (Ďuračiová 2013).

Educational Research on Expertise

This dissertation connects research on GIS education with established educational theory, specifically in the area of novice to expert transitions. There are numerous definitions and descriptions of expertise, varying by discipline (Farrington-Darby and Wilson 2006). Research consensus agrees that experts exhibit certain characteristics, traits, and actions that allow them to complete tasks in their area of specialty with greater accuracy and efficiency than non-experts, or novices (Day 2002). Novices and experts think differently, and extensive studies have sought to determine why (Jewell 2013). The purpose of studying experts is to attempt to define the outstanding qualities that distinguish some individuals in a domain from other non-outstanding individuals (Ericsson and Smith 1991), to help learners move efficiently toward acquiring expertise.

As noted by Downs (2014), identifying the intricacies of GIS expertise presents a major challenge. Attempts to organize GIS knowledge, skills, and practices, such as the GIS Body of Knowledge (BoK) do provide some insight into what a GIS professional should know and be able to do. GIS experts, like experts in all domains, master a great range of topics as they work and consider in their expert practice (Breβler 2012). But part of their practice includes adjusting practices as the core competencies change, and as computer science and programming components change the nature of GIS, the cores of that expertise might change (Huff 2014, DiBiase et al. 2010). Significant work is necessary to continue to understand the actions of GIS experts (Duckham 2015).

Another educational component and concept used in this dissertation is that of learning outcomes, which are sometimes referred to as learning objectives. Learning outcomes describe the essential performances, goals, and knowledge expected from a student at the end of a course (Harden 2002). They have grown in prominence at all levels of education as a tool to organize courses as well as measure and assess student learning (Lacireno-Paquet et al. 2014). Effective outcomes are measureable, observable, and action-oriented (Merchant et al. 2014, Vanblaere and Devos 2015, Savery 2006). In any given course, learning outcomes will span a range of performances from low-cognitive tasks, like remembering, to higher level cognitive tasks such as synthesis and prediction (Van der Kleij et al. 2015). Here, I use Bloom's Revised Taxonomy to arrange the dimensions and levels of learning expected in these objectives (Krathwohl 2002). This taxonomy is widely used as a worthwhile and accessible hierarchy to understand and differentiate between cognitive levels (Adams 2015). Learning outcomes provide a tool to measure student expectations in coursework, and using a widely adopted ranking system like Bloom's allows for some standardized means of discussing course similarities and differences. I engage more with this concept discussing outcomes in collected GIS programming syllabi in Chapter Four.

CyberGIS

CyberGIS is the frontier of GIS. It is the result of an interdisciplinary synthesis and merger of various components of cyberinfrastructure, GIS, spatial analysis, and spatial modeling (Wang 2010). In essence, CyberGIS leverages advances in computer science to enable more sophisticated spatial analysis within a geographic information system. This development has benefited advances in GISci. A major advantage of CyberGIS is its ability to integrate disparate forms of data in a multi-scalar fashion, using high-powered processing capabilities to explore geographic relationships which would be too time or resource consuming to discover through traditional GIS tools or with existing GISci methodologies. For example, TauDEM (Terrain Analysis Using Digital Elevation Models), provides web-based and parallel processed tools to extract hydrologic information from digital elevation models (Tarboton 2015). Normally, this would require extensive local computing resources and data downloads that may impair processing or resulting analyses (Yildirim et al. 2015). The CyberGIS approach uses servers to host this tool and provide analysis distributed using cloud computing. Ultimately, CyberGIS products complete specific workflows (or services) using a diverse set of tools and languages developed by multiple users across a broad community of researchers and analysts. Research to advance CyberGIS offers opportunities for new geospatial applications largely because it allows an increased amount and higher quality of spatial data to be processed, encouraging greater reliance on such data for decision making, and opening more scales of analysis for investigation (Goodchild 2007).

Overview of Literature and Foundations

The purpose of this overview is to provide a meaningful background that enables an understanding of the variant disciplines in play as I address education in CyberGIS. I draw from and connect to literature in geography, GIS, and computer science education, as well as psychological and educational literature on expertise. These connections to known theories of expertise help me build a research-based integration of relevant components. Each of my three investigations shares some of this common background. While specific components are explained in more depth within each individual chapter, the purpose of this review is to identify key findings that inform the general dialogue in these realms. Identifying these vital areas of overlap is necessary to understanding how CyberGIS functions in an educational sense. This presentation summarizes material from the three research chapters.

In this section, I first discuss research on the development of expertise and examine this in the context of GIS and geography education. Then, I discuss research linking computer science, programming, and GIS education, elaborating on potential barriers and issues in their teaching and learning. To close, I reconnect with the literature on CyberGIS, outlining how its development has led to the educational issues now apparent.

Fundamentals of Expertise

At a fundamental level, experts exhibit certain characteristics, traits, and actions which together function to allow them to complete tasks in their domain with greater accuracy and efficiency than non-experts (Ericsson 2008). The superior performance of experts develops from integrations of experience, education, and domain knowledge (Ericsson and Williams 2007). Identifying an expert is not a straightforward task, as conceptions of expertise vary based on field and specialty (Day 2002, Farrington-Darby and Wilson 2008). And while various temporal suggestions for earning expert status exist, such as ten years (Simon and Chase 1973) or 10,000 hours (Ward et al. 2004), these serve as ranges which help control for other variance in performance (Shermer 2013).

Expertise research attempts to define the outstanding qualities that distinguish some individuals in a domain from other non-outstanding individuals (Ericsson and Smith 1991). From studies of world-class performers in music and chess, to understanding performance in sports and medicine, in all domains, high achievement is controlled by practices developed in training over time (Ericsson 2005). These practices include extensive training, practical reasoning experience, and changes in perception and communication of concepts (Shanteau 1988). Together, these allow experts to perform at a high level in their domain by drawing on their years of domain experience (Foley and Hart 1992). Through this experience, experts have a greater capacity to differentiate and apply knowledge in more diverse contexts (Postigo and Pozo 2004). Experts make decisions differently than non-experts (Shanteau and Stewart 1992). In their ability to complete tasks without elaborate referencing, experts capitalize on their extensive knowledge and memory to recall and solve problems (Sternberg and Frensch 1992). However, this expert mindset also results in bias and the potential for functional fixedness (Pleggenkuhle-Miles et al. 2013).

Development of Expertise

The developmental structure of expert tendencies is deliberate practice, the long-term, sustained focus on gaining new knowledge and skills through critical analysis and feedback (Ericsson 1998). Expertise develops over time through deliberate practice and experience (Burgman et al. 2011). This development may take place in specific stages, like novice, competent, and expert (Hoffman et al. 1995). These stages structure knowledge components, perspectives, decision making, and commitments to the domain in question (Dreyfus and Dreyfus, 1986). Observing how experts address non-routine, challenging problems, as well as outlining the tasks that capture deliberate practice and critical expertise, provides insight into the processes by which expertise develops (Ericsson et al. 2009).

Experts have a greater capacity to understand different types of knowledge (Postigo and Pozo 2004). Three types of knowledge are identifiable: explicit, implicit, and conceptual. Explicit knowledge is based on established processes and can be transferred through documents, while implicit or tacit knowledge is based on practice (Smith 2001). Conceptual knowledge is the understanding of underlying structures, relationships, and interconnections in a topic (Eisenhart et al. 1993). Explicit knowledge, easily codified or written in manuals or books, is the easiest to transfer (Ribeiro 2013), while implicit or tacit knowledge requires broader awareness and reasoning abilities (Helie and Sun 2010). Conceptual knowledge requires linkages and abstractions between many information representations (Tyler and Moss 2001).

As novices grow into experts, their organization of knowledge and means of addressing problems change (NRC 2000). These different cognitive approaches reveal meaningful differences across domains, as novices are slower in noticing features and patterns, less flexible in approaching new situations and retrieving and applying knowledge, and demonstrate less depth of understanding in content (NRC 2000, Ericsson et al. 1993, Dror and Cole 2010). A novice's foundational knowledge of a subject plays a role in developing their expertise through deliberate practice (Hambrick and Meinz 2011). For example, while expert performers may spend the same time on performance-related activities, those who complete goal-directed tasks see the greatest improvement (Ericsson 2009). Experts draw on complex relationships between multiple skills and knowledge phases (Schmidt and Boshuizen 1993), so a single metric of analyzing talent or skill within a domain is insufficient to determine expert status (Ackerman 2014). Expert knowledge influences practice and decision-making (Brooks 2010), but determining and developing expert knowledge is an immense challenge. Therefore, a broad focus on the actions related to expertise in geography, GIS, and CyberGIS has merit (Downs 1994).

Instructional method influences the development of expertise (Boshuizen 2009). Generally, active learning constructs encourage novices to engage with content and reduce learning stress (Fee and Holland-Minkley 2010). Well-constructed curricula and texts also assist instructors in building expertise (Bransford and Schwartz 2009). While experts can suffer from a 'blind spot' when discussing, teaching, or designing instructional materials in their domain of expertise, mixed instructional methods and high quality feedback can reduce this concern (Nathan and Petrosino 2003). The overall development of expertise is a social process, in which people who are motivated to learn something have access to relevant teaching expertise and manage their struggle to improve through high-quality teacher-learner feedback (Bransford and Schwartz 2009).

Expertise in GIS

There is no broadly defined or established metric for identifying an expert geographer or GIS user (Downs 2014). While metrics for assessing geographic content knowledge, like the National Geography Standards or the GIS Body of Knowledge (BoK) exist, they function more to structure and organize concepts, terms and ideas rather than provide a framework for expert progression. Huynh and Sharpe (2013) outline a considerable number of tasks, tools, and metrics to assess components of such knowledge, but only begin preliminary investigations into what constitutes expert performance. Experts must balance subject, technical, ethical (Huff 2014), and other types of expertise in their use of GIS, and this expertise might not be evenly developed, as expertise develops across skills unevenly (Breβler 2012). As expertise grows in different components of GIS, users may need to reutilize their conceptions in ways they are not familiar with (Huff 2014, DiBiase et al. 2012). Understanding what concepts or ideas are present in courses and programs, helping to guide and build the development of GIS skill, then serves as meaningful landmarks on the way to understanding GIS expertise.

In geography, as in other domains, estimates of overall time to develop expertise are less important than the use of that time (Ericsson et al. 1993, Hung 2001, Brown et al. 2015). As Downs (2014) argues, even the time involved in earning a PhD may not be enough time to develop a 10,000-hour base of expertise in geography programs. Identifying the characteristics of expertise across the wide knowledge base of GIScientists is an expansive and difficult undertaking. Duckham (2015) proposes five cores of expertise for GIScientists, including having a sophisticated understanding of the structure of geographic information, understanding spatial uncertainty and dynamism, being able to apply aspects of geodesign, and understanding geographic language and cognition. While these broad categories may apply to areas of general expertise in GIS, the specific topics within these areas remain unclear. In defining key terms and explaining GIS from interdisciplinary perspectives, this work seeks to connect what the CyberGIS topics are and how users approach those topics.

Computer Science and Programming in GIS

Computer science and programing are fields, and are taught and used differently in coursework (Franklin et al. 2015). Computer science courses teach fundamental skills and practices to develop perspectives on and understanding of how computers function (Kay et al. 2000). Programming courses, in contrast, focus on developing skills in the semantics and syntax of specific languages to analyze code and generate programs (Van Merrienboer and Krammer 1987). The ability to program, develop applications, and exhibit proficiency in geospatial information technologies are core requirements for the GIS workforce (Mirzoev et al. 2015, DiBiase et al. 2010). Thus, career and post-undergraduate tracks in geography may require programming and computer science knowledge, yet geography students experience considerable anxiety when faced with such instruction (Muller and Kidd 2014, Rickles and Ellul 2014).

Integrating GIS, computer science, and programming instruction remains difficult (Sinton 2009). The GIS skills least developed during GIS courses are those related to programming and computer science (Şeremet and Chalkley 2014). Though typical GIS courses involve instruction across a broad range of concepts, computer science and programming topics are highly complex, especially related to fundamental knowledge and comprehension within the topics (Gasparinatou and Grigoriadou 2011). Learning in computer science contains considerable barriers, including conditions of negative reinforcement (Kinnunen and Simon 2012), impersonal interactions (Barker and Garvin-Doxas 2004), and detachment and de-motivation (Babin et al. 2009). Students need support to overcome these barriers (Robins et al. 2003). Effective support helps learners to develop domain-specific knowledge, regulate their cognition, behavior, and motivation (Devolder et al. 2012), and build viable mental models of key programming concepts (Ma et al. 2011). Methods to adequately support learners in GIS courses that include computer science components recognize the components being taught and connect to established practices teaching computer science (Nuutila et al. 2005). The best practices in expert programming are unsuited for those learning to program (Michaelson 2015).

Further, novice and expert programmers structure their coding activities in different ways, requiring different types of learning support (McKeithen et al. 1981). A point of frustration for GIS students arises as their GIS skill builds, but still face novice challenges in the computer science and programming aspects of their GIS abilities (Etherington 2015). A student’s expert GIS intuition may not be applicable in computer science and programming, slowing their capabilities of application and recall (Ertmer and Newby 1996). Because learning in computer science and programming is challenged by numerous barriers, students need to be motivated about the purpose, value, and utility of concepts within coursework (Carter 2006). Integrating and structuring concepts within the appropriate context is a vital component of merging computer science, programming, geography, and GIS coursework, especially as the research or professional careers these students seek use more programming and require more computer science ability (Merali 2010). This dissertation contributes to the development of materials to help learners structure these useful knowledge components within CyberGIS.

Issues in GIS Education

Course offerings in GIS are rapidly expanding (Lukinbeal and Monk 2015). These courses provide a broad base of knowledge and instruction in diverse components of GIS, including fundamentals in GIS use and GIS theory, the integration of GIS in specific knowledge domains, and the synthesis of computer science concepts, programming, technical architectures, and data acquisition methods (Kopteva et al. 2014). While recent reviews of GIS syllabi reveal some similarities in the general structure of a GIS course (Wikle and Fagin 2014), a lack of understanding concerning how GIS courses at all levels integrate components of computer science and programming prevails. Skills in these specializations, like developing software, automation, coding, and scripting, are quickly becoming fundamental skills of the GIS analyst (Bearman et al. 2015). The utility and ubiquity of GIS has been enhanced by the use and integration of computer science and programming into GIS (Harvey 2013).

GIS is taught in many different content areas and occurs in a wide variety of contexts. Ideally, these courses instill skills relevant not only to GIS but that are transferable to other topics (Şeremet and Chalkley 2014). While GIS is a support system for spatial thinking (National Research Council 2006), the development of spatial abilities is not the primary focus of most GIS coursework. Instead, the diverse uses of GIS allow for vastly different approaches to teaching GIS (Wright et al 1997, Pickles 1997, Whyatt et al 2011). GIS courses can function as introductions to learning specific software (like ArcGIS), descriptions of fundamental spatial topics and theories, or connections to advanced methods, knowledge domains, or technologies often in domains outside of geography (Ellul 2014).

As GIS has diffused from geography to other disciplines in both formal and informal learning environments, issues arise in teaching technical knowledge (Tate and Unwin 2009). A GIS student may not have complementary computer or spatial skills, depending on the home domain of their GIS instruction (Sarkar and Pick 2014). Additionally, as Blaschke and Merschdorf (2014) discuss, there is no wide ranging or extensive agreement about the precise boundaries of GIS and GISci, which makes defining an appropriate learning framework for GIS courses a challenging endeavor. For example, GIS instruction can exist at a basic 'button-pushing' level without developing analytical skills (Bishop 2009), and courses that follow rigid, guided formats achieve less than those that require complex spatial analysis and problem solving using GIS (Theo 2011) or deeply explore the functions of GIS (Miller 2000). Teaching 'with' GIS and 'about' GIS remains difficult to separate (Sui 1995), and adding computer science and programming components greatly enhances a course's complexity.

This duality pervades research in GIS, as quantifying the learning mechanisms and separate components of GIS, whether due to unclear course outcomes or the debate of tool vs. science, has been a difficult prospect (Wright et al 1997, Pickles 1997). An additional component adding to the complexity of GIS instruction is the underlying framework of spatial thinking embedded within GIS. Though GIS fundamentally concerns identifying spatial relationships, and research relating GIS and spatial thinking (Kim and Bednarz 2013, Madsen and Rump 2013, Lee and Bednarz 2009) is producing a growing understanding of how GIS enables spatial thinking, there is not a clear understanding of the optimal means or framework for developing these skills. The growing complications of computer science and programming instruction in geography are also under-addressed (Goodchild 2010).

It may seem needlessly complex to incorporate spatial thinking while considering GIS, geography, computer science, and programming, but working with GIS in classroom settings can develop core spatial skills, since GIS as a tool requires spatial interaction (Goodchild 2011). Integrating GIS into many means of instruction to capitalize on its affordances remains difficult (Sinton 2009). Just as instructors may not be disposed to teaching spatial skills through geography coursework (Jo and Bednarz 2014), they may not be disposed to teaching programming or computer science in the same way (Jacobs et al. 2015). Connecting spatial thought, the GIS interface, and computer infrastructure is also an important concern (Goodchild 2011), connected with disciplinary instructional challenges including lack of fundamental conceptual awareness (Kerski et al. 2013) and reliance on rote instruction rather than problem solving approaches (Read 2010).

As emphasized throughout this work, programming and computer science skills are the least developed subjects in GIS coursework (Şeremet and Chalkley 2014), despite industry evidence suggesting that the computer science and programming components of professional GIS positions are increasing in importance (Hong 2015, Solem et al. 2008). Instruction is complicated by the structure of information learning in computer science and programming (Gasparinatou and Grigoriadou 2011), requiring greater time and focus on these topics in a GIS course (Muller and Kidd 2014). While descriptions of pathways to expertise in GIS are rare (Downs 2014), novice programmers require about ten years, on average, to develop skills and learn content sufficient to become an expert programmer (Robins et al. 2003). Individuals may be both experts in GIS and novices in computer programming. This duality is an educational concern as programming and computer science become more essential to using GIS (Wallentin et al. 2015). Though programming is an essential skill in state-of-the-art GIS and GISci, its implementation into courses is unclear (Gaudet et al. 2003). Research in CyberGIS is confirming the importance of synthesizing computer science, GIS, and advanced computing infrastructure (Wang 2010). With this movement in GIS research and application in mind, the uneven programming and computer science knowledge of GIS learners may limit their progression towards expertise in GIS, especially as computer skills become more essential to the GIS practitioner.

Overlap of GIS and Computer Science in CyberGIS

CyberGIS is a synthesized and interdisciplinary field merging individual components of cyberinfrastructure, GIS, spatial analysis, and spatial modeling (Wang 2010). The merger of these components facilitates advanced spatial analysis and conceptual growth in GIS (Nyerges et al. 2013). CyberGIS integrates multiple forms of data at a range of scales with highly powerful processing capabilities to reveal spatial relationships that would otherwise be too time or resource consuming to discover. The research directions of CyberGIS present new opportunities for geospatial applications. Likewise, an increased amount, reliability, and quality of spatial data encourages greater reliance on such data for decision making, and opens more scales of analysis for investigation (Goodchild 2007).

Traditional desktop GIS are incapable of solving some problems that a CyberGIS easily handles (Wang et al. 2013). The transformation of data with CyberGIS requires extensive skill (Wright and Wang 2011). If CyberGIS incites a transition to a post-GISystems world (Wright 2012), future instruction will have to adapt to a GIS environment in which mastery of components of CyberGIS are the dominant elements to emphasize. CyberGIS requires a different set of knowledge, skills, and practices than traditional GIS. The CyberGIS environment requires a deeper and broader awareness of analytic techniques, as it is an open and collaborative interaction with GIS as opposed to the closed and 'monolithic' tradition of desktop GIS (Wang et al. 2013). CyberGIS requires a user-centered and collaborative mindset for success (Yang et al. 2011), which cloud computing and open source software facilitate. Expertise in CyberGIS not only requires knowledge of traditional GIS, but also aspects of computer science and collaborative design (Nyerges et al. 2013), which manages multiple perspectives from many disciplines to share expertise and build understanding in creating a new product (Du et al. 2012).

While the structural components of CyberGIS have been thoroughly explicated, the same cannot be said for the educational implications of this evolving system. Further, there is no delineation of how course design integrates the unique characteristics of CyberGIS with expert practices. This dissertation reviews diverse viewpoints and scales to determine key components of CyberGIS, to benefit future instructional and research efforts.

Methodology

To answer the research questions raised in this dissertation, I use a mix of qualitative and quantitative methodologies, including the collection and analysis of teaching artifacts, in-depth interviews, and statistical analysis. Thus I focus multiple investigative lenses to answer the research questions. All interactions with human subjects in this research were covered by protocols established with Texas A&M Universities Institutional Review Board (IRB). The IRB approval for data collection in this dissertation is present in Appendix IV.

Chapter Two

To investigate the degree to which computer science and programming are included in undergraduate geography degree programs, and to describe program requirements, I review and analyze course requirements for degree programs at highly ranked universities in the United States. I ask research questions that facilitate an understanding of the formal degree structure surrounding the fusion of geography, GIS, computer science, and programming.

Using rankings distributed by Academic Analytics and the National Research Council, I sought data on degree requirements from 55 undergraduate geography-degree granting departments in the United States. Akin to recent efforts exploring degree structure in master's programs (Lukinbeal and Monk 2015), this portion of my dissertation focuses on the growth of computer science and programming within undergraduate degrees. This focus builds an understanding of what core competencies are expected in these programs, while exploring the status of computer science and programming in degree programs at the selected universities.

Chapter Three

To investigate the prominent components of CyberGIS, I analyze peer-reviewed academic papers concerning CyberGIS to determine core concepts, and interviewed GIS experts to understand their view of these components' relationships. This investigation emerges from questions which seek to understand the components of CyberGIS, and how variant viewpoints of these components function. After constructing a core CyberGIS literature through academic papers using 'CyberGIS' as a keyword, as well as through the foundational papers listed by the CyberGIS institute (CyberGIS 2015), I use machine reading software (the Machine Learning for Language Toolkit (MALLET)) to extract core topics. MALLET allows for a statistically based, natural language form of processing (McCallum 2002). The topics and terms generated with MALLET, which represent key programming languages, computing approaches, analytical methods, and terminology in CyberGIS, form the foundation for understanding how key topics interrelate in CyberGIS.

To understand how experts value these terms, I use q-methodology. Q-method allows for a quantitative assessment of subjective rankings (Wright 2013), facilitating the construction of statistically backed viewpoints through factor analysis. Such evidence-based data is strongly relevant in geography education research as a means to provide methodologically replicable data (Wright 2012). In q-method research, participants rank a set of terms based on their viewpoint of the term's value. After this ranking, they complete a semi-structured validation interview discussing, explaining, and elaborating on their choices. Topics for discussion include examples of the utility of their selections, discussions of how the terms fit into their career, and elaborations on how the process helped reveal their viewpoint. With this information, I produce an in-depth discussion of how expertise interfaces with CyberGIS terminology and structures CyberGIS conceptions.

Chapter Four

In addressing the final questions concerning course topics, and to understand what content GIS programming courses integrate into instruction, I deconstruct syllabi used in teaching GIS courses before or during the fall semester of 2015. With these instructional materials, I seek to answer questions regarding the nature of computer science and programming topics in GIS courses. I contacted course instructors through direct contact at major conferences (2015 Association of American Geographers, 2015 University Consortium of GIS, 2015 ESRI User Conference), through posts to professional and academic geography and GIS listservs, and through social media. Existing public databases like the CyberGIS Fellows Initiative (CyberGIS Center 2014) and the GeoTech Syllabi Repository also provided syllabi. To analyze these components, I compare syllabus construction, comparing topics, learning outcomes, instructional materials, and teaching methods. I use Bloom's Taxonomy to structure and organize outcomes across syllabi, relating organization of topics to learning expectation. This process allows me to consolidate trends and identify patterns in topical structure, materials used in course instruction, and other course details.

Methodological Connections

These methods reflect recent calls for research in geography and GIS education, and draw on recent research in this domain. This work follows the recommendations of the *Geography Education Research Report* roadmap, building clear lines of research, establishing theory-based investigations, and providing materials for future replication and development for continued progress in these areas. By using Q-method, for example, I connect with recent calls for diversification of geography education research (Wright 2012), but also provide materials and documentation for future improvement on this initial research effort. And while Wikle and Fagan's (2014) robust review of over 300 GIS course syllabi provides an immense view of the topics, trends, and inconsistencies within current GIS coursework across the US, my review here of the more narrowly defined GIS programming courses reflects existing scholarship while providing new insights into a key component of the field. In connecting with existing scholarship, I position my work as complementary to existing scholarship while maintaining an innovate reach to necessary subfields in computer science and programming. As Gasparinatou and Grigoriadou (2011) point out, learning in computer science is complex and riddled with obstacles. Looking forward with CyberGIS as a guide, geographers should expect those obstacles as well, coupled with obstacles already present in geography. Building research now is an important step to understanding the GIS future.

Conclusion

This chapter outlined the structure, provided important definitions and background, and defined the research questions and methodologies for the forthcoming content of this dissertation. There is a meaningful lack of research evidence in understanding how computer science and programming topics are integrated into GIS programs, what topics are viewed as most meaningful, and how they are integrated in instruction. This dissertation looks at each of these components by exploring the educational interactions at multiple levels: across programs, across courses, across research literature, and from the viewpoints of varying experts. These variant scales provide diverse avenues to link and discuss conclusions from this research.

The growing utility of CyberGIS and its utility in fusions GIS with computer science and programming is an exciting prospect for many fields in and related to geography. With the analytical capabilities of supercomputers, cloud and web based services and means of managing, collecting, and storing data, the frontiers of GIS are open to continued expansion and growth. Yet, without beginning to understand how new GIS users learning the technology in this CyberGIS epoch approach and learn the variant technologies, any progress in defining key practices, learning frameworks, and other consistent approaches to this domain will remain stunted. There are considerable lessons to be applied from existing research in geography, GIS, and computer science education, but no evidence pointing to how these courses function, and what lessons might be applied in different courses or uses of CyberGIS.

As Goodchild notes in his 1992 paper outlining the fundamentals of the newly reconceived Geographic Information Science,

"few people have had the time to write the textbooks or to identify the intellectual core, or to publish the good examples."

In this case, he referred to the fundamentals of what made the 'S' in GIS 'Science', rather than just the tools and processes of the computer 'System'. The same statement would not be out of place when considering computer science and programming in the field nearly twenty-five years later. More work is necessary and additional contexts must be investigated to continue to understand what GIS 'is', and how computer science and programming fit into it, continuing the long tradition of attempting to answer that question.

It is the fundamental purpose of this dissertation to address this need. By unifying discussions in GIS and computer science, the roles of educational literature in both fields can contribute to guiding future courses and teaching in this realm. By applying further lessons from the wealth of research on expertise, CyberGIS can continue to evolve as a powerful analytical framework with well-researched connections to existing educational practice. Though learning in computer science is fraught with barriers, and GIS education still grapples with teaching the science versus the tool, all of these perspectives need to engage in determining what meaningful lessons can be applied within learning in CyberGIS. In the following chapters, I explore these connections, build evidence-based connections, and further discuss these concepts through novel research tasks, providing much needed depth and detail to the CyberGIS frontier.

CHAPTER II

COMPUTER SCIENCE AND PROGRAMMING COURSES IN GEOGRAPHY DEPARTMENTS IN THE UNITED STATES

Geographic Information Systems and Science (GIS and GISci) are evolving rapidly. GIS is no longer locked to a desktop computer or confined to expensive computer labs (Kong et al. 2015). The affordances of advanced cyber infrastructure, like distributed, high-performance computation, allow a greater flexibility, diversity, and scale of spatial investigation. Fundamental computer science and programming skills are expanding the capabilities of GIS, and have become more prominently essential skills of proficient GIS practitioners (Johnson 2010, Dramowicz et al. 1993, Liu et al. 2012). Data manipulation, analysis, and management, along with system implementation and design, are now core competencies in GIS (Schultze et al. 2013). With this growing emphasis of programming and computer science in the epoch of the internet of things, big data, and web-enabled society, determining how to best incorporate these topics in geography coursework is an essential research effort (Muller and Kidd 2014).

Yet despite underpinning the fundamental technologies used in GIS and GISci, the degree to which computer science and programming instruction are included in degree programs in these domains is unknown. What knowledge, skills, and practices will GIScientists need? What types of curricula, instruction, and learning experiences may develop the broad skillsets necessary for both competent and expert use of GIS? Before being able to answer such questions, we need to understand where we are right now in an educational sense.

Very little is known about the curricula and requirements of degree programs in GIS and GISci. There are a plethora of GIS degrees, certificates, and training venues to obtain such education and training, but no guidelines or structure to the integration of computer science or programming components. Constructing a descriptive analysis of the current state of degree requirements, recommended coursework, and pathways to GIS competence is a useful and worthwhile starting point towards further understanding of how current formal competencies function. The status of computer science and programming instruction, whether 'outsourced' to other academic departments or held within the geography-degree granting department, merits special consideration, particularly due to the changes outlined above. I provide an initial glimpse into the growing role and importance of computer science and programming instruction in current degrees by addressing several questions.

First, are students acquiring computer science and programming knowledge and skills within their primary degree programs in geography or outside of them? There are many venues to learn these knowledge and skills in the university environment, and much like GIS, programming does not 'belong' to any individual entity. How students acquire this knowledge is of interest. Second, what factors influence the number of computer science or programming courses required for geography and GIS degrees? Further, do Bachelor of Science (BS) degrees require more computer science and programming instruction than Bachelor of Arts (BA) degrees? Given the diversity of research, applications, and programs in geography identifying patterns related to the prevalence of computer science and programming coursework in geography and GIS degrees may provide a useful basis for investigating further trends in this realm. Third, do GIS degrees or GIS-specific degree tracks or options require more computer science and programming courses than non-GIS degrees or tracks in the same department? Related to this question is the issue of sequencing; where is computer science and programming coursework typically positioned in the curriculum? Knowing whether there is a consistent sequence of courses or if courses are merely optional rather than required may provide meaningful information about the general integration of computer science and programming in GIS programs. Keeping the variation in program instruction in mind, I intend to explore the disparate student preparation and provide insight into the relative importance of computer science and programming in different programs and degrees. It is my intent to construct a reference useful for departments or colleges considering revising degree plans and programs.

Overall, I seek to understand the position of computer science and programming instruction in GIS and geography programs; is this critical domain being taught in departments of geography, is it housed in traditional computer science departments, or is there a potential point of collaboration between GIScientists and computer scientists? To address these questions, I first explore the intersections between computer science, programming, geography, and GIS. Then, I discuss the selection of universities in this sample and my methodologies for extracting program requirements. After presenting the data discovered in my analysis, I discuss the findings as relevant to my research questions. I conclude by outlining and extending the implications of this work, and by exploring additional research lines for investigation on this topic.

Computer Science and Programming in Geography and GIS

Computer science and programing exist as independent ideas and concepts, and manifest in different coursework. Computer science courses teach fundamental skills and practices to develop perspectives on and understanding of how computers function (Kay et al. 2000). Programming courses, in contrast, focus on developing skill in the semantics and syntax of specific languages to analyze code and generate programs (Van Merrienboer and Krammer 1987). The ability to program, develop applications, and generally to be proficient in geospatial information technologies are core proficiencies required by the GIS workforce (Mirzoev et al. 2015, DiBiase et al. 2010). Thus, many career and post-undergraduate tracks in geography require programming and computer science knowledge, even though geography students experience considerable anxiety when faced with such instruction (Muller and Kidd 2014, Rickles and Ellul 2014). Integrating GIS, computer science, and programming instruction remains difficult (Sinton 2009).

The GIS skills least developed during GIS courses are those related to programming and computer science (Şeremet and Chalkley 2014). Though typical GIS courses involve instruction across a broad range of concepts, the domain information in computer science is considered to be more complex in terms of fundamental knowledge and comprehension (Gasparinatou and Grigoriadou 2011). Learning in computer science is complex and fraught with known barriers including conditions of negative reinforcement (Kinnunen and Simon 2012), impersonal interactions (Barker and Garvin-Doxas 2004), and detachment and de-motivation (Babin et al. 2009). Students need support to overcome these barriers (Robins et al. 2003). Effective support helps learners to develop domain-specific knowledge, to regulate their cognition, behavior, and motivation (Devolder et al. 2012), and to build viable mental models of key programming concepts (Ma et al. 2011).

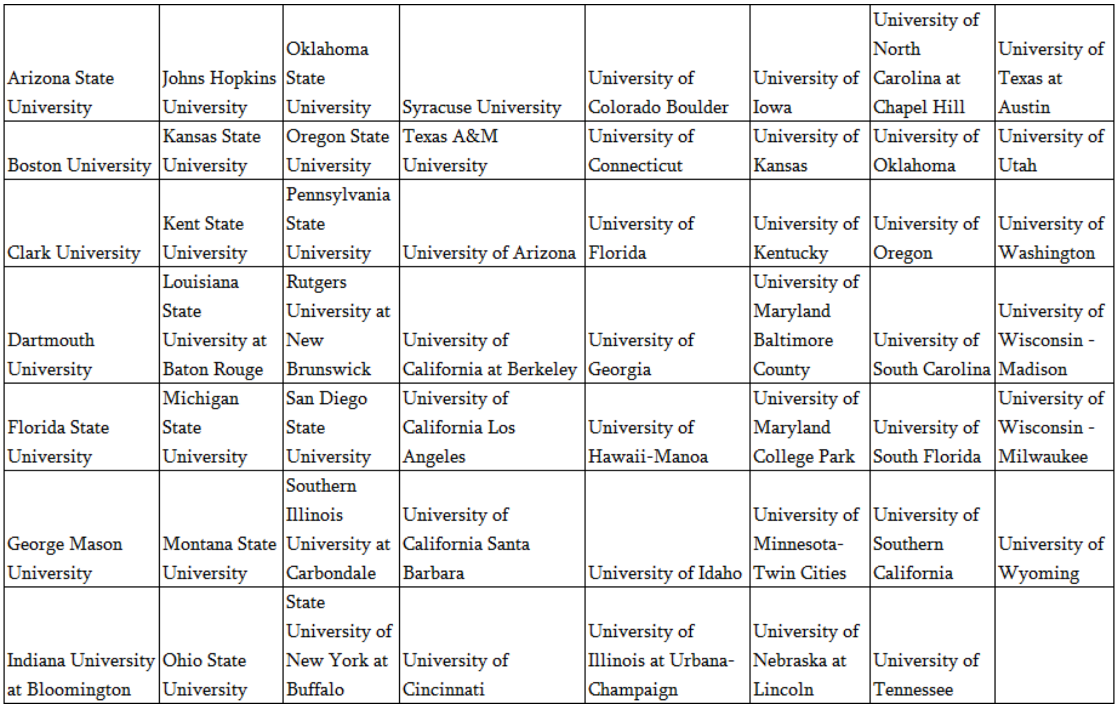
A point of frustration for GIS students arises as their GIS abilities build, yet they are faced with novice challenges in computer science and programming. A student's growing status and intuition as an expert GIS user may not be applicable in computer science and programming, slowing their capabilities of application and recall (Ertmer and Newby 1996). Further, novice and expert programmers structure their coding activities in different ways, requiring different types of learning support (McKeithen et al. 1981). The instructor can serve as a major barrier as well, as expert programmers are often not educators (Robins et al. 2003), and GIS instructors with computer science or programming experience often lack the formal programming training to effectively teach the subject (Muller and Kidd 2013), or may rarely approach the subject in coursework at all (Etherington 2015). Because learning in computer science and programming is challenged by numerous barriers, students need to be motivated about the purpose, value, and utility of concepts within coursework (Carter 2006). Meaningful motivators are essential to computer science and programming learning, whether they be games, explicit disciplinary contexts, or direct connections to professional utility (Papastergiou 2009, Forte and Guzdial 2005). Developing and structuring resources that help students construct useful knowledge while structuring concepts within the appropriate context is a vital component of integrating computer science, programming, geography, and GIS coursework, especially as GIS careers and academic work integrates these concepts.

Design and Methodology

To investigate the degree to which computer science and programming are included in geography degree programs, and to describe program requirements, I surveyed curricula and course offerings in selected geography programs in the United States. Reviews of curricula can provide insight into the scope and nature of academic programs. Course description analysis is an established means of research, useful in diverse fields related to geography, GIS, and computer science. In library science, such analysis has been used to track the emphasis of core topics in library and information science degrees (Irwin 2002); in psychology, it has been used to determine the types of alternative degree options available to undergraduates (Messer et al. 1999); and for establishing a baseline for curriculum design in management of information systems and computer information systems degrees (Kung et al. 2006). In computer science, Davies et al. (2011) discovered a broad 'uniformity' in how most departments teach the first courses in the computer science major sequence. Guo (2014) reported a switch to Python as the first language learned by new students in introductory computer science courses. This type of review is broadly used and useful to understand how courses and programs function.

This review encompasses all geography programs listed by the most recent National Research Council (2010) rankings of geography programs, as well as those programs listed in the set of rankings by academic business intelligence company Academic Analytics (2015), a total of 55 departments (Table 2-1). These rankings provide an easy to use reference of programs in geography to facilitate program analysis.

Table 2-1. List of universities in this sample. I extracted these universities from the 2010 NRC and 2015 Academic Analytics reports.



Arizona State University	Johns Hopkins University	Oklahoma State University	Syracuse University	University of Colorado Boulder	University of Iowa	University of North Carolina at Chapel Hill	University of Texas at Austin
Boston University	Kansas State University	Oregon State University	Texas A&M University	University of Connecticut	University of Kansas	University of Oklahoma	University of Utah
Clark University	Kent State University	Pennsylvania State University	University of Arizona	University of Florida	University of Kentucky	University of Oregon	University of Washington
Dartmouth University	Louisiana State University at Baton Rouge	Rutgers University at New Brunswick	University of California at Berkeley	University of Georgia	University of Maryland Baltimore County	University of South Carolina	University of Wisconsin - Madison
Florida State University	Michigan State University	San Diego State University	University of California Los Angeles	University of Hawaii-Manoa	University of Maryland College Park	University of South Florida	University of Wisconsin - Milwaukee
George Mason University	Montana State University	Southern Illinois University at Carbondale	University of California Santa Barbara	University of Idaho	University of Minnesota-Twin Cities	University of Southern California	University of Wyoming
Indiana University at Bloomington	Ohio State University	State University of New York at Buffalo	University of Cincinnati	University of Illinois at Urbana-Champaign	University of Nebraska at Lincoln	University of Tennessee	

I executed the following steps to extract course data. My work progression involved identifying the college the selected departments were located in, what degree programs they offered, what courses were required for each degree, and obtaining and analyzing the course descriptions from the course website or university catalog. I provide a logic model of this process in Figure 2-1.

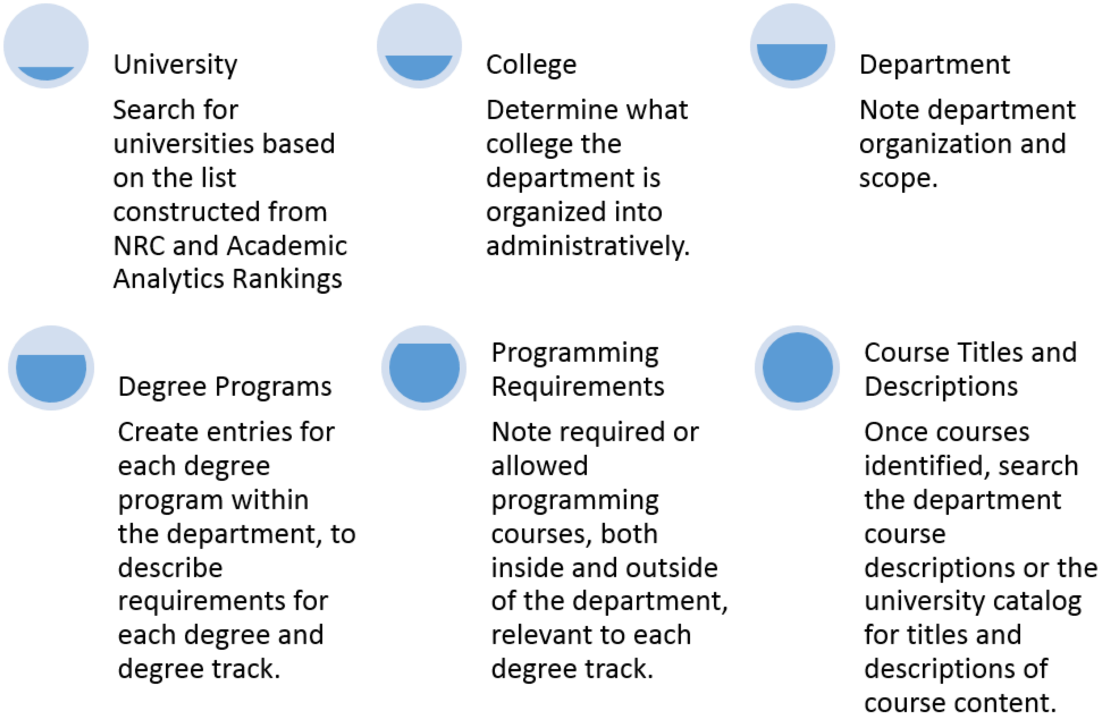

University
Search for universities based on the list constructed from NRC and Academic Analytics Rankings
College
Determine what college the department is organized into administratively.
Department
Note department organization and scope.
Degree Programs
Create entries for each degree program within the department, to describe requirements for each degree and degree track.
Programming Requirements
Note required or allowed programming courses, both inside and outside of the department, relevant to each degree track.
Course Titles and Descriptions
Once courses identified, search the department course descriptions or the university catalog for titles and descriptions of course content.

Figure 2-1. Analytical logic model. A logic model indicating the progression of data collection for this course description. Each step provides greater focus in determining the structure and content of computer science and programming in degree programs.

To begin, I visited each department's website and noted the type of college the department was housed in (College of Geosciences, College of Letters, etc.), listing each degree offered within the department. To determine what courses existed in this sample I inspected each department's website for degree requirements and course listings for all undergraduate degrees offered by the department as well as specializations (listed as 'tracks' or 'options', depending on university terminology) within these degrees. I did not consider minors or certificates for this review. While many departments provided this information on their website, some web site structures required additional reference to course catalogs and broader university resources, like requirements listed by the registrar. For each degree identified, I searched for terms in course titles and descriptions which indicated some form of relevant computer science and programming instruction, including the words ‘programming’, ‘database’, ‘cyber’, ‘web’, ‘model’, ‘computing’, ‘analysis’, and ‘server’. I also searched for terms that indicated specific programming language instruction, like ‘Python’, ‘Java’, ‘JavaScript’, ‘SQL’, and ‘C#’, though the languages ‘FORTRAN’ and ‘MATLAB’ were discovered unexpectedly. I used these terms as indicators for courses focused on computer science and programming applications in GIS.

After analyzing the course descriptions, I determined whether the computer science and programming courses included were required for degrees, ‘elective’ options in degree plans, or not present in degree plans, namely, optional courses not required but offered by the department. Further, I noted any programming or computer science courses offered outside of the department listed as a requirement on each department’s degree plans or degree tracks.

Results

Search Results: Universities, Degrees, and Courses

Of the 55 departments surveyed, 44 offered at least one course with the department prefix with a GIS programming focus. Of the 44 universities with GIS Programming courses, 16 offered only one course within the department. Figure 2-2 shows a count of universities sorted by the number of programming courses offered.

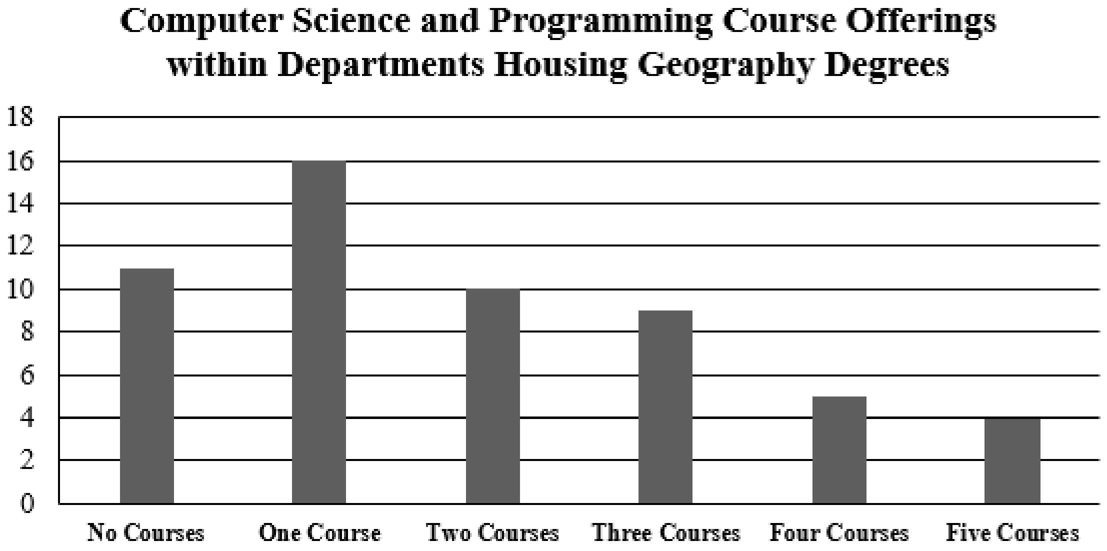
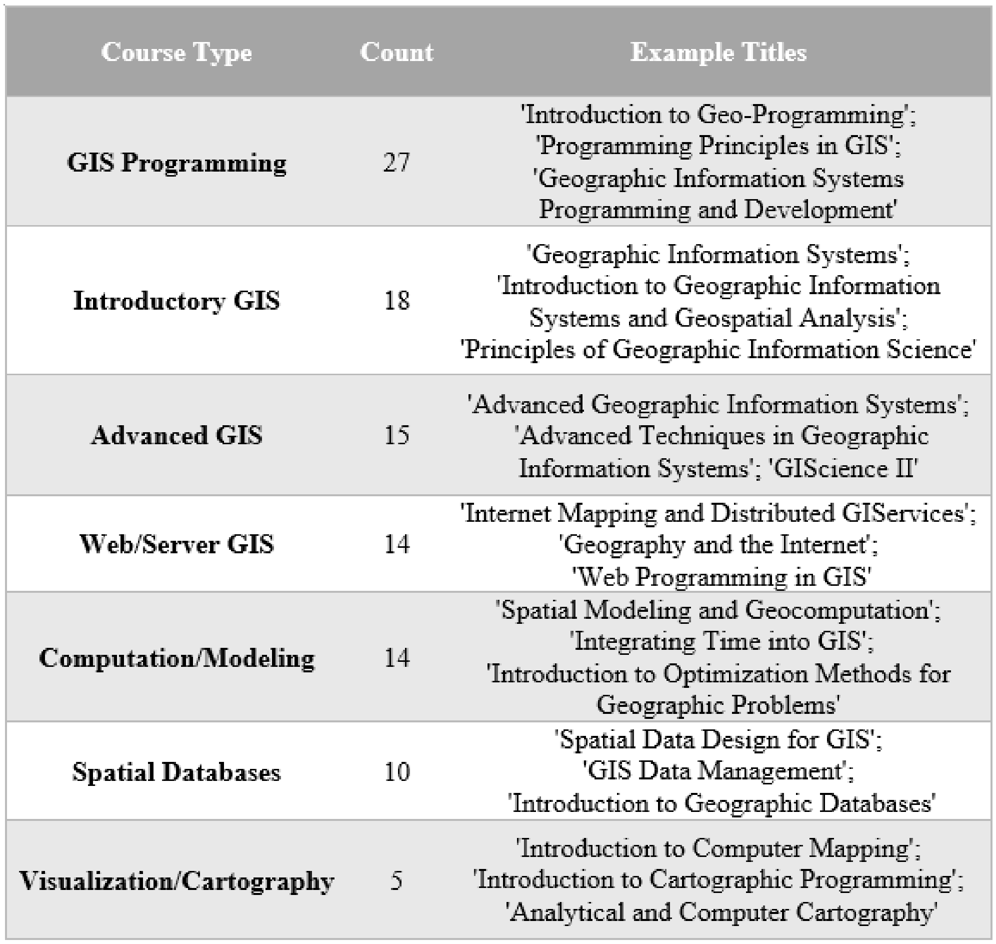


Figure 2-2. Course offerings. Number of universities with amount of computer science and programming courses offered within departments offering geography degrees.

Overall, I identified 103 geography programming courses offered among the 55 universities. Table 2-2 provides course titles and a count of courses. These courses fall into seven distinct categories. Most courses identified exist as specific and intensive GIS programming courses. Though programming components, topics, and instruction are present in many types of GIS and cartography courses, the courses I identify here exist as explicit course constructs with instruction in these topics.

Table 2-2. Computer science and programming course types. Types of computer science and programming courses identified within departments in the sample of 103 courses. Many programming courses in this sample have 'GIS programming' or similar titles. Some introductory and advanced levels of general GIS courses, including 18 courses introducing GIS, contain explicit descriptions of computer science and programming as main components of the course. Specific applications of programming or computer science are also widely prevalent, as web GIS courses, spatial database courses, and computation courses are evident in a broad variety of course types.



Course Type	Count	Example Titles

GIS Programming	27	'Introduction to Geo-Programming';
		'Programming Principles in GIS';
		'Geographic Information Systems Programming and Development'

Introductory GIS	18	'Geographic Information Systems';
		'Introduction to Geographic Information Systems and Geospatial Analysis';
		'Principles of Geographic Information Science'

Advanced GIS	15	'Advanced Geographic Information Systems'; 'Advanced Techniques in Geographic Information Systems'; 'GIScience II'

Web/Server GIS	14	'Internet Mapping and Distributed GIServices';
		'Geography and the Internet';
		'Web Programming in GIS'

Computation/Modeling	14	'Spatial Modeling and Geocomputation';
		'Integrating Time into GIS';
		'Introduction to Optimization Methods for Geographic Problems'

Spatial Databases	10	'Spatial Data Design for GIS';
		'GIS Data Management';
		'Introduction to Geographic Databases'

Visualization/Cartography	5	'Introduction to Computer Mapping';
		'Introduction to Cartographic Programming';
		'Analytical and Computer Cartography'

Some terms and components are more common throughout the data set than others, as shown in Table 2-3 and Figure 2-3. While the length and detail of course descriptions varies, the broad focus of these courses on combining the fundamental spatial components of GIS with programming concepts is clear across the sample. Table 2-3 shows a count of common words used in these course descriptions. These courses have obvious connections to geography and spatial analysis, hence the prominence of the ‘GIS’ terms (geographic, information, science) and ‘spatial’ in the raw count. Other emphases in these courses are evident as well: for example, these courses focus more on ‘application (37)’ than ‘technique (22)’, and are seen at both introductory and advanced levels. To contrast, Figure 2-3 shows a word cloud of terms in course descriptions after removing common words and terms like ‘GIS’, ‘programming’, ‘geographic’, ‘information’, and ‘systems’ to highlight other common content. In this case, the primary centering of these courses on spatial data is apparent.

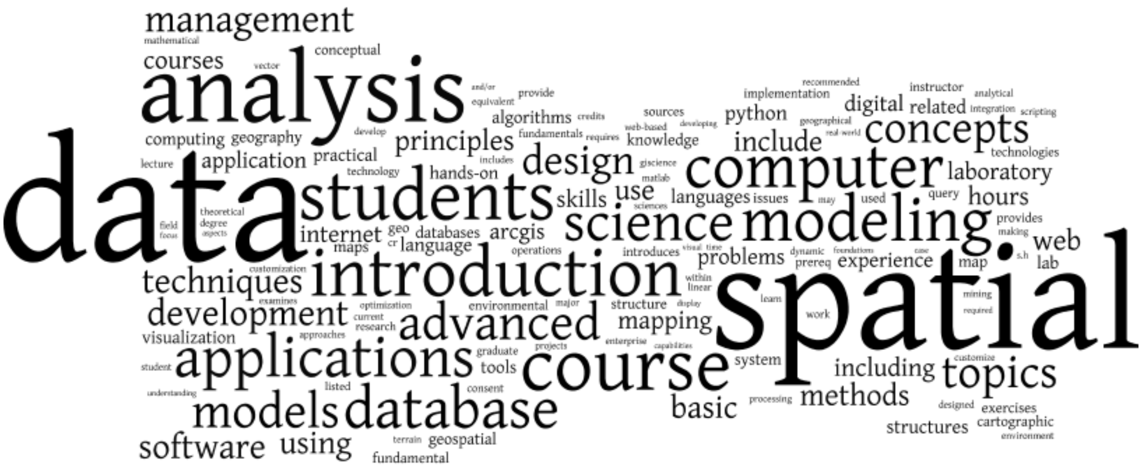
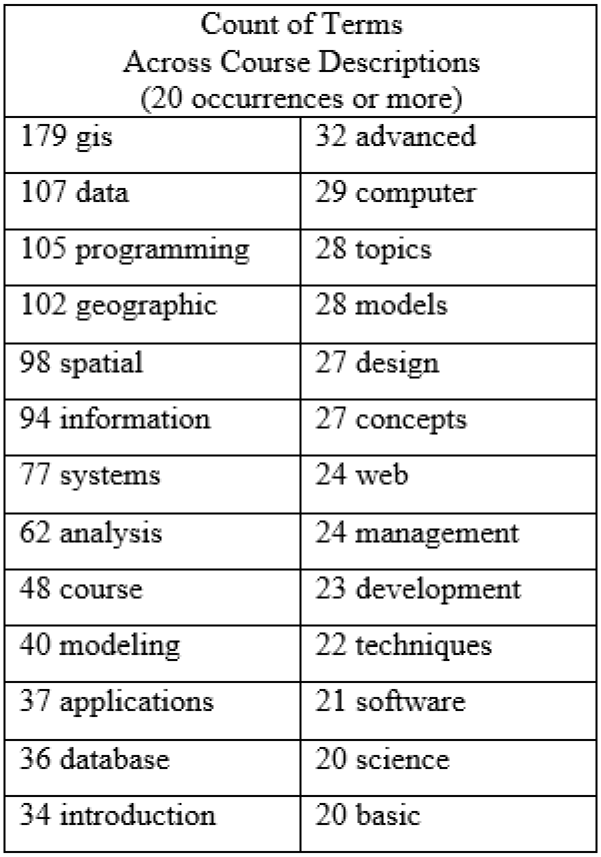


Figure 2-3. Course description word cloud. A word cloud of terms in GIS programming course descriptions. Larger words indicate more occurrences of the word. Many GIS programming courses also use the words databases, spatial data analysis, and data management. Descriptions also situate many of these courses as application based. Some of the technical components of these courses are also evident, like the Python language and use of ArcGIS.

Table 2-3. Terms with 20 or more occurrences in the set of 103 course descriptions. Common words like ‘of’, ‘the’, etc. removed.



Count of Terms Across Course Descriptions (20 occurrences or more)	
179 gis	32 advanced
107 data	29 computer
105 programming	28 topics
102 geographic	28 models
98 spatial	27 design
94 information	27 concepts
77 systems	24 web
62 analysis	24 management
48 course	23 development
40 modeling	22 techniques
37 applications	21 software
36 database	20 science
34 introduction	20 basic

Computer Science and Programming Courses by Degree Requirement

I identified 210 different degree options (noted on some university websites as ‘tracks’ or ‘specializations’) that led to a degree (e.g. BS or BA). This count does not include certificates or minors. As shown in Figure 2-4, only 22 of these degree options required a course teaching GIS programming; an additional 97 tracks allowed a GIS programming course to fulfil a degree requirement. Some of this count, though, is attributable to broad and flexible degree requirements: for example, 'take any 400-level course', which would allow a GIS programming course to count towards the degree. Of all 210 degree options, 15 required computer science or programming courses offered outside of the geography department. Only one degree required both an in-department GIS programming course and a computer science course offered by a department of computer science.

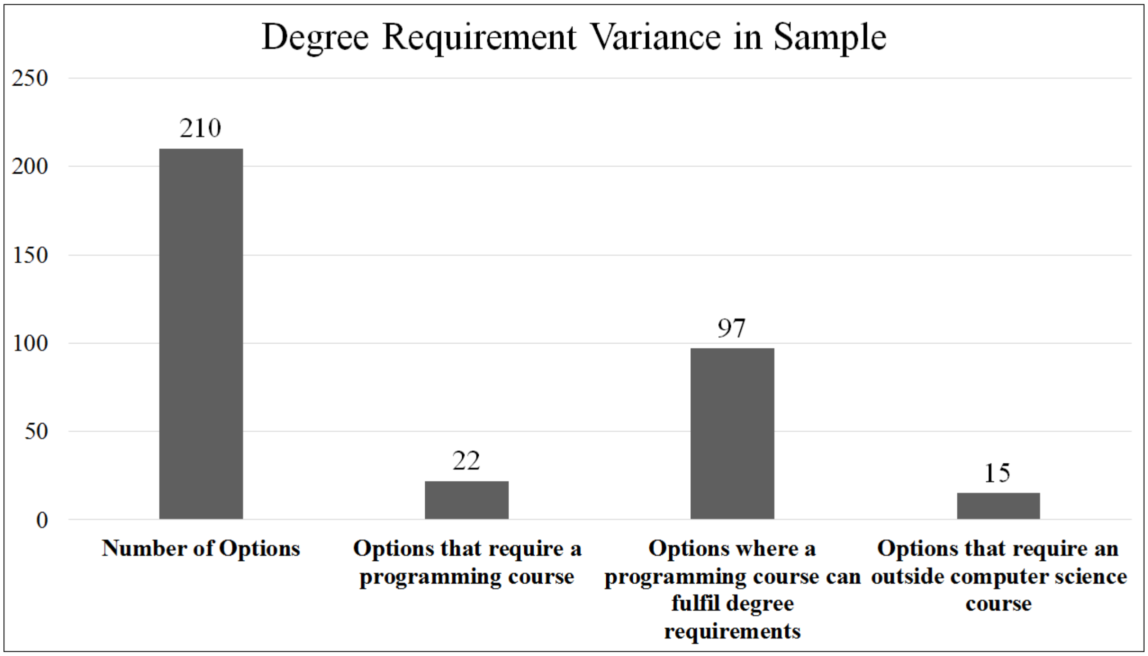
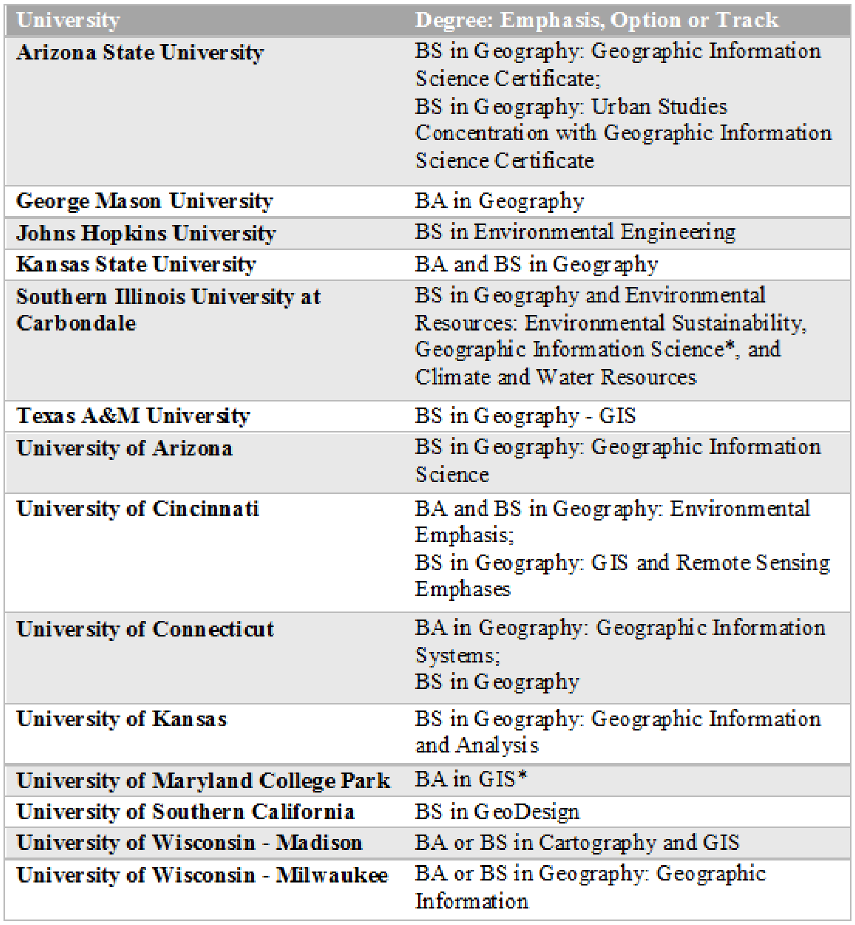


Figure 2-4. Degree requirement variance in sample. Different types of programming course requirements in degree options in the sample.

Computer Science and Programming Courses by Degree Program Type

Of the 22 degree options that required a computer science or programming course, five led to a BA degree, 15 to a BS degree, and two to either a BS or a BA. 12 of the tracks that require computer science and programming courses result in GIS degrees or geography degrees with an explicit GIS emphasis, while five of the tracks confer an environmental emphasis, and three tracks are general geography degrees without any additional emphasis. The remaining two result in a remote sensing emphasis and a GeoDesign degree. Only two tracks require multiple computer science and programming courses as degree requirements. Table 2-4 outlines these degree options requiring these courses by university and emphasis.

Table 2-4. List of degrees and degree options that require a computer science or programming course from within the department. Degrees or options that require multiple courses indicated by a star (*).

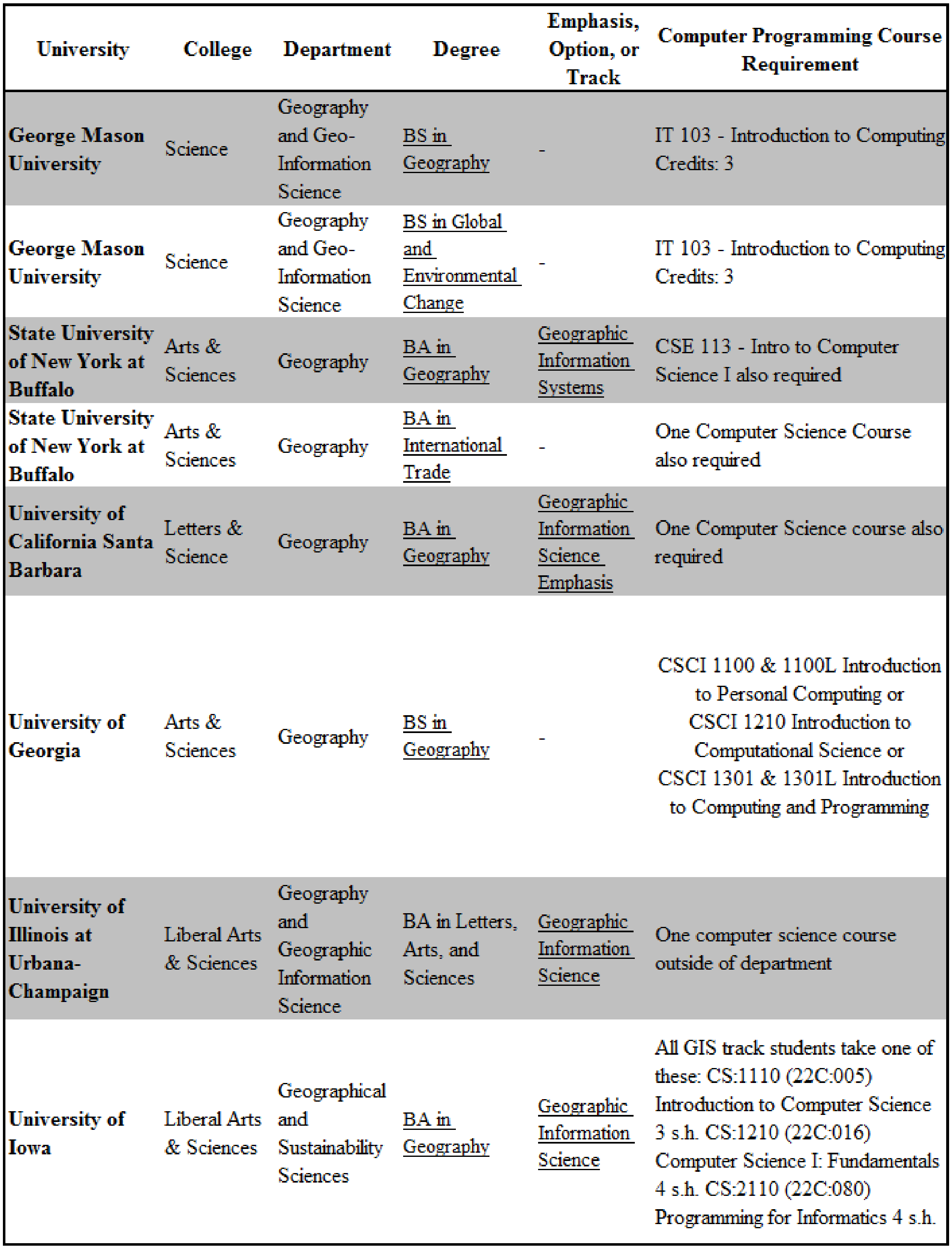


University	Degree: Emphasis, Option or Track
Arizona State University	BS in Geography: Geographic Information Science Certificate; BS in Geography: Urban Studies Concentration with Geographic Information Science Certificate
George Mason University	BA in Geography
Johns Hopkins University	BS in Environmental Engineering
Kansas State University	BA and BS in Geography
Southern Illinois University at Carbondale	BS in Geography and Environmental Resources: Environmental Sustainability, Geographic Information Science*, and Climate and Water Resources
Texas A&M University	BS in Geography - GIS
University of Arizona	BS in Geography: Geographic Information Science
University of Cincinnati	BA and BS in Geography: Environmental Emphasis; BS in Geography: GIS and Remote Sensing Emphases
University of Connecticut	BA in Geography: Geographic Information Systems; BS in Geography
University of Kansas	BS in Geography: Geographic Information and Analysis
University of Maryland College Park	BA in GIS*
University of Southern California	BS in GeoDesign
University of Wisconsin - Madison	BA or BS in Cartography and GIS
University of Wisconsin - Milwaukee	BA or BS in Geography: Geographic Information

Requirements by Course Level

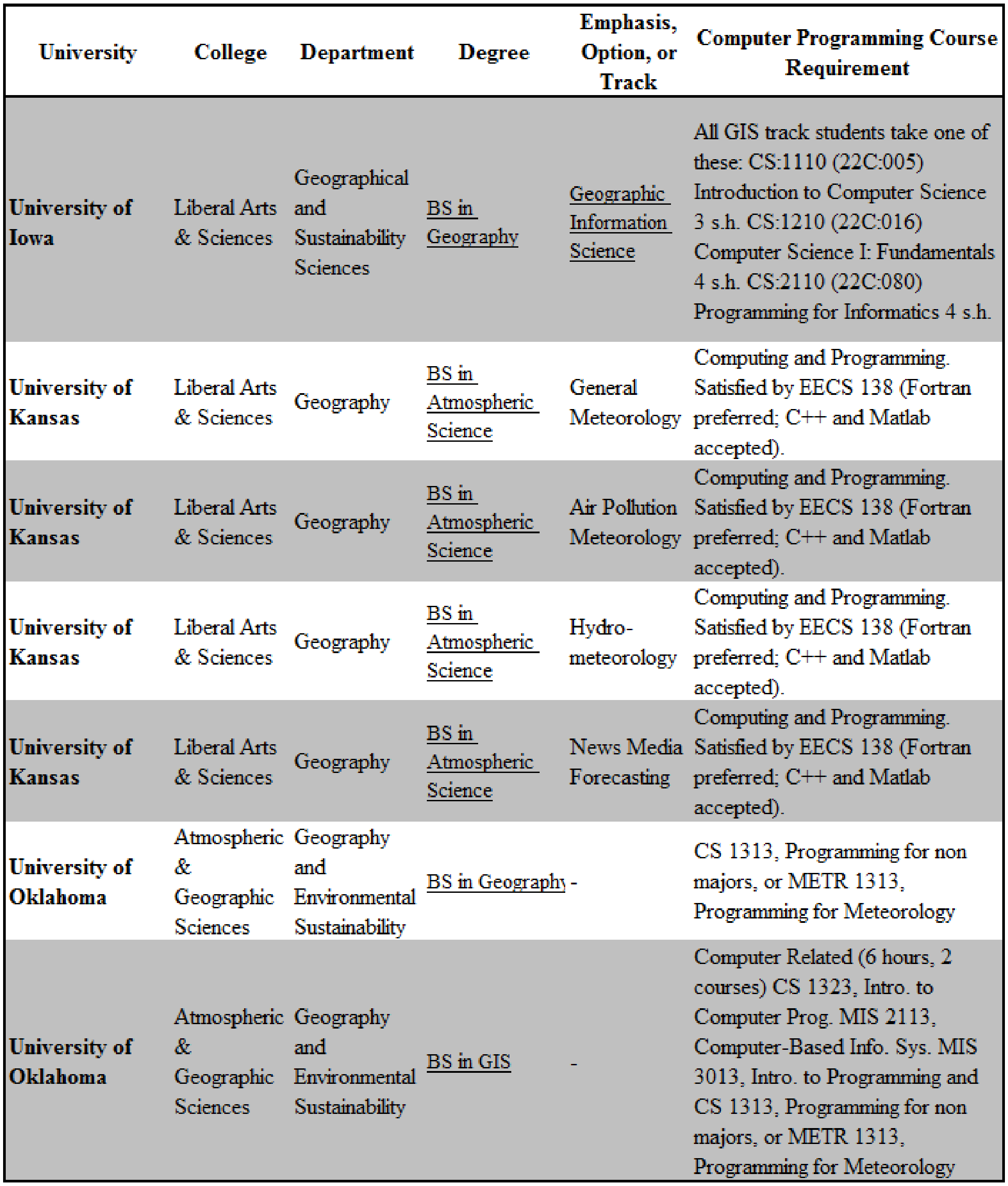
The courses required for geography and GIS degrees and degree options are primarily upper-division undergraduate or undergraduate and graduate cross-listed courses. Only one program offered an introductory course with programming components, an introductory maps and mapping course that addressed web services and geodatabases. The 15 degrees identified that required a computer science or programming course specified only introductory level computer science or computer programming courses. Table 2-5 summarizes the computer science courses required for degrees by university, showing a general preference for courses that provide a broad introduction to computer science and programming.

Table 2-5. Outside computer science and programming courses required by degree and university. Comments in the course requirement column originate from specific department notes about the requirement or commentary from the university catalog. Course numbers and titles retain their originating university's numbering and titling style, as well as any administrative comments.



University	College	Department	Degree	Emphasis, Option, or Track	Computer Programming Course Requirement
George Mason University	Science	Geography and Geo-Information Science	BS in Geography	-	IT 103 - Introduction to Computing Credits: 3
George Mason University	Science	Geography and Geo-Information Science	BS in Global and Environmental Change	-	IT 103 - Introduction to Computing Credits: 3
State University of New York at Buffalo	Arts & Sciences	Geography	BA in Geography	Geographic Information Systems	CSE 113 - Intro to Computer Science I also required
State University of New York at Buffalo	Arts & Sciences	Geography	BA in International Trade	-	One Computer Science Course also required
University of California Santa Barbara	Letters & Science	Geography	BA in Geography	Geographic Information Science Emphasis	One Computer Science course also required
University of Georgia	Arts & Sciences	Geography	BS in Geography	-	CSCI 1100 & 1100L Introduction to Personal Computing or CSCI 1210 Introduction to Computational Science or CSCI 1301 & 1301L Introduction to Computing and Programming
University of Illinois at Urbana-Champaign	Liberal Arts & Sciences	Geography and Geographic Information Science	BA in Letters, Arts, and Sciences	Geographic Information Science	One computer science course outside of department
University of Iowa	Liberal Arts & Sciences	Geographical and Sustainability Sciences	BA in Geography	Geographic Information Science	All GIS track students take one of these: CS:1110 (22C:005) Introduction to Computer Science 3 s.h. CS:1210 (22C:016) Computer Science I: Fundamentals 4 s.h. CS:2110 (22C:080) Programming for Informatics 4 s.h.

(Table 2-5 continued)



University	College	Department	Degree	Emphasis, Option, or Track	Computer Programming Course Requirement
University of Iowa	Liberal Arts & Sciences	Geographical and Sustainability Sciences	BS in Geography	Geographic Information Science	All GIS track students take one of these: CS:1110 (22C:005) Introduction to Computer Science 3 s.h. CS:1210 (22C:016) Computer Science I: Fundamentals 4 s.h. CS:2110 (22C:080) Programming for Informatics 4 s.h.
University of Kansas	Liberal Arts & Sciences	Geography	BS in Atmospheric Science	General Meteorology	Computing and Programming. Satisfied by EECS 138 (Fortran preferred; C++ and Matlab accepted).
University of Kansas	Liberal Arts & Sciences	Geography	BS in Atmospheric Science	Air Pollution Meteorology	Computing and Programming. Satisfied by EECS 138 (Fortran preferred; C++ and Matlab accepted).
University of Kansas	Liberal Arts & Sciences	Geography	BS in Atmospheric Science	Hydro-meteorology	Computing and Programming. Satisfied by EECS 138 (Fortran preferred; C++ and Matlab accepted).
University of Kansas	Liberal Arts & Sciences	Geography	BS in Atmospheric Science	News Media Forecasting	Computing and Programming. Satisfied by EECS 138 (Fortran preferred; C++ and Matlab accepted).
University of Oklahoma	Atmospheric & Geographic Sciences	Geography and Environmental Sustainability	BS in Geography	-	CS 1313, Programming for non majors, or METR 1313, Programming for Meteorology
University of Oklahoma	Atmospheric & Geographic Sciences	Geography and Environmental Sustainability	BS in GIS	-	Computer Related (6 hours, 2 courses) CS 1323, Intro. to Computer Prog. MIS 2113, Computer-Based Info. Sys. MIS 3013, Intro. to Programming and CS 1313, Programming for non majors, or METR 1313, Programming for Meteorology

Search Term Results, False Positives and Negatives

The terms selected for search within course descriptions did return a number of false positives, which required the removal of ten records from the analysis. For example, in the context of computer science the terms 'model' or 'modelling' can refer to computer models or models of workflow. However, this term in geography can refer to hydrological or environmental modeling. While a course in these types of modeling may incorporate GIS programming and application of computer science, without confirmation through deeper syllabus analysis, they were not included in the analysis unless the course description confirmed an explicit computer science or programming component.

False negatives in this analysis resulted from vague course descriptions, no description, inaccurate description or outdated descriptions. While the nature of instruction in GIS courses may change in response to technological innovation, course descriptions may remain static and not reflect what knowledge, skills, and practices are being taught. This may skew the results of this analysis. A lack of updated web-available information may be a reason for the small number of introductory courses located in the sample. For example, Texas A&M University, the home institution of the author, does not list three new programming courses on its webpage due to university constraints on website updating.

Discussion

Review of Guiding Questions

This chapter posed five distinct research questions which guided this research. I recap and discuss those questions below.

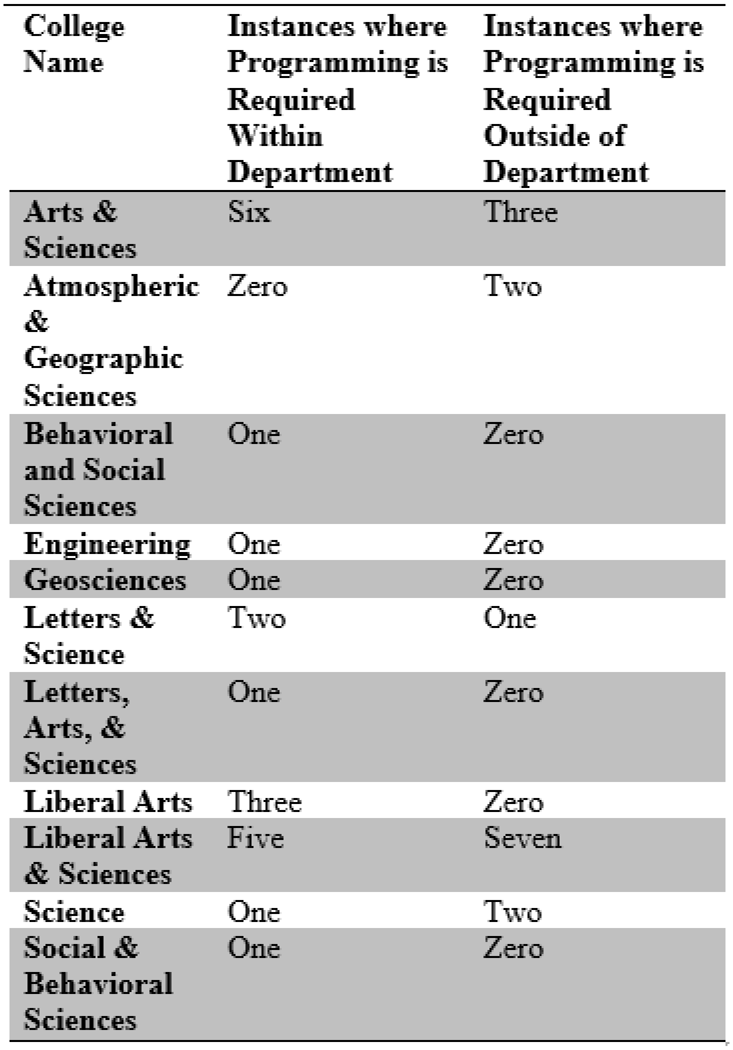
Are students acquiring computer science and programming knowledge and skills within their primary degree programs in geography or outside of them?

Generally, degrees which require a computer science or programming course are more likely to offer that course within the degree-granting department (22 instances) than outside of it (15 instances), with only one degree in this sample requiring both. This instance, however, the BA in Geography from George Mason University, does not proscribe which computer science course to take; instead, it only notes that such a course is required. Based on this work I cannot confirm whether the outside requirements are due to specific crafting of the geography degree curriculum, or tied to broader university 'core' requirements. I can speculate that this almost binary approach may indicate that the strategy of adding programming courses to the degree requirements is more a function of department and university organization, culture, and process rather than explicit curriculum and course review. Nevertheless, as programming language skills like Python are rapidly becoming essential requirements for GIS careers, the lack of required coursework in this area is a concern.

What factors influence the number of computer science or programming courses required for geography and GIS degrees?

There are no clearly evident college or program level influences on programming course requirements. Table 2-6 lists the colleges where programming-requiring departments are housed.

Table 2-6. Breakdown, by college, of degrees, options, or tracks. Course requirements for computer science and programming courses inside or outside of the degree-offering department.



College Name	Instances where Programming is Required Within Department	Instances where Programming is Required Outside of Department
Arts & Sciences	Six	Three
Atmospheric & Geographic Sciences	Zero	Two
Behavioral and Social Sciences	One	Zero
Engineering	One	Zero
Geosciences	One	Zero
Letters & Science	Two	One
Letters, Arts, & Sciences	One	Zero
Liberal Arts	Three	Zero
Liberal Arts & Sciences	Five	Seven
Science	One	Two
Social & Behavioral Sciences	One	Zero

The amalgamation of these colleges (e.g. 'Letters, Arts, and Sciences') precludes much depth of analysis. Department focuses and university requirements, like faculty research areas, and liberal arts or common 'cores', are likely bigger influences on the presence of outside computer science and programming courses in the geography degree. That degree requirement in this rapidly changing field could be governed by rarely addressed and seldom edited administrative oversight indicates a more proactive review and regular revision of degree requirements would benefit students in these programs.

Do Bachelor of Science (BS) degrees require more computer science and programming instruction than Bachelor of Arts (BA) degrees?

Given the wide variety of degree types, degree options, emphases, and the overall construction of degrees, it is not unexpected that large varieties of programming requirements exist. Of particular interest is the difference between a BS degree and a BA degree in geography. Many universities offer both, (29 of the 55 universities in the sample), sometimes with identical tracks for both options. More often, however, different degree options rest inside the disparate degrees: 'Human Geography' as a BA degree, 'Physical Geography' as a BS, for example. GIS tracks are split in a fairly even fashion, with 20 GIS tracks as parts of BS degrees, and 16 GIS tracks as parts of BA degrees.

Often, the BS degrees I investigated required or recommended biology, advanced mathematics, or computer science in the broad 'core' of the degree plan. These are courses taken by all BS majors at a university regardless of specific degree sought. For BA degrees, requirements included competency in a foreign language or world culture courses. For the GIS student, then, exposure to computer science and programming may be more likely within a BS degree, no matter the major or track requirements within the department. The situation and tradition of the department however, is likely a greater influence on whether the geography degree in question is a BA or a BS regardless of the content of that degree.

Do GIS degrees or GIS-specific degree tracks or options require more computer science and programming courses than non-GIS degrees or tracks in the same department?

Overall, 13 of the 22 degree program options identified that require a computer science or programming course offered within the department are GIS-specific, either in degree type or emphasis. Six of the 15 tracks requiring an outside computer science or programming course result in a GIS-specific degree or degree emphasis. Generally, specialized degree or degree tracks with an emphasis in GIS are more likely to require a programming course, while other emphases, especially in Human Geography, are more likely to accept such a course as an option in the degree plan. While many degree plans offer ‘any’ course of a certain level to complete a student’s degree, without knowing what courses students are taking to fulfil that ‘any’ requirement, the degree of penetration of programming courses remains unclear.

Where are computer science and programming coursework typically positioned in the curriculum?

My findings indicate that computer science and programming courses are either taught at the introductory level through an outside department, or as a junior or senior level course within the department. The sequencing of these courses is not clear, however. Do students take the introductory computer science course at the beginning of their degree programs, or do they wait until they are deeper in their major requirements? When do students enroll in a geography degree program, and how does that influence their course selection? There is a clear need for further research in this area, especially considering the inputs of faculty and academic advisors on planning the scope and sequence of the GIS or geography degree. Just as spatial thinking requires specific support structures and instructional methods for student success, topics in computer science, programming, and computational thinking require a different set of these structures and methods. There is no clear trend to the introduction of computer science and programming knowledge in GIS and geography.

Additional Discussion

Web-accessible course descriptions provide a rough, coarsely scaled view of the state of programming instruction in geography and GIS programs. However, broad university requirements and disjointed support and technical capability can also result in less centralized instruction of GIS. A geography department might not be the only 'home' for GIS in a university, so specialized courses (in programming, computer architecture, or specific computer science techniques) might be offered outside of a home geography department. It is not possible to capture the important role of faculty and academic advisors in constructing individual student degree plans through course reviews. Further work involving how these stakeholders shape degree plans is necessary.

Other issues complicate this analysis. Department websites do not always reflect current course offerings. For students interested in learning more about department course offerings, or seeing updated course descriptions, these missing updates reduce the amount of information available to interested students. A similar issue arises with special topics courses. While most departments have a course described as ‘special topics’, the course topics described (or rotated) are not clear. I discovered some instances where special course topics were listed as a degree requirement, but where no course description was available.

Conclusions and Future Directions

I conducted a broad review of course descriptions in NRC and Academic Analytics ranked geography departments in the United States. In reviewing these descriptions, I discovered that most departments (44 of 55) include some type of GIS programming course in their in-department course offerings. However, of the 210 separate degree tracks I identified, only 22 (~10%) required one of these courses for completion of the degree. This lack of programming coursework may stem from many factors: a lack of department awareness of the necessity of these skills in the workforce, no source of motivated and capable instructors to teach in this fused domain, inability to manage the material requirements of these courses, like cost, computer lab space, or otherwise, and other factors, ranging from whether GIS ‘belongs’ to geography to the culture and capabilities of individual departments and instructors.

With the increasing need for graduates trained with a broad set of geospatial skills, and increasing application of geographic data, skills in computer science and programming will continue to require additional emphasis in coursework and degree programs. Determining how to best provide these skills requires attention to the course offerings and content within GIS and geography degrees. Two prominent GIS content resources exist for instructors and departments to determine where their courses and programs fit within established content realms: the Geospatial Technology Competency Model (GTCM), and the GIS&T Body of Knowledge (BoK). These consolidations of GIS practice provide an outline of GIS knowledge, skills, and practices to compare and contrast content within individual courses and program curriculum. The abundant relevant applications of computer science and programming in geography is apparent in these consolidations of knowledge, skills and practices in GIS. While the high percentage (80%) of departments with computer science or programming courses is encouraging, the low percentage (10%) requiring computer science or programming is a curious disconnect worth further investigation.

This review provides a first step into understanding the state-of-the-art in general instruction in GIS. Course descriptions provide a window to understand course offerings and degree requirements at a diverse set of highly ranked geography departments. This course-level scale is relevant in identifying broad trends in course composition and degree requirements, and can serve as one piece of a many-tiered investigation into the content and trends in the integration of computer science and programming. Courses in computer science and programming are clearly present in many geography departments, but are not commonly required for these degrees. To investigate this disparity, future work should focus on other interfaces between computer science, programming, and GIS, like how students respond to computer science and programming courses, reviewing syllabi (including learning outcomes) for content in these courses, and seeking descriptions of skills employers value in students emerging from these degree programs.

More research is necessary to further understand and assess the teaching and learning of programming and computer science in geography. I propose the following foci for work looking forward. One track of research should be conducted within classrooms teaching computer science, programming, and GIS. An evidence-based understanding is vital to determining what advantages or disadvantages might exist for students who take a computer science and programming course outside of geography compared to those who take it within their home department. Considering the differences in instruction, would geography students be better prepared with the general concepts and experience in an introductory computer science and programming course, or with the application focused instruction in a GIS programming course? Is there a set of key skills, actions, or activities of the GIS programmer? Any research and evidence-based documentation of these components would be incredibly valuable to GIS instruction. Additional classroom-based research concerning student learning through different instructional methods would bring important evidence into addressing the best practices in integrating computer science, programming, and GIS instruction.

Another beneficial track of research would consider the content from course syllabi to determine what ideas, concepts, methods, learning outcomes, and so forth, are present in state-of-the-art classes. Since there is no widely agreed upon set of skills or practices to be taught in GIS programming or computer science courses related to GIS, an understanding of existing content would provide guidance for building standards in practice or content to inform instruction or realign course and program content. I undertake such a review in Chapter Four, but repeated studies and varying viewpoints would add valuable viewpoint diversity to this effort. Further research tracks should consider computer science and programming knowledge, skills, and practices in GIS and geography careers, and where academic preparation matches or falls short of professional expectations. The role of academic advisors or faculty in guiding students through their degree plans, selecting courses, and building their academic skills would be a valuable area of focus. There are numerous areas of GIS and geography education research that would be broadly beneficial within this realm.

GIS is the fundamental tool for spatial analysis in geography, yet the core components of how a GIS functions, the computer science and programming concepts, remain inconsistently taught within geography and GIS degree granting departments. Future work should examine the implications of this disconnection, as well as undertake deeper investigations into the structure of these courses. This work must operate on numerous scales, from studying learners within computer science, programming, geography, and GIS classrooms, to analyzing course and degree structures, to academic guidance, professional expectations, and more. Collaborations between geography and GIS education researchers and those investigating computer science education should build cross disciplinary, theory-rich observations beneficial to both fields. This overview takes an important first, but by no means final, step into understanding how geography and GIS courses structure this vital computer science and programming content.

CHAPTER III

VALUABLE COMPONENTS OF CYBERGIS: EXPERT VIEWPOINTS THROUGH Q-METHOD INTERVIEWS

Geographic Information Systems and Science (GIS and GISci) are evolving into Cyber Geographic Information Science (CyberGIS). Traditionally, GIS relied on desktop computers, locally installed software, and powerful hardware to solve complex spatial problems. CyberGIS changes GIS operations by leveraging the advantages of advanced cyber infrastructure (CI). These CI advantages include the ability to process immense data sets, and to analyze these data through computation techniques like parallel processing. Most significantly, CI has changed where analysis occurs; utilizing cloud technologies to deliver GIS via the web. CyberGIS capitalizes on cutting-edge concepts and technologies from computer science and offers users a range of new capabilities. Given the fusion of traditional spatial analysis, traditional GIS components, and advanced concepts in computer science and programming, the scope of potential knowledge and abilities in CyberGIS is immense. The concepts, skills, and practices required to utilize this innovative technology are not well defined or understood.

There is little discussion or research focused on effective ways to prepare students for competency in this cyber-frontier of GIS. This chapter attempts to explore this topic. I begin with a review of CyberGIS, including the foundational components of CyberGIS that distinguish it from traditional spatial analysis. Then, I discuss what it means to be an expert, how expertise develops, and how individuals perceive their learning processes. I then use q-methodology to perform an assessment of the key concepts and practices in CyberGIS using a sample of GIS experts. I investigate three general questions researching the nature of expertise in CyberGIS:

1. What components of CyberGIS are most prominent in the published literature?
2. How do experts value these components based on their experience?
3. How do experts conceptualize CyberGIS?

The purpose of this research is to encourage and direct future teaching and research efforts in CyberGIS and GIS education. I conclude by describing an outline for future research directions and opportunities within this topic.

Defining CyberGIS and Exploring Expertise

CyberGIS

CyberGIS is a synthesized and interdisciplinary field where the individual components of cyberinfrastructure, GIS, and spatial analysis merge (Wang 2010) (See Figure 3-1).

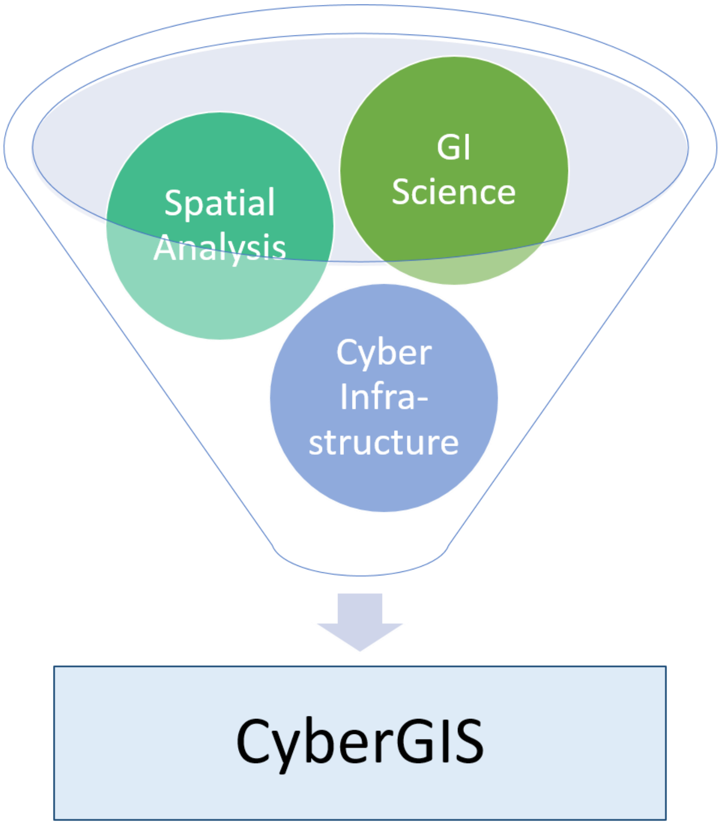
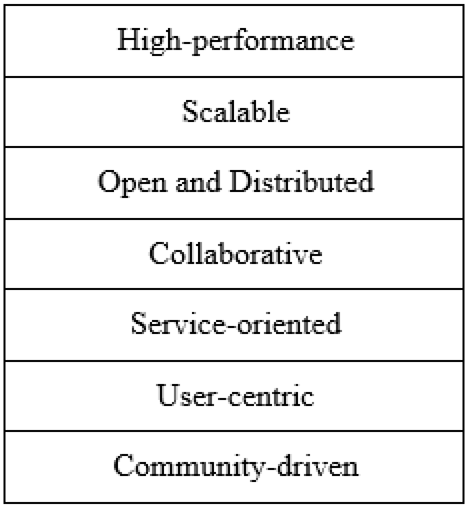


Figure 3-1. CyberGIS components. CyberGIS is a rapidly developing field emerging from concepts and applications in traditional desktop GIS, geographic and spatial analysis, and multiple topics in computer science, including advanced cyberinfrastructure, supercomputers and parallel processing (Wang et al. 2013). Though CyberGIS has a rich technical background, and geography, GIS, and computer science education research forms a background for instruction, there is sparse research educationally for CyberGIS.

The merger of these components allows advanced spatial analysis and has resulted in advances in in GIS (Nyerges et al. 2013). CyberGIS can integrate multiple forms of data at a range of scales through highly powerful processes to reveal spatial relationships that would otherwise be too time or resource consuming to discover. As a result, CyberGIS presents revolutionary new opportunities for geospatial applications. This is especially important as an increased amount of high quality and reliable spatial data has become available, facilitating decision making and problem solving at wider scales of analysis. (Goodchild 2007). Table 3-1 outlines the characteristics CyberGIS exhibits which form the basis for its structure and synthesis.

Table 3-1. Key Characteristics of CyberGIS (Wang et al. 2013). CyberGIS distinguishes itself from its component fields, and especially traditional GIS, by exhibiting these characteristics.



High-performance
Scalable
Open and Distributed
Collaborative
Service-oriented
User-centric
Community-driven

High-performance computing incorporates grid, parallel, and cloud frameworks to improve GIS capabilities, particularly in conducting simulations (Kim and Tsou 2013). Scalability in computing refers to a system's capabilities in handling growing amounts of work (Bondi 2000). Scalable computing improves performance when resources are added, and efficiently processes growing amounts of work as queries are run. Networked computers share processing tasks for greater efficiency, and such open, distributed systems are adept at processing spatial and geo-data (Jhummarwala et al. 2014). All of these characteristics function to allow CyberGIS products to complete specific workflows (or, services) developed by multiple users across a broad community of researchers and analysts. Traditional desktop GIS are incapable of solving some problems that a CyberGIS easily handles (Wang et al. 2013). Overall, managing, transforming, and analyzing data with CyberGIS requires extensive skill (Wright and Wang 2011). If CyberGIS incites a transition to a post-GISystems world (Wright 2012), future instruction will have to adapt to a GIS environment in which mastery of components of CyberGIS are the dominant elements to emphasize.

CyberGIS requires a different set of knowledge, skills, and practices than traditional GIS. As GIS moves forward adapting the affordances of CyberGIS, GIS educators are considering needed changes in instruction, particularly in three areas. First, the CyberGIS environment requires a deeper and broader awareness of analytic techniques, as it is an open and collaborative interaction with GIS as opposed to the closed and 'monolithic' traditional GIS (Wang et al. 2013). Second, CyberGIS requires a user-centered and collaborative mindset for success (Yang et al. 2011), which cloud computing and open source software facilitate. Finally, Expertise in CyberGIS not only requires knowledge of traditional GIS, but also aspects of computer science and collaborative design (Nyerges et al. 2013). Collaborative design manages multiple perspectives from many disciplines to share expertise and build understanding in creating a new product (Du et al. 2012).

The structural and computational components of CyberGIS have been thoroughly explicated in research literature. The educational implications have not, however, creating a large gap concerning the educational implications of this evolving system. This research reviews expert viewpoints on key components of CyberGIS to benefit future instructional and research efforts by outlining potential viewpoints and examining how CyberGIS topics function within those viewpoints.

Development of Expertise

One approach to developing new curricula to prepare students to be competent in CyberGIS is to begin at the end, that is, by understanding the knowledge, skills, and practices of expert users and planning educational experiences backward. Definitions of expertise are not straightforward, vary based on field and specialty, and contain numerous definitions and descriptions (Day 2002, Farrington-Darby and Wilson 2008). As a general rule, experts exhibit certain characteristics, traits, and actions that allow them to complete tasks in their domain with greater accuracy and efficiency than non-experts, or novices. I seek to capitalize on this expert viewpoint to understand how components of CyberGIS are used, valued, and understood to provide evidence for future educational applications. In this section, I review research on expertise in general, considering the history of expertise research especially as relevant to science, geography, and computer science. I also explore the instructional methods that help progress learners on the path to expertise. Considering the nature of expertise, I finish with key insights relevant to this article's focus on CyberGIS.

The study of experts attempts to define the outstanding qualities that distinguish some individuals in a domain from other non-outstanding individuals (Ericsson and Smith 1991). In all domains, from studies of world-class performers in music and chess, to understanding performance in sports and medicine, high achievement is controlled by practices developed in training and development (Ericsson 2005). This extended temporal training allows experts to perform at a high level in their domain by drawing on their years of domain experience (Foley and Hart 1992). This performance includes understanding and applying different knowledge types (i.e. explicit, implicit, and conceptual) in a greater diversity of contexts and situations (Postigo and Pozo 2004). In their ability to complete tasks without elaborate referencing, experts capitalize on their extensive knowledge and memory to recall and solve problems (Sternberg and Frensch 1992).

Expertise in a subject or skill develops over time, and expert status earned through an individual's qualifications, track record, and experience (Burgman et al. 2011). Throughout the progression from novice to expert, time and experience are the primary factors in mastering a subject (Hoffman et al. 1995), structured by deliberate practice (Ericsson 1998). Deliberate practice is the long-term, sustained focus on gaining new knowledge and skills through critical analysis and feedback (Ericsson 1998), connecting with an expert's foundational knowledge of a subject through goal-directed tasks (Hambrick and Meinz 2013, Ericsson 2009). Determining how experts address non-routine, challenging problems, as well as outlining the tasks that capture deliberate practice and critical expertise, allow for the clear development of expert approaches (Ericsson et al. 2009).

Novices address problems and organize domain knowledge in different ways than experts (NRC 2000). Experts exhibit key differences, including noticing features and meaningful patterns, remaining flexible in their approaches to new situations, applying and retrieving their knowledge conditionally based on circumstance, and having a deep understanding of content. Experts draw on complex relationships between multiple skills and knowledge phases (Schmidt and Boshuizen 1993), so a single metric of analyzing talent or skill within a domain is insufficient to determine expert status (Ackerman 2014). Therefore, a broad focus on the actions that help to develop expertise in geography, GIS, and CyberGIS has merit (Downs 1994).

While deliberate practice structures expert growth, the delivery method for knowledge and concepts matters as well. Generally, active learning constructs encourage novices to engage with content and reduce learning stress, as Fee and Holland-Minkley (2010) found in computer science. Problem-based learning (PBL) also serves as a meaningful way to encourage expertise to build (Boshuizen 2009), often with well-constructed curricula and texts (Bransford and Schwartz 2009). While experts can suffer from a 'blind spot' when discussing, teaching, or designing instructional materials in their domain of expertise, mixed instructional methods and high quality feedback can reduce this concern (Nathan and Petrosino 2003). The overall development of expertise is a social process, in which people who are motivated to learn something have access to relevant teaching expertise and manage their struggle to improve through high-quality teacher-learner feedback (Bransford and Schwartz 2009).

The expert geographer or GIS user is undefined (Downs 2014). Expert knowledge influences practice and decision making (Brooks 2010), but determining and developing expert knowledge is an immense challenge. While metrics for assessing geographic content knowledge, like the GIS Body of Knowledge (BoK) exist, they function more to structure and organize concepts, terms and ideas rather than provide a framework for expert progression. Huynh and Sharpe (2013) outline a considerable number of tasks, tools, and metrics to assess components of such knowledge, but only begin preliminary investigations into what constitutes expert performance. Experts must balance subject, technical, ethical (Huff 2014), and other types of expertise in their use of GIS, and this expertise might not be evenly developed, as expertise develops across skills unevenly (Breβler 2012).

In geography and in other domains, estimates of overall time to develop expertise are less important than the use of that time. As Downs (2014) argues, even the time involved in earning a PhD may not be enough time to develop a 10,000-hour base of expertise in geography programs. Identifying the common cores of expertise across the wide knowledge base of GI scientists is an expansive and difficult undertaking. Duckham (2015) proposes five cores of expertise for the GI scientists, including an understanding of the structure of geographic information, the nature of spatial uncertainty and dynamism, aspects of geodesign, and components of geographic language and cognition. While these broad categories may apply to areas of general expertise in GIS, the topics within these areas remain unclear. This research uses GIS user viewpoints to understand the perceived value of components of CyberGIS.

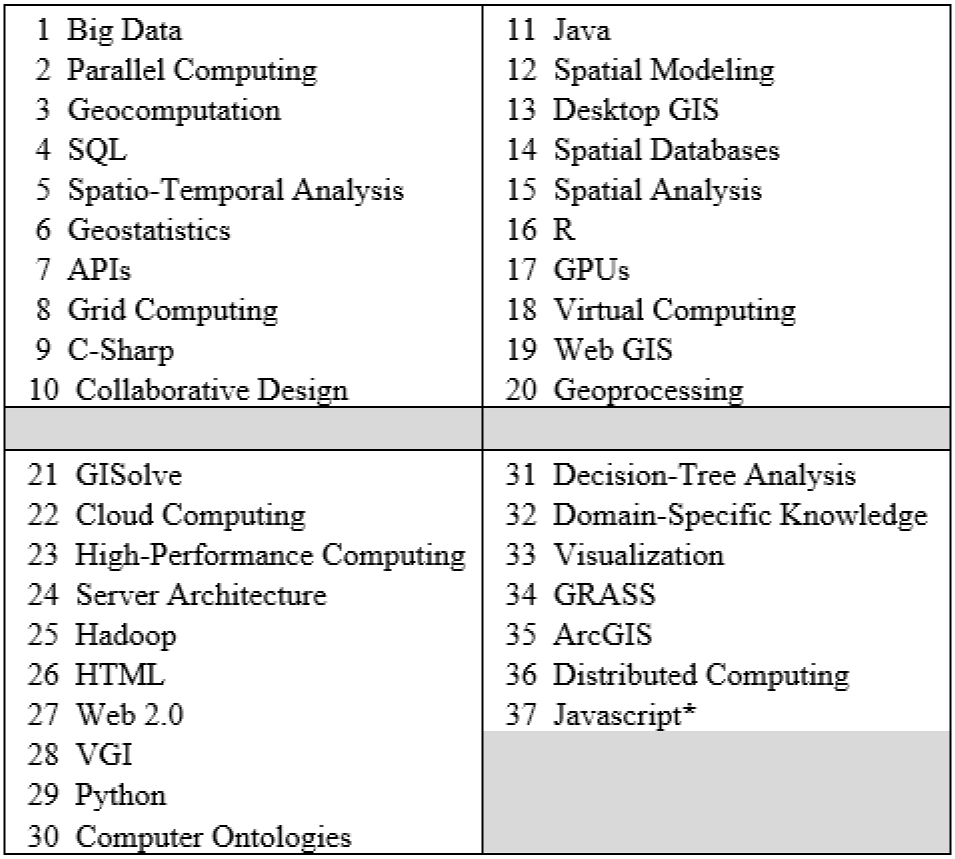
Methodology

Selection of Terms

I began researching on expertise in CyberGIS by developing a set of terms derived from an analysis of CyberGIS literature. I first collected a set of 40 peer-reviewed academic papers concerning CyberGIS. This set included any paper using 'CyberGIS' as a keyword, published in an academic journal, as well as background foundational papers listed by the CyberGIS Center for Advanced Digital and Spatial Studies, the core research, development, and funding organization for CyberGIS activities. These papers are referenced in Appendix I.

I analyzed these articles as converted .txt files through the Machine Learning for Language Toolkit (MALLET). MALLET allows for a statistically based, natural language form of processing, and produces topic models (McCallum 2002). These models are connections of ideas, shared across research articles. After removing journal and author names, 36 terms emerged. These terms, representing key programming languages, computing approaches, analytical methods, and terminology in CyberGIS, capture the key topics of CyberGIS. I added an additional term after preliminary survey testing and sample validation surveys. This set of terms is shown in Table 3-2 below.

Table 3-2. Topic modeling term extraction. The set of terms extracted through topic modelling in MALLET, save for the term marked with a star (*), which was added after preliminary validation efforts. Numbers correspond to the random coding of the term for analysis, and should not be read as an implied hierarchy. These items formed the set of terms in the subsequent q-method surveys.



1	Big Data	11	Java
2	Parallel Computing	12	Spatial Modeling
3	Geocomputation	13	Desktop GIS
4	SQL	14	Spatial Databases
5	Spatio-Temporal Analysis	15	Spatial Analysis
6	Geostatistics	16	R
7	APIs	17	GPUs
8	Grid Computing	18	Virtual Computing
9	C-Sharp	19	Web GIS
10	Collaborative Design	20	Geoprocessing
21	GISolve	31	Decision-Tree Analysis
22	Cloud Computing	32	Domain-Specific Knowledge
23	High-Performance Computing	33	Visualization
24	Server Architecture	34	GRASS
25	Hadoop	35	ArcGIS
26	HTML	36	Distributed Computing
27	Web 2.0	37	Javascript*
28	VGI		
29	Python		
30	Computer Ontologies		

Selection of Participants

For this study, I recruited active GIS researchers, developers, and instructors with extensive GIS experience to participate. I surveyed and interviewed 20 subjects, complying with recommendations of experts in q-method research (Watts and Stenner 2012). These 20 participants attended or presented at prominent national GIS conferences in the United

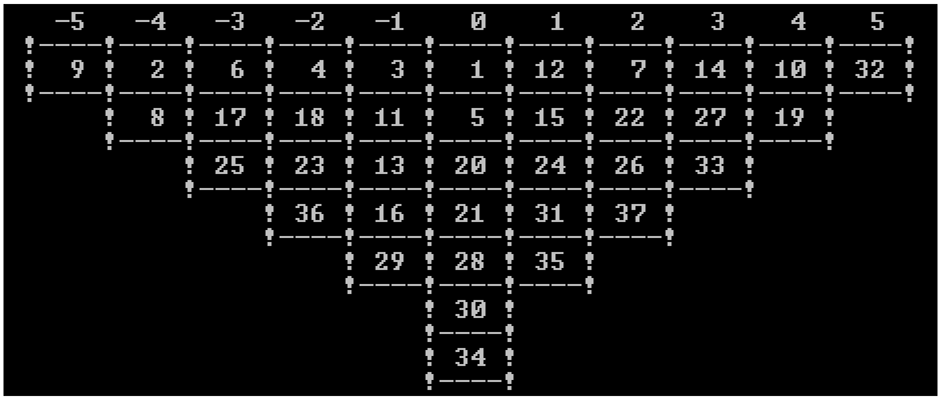
States: the 2015 ESRI International Developers Conference (March 10 – 13), GeoComputation 2015 (May 20 – 23), and the 2015 University Consortium of GIS (UCGIS) Symposium (May 28 – 30). A supermajority of participants were from the 2015 UCGIS symposium, but specific counts of subjects obtained from each conference cannot be provided due to anonymity concerns. Subjects volunteered to participate, and received no compensation for their time.

Q-Methodology

Q-methodology allows a multi-method approach to understanding subjectivity, with a mix of strongly quantitative and qualitative evidence. Q-method is a strategy to devise a quantitative assessment of subjective rankings (Wright 2013), to construct statistically validated rankings. Such evidence-based data is valuable to geography education research (Wright 2012). In q-method research, subjects rank a set of terms (the q-sort, as shown in Table 3-2 above), based on their evaluation of the concepts within the set. Subjects are forced in this methodology to assign ranked value to these terms, creating a continuum that represents how they perceive the terminology as operating in their sense of the subject matter. Through subsequent validation interviews, subjects then explain and discuss their ranking choices, allowing for a qualitative validation of their quantitative rankings.

In this case, I asked subjects to rank the terms following a near-normal distribution (Figure 3-2) corresponding to the importance of each in CyberGIS based on their expertise and experience. Subjects specified which terms were of relative greatest and least value to them by completing this ranking exercise. Figure 3-2 summarizes the results. Participants ranked all 37 terms along this continuum. Numbers below the value spectrum indicators represent the codes for the selected terms as shown in Table 3-2. These codes were randomly generated and do not imply a hierarchy or indicate abundance in the sample. To ease the process of ranking terms, participants first ranked terms into three general groups: valuable, somewhat valuable, and not valuable. This facilitated an easier subsequent sorting into these distinct groups.

For example, this subject ranked term 32, 'Domain-Specific Knowledge,' as the most valuable term in their conception of the field. After ranking all of the terms, participants completed a semi-structured validation interview discussing, explaining, and elaborating on their choices. The questions probed for specific examples defending their selections, and how the concepts fit into their career, and elaborations on how the process helped reveal their viewpoint, among other topics. These interviews after the q-sort ranking exercise helped validate their rankings, and allowed for an in-depth discussion of how their expertise interfaced with these CyberGIS terms.



```
  -5   -4   -3   -2   -1    0    1    2    3    4    5
!----!----!----!----!----!----!----!----!----!----!----!
!  9 !  2 !  6 !  4 !  3 !  1 ! 12 !  7 ! 14 ! 10 ! 32 !
!----!----!----!----!----!----!----!----!----!----!----!
     !  8 ! 17 ! 18 ! 11 !  5 ! 15 ! 22 ! 27 ! 19 !
     !----!----!----!----!----!----!----!----!----!
          ! 25 ! 23 ! 13 ! 20 ! 24 ! 26 ! 33 !
          !----!----!----!----!----!----!----!
               ! 36 ! 16 ! 21 ! 31 ! 37 !
               !----!----!----!----!----!
                    ! 29 ! 28 ! 35 !
                    !----!----!----!
                         ! 30 !
                         !----!
                         ! 34 !
                         !----!
```

Figure 3-2. Sample q-sort. A sample q-sort (sort number ten), showing a sample ranking as transcribed into the analysis program, and the ranking distribution of terms in the study. '-5' represented the term of least value, and '5' represented the term of greatest value, as perceived by each participant. Note that vertical location has no influence on value. All terms ranked in '-1', for example, are given that weight during analysis, regardless of their vertical place in the column.

Analysis

I analyzed the q-sorts using a stand-alone q-method analysis program. This process involves transcribing photographs of the participant's q-sorts through a command line interface, then performing the necessary statistical analysis to create appropriate factor scores for the component terms. I followed the following steps in producing the analysis:

1. After entering numerical sort data into a database, I performed a principal components analysis to derive a correlation matrix with appropriate eigenvalues for selection factors.
2. Then, I performed a varimax rotation of these factors to extract the three factors for analysis.

3. Finally, I flagged defining components of the sorts of these factors, to complete the analysis.

I selected three factors for this analysis, as each of these factors contributed at least 10% to the overall variance in the sample. While I did investigate other factor distributions, the three factor solution formed the simplest and clearest interpretation of the results, per Webler et al.'s (2009) recommendations. While 20 total participants completed the study, two refused to sort some of the terms, and their sorts were removed from the analysis as outlined above. Their statements remain present in the subsequent validation interviews, however.

Results

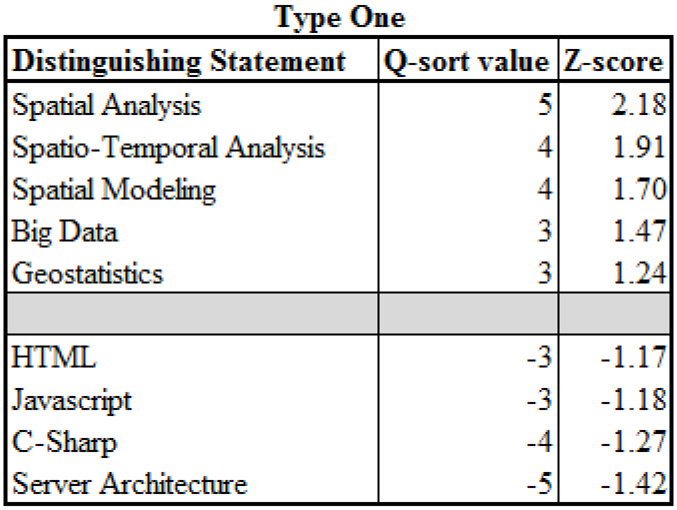
Extracted Viewpoints

I extracted three factors which accounted for 57.8% of the variance in this sample. These factors correspond to three distinct conceptualizations of the expert GIS users in the sample. I interpret these empirically derived viewpoints, or types, as follows:

1. The spatial analyst (traditional GIS user);
2. The domain-specific problem solver;
3. The CyberGIS-enabled computer scientist.

The spatial analyst values the modeling and analytic capabilities within CyberGIS, which help them to process large amounts of data. They find least value in the underlying architecture and programming language that facilitates advanced forms of this analysis within the CyberGIS framework, as shown in Table 3-3.

Table 3-3. Distinguishing statements for type one. Q-sort value indicates the hypothetical position of each statement in a sort representative of this consolidated factor. All of these statements are significant at P < .01. The spatial analyst values most the traditional spatial and analytical capabilities in CyberGIS, as well as access to big data. The technical underpinnings, like programming languages and architectures, which facilitate this analysis, are less valuable.

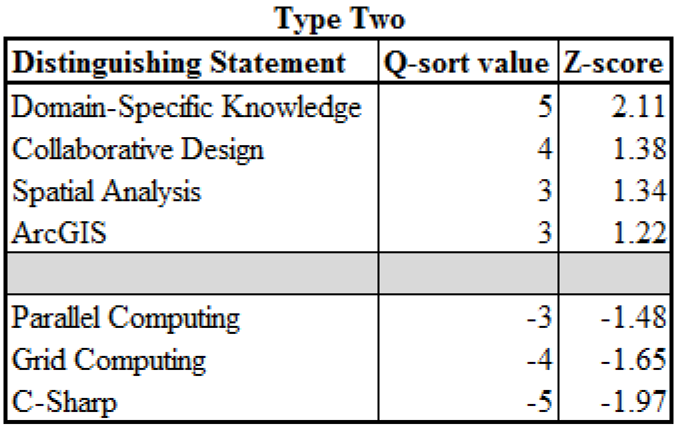


Type One

Distinguishing Statement	Q-sort value	Z-score
Spatial Analysis	5	2.18
Spatio-Temporal Analysis	4	1.91
Spatial Modeling	4	1.70
Big Data	3	1.47
Geostatistics	3	1.24
HTML	-3	-1.17
Javascript	-3	-1.18
C-Sharp	-4	-1.27
Server Architecture	-5	-1.42

The domain-specific problem solver values the knowledge and tools which facilitate their investigation of some problem in a specific knowledge domain. Through fundamental spatial analysis, they design methods to answer specific questions in their domain. The specifics of the computing required to complete such analysis are least valued, as shown in Table 3-4.

Table 3-4. Distinguishing statements for type two. Q-sort value indicates the hypothetical position of each statement in a sort representative of this consolidated factor. All of these statements are significant at P < .01, except for Grid Computing, which is significant to P < .05. The domain-specific problem solver values most concepts and tools that help them to address problems specific to their knowledge domain, while not valuing the specific computing components behind that analysis.

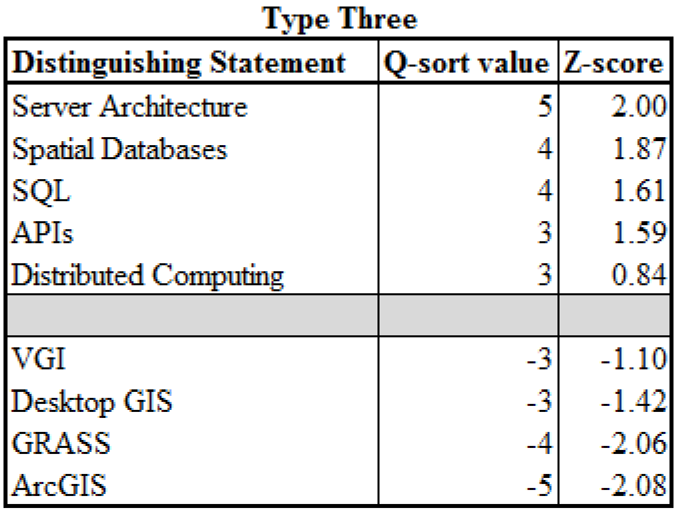


Type Two

Distinguishing Statement	Q-sort value	Z-score
Domain-Specific Knowledge	5	2.11
Collaborative Design	4	1.38
Spatial Analysis	3	1.34
ArcGIS	3	1.22
Parallel Computing	-3	-1.48
Grid Computing	-4	-1.65
C-Sharp	-5	-1.97

The CyberGIS-enabled computer scientist values the technical underpinnings that facilitate advanced spatial analysis. Databases and servers seem especially valuable facilitators in this viewpoint. Conversely, GIS software, whether open source, industry-standard, or generally desktop-locked, are seen as least valuable in this viewpoint, as shown in Table 3-5.

Table 3-5. Distinguishing statements for type three. Q-sort value indicates the hypothetical position of each statement in a sort representative of this consolidated factor. All of these statements are significant at P < .01. The CyberGIS-enabled computer scientist values most the technical components that enable advanced spatial analysis, while valuing less the precise software that runs the analysis. They also do not value Volunteer Geographic Information (VGI), perhaps due to its unreliability.



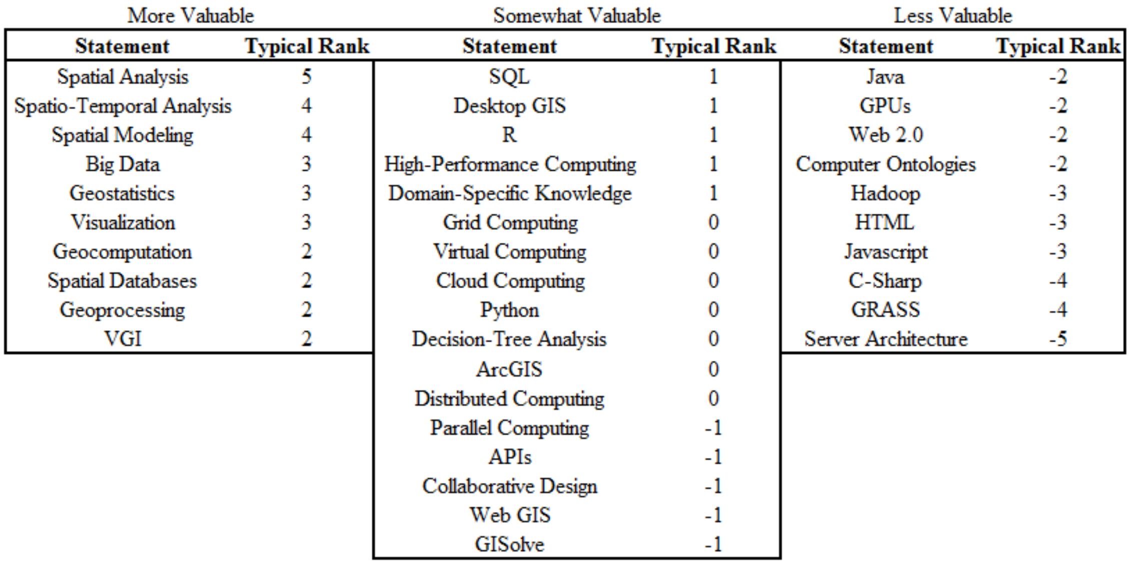
Type Three

Distinguishing Statement	Q-sort value	Z-score
Server Architecture	5	2.00
Spatial Databases	4	1.87
SQL	4	1.61
APIs	3	1.59
Distributed Computing	3	0.84
VGI	-3	-1.10
Desktop GIS	-3	-1.42
GRASS	-4	-2.06
ArcGIS	-5	-2.08

Ranking of Terms

Q-method analysis produces a 'typical' sort for each type based on the viewpoints of the participants. Each of these sorts represents a typical distribution of the terminology based on the factor analysis. While no sort matches these ranks precisely, they do typify the patterns and rankings of participants in the study, and provide a useful lens to understand the relationships between components within each type. Table 3-6 shows a typical sort for the first type, the spatial analyst. By leveraging analytical techniques with spatially focused computing power, data, and data sources, practitioners adhering to this view seek to solve spatial problems. Some tools, like the R programming language, Python, and traditional desktop GIS are less valued but still relevant to this perspective. Other languages, like C-Sharp and JavaScript, as well as technical details like server construction, are not viewed as valuable.

Table 3-6. Type one q-sort values. As noted in Table Three, the spatial analyst values most the traditional spatial and analytical capabilities in CyberGIS. However, they appear to complement this spatial core with additional methods and tools for data analysis, like geoprocessing. The components of computing the analysis show less relative utility, but are still rated as more valuable than the technical computer languages and ontological specifics.

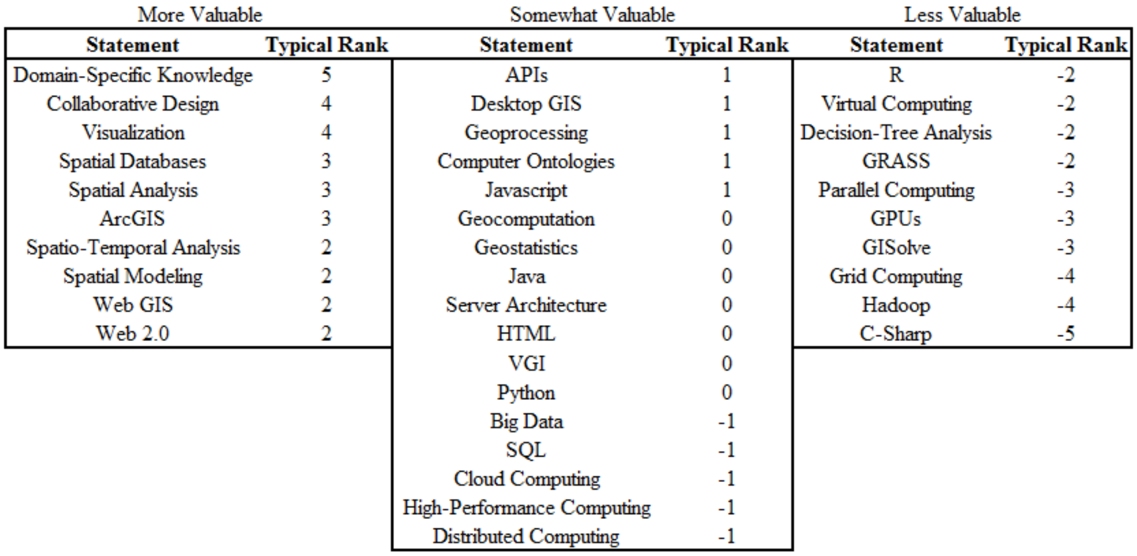


More Valuable		Somewhat Valuable		Less Valuable	
Statement	**Typical Rank**	**Statement**	**Typical Rank**	**Statement**	**Typical Rank**
Spatial Analysis	5	SQL	1	Java	-2
Spatio-Temporal Analysis	4	Desktop GIS	1	GPUs	-2
Spatial Modeling	4	R	1	Web 2.0	-2
Big Data	3	High-Performance Computing	1	Computer Ontologies	-2
Geostatistics	3	Domain-Specific Knowledge	1	Hadoop	-3
Visualization	3	Grid Computing	0	HTML	-3
Geocomputation	2	Virtual Computing	0	Javascript	-3
Spatial Databases	2	Cloud Computing	0	C-Sharp	-4
Geoprocessing	2	Python	0	GRASS	-4
VGI	2	Decision-Tree Analysis	0	Server Architecture	-5
		ArcGIS	0		
		Distributed Computing	0		
		Parallel Computing	-1		
		APIs	-1		
		Collaborative Design	-1		
		Web GIS	-1		
		GISolve	-1		

Type One Typical Rankings

Table 3-7 shows a typical sort for the second type, the domain-specific problem solver. Notably, spatial analysis terms are highly ranked as in the spatial analyst type, but domain-specific knowledge is considered more valuable. Spatial computing methods or tools which address domain problems are less valued, but may be applicable in certain contexts or considerations. Again, technical details and certain computer programming languages are viewed as less valuable components.

Table 3-7. Type two q-sort values. As noted in Table Four, the domain-specific problem solver values most concepts and tools that help them to address problems specific to their knowledge domain. They complement this problem solving with tools and methods of analysis to benefit their investigation. Technical computer science details, and methods of computing, are of less concern, with certain components like C-Sharp and Hadoop being the least useful of the broad suite of components.

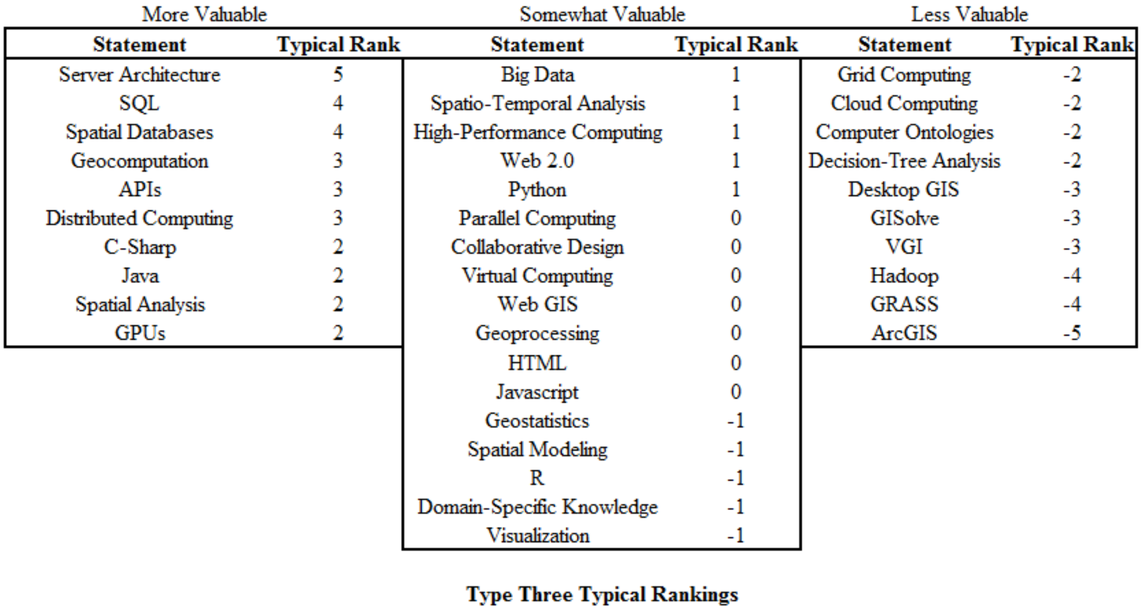


More Valuable		Somewhat Valuable		Less Valuable	
Statement	**Typical Rank**	**Statement**	**Typical Rank**	**Statement**	**Typical Rank**
Domain-Specific Knowledge	5	APIs	1	R	-2
Collaborative Design	4	Desktop GIS	1	Virtual Computing	-2
Visualization	4	Geoprocessing	1	Decision-Tree Analysis	-2
Spatial Databases	3	Computer Ontologies	1	GRASS	-2
Spatial Analysis	3	Javascript	1	Parallel Computing	-3
ArcGIS	3	Geocomputation	0	GPUs	-3
Spatio-Temporal Analysis	2	Geostatistics	0	GISolve	-3
Spatial Modeling	2	Java	0	Grid Computing	-4
Web GIS	2	Server Architecture	0	Hadoop	-4
Web 2.0	2	HTML	0	C-Sharp	-5
		VGI	0		
		Python	0		
		Big Data	-1		
		SQL	-1		
		Cloud Computing	-1		
		High-Performance Computing	-1		
		Distributed Computing	-1		

Type Two Typical Rankings

Finally, Table 3-8 shows the typical sort for the third type, the CyberGIS-enabled computer scientist. Here, technical terms and competencies are of most value; computing approaches, programming languages, and spatial methods seen as highly relevant. These terms continue to mix as perceived value decreases, but certain types of computing, and especially desktop based GIS systems, are valued less.

Table 3-8. Type three q-sort values. As noted in Table Five, the CyberGIS-enabled computer scientist values most the technical components that enable advanced spatial analysis. The programming languages that facilitate analysis are of more value in this viewpoint, though some fall lower on the value continuum. Specific computing construct, as well as GIS software, have low value in this view.



More Valuable		Somewhat Valuable		Less Valuable	
Statement	**Typical Rank**	**Statement**	**Typical Rank**	**Statement**	**Typical Rank**
Server Architecture	5	Big Data	1	Grid Computing	-2
SQL	4	Spatio-Temporal Analysis	1	Cloud Computing	-2
Spatial Databases	4	High-Performance Computing	1	Computer Ontologies	-2
Geocomputation	3	Web 2.0	1	Decision-Tree Analysis	-2
APIs	3	Python	1	Desktop GIS	-3
Distributed Computing	3	Parallel Computing	0	GISolve	-3
C-Sharp	2	Collaborative Design	0	VGI	-3
Java	2	Virtual Computing	0	Hadoop	-4
Spatial Analysis	2	Web GIS	0	GRASS	-4
GPUs	2	Geoprocessing	0	ArcGIS	-5
		HTML	0		
		Javascript	0		
		Geostatistics	-1		
		Spatial Modeling	-1		
		R	-1		
		Domain-Specific Knowledge	-1		
		Visualization	-1		

Type Three Typical Rankings

Consensus Rankings

Each type exists with its own unique ranking of the terms of the sort. However, terms share similarity or dissimilarity across these types. Table 3-9 below ranks the terms by the variance of their z-scores, identifying terms with consensus and disagreement. The terms with major differences may help define the components most unique to specific types, while the consensus terms represent topic of consistent value.

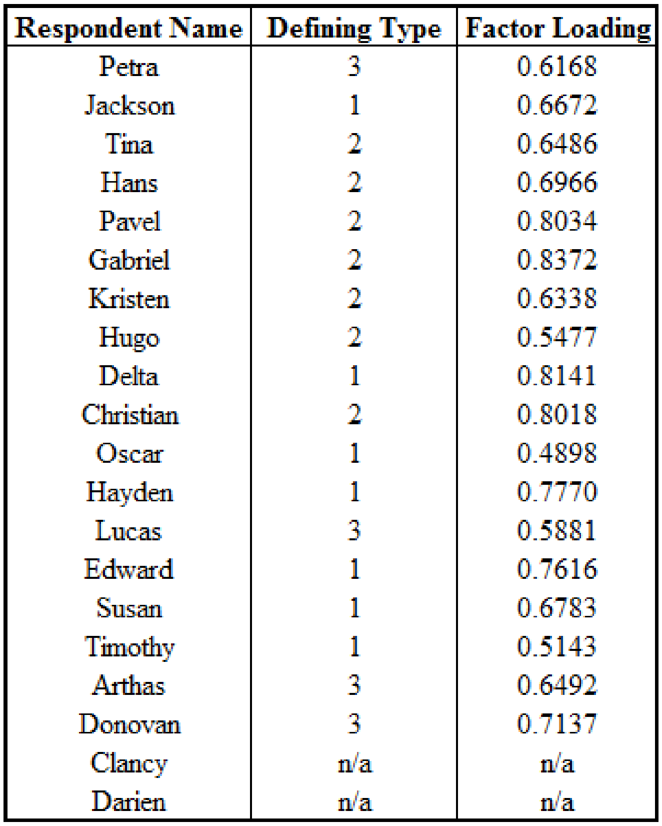
Discussion

I conducted extensive post-sort interviews with participants to build a qualitative understanding of their viewpoints to supplement the quantitative q-method analysis. These interviews provided additional evidence to confirm the results of the factor interpretation. For each type, I provide context from typical sorts (see Table 3-10 for participant information), as well as commentary on specific components.

Table 3-9. Terms ranked by consensus. Terms with greater consensus, as measured by the variance across z-scores, are at the top of the table, which terms with greater disagreement are at the bottom of the table. Controversial terms at the bottom of the table show wide disagreement between factors on terminology value. For example, ArcGIS, the prominent commercial software for GIS analysis, is ranked of middle importance by factor one viewpoints, of higher importance in factor two, but of least importance of factor three. This disparity indicates broad disagreement over the relative value of ArcGIS within this set of CyberGIS terminology.

Statement	Type 1 Rank	Type 2 Rank	Type 3 Rank
Cloud Computing	0	-1	-2
Python	0	0	1
Hadoop	-3	-4	-4
Geoprocessing	2	1	0
GISolve	-1	-3	-3
Virtual Computing	0	-2	0
Geocomputation	2	0	3
GRASS	-4	-2	-4
Decision-Tree Analysis	0	-2	-2
Web GIS	-1	2	0
Spatial Databases	2	3	4
Geostatistics	3	0	-1
High-Performance Computing	1	-1	1
R	1	-2	-1
Grid Computing	0	-4	-2
Distributed computing	0	-1	3
Java	-2	0	2
HTML	-3	0	0
Spatio-Temporal Analysis	4	2	1
Spatial Analysis	5	3	2
Web 2.0	-2	2	1
Parallel Computing	-1	-3	0
VGI	2	0	-3
Visualization	3	4	-1
GPUs	-2	-3	2
Computer Ontologies	-2	1	-2
Javascript	-3	1	0
Big Data	3	-1	1
SQL	1	-1	4
Spatial Modeling	4	2	-1
APIs	-1	1	3
Desktop GIS	1	1	-3
Collaborative Design	-1	4	0
Domain-Specific Knowledge	1	5	-1
C-Sharp	-4	-5	2
ArcGIS	0	3	-5
Server Architecture	-5	0	5

Table 3-10. Participant information. Names are randomly generated pseudonyms with no connection to cultural background or gender. Defining factor indicates which factor the sort contributed most towards defining. Factor loading indicates how much their sort contributed to the definition of the type in the factor analysis (maximum of one).



Respondent Name	Defining Type	Factor Loading
Petra	3	0.6168
Jackson	1	0.6672
Tina	2	0.6486
Hans	2	0.6966
Pavel	2	0.8034
Gabriel	2	0.8372
Kristen	2	0.6338
Hugo	2	0.5477
Delta	1	0.8141
Christian	2	0.8018
Oscar	1	0.4898
Hayden	1	0.7770
Lucas	3	0.5881
Edward	1	0.7616
Susan	1	0.6783
Timothy	1	0.5143
Arthas	3	0.6492
Donovan	3	0.7137
Clancy	n/a	n/a
Darien	n/a	n/a

Type One

I describe type one as the spatial analyst, or traditional GIS user. From this viewpoint, the spatial core is of greatest value, as Delta noted:

"Spatial modeling, spatial-temporal analysis, visualization, those things will be around for a long time. They will probably be the same thing in their core. (We discuss) changes in chip architecture from 32 to 64-bit and parallel and distributed computing aspects, but maybe we won't be talking about those terms in the future in the same way. It would be like talking about some other medium that is so ubiquitous to stress the importance of it."

For Hayden, 'spatiotemporal analysis' held the top spot in their sort, eliciting a positive reflection on the combination of those terms:

"If you notice, the top (term) is 'spatiotemporal'. To me that combination for so long was left out, it wasn't always temporal. A lot of research involved time studies and other research looked at spatial or using location based type things. It just seems like people are finally bringing it together, but it's been a long time coming I think."

When discussing terms in the middle values of their sort, like HTML, they recognized the utility without dismissing them entirely, saying:

"I'm kind of neutral about those. These feel more in the past or unsecure and in the middle these are not the essence of GIS, but you'll need them."

The idea of 'needing' terms ranked in the middle of the sorts was a common theme among respondents across types. Respondents regularly indicated that their toolbox of necessary skills or abilities varied based on their focus or the problem at hand, often shifting as their needs changed. While the most valuable terms in each sort give a strong indication of the primary focus of the respondents, how the middle terms vary provides valuable insights as well.

Jackson placed 'spatial analysis' as the most valuable term, followed by 'spatial modeling' and 'domain-specific knowledge'. They explained this preference, saying:

"GIScience and technology helps us advance the science and domain specific knowledge. And that has been the focus of my professional career... in terms of using imagery or digital imagery and maps, to understand something about the environment. So spatial analysis and spatial modeling has been pretty much what I do, and what I use the computer technology (for)."

Timothy explained the lower relative value of some terms not for lack of utility, but for lack of focus:

"For example, HTML is important if you are a web designer, but I don't think we are web designers. Also, I think HTML is easy to read. I've had to use HTML before and I always feel it is something that if you need to use it, you look it up and use it, but then forget about it. VGI and Web2.0: these are amazing fields, just not what I'm currently working on, but if I had the chance I would."

Edward built their rankings holistically, chaining their conception of the value of the terms in a clear process-based hierarchy:

"So from the holistic perspective this is how I think. I'm doing spatial modeling with spatial analysis and it needs to be visualized and I use spatial temporal analysis and databases, and I'm working with big data and computation statistics, processing and machine learning. By using the programming and certain data sources... it flowed out of my processing, how I do research. So I've taught programming in GIS and that side of it but I think of it more as a tool.

Teaching concerns also weigh into the ranking here, as Delta notes:

"From an educator's standpoint, you've got these two big pressures. One is that at a state university, especially in the context of constrained resources, tangible skills that will lead to job market potential for current students are a concrete goal, but most people who do technologically enabled things did not learn them in a formal setting. So are we giving tasks to people or are we building potential? Where are the timeless and persistent themes that we can address?"

Delta reflects the overall scheme of rankings in this type, where the 'timeless' concepts of spatial analysis and modeling prove most valuable within the CyberGIS construct. The details of how to achieve those goals are valid in support of the analysis, but are easily forgotten or shifted as new technologies, tools, or methods develop.

Type Two

I describe type two as the domain-specific problem solver. To this type of practitioner, the primary value of CyberGIS components is to solve problems in a specific domain through spatial analysis and with tools like ArcGIS. As Gabriel states at the beginning of their interview:

"I return to think over and over again GIS is in many ways a tool, and while there (is) science around it, and I've done that science, I still feel in order to do it you need to understand the stuff you're going to do it with. So domain specific knowledge is not domain as in knowing the GIS domain, but really some part of the geography. If you're a soil scientist or biologist or whatever, (that knowledge is) something tangible that you have core knowledge of. I think that is key to becoming a good GI scientist. Producing maps and visual information and also the term spatiotemporal analysis is in everything we do in GIS."

This core idea of having a problem to solve in some specific domain was a clear theme in Hugo's interview as well:

"If you think about GIS, whether it's CyberGIS or anything; they're a tool, and without having a problem to solve, our tools are meaningless. So (whatever) your domain knowledge is, there has to be some knowledge about what the tool needs to solve. It can't just be done in the abstract."

This fusion of important components, connecting domain knowledge to spatial analysis, was shared by Kristen, who detailed their ranking process through clarity of utility, saying:

"I use spatial analysis as one of the central approaches within my research. I also almost exclusively use desktop GIS, so that part was really clear... I am more of an applied GI Scientist, so domain specific knowledge is very central to my research. Those three things were most clear to me. Moving down the aisle so to speak, I see a lot of collaborative research. Collaborative design enables that and I use a variety of different technologies to facilitate collaboration both with GIS software and outside of it too."

The broad set of terms was appreciated by Pavel, who connected the terms present to a larger idea of geodesign:

"I place collaboration in a ramp with communication, cooperation, coordination, and collaboration so that there is a participatory ramp, and I've unpacked the nature of those things. And design is near and dear to my heart because this is the innovation of ideas…. So adding the collaborative aspect to that design is a huge thing, submerging thing these days in terms of geodesign. Geodesign is very large I think going to be an extremely important field. And doing all of this in regard to (geodesign) is going to be important. There isn't anything here that is not significant in some way to that because that's the nature of computation."

This framework of CyberGIS components existing to facilitate problem solving is a complementary approach to considering the domain-specific problem solver. In this case, the domain is very broad and connected computationally. These components could be interpreted as an extension of the current state of GIS, as noted by Tina:

"To me (the term) 'CyberGIS', I have a hard time seeing it as not part of GIS. It is the natural evolution of GIS. I don't believe that any advance within the field needs to declare itself a new field as opposed to a natural evolution in the field. CyberGIS is that (declaration)."

They continued to note that even the suite of problem-solving tools and components within the set of terms was difficult to settle on, saying:

"When students ask which one language they should learn, I tell them I don't know. But if you learn two languages, one of them should be Java, and if you learn three, two of them should be Java and Python. If you learn four, three of them should be Java, Python and R."

For the domain-specific problem solver, two dual identities may exist. Those who use CyberGIS components to solve problems from other domains through the tools and affordances of CyberGIS, but without incredible distinction between the components, like Gabriel:

"So WebGIS and Web2.0: they aren't the same, but today they feel the same. I'm not sure how important it is for me to separate, Java, JavaScript, C#, HTML... I see those as a collection that supplement each other. The platform we are developing is using C#, HTML, JavaScript and MySQL to accomplish what it does. So they go together so it's hard to say one is more important."

And those whose problems are facilitated by these advanced components, like for Hans:

"I have this split identity in a way between behavioral geography and GIS; I'm very much interested in technology mediated individual decision-making. So in a way it's more a nexus of elements that are important."

The domain-focused problem solvers value anything that can help their pursuit for knowledge, and CyberGIS broadly provides elements of varying value in that pursuit.

Type Three

I describe type three as the CyberGIS-enabled computer scientist. Only four respondents fit this characterization, and their preference of the computer science and programming topics showed a clearly unique set of viewpoints in the sample. Petra viewed server architecture as the most valuable component, saying:

"Without (understanding) how the servers are built, whether that is a stack of servers or a series of servers to create a more efficient computational aspect, then you are dead in the water. You can use parallel computing, but once you're getting to petabytes of data, then your server architecture really comes into play and that's very valuable."

And even though servers were not currently key components of their research plan, Petra was aware of the need for a long-term consideration:

"I am thinking forward because I use a lot of data and I also want to incorporate imagery for ten years of data. At the point where I start incorporating the imagery and I need to process all that imagery. My little 2 terabyte gaming computer might not work."

In this case, the processing and analysis Petra wants to undertake is only possible with a computational component of CyberGIS: server architecture. Lucas agreed with this enabling concept of CyberGIS, connected to geocomputation:

"Well, geocomputation to me is where the computer drives the analysis. So the computer is not just a convenient way of doing it. For example, doing a regression analysis, you could theoretically at least solve a regression analysis with pencil and paper, but the computer makes it convenient. Geocomputation is where the computer enables new types of analysis and that's why I think geocomputation and CyberGIS are coupled. They are almost the same thing."

Lucas takes geocomputation as a synonym term for CyberGIS as well:

"My view of CyberGIS is geocomputation enabled by distributed computing. And I do spatial temporal analysis so the whole point, to me, of CyberGIS, would be to enable spatial-temporal analysis."

CyberGIS enables analysis that would be impossible otherwise. As Lucas noted, regressions can be solved on paper, but computers facilitate such analysis in a more efficient and revealing way. Donovan saw outputs and products from these components as the primary goal of CyberGIS components:

"Server architecture, databases, big data, SQL, everything that it takes to compute the make queries, optimize for speed, that's our main goal: the data output."

Arthas interpreted this facilitation through the programming side of the components, however:

"API is number one just because it's less for me to have to do, might as well look for stuff that other people have done. Why reinvent the wheel if there's something that is capable that you can use? If I ever need to look for something I always look to see if someone's done it first. C# and SQL, you pretty much need that for anything. That's pretty much your entire database, your entire back end. Without C# or SQL you wouldn't have much of anything."

Overall, these CyberGIS-enabled computer scientists see the infrastructure and computational components as the true drivers of production and research within GIS. Traditional GIS is valuable, but not as valuable as the tools to facilitate faster, and more capable, GIS production. The components of CyberGIS that allow this progression into 'better' analysis have precedence in this view.

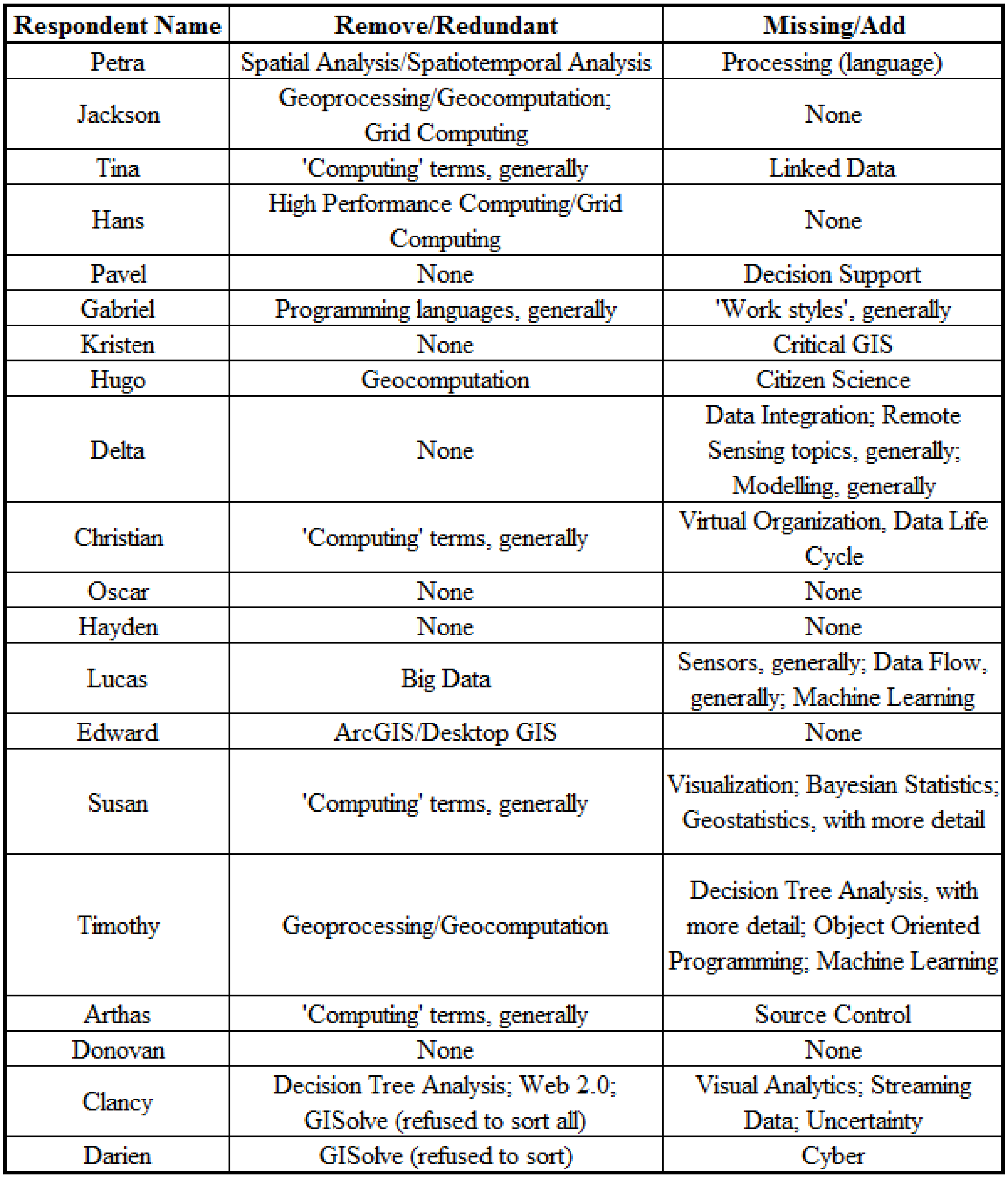
Term Specific Comments

Respondents recommended topics to be removed or added to future iterations of the q-set in Table 3-11. Terms that were viewed as too redundant or with too much overlap, as well as missing terms of high importance to the respondents are presented here. These revisions reveal some redundancies in the set of terms developed by the machine reading approach, but also outline emerging areas of importance in CyberGIS. To rectify the issues raised in the interviews, I provide a revised set of terms in Appendix Two. These expert based revisions provide further clarity to the terms that make up key components in CyberGIS.

Conclusion

In this article, I describe the construction of key terms in CyberGIS, and how expert GIS users value those terms in their experiences. Initially, I used machine reading to extract and identify key terms and topics from CyberGIS literature. Using q-methodology, I constructed statistically distinct viewpoints based on expert sorts of these key terms. I identified three distinct types representative of expert views of CyberGIS components; the spatial analyst, the domain-specific problem solver, and the CyberGIS enabled computer scientist. The components of CyberGIS I identified are valued unevenly across viewpoints, with major disagreements on some GIS and GISci fundamentals, like ArcGIS, spatial modelling, and the role of computer science and programming. The variant viewpoints concerning CyberGIS provide meaningful insight into how experts with different domain focuses view and understand CyberGIS.

Table 3-11. Terms recommended for removal or addition based on respondent feedback. Terms with a '/' are terms that were indicated by respondents to have significant redundancy or overlap. Individual terms are separated by a semicolon. In most cases, recommendations for terms to add were for general topics, except for Timothy's suggestion of detailed terminology concerning decision tree analysis.



Respondent Name	Remove/Redundant	Missing/Add
Petra	Spatial Analysis/Spatiotemporal Analysis	Processing (language)
Jackson	Geoprocessing/Geocomputation; Grid Computing	None
Tina	'Computing' terms, generally	Linked Data
Hans	High Performance Computing/Grid Computing	None
Pavel	None	Decision Support
Gabriel	Programming languages, generally	'Work styles', generally
Kristen	None	Critical GIS
Hugo	Geocomputation	Citizen Science
Delta	None	Data Integration; Remote Sensing topics, generally; Modelling, generally
Christian	'Computing' terms, generally	Virtual Organization, Data Life Cycle
Oscar	None	None
Hayden	None	None
Lucas	Big Data	Sensors, generally; Data Flow, generally; Machine Learning
Edward	ArcGIS/Desktop GIS	None
Susan	'Computing' terms, generally	Visualization; Bayesian Statistics; Geostatistics, with more detail
Timothy	Geoprocessing/Geocomputation	Decision Tree Analysis, with more detail; Object Oriented Programming; Machine Learning
Arthas	'Computing' terms, generally	Source Control
Donovan	None	None
Clancy	Decision Tree Analysis; Web 2.0; GISolve (refused to sort all)	Visual Analytics; Streaming Data; Uncertainty
Darien	GISolve (refused to sort)	Cyber

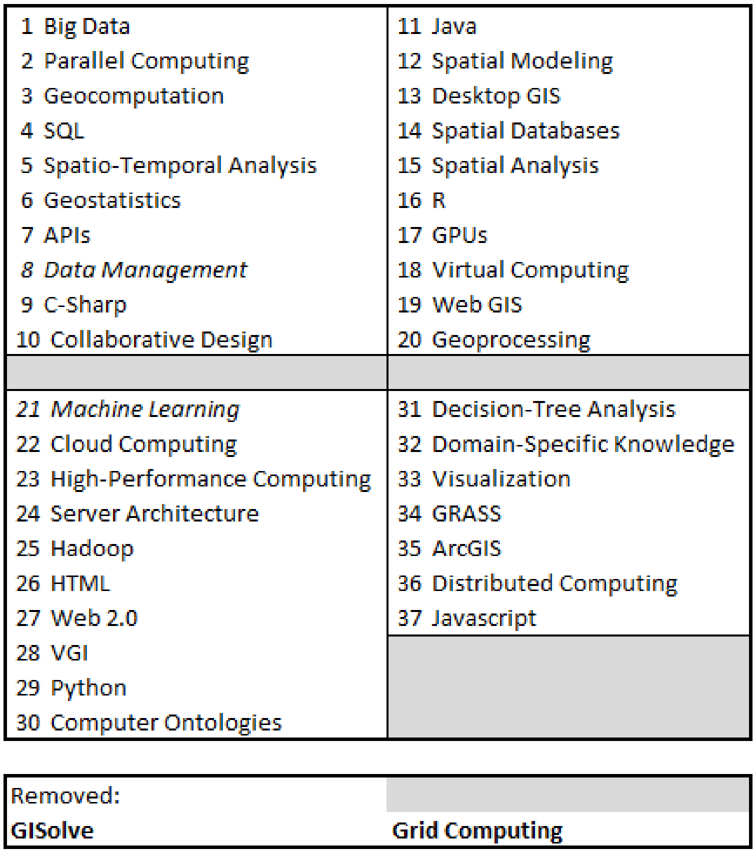
Review of Research Questions

I proposed three major research questions which informed this investigative approach. Through my review of published CyberGIS research and the q-method research, I begin to asses not only expert viewpoints of CyberGIS components, but also how CyberGIS fits together the diverse domain aspects of computer science, programming, geography, and GIS.

What components of CyberGIS are most prominent in the published literature?

I constructed a set of 37 terms, using a machine reading approach, which emerged as the dominant key topics in CyberGIS. These topics focus mostly on computing terms and programming languages, but also incorporate core spatial actions and technologies. As shown in Table 3-12, after interviewing participants and revising the set of terminology, I emerge from this work with 37 revised terms that represent the state-of-the-art in CyberGIS. The fusion of computer science, programming languages, and fundamental geographic and spatial actions is striking. If CyberGIS is the future conception of GIS, or if it only represents the growing importance of computer science and programming in geography, the landscape of what it means to be a GIS expert or GIS user is facing a major shift in content.

Table 3-12. Revised set of q-sort terms based on respondent feedback. New terms noted in *italics*, removed terms listed at the bottom in bold. Numbers correspond to the random coding of the term for analysis, and should not be read as an implied hierarchy. I recommend that these items form the set of terms in future q-method surveys.



1 Big Data	11 Java
2 Parallel Computing	12 Spatial Modeling
3 Geocomputation	13 Desktop GIS
4 SQL	14 Spatial Databases
5 Spatio-Temporal Analysis	15 Spatial Analysis
6 Geostatistics	16 R
7 APIs	17 GPUs
8 Data Management	18 Virtual Computing
9 C-Sharp	19 Web GIS
10 Collaborative Design	20 Geoprocessing
21 Machine Learning	31 Decision-Tree Analysis
22 Cloud Computing	32 Domain-Specific Knowledge
23 High-Performance Computing	33 Visualization
24 Server Architecture	34 GRASS
25 Hadoop	35 ArcGIS
26 HTML	36 Distributed Computing
27 Web 2.0	37 Javascript
28 VGI	
29 Python	
30 Computer Ontologies	

Removed:	
GISolve	**Grid Computing**

How do experts value these components based on their experience?

While the CyberGIS framework is facilitated by advanced computer infrastructure, two of the viewpoints I found (the spatial analyst and the domain-specific problem solver) consider the fundamental spatial methods or domain-relevant techniques as having greater value than the underlying technologies. These components are not at the forefront of their views of GIS, instead existing as components which serve to functionalize their analysis. In this view, the technical details are not valuable to consider, though certain programming languages or computing approaches may have some value in achieving their overall goals.

In contrast, the CyberGIS computer scientist values the computational advantages which facilitate advanced analysis. The use, development, and progression of GISci within these components is a major motivation of this conception of value. Computer science and programing concepts enable advanced forms of GIS, and this viewpoint values this enabling, rather than the end product.

In essence, the overall value of GIS, GISci, and spatial problem solving is viewed through these types in with distinct component values. One type values the underlying technologies (type three), which allow a group of GIScientists (type one) to produce advanced GIS/Sci methods and processes, which are then utilized by a group to solve domain-based problems (type two).

How do experts conceptualize CyberGIS?

While CyberGIS exists as a framework to capitalize on advanced infrastructure while advancing GIS/Sci simultaneously, there appears to be a continuing specialization focus in play within this data. Rather than committing to an exploration of both components, participants here remain focused on components that help facilitate their current interests, research programs, and teaching responsibilities. What was valued at the time of interview was components that helped achieve their goals, and the holistic fusion of computer science, programming, and GIS was irrelevant. What remains unclear is how this specialization benefits or detracts from utilizing the advantages that CyberGIS affords.

Future Work

Future analysis should continue to investigate the viewpoints of other stakeholders concerning the components of CyberGIS. I recommend replication of this study with additional, diverse participant sets to further identify factors and viewpoints concerning CyberGIS terminology. Further, I suggest future studies use the adapted q-set shown in Appendix II. I have supplied my list of questions for participants for use or adaptation in Appendix III. How CyberGIS components are integrated into industry and their broad benefits to students as they engage with the professional workforce, is also of interest.

Components of CyberGIS do facilitate spatial analysis and problem solving that would otherwise be impossible. However, this research indicates that significant differences exist in how experts view and utilize these components. GIS is a fundamental information technology, but the landscape of what makes a GIS expert, and what skills or abilities are necessary for competence in GIS, is changing rapidly. While this article presents an important first step in understanding how this domain fusion functions in variant viewpoints and uses of these components, it also highlights a growing lack of understanding in how GIS is stretching to apply in other domains and contexts. More research is needed to begin to understand the frontiers of this domain.

CHAPTER IV

COURSE SYLLABI IN GIS PROGRAMMING: TRENDS AND PATTERNS IN THE INTEGRATION OF COMPUTER SCIENCE AND PROGRAMMING

The growing subfield of Cyber Geographic Information Systems (CyberGIS) is the frontier of GIS coursework, where spatial analysis and advanced computer modeling merge (Wang 2010). The skillset for this frontier is constantly being reimagined, and there is little empirical understanding of the content or function of GIS programming courses, and how they integrate computer science. Course offerings in Geographic Information Systems and Science (GIS and GISci) provide a broad base of knowledge and instruction in diverse components of GIS, including fundamental theories, advanced methods of integrating GIS with specific knowledge domains, and synthesis of computer science and programming concepts, technical architectures, and data acquisition methods (Lukinbeal and Monk 2015). The number of such courses is expanding rapidly as GIS expands in higher education. While recent reviews of GIS syllabi reveal some similarities in the structure of introductory GIS courses in the United States (Wikle and Fagin 2014), there remains a lack of understanding of how GIS courses at all levels integrate basic components of computer science and programming. Skills in these specializations, like developing software, automation, coding, and scripting, are quickly becoming fundamental skills of the GIS analyst (Bearman et al. 2015). Further, the use and integration of computer science and programming into GIS is transforming the utility and ubiquity of GIS and GISci (Harvey 2013). This transformation is driven by diversified applications of GIS, enabled by cloud and server computing, easier access to GIS, enabled by greater availability of technology, and greater customization of GIS capabilities, enabled through programming extensions (Harvey 2013, Wright 2012, Sui 2014).

The purpose of this paper is to analyze course syllabi concerning GIS programming and computer science integrations explore the nature of these courses, specifically how they integrate the topics of computer science and programming and their instructional configuration. I seek to broadly understand how existing courses balance core GIS topics like spatial analysis with key computer science and programming skills, and the learning outcomes that dominate these courses. Further, derived from my analysis of GIS syllabi, I seek to characterize the evolving curriculum in computer science and programming within GIS courses. I seek to outline common instructional materials or teaching methods relating GIS with computer science and programming, using syllabi to gather clues about the evolving curriculum in GIS. I consider the following research questions in my investigation:

First, what is the nature of computer science and programming topics in GIS programming courses? A considerable library of programming languages, computing tasks, computer science concepts, and so forth are relevant to GIS, but understanding which components are most common and thus considered most relevant is an important step in describing overall patterns in course construction.

My second question asks: how does the nature of these courses support the development of key GIS competencies given growing use of computer science and programming in GIS? What students are expected to do with their new GIS programming and computer science knowledge is of considerable interest. Advanced computing facilitates the expansion of GIS into new areas. Determining what GIS users are expected to know, and what tasks they are expected to perform, helps scaffold and define an understanding of how these computer science and programming topics function.

My final question considers how topics are taught within these courses. What is the sequence of learning in these courses, and is there any curricular coherence in how these courses are presented? Building an understanding of the content, teaching methods, and applications of computer science and programming in GIS reveals how these courses function, and what components might serve as common components across courses. Overall, these questions serve to further an evidence-based understanding of how computer science and programming function in a GIS context, providing more insight for course development and designs fusing GIS, geography, computer science, and programming.

To achieve these goals, I first provide a brief review of research on GIS curriculum and instruction, focusing on the integration of computer science and programming, and its relation to the ongoing development and importance of CyberGIS. Next, I present research analyzing GIS syllabi. To conclude, I discuss the findings as they relate to the current landscape of GIS education.

Components of GIS, Computer Science, and Programming

GIS education has long attempted to balance the competing needs of building theoretical competencies in the nature of spatial analysis with the ability to use the software and programs associated with such analysis (Wright et al. 1997, Pickles 1997). There is a tension between the idea that use of GIS serves as a tool or as a component of scientific investigation (Walsh 1992). Competing emphases in GIS education results in courses a large number of divergent goals. GIS courses therefore are structured to accommodate the individual preferences of GIS instructors, and to build a range of professional and academic skills and capabilities in GIS students. Coupling GIS theory with real-world applications in GIS curriculum remains a subject of considerable debate (Kemp and Frank 1996, Bearman et al. 2015), and contributes to the overall complexity of learning outcomes within GIS instruction (Whyatt et al. 2011).

Due to the diverse applications and situations of GIS education, instruction exists in diverse domains and occurs in a wide variety of contexts. Ideally, a GIS course instills skills relevant not only to GIS-specific applications, but transferable to other areas of spatial thinking and domain-based problem solving (Şeremet and Chalkley 2014). GIS courses may focus on numerous topics, functioning as introductions to learning specific software (like ArcGIS), descriptions of fundamental spatial topics and theories, or connections to advanced methods, knowledge domains, or technologies. Further diversifying the nature of GIS education, GIS is also not 'locked' to geography, allowing a wide range of disciplinary approaches to GIS applications and instruction (Ellul 2015).

Curriculum efforts attempt to organize this diversity of GIS instruction, but many challenges remain. Quantifying the learning mechanisms and separate components of GIS as a 'tool' or as a 'science' has been a difficult prospect (Wright et al. 1997, Pickles 1997). As

GIS spreads from geography to other disciplines in both formal and informal learning environments, issues arise in teaching technical knowledge (Tate and Unwin 2009). A GIS student may not have complementary computer and spatial skills, depending on the home domain of their GIS instruction (Sarkar and Pick 2014). Additionally, as Blaschke and Merschdorf (2014) discuss, there is no wide ranging or extensive agreement about the precise boundaries of GIS and GISci, which makes defining an appropriate learning framework for GIS courses a challenging endeavour. For example, GIS instruction can exist at a basic 'button-pushing' level without developing analytical skills (Bishop 2009), and courses that follow rigid, guided formats achieve less than those that require complex spatial analysis and problem solving using GIS (Theo 2011) or deeply explore the functions of GIS (Miller 2000). Teaching 'with' GIS and 'about' GIS remains difficult to separate (Sui 1995), and adding computer science and programming components greatly enhances a course's complexity, as illustrated below in Figure 4-1.

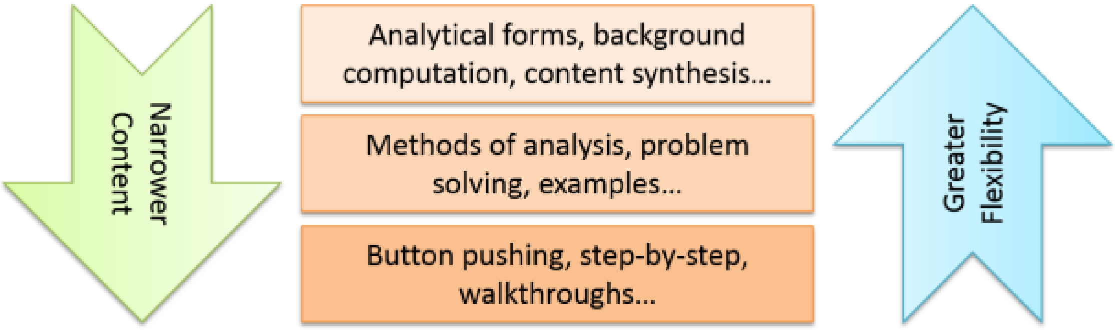

Narrower Content
Analytical forms, background computation, content synthesis...
Methods of analysis, problem solving, examples...
Button pushing, step-by-step, walkthroughs...
Greater Flexibility

Figure 4-1. Possible topics in GIS course variants. Courses that teach 'button-pushing' will have a narrower focus on producing specific outputs. Courses that consider the analysis behind the buttons may promote different problem approaches. Further, courses that describe how the analytics function will have a deeper level of content to assimilate (Kostelnick et al. 2009, Kerski et al. 2013).

One possible unifying aspect of GIS is the affordances it provides in developing spatial thinking. While GIS is a support system for spatial thinking (National Research Council 2006), the development of spatial abilities is not the primary focus of most GIS coursework. Though GIS fundamentally concerns the identification of spatial relationships, and research relating GIS and spatial thinking is producing a growing body of literature (Kim and Bednarz 2013, Madsen and Rump 2012, Lee and Bednarz 2009), there is not a clear understanding of the optimal means for developing spatial skill and GIS tool skill simultaneously. Working with GIS in classroom settings can develop core spatial skills, as the GIS 'tool' is a non-neutral method of interacting with spatial data (Goodchild 2011). Integrating GIS into instruction to capitalize on its affordances remains difficult (Sinton 2009), as instructors may not be disposed to teaching spatial skills through geography coursework (Jo and Bednarz 2014). The spatial aspects of learning with GIS rely on the transparency of spatial functions within GIS to function. Challenges include lack of fundamental conceptual awareness (Kerski et al. 2013) and reliance on rote instruction rather than problem solving approaches (Read 2010). Additional concerns exist in connecting spatial thought, computer infrastructure, and the GIS interface (Goodchild 2011). GIS and geography are no longer disciplinarily locked together, and GIS without geography is growingly commonplace. Whether it is as effective requires further study.

Two major structures exist which aim to structure and identify the essential skills in GIS. The Geographic Information Science and Technology (GIS&T) Body of Knowledge (BoK) serves as a reference for prominent GIS components, outlining key methods and topics in academic and professional use of GIS. Its organization based on knowledge areas serves as point of focus for curriculum planners (DiBiase et al. 2006). A subsequent, more industry centered resource is the Geospatial Technology Competency Model (GTCM). The GTCM outlines tiers of competencies relevant in many categories, from specific industries to personal and academic effectiveness. Both of these structures provide form and function for curriculum and course design in GIS, and also reveal the important function of programming and computer science within GIS. Coupled with recent investigations in workforce needs in the GIS domain (Hong 2015, Wikle and Fagin 2015), the growing importance of programming and computer science in GIS is apparent through these consolidations.

Programming and computer science skills are under-developed within academic GIS coursework (Şeremet and Chalkley 2014). However, growing industry evidence suggests that the computer science and programming components of professional GIS positions are increasing in importance (Hong 2015). Further complicating the fusion of GIS, computer science, and programming is that comprehension and knowledge may be more complex in terms of domain information in computer science, requiring more instruction and more direct interaction with these topics within a GIS course (Gasparinatou and Grigoriadou 2011). A growing field of GIS research known as CyberGIS is further investigating the importance of synthesizing and taking advantage of the merger of computer science, GIS, and advanced computing infrastructure (Wang 2010). With this movement in GIS research and application in mind, uneven programming and computer science knowledge of GIS learners may limit their progression towards expertise in GIS, especially as computer skills become more essential to the GIS practitioner. While trajectories to expertise in GIS are sparsely researched (Downs 2014), novice programmers require about ten years, on average, to develop skills and learn content sufficient to become an expert programmer (Robins et al. 2003). Experts in using GIS may be novice programmers, a concerning duality as programming and computer science become more essential to using GIS. Though programming is an essential skill in state-of-the-art GIS and GISci, how it is taught and learned in GIS degree programs is not clear.

Analyzing course structure builds evidence of how instructors emphasize and value course topics. In geography, review of course content has led to discussion on incorporating more technical instruction relevant to GIS (Marti et al. 2014). Given the rise in the utility of GIS in the classroom, its use as an instructional tool is growing, though what classroom conditions support its use requires significant attention (Favier and van der Shee 2012).

Texts used in courses are meaningful to course design as well, as noted by Lee and Catling (2015). Generally, the use of GIS in courses is influenced by how instructors perceive GIS to be useful (Lay et al. 2013). Though GIS courses demonstrate a wide variance in instruction, in computer science, there is evidence of a broad uniform nature in how most departments handle initial courses in the computer science major sequence (Davies et al. 2011). Guo (2014) reported a transition to Python as the first language learned by new students in introductory computer science courses. While the topical focus of computer science and programming knowledge changes rapidly, some consistency exists entering that domain. How GIS programming courses balance the variety of approaches in GIS with more standard computer science and programming knowledge is an important component of this domain fusion worth considering.

Measurement of the way these courses emphasize topics or expect learners to understand them is achievable by analyzing learning outcomes or learning objectives. Statements of learning expectation describe the essential course outputs instructors expect learners to take with them at the close of a course (Harden 2002). They have grown in prominence across education levels as a tool to measure and assess student learning and teaching practices (Lacireno-Paquet et al. 2014). Effective outcomes and objectives are measurable in regular intervals, encourage reflection and self-efficacy, and can be adjusted rapidly in response to new teaching methods or technological innovations (Merchant et al. 2014, Vanblaere and Devos 2015, Savery 2006). As an added benefit, outcomes target different types of knowledge retention and can function at different hierarchical levels (Van der Kleij et al. 2015). Bloom's Revised Taxonomy is a common knowledge hierarchy used widely across education to arrange learning expectations (Krathwohl 2002). This taxonomy is widely used as an accessible hierarchy for cognitive differentiation (Adams 2015). This widely adopted ranking system allows for some standardization in deconstructing learning outcomes and objectives, while maintaining a standardized means of discussing similarities and differences in how courses function.

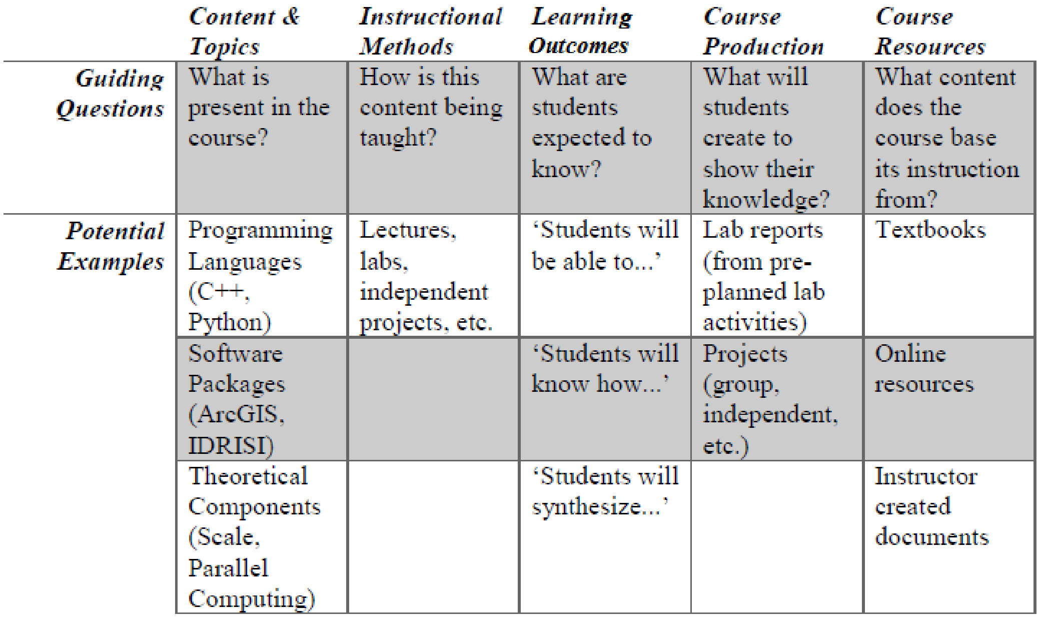
Educators in GIS have a long history of grappling with the complexity of topics taught in the GIS classroom. Efforts to collect and consolidate this range of approaches have yielded useful guides, like the GIS&T BoK and the GTCM. But the renewal of computer science and programming as key topics in GIS, and the growing importance of these skills professionally, requires an additional focus on how computer science and programming are structured in GIS programming courses. The remainder of this chapter addresses this need.

Methodology

Syllabus review provides a valuable lens into the knowledge, skills, and practices valued in courses. To a certain extent, syllabi can also reveal how students are expected to acquire these capabilities. To understand the nature of GIS, computer science and programming courses, this study collected and analyzed syllabi that were being used or were planned for use before or during the fall semester of 2015. While syllabi are not perfect representations of how courses function, they do represent course content in a manner useful for comparison (Jo et al. 2012). Syllabi serve as an initial communication tool for students and formal representation of course content, and viewing them as important documents relating the attributes and characteristics of courses enables an understanding of topics, resources, and methods planned for use in these courses (Eberly et al. 2001). I recruited participants through direct contact at major conferences (2015 Association of American Geographers, 2015 University Consortium of GIS, 2015 ESRI User Conference), through posts to professional and academic geography and GIS listservs, and through social media. These outlets are foci for active scholars and educators who are most deeply engaged in discussions concerning the integration of GIS, computer science, and programming. Existing public databases such as that developed by the CyberGIS Fellows Initiative (CyberGIS Center 2014) and the GeoTech Syllabi Repository were additional sources of syllabi. In total, I obtained forty syllabi for use in this analysis. These syllabi represent a wide variety of course topics and structures and form a diverse set of artifacts. I also invited syllabus contributors to complete a web-based interview and questionnaire, to clarify syllabus content and to elicit additional materials or insight into the courses. In all cases, participants enrolled on a volunteer basis only, and did not receive compensation.

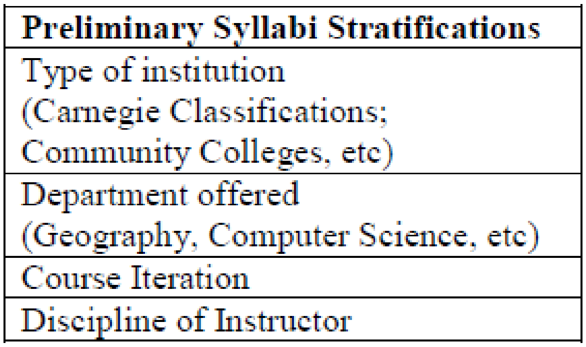
The syllabi and interviews were organized, coded, and classified into a cohesive database for comparison and analysis. Using syllabi provides unique opportunities to work with primary source data on course content and delivery approaches, but also presents challenges in representing course structures (Boss and Drabinski 2013). Syllabus requirements vary with universities and departments, and components of a syllabus (i.e., learning outcomes or content schedules) are inconsistently constructed. Other factors, such as how many sections of the course exist, instructor experience, etc. will also influence syllabus content. I built an initial structure for syllabus deconstruction as shown in Table 4-1 below. A further stratification framework allowed additional distinction by institutional types, academic departments, and other factors as listed in Table 4-2. To compare and contrast syllabi collected in this study, I used these initial stratifications as guides for comparisons between course originations. Carnegie classifications allow for a standardized metric for organizing institutions by size, research output, and other factors, to determine what types of institutions are represented in this sample (Carnegie Foundation 2011). I use department of origin and instructor discipline to outline the backgrounds these courses originate from, while noting course iteration for insight into whether repeated, and potentially revised courses show any significant differences to new course designs. I represent my overall workflow graphically in Figure 4-2.

Table 4-1. Areas of initial syllabus content analysis. Determining and describing the characteristics of computer science and programming in GIS courses is an important first step to discovering their integration into the current state-of-the-art of instruction. Certain topics, like programming languages, may be easier to discover in course syllabi as they require specific notation as a course topic. However, components like 'collaboration' may be more difficult to determine, as aspects of collaboration may not be specifically outlined in a given syllabus.



	Content & Topics	*Instructional Methods*	*Learning Outcomes*	*Course Production*	*Course Resources*
Guiding Questions	What is present in the course?	How is this content being taught?	What are students expected to know?	What will students create to show their knowledge?	What content does the course base its instruction from?
Potential Examples	Programming Languages (C++, Python)	Lectures, labs, independent projects, etc.	'Students will be able to…'	Lab reports (from pre-planned lab activities)	Textbooks
	Software Packages (ArcGIS, IDRISI)		'Students will know how…'	Projects (group, independent, etc.)	Online resources
	Theoretical Components (Scale, Parallel Computing)		'Students will synthesize…'		Instructor created documents

Table 4-2. Stratification classifications for syllabi collected in this study. These stratifications allow comparisons between similar types of syllabi for review.



Preliminary Syllabi Stratifications
Type of institution (Carnegie Classifications; Community Colleges, etc)
Department offered (Geography, Computer Science, etc)
Course Iteration
Discipline of Instructor

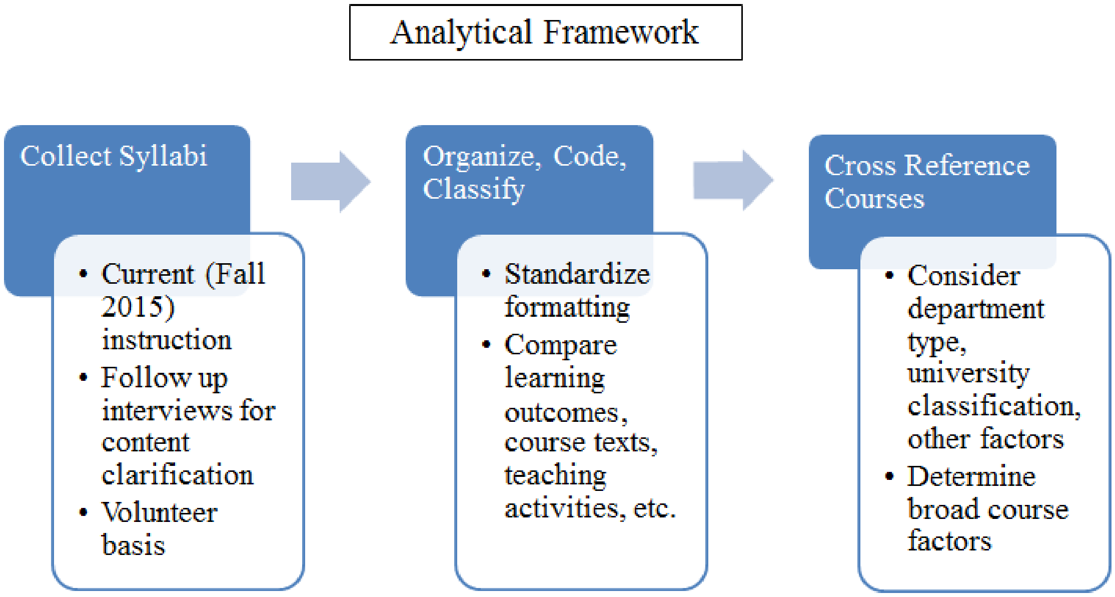

Analytical Framework
Collect Syllabi
Current (Fall 2015) instruction
Follow up interviews for content clarification
Volunteer basis
Organize, Code, Classify
Standardize formatting
Compare learning outcomes, course texts, teaching activities, etc.
Cross Reference Courses
Consider department type, university classification, other factors
Determine broad course factors

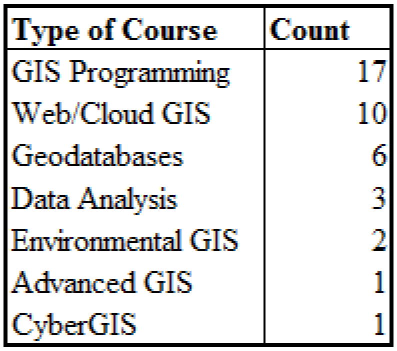
Figure 4-2. Analytical framework. This analytical process seeks to understand topics taught in GIS programming courses by extracting key components from course syllabi.

Results

Syllabi in Sample

Courses in this sample focus on a range of topics in GIS programming and computer science, as shown in Table 4-3. Nearly half of the courses (17) are stand-alone GIS programming courses, with varying titles like 'Programming in Python for GIS', 'GIS Programming', and 'Computer Programming for GIS'. WebGIS courses account for a quarter of the sample (10), while courses concerning databases (6), environmental GIS (2), and other topics round out this data. The syllabi from environmental GIS and advanced GIS courses show the greatest domain-based applications of GIS, using programming techniques to facilitate problem solving. The single course in CyberGIS represents the greatest fusion of multiple computer science and programming components. These four courses are unique outliers in that they stretch the conception of a GIS programming course in highly specialized ways. They are included in this analysis to represent the diverse ways computer science and programming is being taught in GIS. Throughout my discussion of these courses, I will generalize course titles and identifying details to preserve the anonymity of submissions.

Table 4-3. Generalized course titles of syllabi received in this sample. Most courses are either GIS programming or Web/Cloud GIS courses. Some very specialized courses, like Environmental GIS and CyberGIS also appear.

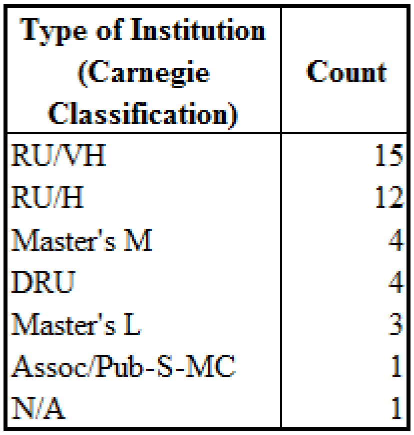


Type of Course	Count
GIS Programming	17
Web/Cloud GIS	10
Geodatabases	6
Data Analysis	3
Environmental GIS	2
Advanced GIS	1
CyberGIS	1

The courses come from a variety of institutions (Table 4-4), and are largely targeted towards advanced undergraduate or graduate students. Notably, courses in this sample generally come from so called R1 (research intensive) universities. The small number of programming courses found in Master's degree programs or community colleges may be the result of faulty data collection, or indicate that smaller, less research focused institutions face additional hurdles to develop specialized courses of this nature.

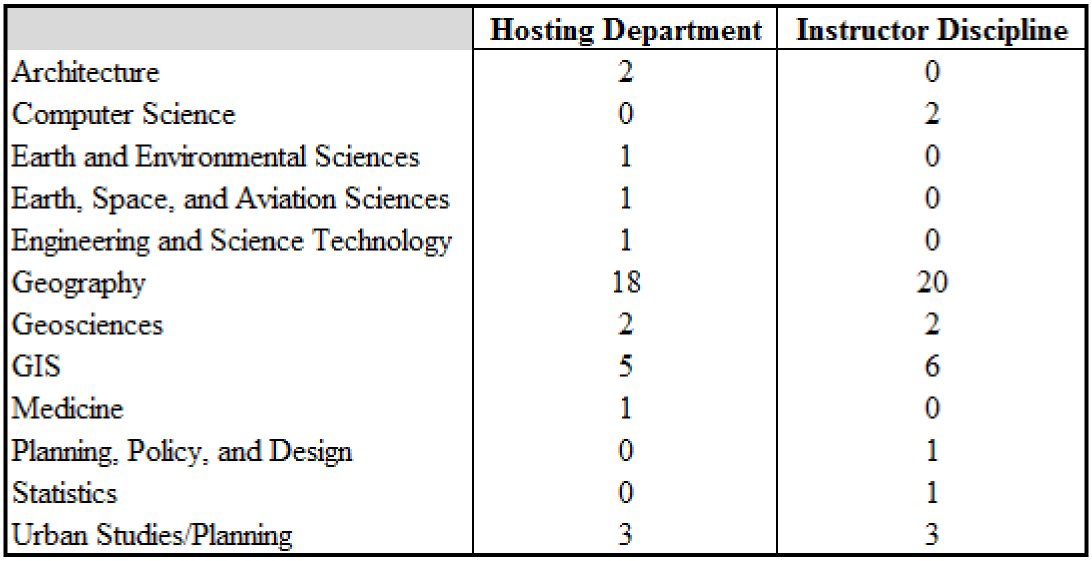
A range of departments offering these courses, but the majority of instructors who teach these courses report a geography or GIS background (Table 4-5). Nine different types of department, mostly in geography (18 courses) and GIS (five courses), host these courses. However, departments of architecture, medicine, and urban studies/planning also host courses in GIS programming. The instructor of record listed for the courses in this sample hold their terminal degrees in seven different fields, and while are again mostly geographers (20 instructors) or have GIS degrees (6 instructors), other backgrounds include statistics and computer science. Programming applications for GIS are not exclusively the concerns of geographers.

Table 4-4. Institutions of origin for syllabi in the sample. Most of the syllabi in this sample come from research universities (RU). Classifications based on current Carnegie Classification framework (Carnegie Foundation 2011).



Type of Institution (Carnegie Classification)	Count
RU/VH	15
RU/H	12
Master's M	4
DRU	4
Master's L	3
Assoc/Pub-S-MC	1
N/A	1

Table 4-5. Hosting department and instructor disciplines for courses in this sample. Syllabi with unlisted departments or affiliations not listed. Most GIS programming courses are housed in and taught by geographers and GIS practitioners, while Urban Studies and Planning also serve as 'homes' for these courses.

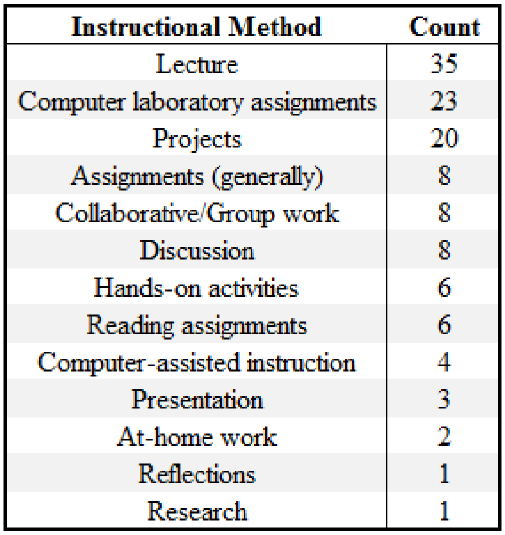


	Hosting Department	Instructor Discipline
Architecture	2	0
Computer Science	0	2
Earth and Environmental Sciences	1	0
Earth, Space, and Aviation Sciences	1	0
Engineering and Science Technology	1	0
Geography	18	20
Geosciences	2	2
GIS	5	6
Medicine	1	0
Planning, Policy, and Design	0	1
Statistics	0	1
Urban Studies/Planning	3	3

Nature of Course Structures

As indicated from the analysis of the syllabi, courses in this sample followed different formats, used different instructional materials, and had different strategies to attain education goals. As shown in Table 4-6, most courses rely on lectures and computer laboratory assignments to drive student learning. Nearly half required a summative project. These activities target different competencies and knowledge delivery for learners in the courses. Lectures provide foundational knowledge for students, while lab assignments offer students the opportunity to acquire the skills and practices of GIS. Independent, summative projects build a deeper, integrative understanding of the knowledge, skills and practices at hand (Lee and Bednarz 2009). A consolidated example of a typical GIS programming course in this sample uses lectures to introduce and reinforce foundational information relevant to the emphasis of the course (scripting, databases, etc.). Students are then expected to apply this knowledge through lab assignments, occasionally participating in discussions or collaborative work to further their comprehension. A major course project then attempts to consolidate all of this knowledge delivery into one broader creation activity, where students complete some task or develop some process relevant to the course topic.

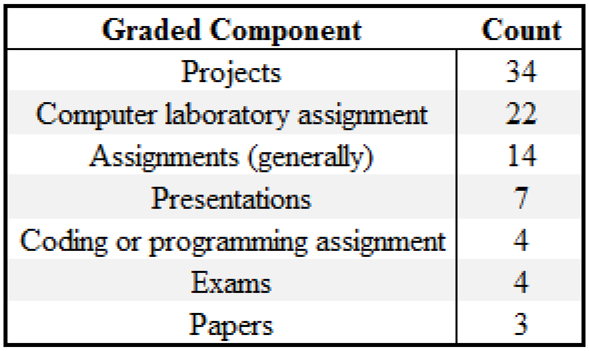
Table 4-6. Instructional methods used in GIS programming courses. Most courses highlight some form of lecture or lab assignments as part of the learning tasks in the course, while nearly half require students to accomplish some holistic task in a course project. Less prominent activities like reading assignments, discussion, and collaborative work, are present in around a quarter of the courses. Note that hands-on activities are differentiated from computer laboratory assignments as they were assigned in explicitly separated fashions. The count for projects includes single instances of real world problem solving activities and case studies.



Instructional Method	Count
Lecture	35
Computer laboratory assignments	23
Projects	20
Assignments (generally)	8
Collaborative/Group work	8
Discussion	8
Hands-on activities	6
Reading assignments	6
Computer-assisted instruction	4
Presentation	3
At-home work	2
Reflections	1
Research	1

Students were assessed primarily by project and laboratory grades (Table 4-7); exams and papers were used less commonly to evaluate students. The courses analyzed used different textbooks and instructional materials. The syllabi collected which included explicit details on resources for discussion indicated that readings selected by the instructor would be included as components of the course, and eleven syllabi required only instructor selected readings, and no textbook. The remaining 29 courses required or recommended 36 different textbooks overall.

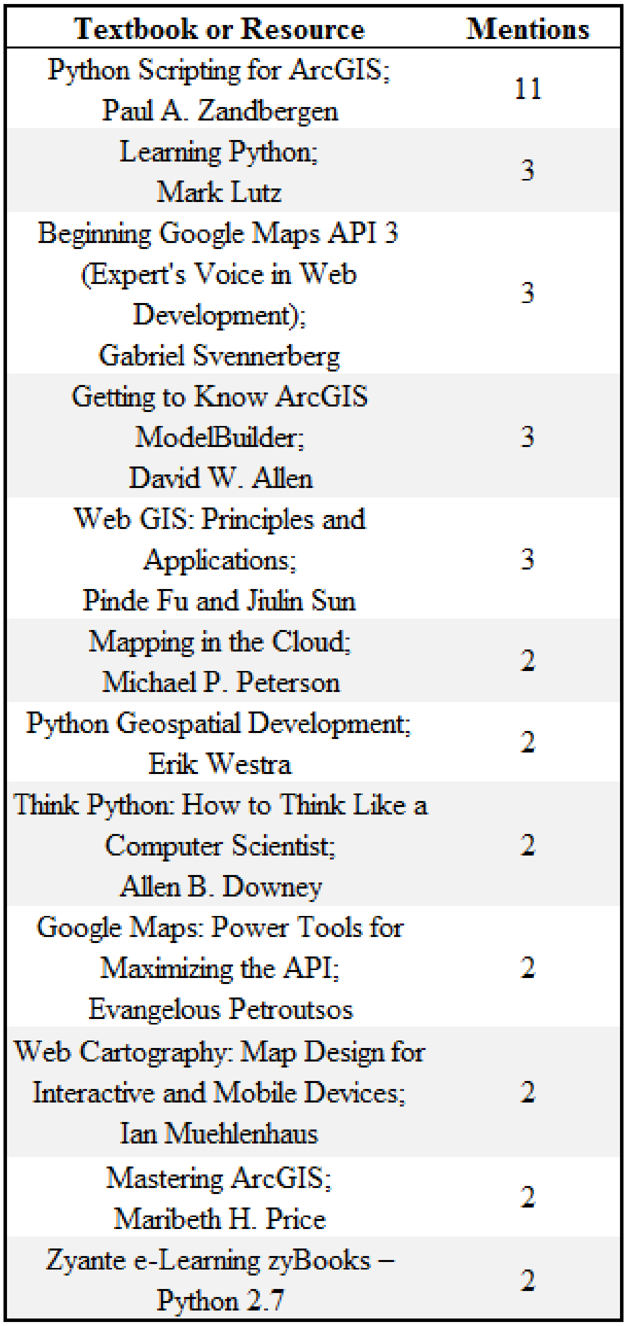
Table 4-7. Evaluation methods used in GIS programming courses. Students are generally evaluated through projects and lab exercises, with traditional methods like exams and papers seeing little use in these courses. Assignments of many types are also present, and include homework, discussions, reading, and other tasks.



Graded Component	Count
Projects	34
Computer laboratory assignment	22
Assignments (generally)	14
Presentations	7
Coding or programming assignment	4
Exams	4
Papers	3

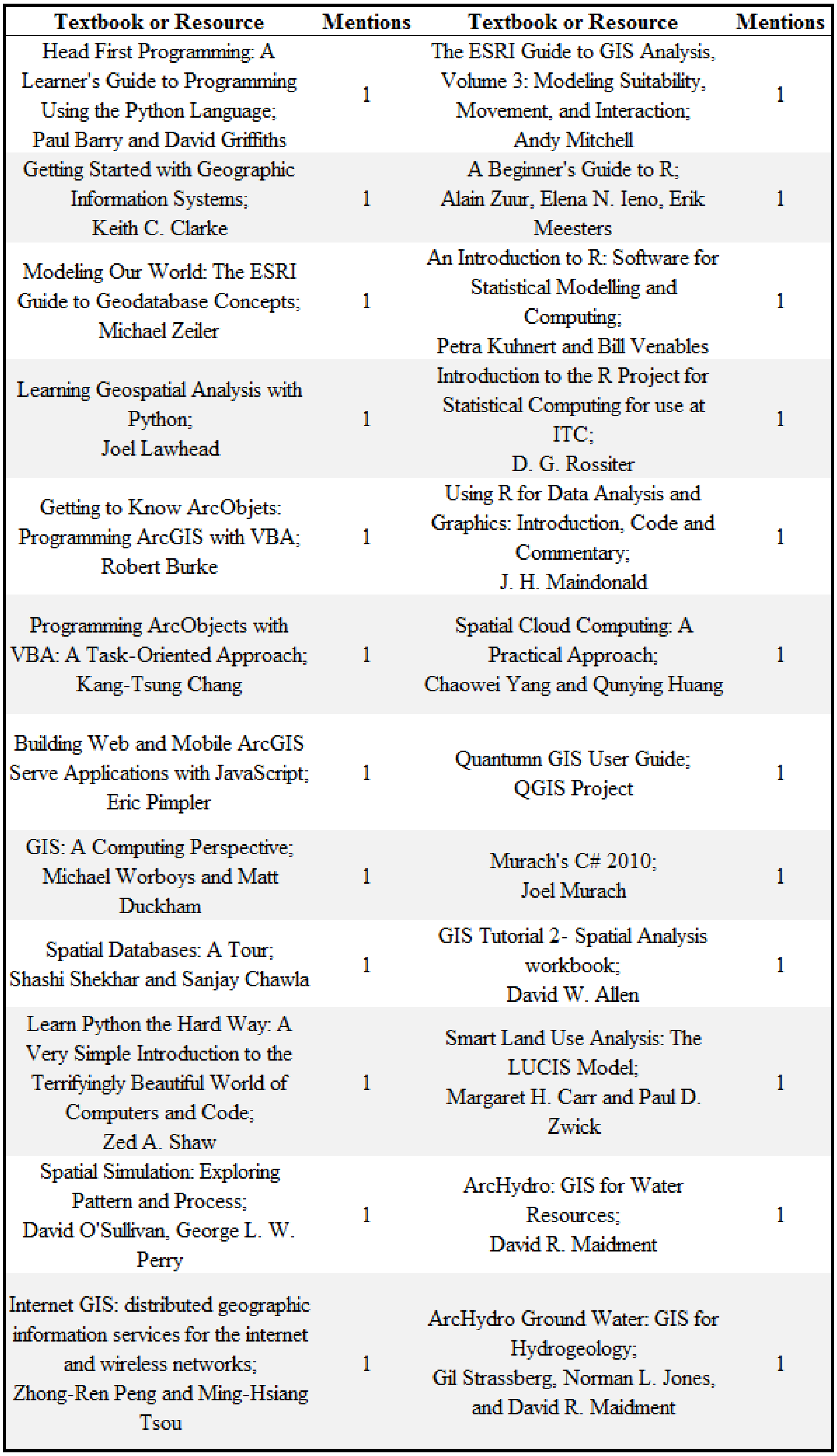
Textbooks used in at least two courses are shown in Table 4-8, and textbooks used in single courses are shown in Table 4-9.

Table 4-8. Textbooks used in GIS programming courses in multiple courses. While Python as a language is present throughout the sample of texts, only one text is present in more than a quarter of the courses. There are diverse instructional materials available for teaching computer science and programming topics in a GIS context, both for broad understanding of computer science, programming, and GIS interrelations, and for specific application focuses. This diversity is reflected here and in Table 4-9.



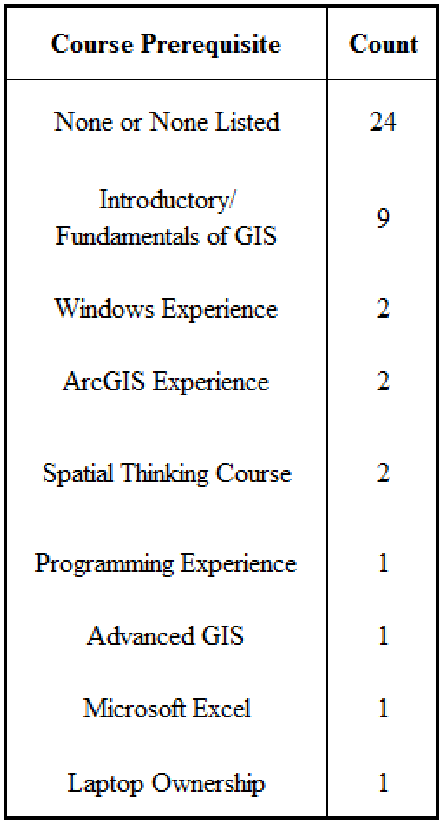
Textbook or Resource	Mentions
Python Scripting for ArcGIS; Paul A. Zandbergen	11
Learning Python; Mark Lutz	3
Beginning Google Maps API 3 (Expert's Voice in Web Development); Gabriel Svennerberg	3
Getting to Know ArcGIS ModelBuilder; David W. Allen	3
Web GIS: Principles and Applications; Pinde Fu and Jiulin Sun	3
Mapping in the Cloud; Michael P. Peterson	2
Python Geospatial Development; Erik Westra	2
Think Python: How to Think Like a Computer Scientist; Allen B. Downey	2
Google Maps: Power Tools for Maximizing the API; Evangelous Petroutsos	2
Web Cartography: Map Design for Interactive and Mobile Devices; Ian Muehlenhaus	2
Mastering ArcGIS; Maribeth H. Price	2
Zyante e-Learning zyBooks – Python 2.7	2

Table 4-9. Textbooks used in only one course. These texts further outline the wide variety of topics relevant to GIS programming, from further technical investigations domain-based applications.



Textbook or Resource	Mentions	Textbook or Resource	Mentions
Head First Programming: A Learner's Guide to Programming Using the Python Language; Paul Barry and David Griffiths	1	The ESRI Guide to GIS Analysis, Volume 3: Modeling Suitability, Movement, and Interaction; Andy Mitchell	1
Getting Started with Geographic Information Systems; Keith C. Clarke	1	A Beginner's Guide to R; Alain Zuur, Elena N. Ieno, Erik Meesters	1
Modeling Our World: The ESRI Guide to Geodatabase Concepts; Michael Zeiler	1	An Introduction to R: Software for Statistical Modelling and Computing; Petra Kuhnert and Bill Venables	1
Learning Geospatial Analysis with Python; Joel Lawhead	1	Introduction to the R Project for Statistical Computing for use at ITC; D. G. Rossiter	1
Getting to Know ArcObjets: Programming ArcGIS with VBA; Robert Burke	1	Using R for Data Analysis and Graphics: Introduction, Code and Commentary; J. H. Maindonald	1
Programming ArcObjects with VBA: A Task-Oriented Approach; Kang-Tsung Chang	1	Spatial Cloud Computing: A Practical Approach; Chaowei Yang and Qunying Huang	1
Building Web and Mobile ArcGIS Serve Applications with JavaScript; Eric Pimpler	1	Quantumn GIS User Guide; QGIS Project	1
GIS: A Computing Perspective; Michael Worboys and Matt Duckham	1	Murach's C# 2010; Joel Murach	1
Spatial Databases: A Tour; Shashi Shekhar and Sanjay Chawla	1	GIS Tutorial 2- Spatial Analysis workbook; David W. Allen	1
Learn Python the Hard Way: A Very Simple Introduction to the Terrifyingly Beautiful World of Computers and Code; Zed A. Shaw	1	Smart Land Use Analysis: The LUCIS Model; Margaret H. Carr and Paul D. Zwick	1
Spatial Simulation: Exploring Pattern and Process; David O'Sullivan, George L. W. Perry	1	ArcHydro: GIS for Water Resources; David R. Maidment	1
Internet GIS: distributed geographic information services for the internet and wireless networks; Zhong-Ren Peng and Ming-Hsiang Tsou	1	ArcHydro Ground Water: GIS for Hydrogeology; Gil Strassberg, Norman L. Jones, and David R. Maidment	1

Table 4-10. Prerequisites for GIS programming courses. Generally, most courses do not have a clearly defined set of requirements for entry into the programming courses in this sample. Of those that do, most seek students with previous GIS experience, though various specific technical experiences, skills, and familiarities are required as well.



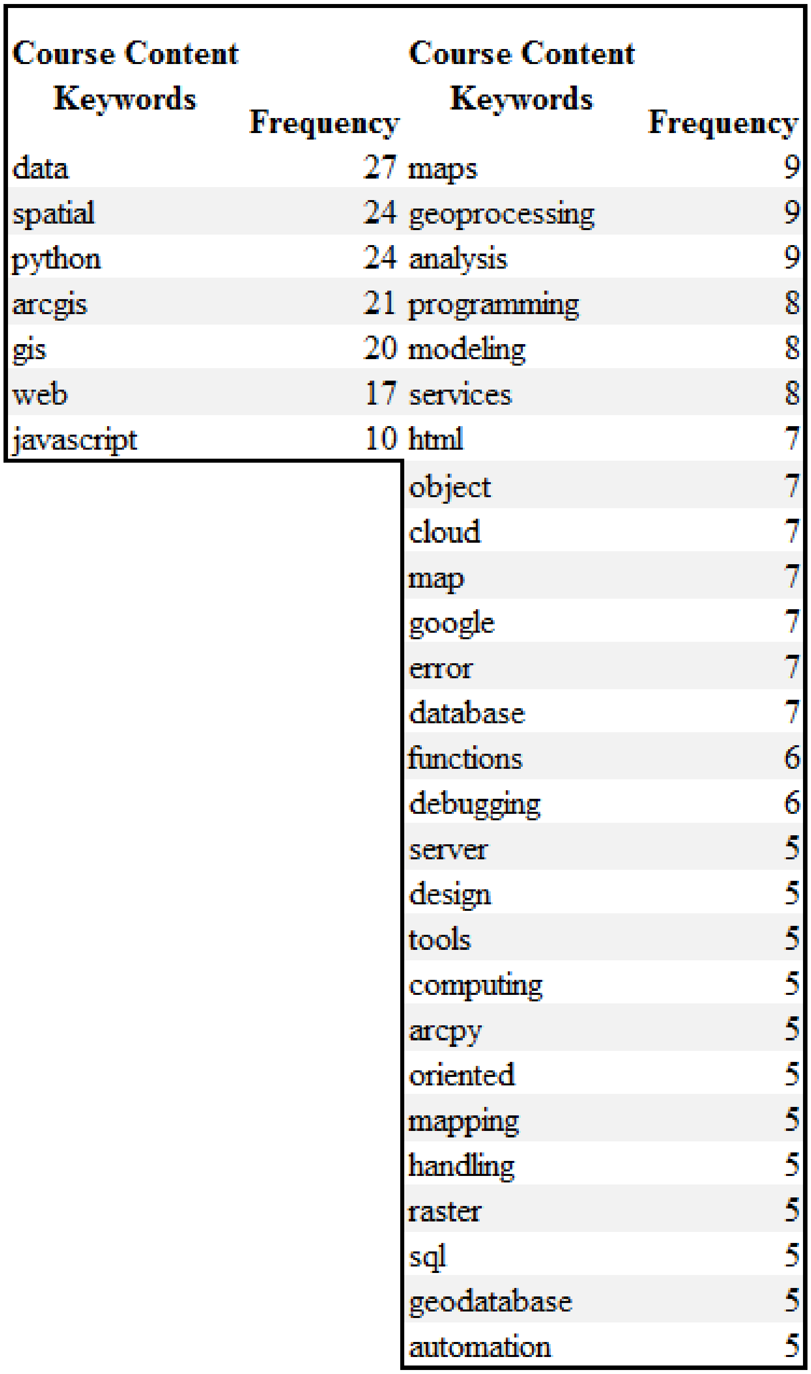
Course Prerequisite	Count
None or None Listed	24
Introductory/ Fundamentals of GIS	9
Windows Experience	2
ArcGIS Experience	2
Spatial Thinking Course	2
Programming Experience	1
Advanced GIS	1
Microsoft Excel	1
Laptop Ownership	1

Course pre-requisites also vary widely. As seen in Table 4-10, 24 courses list no requirements for enrollment in GIS programming courses, while nine required some introductory GIS knowledge, far more often than an introductory computer science or programming course (one case). A few syllabi indicated technical knowledge or hardware were required for their courses, such as capability with specific spreadsheet programs, operating systems, or laptop ownership.

Topics and Learning Outcomes

Topics included in GIS programming courses vary widely based on the focus of the course, the background of the instructor, and the balance of computer science, programming, and GIS topics. I list the 34 most common topics in these courses in Table 4-11.

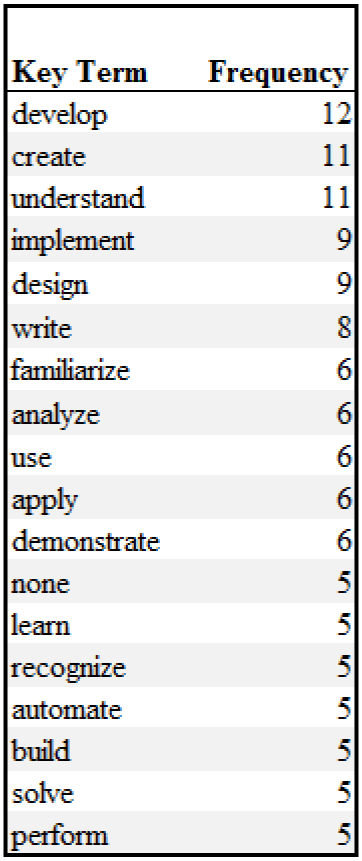
Table 4-11. Topics in GIS programming courses. Broadly focused on using spatial data, courses generally relate Python as the language of choice in GIS programming, though JavaScript, HTML, and SQL are present as well. Topics listed in fewer than five courses are omitted, and topics only are counted once per syllabus.



Course Content Keywords	Frequency	Course Content Keywords	Frequency
data	27	maps	9
spatial	24	geoprocessing	9
python	24	analysis	9
arcgis	21	programming	8
gis	20	modeling	8
web	17	services	8
javascript	10	html	7
		object	7
		cloud	7
		map	7
		google	7
		error	7
		database	7
		functions	6
		debugging	6
		server	5
		design	5
		tools	5
		computing	5
		arcpy	5
		oriented	5
		mapping	5
		handling	5
		raster	5
		sql	5
		geodatabase	5
		automation	5

The practice of manipulating spatial data with Python, a programming language, emerges as a common outcome in these courses. This Python based data manipulation and scripting occurs most often through GIS and using the dominant GIS software, ArcGIS. As also noted in Table 4-3, WebGIS is a second area of emphasis. While Python is the dominant language taught and used in these courses, JavaScript, HTML, and SQL also appear as common programming languages in course instruction. JavaScript and HTML both serve to web-enable GIS for data collection and display, among other uses. SQL, a database access language, serves as a management tool for spatial data. These three themes: Python scripting, Web-enabled GIS, and database access dominate the course structures. Students are expected to explore and become functional practitioners in one of these themes through their course experience. Regardless of course focus, however, students are generally expected to be able to 'develop' 'create' and 'understand' this content through the learning outcomes of the courses (Table 4-12). Overall, I identified 89 different terms indicating levels or types of performance in the learning outcomes in this sample, even though five courses included no learning outcomes at all.

Table 4-12. Key terms used in learning outcomes in GIS programming courses. I identified 238 total terms used in syllabi, with 89 unique terms used. These terms relate to objective verbs used in Bloom's Hierarchy which relate to cognitive levels of information recall. Syllabi often ask students to 'develop', 'create', or 'understand' components of GIS programming, though those terms are hardly dominant among syllabi reviewed.

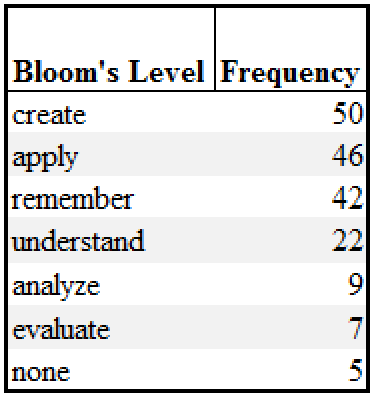


Key Term	Frequency
develop	12
create	11
understand	11
implement	9
design	9
write	8
familiarize	6
analyze	6
use	6
apply	6
demonstrate	6
none	5
learn	5
recognize	5
automate	5
build	5
solve	5
perform	5

Levels of Performance

Learning outcomes can be arranged in knowledge hierarchies. I used Bloom's Revised Taxonomy to outline the dimensions and levels of learning expected in these objectives (Krathwohl 2002). As outlined in Table 4-13, of the 176 learning outcomes analyzed in the 35 syllabi that provided outcomes, students are most often asked to 'create' (joining elements together, the highest level on the taxonomy), ‘apply’ (using a situation-appropriate procedure, the fourth-highest level on the taxonomy), or ‘remember’ (retrieving information from memory, the lowest level on the taxonomy) course content.

Table 4-13. Frequency of Revised Bloom’s Levels in GIS programming course objectives. The fundamental ideas, application of concepts, and creation of unique GIS interfaces that integrate topics in computer science and programming are dominant. Analysis and evaluation are less prominent, and five syllabi in this sample contained no learning outcomes at all.



Bloom's Level	Frequency
create	50
apply	46
remember	42
understand	22
analyze	9
evaluate	7
none	5

Figure 4-3 displays these levels within the revised Bloom’s hierarchical structure. This hierarchical organization allows a view of the set of expectations within this sample, and those expectations cluster at the base and peak of the hierarchy. The gaps in analysis and evaluation can diminish students’ creative abilities, especially relating to connecting components within a new construct, or defending the choices they make in constructing a new workflow, model, or other course output.

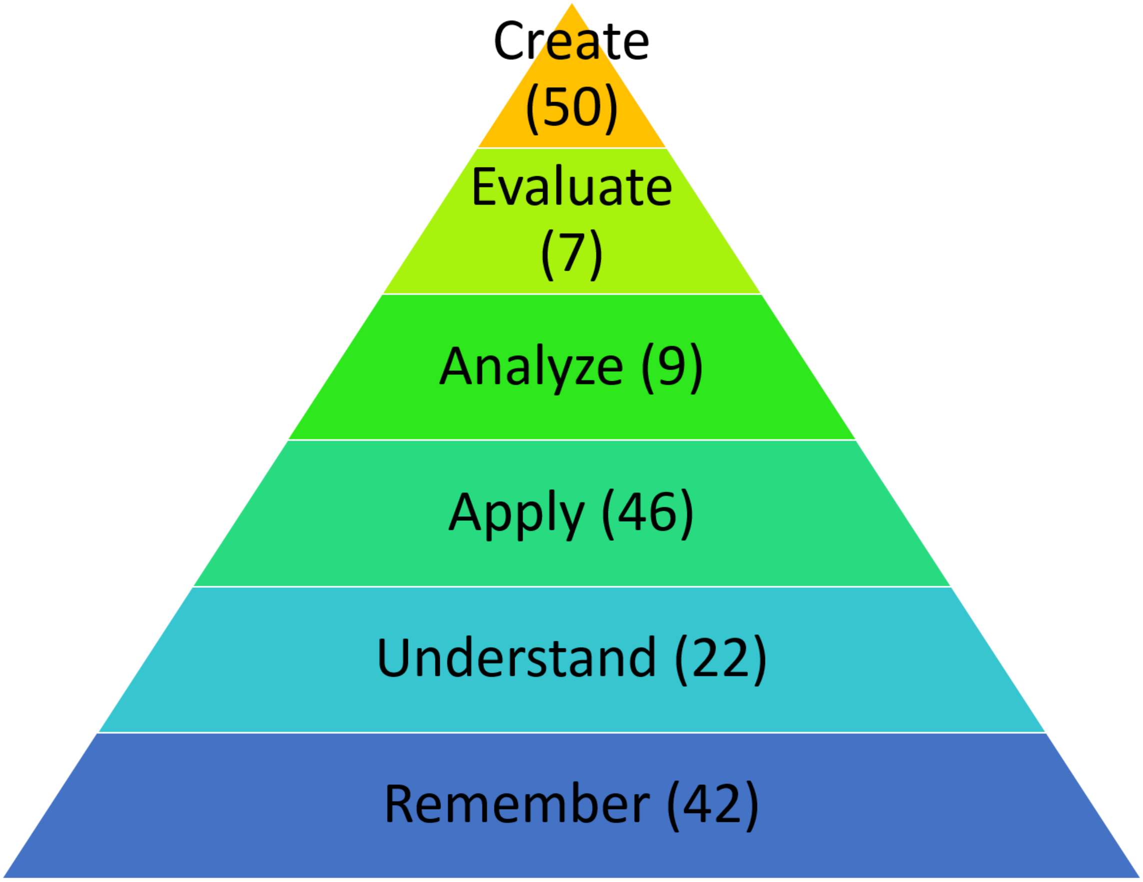
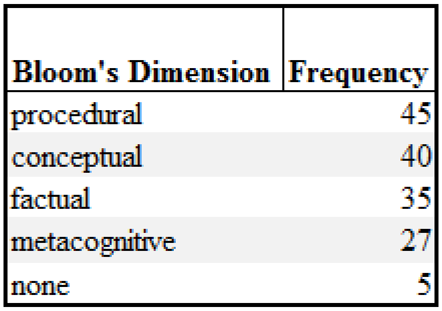


Figure 4-3. Bloom's revised cognitive hierarchy with count of terms in this sample. Most learning outcomes in this sample are at the base (remembering) and peak (creating), and middle (applying) of the hierarchy, with gaps in the higher levels of analyzing and evaluating. The five submissions which did not list learning outcomes are omitted here.

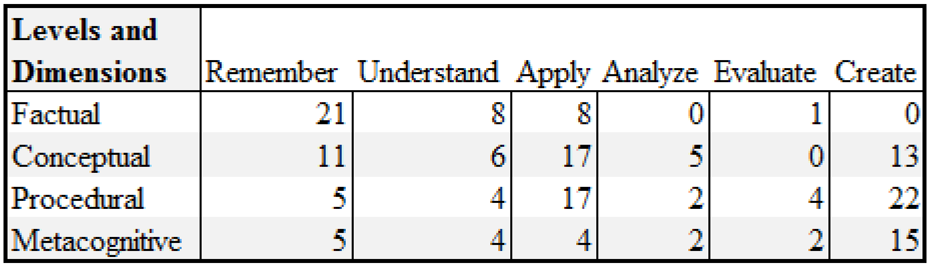
These cognitive processes also contain expectations for knowledge level as defined by the outcomes. These four levels, factual, conceptual, procedural, and metacognitive, represent what elements of knowledge the outcomes seek through the cognitive process. Factual knowledge includes basic elements of components, and was the focus of 35 objectives. Conceptual knowledge interrelates elements to build an understanding of how knowledge components function together, and was the focus of 40 objectives. The most common dimension was procedural knowledge (45 objectives), the methods and criteria which help define how to do something. Metacognitive knowledge, or knowledge about how cognition functions in the domain, is least present in these courses (27 objectives), contrasting the similar frequencies of the other dimensions (Table 4-14). Considering these components together (Table 4-15), lower items on the cognitive domain match with simpler dimensions, while higher items on the cognitive domain match with more complex dimensions.

Table 4-14. Frequency of Bloom's dimensions of learning outcomes. No dimension is dominant, though 'procedural' and 'conceptual' dimensions of course objectives are more common. Integrating programming and computer science skills with GIS requires certain capabilities in understanding coding processes and the ontologies of computer structure, perhaps accounting for the dominance of 'procedural' and 'conceptual' dimensions.



Bloom's Dimension	Frequency
procedural	45
conceptual	40
factual	35
metacognitive	27
none	5

Table 4-15. Revised Bloom's levels and dimensions. While 'create', 'apply', and 'remember' are the most common Bloom's levels (as shown in Table 4-13), and Bloom's dimensions are distributed fairly evenly (Table 4-14), they function mostly increasing in step in complexity. Most 'remember' objectives function at a factual level, while most 'create' objectives function at a 'procedural' and 'metacognitive' level. Determining how these objectives play out in a classroom setting is a notable area for further investigation.



Levels and Dimensions	Remember	Understand	Apply	Analyze	Evaluate	Create
Factual	21	8	8	0	1	0
Conceptual	11	6	17	5	0	13
Procedural	5	4	17	2	4	22
Metacognitive	5	4	4	2	2	15

Content Presentation by Syllabi and Instructors

This research discovered wide variation in the curricula, structures, and foci of these courses, aimed broadly at facilitating computer science and programming knowledge in the context of GIS. As a general goal, GIS courses grow student knowledge, skills, and practices in computer science and programming within a specific context. For example, the course in environmental planning challenges students to:

"think critically about environmental problems, break them into individual components (and represent them) using geospatial technologies."

In this case, the course expects students with a strong environmental planning background to apply their new and novel knowledge, skills, and practices related to geospatial technologies, including computer science and programming components, to critical environmental problem solving. Other courses take an approach of using a specific programming language (often Python) as a way to expand students' spatial analysis capabilities, arguing that:

"Python is a very popular and easy to learn language. Knowledge of Python is a highly desirable skill for GIS analysts, while the automation makes tedious GIS tasks easier, faster and more accurate."

There is little consensus on the scope of these courses. The framing and situation of these courses within the larger knowledge base of geography, GIS, computer science, and programming yields a tremendous variety of approaches. As noted in the results above, instructors in these courses have variant learning expectations for their students, who encounter fundamentally different exposures to GIS, computer science, and programming, and may use highly disparate core instructional materials to develop knowledge. In this section, I outline these characteristics towards an understanding of the structure of courses in this set of syllabi, outlining their similarities and differences, and informing my core research questions.

Overview of Course Structures

As noted in Table Three, these courses vary in focus but all broadly center on teaching GIS with computer science and programming. Even the most typical course in GIS programming shows noticeable variations in learning outcomes, instructional materials, and methods. One syllabus introduces the course by saying:

"Python is a free and open-source scripting language. Python scripting is used to automate tasks, making work in ArcGIS faster, easier, and often more accurate. Python scripting allows ArcGIS users to automate complex processes and carry out sophisticated data analyses. Knowledge of Python scripting is in high demand in the GIS job market."

Python is the obvious focus of this course. Here, Python scripting is a practice for easier, faster, and more accurate work within an essential tool of many GIS users: ArcGIS. Another syllabus shares this conception:

"GIS users often find reasons to invest time in the automation of data management, processing, analysis, or visualization tasks. Reasons for automation might include carrying out repetitive tasks, executing a task on a routinely scheduled basis, processing very large datasets, or stringing together multiple processing steps."

Again, Python serves as a facilitator, providing processing powers and reducing the tedium of GIS analysis. Introductory courses in this sample share this framework, that programming capability, especially in Python, is an advantageous and meaningful benefit to of a student in GIS. While some previous experience with GIS is noted as valuable to students entering the realm of GIS programming, prior programming experience is not a requirement. Indeed, some syllabi emphasize that no such experience is necessary, with one syllabus stating:

"You do not need any experience in writing code, as you will learn the fundamentals of programming in this course."

These courses build key GIS skills and expose students to fundamental skills in computer science and programming. While many approaches exist in introducing and building student knowledge in this area, a common structure and organization emerges surrounding the textbook used most in courses in this sample: *Python Scripting for ArcGIS* by Paul A. Zandbergen. Of the 11 mentions of this text as a component of courses in this sample (Table 4-8), eight of the courses are general GIS programming courses, and five of those follow a generally linear path through the content of the textbook, with some shuffling of the later chapter content.

A number of different formats for WebGIS and Cloud GIS courses also exist in this sample. These courses seek to build students' capabilities in hosting spatial data on websites or in performing spatial analysis through distributed computing solutions. In many cases, these courses practice GIS removed from the traditional desktop computer environment, and diversifying occurrences of spatial analysis. These new considerations of spatial analysis capabilities include technologies like servers, mobile devices, and other non-desktop entities. The introduction provided in one syllabus for a WebGIS course states:

"This course is intended to provide a broad introduction to the societal implications of and technologies behind non-desktop based mapping and GIS applications. The evolution of the web has led to a movement from a simple information delivery platform to one in which rich content and functionality can be delivered to any device with sufficient connection speeds. Geospatial applications are increasingly becoming a significant part of this evolution with an ever growing number of mapping and GIS service applications becoming available."

Thus it appears that knowledge of web and cloud applications are growing in significance in GIS curriculum. The broad appeal and utility of web maps and applications of cloud data are highly motivating factors in including capabilities in these areas in courses. As one syllabus explained:

"Learning to program innovative web-based mapping applications facilitates sharing and dissemination of your work, and at the same time vastly expanding your overall application development skillset. Familiarity with web scripting languages and how these are utilized to implement Web GIS applications provides in-depth insight into how many government and commercial organizations as well as individuals develop these tools."

A wide variety of web tools and languages exist to develop web and cloud-enabled content. Syllabi in this sample focus on relevant development capabilities that interact with key platforms for delivering spatial content like Google Maps and ArcGIS Online. Rather than Python, these courses in contrast focus on JavaScript, HTML, and Application Program Interfaces (APIs). All of the ten mentions of JavaScript and seven mentions of HTML as shown in Table 4-11 are in Web or Cloud GIS courses. Contrasting with the introductory programming courses, these courses lack a cohesive text or set of instructional materials.

The third common curricular focus is that of geodatabases. Spatially enabled databases drive the storage and manipulation of information used by GIS. Spatial databases require care in design to facilitate spatial queries and spatial manipulation of data, a key technical underpinning of GIS. One course emphasized understanding data structure introducing the topic as a core effectiveness of the GIS user:

"In order to be an effective GIS (Geographic Information Systems) analyst, one must not only be aware of the available GIS tools and capabilities. Rather, one must also be able to acquire, create, and manage spatial data. Ultimately, this knowledge will result in time and monetary savings, as well as minimize analytical errors."

In this case, the practice of knowing how data is acquired and stored is identified as an important capability of a GIS analyst. Also notable is that this syllabus uses a systems approach for teaching GIS, placing additional emphasis on the computer science and programming relevance in this course, rather than the spatial science aspects of GIS. Using computer science concepts and programming languages like SQL, PostgreSQL, and database management software, these courses face the challenge of balancing the complexities of database design with the necessary spatial nature of GIS data. Another syllabus outlined these necessary competencies by saying:

"The main objective of this course is to introduce students to scripting/programming techniques designed to describe, analyze, transform, display, and make inferences from geospatial datasets. Students will extend the capabilities of commercially available geographic information systems (GIS) software packages through the development of custom/tailored computer programs."

Students in this course develop and extend functionalities as well, all centered on the key technology of geospatial datasets. While the specific software present are not noted, the note of 'commercially available' indicates that students will interact with software commonly used in GIS practice as a platform to demonstrate their computer science and programming abilities. However, these courses and the others submitted do not reveal any shared instructional materials or content that provides considerable enlightenment on how best to approach this fusion of content.

Instructor Comments

Some instructors provided additional context as to how their courses function. Comments on challenges that instructors perceive for their students revealed some of the difficulties in building these courses. One instructor of a WebGIS course noted:

"Students struggle to think in an automated way -e.g. how to dynamically set file names in a loop for created data. Students get frustrated with errors."

The negative feedback associated with learning in computer science is a major challenge for novices learning the fundamentals of computer science and programming. The frustration that arises from failure in this domain serves as a significant barrier for students as they attempt to grasp fundamental tasks. Another WebGIS instructor added that:

"Some students already have programming experience. Others have none. For many the class is too fast, for others it's too slow."

This unevenness in preparation can complicate already challenging fundamentals of class structure, like one introductory GIS programming instructor noted:

"I'm having a hard time cramming all the content I would like into each two-and-a-half hour class meeting. We start with questions, then quiz, then in-class activity. However, sometimes we spend 30 minutes or more on questions; by the time the quiz is done (and other housekeeping details are finished) we only have (an hour and a half) left for the exercise, which often times is not enough."

However, this instructor found success with a specific method of course construction:

"This year I implemented the flipped classroom, which has been great. Far fewer frustrated students, and they seem more engaged in the material. This has been reflected in their first round of presentations, which have been much better than last year."

Flipped courses rely on students to watch lectures or consider foundational materials outside of the classroom while completing hands-on exercises during class meetings (Tucker 2012). Combined with the use of projects, group work, and other non-lecture based methods, these interactive, problem-based methods of instruction are commonplace within the syllabi collected here.

Integration of Computer Science, Programming, and GIS

Each of these courses seeks to integrate some detailed understanding of computer science relevant to advancing or informing spatial analysis capabilities in GIS. These courses all view the relevance of the computer science or programming topic through the lens of its usefulness to GIS, except for the courses that situate GIS, computer science, and programming as relevant to environmental studies. Since the instructors of these courses are generally not computer scientists, but geographers and GIS practitioners, these courses generally describe the computer science and programming components in terms of their usefulness to GIS.

The courses in this sample generally focus on building fundamentals of the computer science topic in the course lectures and the overall course structure, but integrate that computer science topic through assignments, projects, and other creative activities. Many courses ask students to create content, either in developing functioning systems for GIS analysis with servers, scripting, or other components, or by providing a solution to some problem or objective solvable with the computer science or programming topic in the course. As noted above, the topical focus of a course will change depending on the capabilities of GIS to be extended. Python is a scripting language that allows extension and expedition of analytical capabilities; JavaScript and HTML function to web-enable spatial analysis and data hosting, while SQL helps build databases to facilitate spatial data storage. So, though Python is the most common language in this sample, it serves as one pillar of an emerging GIS programming trinity that must also include SQL and JavaScript. HTML, as a markup language, might be considered for geographers as important as knowledge in ArcGIS or common spreadsheet or word processing programs: a fundamental technology utilized in the professional and academic realm.

Though there is no clear and distinct language or utilization that dominates the integration of computer science, programming, and GIS, these all merge under the general umbrella of cyberinfrastructure. As noted by one syllabus that introduces the term:

"Just as physical infrastructure provides services such as electricity, plumbing, and road networks to communities across the world, cyberinfrastructure has emerged to provide computational services and capabilities to scientific communities. Cyberinfrastructure integrates high-performance computing, digital sensors, virtual organizations, and software tools and services to facilitate computationally-intensive and collaborative scientific research."

The topics in the syllabi I collected all relate to providing geographers access to the wide suite of computing infrastructure available in industry and research. The course noted above describes this realm as CyberGIS, a conception of GIS that integrates many of the topics described in this article. As this CyberGIS course syllabus continues;

"CyberGIS, broadly defined as cyberinfrastructure-based geographic information systems, integrates cyberinfrastructure, geographic information systems (GIS), and spatial analysis to enable collaborative geographic problem solving."

While CyberGIS is a growing field of research and discussion in how to conceive GIS, I only received a single syllabus on the topic. Positioning and exploring the computing infrastructure as a backbone of facilitating modern knowledge of GIS could serve as a holistic method to introduce and integrate GIS and computer science concepts. An awareness of the relevance of these topics in GIS could serve as an important base for deeper, more application-focused courses in later academic work.

Conclusion

I conducted an investigation of course syllabi in GIS programming. Since programming and computer science skills and abilities are rapidly growing in relevance to GIS users, I sought to understand what components make up GIS programming courses, what common instructional materials exist in these courses, and what general structure is present in these courses. Generally, syllabi in this sample show a preference for Python, and follow a structure that emphasizes the specific programming topic in lectures, but encourages creation and integration of GIS and that topic in exercises and projects. While one Python-centered textbook was most common across these courses, it existed mostly in introductory GIS programming courses, which was the primary course type in the sample. No other common resource or structure of courses was apparent. Finally, though considerable variation in learning outcomes and tasks are present, the development, creation, understanding, and remembering of key components in these domains are dominant.

Research Question Summary

I posed three distinct research questions which structured this analysis. I recap and discuss them below.

What is the nature of computer science and programming topics in GIS programming courses?

I identified 34 topics present in at least five syllabi in this sample. These topics range from specific programming languages, of which Python was the most popular, to general computing terms like data and modeling, to GIS terms with which computer science and programming would be used to analyze, extend, or otherwise synthesize, like ArcGIS and geoprocessing. Broadly, these courses concern processing spatial data with various computer science and programming skills. I identified three distinct structures of integration within GIS for these topics. Most common was a general GIS programming structure that uses Python to introduce topics in scripting and extending core functions of ArcGIS and other desktop GIS programs. Another uses JavaScript and HTML to web-enable GIS for use in cloud and server based hosting and analysis. A final structure uses SQL to facilitate learning of database structure in and data management for GIS. While other course structures did appear in small numbers, the repetition of these course structures provided the clearest understanding of how to extend fundamental GIS knowledge into a more specific computer science realm.

How does the nature of these courses support the development of key GIS competencies given growing use of computer science and programming in GIS?

Using Bloom's Revised Taxonomy, courses in this analysis generally structure information at the base and peak of the knowledge hierarchy. This means that students are more often asked to 'understand' or 'remember' components of these courses, aimed at basic recall of information, or 'create' and 'develop' integrated GIS and computer science applications, aimed at high level understanding of knowledge interaction. Since these courses are generally structured within geography departments and taught by geography or GIS faculty, care in scaffolding student learning to build towards high-level outcomes with computer science is necessary, as these courses generally required no experience in coding, programming, or computer science for student enrollment.

What is the sequence of learning in these courses, and is there any curricular coherence in how these courses are presented?

Aside from the course structures noted above, no clear structures or models were evident in this analysis. While one textbook, *Python Scripting for ArcGIS* by Paul A. Zandbergen, was used in about a quarter of the courses in the sample, these courses were primarily introductory GIS programming courses, and no resource or topic was prevalent in the other structures or across course types. Most courses rely on lectures to deliver fundamental knowledge and structure projects and assignments with expectations for creation and development of course content.

Recommendations

There is a distinct need for computer science and programming in GIS. The affordances of advanced computer infrastructure, computing capabilities, and programming knowledge are vital to extending and expanding the reach, influence, and capability of spatial analysis in GIS. While this analysis reveals three distinct topical structures for content and information in this realm, it raises more questions about how these courses present information and are structured within the general GIS knowledge base. Since no course in this sample explores components of all three of these areas of interest (scripting, web-enabling, databases), students might remain unprepared in two key areas of GIS programming if they take a course that focuses on one side of the GIS-computer science interface. Students familiar with diverse applications of computer science in GIS should have an advantage in understanding different ways GIS and computer science integrate. I therefore recommend investigations into what components of programming and computer science are most vital for the GIS user, whether in academics, professional employment, or otherwise, to build on this syllabus understanding to propose, design, or reconstruct academic training moving forward.

Another disparity of note worth further investigation is the way syllabi are constructed, especially in terms of learning outcomes, learning progressions, and scaffolding of knowledge in the GIS and computer science realms. Some syllabi in this sample include no learning outcomes at all, and others do not describe course materials or reference resources outside of general readings selected by the instructor. Research on individual courses, including case studies of how GIS programming courses function, would provide meaningful evidence regarding what objectives are appropriate for novices in computer science with an extensive GIS background.

Finally, work needs to begin on identifying instructional materials usable in these contexts. Zandbergen's Python book is relatively commonplace in this sample serving as a textbook which instructors use to structure syllabi. Determining what other materials exist for complementary topics, including additional resources in computer science and programming which are relevant to the other instructional foci discovered here, will be important areas for investigation. Considering that I identified 38 explicitly identified instructional materials across 40 syllabi, and that some syllabi outlined no materials at all in their construction, any evidence-based understanding of how course topics align instructional materials and course outcomes is vital. Further guidance to text relevance might be obtained by asking professional GIS users what text, manuals, or guides shape their post-academic learning of GIS, computer science, and programming.

GIS, computer science, and programming are not easily separated in state-of-the-art instruction. While this work begins an understanding of how these topics merge, more work is necessary to strengthen our conceptions of how GIS, computer science, and programming function in an academic sense.

CHAPTER V

CHAPTER SUMMARIES AND CONCLUSION

Chapter Summaries

This dissertation investigates the many connections and interrelations among GIS, computer science, programming, and GIS education through an analysis of courses, degree plans, and user viewpoints. Here, I first summarize the findings of my previous research chapters, while also expanding on the connections and linkages between the parallel studies composing this dissertation. With this context of new discovery and interconnection in mind, I also outline potential research lines to build off the discoveries outlined in this dissertation. Additionally, I discuss further work underway that is relevant to expanding on the research articles herein. By synthesizing the studies, I summarize my findings and establish a context for the next phase of investigations. Overall, this chapter highlights the vibrant nature of this work in a holistic, consolidated fashion.

Major Findings

CyberGIS, a synthesized and interdisciplinary field where the individual components of cyberinfrastructure, GIS, and spatial analysis merge, is the frontier of GIS. This dissertation investigated curriculum and instruction in this rea through three means: a review of the ways computer science and programming is being integrated into GIS and geography degrees, by interviewing experienced GIS users to determine their viewpoints on the value of CyberGIS components, and by examining programming syllabi to determine what instructional materials, methods, and course components are dominant in state-of-the-art coursework. In this chapter, I summarize the major findings and conclusions that emerged from answers to my guiding research questions. Overall, computer science and programming courses containing instruction relevant to the core components which make up the enabling technologies of CyberGIS are widely present, but not widely required, in GIS degrees. In outlining three different viewpoints of the components of CyberGIS, this is not surprising, as some experienced GIS users see the computer science and programming components merely as topics to be utilized when focusing on more important spatial or domain-based problem solving. Those who do value the computer science and programming topics more might be teaching GIS courses like those collected here, but they might also be teaching them in distinctly different ways. There are numerous ways to connect GIS and geography to computer science and programming ideas and skills, and courses in scripting, web-enabling, and databases all show the vast reliance of GIS on computer science knowledge, but also the grand opportunities for collaborations and developments of impactful learning opportunities.

Primary Findings and Implications: Chapter Two

The first article, in Chapter Two, considered the construction of geography and GIS degrees through course descriptions in highly ranked geography departments in the United States. This review discovered that most departments (44 of 55) include some type of GIS programming course within their course catalog. However, of the 210 separate degree tracks identified among these departments, only 22 (~10%) required one of these courses for completion of the degree. Many factors may contribute to this dearth of this requirement of programming coursework: no awareness at the administrative or degree planning level of the necessity of these skills in the workforce; no source of motivated and capable instructors to teach these courses or who feel comfortable developing these courses; inability to manage the material requirements of these courses, in cost, computer lab space, developmental time, or otherwise, and other factors, ranging from whether GIS ‘belongs’ exclusively to geography programs, to the culture and capabilities of individual departments and instructors.

This review provides a first step into understanding the state-of-the-art in general instruction in GIS, as these course descriptions form a useful platform to discover broad trends across a diverse set of geography departments. This scale is relevant for identifying patterns in course composition and degree requirements, and can serve as a starting point for future research along this track, potentially researching whether structural differences in these departments or universities (i.e. land grant, flagship, etc.) contribute to these differences, or if with further review of geography department structures, more patterns can be discovered to indicate where computer science and programming courses are more commonly required. Courses in computer science and programming are clearly present in most departments, but why they are not required remains an important mystery to resolve.

Primary Findings and Implications: Chapter Three

In the second article, Chapter Three, I constructed a table of key terms in CyberGIS, and explored how users value those terms in their experiences. Initially building the set of terms through a detailed literature review and the assistance of machine reading, these terms then formed the basis of interviews using q-methodology. Through these q-method interviews, I identified three distinct types which represent expert views of CyberGIS components. These views, titled the spatial analyst (type one), the domain-specific problem solver (type two), and the CyberGIS enabled computer scientist (type three), reveal quantitatively distinct ways of conceptualizing terminology in CyberGIS, which function in different relative views of topical importance. The identified components of CyberGIS are valued unevenly across viewpoints, with major disagreements on some GIS and GISci fundamentals, like ArcGIS, spatial modelling, and the role of computer science and programming. These variant viewpoints provide meaningful insight into how experts with different domain focuses view and understand CyberGIS.

In essence, the overall value of GIS, GISci, and spatial problem solving is viewed through these factors. One group values the underlying technologies (type three), which allow a group of GIScientists (type one) to produce advanced methods and processes, which are then utilized by a group to solve domain-based problems (type two). Components of CyberGIS do facilitate spatial analysis and problem solving that would otherwise be impossible. However, this research indicates that significant differences exist in how experts view and utilize these components, due to their experiences and focuses working with the various technologies and topics that compose the CyberGIS core.

Primary Findings and Implications: Chapter Four

The final article, Chapter Four, analyzed the structure and content of GIS course syllabi. Since programming and computer science skills and abilities are rapidly growing in relevance to GIS users, this chapter functions as a beginning investigation into what components help build an understanding of GIS programming, how courses in this realm are composed, what common instructional materials exist in these courses, and if a general structure of how these courses function exists. Generally, syllabi in this sample show a preference for the Python programming language, but are not exclusively Python based. Each course emphasizes the language of choice primarily through lectures, but encourages creation and integration of GIS and the programming topic in exercises and projects. While one Python-centered textbook was most common across these courses, it existed mostly in introductory GIS programming courses, which was the primary course type in the sample. No other common resource or structure of courses was apparent. Finally, though considerable variation in learning outcomes and tasks are present, the development, creation, understanding, and remembering of key components in these domains are dominant. This variation reveals a gap between the cognitive skill levels present in these courses, indicating a need for more structure in guiding students along their learning outcomes.

Limitations

Despite the wide-ranging nature of these investigations, some limitations, both in methodology and in applying research findings are worth noting and discussing in more depth. Considering the research holistically, it is important to note that while this document operates at different scales and uses different types and sources of data, that the education materials, interviews, and information gathered here may not be representative of the current geographic landscape as a whole. Future research must continue to expand the reach of these analyses to better represent the full spectrum of instruction and geography coursework. This research serves as a meaningful first step in understanding the fusion of GIS, programming, computer science, and geography, but is by no means meant as an authoritative summary of all education actions present in these domains.

Chapter Two uses course descriptions from ‘highly ranked’ geography departments to build an understanding of how programs integrate programming coursework. However, academic rankings are controversial and lack consistent application. While I used two separate ranking systems, that generally overlapped in the institutions they contained, this was done out of convenience and to rely on other entity’s established determinations of ‘highly ranked’, rather than as an endorsement of their rankings. Determining how geography ranking metrics correspond to potential GIS rankings deserves further consideration. Another issue with finding geography programs is the growing separation of GIS degrees or programs from geography departments. There is no guarantee GIS programs or degrees in other departments share the same structures outlined here, but those programs would provide interesting comparisons for future analyses.

Chapter Three uses machine reading to infer topics gathered from an extensive review of CyberGIS literature. Even with the advantages of topic modeling through MALLET, some terms of importance may have escaped my analysis. This limitation revealed itself in early validation testing, which necessitated the addition of the term ‘JavaScript’ even though it was not noted prominently in the early analysis. While the additional validation and revision of the term list that resulted from the q-methodology work is further improved, it may need revisions as well as technology and integrations progress. These may be considered snapshots of key concepts, and will evolve and shift over time and with further research.

While robust in determining quantitative descriptions of subjective ideas, the q-method portion of Chapter Three is limited to describing the participants in the sample. While the viewpoints described in the q-method analysis likely exist outside of the sample, the landscape of viewpoints is considerably more complex. How these viewpoints change based on individuals' backgrounds (geographers vs. non-geographers) could provide additional depth for future studies, for example. Q-method is designed for small sample sizes and qualitative validation, so additional efforts, using the revised set of q-terms, should seek out further disparate views and considerations of how geography, computer science, programming, and GIS function. This investigation begins the discussions of how to consider these components, but should not remain the only framework of understanding these viewpoints.

The syllabi and instructional material review in Chapter Four uses numerous syllabi and types of course constructions to build conclusions, but syllabi may not always represent course learning or course content in a direct and clear fashion. Adapting to student needs, adjusting course content based on student abilities and learning progressions, or other shifts in course construction throughout a semester are meaningful and important pieces to how courses function. Though syllabi are important artifacts of course plans and designs, they should not be taken as completely pure and representative descriptions of how the course existed. Instructors who create these syllabi may not even be familiar with writing effective learning outcomes, making representations with Bloom's inaccurate for considering the nature of these courses. More in-class research is necessary to understand how these courses develop over time.

Linkages

While the research articles in Chapters Two, Three, and Four function as individual entities, they also provide insight into how these components function together across the dimensions and scales of GIS education. The design of this dissertation was to investigate key educational components through different instructional scales, theories, and documentation, providing multiple means of understanding the GIS, geography, computer science, and programming fusion present both in the CyberGIS framework and evident in current instruction. These studies, when considered together, also reveal interesting relationships between components. In this section, I explore the connections among the papers, and ideas from the dissertation when considered as a whole, to deepen the findings present across the document and supplement the connections as a whole.

Cross-Component Connections

These articles form a cohesive argument outlining the importance of computer science and programming to GIS and geography degrees. With little work exploring these key areas, this dissertation provides meaningful insight and guidance in the field. There is no consistent conception of GIS. The applications, capabilities, and extensions of GIS have made it an ephemeral, fuzzy entity that reaches and extends into many domains and applications. Perhaps because of this underlying uncertainty, there does not appear to be any meaningful consistency in how courses are taught or programs are conceived, perhaps due to the strong variance in the ways experts value their component knowledge of GIS. Beginning to discern the footing of the GIS landscape with more conceptual and ideological depth would be of great assistance to explore the foundations of how GIS functions academically. These studies indicate that factors which compose the GIS skill set are not consistently taught, required as part of degree programs, or even valued by experts. This disparity is quite interesting, and delineating how these disparities manifest in other ways is an intriguing prospect.

Chapters Two and Three

As geography and GIS degrees change in their requirements and emphases, the viewpoints of the stakeholders who determine how those degrees function will play an important role. Given the three viewpoints of the value of CyberGIS components, a potential projection of degree composition is possible. Right now, I identified few departments that actively require any form of GIS programming in a geography degree. And if traditional spatial analysis or domain-based problem solvers were to serve in those positions of power that determine the structure of degrees, this trend would likely continue. In both of those viewpoints, the underlying computer science and programming components of CyberGIS are not viewed as vital, valuable components of GIS, but instead as minor components that exist to complete certain tasks or assist with more valuable spatial or domain-based analysis. In essence, these views see computer science and programming with specific purposes rather than as essential underpinnings of GIS function and ability. These conceptions might undervalue the programming components of GIS, even as they become more essential, and miss adding programming or computer science relevant components to the degree plan.

Conversely, while programming and computer science are becoming more important in professional GIS, especially as an enabler of advanced spatial analysis, web mapping, big data and other advanced capabilities, the core of GIS is still geographic, and still spatial. Training GIS users only to push buttons without an understanding of the deeper spatial implications of their decision-making is a worrisome concept, and even if students knew both what process to run and the code behind that process, they still might lack an awareness of the spatial perspectives of the system. So the prominence of computer science and programming for factor three viewpoints could swing the GIS requirements too far in favor of those topics, despite their relative scarcity now. As degree requirements evolve, some balance in the administrative and faculty oversight of these degrees must exist between different conceptions of GIS. The traditional spatial analyst may not be a capable GIS user in a development heavy environment, and someone emerging from a domain focused program might not have the versatility to succeed in a broader spatial problem solving environment. The manifestation of expert preferences and blind spots in the composition of degrees will be an important focus moving forward. I outline these differences further in Figure 5-1, but the coupling of faculty focus, department culture, and degree requirements cannot be discounted.

The types I identified in chapter three will also interrelate with how degree programs function in a given department. For example, type one practitioners, the traditional spatial analysts, may emphasize more courses and degree requirements focusing on the theory of GIS rather than explicit practices inside of domains or with computer science and programming connections. Type two practitioners, who value the domain-based problem solving capabilities of GIS, might structure courses and degree requirements to focus on applications, especially applications within their domain specialty, but without a broader situation of GIS in other fields or with connections to GIS and computer science. The third type, the CyberGIS enabled computer scientists, may emphasize programming and computer science coursework, but diminish the core spatial or domain coursework in the overall degree. Each of these types could structure and engender fundamentally different degrees. The growing landscape of GIS degrees may not be preparing students equally, and the influence of viewpoint of GIS on GIS curriculum design is a valuable area of future analysis.

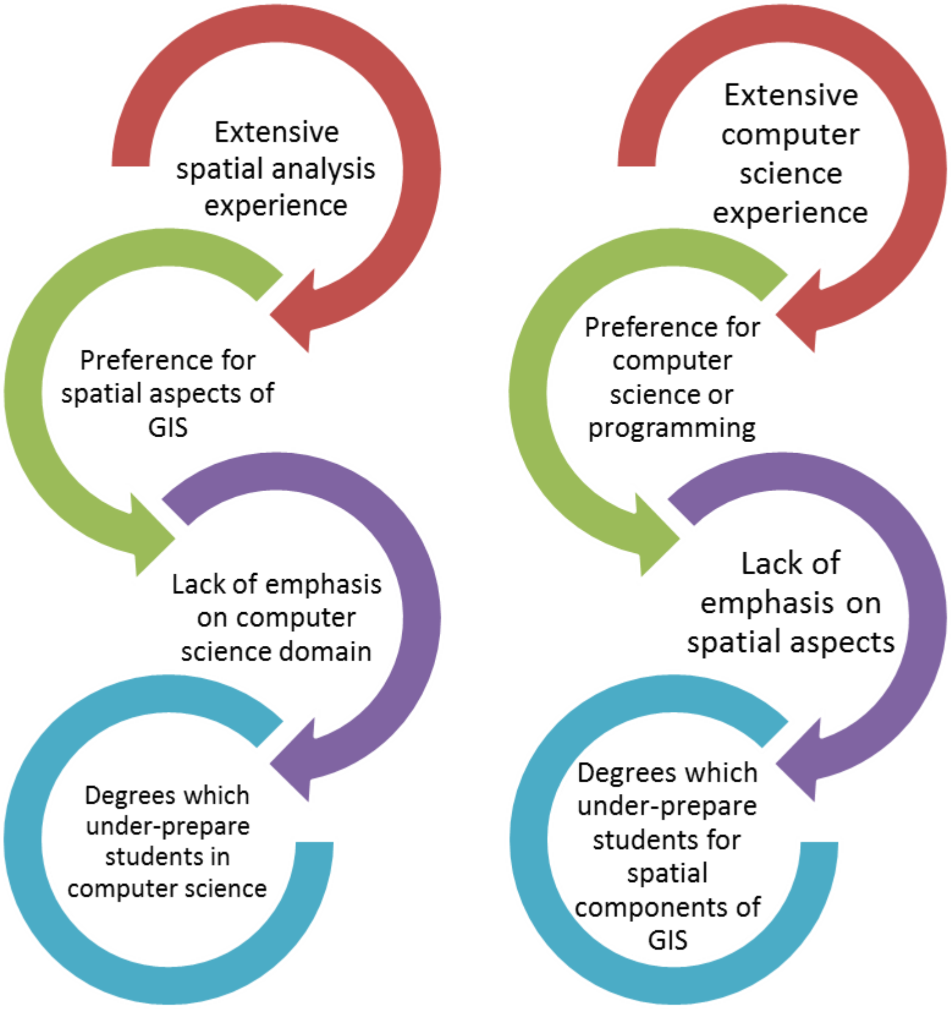

Extensive spatial analysis experience
Preference for spatial aspects of GIS
Lack of emphasis on computer science domain
Degrees which under-prepare students in computer science
Extensive computer science experience
Preference for computer science or programming
Lack of emphasis on spatial aspects
Degrees which under-prepare students for spatial components of GIS

Figure 5-1. The expert blind spot spiral. While faculty, administrators, or other stakeholders with power in determining academic degree structure rely on their expertise to construct degrees, their biases, or blind spots, based on their experiences may result in under-preparation or under-exposure in other key areas of GIS.

Chapters Two and Four

While providing a diversity of courses benefits students in providing multiple options and interests for them to pursue, determining how students build their degrees, and what influences their decision making when faced with degree options like 'pick any 400-level course' is a high-need research area. For students who have the option of picking a GIS programming course, the skills or abilities that course facilitates may vary widely depending on the core languages and skills discussed in that class. As Chapter Four outlines, there are considerable differences in the way GIS programming courses are constructed. For departments where multiple programming courses are offered, or as more courses are developed, all of these possibilities might not fit into degree requirements. In competing for required course slots with foundational GIS, geography, and other specialized courses, students may have to select one programming course with one language and application focus, rather than diversifying their abilities in Python and SQL, for example. And since only one degree in this investigation discovered a multiple computer science and programming course requirement, it seems like any expectation of a rounded, diverse programming course background will fall to students, academic advisors, or faculty for course opt-ins, rather than degree requirements. Thus, there should be some coherent alignment to practice present in these degree programs, informed by efforts like the GIS Body of Knowledge, to build a coherent sequence of courses for students to build their GIS skillset, rather than a convenience of courses for students to pick from with an unknown amount of faculty or advisor guidance.

Collaborating with computer science departments to incorporate more introductory components into liberal arts cores or other common, university wide course requirements may be another method of opening the wide range of topics to students studying geography. Further, regional and local factors must be accounted for, taking into consideration where students matriculate after completing their geography or GIS degrees, and what needs are noted in their future academic or professional work. Figure 5-2 notes some potential lessons to be learned from course constructs applicable to degree program construction, while Figure 5-3 considers the inverse, how the construction of degree programs might influence development of future programming courses. Overall, there is significant overlap in understanding what courses exist and how to integrate them into degree programs. What remains is identifying the key actors in integrating these levels of geography in higher education.

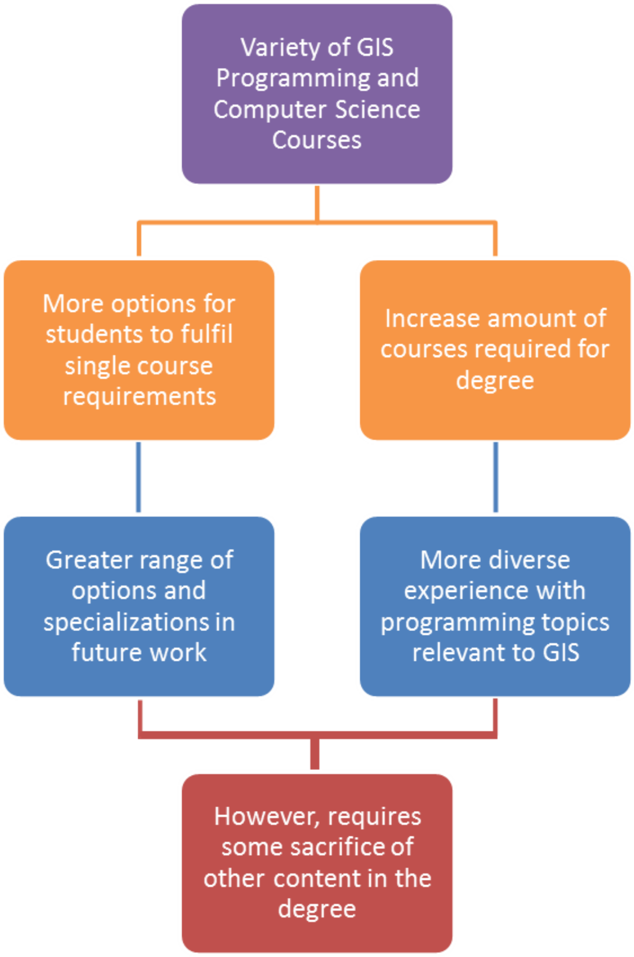

Variety of GIS Programming and Computer Science Courses
More options for students to fulfil single course requirements
Increase amount of courses required for degree
Greater range of options and specializations in future work
More diverse experience with programming topics relevant to GIS
However, requires some sacrifice of other content in the degree

Figure 5-2. Potential influences of programming course variety on GIS degrees. With more programming courses available, degrees may diversify requirements or course options to better prepare students. However, this might come at the cost of some other departmental or university requirements.

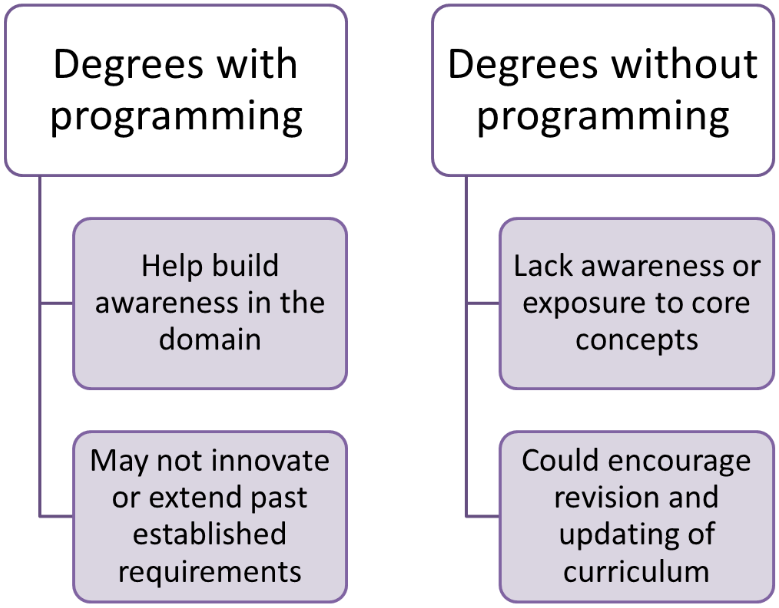
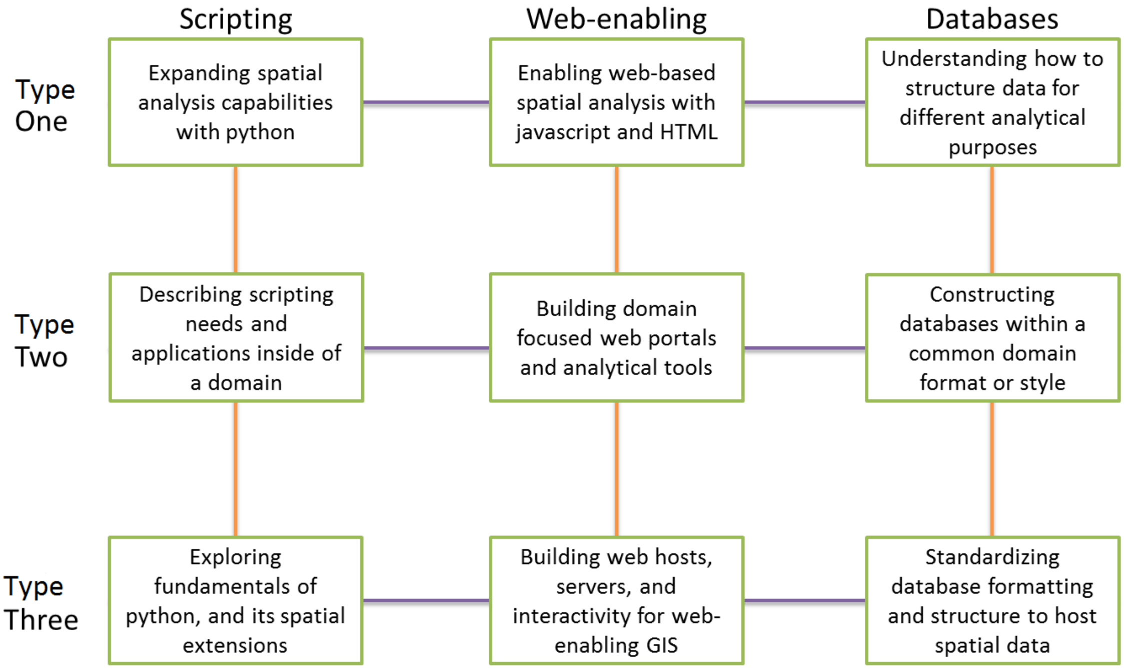

Degrees with programming
Help build awareness in the domain
May not innovate or extend past established requirements
Degrees without programming
Lack awareness or exposure to core concepts
Could encourage revision and updating of curriculum

Figure 5-3. Potential influences of degree structures on GIS programming courses. Some degree forms could encourage the development or restructuring of GIS courses, while others may prove too restrictive.

Chapters Three and Four

Unintentionally, Chapters Three and Four both discuss three primary categorizations of the investigated subjects. For Chapter Three, I outlined three prominent viewpoints as determined by q-method analysis help by experts regarding the components of CyberGIS. In Chapter Four, I discussed three prominent GIS course languages and application types; Python for scripting and extending GIS functions, JavaScript for building web-enabled GIS constructs, and SQL for structuring spatial databases. As I describe in Figure 5-4, considering how each of these viewpoints might manifest in coursework is a worthwhile thought experiment. A course in Python taught by a traditional spatial analyst would look much different than one taught by a CyberGIS enabled computer scientist, and while the outcomes, examples, and course structures would likely be significantly different, both would be relevant if delivered in an appropriate context. These different approaches may seem to preclude any one-size-fits-all approach to building curriculum and focusing instruction, but further exploration of how this synthesis functions in the classroom will help identify approaches applicable across domains and throughout diversity applications of GIS instruction.



	Scripting	Web-enabling	Databases
Type One	Expanding spatial analysis capabilities with python	Enabling web-based spatial analysis with javascript and HTML	Understanding how to structure data for different analytical purposes
Type Two	Describing scripting needs and applications inside of a domain	Building domain focused web portals and analytical tools	Constructing databases within a common domain format or style
Type Three	Exploring fundamentals of python, and its spatial extensions	Building web hosts, servers, and interactivity for web-enabling GIS	Standardizing database formatting and structure to host spatial data

Figure 5-4. Manifestations of viewpoints in course designs. As different views of the utility of computer science and programming components manifest, different types or focuses of coursework can develop despite any similarity in the computer science or programming topic.

As the need (and demand) for courses in GIS programming and computer science continues to rise, it will be necessary to consider the background required of instructors. The specialized nature of the applications in this area may lead to highly specialized courses, with great focus on specific applications, but less integrations with other areas of programming, computer science, GIS, or geography. The framework and biases that an instructor brings to their coursework will influence the course's design, and how those choices influence content will have meaningful impacts on how students learn content, and what they take away from the course overall. Further, if a course exists in some form but a new instructor with a new perspective or different values takes over the course, that course will transition to different emphases, perhaps changing the fundamental nature of instruction. A further emphasis on classroom instruction can help reveal patterns and divergences among instructors with different backgrounds and GIS values. Understanding how viewpoints shape content remains a valuable area of synthesis considering these components.

Agenda for Future Research

Within these articles, I propose additional research to further understand and establish meaningful, research-based evidence on how curriculum and instruction in GIS, programming, and computer science can be understood on the CyberGIS frontier.

Proposed Research Areas

More research is necessary to further understand and assess the teaching and learning of programming and computer science in geography. CyberGIS functions as useful entity as the frontier of the integrations of GIS, computer science, programming, and geography. Organizing GIS education research around its synthesis should prove beneficial as well. As outlined by Baker et al. (2015), geospatial learning knowledge remains sparse, but the directions posed by this and other roadmaps for GIS education research are meaningful structures when considering the components of CyberGIS as well. Considering the need for structured, replicable, interdisciplinary, and coordinated approaches concerning geospatial learning, and to contribute meaningful research to advance the next wave of GIS education research, I propose the following areas of future work.

One track of research should be conducted within classrooms teaching computer science, programming, and GIS in concert. An evidence-based understanding is vital to determining what advantages or disadvantages might exist for students who take a computer science and programming course outside of geography compared to those who take it within their home department. Through classroom observations, participant observations, and other methods, determining how these courses function using the rich methodologies available can help build constructions of current practice to connect to existing theoretical frameworks. Considering these differences in instruction, would geography students be better prepared with the general concepts and experience in an introductory computer science and programming course, or with the application focused instruction in a GIS programming course? What does training in programming do for GIS students that makes them think more efficiently or more spatially? These questions can lead to connections with expert approaches or progressions, and how the multi-faceted expertise necessary in GIS practice develops. Any research and evidence-based documentation of these components would be incredibly valuable to GIS instruction. Additional classroom-based research concerning student learning through different instructional methods would bring important evidence into addressing the best practices in integrating computer science, programming, and GIS instruction.

Further research tracks along this training line should consider computer science and programming knowledge, skills, and practices in GIS and geography careers, and where academic preparation matches or falls short of professional expectations. This can begin with additional replications and confirmations of the types outlined in Chapter Three, through additional q-method research on the importance of computer science and programming in GIS right now. A careful analysis of the practices in GIS programming and computer science, informs their function in GIS applications. Understanding this practice aspect provides insight into the approaches to 'doing' GIS that can then be reflected onto how GIS is taught or organized academically. From there, the role of academic advisors or faculty in guiding students through their degree plans, selecting courses, and building their academic skills would be a valuable area of focus. Tying expert viewpoints to functional curriculum and instruction approaches in GIS would bridge these component domains in a meaningfully holistic fashion.

In concert, future analysis should continue to investigate the viewpoints of other stakeholders concerning the components of CyberGIS. Replicating this study with further diverse interests in GIS, CyberGIS, geography, and other component domains will continue to gather viewpoints and identify what other conceptions of geography may be present in the current framework. Further, future studies should use the adapted q-set developed and shown in Appendix B. And while q-method is a powerful methodology for building statistically based understandings of subjectivity, other promising methods for conducting research should be considered as well. How CyberGIS components are integrated into industry and their broad benefits to students as they engage with the professional workforce is also of interest, and considering the needs of professional GIS, as opposed or aligned to academic GIS, could provide a fruitful area for collaboration and content.

The growing literature in CyberGIS, as well as studies in profession GIS skillsets, indicates there is a distinct need for computer science and programming in GIS. The affordances of advanced computer infrastructure, computing capabilities, and programming knowledge are vital to extending and expanding the reach, influence, and capability of spatial analysis in GIS. While Chapter Four outlines three distinct types for content and information in this realm, it raises more questions about how these courses present information and are structured within the general GIS knowledge base. Since no course sampled explores components of all three of these courses (scripting, web-enabling, databases), students might remain unprepared in two key areas of GIS programming when taking a course in one. Therefore, I recommend investigations into what components of programming and computer science are most vital for the GIS user, whether in academics, professional employment, or otherwise, to build on this syllabus understanding to propose, design, or reconstruct academic training moving forward.

Another disparity of note worth further investigation is the way syllabi are constructed, especially in terms of learning outcomes, learning progressions, and scaffolding of knowledge in the GIS and computer science realms. Some syllabi in this sample include no learning outcomes, and others do not describe course materials or content outside of general readings selected by the instructor. Research on individual courses, including case studies of how GIS programming courses function, would provide meaningful evidence regarding what objectives are appropriate for novices in computer science with an extensive GIS background.

Finally, work needs to begin on identifying instructional materials and texts usable in these contexts. Zandbergen's Python text is relatively widely used, and instructors that use it as an instructional material tend to structure their syllabus with its chapter organization as a guide. Determining what other resources exist for complementary topics, or whether additional topics in computer science and programming are relevant areas for further investigation. Considering that 38 texts exist across 40 syllabi, and that some syllabi outlined no instructional materials in their construction, future work in understanding existing instructional materials would be very beneficial in understanding what common topics are shared across courses and within course materials. Just as previous analytical efforts studies the spatial nature of GIS tests (Jo et al. 2012), a similar effort on the computer science and programming nature of texts would be an important step for consolidating instructional resources, and outlining potential curriculum goals for GIS courses of this nature. Further guidance to text relevance might be obtained by asking professional GIS users what text, manuals, or guides shape their post-academic learning of GIS, computer science, and programming.

Current Complementary Work

My current projects are working to understand how computer science and programming skills are represented in professional GIS. I am working with other researchers to determine what hard and soft skills are present in GIS job ads, as well as what educational competencies, experience, and other requirements these jobs have, to understand what a GIS job is composed of, and whether the current educational framework allows GIS or geography students to be competitive for those jobs. One worry is that GIS jobs are becoming more and more jobs for the computer scientist, with employment emphasis focused on programming, knowledge of computer ontologies, and software skills rather than spatial analysis or understanding. Two separate efforts seek to explore this situation and follow the recommendations above. First, a collection of GIS job postings is being prepared for analysis to determine any trends, patterns, or dominant constructs in how GIS jobs are changing and who might be qualified for them. Second, interviews and surveys with current GIS professionals in specific industries are being developed to gather additional viewpoints on what GIS experts do and what they look for in expanding their skills and abilities. These components will provide a fruitful base of research for my future research agenda.

Closing

GIS can be conceptualized in many ways. As a tool, the analytical capabilities of GIS allow many types and forms of spatial analysis that extend and expedite our ability to understand how spatial processes function; as a system, GIS allows storage and management of spatial data to help structure these analyses; and as a science, GIS changes the way users consider spatial questions, and enables new ways of framing inquiries to understand spatial processes. All of these approaches have rich supporting literature and discussion of best practices considering education through these lenses. However, as they all continue to be redeveloped and reconceptualized with continuous advances in technology and methodologies, how they function remains fluid. The growing literature and research in CyberGIS seems to point the way for the technological future of GIS, yet the implications of that path for education are only beginning to be discovered.

CyberGIS seems to promise a massive jump ahead in the way spatial data is conceptualized. By utilizing faster processors, handling and processing more data, and allowing greater use and access to spatial analysis, the advanced computing infrastructures on which CyberGIS relies support an amazing reimagination of the function of spatial data. Web GIS today offers capabilities in map making, data handling, and analysis that required expensive desktop installations merely ten years ago, and were impossible outside of exclusive computer and data centers 25 years ago. With the first dedicated GIS supercomputer now functional, further advances in fusing advanced computing and cyberinfrastructure with GIS will continue to emerge as the cutting edge of GIS practice and filter into the activities of the daily GIS user.

It is important to reemphasize that CyberGIS is systems focused. Indeed, the 'GIS' of CyberGIS is Geographic Information Systems. While the technical and computing aspects of GIS are certainly fast-evolving, sound scientific decision-making remains essential to achieving meaningful outputs with GIS. For example, the ability to access millions of geocoded tweets does not guarantee they will be analyzed or displayed in a scientifically or methodologically valid way. As computer science and programming systems knowledge, skills, and practices emerge from CyberGIS, attention should be paid to maintaining and developing other components of GIS knowledge as well. If CyberGIS is the future of how GIS functions, it nonetheless needs to retain the lessons learned from previous discussions on GIS as a science and not simply a tool.

The first Geographic Information System sought to solve a simple problem: how to store, measure, and analyze data (Tomlinson 1967). This mission statement still rings true today. GIS remains a growing and vibrant area of study. As GIS expands across industries, becomes easier to use, and strengthens its position as a fundamental information technology, the spatial capabilities it affords contribute to this growth. Teaching and learning in GIS will continue to adapt as new software packages dominate the use of GIS professionally, new computing methods are required for functional use of GIS, and new approaches and discoveries are incorporated into GIS practice. Instead of blindly adapting to these changes as each new technology or approach emerges, building evidence of how GIS instruction is adapting, and how new approaches or technologies are integrated, allows us a more meaningful view of the knowledge, skills, and practices in the field as a whole.

This dissertation focuses on CyberGIS as a nexus for reconceptualizing GIS with added emphasis on computer science and programming components. Regardless of whether CyberGIS is a new form of GIS or the natural evolution of GIS into modern infrastructure, by outlining the viewpoints of CyberGIS components and discovering what topics exist in courses and programs, the information presented here serves as a guidepost in understanding how GIS functions educationally and academically. The field will be different in the future, but this work builds a basic level of understanding of what GIS is now: increasingly integrated with computer science, reliant on connections to infrastructure with programming, and fundamentally spatial.

REFERENCES

Academic Analytics. 2015. Online at: http://www.academicanalytics.com/ (last accessed 13 August 2015).

Ackerman, P. L. 2014. Nonsense, common sense, and science of expert performance: Talent and individual differences. *Intelligence* 45:6-17.

Adams, N. E. 2015. Bloom's taxonomy of cognitive learning objectives. *Journal of the Medical Library Association* 103:152-153.

Agrawal, S., D. K. Meena, and R. D. Gupta. 2015. Comprehensive review of GIS and Web GIS. *Discovery* 41:130-135.

Babin, L-M, A. Tricot, and C. Mariné. 2009. Seeking and providing assistance while learning to use information systems. *Computers & Education* 53:1029-1039.

Baker, T. R., S. Battersby, S. W. Bednarz, A. M. Bodzin, B. Kolvoord, S. Moore, D. Sinton, and D. Uttal. 2015. A research agenda for geospatial technologies and learning. *Journal of Geography* 114:118-130.

Barker, L. J. and K. Garvin-Doxas. 2004. Making visible the behaviors that influence the learning environment: A qualitative exploration of computer science classrooms. *Computer Science Education* 14:119-145.

Bearman, N., P. Munday, and D. McAvoy. 2015. Teaching GIS outside of geography: A case study in the School of International Development, University of East Anglia. *Journal of Geography in Higher Education* 39:237-244.

Berland, M., T. Martin, T. Benton, C. P. Smith, and D. Davis. 2013. Using learning analytics to understand the learning pathways of novice programmers. *Journal of the Learning Sciences* 22:564-599.

Bishop, M. P. 2009. International multidisciplinary research and education: A mountain geography perspective. *Journal of Geography* 108: 112-120.

Blaschke, T. and H. Merschdorf. 2014. Geographic information science as a multidisciplinary and multiparadigmatic field. *Cartography and Geographic Information Science* 41:196-213.

Bondi, A. B. 2000. Characteristics of scalability and their impact on performance. *Proceedings of the 2nd International Workshop on Software and Performance* 195-203.

Boshuizen, H. P. A. 2009. "Teaching for expertise: Problem-based methods in medicine and other professional domains," in *Development of Professional Expertise: Toward Measurement of Expert Performance and Design of Optimal Learning Environments,* edited by K. A. Ericsson, 379-404. Cambridge: Cambridge University Press.

Boss, K. and E. Drabinski. 2013. Evidence-based instruction integration: A syllabus analysis project. *Reference Services Review* 42:263-276.

Bransford, J. D. and D. L. Schwartz. 2009. "It takes expertise to make expertise: Some thoughts about why and how and reflections on the themes in chapters 15-18," in *Development of Professional Expertise: Toward Measurement of Expert Performance and Design of Optimal Learning Environments*, edited by K. A. Ericsson, 432-448. Cambridge: Cambridge University Press.

Breßler, J. 2012. "The development of expertise in education and training management" in *Knowledge and Learning: Global Empowerment; Proceedings of the Management, Knowledge and Learning International Conference 2012*, 403-412. Celje, Slovenia: International School for Social and Business Studies.

Brooks, C. 2010. Why geography teachers' subject expertise matters. *Geography* 95:143-148.

Brovelli, M. A., M. Minghini, and G. Zamboni. 2015. Public participation in GIS via mobile applications. *ISPRS Journal of Photogrammetry and Remote Sensing* In Press.

Brown, R. M., R. J. Zatorre, and V. B. Penhune. 2015. Expert music performance: Cognitive, neural, and developmental bases. *Progress in Brain Research* In press.

Burgman, M. A., M. McBride, R. Ashton, A. Speirs-Bridge, L. Flander, B. Wintle, F. Fidler, L. Rumpff, and C. Twardy. 2011. Expert status and performance. *PLoS One* 6, Published online July 29, 2011.

Carnegie Foundation for the Advancement of Teaching. 2011. *The Carnegie Classification of Institutions of Higher Education*, *2010 edition*. Menlo Park, CA: Carnegie Foundation for the Advancement of Teaching.

Carter, L. 2006. Why students with an apparent aptitude for computer science don't choose to major in computer science. *ACM SIGCSE Bulletin* 38:27-31.

Davies, S., J. A. Polack-Wahl, and K. Anewalt. 2011. A snapshot of current practices in teaching the introductory programming sequence. *SIGCSE* 2011.

Day, J. 2002. What is an expert? *Radiography* 8:63-70.

Devolder, A., J. van Braak, and J. Tondeur. 2012. Supporting self-regulated learning in computer-based learning environments: Systematic review of effects of scaffolding in the domain of science education. *Journal of Computer Assisted Learning* 28:557-573.

DiBiase, D., T. Corbin, T. Fox, J. Francia, K. Green, J. Jackson, G. Jeffress, B. Jones, B. Jones (sic), J. Mennis, K. Schuckman, C. Smith, and J. Van Sickle. 2010. The new geospatial technology competency model: Bringing workforce needs into focus. *URISA Journal* 22:55-72.

Downs, R. M. 1994. Being and becoming a geographer: An agenda for geography education. *Annals of the Association of American Geographers* 84:175-191.

Downs, R. M. 2014. Time and practice: Learning to become a geographer. *Journal of Geography* 113:189-197.

Dramowicz, K., J. F. Wightman, and J. S. Crant. 1993. Addressing GIS personnel requirements: A model for education and training. *Computers, Environment and Urban Systems* 17:49-59.

Dreyfus, H. L., & Dreyfus, S. E. (1986). *Mind over machine: The power of human intuition and expertise in the era of the computer*. New York: The Free Press.

Dror, I. E. and S. A. Cole. 2010. The vision in "blind" justice: Expert perception, judgment, and visual cognition in forensic pattern recognition. *Psychonomic Bulletin & Review* 17:161-167.

Du, J., S. Jing and J. Liu. 2012. Creating shared design thinking process for collaborative design. *Journal of Network and Computer Applications* 35:111-120.

Duckham, M. 2015. GI expertise. *Transactions in GIS* 19:499-515.

Ďuračiová, R. 2013. Querying uncertain data in geospatial object-relational databases using SQL and fuzzy sets. *Slovak Journal of Civil Engineering* 21:1-12.

Eberly, M. B., S. E. Newton, and R. A. Wiggins. 2001. The syllabus as a tool for student-centered learning. *Journal of General Education* 50:56-74.

Eisenhart, M., H. Borko, R. Underhill, C. Brown, D. Jones, and P. Agard. 1993. Conceptual knowledge falls through the cracks: Complexities of learning to teach mathematics for understanding. *Journal for Research in Mathematics Education* 24:8-40.

Ellul, C. 2015. Geography and geographical information science: Interdisciplinary integrators. *Journal of Geography in Higher Education* 39:191-194.

Ericsson, K. A. 1998. The scientific study of expert levels of performance: General implications for optimal learning and creativity. *High Ability Studies* 9:75-100.

Ericsson, K. A. 2005. Recent advances in expertise research: A commentary on the contributions to the special issue. *Applied Cognitive Psychology* 19:233-241.

Ericsson, K. A. 2008. Deliberate practice and acquisition of expert performance: A general overview. *Academic Emergency Medicine* 15:988-994.

Ericsson, K. A. and A. M. Williams. 2007. Capturing naturally occurring superior performance in the laboratory: Translational research on expert performance. *Journal of Experimental Psychology: Applied* 13:115-123.

Ericsson, K. A., and J. Smith. 1991. *Toward a general theory of expertise: Prospects and limits*. Cambridge: Cambridge University Press.

Ericsson, K. A., R. S. Perez, D. W. Eccles, L. Lang, E. L. Baker, J. D. Bransford, K. VanLehn, and P. Ward. 2009. "The measurement and development of professional performance: An introduction to the topic and a background to the design and origin of this book" in *Development of Professional Expertise: Toward Measurement of Expert Performance and Design of Optimal Learning Environments*, edited by K. A. Ericsson, 1-24. Cambridge: Cambridge University Press.

Ericsson, K. A., R. Th. Krampe, and C. Tesch-Römer. 1993. The role of deliberate practice in the acquisition of expert performance. *Psychological Review* 100:363-406.

Ertmer, P. A. and T. J. Newby. 1996. The expert learner: Strategic, self-regulated, and reflective. *Instructional Science* 24:1-24.

ESRI. 2015. *What is ArcPy?* Available online at: http://pro.arcgis.com/en/pro-app/arcpy/get-started/what-is-arcpy-.htm

Etherington, T. R. 2015. Teaching introductory GIS programming to geographers using an open source Python approach. *Journal of Geography in Higher Education*, published online 19 October, 2015.

Farrington-Darby, T., and J. R. Wilson. 2006. The nature of expertise: A review. *Applied Ergonomics* 37:17-32.

Favier, T. T. and J. A. van der Schee. 2012. Exploring the characteristics of an optimal design for inquiry-based geography education with Geographic Information Science. *Computers & Education* 58:666-677.

Fee, S. B. and A. M. Holland-Minkley. 2010. Teaching computer science through problems, not solutions. *Computer Science Education* 20:129-144.

Foley, M. and A. Hart. 1992. "Expert-novice differences and knowledge elicitation" in *The Psychology of Expertise: Cognitive Research and Empirical AI*, edited by R. R. Hoffman, 233-244. New York: Spring-Verlag.

Forte, A. and M. Guzdial. 2005. Motivation and nonmajors in computer science: Identifying discrete audiences for introductory courses. *IEEE Transactions on Education* 48:248-253.

Foster, I. 1995. *Designing and Building Parallel Programs.* Argonne, IL: Argonne National Laboratory.

Foster, I., C. Kesselman, and S. Tuecke. 2001. The anatomy of the grid: Enabling scalable virtual organizations. *The International Journal of High Performance Computing Applications* 15:200-222.

Franklin, D., C. Hill, H. Dwyer, A. Iveland, A. Killian, and D. Harlow. 2015. Getting started in teaching and researching computer science in the elementary classroom. *SIGCSE '15* Published online May 2015.

Gasparinatou, A. and M. Grigoriadou. 2011. Supporting students' learning in the domain of computer science. *Computer Science Education* 21:1-28.

Gaudet, C.H., H. M. Annulis, and J. C. Carr. 2003. Building the geospatial workforce. *URISA Journal* 15:21-30.

Goldberg, D., M. Olivares, Z. Li, and A. G. Klein. 2014. Maps & GIS libraries in the era of big data and cloud computing. *Journal of Map and Geography Libraries* 10:100-122.

Goodchild, M. 1998. GIS and geography: Elements of a debate. *Yearbook of the Association of Pacific Coast Geographers* 60:150-157.

Goodchild, M. 2007. Citizens as voluntary sensors: Spatial data infrastructure in the world of Web 2.0. *International Journal of Spatial Data Structures Research* 2:24-32.

Goodchild, M. 2011. Scale in GIS: An overview. *Geomorphology* 130:5-9.

Goodchild, M. F. 1992. Geographical information science. *International Journal of Geographical Information Systems* 6:31-45.

Goodchild, M. F. 2009. Geographic information systems and science: Today and tomorrow. *Annals of GIS* 15:3-9.

Goodchild, M. F. 2010. Twenty years of progress: GIScience in 2010. *Journal of Spatial Information Science* 1:3-20.

Guo, P. 2014. Python is now the most popular introductory teaching language at top US universities. *Communications of the ACM,* 7 July 2014. Online at: http://cacm.acm.org/blogs/blog-cacm/176450-python-is-now-the-most-popular-introductory-teaching-language-at-top-us-universities/fulltext (last accessed 13 August 2015).

Ha, T. S. 2015. Learning stories from IT workers – development of professional expertise. *Studies in Continuing Education* 37:79-98.

Hambrick, D. Z. and E. J. Meinz. 2011. Limits on the predictive power of domain-specific experience and knowledge in skilled performance. *Current Directions in Psychological Science* 20:275-279.

Harden, R. M. 2002. Developments in outcome-based education. *Medical Teacher* 24:117-120.

Harvey, F. 2013. A new age of discovery: The post-GIS era. *GI_Forum* 2013:272-281.

Hélie, S. and R. Sun. 2010. Incubation, insight, and creative problem solving: A unified theory and a connectionist model. *Psychological Review* 117:994-1024.

Hoffman, R. R., N. R. Shadbolt, A. M. Burton, and G. Klein. 1995. Eliciting knowledge from experts: A methodological analysis. *Organizational Behavior and Human Decision Process* 62:129-158.

Hong, J. E. 2015. Identifying skill requirements for GIS positions: A content analysis of job advertisements. *Journal of Geography* Published online 14 Oct 2015.

Houlton, H. R. 2015. Thinking critically about the student-to-professional transition: Alignments of skills qualifications of geoscience master's students with current workforce employment. *The Leading Edge* October 2015:1220-1229.

Huff, C. 2014. From meaning well to doing well: Ethical expertise in the GIS domain. *Journal of Geography in Higher Education* 38:455-470.

Hung, S-Y. 2001. Expert versus novice use of the executive support systems: An empirical study. *Proceedings of the 34th Hawaii International Conference on System Sciences – 2001* 1-9.

Huynh, N. T. and B. Sharpe. 2013. An assessment instrument to measure geospatial thinking expertise. *Journal of Geography* 112:3-17.

Irwin, R. 2002. Characterizing the core: What catalog descriptions of mandatory courses reveal about LIS schools and librarianship. *Journal of Education for Library and Information Science* 43:175-184.

Jacobs, J. C. G., A. M. M. Muijtens, S. J. Van Luijk, C. P. M. Can Der Vleuten, G. Croiset, and F. Scheele. 2015. Impact of institute and person variables on teachers' conceptions of learning and teaching. *Medical Teacher* 37:738-746.

Jazayeri, M. A., S. H. L. Liang, and C-Y Huang. 2015. Implementation and evaluation of four interoperable open standards for the internet of things. *Sensors* 15:24343-24373.

Jewell, A. 2013. Supporting the novice nurse to fly: A literature review. *Nurse Education in Practice* 13:323-327.

Jhummarwala, A., M. B. Potdar, and P. Chauhan. 2014. Parallel and distributed GIS for processing geo-data: An overview. *International Journal of Computer Applications* 106:9-16.

Jo, I., A. Klein, R. S. Bednarz, and S. W. Bednarz. 2012. An exploration of spatial thinking in introductory GIS courses. In *Teaching Geographic Information Science and Technology in Higher Education*, 211-229. Oxford: Wiley and Sons.

Johnson, J. 2010. What GIS technicians do: A synthesis of DACUM job analyses. *Journal of the Urban and Regional Information Systems Association* 22:31-40.

Kay, J., M. Barg, A. Fekete, T. Greening, O. Hollands, J. H. Kingston, and K. Crawford. 2000. Problem-based learning for foundation computer science courses. Computer Science Education 10:109-128.

Kemp, K. K. and A. U. Frank. 1996. Toward consensus on a European GIS curriculum: The international post-graduate course on GIS. *International Journal of Geographical Information Systems* 10:477-497.

Kerski, J. J., A. Demirci, and A. J. Milson. 2013. The global landscape of GIS in secondary education. *Journal of Geography* 112:232-247.

Kim, I-H. and M-H. Tsou. 2013. Enabling Digital Earth simulation models using cloud computing or grid computing – two approaches supporting high-performance GIS simulation frameworks. *International Journal of Digital Earth* 6:383-403.

Kim, M. and R. Bednarz. 2013. Effects of a GIS course on self-assessment of spatial habits of mind (SHOM). *Journal of Geography* 112:165-177.

Kinnunen, P. and B. Simon. 2012. My program is ok - am I? Computing freshmen's experiences of doing programming assignments. *Computer Science Education* 22:1-28.

Kong, N., T. Zhang, and I. Stonebraker. 2015. Evaluation of web GIS functionality in academic libraries. *Applied Geography* 60:288-293.

Kopteva, I. A., D. Arkowski, and E. L. Craft. 2015. Tiered internship model for undergraduate students in geospatial science and technology. *Community College Journal of Research and Practice* 39:34-43.

Krathwohl, D. R. 2002. A revision of Bloom's taxonomy: An overview. *Theory Into Practice* 41:212-264.

Kung, M., S. C. Yang, and Y. Zhang. 2006. He changing information systems (IS) curriculum: A survey of undergraduate programs in the United States. *Journal of Education for Business* 81:291-300.

Lacireno-Paquet, N., C. Morgan, and D. Mello. 2014. *How states use student learning objectives in teacher evaluation systems: A review of state websites*. Regional Educational Laboratory 2014-013.

Lay, J-G., Y-L. Chi, Y-S. Hsieh, and Y-W. Chen. 2013. What influences geography teachers' usage of geographic information systems? A structural equation analysis. *Computers & Education* 62:191-195.

Lee, J. and R. Bednarz. 2009. Effect of GIS learning on spatial thinking. *Journal of Geography in Higher Education* 33:183-198.

Lee, J. and S. Catling. 2015. Some perceptions of English geography textbook authors on writing textbooks. *International Research on Geographical and Environmental Education.* Published online 02 November 2015.

Liu, L., W. Li, H. Qiu, and C. Sun. 2012. Research and realization of GIS network education system. *IERI Procedia* 2:276-281.

Lukinbeal, C and J. J. Monk. 2015. Master's in geographic information systems programs in the United States: Professional education in GIS and geography. *The Professional Geographer* 67:482-489.

Ma, L., J. Ferguson, M. Roper, and M. Wood. 2011. Investigating and improving the models of programming concepts held by novice programmers. *Computer Science Education* 21:57-80.

Madsen, L. M. and C. Rump. 2013. Considerations of how to study learning processes when students use GIS as an instrument for developing spatial thinking skills. *Journal of Geography in Higher Education* 36:97-116.

Martí. C., J. Feliu, and D. Varga. 2014. Geographic information technology and innovative teaching: Keys to geography degree curriculum reform. *Journal of Geography* 113:118.128.

McCallum, A. K. 2002. *MALLET: A Machine Learning for Language Toolkit*. Available online at: http://mallet.cs.umass.edu.

McKeithen, K. B., J. S. Reitman, H. R. Rueter, and S. C. Hirtle. 1981. Knowledge organization and skill differences in computer programmers. *Cognitive Psychology* 13:307-325.

Merali, Z. 2010. …ERROR …why scientific programming does not compute. *Nature* 467:775-777.

Merchant, Z., E. T. Goetz, L. Cifuentes, W. Keeney-Kennicutt, and T. J. Davis. 2014. Effectiveness of virtual reality-based instruction on students' learning outcomes in K-12 and higher education: A meta-analysis. *Computers and Education* 70:29-40.

Messer, W. S., R. A. Griggs, and S. L. Jackson. 1999. A national survey of undergraduate psychology degree options and major requirements. *Teaching of Psychology* 26:164-171.

Michaelson, G. 2015. Teaching programming with computational and informational thinking. *Journal of Pedagogic Development* 5:51-66.

Mirzoev, R., A. Moore, B. Pryzbysz, M. Taylor, and J. Centeno. GIS as a job growth area for IT professionals. 2015. *World of Computer Science and Information Technology Journal* 5:98-111.

Monk, J. J. and K. E. Foote. 2015. Directions and challenges of master's programs in geography in the United States. *The Professional Geographer* 67:472-481.

Muller, C. L. and C. Kidd. 2014. Debugging geographers: Teaching programming to non-computer scientists. *Journal of Geography in Higher Education* 38:175-192.

Nathan, M. J. and A. Petrosino. 2003. Expert blind spot among preservice teachers. *American Educational Research Journal* 40:905-928.

National Center for Supercomputing Applications. 2014. ROGER System Information Wiki. University of Illinois. Available online at: https://wiki.ncsa.illinois.edu/display/ROGER/ROGER+System+Information

National Research Council. 2000. *How People Learn: Expanded Edition* Washington, D.C.: National Academy Press.

National Research Council. 2006. *Learning to Think Spatially*. Washington, D.C.: National Academies Press.

National Research Council. 2010. Doctoral Program Rankings. Online at: http://chronicle.com/article/NRC-Rankings-Overview-/124734/ (last accessed 13 August 2015).

Nuutila, E., S. Törmä, and L. Malmi. 2005. PBL and computer programming – The seven steps method with adaptations. *Computer Science Education* 15:123-142.

Nyerges, T. L., M. J. Roderick, and M. Avraam. 2013. CyberGIS design considerations for structured participation in collaborative problem solving. *International Journal of Geographical Information Science* 27:2146-2159.

O'Keefe, R. D., J. R. Lopez, J. Xu., and R. K. Lall. 2015. Teaching and learning objectives: The first step in assessment programs. *Journal of Learning in Higher Education* 11:57-66.

Papastergiou, M. 2009. Digital game-based learning in high school computer science education: Impact on educational effectiveness and student motivation. *Computers & Education* 52:1-12.

Pickles, J. 1997. Tool or science? GIS, technoscience, and the theoretical turn. *Annals of the Association of American Geographers* 87:363-372.

Pleggenkuhle-Miles, E., T. A. Khoury, D. L. Deeds, and L. Markoczy. 2013. Exploring cognitive bias: Expert ratings of business schools. *Management Decision* 51:1905-1927.

Postigo, Y. and J. I. Pozo. 2004. On the road to graphicacy: The learning of graphical representation systems. *Educational Psychology* 24:623-644.

Read, J. M. 2010. Teaching introductory geographic information systems through problem-based learning and public scholarship. *Journal of Geography in Higher Education* 34:379-399.

Ribeiro, R. 2013. Remarks on explicit knowledge and expertise acquisition. *Phenomenology and the Cognitive Sciences* 12:431-435.

Rickles, P. and C. Ellul. 2014. A preliminary investigation into the challenges of learning GIS in interdisciplinary research. *Journal of Geography in Higher Education*. Published online Sep 19, 2014.

Robins, A., J. Rountree, and N. Rountree. 2003. Learning and teaching programming: A review and discussion. *Computer Science Education* 13:137-172.

Sarkar, A. and J. Pick. 2014. A model and case analysis of geographical information systems curricula in management schools. *Twentieth Americas Conference on Information Systems* 2014.

Savery, J. R. 2006. Overview of problem-based learning: Definitions and distinctions. *Interdisciplinary Journal of Problem-Based Learning* 1:9-20.

Schmidt, H. G. and H. P. A. Boshuizen. 1993. On acquiring expertise in medicine. *Educational Psychology Review* 5:205-221.

Schulze, U., D. Kanwischer, and C. Reudenbach. 2013. Essential competences for GIS learning in higher education: A synthesis of international curricular documents in the GIS&T domain. *Journal of Geography in Higher Education* 37: 257-275.

Seiter, L. and B. Foreman. 2013. Modeling the learning progressions of computational thinking of primary grade students. *ICER '13:*59-66.

Şeremet, M. and B. Chalkley. 2014. Student perspectives on the teaching of geographical information systems (GIS) in geography degrees. *Journal of Geography in Higher Education* Published online Oct 1, 2014.

Shanteau, J. 1988. Psychological characteristics and strategies of expert decision makers. *Acta Psychologica* 68:203-215.

Shanteau, J. and T. R. Stewart. 1992. Why study expert decision making? Some historical perspectives and comments. *Organizational Behavior and Human Decision Processes* 53:95-106.

Shermer, M. 2013. The 10,000-hour rule debunked. *Skeptic Magazine* 18:57-58.

Simon, H. A. and W. G. Chase. Skill in chess. 1973. *American Scientist* 61:394-403.

Sinton, D. S. 2009. Roles for GIS within higher education. *Journal of Geography in Higher Education* 33:s7-S16.

Smith, E. A. 2001. The role of tacit and explicit knowledge in the workplace. *Journal of Knowledge Management* 5:311-321.

Solem, M., I. Cheung, and M. B. Schlemper. 2008. Skills in professional geography: An assessment of workforce needs and expectations. *The Professional Geographer* 60:356-373.

Sternberg, R. J. and Frensch, P. A. 1992. On being an expert: A cost-benefit analysis. In *The psychology of expertise*, 191-203. New York: Springer.

Sui, D.Z. 1995. A pedagogic framework to link GIS to the intellectual core of geography. *Journal of Geography* 94:578-591.

Sui, D. and M. Goodchild. 2011. The convergence of GIS and social media: Challenges for GIScience. *International Journal of Geographical Information Science* 25:1737-1748.

Tarboton, D. 2015. TauDEM Version 5. Available online at: http://hydrology.usu.edu/taudem/taudem5/documentation.html

Tate, N. J. and D. J. Unwin. 2009. Teaching GIS&T. *Journal of Geography in Higher Education* 33:S1-S6.

Theo, L. 2011. Simplifying central place theory using GIS and GPS. *Journal of Geography* 110:16-26.

Tomlinson, R. 1967. *Data for Decision*. Available online at: http://video.esri.com/watch/128/data-for-decision_comma_-1967-short-version

Tucker, B. 2012. The flipped classroom. *Education Next*, 12:82-84.

Tyler, L. K. and H. E. Moss. 2001. Towards a distributed account of conceptual knowledge. *Trends in Cognitive Science* 5:244-252.

Van der Kleij, F. M., R. C. W. Feskens, and T. J. H. M. Eggen. 2015. Effects of feedback in a computer-based learning environment on students' learning outcomes: A meta-analysis. *Review of Educational Research* 85:475-511.

Van Merrienboer, J. J. G., and H. P. M. Krammer. 1987 Instructional strategies and tactics for the design of introductory computer programming courses in high school. *Instructional Science* 16:251-285.

Vanblaere, B. and G. Devos. 2015. Exploring the link between experienced teachers' learning outcomes and individual and professional learning community characteristics. *School Effectiveness and School Improvement* Published online 13 Jul 2015.

Wallentin, G., B. Hofer, and C. Traun. 2015. Assessment of workforce demands to shape GIS&T education. *Transactions in GIS* 19:439-454.

Walsh, S.J. 1992. Spatial education and integrated hands-on training: Essential foundations of GIS instruction. *Journal of Geography* 91:54-61.

Wang, P. and L. Dong. 2014. Web GIS research and application based on SVG. *Applied Mechanics and Materials* 513-517:2004-2007.

Wang, S. 2010. A CyberGIS framework for the synthesis of cyberinfrastructure, GIS, and spatial analysis. *Annals of the Association of American Geographers* 100:535-557.

Wang, S. L. Anselin, B. Bhaduri, C. Crosby, M. F. Goodchild, Y. Liu, and T. L. Nyerges. 2013. CyberGIS software: A synthetic review and integration roadmap. *International Journal of Geographic Information Science* 27:2122-2145.

Ward, P., N. J. Hodges, A. M. Williams, and J. L. Starkes. 2004. Deliberate practice and expert performance: Defining the path to excellence. In *Skill acquisition in sport: Research, Theory, and Practice.* A.M. Williams & N.J. Hodges (Eds.), 231–258. New York: Routledge.

Warren, S. 1998. GIS, cartography, and geography: The view from the other side. *Yearbook of the Association of Pacific Coast Geographers* 60:162-178.

Watts, S. and P. Stenner. 2012. *Doing Q Methodological Research: Theory, Method and Interpretation* London: SAGE.

Webler, T., S. Danielson, and S. Tuler. 2009. *Using Q method to reveal social perspectives in environmental research.* Greenfield, MA: Social and Environmental Research Institute. Retrieved from http://www.seri-us.org/sites/default/files/Qprimer.pdf.

Whyatt, D., G. Clark, and G. Davies. 2011. Teaching geographical information systems in geography degrees: A critical reassessment of vocationalism. *Journal of Geography in Higher Education* 35:233-244.

Wikle, T. A. and G. A. Finchum. 2003. The emerging GIS degree landscape. *Computers, Environment, and Urban Systems* 27:107-122.

Wikle, T. A. and T. D. Fagin. 2014. GIS course planning: A comparison of syllabi at US college and universities. *Transactions in GIS* 18:574-585.

Wright, D. J. 2012. Theory and application in a post-GISystems world. *International Journal of Geographical Information Science* 26:2197-2209.

Wright, D. J. and S. Wang. 2011. The emergence of spatial cyberinfrastructure. *Proceedings of the National Academy of Sciences* 108:5488-5491.

Wright, D., M. F. Goodchild, and J. D. Proctor. 1997. Demystifying the persistent ambiguity of GIS as "Tool" or "Science". *Annals of the Association of American Geographers* 87:346-362.

Wright, P. N. 2013. Is q for you?: Using q methodology within geographical and pedagogical research. *Journal of Geography in Higher Education* 37:152-163.

Yang, C., M. Goodchild, Q. Huang, D. Nebert, R. Raskin, Y. Xu, M. Bambacus, and D. Fay. 2011. Spatial cloud computing: How can the geospatial sciences use and help shape cloud computing? *International Journal of Digital Earth* 4:305-329.

Yildirim, A. A., D. Watson, D. G. Tarboton, and R. M. Wallace. 2015. A virtual tile approach to raster-based calculations of large digital elevation models in a shared-memory system. *Computers & Geosciences* 82:78-88.

Zandbergen, P. A. 2013. *Python Scripting for ArcGIS*. Redlands, CA: ESRI Press.

Zhou, X., C. Xu, and B. Kimmons. 2015. Detecting tourism destinations using scalable geospatial analysis based on cloud computing platform. *Computers, Environment, and Urban Systems* 54:144-153.

APPENDIX I

PAPERS USED TO GENERATE LIST OF CYBERGIS TERMS

To generate this list of CyberGIS terms, I extracted content from the papers listed below. To select these papers, I included papers recently published with the key word 'CyberGIS', papers published by the CyberGIS project, and background documents as listed on the CyberGIS project website. These selections resulted in a list of 40 papers used for my analysis. I provide the list here separate from the references for ease of review and replication.

Ahearn, S. C., I. Icke, R. Datta, M. N. DeMers, B. Plewe, and A. Skupin. 2013. Re-engineering the GIS&T Body of Knowledge. *International Journal of Geographical Information Science* 27:2227-2245.

Anselin, L. 2012. From SpaceStat to CyberGIS: Twenty years of spatial data analysis software. *International Regional Science Review* 35:131-157.

Anselin, L., S. J. Rey, and W. Li. 2014. Metadata and provenance for spatial analysis: The case of spatial weights. *International Journal of Geographical Information Science* 28:2261-2280.

Cao, G., S. Wang, M. Hwang, A. Padmanabhan, Z. Zhang, and K. Soltani. 2015. A scalable framework for spatiotemporal analysis of location-based social media data. *Computers, Environment, and Urban Systems* 51:70-82.

Goodchild, M. F. 2013. Prospects for a space-time GIS. *Annals of the Association of American Geographers* 103:1072-1077.

Goodchild, M. F., and R. P. Haining. 2004. GIS and spatial data analysis: Converging perspectives. *Papers in Regional Science* 83:363-385.

Gui, Z., C. Yang, J. Xia, K. Liu, C. Xu, J. Li, and P. Lostritto. 2013. A performance, semantic and service quality-enhanced distributed search engine for improving geospatial resource discovery. *International Journal of Geographic Information Science*27:1109-1132.

Guo, M., Y. Huang, and Z. Xie. 2014. An efficient approach to load balancing of vector maps in cyberGIS cluster environment. *Geomatica* 68:129-134.

Guo, M., Y. Huang, and Z. Xie. 2015. A balanced decomposition approach to real-time visualization of large vector maps in CyberGIS. *Frontiers in Computer Science* 9:442-455.

Huang, Q., C. Yang, K. Benedict, S. Chen, A. Rezgui, and J. Xie. 2013. Utilize cloud computing to support dust storm forecasting. *International Journal of Digital Earth* 6:338-355.

Kim, I-H., and M-H. Tsou. 2013. Enabling Digital Earth simulation models using cloud computing or grid computing – two approaches supporting high-performance GIS simulation frameworks. *International Journal of Digital Earth* 6:383-403.

Laura, J., W. Li, S. J. Rey and L. Anselin. 2015. Parallelization of a regionalization heuristic in distributed computing platforms – a case study of parallel-*p*-compact-regions problem. *International Journal of Geographical Information Science.* Published online May 6th, 2015.

Li, J., L. Meng, F. Z. Wang, W. Zhang, and Y. Cai. 2014. A map-reduce-enabled SOLAP cube for large-scale remotely sensed data aggregation. *Computers & Geosciences* 70:110-119.

Li, Z., C. Yang, B. Jin, M. Yu, K. Liu, M. Sun, and M. Zhan. 2015. Enabling big geoscience data analytics with a cloud-based, mapreduce-enabled and service-oriented workflow framework. *PLoS One* 10:1-23.

Li, J., Y. Jiang, C. Yang, Q. Huang, and M. Rice. 2013. Visualizing 3D/4D environmental data using many-core graphics processing units (GPUs) and multi-core central processing units (CPUs). *Computers & Geosciences* 59:78-89.

Li, W., L. Li, M. F. Goodchild, and L. Anselin. 2013. A geospatial cyberinfrastructure for urban economic analysis and spatial decision-making. *ISPRS International Journal of Geo-Information* 2:413-431.

Liu, Y, A. Padmanabhan, and S. Wang. 2015. CyberGIS gateway for enabling data-rich geospatial research and education. *Concurrency and Computation: Practice and Experience* 27:395-407.

Nyerges, T. L., M. J. Roderick, and M. Avraam. 2013. CyberGIS design considerations for structured participation in collaborative problem solving. *International Journal of Geographical Information Science* 27:2146-2159.

Padmanabhan, A., S. Wang, G. Cao, M. Hwang, Z. Zhang, Y. Gao, K. Soltani, and Y. Liu. 2014. FluMapper: A cyberGIS application for interactive analysis of massive location-based social media. *Concurrency and Computation: Practice and Experience* 26:2253-2265.

Richardson, D. B. 2013. *Real-time* space-time integration in GIScience and geography. *Annals of the Association of American Geographers* 103:1062-1071.

Shi, X. and S. Wang. 2015. Computational and data sciences for health-GIS. *Annals of GIS* Published online March 26th, 2015.

Shook, E., S. Wang, and W. Tang. 2013. A communication-aware framework for parallel spatially explicit agent-based models. *International Journal of Geographical Information Science* 27:2160-2181.

Sui, D. 2014. Opportunities and impediments for open GIS. *Transactions in GIS* 18:1-24.

Sui, D. and M. Goodchild. 2011. The convergence of GIS and social media: Challenges for GIScience. *International Journal of Geographical Information Science* 25:1737-1748.

Wang, S. 2010. A CyberGIS framework for the synthesis of cyberinfrastructure, GIS and spatial analysis. *Annals of the Association of American Geographers* 100:535-557.

Wang, S. 2013. Editorial. CyberGIS: Blueprint for integrated and scalable geospatial software ecosystems. *International Journal of Geographical Information Science* 27:2119-2121.

Wang, S. and M. P. Armstrong. 2009. A theoretical approach to the use of cyberinfrastructure in geographical analysis. *International Journal of Geographical Information Science* 23:169-193.

Wang, S. and Y. Liu. 2009. TeraGrid GIScience gateway: Bridging cyberinfrastructure and GIScience. *International Journal of Geographical Information Science* 23:631-656.

Wang, S., Y. Liu, N. Wilkins-Diehr, and S. Martin. 2009. SimpleGrid toolkit: Enabling geosciences gateways to cyberinfrastructure. *Computers & Geosciences* 35:2283-2294.

Wang, S., N. R. Wilkins-Diehr, and T. L. Nyerges. 2012. CyberGIS – Toward synergistic advancement of cyberinfrastructure and GIScience: A workshop summary. *Journal of Spatial Information Science* 4:125-148.

Wang, S., L. Anselin, B. Bhaduri, C. Crosby, M. F. Goodchild, Y. Liu, and T. L. Nyerges. 2013. CyberGIS software: A synthetic review and integration roadmap. *International Journal of Geographical Information Science* 27:2122-2145.

Wilkins-Diehr, N., D. Gannon, G. Klimeck, S. Oster, and S. Pamidighantam. 2008. TeraGrid science gateways and their impact on science. *Computer* 41:32-41.

Wright, D. J. 2012. Theory and application in a post-GISystems world. *International Journal of Geographical Information Science* 26:2197-2209.

Wright, D. J., and S. Wang. 2011. The emergence of spatial cyberinfrastructure. *Proceedings of the National Academy of Sciences* 108:5488-5491.

Xia, J., C. Yang, Z. Gui, K. Liu and Z. Li. 2014. Optimizing an index with spatiotemporal patterns to support GEOSS clearninghouse. *International Journal of Geographical Information Science* 28:1459-1481.

Yang, C. (P.), D. W. Wong, R. Yang, M. Kafatos, and Q. Li. 2005. Performance-improving techniques in web-based GIS. *International Journal of Geographical Information Science* 19:319-342.

Yang, C., R. Raskin, M. Goodchild, and M. Gahegan. 2010. Geospatial cyberinfrastructure: Past, present and future. *Computers, Environment and Urban Systems* 34:264-277.

Yang, C., D. Nebert, and D. R. F. Taylor. 2011. Establishing a sustainable and cross-boundary geospatial cyberinfrastructure to enable polar research. *Computers & Geosciences* 37:1721-1726.

Yang, C., Y. Xu, and D. Nebert. 2013. Redefining the possibility of digital Earth and geosciences with spatial cloud computing. *International Journal of Digital Earth* 6:297-312.

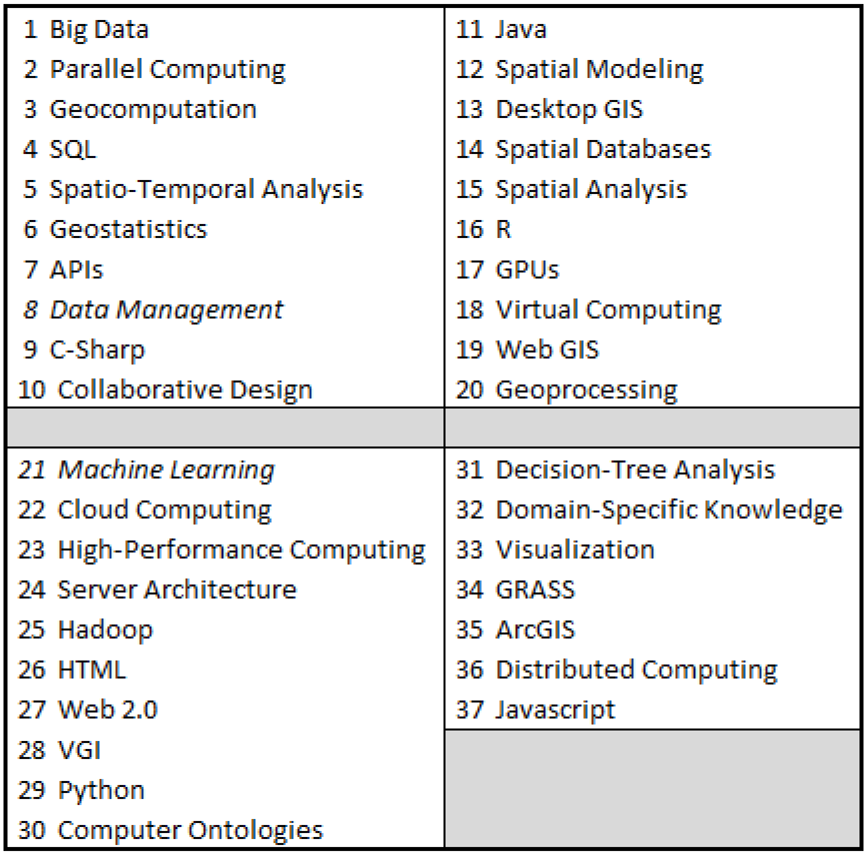
Zhao, Y., A. Padmanabhan, and S. Wang. 2013. A parallel computing approach to viewshed analysis of large terrain data using graphics processing units. *International Journal of Geographical Information Science* 27:363-384.

APPENDIX II

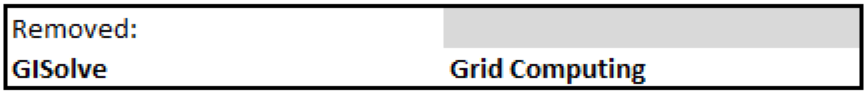
RECOMMENDED TERMINOLOGY FOR FUTURE Q-SORTS

For subsequent q-method research extending this work, I recommend the following set of terms (Table AII, below) to be used in place of the terms in Table 3-2. These updated terms reflect the feedback from participants in revising key terms in CyberGIS. I removed the terms ‘GISolve’ and ‘Grid Computing’, and added the terms ‘Data Management’, and ‘Machine Learning’. Based on feedback from the respondents, GISolve was considered too precise and too specific within certain constructions of the CyberGIS framework, and not relevant to other ways of interacting with CyberGIS systems. While many noted redundancies in types of computing in the sort, Grid Computing was most often referred to as redundant or not as relevant in the sample. The types of computing used in GIS are of interest, and future work might investigate that topic alone to determine how GIS users integrate computing and GIS topics.

Table AII. Revised set of q-sort terms based on respondent feedback. New terms noted in *italics*, removed terms listed at the bottom in bold. Numbers correspond to the random coding of the term for analysis, and should not be read as an implied hierarchy. I recommend that these items form the set of terms in future q-method surveys.



1 Big Data	11 Java
2 Parallel Computing	12 Spatial Modeling
3 Geocomputation	13 Desktop GIS
4 SQL	14 Spatial Databases
5 Spatio-Temporal Analysis	15 Spatial Analysis
6 Geostatistics	16 R
7 APIs	17 GPUs
8 Data Management	18 Virtual Computing
9 C-Sharp	19 Web GIS
10 Collaborative Design	20 Geoprocessing
21 Machine Learning	31 Decision-Tree Analysis
22 Cloud Computing	32 Domain-Specific Knowledge
23 High-Performance Computing	33 Visualization
24 Server Architecture	34 GRASS
25 Hadoop	35 ArcGIS
26 HTML	36 Distributed Computing
27 Web 2.0	37 Javascript
28 VGI	
29 Python	
30 Computer Ontologies	



Removed:	
GISolve	**Grid Computing**

I added data management based on feedback and responses that focused on how CyberGIS users structure and organize their data. This organization plays a major role in analysis, and respondents noted it would be an item of high value to them. Advances in machine learning, which informed not only the creation of my initial sample, but research and development in the CyberGIS realm, also received comments for inclusion. Similarly, the aspects of visualizing such data and analysis were also mentioned as important but missing from this sample, but determining what types of visualization were most important (data, cartographic, etc.) proved difficult. Since visualization already exists as a term in the sort, I did not want to duplicate it; instead, it may be an area of emphasis worth further focus in future realignments of the terminology. Based on statements from the respondents and my own reflections on the content, I believe this revised set of terms will serve future researchers well.

APPENDIX III

QUESTIONS FOR Q-METHOD INTERVIEWS

I used the following questions as guidelines for discussion with participants in the interview portion of the q-method activity. These questions serve as general guidelines for discussion, and were used to help guide discussions with participants.

1. Which term that you placed are you the most confident about?

2. Why did you place this term (pointing to the first term sorted) down first?

3. Why did you place this term (pointing to the last term sorted) down last?

4. Which terms were particularly challenging to place?

5. Which terms were easy to place?

6. Looking at your sort, do you see any patterns that stick out to you?

7. Do you think your colleagues would sort these terms the same way? Why?

8. Are there any terms you feel are redundant in this sort?

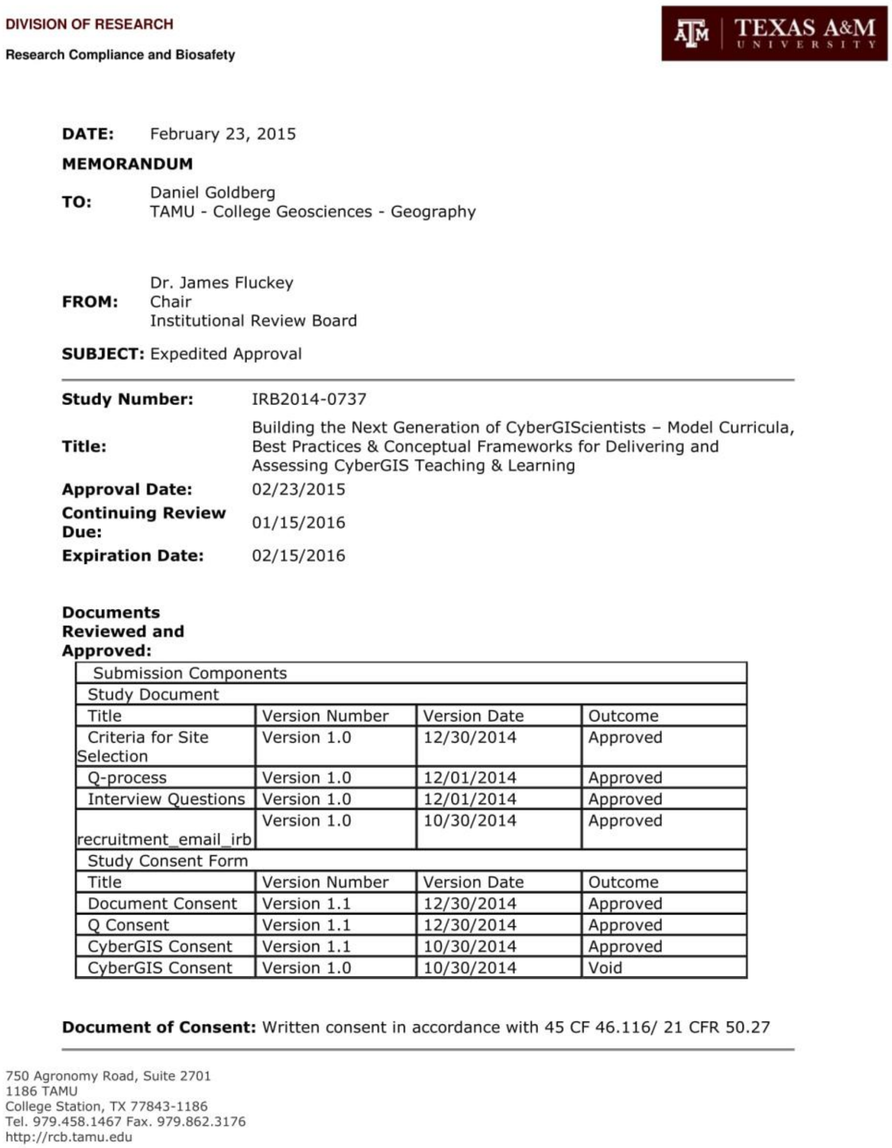
9. Are there any terms you feel are missing?

10. Do you have any other comments about the sorting activity?

APPENDIX IV

INSTITUTIONAL REVIEW BOARD APPROVAL

Content on the following pages display the Texas A&M Institutional Review Board approval for human subjects research in this dissertation.



DIVISION OF RESEARCH

Research Compliance and Biosafety

TEXAS A&M UNIVERSITY

DATE: February 23, 2015

MEMORANDUM

TO: Daniel Goldberg
TAMU - College Geosciences - Geography

FROM: Dr. James Fluckey
Chair
Institutional Review Board

SUBJECT: Expedited Approval

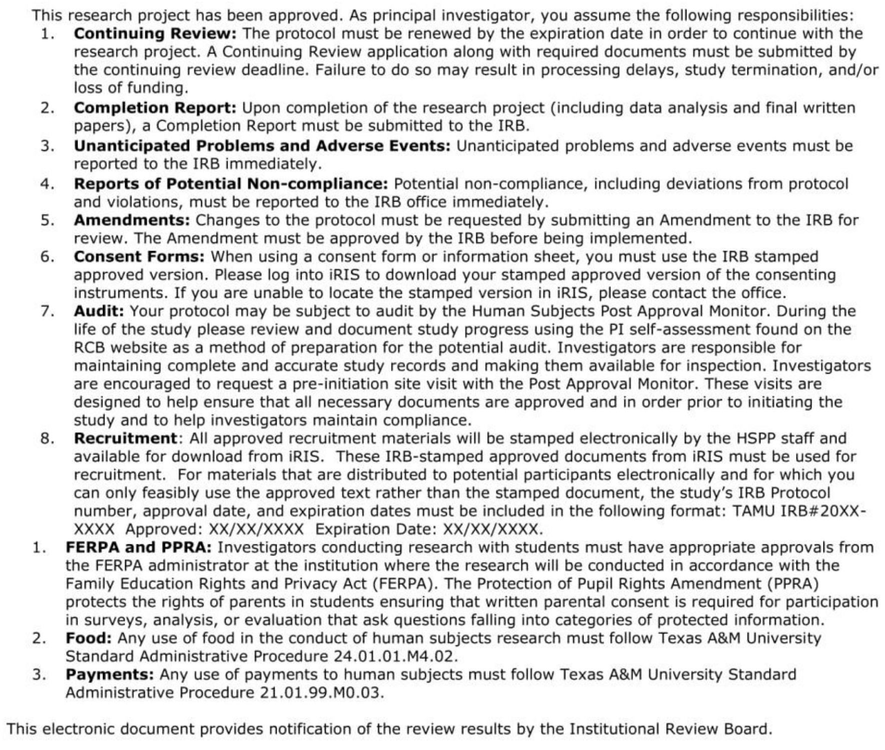
Study Number:	IRB2014-0737
Title:	Building the Next Generation of CyberGIScientists – Model Curricula, Best Practices & Conceptual Frameworks for Delivering and Assessing CyberGIS Teaching & Learning
Approval Date:	02/23/2015
Continuing Review Due:	01/15/2016
Expiration Date:	02/15/2016

Documents Reviewed and Approved:

Submission Components			
Study Document			
Title	Version Number	Version Date	Outcome
Criteria for Site Selection	Version 1.0	12/30/2014	Approved
Q-process	Version 1.0	12/01/2014	Approved
Interview Questions	Version 1.0	12/01/2014	Approved
recruitment_email_irb	Version 1.0	10/30/2014	Approved
Study Consent Form			
Title	Version Number	Version Date	Outcome
Document Consent	Version 1.1	12/30/2014	Approved
Q Consent	Version 1.1	12/30/2014	Approved
CyberGIS Consent	Version 1.1	10/30/2014	Approved
CyberGIS Consent	Version 1.0	10/30/2014	Void

Document of Consent: Written consent in accordance with 45 CF 46.116/ 21 CFR 50.27

750 Agronomy Road, Suite 2701
1186 TAMU
College Station, TX 77843-1186
Tel. 979.458.1467 Fax. 979.862.3176
http://rcb.tamu.edu



This research project has been approved. As principal investigator, you assume the following responsibilities:

1. **Continuing Review:** The protocol must be renewed by the expiration date in order to continue with the research project. A Continuing Review application along with required documents must be submitted by the continuing review deadline. Failure to do so may result in processing delays, study termination, and/or loss of funding.
2. **Completion Report:** Upon completion of the research project (including data analysis and final written papers), a Completion Report must be submitted to the IRB.
3. **Unanticipated Problems and Adverse Events:** Unanticipated problems and adverse events must be reported to the IRB immediately.
4. **Reports of Potential Non-compliance:** Potential non-compliance, including deviations from protocol and violations, must be reported to the IRB office immediately.
5. **Amendments:** Changes to the protocol must be requested by submitting an Amendment to the IRB for review. The Amendment must be approved by the IRB before being implemented.
6. **Consent Forms:** When using a consent form or information sheet, you must use the IRB stamped approved version. Please log into iRIS to download your stamped approved version of the consenting instruments. If you are unable to locate the stamped version in iRIS, please contact the office.
7. **Audit:** Your protocol may be subject to audit by the Human Subjects Post Approval Monitor. During the life of the study please review and document study progress using the PI self-assessment found on the RCB website as a method of preparation for the potential audit. Investigators are responsible for maintaining complete and accurate study records and making them available for inspection. Investigators are encouraged to request a pre-initiation site visit with the Post Approval Monitor. These visits are designed to help ensure that all necessary documents are approved and in order prior to initiating the study and to help investigators maintain compliance.
8. **Recruitment**: All approved recruitment materials will be stamped electronically by the HSPP staff and available for download from iRIS. These IRB-stamped approved documents from iRIS must be used for recruitment. For materials that are distributed to potential participants electronically and for which you can only feasibly use the approved text rather than the stamped document, the study's IRB Protocol number, approval date, and expiration dates must be included in the following format: TAMU IRB#20XX-XXXX Approved: XX/XX/XXXX Expiration Date: XX/XX/XXXX.

1. **FERPA and PPRA:** Investigators conducting research with students must have appropriate approvals from the FERPA administrator at the institution where the research will be conducted in accordance with the Family Education Rights and Privacy Act (FERPA). The Protection of Pupil Rights Amendment (PPRA) protects the rights of parents in students ensuring that written parental consent is required for participation in surveys, analysis, or evaluation that ask questions falling into categories of protected information.
2. **Food:** Any use of food in the conduct of human subjects research must follow Texas A&M University Standard Administrative Procedure 24.01.01.M4.02.
3. **Payments:** Any use of payments to human subjects must follow Texas A&M University Standard Administrative Procedure 21.01.99.M0.03.

This electronic document provides notification of the review results by the Institutional Review Board.