Innovation-Directed Experiential Learning Using Service Blueprints

Jayashree Ramanathan, Rajiv Ramnath, Michael J. Herold, Benjamin J. R. Wierwille
Department of Computer Science & Engineering
The Ohio State University
Columbus, Ohio 43210–1277
jayram@cse.ohio-state.edu, ramnath@cse.ohio-state.edu, herold.870@osu.edu, wierwille.3@osu.edu

***Abstract*—An analysis of hiring patterns showed emerging trends: the complexity of information technology (IT) is shifting from development to post-deployment and integration needed for services. Given the complexity of deployed service systems, generated big data, and the national dialogue on educating engineers, we asked ourselves related questions. Do our graduate students have evaluation skills needed to work at the most advanced level of Bloom's taxonomy? Can they learn to frame and solve the problems within complex industry environments while applying the current research? How do we structure a graduate curriculum and an environment that provides experiences in innovation within the constraints of the academic calendar? Here we present an interdisciplinary curriculum comprised of three components: a service interaction blueprint for framing the industry problem, agile principles focusing on aspects of the solution, and Christensen's theory-building to frame the next iteration of research. The environment for industry problems was created through an National Science Funded Industry & University Cooperative Research Center. The feedback from a pilot graduate-level class is positive and provides insights for further research. We show through feedback discussions that it is possible to have translational activity at the industry-university enterprise boundary resourced in by advanced experiential learning.**

Keywords*—complex systems, design, innovation, performance metrics, services science, technology management

I. Innovations for Complex Enterprise Systems

As early as 2004, an analysis of graduate student hiring patterns within industry showed several emerging trends: the complexity of information technology (IT) was shifting from the development phase to post-deployment. Enterprise IT departments were becoming brokers of cloud, social, mobile and information services [3]. And there is significant interest in converting data pumped out by these service complexes into actionable intelligence within enterprise workflows. This is now popularly known as the *big data problem* [4]. At the same time, we found little within the academic programs that teach the skills to frame problems related to the use of complex systems, learn from solutions and failures, and extract research that can provide real, perceptible value to their sponsors.

Computer science graduate students enter their second year with little insight into what it means to deliver services in the context of a complex enterprise system. They are mostly familiar with development methods and typically take the viewpoint of a programmer when approaching projects. . We use the term *complex enterprise system* here to mean both intra-enterprise and inter-enterprise services that collectively enable a business goal. As in [1, 2], we view these as networks (e.g. a value chain, supply chain, or service value network) with nodes as agents (i.e. humans, organizations, software and hardware) that are all creating, communicating, or consuming information.

The environment of the case study reported here is the specific National Science Foundation Industry & University Cooperative Research Center (I/UCRC) for Experimental Research in Computer Systems (CERCS). This is a multi-university center with CERCS at Georgia Tech researching complex enterprise systems, including their hardware, communications and system-level software, and applications. Complementary to this, the CERCS research site at The Ohio State University, the Center for Enterprise Transformation and Innovation (CETI), studies the applications of technology to achieve innovation. Thus by emphasizing the experimental method, CERCS promotes the creation of knowledge through the design, implementation, and measurement of large-scale systems. The goal is to conduct research that involves graduate students in industry research projects, yet is of interest to both the industry and the university [5].

Bridging the gap between technology consumers and technology providers is a key goal of CETI and the curriculum design discussed here. To accomplish this, graduate students need to think of how their services, when implemented, will benefit their sponsors and consumers, rather than focus on providing a technology implementation alone. However, we found that dealing with continually increasing dynamic complexity within the enterprise and its accompanying *wicked problems* [6] is rarely addressed in the typical graduate computer science curriculum. Often, even when an idea is adequately developed, the solution may not yield perceptible value to the enterprise, in that it fails to deal with the reality of delivering services with product and process innovations. Nevertheless, this is still good feedback for researchers and technology developers.

In order to address these gaps, we developed a graduate-level computer science course in applied information technology. We selected a set of interdisciplinary methods that focus on innovation-directed experiential learning and teach problem framing and

978-1-4673-5261-1/13/$31.00 ©2013 IEEE

research skills. Keeping in mind the time constraints present in the academic curriculum, we used a set of simple yet effective frameworks comprised of:

- *service blueprints* [7] for problem framing;
- *agile development practices* [8] as the process for delivering value to sponsor; and
- Christensen's *theory building process* [9] to extract the research statement from the project.

We assume here that by 1) framing problems to derive an innovation and development plan that applies within their sponsor companies, and 2) identifying research towards improving existing theories, the students will demonstrate evaluation skills to deliver improvements to complex systems. Thus they will function at the highest level of Bloom's taxonomy [10].

In this paper we begin with how we used the service blueprint to bridge from the *technology providers* (e.g. industry research labs, engineers in academia) to *technology consumers* within complex enterprise systems. Using this framework, we show how students use agile practices in their projects and extracted research statements from their project, all within one semester. We present feedback from students and offer insights toward improvements. We conclude with discussions of how graduate students can benefit project stakeholders (including themselves) through innovation-directed experiential learning.

II. Design of Curriculum Bridging Technology Consumers and Providers

We begin with issues in experiential learning and then introduce and justify the methods for use in the curriculum.

A. Importance of Context for Design

It is now well established that engineering design requires 1) an understanding of the context or environment for the correct framing of the problem, and 2) interdisciplinary approaches to solution development. For example, architectural design is viewed as its own integrated field of study, and is compared to other engineering disciplines; it is multi-disciplined, since the field seeks integration of electrical, plumbing, lighting, and other systems within its overall building design. Capture of contextual knowledge as *design patterns* was also first introduced by the architect Christopher Alexander [11, 12]. These design patterns are a way of making tacit knowledge explicit. Design patterns have been enthusiastically endorsed by software engineers leading to many framework technologies that improved the development of software. However, far less has been done in integrating interdisciplinary frameworks[1] related to complex enterprise services within the graduate software and systems engineering curricula.

Recent NSF workshops have looked at the process of introducing design thinking into the engineering curriculum [18, 19]. In addition, *grounded theory* provides a systematic methodology in the social sciences involving the discovery of theory through the analysis of data captured in the field [20]. It is mainly used in qualitative research, but is also applicable to quantitative data. All these ideas and methods bridge the gap between technology development and technology consumption in industry.

Finally, related to technology, engineering research continues to focus primarily on normative theories related to technology while business, such as management information systems, and healthcare focus more on descriptive or operational aspects, leaving a gap to be bridged. To address this *design science* [21] emphasizes the need for the information systems researcher to bridge the gap between existing knowledge and its context of use.

B. Experiential Curriculum Needs

In experiential learning, students learn to solve an industry-sponsored problem and the solution and outcomes are not predetermined. Many have identified the need to develop an advanced engineering "workforce of the future" capable of working more effectively with complex systems [22], the related need for advanced education that creates "T-shaped" individuals in technology areas [23], and new IT roles, such as a data architect or data scientist, with the appropriate complex skills [24].

While there is considerable focus on undergraduate education and accreditation, less national dialogue exists at the master's level. Related to software engineering master's requirements, the Integrated Software and Systems Engineering Curriculum has identified experiential capstone skills that "[can] reconcile conflicting project objectives, finding acceptable compromises within limitations of cost, time, knowledge, existing systems, and organizations. [25]" This type of work is at the highest levels of Bloom's taxonomy.

To address this curriculum need, we return to the technology consumer-provider gap and note that traditional engineering curricula have focused primarily on the creational (technology provider) aspects of software and hardware. To some extent, courses in management information systems cover operational (technology consumer) aspects. With the exception of industry best practices - TOGAF and ITIL – Enterprise Architecture curriculum assets that cover complex technology infrastructures and their related continuous improvement are limited. The challenges of enhancing the overall functionality and performance of a deployed complex system are left unaddressed within the curricula, with negative consequences illustrated below.

Problem-driven experiential learning is often interdisciplinary and questions that arise require faculty and

[1] Examples used by CETI include basic frameworks include: lean software development [13], TOGAF [14], ITIL [15], the balanced scorecard [16], and Porter's Five Forces model [17]. These frameworks are covered within different disciplinary silos.

practitioners from different disciplines to at least start with a shared understanding to develop solutions in a timely manner. To use a simple example, for a typical software engineer the word "activity" refers to a node in a finite state machine; for an industrial engineer this refers to a workflow task; and for a typical business graduate this term refers to an entire organization. While these terms are related in the field, they are rarely related in the disciplinary curricula.

Similar points are made in the emerging interdisciplinary field of Services Science, Management, Engineering and Design. Educators and researchers have pointed out that despite the increasing contributions of services to economic growth, there is no common understanding of what phenomena underlying services and the dynamics of services ecosystems create and drive value [26]. It has been argued [27] that existing models, traditionally used for describing the exchange of physical products, will not apply in the services context, in which close interactions between suppliers, service providers, and customers exist, where knowledge is created and exchanged, and experiences, capabilities, and relationships are an integral part of the transaction. In summary, an advanced experiential curriculum is important, difficult to resource, and demanding in terms of knowledge and needed skills.

Finally, students have limited available credit hours in their programs for the experience needed to correctly frame problems and solve them. Thus the methods selected for the course had to be simple to introduce, but effective in providing a structure for future growth as innovators. These methods are presented below.

C. Service Blueprint

In an effort to capture the needs of the service consumers of a complex enterprise, we selected a tool that is used in consumer sciences known as the service blueprint [7]. This framework identifies the core, peripheral, and enabling services within the enterprise. In the service blueprint, the core requires the student to identify the customer experience, from a tangible and emotional perspective. The blueprint is augmented with other services (organizational and technology) that have to be in place, thus requiring the student to elaborate on how a customer interaction moves through the organization. The service blueprint builds on methods of both computer and consumer sciences.

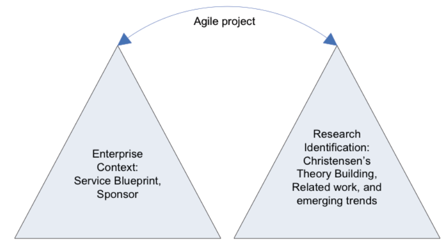

Fig. 2: The course frameworks and relationships: Service Blueprint for discovery, Agile for development; and Christensen for research identification.

D. Translational Process

Translation usually uses research to improve the methods of working to producing goods and services. The result is an innovation. The term *complex innovation* is innovation related to complex enterprise services. Complex innovation requires students to fully understand the underlying translational process cycles that are needed. These cycles often start from *in situ* observations and

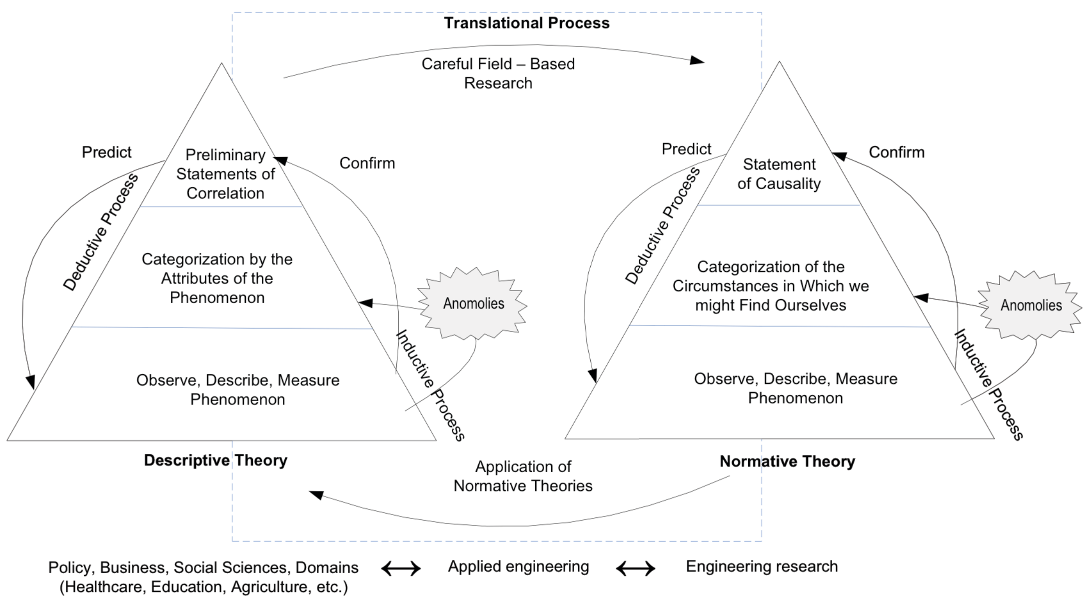

Fig. 1: The Engineering Translational Process bridging from descriptive to normative theories, based on [9].

understandings, i.e. development of descriptive theories (Fig. 1, left). Next, these lead to subsequent cycles of theory building that are well explained in [9]. In particular we next wish to build on key points about theory building that Christensen makes and relate them to the translational process involving, in particular, engineering innovation research.

First, the descriptive stage of theory building (Fig. 1, left) is viewed as a preliminary stage because translational engineers must understand the context and causalities that occur in the real world, e.g. the complex healthcare system. They then pass this understanding on to the development of a normative theory (Fig. 1, right). The right side is the typical realm of more academic engineering researchers. Normative theories have the ability to predict outcomes of interest, not just what is correlated in the field. Anomalies are treated with great interest as they often refine theories. We use the term *translation* here for the process bridging between descriptive and normative theories in order to cycle between understandings to causalities.

Second, the translational process (as in Fig. 1) is not simply a one-way process of taking ideas from university labs out to be applied in industry. Careful field-based research, anomalies and correlations must also move back to inform technology and engineering research in the university.

This deeper, normative understanding of predictability and repeatability is exactly what is needed as the basis for service automation and the development of enabling technology. For example, the design of telemedicine solutions must be based only on theories of automation because they are also repeated in human interactions. Care must be taken not to over-automate at the expense of usability. This type of nuanced understanding begins with observations and the development of a descriptive theory regarding the context of use.

The typical innovation goal is often initially stated in broad terms (e.g. social media can be used to improve the health care enterprise). Such a claim may never be warranted by existing underlying research, yet the innovation might be successful. In contrast, each academic research hypothesis and its accompanying goals are typically narrow and rigorous (e.g. text analytics tools can be used to extract the positive sentiment of entities in micro-blogs). In other words, within complex systems we may never complete all the research (e.g. due to wicked problems [6]) and we may have to acknowledge that a single good research result might not have a perceptible impact on the economic performance of the whole system.

There also is no guarantee that research investment will be successful in practice. Innovation may be the result of research and development, but also more rarely, it could be due to a "Eureka" moment. Thus, many wicked problems and different disciplinary perspectives need to be addressed for a typical complex system innovation. And, to achieve results, many translational cycles will typically be involved. Ways to enable translational activity within the university are discussed in [28].

E. Agile Translation

Last but not least we introduced agile development practices, which form the process used to conduct the projects. It consists of the following principles, based on [8]. Since the industry sponsor approves all work products based on achieving a translational goal state that is also funded, we point out that the work products are of value to an industry process or product. Thus, each work product contributes incrementally to a final innovation.

III. Implementing Innovation – Directed Experiential Learning

The students taking the course in Advanced Enterprise Architectures in Services were first- and second-year master's and doctoral students seeking real-world experience. Most were new to research and most had a year of previous work or project experience. The steps in the course (see Fig. 2) and curriculum is as follows:

1) Project triage: At the start of class the students were assigned projects and industry sponsors to work with. These projects are typically in the domains of healthcare, education, insurance and finance, and government services. The industry projects serve as a living laboratory or sandbox, meaning the projects represent the full messiness and complexity of real-world enterprises. In this early phase, the students were mainly introduced to the problem.

We assigned team members as follows: the team leader was a doctoral or second year master's student. The remaining team members were first year students with an interest in the project topic. Thus there was a mentoring relationship.

2) Develop service blueprint: In this phase, the students had to distill the messiness of the problem into an abstraction that identified, from the consumer's perspective, the true value of the final project. They were given a lecture on Service Blueprint.

3) Follow agile practices: On an on-going fashion, the students interacted with their sponsors and created project implementations. Concurrently, they did background and secondary research on related emerging trends.

4) Identify research: Using Christensen's theory building framework, the students began to understand and document where the unique contributions of their work might lie and developed a future project plan. They were introduced to Christensen's theory in two lectures.

5) Wrap-up: The groups present to their sponsors the project plan and the value of the project. And, they produce a research report describing their publishable work.

The course format was a hybrid online-offline classroom. Required reading materials were posted to the course website prior to class and five classes (of a total twenty-four) were spent on lectures. Thus more time was spent on in-class project discussions applying theory building and service blueprint methods.

The project environments included weekly activity tracking of students, kept private between the student and faculty advisor, and a project environment with blogging for team collaboration. Google sites were used for access to the work products by sponsors.

IV. Results

Twenty students took the pilot course in Spring 2013: six doctoral students, five second-year master's students, eight first year master's students, and one non-traditional undergraduate. The course was being offered for the first time so we had little understanding on what to expect. Thus we viewed the course as a way to discover and frame hypotheses for future research.

To assess the effectiveness of this course and the instruction method, we therefore prepared a survey for a longitudinal study to capture improvement in students' evaluation skills consequent to their instruction with the service blueprint. Christensen's framework was presented at the end of the class (they had previous knowledge of Agile).

We asked the same questions week three and week ten:

How do you define computer science research? How well do you feel you are equipped with methodological knowledge for research? How do you define computer science innovation? How well do you feel you are equipped with methodological knowledge for coming up with innovations? Do you consider innovation skills to be essential for employment? Do you consider research skills to be essential for employment?

The three groups are – *group 1:* nine first year masters students including one senior, *group 2:* nine second year masters, and *group 3:* three doctoral (playing mentor roles).

We administered these questions at the start and towards the end of the semester. Given small numbers and absences, we found no statistically significant change between the pre-instruction and post-instruction surveys. But taken the responses taken along with additional feedback solicited at the end of the class from each group was insightful and useful for developing hypotheses for further research. We have thus organized our results from the perspective of hypotheses forming evidence below.

Hypothesis: Graduate students understand the need for evaluation skills at the highest level of Bloom's taxonomy. They can define terms like service, innovation, and research and state the importance.

Group 1: Several saw the need for project skills but did not see the value of research and innovation skills for their future job search. They saw that the structure of the course as "different", "more autonomous", "bigger picture', and "not just "start coding". They could see that they have to gather requirements and understand what they are doing. Relationship with companies leading to internships is a plus; multidisciplinary perspective is a plus (much higher-level class, more than coding, gives aspects of industry that s/he's never seen). Suggestions for the course: "precisely define the terms", "give examples of hypothesis – unsure what it looks like", "good on-boarding", "started off seeming like an implementation course, and not a research course".

Our observations: We did not structure the relationships between project and research adequately enough to ask more precise questions. In retrospect, we think this was because the class was a combination of project and lecture based, with project discussion dominating. Thus the students focused on the Agile project aspects, but did not focus on the vocabulary even though that was presented and tested.

Hypothesis: Technology Graduate students can be taught to focus on technology consumers and innovation with Service Interaction modeling.

We noted by examining the project work products, that with the introduction of the service blueprint each team shifted its thinking process from implementation (technology provider mentality) to the consumer value. Most teams modified their blueprints several times as they thought more globally. In addition three Group 2 masters students developed satisfaction questionnaires for the technology consumers of their own innovations based on this. This was an important behavioral shift we were attempting to achieve. Consequently in the next iteration of the class, we plan to have work product related questions.

Hypothesis: Mentoring is helpful.

Group 1: "discussions with the team were helpful", "enjoyed working with partner (mentor), who is helpful", "class lectures were unclear; I now understand enterprise projects, so the class was helpful".

Group II: Several group II students mentioned "benefitted a lot from working with a Ph.D. student". Suggestions for the course: "start earlier with Christensen's Theory Building", "too long of a warm-up period, not enough time to do the research", "no direct tie between the EA (enterprise architecture) foundation and the class work, improve teaching of EA".

Hypothesis: Graduate students need to be taught all three frameworks and the relationships between them.

Group III: "Was previously unsure of how to write an academic paper", "the time wasn't too short, as s/he came in with more preparation"; "got good insights for own personal research"; "didn't know previously how to proceed with research, now has some sense on how to do research", "difficult to extract research from projects". Out observations, Group II and Group III were more able to see the benefits of the service interaction model and the relationships to the Christensen's model.

These responses provide some overall guidance on how to improve the experiential learning:

1) Include more explicit in-class discussions on the terminology.

For instance, include an explicit discussion about both Christensen's theory building model and innovation versus research. One way to proceed, given limited time, is to introduce a class simulation that illustrates the differences between research and innovation and the different types of reasoning processes.

2) Make the doctoral student experience more explicit.

Doctoral students gain because they get several teams working on exploring different ideas related to their research. This needs to be made more explicit.

3) Poor access to relevant knowledge.

We also make other, broader observations related to the environment. The conduct of an experiential curriculum and its accompanying translational activities is knowledge intensive and the resourcing has related challenges, as the faculty member has to have both explicit research background and deep industry experience. Confronted with typical sponsor needs in industry, we have also found that the typical academic research paper is often too narrow to solve the whole problem and show benefits to the end consumer. This, we believe, makes it increasingly difficult to derive innovations in short academic timeframes.

For example, during the problem framing stage it is typical to find thousands of academic publications that match keywords of interest. A simple search in Google Scholar for "adaptive complex enterprise" yields over six million results. Even so, most academic research papers do not address the whole problem and are too discipline-specific. Practice papers are too generic or vendor-specific. More integrative approaches to presenting research and practice must be explored to further the translational process. In particular, new approaches to the publication of translational work products and data sets related to the paper should be considered.

V. Conclusions

There is a growing gap between technology consumers and technology providers. This is also mirrored as the related gap between academic research rigor and effectiveness in practice. Thus, from an I/UCRC perspective, we are interested in asking: can advanced graduate students within experiential courses become the engine for translational activities and innovation related to complex systems?

The challenges to be addressed include lack of understanding of the 1) enterprise context, and 2) research and innovation processes. To address this we have proposed an advanced interdisciplinary curriculum framework integrating service blueprints from consumer science, Christensen's theory building from business, and agile practices from software engineering. We provide specifics of implementation and observations based on a pilot implementation.

Our preliminary results show that, while the application of technology increasingly depends on consumer context, this context can be quickly abstracted using available tools like service blueprints. Subsequent agile management of projects and extraction of research abstractions using Christensen's theory building is feasible within the constraints of one semester.

This approach has the potential of a sustaining industry-university structure that can benefit graduate student learning experiences and translational researchers. We have shown that students can acquire all the needed knowledge and deliver value within a limited time.

Acknowledgment

This work was done under the aegis of the Center for Experimental Research in Computer Science at Georgia Tech and The Ohio State University, funded by the National Science Foundation's I/UCRC program (NSF IIP 0753710). Funding was also provided by The Ohio Board of Regents Computer Science Graduate Education program and by the National Science Foundation Course, Curriculum, and Laboratory Improvement program (NSF CCLI Phase 0837555). We wish to acknowledge the helpful ideas by our colleagues, especially Dr. Jay Kandampully for his presentations and insights into the service blueprint, CETI members, and sponsors.

References

[1] R. C. Basole and W. B. Rouse, "Complexity of service value networks: Conceptualization and empirical investigation." IBM Systems Journal, vol. 47, no. 1, pp. 53–70, 2008.

[2] J. Ramanathan, "Fractal architecture for the adaptive complex enterprise," Communications of the ACM, vol. 48 no. 5, pp. 51–57, May 2005.

[3] J. McKendrick, "Line-of-business tech budgets may soon top IT department budgets: Gartner,". ZDNet, December, 2011.

[4] Gartner, "Big data," Retrieved March 30, 2013, from http://www.gartner.com/technology/topics/big-data.jsp.

[5] National Science Foundation, "NSF I/UCRC," Retrieved March 30, 2013, from http://www.nsf.gov/eng/iip/iucrc.

[6] S. S. Batie, "Wicked problems and applied economics," American Journal of Agricultural Economics, vol. 90, pp. 1176–1191, 2008.

[7] L. G. Shostack, "Designing services that deliver," Harvard Business Review, vol. 62, no. 1, pp. 133–139, 1992.

[8] K. Beck, and C. Andres, Extreme Programming Explained: Embrace Change. Boston, MA: Addison-Wesley, 2004.

[9] C. Christensen, "The ongoing process of building a theory of disruption," Journal of Product Innovation Management, vol. 23, pp. 39–55, 2006.

[10] B. S. Bloom, Taxonomy of Educational Objectives. Boston, MA: Allyn and Bacon, 1956.

[11] Wikipedia. “Christopher Alexander - Wikipedia, The Free Encyclopedia,” Retrieved March 30, 2013, from http://en.wikipedia.org/wiki/Christopher_Alexander.

[12] G. Bryant, “The Oregon Experiment after twenty years,” RAIN Magazine, vol. XIV, no. 1, 1991.

[13] M. Poppendieck and T. Poppendieck, Lean Software Development: An Agile Toolkit. Boston, MA: Addison-Wesley, 2003.

[14] D. Hornford, TOGAF Version 9.1. Zaltbommel, NL: Van Haren Publishing, 2011.

[15] R. A. Steinberg, C. Rudd, S. Lacy, and A. Hanna, ITIL Service Operation. London, UK: TSO, 2011.

[16] R. S. Kaplan and D. P. Norton, “The balanced scorecard: measures that drive performance,” Harvard Business Review, vol. 70, no. 1, pp. 71–79, 1992.

[17] M. E. Porter, Competitive Strategy: Techniques for Analyzing Industries and Competitors. New York, NY: The Free Press, 1998.

[18] W. Chen, E. Colgate, D. Norman, A. McKenna, E. Gerber, “NSF workshop on driving innovation through design: Engineering in the 21st century,” NSF Directorate for Engineering, 2010-2011.

[19] IDEO, “A design and innovation consulting firm,” Retrieved March 30, 2013, from http://www.ideo.com.

[20] B. Glaser and A. Straus, The Discovery of Grounded Theory: Strategies of Qualitative Research. London, UK: Wiedenfeld and Nicholson, 1967.

[21] A. R. Hevner, J. Park, S. T. March, and S. Ram, “Design science in information systems research,” MIS Quarterly, vol. 28, no. 1, pp. 75–105, March 2004.

[22] National Academy of Engineering, The Engineer Of 2020: Visions Of Engineering In The New Century. Washington, DC: National Academies Press, 2004.

[23] T. Kucharvy, “IBM’s role in creating the workforce of the future,” Beyond IT, 2009.

[24] C. Pettey and R. van der Meulen, “Enterprise architecture practitioners significantly influenced $1.1 trillion,” Retrieved March 30, 2013, from http://www.gartner.com/newsroom/id/2303215.

[25] iSSEc Project, “Graduate Software Engineering 2009 (GSwE2009),” Hoboken, NJ: Stevens Institute of Technology, 2009.

[26] A. M. Chaker, “Companies design, fund curricula at universities,” Pittsburgh Post-Gazette, September 11, 2006. Retrieved March 31, 2013, from http://www.post-gazette.com/pg/06254/721153-28.stm.

[27] “Service science,” IBM Research Watson. Retrieved March 31, 2013 from http://www.research.ibm.com/ssme/.

[28] J. J. Duderstadt, “Engineering for a changing world: A roadmap to the future of engineering practice, research, and education,” The Millennium Project the University of Michigan, pp. 131–156.