MAKING SERVICE ORIENTED ARCHITECTURE RELEVANT USING A MULTIDISCIPLINARY APPROACH

Thomas Way and Vijay Gehlot
Center of Excellence in Enterprise Technology
Department of Computing Sciences
Villanova University
Villanova, PA 19085
(610) 519-7307
thomas.way@villanova.edu
vijay.gehlot@villanova.edu

ABSTRACT

The use of a Service Oriented Architecture (SOA) approach to the design of enterprise-scale applications is increasing in industry, yet there are few sufficiently multidisciplinary college courses available for students that cover the topic the way students will see it in practice. In this paper, we present the design of a graduate level course that integrates elements of business and engineering into the computer science foundation. The original course design is motivated, and experiential data from the first offering is given. An analysis is provided, leading to guidelines for a relevant SOA course. Modifications for future offerings of the course, including an undergraduate version, are given based on our first offering experiences.

1. INTRODUCTION

Service Oriented Architectures (SOAs), the techniques for developing and deploying loosely coupled, interoperable, implementation independent network services that exchange data with each other [2], are continuing a trend toward increased adoption by industry [1]. The technology is sufficiently mature, and the available software tools make it possible to develop SOA services rapidly. Although the underlying design, development and networking issues are often complex, well-trained software engineers can master SOA and most of the technical challenges are

quite solvable. Thus, the key issue preventing broader adoption of SOA is the practice of governance [5], which includes establishing agreed upon standards for interoperability and the significant challenge of balancing the competition of the marketplace with the cooperation and collaboration required for SOA to be successful.

It is into this marketplace of significant but solvable software engineering challenges and identifiable but potentially intractable business practice conundrums that our graduates enter. The best prepared graduates who seek careers in SOA should be astute on the technical issues of enterprise-scale, networked, software design and deployment and also on the motivating business aspects that are manifest in the natural tension between competition and collaboration. Opportunities for education are few, but growing as the demand increases. In addition to a small number of vendor-offered, industry-oriented training seminars, when the proposal to develop this course was first made only Brandeis University offered a course specifically on SOA. In a recent online search, just six universities were found that offered a graduate or undergraduate course specifically on SOA, with a focus almost exclusively on technical aspects.

Software engineering curricula includes technical topics of software tools, methods and design, and typically acknowledges the business aspects needed to manage a software project [7]. The incorporation of sufficiently complex, real-world projects [4] and exposure to industry practices and experts [8] into a software engineering course can provide students with valuable experience applying technical concepts. To fully understand SOA, one also must be facile with its business aspects. The current dearth of formal SOA coursework, coupled with the expertise developed by faculty researchers at our institution gained through a DoD-funded, enterprise-architecture research project, motivated the development of a multidisciplinary course for graduate, and future upper-level undergraduate, computer science students.

2. COURSE DESIGN

This semester-long course incorporates three learning objectives supported by outcomes and a broad selection of content, met one evening per week for 2¾ hours, and had an expectation of student's performing significant outside-of-class work.

2.1 Learning Objectives

Three learning objectives, and corresponding outcomes, covering computer science, engineering and business aspects of SOA, were developed to provide students with the technical and theoretical foundations of SOA.

Objective 1 is to survey the topic of Service Oriented Architecture (SOA), including related aspects of computer science, engineering and business, and how these all relate within the larger topic of software engineering. Outcomes are that students will demonstrate an understanding of a broad range of SOA topics through a wide variety of course content and activities.

Objective 2 is to provide experience developing SOA applications individually or as part of a team. This objective and its outcomes were accomplished with a large-scale, team-based project, and numerous related lectures, laboratory exercises and reading assignments, giving students first-hand experience with the business and

people aspects involved with managing a team working toward a common goal and of encouraging collaboration among competitors, in this case among other teams in the class.

Objective 3 is to learn about modeling and simulation of Service Oriented Architecture, and gain experience using current, professional modeling and simulation software. This third objective and its outcomes were accomplished using CPN Tools and OPNET modeling software to provide hands-on application of SOA.

2.2 Content Development

Lacking a single, comprehensive SOA textbook and little existing classroom tested course material presenting the subject from computer science, engineering and business perspectives, a significant quantity and variety of course content was developed.

- Two **textbooks** on SOA were selected to provide a foundation, broad context and practical implementation techniques on the subject, although the texts were augmented with online readings. The texts used were *Service-Oriented Architecture (SOA): Concepts, Technology, and Design* by Thomas Erl [2] and *SOA: Using Java Web Services* by Mark D. Hansen [3].
- **Selected readings** from online sources supported lecture topics. Online sources were used to provide material that is as current and relevant as possible.
- **Lectures** on technical and business aspects of SOA were presented for a portion of each class meeting to introduce and reinforce material from assigned readings, and were made available to students ahead of time to assist with note-taking.
- **Guest speakers**, who were paid a small stipend, presented first-hand accounts of their professional, insider experiences working with technical, engineering and business aspects of SOA in industry. Scheduling suitable guest speakers was very time-consuming, and a number of ideal speakers were unavailable at class time.
- **Laboratory activities** enabled learning of the practical aspects of SOA software development using industry standard tools, with a focus on NetBeans. Students also modeled SOA using professional modeling software CPN Tools and OPNET.
- **Topic presentations** were assigned, with each student selecting a relevant, advanced SOA topic and preparing and delivering a short talk to the class on that topic, giving students deeper exposure to a variety of advanced topics.
- **Team-based projects** had students practicing real-world collaborative skills in the development of models and SOA services, equipping students with the technical skills and experience needed to work together on the final project, where student teams created interoperable services as part of a “smart house” simulation. Teams implemented services to produce and consume “live” house environment management data such as banking balances, grocery ordering, and power use billing. This project gave students first-hand experience with governance and collaboration issues that are critical to the use of SOA services in the marketplace. Details of this project, as well as plans for a variety of additional projects, are omitted here for space considerations but are available from the authors.

3. ASSESSMENT

To assess student learning and measure the effectiveness of the course design, the 13 graduate students in the course were given an initial survey to gather basic demographic information. Of the 13 students, 11 had earned a BS degree in computer science, with one BS each in business administration and finance. Student experience with software engineering was distributed relatively evenly, from no experience to more than 5 years experience. The majority indicated that they had some professional business management experience with 3 having served as managers and 8 (including the 3 managers) having performed project leadership duties. The remaining 5 students reported no previous business experience. All had experience programming in Java, most also had experience with C/C++, and about half with web applications and a variety of other programming languages.

To measure learning, students were surveyed at the start and finish of the semester regarding their understanding of SOA with regards to software design, business issues, engineering, networking and hardware issues, and general terminology or SOA literacy. For each topic, students began the semester distributed roughly evenly across the levels of understanding, from poor to excellent. At the conclusion of the semester, all students reported significant improvement to their understanding, with all winding up in the good to excellent range (Fig. 1).

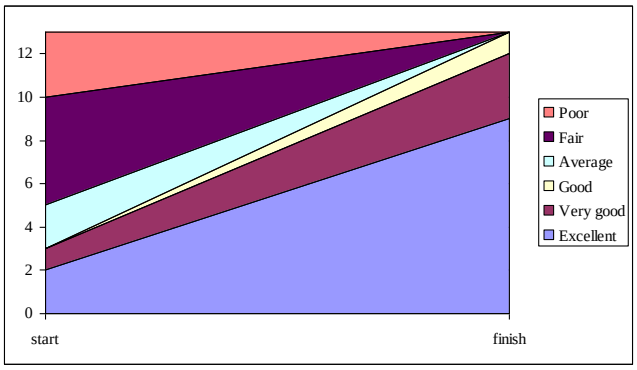

Fig. 1. Student self-assessment of familiarity with SOA terminology and general SOA literacy at start and finish of semester, a typical distribution for all topics surveyed.

As part of the semester-concluding survey, students were asked for feedback on various aspects of course content, specifically: topic coverage, reading material, guest lectures, exposure to current tools and technologies and SOA programming exercises. All 13 students reported that topic coverage was adequate, with numerous comments about liking the wide variety of topics. One student suggested an increased emphasis on SOA business process management to benefit students entering the workforce.

Reading material was judged adequate by 11 students, and inadequate by 2, although for an unexpected reason. Some felt that the textbook was not needed because of the wide availability of material online. The 2 inadequate ratings resulted from material being too advanced for students with less technical backgrounds. Two guest lecturers spoke during the semester, and 10 students found them to be adequate, serving as a “good break” from the other course activities, with inadequacies including topic repetition overly advanced material.

Current software tools and SOA technologies were covered, with 12 students finding it adequate. Students felt they gained a broad appreciation of the technology, security and business issues, and found the projects to be the most valuable activity to improve understanding. Shortcomings included the steep SOA learning curve, need for more lab time, and a desire for more coverage of SOA programming in lectures. Students suggested a desire for more and earlier programming projects, better team organization of the projects, fewer or no student presentations, more in-class activities and labs, and a better defined structure for the overall course.

4. ANALYSIS AND ADAPTATIONS

Student satisfaction with the course and its content was very high, with a few noted exceptions. Reading assignments from the textbook and online sources were judged valuable, but too advanced at times, and the textbooks were not necessary due to the preponderance of free and up-to-date reading material online. As a result of this feedback, a new text for the second offering of the course. *Applied SOA: Service-Oriented Architecture and Design Strategies* by Michael Rosen, et al. [6], was chosen for its accessible presentation of material, currency, good overview and design-oriented approach.

Lectures, homework and laboratory assignments were appreciated, with students suggesting more hands-on work early in the semester. In order to better prepare all students for the semester project, laboratory activities will be assigned earlier in the next offering to get students working with the technology earlier. Guest speakers were important to students, yet time-consuming to identify and schedule, so in the future we will begin earlier to line them up. Programming projects, in particular the large-scale final project, were viewed by students as invaluable to their learning, with the plan to introduce them earlier in a future offering.

Colleagues in academia and industry expressed satisfaction with the preparation students were receiving in the course. Viewed as most valuable to industry was the multidisciplinary “big picture” that includes business and engineering, and modeling and simulation (i.e., Formal Methods) as risk-management design approaches. As a result, we are producing a variety of lab assignments in modeling and simulation suitable for all levels of computer science students.

5. CONCLUSIONS

This paper reported on the design and first offering of a course for graduate students on the conjunction of computer science, engineering and business aspects of the emerging science of Service Oriented Architectures. Student learning was significant and all objectives were met. There was a variety of shortcomings and ideas for improvement expressed by the students, and by the instructor, and this feedback was actively used to design a second offering of the course which is currently in progress. We plan a comprehensive follow-up study to compare the results of the second offering with the version reported in this paper, and to continue to incorporate feedback of all sorts into further development of the course.

Students particularly appreciated any attempts to provide real-world experiences, through topical lectures, large scale projects and guest speakers, and viewed such content as invaluable foundation for their current or future careers. Because the course was offered at the graduate level only due to course scheduling constraints, a number of options for adapting the course to upper level undergraduates were noted and will be incorporated. For example, undergraduates will be given more, shorter programming projects, less advanced readings, and additional introductory material.

As a graduate course, a multidisciplinary course on SOA fits well into our Software Engineering and general computer science MS sequences. For undergraduates, we suggest that it be offered as an elective for a Software Engineering track. We are currently developing lab modules that cover the technology, business and engineering elements of SOA, with a particular emphasis on modeling and simulation that can be integrated appropriately throughout a computer science curriculum. Because SOA is increasingly widespread in industry, we believe that the best prepared graduates should have ample exposure to this important subject within the context of the multidisciplinary nature of SOA.

6. ACKNOWLEDGEMENTS

Support for the development of this course was provided by a Villanova Institute for Teaching and Learning (VITAL) Minigrant, with thanks to Dr. Carol Weiss for her suggestions and comments that helped with the initial course offering. Thanks also are due to members of the Villanova Center of Excellence in Enterprise Technology.

7. REFERENCES

[1] Michael J. Carey. SOA What? Computer, 41:3, March 2008, pp. 92-94.

[2] Thomas Erl. Service-Oriented Architecture: Concepts, Technology, and Design. Prentice Hall, Upper Saddle River, NJ, 2005.

[3] Mark D. Hansen. SOA: Using Java Web Services. Prentice-Hall, Upper Saddle River, NJ, 2007.

[4] Chang Liu. Enriching software engineering courses with service-learning projects and the open-source approach. In ICSE ’05: Proceedings of the 27th international conference on Software engineering, 2005, pp 613–614.

[5] Eric A. Marks. Service-Oriented Architecture (SOA) Governance for the Services Driven Enterprise. Wiley Publishing, 2008.

[6] Michael Rosen, Boris Lublinsky, Kevin T. Smith and Marc J. Balcer. Applied SOA: Service-Oriented Architecture and Design Strategies. Wiley Publishing, Inc., June 2008.

[7] Mary Shaw. Software engineering education: A roadmap. In: A. Finkelstein (ed.): The Future of Software Engineering: the 22nd International Conference on Software Engineering, New York, NY, ACM Press, 2000, pp. 371-380.

[8] Rayford B. Jr. Vaughn and Jeffrey Carver. "Position Paper: The Importance of Experience with Industry in Software Engineering Education. Conference on Software Engineering Education and Training Workshops, 2006, pp. 19.