JAD on a Shoestring Budget

Dr. Mario J. Spina
The George Washington University

John A. Rolando
Science Applications International Corporation

Why is it seemingly so difficult to adequately address that first and all-important development phase – gathering and defining software requirements? The use of Joint Application Development offers more than a worthwhile and proven approach; it can be adapted to accommodate the business and administrative challenges in requirements gathering when there is seldom enough time and resources available to do it right the first time. The following article discusses the details of how the Science Application International Corporation in McLean, Va. was able to inexpensively bring together users and developers to define complex and disparate requirements in a disciplined and effective manner. This laid a foundation for successful integration of application needs with existing commercial off-the-shelf and government off-the-shelf software tools and products.

The complexities of software development provide fertile ground for debate regarding which activities constitute its most critical steps and processes. However, many discussions on this subject suggest that the final frontier for successful software development is now, and may continue to be, the requirements gathering process.

For better or worse, gathering requirements demands involving the software application users, many of whom are neither educated nor experienced in the software development enterprise. Many software customers mistakenly believe that the up-front time spent in requirements gathering and analysis simply translates to an equivalent time delay in product delivery. This belief holds irrespective of studies and evidence showing that costs associated with correcting errors traceable back to poor up-front requirements, after fielding, can range from 68 to 200 times higher than the preventive costs associated with catching the errors during the requirements analysis phase [1].

With so much at stake, it is still surprising to learn how divergent the methodologies presented in textbooks and software journals are when discussing ways to elicit and rationalize software requirements. Few software engineering texts seem to provide much detail on how to elicit requirements, or otherwise may do so in one or more brief chapters or paragraphs offering only a short list of steps to consider in the process. Although referred to by a variety of names, a software requirements elicitation process called Joint Application Development (JAD) recognizes that requirements gathering is a very social endeavor [2]. It proposes a disciplined process for collecting, understanding, and organizing the innumerable and enigmat-
JAD on a Shoestring Budget

JAD in Action

Each day of the JAD conference began with a keynote talk by a prominent industry figure having a strong grasp of technology trends in the field. The first two mornings, attendees viewed product presentations from each vendor in four separate conference rooms, one for each of the business process specialties: procedure development, document management, mobile procedure execution, and data recording and reporting. The first two afternoons attendees were asked to indicate what they liked and did not like about the products they had seen during the morning sessions.

Professional facilitators worked to cultivate customer and vendor interchange and turn customer comments into system requirements. The facilitators were supported by stenographers and employed an on-screen requirements database. The full-screen view of each requirement, displayed as simple, standalone statements allowed customer groups an opportunity to specify, negotiate, and validate the precise phrasing of each system requirement. The facilitators focused upon building requirements consensus through technical discussion and point negotiations.

On the third and final day, the results of the previous two days were consolidated and presented in summation to the collective audience. Before lunch, customers were asked to rate the products they had seen, identify a value and a desire to purchase such products, and indicate their willingness to fund a political action committee effort to develop a product that would achieve the set of requirements identified at the conference. As a structured workshop approach, JAD is normally performed in relatively small classroom-sized settings. Having 80 participants placed a premium on visual aids, experienced facilitation, and opinion management. The large full-screen display was critical.

Consortium Results

The full affair was well done with three quality meals each day and full-time coffee, tea, and soda service. The total bill for the event was approximately $35,000, but SAIC paid only a small portion of the costs. The product vendors were willing to sponsor meals and advertisement banners, and attendees were willing to pay a small $100 conference fee. Overall, the venture was well worth the effort, not only for its informational value but also for the industry goodwill it engendered and the networking opportunities it offered.

The data from the survey was extensive. Customers had not only indicated their requirements and the products they preferred, but also indicated an initial willingness to purchase the associated software solution. SAIC was thereby armed with the information needed to select integration partners, the product requirements, the purchase price, and the business probability data from which to make a project go/no-go decision.

With this information, a COTS application architecture was defined; key vendors provided cost estimates for the licensing, interface development, and integration of their products into a final solution. With an assumption that initial development and roll-out costs could be spread across the set of customers who indicated a high likelihood of product purchase, a per customer solution cost was estimated. From this estimate, the SAIC architect was quickly able to determine that, unfortunately, product costs significantly exceeded customer perceived value. Accordingly, this product development effort was cancelled.

While this JAD did not result in a successful software product development, it presented some valuable information and experience. The effort showed how the JAD approach could be used when there is a need to develop and refine a software solution that integrates an existing product base. It also showed how a software requirements elicitation involving software users, product vendors, and a systems integrator could be accomplished on a shoestring budget. Lastly, there was the advantage of having multiple poten-
Epilogue

JAD is, of course, one of a multitude of requirements elicitation techniques. We have found no actual software development data comparing the use of JAD with alternative procedures and techniques in terms of relative successes or failures. In addition, since all projects are somewhat unique, we believe any comparative empirical data would somehow have to be adjusted for the multitude of other diverse variables to be meaningful.

It would seem intuitive that there would be advantages in having a dedicated gathering of users, developers, and customers together in a structured setting, compared with shorter piecemeal sessions or multiple one-on-one sessions. However, it is also easy to imagine circumstances whereby just the converse would be true, i.e., that shorter piecemeal sessions or multiple one-on-one sessions would be better if, for example, the JAD facilitator was somehow skewing inputs directly or indirectly via the recording process [8]. Thus, before any reader commits to using the JAD approach, we suggest they perform their own analysis and thought. We offer the following list of sources and Internet sites for further research.◆

References


Additional Reading


Web Sites

- <www.utexas.edu/hr/is/pubs/jad.html>
- <www.dcm.com/events/jad>
- <www.credata.com/research/jad.html>
- <www.russellmartin.com/courses/JAD.htm>
- <www.carolla.com/wp-jad.htm>
- <www.csweb.cs.bgu.ac.il/maner/domas/RAD.htm>
- <www.trainersdirect.com/outlines/JAD.htm>
- <www.sisg.com/JAD.htm>
- <www.verhofe.com/cs/njrad.htm>
- <www.computer.muni.cz/espresse/CATALOG/rs00072.htm>
- <www.mgromanlaw.com/articles/mhl/joint_application_development.htm>
- <www.mgrossmanlaw.com/articles/joint_application_development.htm>
- <www.csweb.cs.bgu.ac.il/maner/domas/RAD.htm>
- <www.trainersdirect.com/outlines/JAD.htm>
- <www.sisg.com/JAD.htm>
- <www.verhofe.com/cs/njrad.htm>
- <www.computer.muni.cz/espresse>CATALOG/rs00072.htm>
- <www.mgromanlaw.com/articles/mhl/joint_application_development.htm>

About the Authors

Mario J. Spina, D.Sc., is a part-time faculty for the George Washington University School of Business and Public Administration and the School of Engineering and Applied Science, as well as deputy division manager for Science Applications International Corporation’s (SAIC) Aerospace Technology Applications Division. As a new product development manager for SAIC, Dr. Spina led the company’s charge into the wireless mobile worker software market. He has a doctorate in science degree in engineering management from The George Washington University, a master’s of science degree in electrical engineering from the University of Southern California, and a bachelor’s of science degree in mechanical engineering from California State University, Northridge.

310 Cabin Rd. SE
Vienna, VA 22180
Phone/Fax: (703) 242-6732
E-mail: mspina@gwu.edu

John A. Rolando is currently a senior operations research analyst for Science Applications International Corporation (SAIC) Information Technology Solutions Group. He has been directly supporting communications interoperability analyses and the development of two detailed communications models for the Command, Control, Communications, and Computer Systems Directorate, Joint Staff, the Pentagon, since 1996. He has a bachelor’s degree in economics from LeMoyne College, Syracuse, a master’s degree in public administration from the University of Northern Colorado, and a master’s degree in business administration from Marymount University, Arlington, Va.)

Science Applications International Corporation
1710 SAIC Drive
McLean, VA 22102
Phone: (703) 676-4682
Fax: (703) 356-2534
E-mail: john.a.rolando@saic.com