JOURNAL OF OBJECT TECHNOLOGY

An Increase In Value

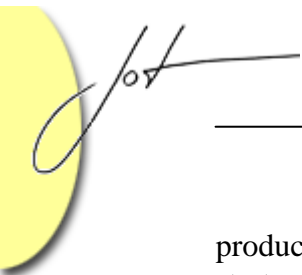**John D. McGregor**, Clemson University and Luminary Software LLC, U.S.A.

Abstract

Value is not always created as explicitly or as tangibly as with a product that benefits its user and its creator. Increasingly, value emerges from a set of activities and the knowledge of how to perform those activities. There is also value in espousing a set of values that guide the operation of an enterprise. In this issue of Strategic Software Engineering I want to expand on a previous column on Value, thereby Increasing its Value. I will also explore the relationships between the different ways that value is created.

1   INTRODUCTION

As I start the first draft of this article I am teaching a two-day course for software professionals at the SEI. And I routinely teach courses at Clemson University on topics such as software architecture. I feel that both of these activities add value to the lives of the students in the course, but the creation of this value is difficult to plot on the value chain that I discussed in the November-December 2007 issue of Strategic Software Engineering [McGregor07]. You can plot the creation of value of the course as a product of Clemson University, but not the increase in value of the students to future employers or the value to the students of further study.

One of the comments that I received after the first column on value was from a consultant who had a different view of how he is of value to a client. To a degree the difference is the difference between producing a product and providing a service. The economy of the world, and in particular the economy of the United States, has moved to become more of a service economy. "Knowledge workers" is a term heard often to describe one segment of the service economy.

In some cases the service economy dominates an activity. Take the cellular phone industry. Many of us, who use the phone as just a phone, pay for the service but never pay for the cell phone product. The phone is given to us in order for us to use, and pay for, the service. A service is much more difficult to price than a product that has constituent pieces that have material cost; however, IBM has found that it is much harder to reduce prices of services than for products. The cost of services, at least until robotics advances a bit more, are dominated by labor costs. In this environment, an increase in

productivity of workers is an increase in value. In a software product line organization the strategic levels of reuse reduce the labor content of each product and thus the cost of the product.

Another change in business is the emphasis on collaboration. Even when a product is created, the value of the product may not explicitly benefit the producers of the product. Open source projects increasingly count on industry contributions of labor to accomplish their agendas. These projects bring together a large number of organizations in a setting in which none of the companies own the intellectual property but all will benefit from it. They benefit by choosing open source projects in which to invest that will add value to their other products and services.

There are new models of value that correspond to the changes in business models. In addition to value chains, there are the value shop and value network models. These models are closely related and have much in common. In the next section I will explore these models. After that I will apply these ideas to software development.

2   VALUE CHAINS, VALUE SHOPS, AND VALUE NETWORKS

I gave extensive coverage to value chains in the previous value-oriented column so I will only give a brief overview followed by more detailed descriptions of value shops and value networks.

Value chain

Porter's value chain model sequences the set of activities in creating a product and describes how each activity adds value to the final product [Porter 98]. The model is really intended to describe the manufacture of a products as a sequence of activities and does not address the highly iterative process of innovation. The primary activities in Porter's value chain are:

Inbound Logistics – the initial handling of raw materials prior to producing products

Operations – the product manufacturing process

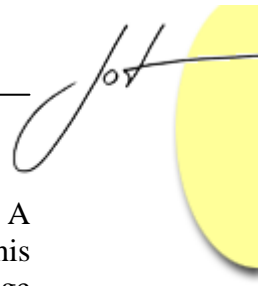**Outbound Logistics** – the management of finished products until they are delivered

Marketing & Sales – identifying customer needs

Service – support of customers after the sale

Even though Porter's model mentions service, it is at the end of the chain and describes how the company supports a user of its product rather than how to provide a service.

The value in the value chain is in the output of the activities in the chain. The demand for that output determines how great that value is.

Value shop

The value shop is an organization that adds value through solving customer problems. A basic representation is shown in Figure 1 [Stabell 98]. The circular nature of this representation shows that as the solution is refined, more specific levels of knowledge may be identified and brought to bear on the problem. Often the value is added through consultation or contracting services. Luminary Software, LLC, mentioned in my biography at the end of this column, is a value shop. We provide value through advice on technical and some managerial issues related to the adoption of the software product line strategy. As a consulting company we do not actually build a product, either in-house or or not, we assist the customer in solving problems by directing them in what decisions to make and how to make them.

Contracting companies provide skilled people to augment existing staff capabilities. A contractor typically does not have primary product responsibility but the goal of the value shop company is to provide whatever the customer needs. In both consulting and contracting models, the value added depends on the value of the supplied personnel not the value of a product provided. Many out-sourcing ventures have been with companies that follow this model. Where these efforts have failed it has often been because either the resource provided was not as valuable as the value shop promised or because the customer felt they could drop off a problem and pick up a finished product.
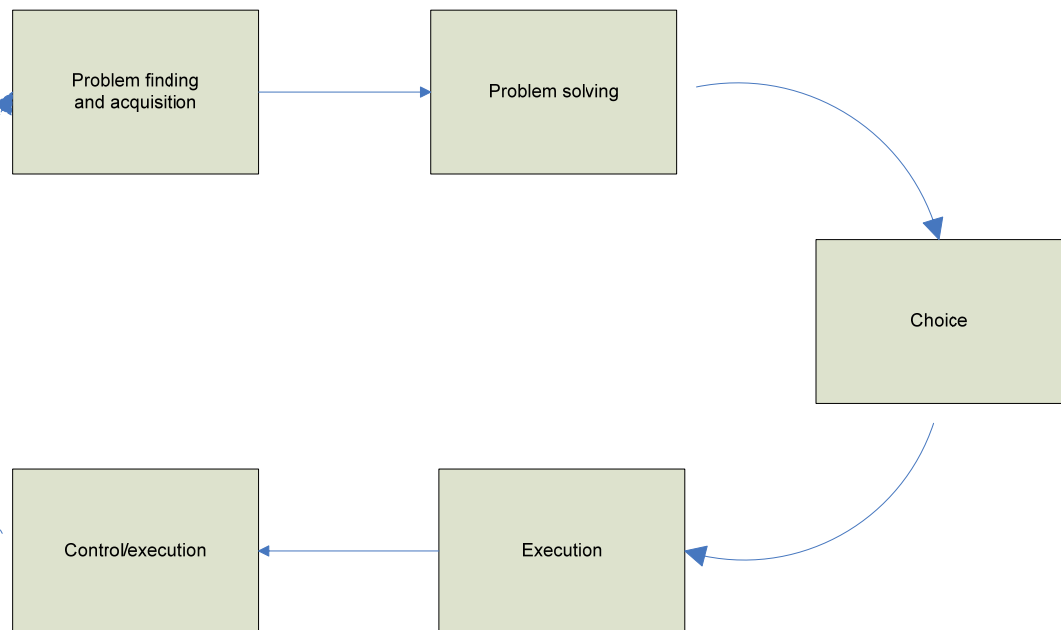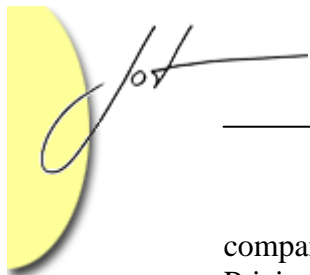

Figure 1 Value Shop diagram

A value shop increases its value to customers by increasing its abilities at solving relevant problems. The shop can hire people with skills that complement those already on board and the shop can train existing people in areas in which they are not currently competent. The implication here is that the value resides in the person, not the company, so the value of the company fluctuates based on the persons currrently employed. The value in the

company is a residual value that comes from a reputation for hiring competent people. Pricing a value shop for sale is part estimating how many people will remain under the new management, part valuing the materials that have captured the information in the heads of the employees, and partly estimating the residual value of returning customers who continue to use the services of the company under the new ownership.

The value in a value shop is in its people and their expertise. The demand for that expertise determines how great that value is.

Value network

The value network creates value for the participants in the network through the associations established among network members. A basic representation given in [Stabell 98] is shown in Figure 2. I am a member of LinkedIn, a value network for professionals. I am linked to a number of people whom I have known for many years and some whom I have met professionally recently. I can request information, help, employment or assistance finding an employee. The value of LinkedIn to me is in part based on how often I am able to satisfy a need I have and in part on how often I reap benefit by helping someone else in the network. The size of the network and the relevance of its members to me are its value.

Network promotion and contract management
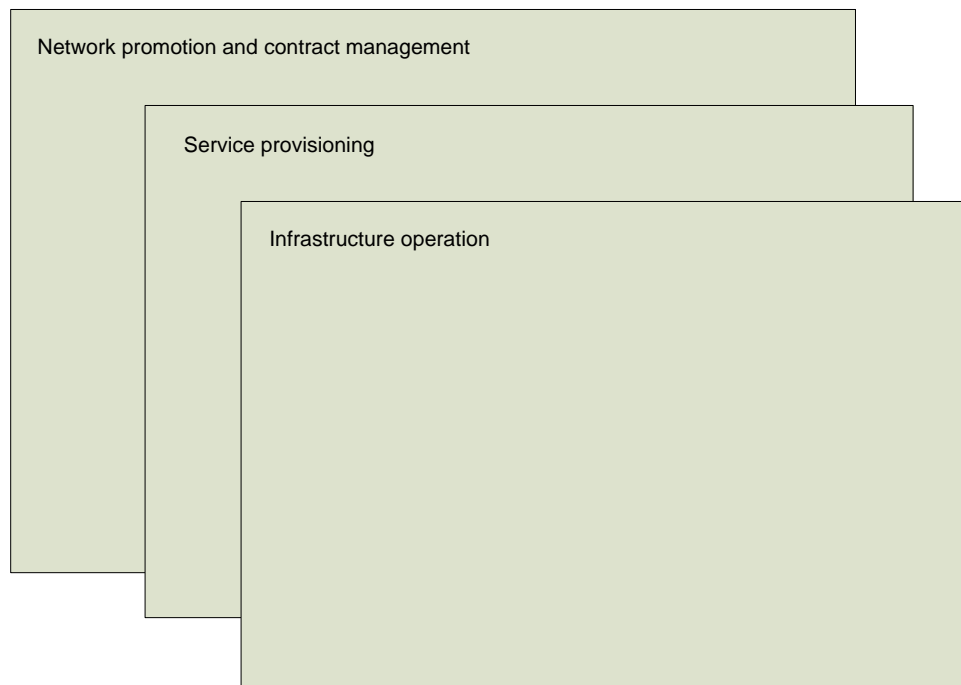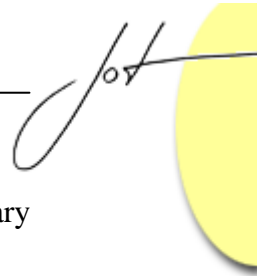
Service provisioning

Infrastructure operation

Figure 2 Value network diagram

I can create value by inviting people to join the network. Most networks have mechanisms explicitly for that purpose. It is valuable to have the invitee join and even more valuable when the invitee brings along other new members to the network. The

network increases in its value to me without me doing any work when these secondary people participate.

Notice that I speak of the value of the network. I have a hundred or more links but each of them has links so my connections are multiplied across the network. In fact I gain very little direct value from linking to people I knew before joining the network. It is the full connected graph, with thousands of people with whom I would have had no connection otherwise, that potentially has value to me, not just my direct connections.

Some networks give the user explicit access to these links for broadcasting or searching. In fact most networks are free to join but the more tools you access to extract value from the network, the more you pay. Advanced search tools provide ways of accessing the characteristics of people in the network and allow me to quickly determine the appropriate people to contact.

A new form of value network is emerging. Cloudsmith is an example of a company that is providing an infrastructure in which users can build networks of software components, libraries, and repositories [Cloudsmith 07]. The value in this network is in the content, being able to find what I need to solve a problem, and in the speed with which I can collect what I need. Cloudsmith also provides a means of aggregating elements of the network into distros and adding them to the network (or cloud). This type of value network can provide real value by collecting together often used combinations of open source products, but it can also be of value behind the firewall in a company by supporting the formation of clouds that cross business unit boundaries, or other artifical obstacles, to make software assets available across a company or around the world.

The value in the value network is in the associations. The relevance of the relationships determines just how great that value is.

3   VALUE IN SOFTWARE DEVELOPMENT

All three of the value models described above are relevant in software development. Everyone believes that the value model they are using is the best but I believe the three are complementary, intertwined, and that none gives a complete model for the value in modern software engineering by itself. In this section I want to describe how all three are applicable to software development and then in the next section I will discuss how the models are useful in organizing the activities in a software product line.

First, I want to distinguish between a software development model and a business model. Interest in iterative, incremental development or agile development methods is different from the business model by which the company interacts with customers and deploys products. The fact that the value chain business model is somewhat sequential does not limit a product company that uses the value chain to model its business value to sequential software development. The operations phase of the value chain model encapsulates whatever development strategy is being used.

There are many software development companies that still sell software products using an approach for which the value chain model is viable from a business perspective. They manufacture software for general sale or by contract. They have logistics issues such as when to introduce the latest version of a compiler into a team. They have operational issues such as network availability for a world-wide development organization. They have orchestration issues among globally distributed, multi-cultural teams. Their product has value to the producing company in terms of the revenue derived from it through licensing fees and maintenance subscriptions. The product is intended to create value for customers by supporting a reduction in personnel needed for a specific task, or by helping existing personnel do more, more quickly, or improving the quality of the current business process.
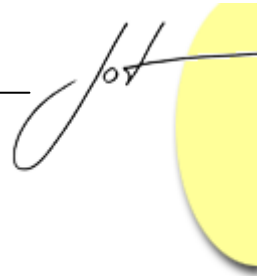
Many of these product companies also consult in the area of their domain expertise acting as a value shop. The business model is expanded to include this line of business in which their employees act as development resources. The two lines are mutually supportive. While consulting, the employees are able to identify additional needs in existing products or new ones. They also have the opportunity to update and expand their domain expertise.

Companies whose business model involves value networks, such as Cloudsmith and others, often decide to productize their infrastructure. They package and sell the software that supports their network operations. They become product companies for a second stream of revenue and have a separate value proposition for that line of business.

Open source software development is a good example of two or more value models coming together in a business strategy. The open source strategy does not require any specific software development style although many projects have adopted a similar style. There are a number of companies that act as value shops by providing assistance with obtaining, installing, and using fundamental software such as Linux. In order to ensure a steady supply of customers needing assistance, these same companies donate labor to the value chain that is formed around the open source product development. The contributions developed by these companies may have resulted from agile or iterative styles of development.

Companies increasingly rely on a network of alliances and dependencies among people and companies to obtain both the components they need and the advice they need. These business interactions do not require that the software development method be some type of networking approach. Many of the issues about licenses and protection of intellectual property have been eliminated and replaced with a network of alliances built around open source software development efforts in which none of the parties own the intellectual property.

Most companies use each of the three models at a certain point in the maturity of their organization and their product base. The different models produce value in different ways and require different actions on the part of the companies but they are mutually dependent. For example, the process of innovating to define a new product relies heavily on the skills of the people involved acting as a value shop to supply the inputs to the value chain that eventually produces the product.

4 VALUE IN A SOFTWARE PRODUCT LINE

In the previous issue on Value, I briefly described some initial ideas about value in a software product line organization. Now I am going to expand on those ideas.

The value chain model provides a reasonable model for the development of the individual products. Our work in production plans has illustrated that with planning, the similarity among the products in a product line supports the development of efficient development streams [Chastek 06]. Some product line organizations have achieved high degrees of automation in product production.

The value network is a useful way to think about the value built up in the virtual organization formed by the core asset and product building groups in a software product line. Globally distributed product line organizations develop a variety of relationships among their units with many different governing structures.

Hecker [Hecker 07] analyzes the value network for Firefox using Christensen's categories [Christensen 03] and I will use those same categories to evaluate the Eclipse ecosystem, which I have previously discussed as a product line [Eclipse 07]. The Eclipse ecosystem constitutes a rapidly expanding value network. Christensen's view of a value network includes "upstream suppliers; its downstream customers, retailers, and distributors; and its partners and ancillary industry players" A brief analysis of Eclipse's ecosystem from the network perspective yields:

Upstream suppliers: Eclipse projects use the output from numerous open source projects other than Eclipse. These projects coordinate the efforts of numerous individual contributors and workers assigned by their employers.
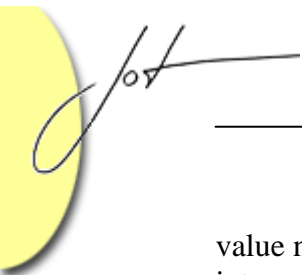
Downstream customers: Eclipse software is used by numerous other software development projects including some commercial software tool vendors.

Retailers: While Eclipse is not sold, a number product integrators use various outputs from the Eclipse organization to provide solutions to their customers. The Eclipse License makes this possible.

Distributors: The Eclipse ecosystem includes a number of mirror sites. These sites provide this service for a variety of reasons. Research universities may do it to reduce the load on the infrastructure when a large number of students on campus – one of my classes – need to accomplish a substantial download. For a smaller company this may bring the company's name in front of users of products complementary to the company's products.

Partners: Clemson University is an academic member of Eclipse. The Eclipse Foundation members wish to support, with advice and dues, the work of the Foundation. They also want to have input into future directions that are of greater weight than using Bugzilla to request new features.

Ancillary industry players: There are numerous numerous downloaders use the Eclipse software for a variety of purposes. These users form part of the ecosystem and a

value network. They participate through user forums in which users and project personnel interact. Future contributors come from this interaction.

The product line organization seeks to create value by separating concerns about reuse from concerns about product production. The producers of reusable assets take a breadth-first, investment view of producing value. They build the value network that supports the complete product line. The product builders take a depth-first, expedient view. A product team forms a value chain from the value network of assets to produce a product as rapidly as possible.
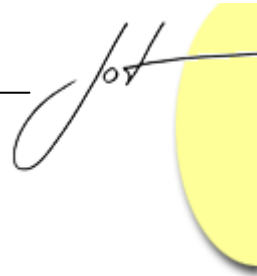
5   VALUE IN VALUES

The values of the company, its people, and its customers contribute to the value of its products. We have all seen the values statement of one organization or another. In some cases lip service is paid to those values, in some cases they are forgotten, but in some cases they are the guide to decision making.

Employees that are doing well have values that align with the organization with which they are affiliated. One of the reasons why open source software has been successful is that what is valued in an open source project aligns with the values of many developers. They can focus on the solution to a problem. Open source projects such as Eclipse where a large percentage of the contributors are actually assigned to the project by an employer worry less about aligning values although there is still a different culture on that project from proprietary projects.

I have consulted with many companies that valued meetings. Being "in a meeting" was a valid excuse for missing a deadline. Many of these companies are dead or in the process of resurrecting themselves from near death experiences. Companies that are doing well today value getting products with the right features into the hands of customers quickly.

It is not an easy task to create an environment in which people value what they do but it certainly adds value to the organization and indirectly to the product when that type of environment exists. It is easy to achieve with staff that are early in their career or with new work involving new technology, but is it sustainable? I am not certain but I do know it is easier for someone to value their work if they at least have input, if not some control over what they do, when they do it, and where. Open source projects maintain lists of needed items and allow contributors to pick what they will work on. It is not clear how this would play in a project with a tight schedule or a large amount of mundane work.

It is also easy to get people to value their work if they see that their work is valued by others. Many open source projects operate as a meritocracy. The project rewards value by promoting contributors to become committers and by giving committers more authority based on their contributions to the project. This sends the clear message that the work is valued more than the ability to get one's self invited to so many meetings there is no time to do any productive work.

6 SUMMARY

Value may accure by accident but not often. Professionals must intentionally seek to create value. Its easy for this pursuit to get lost in the day-to-day effort to meet deadlines and resolve issues. It is not an attribute that jumps out at you during development. It isn't usually visible of its own accord until the product is completed and the absence of value becomes all too apparent and all too difficult to to fix.

A software product line organization adopts certain practices because they create value in a variety of ways. Customer interface management is an obvious practice that, when handled effectively, can add real cash value to the organization. Structuring the organization to effectively manage the long term reuse and short term product production perspectives adds value more implctly by supporting a faster rhythm.
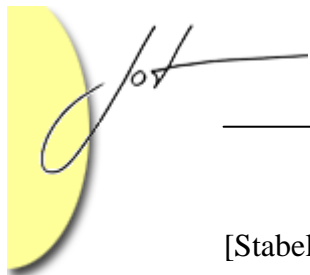
Value is everyone's business. A strategic, thoughtful approach to your work will accrue value for both yourself and your organization.

ACKNOWLEDGEMENTS

REFERENCES

[Chastek 06]   Gary Chastek and John D. McGregor. Guidelines for Developing a Product Line Production Plan (CMU/SEI-2002-TR-006).

[Chastek 07]   Gary Chastek, Linda M. Northrop, and John D. McGregor, "Observations from Viewing Eclipse as a Product Line", OSSPL07 Asia, 2007.

[Christensen 03]   Clayton M. Christensen, The Innovator's Dilemma: The Revolutionary Book that Will Change the Way You Do Business, Collins Publishers, 2003.

[Cloudsmith 07]   www.cloudsmith.com

[Hecker 07]   Frank Hecker. The Firefox value network, www.hecker.org, 2007.

[McGregor 07]   John D. McGregor. "Openness", in *Journal of Object Technology*, vol. 6, no. 6, July - August 2007, pp. 7 – 14, http://www.jot.fm/issues/issue_2007_05/column1.

[Porter 98]   Michael E. Porter. Competitive Advantage: Creating and Sustaining Superior Performance, Free Press, 1998.

[Stabell 98]   Charles B. Stabell and Oystein D. Fjellstad. "Configuring value for competitive advantage: On chains, shops, and networks", Strategic Management Journal, 1998.

About the author

Dr. John D. McGregor is an associate professor of computer science at Clemson University and a partner in Luminary Software, a software engineering consulting firm. His research interests include software product lines and component-base software engineering. His latest book is *A Practical Guide to Testing Object-Oriented Software* (Addison-Wesley 2001). Contact him at johnmc@lumsoft.com.