Journal of Operations Management 29 (2011) 376–390

Contents lists available at ScienceDirect

Journal of Operations Management

journal homepage: www.elsevier.com/locate/jom

Lean principles, learning, and knowledge work: Evidence from a software services provider

Bradley R. Staats[a,*], David James Brunner[b,1], David M. Upton[c,2]

[a] University of North Carolina at Chapel Hill, Campus Box 3490, McColl Building, Chapel Hill, NC 27599-3490, United States
[b] 27275 Byrne Park Lane, Los Altos Hills, CA 94022, United States
[c] Oxford University, Park End Street, Oxford OX1 1HP, UK

ARTICLE INFO

Article history:
Available online 25 November 2010

Keywords:
Lean production
Knowledge work
Learning
Operations strategy
Software

ABSTRACT

In this paper, we examine the applicability of lean production to knowledge work by investigating the implementation of a lean production system at an Indian software services firm. We first discuss specific aspects of knowledge work—task uncertainty, process invisibility, and architectural ambiguity—that call into question the relevance of lean production in this setting. Then, combining a detailed case study and empirical analysis, we find that lean software projects perform better than non-lean software projects at the company for most performance outcomes. We document the influence of the lean initiative on internal processes and examine how the techniques affect learning by improving both problem identification and problem resolution. Finally, we extend the lean production framework by highlighting the need to (1) identify problems early in the process and (2) keep problems and solutions together in time, space, and person.

1. Introduction

Lean principles, exemplified by the Toyota Production System (TPS), continue to greatly interest the operations community. Many credit Toyota's sustained success to their persistent and pervasive application of these ideas to manufacturing and management systems (Hino, 2006; Liker, 2004). This thinking has motivated many manufacturing companies to imitate, either wholesale or in part, lean principles in their improvement programs. While lean production has led to improved performance in many cases (Li et al., 2005; Shah and Ward, 2007), failed implementations are common, and as Shah and Ward (2007, p. 785) note, there is significant "confusion and inconsistency" in how lean production works and how it is best implemented.

In recent years, organizations have sought to apply lean production to knowledge work (e.g., Poppendieck and Poppendieck., 2003; Schutta, 2005). While almost all work consists of manipulating both physical goods and information, work referred to as "knowledge work" primarily involves the use of information (Drucker, 1999). The utility and impact of lean production in non-manufacturing contexts remain points of contention, leaving many managers to wonder if they are merely applying inappropriate and faddish ideas while others argue that lean principles have universal applicability (c.f. Sousa and Voss, 2001). In this paper, we ask two related questions: (1) Do principles of lean production apply to knowledge work? (2) How can we extend the existing framework of lean production to a new context that differs substantially from that in which lean was developed? To answer these questions, we report our observations and analysis of the application of lean production at Wipro Technologies, a large Indian firm competing in the global software services industry (i.e., custom software development).

In the following section we examine lean production in the context of knowledge work and identify this paper's contribution to the literature, before discussing the principles of lean production as identified in manufacturing in Section 3. Section 4 details our case study research design while Section 5 uses quantitative data from Wipro to examine the performance of lean software development projects as compared to non-lean software development projects. Projects at Wipro are the primary way that work is delivered to customers. A *non-lean* project is executed in a traditional manner, while a *lean* project is delivered using lean principles. Section 6 then qualitatively examines how Wipro's lean initiative changed the way that the firm operated. In Section 7 we discuss extensions to the lean production model while Section 8 offers concluding remarks.

2. Lean in knowledge work

Prior theorists note that implementations of lean production may vary across different manufacturing settings due to contextual

* Corresponding author. Tel.: +1 919 962 7343.
E-mail addresses: bstaats@unc.edu (B.R. Staats), dbrunner@hbs.edu (D.J. Brunner), david@upton.com (D.M. Upton).
[1] Tel.: +1 617 276 5492.
[2] Tel.: +44 0 1865 288800.

0272-6963/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.jom.2010.11.005

differences (de Treville and Antonakis, 2006). Knowledge work not only has a context separate from manufacturing, but also differs fundamentally in structure, calling into question lean principles' universal applicability. Shah et al. (2008) use a services context to document how unstable and uncertain demand does not preclude the use of lean principles. At least three more differences between manufacturing and knowledge work need to be addressed, however. First, knowledge work typically takes on a character more dynamic than that generally associated with manufacturing. Once Toyota begins producing a Camry sedan, they do not try changing it into a Tacoma truck halfway down the line if the customer changes her mind. In software services, however, such a problem is common as customers often change requirements during production. This task uncertainty is not limited to customer requirements; it can also arise from the underlying technology or external environment.

Second, knowledge work processes and their connections are often invisible. While in manufacturing it may be possible to see all pieces in a process and how they fit together, this is not the case in knowledge work. Important pieces are kept in individuals' heads or represented in symbols inside computers. While a software manager might want to view a software engineer's work in process, in many cases that is not possible until the work is complete or nearly so. Thus, process invisibility prevents problems from being identified early enough to be solved efficiently and effectively.

Finally, knowledge work is often a design task that spans high-level architecture to low-level details (Boone et al., 2008; Clark, 1985). In other words, while a manufacturing process might consist primarily of low-level exploitation (i.e., using the same knowledge and processes repeatedly), knowledge work may undertake both high- and low-level *exploration*, sometimes simultaneously (c.f. March, 1991; March and Simon, 1993). By high- and low-level exploration we are referring to where in the process architecture that exploration takes place. For example, in the context of manufacturing, while low-level exploration might involve changes at the level of an individual worker (e.g., a new sequence of steps for installing a part), high-level exploration may involve a complete redesign of the assembly process. High-level exploration is difficult since many flaws of an existing design (the output from high-level exploration) become clear only after low-level details are resolved. An architect might create an elegant high-level design; however, during low-level work (or worse, after that work is completed), a flaw in the design may be revealed. In the case of the now "Leaning" Tower of Pisa, the flaws of an inadequate design (e.g., a foundation too small for unstable ground) were not revealed until the tower's third story was added—five years into building (Encyclopædia Britannica, 1989).

Prior research on lean has been premised on notions that an established architecture will not change rapidly; that task uncertainty is low, facilitating task specification; and that workers can identify problems rapidly and accurately (Spear and Bowen, 1999; Spear, 1999). The higher degrees of task uncertainty, process invisibility, and architectural ambiguity that arise in knowledge work may therefore impede or even prevent application of lean principles to knowledge work.

This paper makes four primary contributions to the existing body of research. First, we identify challenges posed by using ideas from lean production in a knowledge work setting. Lean production is premised on the specification of both outcomes and behaviors (i.e., the work to be completed including tasks, connections, and architectures; Nidumolu and Subramani, 2003; Spear and Bowen, 1999; Spear, 1999). However, the lack of task repetition within software services obscures the degree to which tasks can be specified and then standardized. Second, we document, quantitatively, the applicability of lean principles to knowledge work. Our empirical analysis confirms that Wipro's lean software projects achieved higher, less variable performance than did a matched comparison set on most, but not all, performance outcomes. Third, we use descriptive analysis to examine how the challenges identified above were overcome. Our qualitative work illustrates how lean production improves both problem identification and problem resolution within knowledge work. The detailed specification of tasks, connections, and architectures that lean production requires (Spear and Bowen, 1999; Spear, 1999) creates opportunity to effectively deploy an iterative development model in this context. Fourth, besides offering us a chance to examine whether lean production principles apply to knowledge work, our unique context creates opportunity to extend the lean production framework.

3. Principles of lean production

Toyota's successful journey of more than 50 years to become the world's most profitable auto company (Taylor, 2007) is often credited to the company's manufacturing prowess resulting from the Toyota Production System (Hino, 2006; Liker, 2004). TPS was not the product of a single conceptual breakthrough. Rather, it developed gradually over many years as the accumulation of a series of small innovations (Fujimoto, 1999). Early on, Toyota leaders did not have the economies of scale enjoyed by Ford or General Motors and believed they could not attain these, so they tried to develop a system that they imagined Henry Ford might have used in their situation (Ohno, 1988).

In the early 1980s, as Toyota and other Japanese manufacturers made inroads into global markets, the call was sounded to study these Japanese companies in depth (e.g., Hayes, 1981). This led to books on TPS by its creators, as well as program launches to study lean principles at multiple universities (e.g., Clark and Fujimoto, 1991; Monden, 1983; Ohno, 1988; Womack et al., 1990). In an attempt to generalize the work of Toyota for other manufacturing settings, Krafcik (1988) coined the term "lean" to highlight the principles of limiting inventory and excess workers, or "waste", as opposed to other auto manufacturers' "buffered" approaches (Hopp and Spearman, 2004).

Despite significant study, the field has struggled with a lack of clarity about what lean production is and what it is not (Shah and Ward, 2007). Significant recent work has defined the "how" of lean production in manufacturing (MacDuffie, 1995; Narasimhan et al., 2006; Shah and Ward, 2003, 2007). Nevertheless, bundles of practices have not been identified outside of manufacturing. Therefore, we examine the Wipro implementation through the lens of prior work that has identified "lean" principles. This brief review does not survey the literature on lean production fully, but rather focuses on work done to identify its underlying principles (see Hopp and Spearman, 2004; Shah and Ward, 2007 for surveys).

Two early works on lean principles were Ohno's (1988) *Toyota Production System: Beyond Large-Scale Production* and Womack and Jones' (1996) *Lean Thinking*. Ohno, the principal creator of TPS, specifies two basic criteria characterizing lean production: just in time (JIT) and autonomation. JIT is a "pull" system in which production at each step begins only when signaled for by the customer downstream. To support JIT, Ohno developed the concept of a *kanban*, with six accompanying rules. Autonomation (sometimes called *jidoka)*, captures the notion of automation including a role for operators. The goal is not to eliminate production workers, but rather to focus them on aspects of the practice valued most highly.

The book *Lean Thinking* arose from work done by the MIT International Motor Vehicle Program (IMVP). While studying all aspects of the auto industry, the IMVP focused on automobile production. In addition to the prolific output of Womack and Jones, 1994, 1996, 2005; Womack et al., 1990), other key works associated with the IMVP include Krafcik (1988), MacDuffie (1995, 1997), MacDuffie et al. (1996), and Cusumano and Nobeoka (1998). In *Lean Think-*