Association for Information Systems
AIS Electronic Library (AISeL)

ICIS 2009 Proceedings

International Conference on Information Systems (ICIS)

2009

A Threesome Dance of Agency: Mangling the Sociomateriality of Technological Regimes in Digital Innovation

Fredrik Svahn
Viktoria Institute, fredrik.svahn@ait.gu.se

Ola Henfridsson
Viktoria Institute and the University of Oslo, ola.henfridsson@wbs.ac.uk

Youngjin Yoo
Temple University, youngjin.yoo@temple.edu

A THREESOME DANCE OF AGENCY: MANGLING THE SOCIOMATERIALITY OF TECHNOLOGICAL REGIMES IN DIGITAL INNOVATION

Completed Research Paper

Fredrik Svahn
Viktoria Institute
Gothenburg, Sweden
fredrik.svahn@viktoria.se

Ola Henfridsson
Viktoria Institute & University of Oslo
Gothenburg, Sweden
ola.henfridsson@viktoria.se

Youngjin Yoo
Temple University
Fox School of Business
Philadelphia, USA
youngjin.yoo@temple.edu

Abstract

In this paper, we develop a sociomaterial perspective for appreciating tensions between different technological regimes in digital innovation. Our case study research specifically looks at the tension between the deep-rooted component-based logic of two automakers' innovation practices and their attempt to introduce a new software architecture based on service orientation. Our evidence suggests that digital architectures need to materialize and be shaped in a dialectical way in the mangle of both existing regimes. We argue that the threesome dance of physical material agency, digital material agency and human agency can explain this finding and yield implications for our understanding of digital innovation in the traditional industries. Digital innovation is a result of a dialectical process, resolving various elements of resistance, subjection, and accommodation across the three types of agency.

Keywords: Innovation, Sociomateriality, Service oriented architecture (SOA), Software architecture, Software platform

Introduction

Computing and networking capabilities are of increasing importance to product innovation in industries such as avionics, automotive, and consumer electronics. This importance is not only manifested in increasing spending on digital technology over the last 20 years but also in the emergence of new business concepts turning physical products into digital products and services (Andersson et al. 2008; Barabba et al. 2002; Jonsson et al. 2008). The digitization of products, what we refer to as *digital innovation* in this paper, creates new waves of organizational, technical and cognitive challenges in organizations (Yoo et al. 2008).

In particular, the digitization of physical products entails inherent tensions between a long-established manufacturing paradigm that is hardware-based and an emerging software logic that is service-based (Andreasson and Henfridsson 2009). We trace these tensions to the co-existence of two heterogeneous innovation regimes (Godoe 2000) with different social structures and technical materiality (Orlikowski and Scott 2008, Pickering 1993). On one hand, the manufacturing paradigm is characterized by an innovation regime centered on *component-based modularity* (Garud et al. 2003). With physical components as a material basis, modularity has proved useful in the design of complex systems, such as cars or airplanes, by establishing interdependence within and independence across product components (Baldwin and Clark 1997; Ulrich 1995). Furthermore, it provides a template with which social structures with suppliers and sub-contractors are established. It further facilitates specialization and division of expertise in a hierarchical way. Such hierarchical control, combined with modular design principles, facilitates flexible coordination of loosely coupled business and sourcing relations (Sanchez and Mahoney 1996). To the contrary, *service-based modularity* is an innovation regime that seeks to unbundle software modules from physical constraints by promoting architectures that are independent from specific platforms (Allen 2006). Drawing on the materiality of software-based computing and communication capability (Zammuto et al. 2007), service-based modularity facilitates new forms of social structures based on coordination among distributed and heterogeneous actors. Generic platforms and reduced communication costs allow for integration of previously unconnected activities and artifacts (Yoo et al. 2008), tapping into a new source of creativity in product design (Yoffie 1997; Zittrain 2006). Together, computing and communication opens up for radical "reconfigurations of an established system to link together existing components in a new way" (Henderson and Clark 1990).

In this paper, we examine the tension between component-based modularity and service-based modularity as an important consequence of digital innovation in the area of car infotainment system. Appreciating that innovation regimes would not be developed, maintained, or modified without social practices (Godoe 2000; Nelson and Winter 1982), our analysis is geared towards the way that this tension plays out in the social practice of product developing firms. In doing so, we draw on a sociomateriality perspective (Orlikowski 2007; Orlikowski and Scott 2008, Pickering 1993) for analyzing the adoption of service-based modularity into innovation practices characterized by component-based modularity. A sociomateriality perspective enables us to see innovation as a continuing dialectic process through *resistance* and *accommodation*, i.e., a *mangling* process. In extant research, sociomateriality lenses have been employed to study the introduction of single technologies (Orlikwoski 2007, Pickering 1993, 1995). Our work extends earlier literature on sociomateriality by examining the tension between the two different materiality regimes and their interplay with human agencies as digitization of products unfolds. In addition, we note that digitization do not lead to a replacement of one regime for another. After all, despite increasing digital content, products such as cars, airplanes, and household appliances will always consist of physical parts. In other words, digital innovation entails the co-existence of dual sociomaterial practices, forming an emerging configuration of sociomateriality blurring the boundary between the physical and the digital.

We conducted case study research (Yin 2009) at two automakers, CarCorp and AutoInc, where a new service-based architecture, MOST (Media Oriented Systems Transport), was introduced to facilitate digital innovation in the area of infotainment systems. Through data collection methods such as semi-structured interviews, participant observations, thematic workshops, and document analysis, we explored how these two innovation regimes were clashing in the dynamics of sociomaterial practice of product innovations. Our study illustrates how the design vision, grounded in service-based modularity, confronted persistent resistance, grounded in the existing sociomateriality of component-based modularity, and how designers had to accommodate the materiality of hardware components throughout the design process. To resolve such tensions, software engineers had to set their visions of service-orientation aside, and find solutions that reasonably aligned with the dominant, component-based modularity. Dialectics between resistance and accommodation, then, is at the core of continuing evolution of digital innovations which emerge out of *impure* dynamics of mangling of two different forms of materiality and human agency (Pickering 1993).

The remainder of the paper is organized as follows. The next section provides an overview of component-based modularity and service-oriented computing as two competing innovation regimes. Then, we present sociomateriality as a theoretical lens with which we analyze the dynamics of sociomaterial practice in manufacturing. We then outline our research methodology, followed by a presentation of our case study of MOST in the automotive industry. Finally, we discuss some theoretical implications of the case study.

Tensions in Innovation Regimes

We refer to digital innovation as the new waves of organizational, technical, and cognitive innovation practices that follow the digitization of physical artifacts (cf. Andersson et al. 2008; Boland et al. 2007; Henfridsson et al. 2009; Yoo et al. 2008; Zammuto et al. 2007). In this regard, digital innovation can be positioned at the intersection of the technology management literature (e.g., Baldwin and Clark 2000; Garud et al. 2003; Henderson and Clark 1990; Murmann and Frenken 2006) and the IT innovation literature (e.g., Fichman 2004; Lyytinen and Rose 2003; Swanson 1994; Swanson and Ramillier 2004). This emerging research stream strikes a useful balance between the physical and the digital in approaching innovation.

To this end, this section briefly describes component-based and service-based computing as innovation regimes. Godoe (2000) defines innovation regimes as “principles, norms and ideology, rules and decision-making procedures forming actors’ expectations and actions in terms of the future development of a technology” (p. 1034). In this study, we see an innovation regime as an example of mangling process as it entails the materiality of technology being developed and the social practices in the development process. Following this focus on innovation regimes, our examination of these two forms of modularity is not intended to be exhaustive but to provide a basis for appreciating their underlying assumptions and influence on the product innovation process.

Component-Based Modularity

Product architecture is imperative to firms operating in competitive markets. Accordingly, the innovation literature has paid a lot of attention to different architecture types and their specific firm implications (see e.g., Baldwin and Clark 1997; Ulrich 1995; Ulrich and Eppinger 2003). One of the most influential architectures in manufacturing is component-based modularity (Garud et al. 2003).

Modularity has proved useful in the design of complex systems. In particular, it establishes interdependence within and independence across product components (Baldwin and Clark 1997; Ulrich 1995). It facilitates control over complex systems, allowing for concurrent design, and accommodates uncertainty (Baldwin and Clark 2000; Parnas 1972). In this vein, modular design facilitates product change and flexibility (Sanchez and Mahoney 1996), allowing incremental improvement of the product design over time. In addition, a modular product architecture constitutes a template with which social relationships with suppliers and sub-contractors can be enacted. It facilitates specialization and division of expertise in a hierarchical way. Such hierarchical control, combined with modular design principles, facilitates flexible coordination of loosely coupled business and sourcing relations (Sanchez and Mahoney 1996). Consequently, manufacturing firms use component-based modularity to establish horizontal loose coupling between components and vertical tight integration of the product design hierarchy. While components, and their inherent functionality, are managed autonomously by suppliers, system integration is strictly controlled by the manufacturer.

Enabled by component-based modularity, innovation processes typically follow a linear logic powered by waterfall models of product development (Boehm 1976; Royce 1970). Requirements are therefore gradually broken down according to the design hierarchy, reflecting a distributed nature of innovation. Business objectives, overall system topics, and significant functional properties, are managed by the manufacturer, while the design of components and detailed functionality is assigned to highly autonomous suppliers further down the hierarchy. While these remote ”islands of innovation” are in a subordinate position, it is clear that product-lead firms are highly reliant on their long-term generative capacity for securing competitiveness over time.

Service-Oriented Modularity

While component-based modularity has its basis in physical components, the unit of innovation in service-oriented computing is the software service. Such services are self-describing, open components that support rapid and low-cost composition of distributed applications (Papazoglou and Georgakopolous 2003). Rather than focusing on controlling the innovation process through decomposition, the main emphasis of service-oriented computing is agility. Service-oriented computing is frequently portrayed as a response to modern organizations' needs of speed, cost-effectiveness, accuracy, and flexibility when dealing with ever-changing demands (Allen 2006).

With agility as a value basis, the service-oriented innovation regime views a service, or a module, as a temporary solution that is scrutinized to continuous reassessment within a non-linear and relatively open innovation network. The service is a reusable unit of business-complete work (Papazoglou and van den Heuvel 2007) and its ultimate prosperity is determined by its ability to provide user value. Because of their relative independence of physical constraints, software services can be easily recombined to deliver new functionality and user value. In this regard, a service-oriented innovation regime enables sense-and-respond capability, releasing firms from the burden of predicting services of tomorrow. Achieving such agility requires a service-oriented architecture (SOA) that aligns technology with business goals (Allen 2006).

Two fundamental elements of SOA can be distinguished: structure and policy (Allen 2006). With service orientation the dominant structure is manifested in software, rather than hardware. Functionality is implemented as loosely-coupled software services, relatively independent from the underlying hardware and operating system. Important structural concepts of SOA are encapsulation, abstraction, reusability, composability (several services can be combined), autonomy and granulation (the service show functionality at a granularity recognized by the user). Largely, the structural principles of SOA are inherited from earlier paradigms, such as object-oriented design (Mathiassen et al. 2000) and component-based software engineering (Heineman and Councill 2001), but we also recognize many concepts from component-based modularity.

As pointed out by Huhns and Singh (2005), SOA also depends on how well services can be placed into a cohesive framework. In contrast to component-based modularity, where product design and organizational design are tightly inter-related (Sosa et al. 2004), service-oriented computing devises software policy to control the distributed computing environment. Such policy covers issues such as quality of service, design, sourcing and usage, and technology. For instance, sourcing policy concerns the way that services should be purchased and supplied. Important issues include whether services should be insourced or outsourced.

Sociomateriality

While the innovation regimes of component-based modularity and service-oriented computing are inherently different forms of modularity, physical products with embedded computing and communication capability cannot escape either of them. A significant challenge in digital innovation is, then, to explicate the co-existence of these two different forms of sociomateriality. As noted above, materiality of technology alone does not determine the fate of innovations. In addition to different material basis (physical components and software), different innovation regimes are characterized by different social structures that include heterogeneous norms and values that play out differently in innovation practices enacted by actors in the innovation process. These social structures are continuously reshaped through the interplay with technical materiality, forming the mangle of sociomateriality that underpins an innovation regime.

In order to understand the adoption of service-based modularity into innovation regimes characterized by component-based modularity, we therefore turn to the literature on sociomateriality (Orlikowski 2007; Orlikowski and Scott 2008). The sociomateriality perspective is a useful way of conceptualizing the interplay between human agency and material agency in organizational life. In particular, we pick up three interrelated concepts within this relatively broad body of literature. First, we have adopted the idea of *sociomaterial assemblages*. In synthesizing a stream of research dealing with the mutual interdependence between social and technical elements (see e.g., Barley 1988; Orlikowski 1992, 2000; Monteiro and Hanseth 1995; Rose et al. 2005) this concept underlines the *constitutive entanglement* of material and human agency. Drawing on new perspectives on agency in sociology, such as Actor-Network Theory (Callon 1986; Latour 1987), socio-technical ensembles (Bijker 1995), mangle of practice (Pickering 1995), relational materiality (Law 2004), and IS researchers' conceptualizing of the mutual entanglement of technology and human action (Jones 1998; Kallinikos 2006; Latham and Sassen 2005; Monteiro and Hanseth

1995; Orlikowski 2007; Orlikowski and Iacono 2001; Orlikowski and Scott 2008), the sociomaterial perspective opposes the ontological separation between people and technology as primarily self-contained entities that influence each other (Slife 2004). Instead, it highlights that technology is not ready-made, but shaped by humans situated in a network of relations and artifacts. Conceptualizing the relations between organization and technology through sociomaterial assemblages denotes that material agency and human agency are saturating each other to the extent that previously taken-for-granted boundaries are dissolved (Orlikowski and Scott 2008).

Second, we adopt the idea that materiality can be viewed as *performed relations*, rather than as pre-formed substances. While the constitutive entanglement of agency is a central element of sociomateriality, it gives poor guidance in understanding the role and meaning of material agency. Unlike humans, material agents are not capable of setting up goals that refer to non-existing future states and then seek to realize them. Still, the sociomateriality literature argues that we can take material agency as seriously as human agency, as long as we consider it *temporally emergent in practice*. The contours of material agency are never decisively known in advance, but have to be explored in practice over time (Pickering 1993). Therefore, material agency can be viewed as performed relations, emerging through impure dynamics that is "situated within a space of human purpose, goals, and plans" (Pickering 1993, p. 577).

Third, we turn to the *mangle of practice* (Pickering 1993) to understand how the interplay between material and human agency shape trajectories of change. In viewing materiality as performed relations embedded in sociomaterial assemblages, we have a model to understand material and human agency, as well as the interplay between them. With the notion of 'mangle', Pickering underlines the ambiguity inherent in this dialectical dance of human and material agency. He argues that "the trajectories of emergence of human and material agency are constitutively enmeshed in practice by means of a dialectic of *resistance* and *accommodation* (emphasis is ours)" – the mangle of practice. Here, resistance denotes "the failure to achieve an intended capture of agency in practice", while accommodation means "an active human strategy of response to resistance, which can include revisions to goals and intentions as well as to the material form of the machine in question to the human frame of […] social relations that surround it" (Pickering 1995, p.22). The metaphor of mangle conjures up "the image of the unpredictable transformations worked upon whatever gets fed into the old-fashioned device of the same name used to squeeze water out of the washing" (Pickering 1993, p. 567). At the same time, he underlines that there are aspects of human agency that has no material counterpart. In arguing that humans' ability to put intentions behind our actions is central to the 'mangle' he is breaking with the strict symmetry of actor-network theory. Our ability to set up goals and create models that refer to non-existing future states is central in the formation of accommodation strategies.

The digitization of car infotainment systems through the introduction of service-based modularity into traditionally hardware-oriented product development is not done by coincidence. It is a deliberate attempt to change an existing practice, aiming for specific strategic goals. At the same time, we see rich evidence that this process is characterized by significant ambiguity and unpredictability. Original goals tend to be significantly translated and delineated in the clash with an existing sociomateriality, i.e., the component-based innovation regime. In this paper, we adopt sociomateriality as a lens to understand the tensions between these two innovation regimes that are involved in digital innovation. We argue that component-based modularity and service-oriented computing can be seen as distinct material agencies, formed in different social contexts. In the automotive industry, component-based modularity is deeply mangled with centrally controlled social practices. A pressing issue for firms engaged in digital innovation is how to introduce a foreign form of materiality into this existing mangling of sociomateriality.

In pursuing the idea of digital innovation, we expand the idea of the mangle of practice by noting how one form of material agency subjects itself to human agencies as it competes with another form of material agency in digital innovation. We propose a perspective of digital innovation that builds on this threesome dance of agency. In the following sections, we present a case study of two automakers, AutoInc and CarCorp, and their attempt to appropriate a service-oriented technology called MOST for promoting agility and flexibility in their development of infotainment systems. The case study will demonstrate the challenges involved in developing co-existing, multiple practices within the same organizations and how these challenges can be traced to materiality.

Research Methodology

We conducted case study research at AutoInc and CarCorp, which are manufacturing firms in the automotive industry that develop, produce, market, and sell cars on the global market. In distinguishing between different styles of researcher involvement in case study research, we were "involved researcher" rather than "outside researchers"

(Walsham 2006). In this regard, our research was interpretive (Klein and Myers 1999; Walsham 1993) in nature and focused on the interplay between actors and technology over time (Langley 1999; Markus and Robey 1988).

There were two reasons why this approach to the study of sociomaterial practices was considered useful. First, a sociomaterial lens requires a methodology that focuses on the activities, behaviors, and events through which actors interpreted the new technology over time (Langley 1999). Earlier studies of sociomateriality (see e.g., Pickering 1995), as well as other forms of practice research (e.g., Schultze and Orlikowski 2004), have adopted this kind of case-near research. Second, because sociomateriality still is a relatively unexplored research topic, a revelatory case study (Yin 2009) can be a powerful way of illustrating the concept and its implications for IS research.

Our data collection can be described as a two-step process. First, data was collected between 2002 and 2003, when the first author conducted participant observation of AutoInc's adoption of MOST. Apart of participant observation, one important data source was technical specifications. This early data collection was later complemented with 7 interviews involving key personnel from AutoInc's MOST project. The initial study at AutoInc guided our interest towards tensions between technological regimes in digital innovation. Second, to explore the nature of these tensions, a two-year (2008-2009) intensive field study was conducted at CarCorp. It generated significant data on sociomaterial practices enacted in the appropriation of MOST. Our data collection at CarCorp included 27 recorded and transcribed semi-structured interviews, ranging between 40 minutes and 2 hours. Respondents covered different roles at CarCorp, ranging from managers to developers, and they covered expertise such as software, architecture, graphical interfaces, ergonomics, design, and market. Furthermore, more than 50 work meetings at the infotainment department were attended. These meetings were documented by taking extensive meeting notes. We also organized 5 thematic workshops, focusing on different architectural challenges. In addition, we studied the 30 most central specifications from CarCorp's original MOST project.

The various data sources were repeatedly read and coded with an open-ended approach to identify key themes from major events, activities, and technology choices that emerged over time (Langley 1999). At a relatively early stage, modularity came out as a central concept in understanding digital innovation and the different contradictions following on the clash between technological regimes. Following Strauss and Corbin (1998), we revisited the data, doing selective coding, to verify observations and develop a theoretical understanding of the mangling between two different forms of materiality.

MOST at AutoInc and CarCorp

MOST as a Solution to Component-Based Modularity Concerns

At the turn of the century, AutoInc and CarCorp became increasingly aware that the growing number of infotainment functions challenged established development practices that were anchored in component-based modularity. The hitherto loose coupling between components did not fully make sense anymore, when a wide range of co-existing applications required the same basic resources. Speakers, displays, controls, and various sensors simply had to be shared over the full range of infotainment applications to support customer satisfaction and economy of scale.

Following the deep-rooted logic of modularity AutoInc and CarCorp started to define groups of components to hide the increasing interdependence within sub-systems. While initially boosting functional growth, this strategy soon turned into a burden for the automakers. With modularity being a central element in enforcing hierarchical control over suppliers and sub-contractors, CarCorp and AutoInc largely found themselves being in the hands of the suppliers. Amplifiers, radios, CD players, etc, remained separate physical entities, but highly intertwined through various proprietary and largely unknown networks, protocols, and harnesses from a few major suppliers. At both firms, R&D staff perceived decreasing control of system design, product planners of upcoming functionality, and purchasers of the sourcing process. The rapidly increasing coupling simply had to be addressed in a new way to reclaim control and secure future growth. At the turn of the century one industry-wide initiative had achieved enough momentum to be hailed as a solution to these problems: MOST. A senior AutoInc systems architect recall the early discussions promoting MOST as an interesting general purpose network concept, supporting the domain specific requirements and thereby further growth:

> *We all saw the transformation of infotainment. It was a remarkable change, and growth, and new lifecycles of the products. We needed an infrastructure to support this. MOST was [already] selected by*

> *BMW, with others talking about it. Somehow it should support this domain, with needs beyond body [electronics] and powertrain.*

The adoption of MOST was preceded by investigations and debates at the two auto manufacturers. At AutoInc an official pre-study was launched, in the form of an advanced engineering project. CarCorp adopted a less formal approach, evaluating the technology on the basis of a technology review. Overall, at both companies three different perspectives emerged from the process; *MOST as architecture*, *MOST as standard*, and *MOST as a business driver*.

MOST as Architecture

In the mid 90th AutoInc and CarCorp successfully established the controller-area network (CAN) as an in-vehicle communication backbone. A wide range of artifacts (sensors, actuators, computing, etc) were now interconnected via this dual wire serial bus. However, with its relatively low bandwidth, the CAN network quickly became a limiting resource. The growing number of infotainment components, requesting network capacity for audio, graphics, video and human-machine interaction (HMI), could not benefit from the CAN concept. Instead various *ad hoc* solutions emerged, often designed by suppliers for a specific purpose or project. With a growing number of suppliers and components the overall system solution became inherently complex and hard to manage. The various sub-systems simply did not fit well together.

The technology reviews suggested that MOST offered promising solutions for the almost desperate need of *structure and reduced complexity*. A technical fellow and later acting project manager for AutoInc' MOST project explain the motives:

> *It was all about bringing things together. To get control [data], signals, audio, and, as we expected, also video into the same bus concept. This in contrast to a mess of different harnesses and cables. It would have simplified the system dramatically, as with a computer you are plugging into the wall. You don't have one network for control signals, one for streaming audio, and one for streaming video. You've got ONE Ethernet connector.*

Another consequence of the heterogeneous, supplier driven architecture was its inability to support change and modifications. Component changes often brought costly and time-consuming system level modifications. Therefore, systems architects identified *flexibility and modularity* as important primitives of an upcoming infotainment architecture. A senior AutoInc architect reflects back on the arguments at the time:

> *It [MOST] will come [on the market], and we need to approach it, prepare ourselves in order to get access to such flexibility – that is probably an interesting concept here – to be able to produce information anywhere in the car and consume it somewhere else, in a simple way.*

A third architectural motive presented in favor of MOST was *capacity*. In extrapolating the functional growth of the 90th, bandwidth was standing out as a critical issue. CarCorp's former MOST project manager recalls that:

> *We had remarkable ambitions. We planned for video screens in the back seat and support for external video sources, delivering services such as park assistance. It should be a pretty high level of functionality. And when we looked at the different things customers should be able to do concurrently – it was a concept work I guess – we found that CAN wouldn't do. We needed a really powerful bus concept to survive that. It should be able to support graphics, while simultaneously transmitting a burst of navigation data.*

MOST as a Standard

The work of the industry-spanning MOST consortium aimed for standardized technology, from the lowest physical layers all the way up to the application layers. Dedicated hardware, middleware for network management and an extensive application framework should secure a common – and thereby compatible – approach to vehicle infotainment. Considering a reflection from the acting AutoInc MOST project manager, it is clear that standardization was an important argument in promoting MOST:

> *This idea about common specifications on functions and interfaces, that's a major benefit. More or less being able to buy a component [off the shelf], like a radio tuner, developed for one manufacturer, but applicable to another since it's a common interface specification.*

Significant adoption of a standardized technology was considered a great potential, not least in the sourcing process. With the traditional, proprietary system solutions CarCorp and AutoInc could possibly benefit from competition at the time of sourcing, but not over the product life cycle. Major investments in systems integration effectively prevented re-sourcing of components, causing lock-in effects where the manufacturer had no option but to stick with existing suppliers. With standardized components CarCorp and AutoInc saw a potential to dramatically increase competition, with lower thresholds for re-sourcing.

MOST as a Business Driver

The third perspective falling out of the technology reviews referred to MOST's ability to transform business in scope and form. The inherent ability to support service orientation played a minor role in this discussion. Instead of seeing software in itself as a potential revenue generator, this stream of proponents suggested that MOST would allow modern, distributed software development to fit into the established paradigm of component-based modularity. Together, the generic fiber-optical bus and the object-oriented, event-driven application framework were expected to give a healthy separation between hardware and software. Suddenly, it was possible to see the increasingly problematic issue of integration through the lens of software, rather than costly and inflexible hardware structures.

With frustration CarCorp and AutoInc's product planners had observed how these increasingly monolithic hardware structures destroyed attractive business models. Instead of an open option list model, enabling customer unique combinations, these interdependent systems forced the auto makers to bundle functionality in a few predefined offerings. In addition, such solutions complicated the lucrative aftermarket business. With tight, physical integration between components, it was more or less impossible to extend or change the configuration over the vehicle's life cycle. It was argued that MOST would break up the monoliths, allowing unlimited variations in functionality through a wide range of specialized, independent components.

In summary, this perspective suggested that the MOST concept could be applied as to support the traditional component-based modularity. Extensive bandwidth, highly specialized and independent components, and software enabled functionality would secure future *growth and diversity* for infotainment functions, both as options (built-in at the time of production) and accessories (aftermarket extension).

Adopting MOST

With the decisions to adopt MOST as the new infotainment backbone CarCorp and AutoInc entered a rather painful path, unfolding in the clash of technological paradigms. A potentially service-oriented approach – characterized by a combination of extensive, generic communication capacity and object-oriented, event-driven computing – was about to be applied in a domain based on component-based modularity. The materiality of digital, software-based product development was confronted with solid materiality of hardware-centric structures, which is deeply ingrained into the very sociotechnical fabric of the existing organizations and products. As illustrated by AutoInc's project manager, the new technology brought a wide range of challenges, from technology design to processes and organizations.

> *We had no idea what we were about to engage in… We simply bit off more than we could chew. First, introducing a new bus [concept], then taking systems responsibility – previously allocated to an external supplier – and, finally, making it a lot more complex through distribution. It was a major challenge.*

As these various challenges unfolded, the automakers initiated a range of actions. First, they had to acquire and *develop new knowledge*. Second, the designers found existing design hierarchies suboptimal in supporting the new technology. Therefore, as they developed basic knowledge, their attention was gradually shifted toward the establishment of *new forms for collaboration* within their own organization and towards suppliers. Finally, with many critical tensions resolved, CarCorp and AutoInc started to set up *new design practices* for MOST-based infotainment.

Developing New Knowledge

The MOST concept brought several new technologies with impact on component design. For example, the novel integrated circuits that enables access to the optical network were not yet stable, thus causing major trouble to both manufacturers. Although learning how to manage fiber optics in an automotive context was demanding, the critical challenges were related to architectural design, rather than component design. With MOST, the notion of architecture became blurred to designers, and gradually loaded with new meaning. The traditional rationale behind architectural work – hiding complexity, division of labor, reuse, etc – was extended with a new, partly incompatible logic. With software-based functionality distributed over several physical components, other properties became salient. The new infotainment architecture became an *enabler of functionality,* largely defining the shape and form of this distributed computing environment. Systems architects turned into platform designers. The architecture came to manifest a design philosophy and generic system level services, rather than the structure of components. Although this transition was highlighted in the original MOST concept, the auto makers underestimated the challenges of discovering, understanding, and implementing this design philosophy. A senior systems architect at CarCorp remembers his disappointment, when discovering that the architectural concept was far from solid:

> *They [MOST cooperation] promoted MOST as a new system-level model, a new kind of thinking, a new philosophy for design. But this model was never written down. It was BMW and Becker running it, but not in public. […] we could see how it was designed, I mean the result of the MOST interface definition, but we never understood the [deeper] thinking, and how they intended to evolve it. That made many of us, implementing at the time, doing extensions of our own, tweaking around, and creating solutions which probably did not align with the visions.*

On a general level, systems architects and designers were trapped between two different materialities. On the one hand, they had to adopt a more service-oriented approach to infotainment development. There was a consensus among engineers that the established component-based modularity would not be able to secure future growth for this family of increasingly changing applications. On the other hand, they were still embedded in a product development context that is tightly entangled with hardware-centric component-based modularity. A massive body of existing requirements was derived on the premises of this sociomateriality of component-based modularity. Further, both suppliers and the auto makers' own purchasing were reluctant to adopt software-driven business models. So were the product planners, showing marginal interest in software as a future revenue generator. With such a range of path-dependent forces, the lack of clear and unambiguous design vision became highly problematic. Architectural knowledge and new practices materialized with a bottoms-up logic, driven by designers' local problems and challenges, rather than top-down, guided by strategic management. One such driver was an increasing awareness that existing processes gave limited support to the emergent forms of collaboration between different actors in existing design hierarchies.

Seeking New Forms for Collaboration

The automotive industry's component-based modularity, refined over a hundred years, is essentially tightly intertwined with strict hierarchies both in product and organization structures. Product structures are hierarchical, with horizontal independency between components. In the same way, organizations are hierarchical, dividing relatively independent branches of labor. In order to manage such design hierarchies CarCorp and AutoInc followed strictly linear innovation processes, with a dynamics powered by waterfall models (Boehm 1976; Royce 1970) of product development. In practice, requirements were gradually broken down alongside the design hierarchy, reflecting a distributed nature of innovation. Business objectives, general system topics, and overall functional properties were managed by the manufacturers, while the design of components and detailed functionality was assigned to highly autonomous suppliers, further down the hierarchy.

With the introduction of MOST, it soon became obvious that these remote "islands of innovation" did not perform anymore. Still being trapped in hardware-centric sociomateriality, functionality spanned several physical components and, thereby, suppliers. The automakers saw no other option than bridging the gap between suppliers themselves by specifying not only interfaces between components, but also the system level behavior of component-spanning functions. As illustrated by a project manager at CarCorp, this approach had significant implications for the collaboration within established hierarchies:

> *You are taking a [new] responsibility as a manufacturer, when specifying this stuff. It becomes… I mean, they [suppliers] CANNOT even do anything! When I think about it, it's not them rejecting responsibility, it's us taking it from them. Yes, that's what it is. We are telling them that "the only thing you're about to do is to support this [our solution].[…] Earlier, when things were more component-oriented, they had an opinion of their own on things, they had tested it – possibly with other*

> *manufacturers – and knew what was good and what was bad. With this approach [MOST] we more or less lost such feedback.*

These problems were grounded in an emerging and fundamental mismatch between the existing social structures and the emerging materiality of service-based modularity. That is, as engineers try to follow service-based modularity, they had to increasingly background the physical hardware, while focusing on software-enabled functionality that span several components. At the same time, the same engineers remained organized according to the component-based modularity.

Knowing that this mismatch could not be easily resolved, the two auto makers initiated two different measures to smooth the implementation of a MOST-based infotainment solution. First, they reorganized the workforces at a local level to meet the new commission. At AutoInc infotainment managers decided to replace the two existing component subunits with four new units. In the original organization, audio components were allocated to one subunit, while telematics, navigation and telephony were handled by the other. The management realized that the conception of component was less important with the new technology and architecture. Therefore, in the new structure, an increased need for system level control was met by a new subunit, responsible for system and functional design. This subunit hosted existing roles, as well as new ones. For example, an entirely new role – labeled “infotainment complete” – was created to manage a growing number of generic services, such as resource management and application coordination. The other new unit was created in response to the increasingly challenging task of testing. Distributing functionality over various components not only increased the complexity in interfaces among components, complicating integration, but also redefined the meaning of component testing. Behavior of functions could no longer be validated on a component level, leaving this task to the systems owner – the manufacturer.

Although CarCorp did not implement AutoInc's formal change of the organization, they also went through very similar experiences, albeit somewhat informally. The acting project manager for MOST industrialization reflected on this topic:

> *Originally, it was a component-oriented group. They were expected to work with functional specifications as well. Later on, this didn’t work out, so they invited some people working with functions only. They needed more and more such people and, eventually they were a group of their own. Probably 10-12 [persons], maybe even more. Most of them were consultant since it was running so fast, and we wanted it implemented. We underestimated the efforts significantly*

Taken together, rather than obliterating the hierarchical structure, the two manufacturers rebalanced the workforce, with old roles and levels of the hierarchy essentially remaining the same, while the locus of design activities moved upwards in the hierarchy, from the component level to the functional level.

Second, designers needed to break with the strictly linear models of innovation. The new situation pushed new forms for collaboration and new relations – some temporary and some more permanent – between actors that were not supported by the official hierarchy. Moreover, with functionality becoming a system-level issue, it was necessary to adopt iterative approaches to innovations. While the official development processes stated very few recursions, each resulting in the production of a pre-series car, the new way of designing infotainment seemed to call for an endless series of iterations. While the reorganization was formally approved by management, solutions to these challenges emerged bottom-up, from designers’ daily need to make progress in the development process. When specifications were ambiguous to suppliers, workshops were initiated with relevant stakeholders. When supplier implementations failed due to various misconceptions, the automakers built extensive system-level test environments to identify and solve problems collectively. When progress was too slow, the number of iterations increased dramatically, sometimes exceeding one software release a week. Such figures are in stark contrast to the official development process, stating just a handful of releases for an entire 3-4 year car project.

The introduction of service-based modularity led to the emergence of a mixed form of sociomateriality. On the one hand, traditional sociomateriality based on hierarchies and component-based modularity remained. Formal specifications were written, broken down to a component level and, eventually, sourced to various suppliers according to existing principles. On the other hand, much of the critical work was performed in a fluid structure of more or less temporary, cross-organizational design teams. Relations between actors and arenas for collaboration were established and destroyed according to project needs. Together these informal teams and processes made up a network-based model for innovation, augmented to the formal hierarchy.

Balancing these two, partly incompatible sociomaterialities was highly challenging to designers and architects. To support the network-oriented daily work, the auto makers had to create new design practices, improving the collaborative visibility. At the same time, to enforce the formal hierarchies they had to find new practices for the deployment of the growing functional designs to physical components.

Setting up New Design Practices

Systems architects at CarCorp and AutoInc had studied new design practices from the software industry even before the introduction of MOST. Since they had already seen increasing interdependencies with the low bandwidth CAN networks, they were attracted by the ideas behind service-based modularity design and the ontological separation between software and hardware. With the decision to adopt MOST technology, bringing object-orientation and event-driven design, such ideas became legitimate and apparently useful.

First, the two auto makers revised their definitions of architecture. In the architecture specification for the new infotainment system, CarCorp developed the notion of components, now referring to them as either logical entities or physical nodes. On the basis of this extended notion, they defined architecture “as the structures of the components of the system, their interrelationships, and principles and guidelines governing the design and evolution over time”. In contrast to prior architectural approaches which more or less addressed the decomposition of systems in independent parts, this definition significantly changed the locus of architectural work. In including the dynamics of interconnected components and principles for development, it made system architecture a matter for designers in their daily work.

Second, with the logical view of system design in place, both CarCorp and AutoInc adopted new CASE tools supporting a model-based approach to system design. Both firms decided to use the unified modeling language (UML) as a basis for modeling. However, while CarCorp focused on the cognitive challenges, using Microsoft Visio to capture, describe and illustrate the complex system, AutoInc wanted to take modeling one step further. They adopted the more extensive Rose Suite from Rational in order to get better support for the deployment of logical designs on physical components. This tool enabled AutoInc to generate component level specifications and interface specifications automatically from the logical designs. Consequently, the role of component engineers transformed radically. Their prior role, interpreting information and compiling specifications, was essentially reduced to editorial work, including various non-functional requirements. Therefore, AutoInc’s approach to modeling supported not only the cognitive aspects of system design, but also the more organizational challenges of rebalancing the workforces.

Finally, the two manufacturers spent considerable time and energy in trying to establish and spread various design patterns (cf. Gamma et al. 1995) across the formal and informal organization structures. One such key strategy was founded on the architectural pattern model-view-control (MVC). Here the logical *model* objects corresponded to basic functionality, such as navigation routing or digital music decoding. *View* objects implemented the user interface, while *control* captured the dynamic properties. Along with this central idea, both auto-makers implemented the observer pattern (sometimes labeled publisher-subscriber) to facilitate event-driven interaction between an increasing amount of distributed objects. Basically, this pattern identified controllers and views as subscribers of events at the models, creating a hierarchy between objects. As described by CarCorp in its architecture and design strategy, these approaches aimed for an important isolation between user interface issues and basic functional issues:

> *The infotainment system is a user interactive and user intensive system (application) with continuously changes in the user interface but with core functionality that in some degree is defined as stable. Therefore it is a good idea to split the core functionality from the user interface.*

Although resolving some major issues of complexity and reuse of generic functionality, this approach created new challenges at the time of deployment. The formal hierarchies preserved the physical component as the central unit of design, manufacturing and sourcing. With a service-oriented business model beyond reach, the auto makers saw no option but to deploy the model objects on various remote components, while view and control objects were allocated to a few central components. Generic navigation or media functionality were sourced to dedicated specialists, while their respective user interface were centralized to coordinate user interaction (Broström et al. 2006) and aligned look-and-feel. AutoInc even considered the control over user interfaces critical enough to develop the infotainment control module (ICM) in-house.

As a consequence, this deployment strategy broke more or less every function apart in two non-trivial units, boosting distribution one step further. The system became highly interactive in terms of network communication, causing major system level problems with latency, timing, etc. Most of the problem seemed to derive from the boundaries raised at deployment. Suppliers simply did not understand each other and made different interpretations of the same information.

Consequences

Over a rather painful period of approximately 2 years, CarCorp and AutoInc revised their architectural knowledge, established new forms for collaboration and launched new design practices. These initiatives had significant impact on the three dimensions of MOST as architecture, standard, and business drive.

First, the MOST architecture significantly challenged and transformed the existing architectural approach at CarCorp and AutoInc. It established the logical view on systems design, releasing significant power of software, until then bounded to independent physical components. It changed the rationale behind architecting, making platform designers out of architects.

Second, and perhaps most obvious, the idea of MOST as a standard faded away as soon as the designers were squeezed between an immature concept and established designs. Extremely tight project schedules did not leave the option of taking problems back to the MOST cooperation for reconsideration. Instead, CarCorp, AutoInc and most other auto makers implemented proprietary solutions on top of the core MOST concept. Off-the-shelf components did not emerge at the time of the first MOST projects, nor later.

Finally, with MOST as a business driver, the outcome is somewhat more subtle and ambiguous. The new infotainment systems became a lot more distributed, in sense of physical components with dedicated functions. These remote components were definitely less interconnected physically, more or less only reliant on power supply and a fiber optical cable. However, logically they were highly interdependent with a few central components, managing system properties and user interfaces. Although slightly increasing the configurability at the time of car sale, this rather centralized solution effectively restrained the anticipated after-market business.

Discussion

In this paper, we seek to understand the mangle of two different forms of materiality with human agencies during the digital innovation process. Specifically, we examine how the existing sociomateriality established with the materiality of hardware-centric component-based modularity respond to the designers' attempts to introduce a new form of materiality, based on software-centric service-based modularity. Consistent with the core concept of sociomateriality, our case study shows that technological change cannot be foreseen or understood unless seeing technology and social structures as a whole. Although MOST in itself has every hallmark of a truly service-oriented technology, it did not feed a service-oriented infotainment business for CarCorp or AutoInc. Social structures of these two organizations are deeply entangled with the materiality of component-based modularity. The established sociomateriality resisted engineers' efforts to introduce service-oriented modularity. Consequently, engineers had to accommodate the principles of MOST evolved through a continuous compromise and adaptations of materiality, creation of new design practices, and establishment of new social structures. This mangling process led to the hybridization of materiality, reflecting both existing, familiar practices and the selective elements of MOST technology. As a technology, the new MOST concept became local, rather than global over the entire organization, and had to emerge in some harmony with existing sociomaterial practices, rather than replacing them. Although infotainment is a high profile application area visible to end-users, it makes only a minor element of a car. In such an embedded context, designers may seek software-centric architectures that can improve system performance or the execution of a local application's functions – protocols, signal processing, user interface, and so on – but only while meeting all performance and resource requirements (space, weight, power consumption, electromagnetic compatibility, etc) of the overall system (Wolf 2002). Essentially, these non-functional requirements are defined by the existing physical materiality of hardware.

This tension between a hardware-centric, physical materiality and an emerging software-centric digital materiality is present throughout the entire process of adopting and appropriating MOST. Indeed, there are frequent elements of gradual learning and dialogue in our case story. Still, as exemplified by table 1, almost any central decision can be traced to a series of mangle of two forms of materiality and human agency in the form of resistance and accommodations, playing out at three different layers in product development: architecture, design, and organization. First, MOST introduced a radically different approach to *architecture*, focusing on the structures of software-based logical elements, rather than physical components. The traditional architectural approach, grounded in component-based modularity, aimed for decomposition of the product in a hierarchy of relatively independent modules. This enabled the automakers to exercise control over the product through strict supervision of the interfaces. At the same time, it encouraged innovation by allowing for extensive autonomy within components. In contrast, the new architectural concept, grounded in service-oriented modularity, aimed for a decoupling between software structures and hardware structures. It promised a functional design practice without strict encapsulation in components. CarCorp and AutoInc would be able to exercise control by taking a more active role in logical design, although leaving physical design to tier-1 suppliers. It was also expected to encourage architectural innovation by giving the designers access to heterogeneous resources over the entire system. The main challenge, also making a significant force of resistance, turned out to be the act of deployment. With component-based modularity remaining the dominant architectural model for vehicle design in general, it kept making a template for supplier relations also in the area of infotainment. Given that suppliers were contracted to deliver physical components, not software, engineers feared that the new concept would bring a counterproductive separation of distributed functions, for implementation by different suppliers, at the level of components. To preserve reasonable agility promised by the logical architecture, the automakers accommodated this resistance through a platform approach to architectural design. The system, as a whole, was designed to enable generic, non-functional services, such as service discovery, resource management or application coordination. While accessible over the entire system, these central elements were allocated by architects to one (AutoInc) or two (CarCorp) strategic components, under extensive control by the auto makers. Although software implementation essentially remained a task for component suppliers, this platform concept allowed the manufacturers to easily reallocate functions across components.

Second, with the platform philosophy in place, the challenge of balancing digital dimension with physical was shifted to the level of *design practice*. With a traditional, component-based innovation paradigm, designers are trained to keep the interdependency between component low. Therefore, a superior objective is to establish local encapsulation of functionality at the level of components. However, service-orientation encourages designers to build distributed functionality, allocated across components. With this architectural approach, they can take a system-level perspective on design, where allocation is guided by the need for simplicity, performance, information, etc. While systems architects met resistance in the act of deployment, the designers saw a related, potentially massive problem in the transition towards a logical design practice. Relying on deep-rooted organizational structures, where suppliers are contracted to build components, not software, they predicted a severe cognitive challenge. Essentially, this resistance developed in the fear that coherent system-level designs were running the risk of being broken apart in abstract pieces, more or less incomprehensible on a component level. Addressing such misgivings, CarCorp and AutoInc accommodated by introducing model-based design methodology, supported by new CASE tools. They also launched several new design patterns to guide designers in their modeling work. In addition to their normative effects on practice, these efforts injected invaluable insights across the entire design hierarchy, giving all actors a better understanding of the interrelations between system-level designs and component-level implementations.

Finally, the transformation of architectures and design practices highlighted that the experienced resistances derived from a mismatch between *organizational structures* and the emerging product structures. Essentially, the established, component-based innovation regime is built on the concept of hierarchy, where entities on the same level of aggregation are largely independent. At the same time a component has a well defined fit into the next level of aggregation. Over the years it has proven highly efficient to reflect this structure also in the organization. Therefore, manufacturing firms tend to feed innovations linearly through a vertical hierarchy. A service-oriented innovation regime, on the other hand, put technological change and organizational agility in focus, rather than specialization, expertise, and division of labor. Through loosely connected networks of internal and external contributors it offers the manufacturing firm a non-linear innovation model, better suited for the exploitation of heterogeneity and multiplicity across horizontal layers. While engineers had more or less unrestricted authority over the local infotainment architecture, they were not able change the organizational structures. Sourcing was a matter for central purchasing departments, and business models were centered on the idea of selling components. Trying to establish a radically new organization, applicable to the local domain of infotainment was not only impractical, but virtually hopeless. Accommodating this resistance, the infotainment managers decided to rebalance their own formal organization, instead of breaking it up. While roles and hierarchy structure essentially remained untouched, the locus of design activity moved upwards in the hierarchy. As CarCorp and AutoInc got more involved in system-level designs, resources were transferred from the component level to the functional level. In addition to such top-down initiatives, launched by management, designers built informal innovation networks, augmented to the formal organizations. These bottoms-up initiatives emerged as a response to their need to reach beyond the local organization. Unable to reform supplier organizations, they simply enforced a new, organization-spanning design practice, where engineers from different organizations worked together, side by side.

Table 1. Resistance and accommodation in the mangling of sociomaterial practice at CarCorp and AutoInc.

	Logic of the established sociomateriality	***Promise of the emerging sociomateriality***	***Resistance***	***Accommodation***
Architecture	With traditional, physical architectures, based on component-based modularity, CarCorp and AutoInc are able to exercise product *control through strict supervision of the interfaces*. It also encourages *component innovation* by allowing for extensive autonomy at the level of suppliers.	A logical architecture, based on service-oriented modularity, invites the automakers to exercise *control by taking a more active role in logical design*. It also encourages *architectural innovation* by giving the designer access to heterogeneous resources across the system.	The emerging logical architecture met significant resistance in the act of *decomposition*. Architects feared that the agility would be lost as logical designs were decomposed to fit a traditional physical architecture, with several components, delivered by different suppliers.	To preserve agility, while still avoiding functional fragmentation, the automakers introduced a *platform approach* to architectural design. Containing service discovery, application coordination, etc, the platform enabled rapid reallocation of functions across components.
Design	Design practices developed under a component-based innovation paradigm seek low interdependency between components. Therefore, functional designers tend to strive for local *encapsulation* at the level of components.	Design practices developed under a service-oriented innovation paradigm encourages *distribution* of functionality, across components. In doing so, it opens up for a system-level perspective on design, where allocation is guided by the need for simplicity, performance, information, etc.	The emerging design practice was resisted in that designers envisioned massive *cognitive difficulties*, across design hierarchies. They feared that coherent system-level designs would be broken apart in abstract pieces, more or less incomprehensible on a component level.	To resolve some of the cognitive problems, yet adopting a system-level, logical design practice, the automakers introduced *model-based design* principles and generic *design patterns*. These tools gave a common overview and valuable insights across the organization-spanning design practice.
Organization	A hierarchical organization structure enables a *linear model of innovation*, where information flows vertically. This allows for specialization and division of labor, being critical factors in a component-based innovation paradigm.	A networked organization encourages technological change, rather than specialization and expertise. Such loosely connected networks of internal and external contributors allow for *non-linear innovation* models, able to cater for heterogeneity and multiplicity across horizontal layers.	The networked organization was resisted in that local infotainment needs clashed with firm-level organizational strategies. Infotainment managers had extensive authority over technology, but little influence on purchasing or business strategies, tuned for component-based innovation.	Failing to change the firm-spanning organization structure, infotainment managers launched a *rebalancing of the local organization*. In addition, designers built *informal innovation networks*, augmented to the formal design hierarchies.

The resistances of physical materiality of component-based modularity at each layer are central in developing new sociomaterial practices, embracing selective elements of digital materiality of service-based modularity. Although designers are constrained by the established sociomateriality that limits their design options, they actively and artfully exploit accommodate these resistance in order to seek alternative sociomateriality. Therefore, the threesome dance of agency – the mangle of sociomaterial practice among digital and physical materiality and human agency of engineers – is the central force that fuels the digital innovation of infotainment at both CarCorp and AutoInc.

Indeed, design practices changed with MOST, but what did this threesome dance of agency in the mangle of sociomateriality mean to digital innovation? First, we can establish that the local MOST architecture did not obliterate any architectural knowledge across the global organizations. Component-based modularity remained the dominant sociomateriality in the overall design of cars. Although making use of remote sensors and data, MOST did not generate architectural innovations at this level. However, the new concept did radically change the domain of infotainment, forming new sociomateriality locally. Designers and architects were given a wide range of new tools supporting innovation, resulting in novel HMI solutions and extensive alignment and coordination between functions. At the same time, the potential power of MOST was never fully utilized, since the hierarchical organizations remained. In practice, the locus of design was moved upwards in the hierarchy from the component level to the functional level. Ironically, innovation ended up in the hands of fewer people, not more. Fostered in a component-based innovation paradigm, where control is a central aspect of architectural ideas, they exploited the logical perspective for centralization, rather than multiplicity. Finally, our study suggests that modular innovation that was traditionally performed by suppliers was obstructed by the new MOST concept. With suppliers becoming less independent, they took a more passive role in innovation, leaving more of the design work for the auto makers.

Our study has two important implications for the use of sociomateriality in digital innovation. First, as we pointed out, hardware-centric component-based modularity and software-centric service-based modularity represent two different forms of materiality. In studying digital innovations, where a key challenge is to merge digital components with physical components, researchers must carefully separate differences in physical and digital materiality (Leonardi 2007, Leonardi and Barley 2008). As earlier studies of sociomateriality often focus on a single technological innovation, thus directing the attention to the interplay between material agency of a single technology and human agency, this suggests a rather significant theoretical extension to the existing sociomateriality works. Second, our study also suggests a possible third element in the dynamics of mangle in the context of digital innovations. In our study, while the physical material agency of component-based modularity resisted the human agency of designers who wanted to introduce MOST, the digital material agency of service-based modularity subjected itself to human agency, becoming a part of accommodation strategy. This threesome dance of agency among physical materiality, digital materiality and human agency through resistance, subjection and accommodation seems to be a fundamental mechanism in the evolution of digital innovations.

Finally, a sociomateriality perspective suggests that the contours of digital innovations are temporarily emergent. It is never fully known ahead of time, only situated in response to human purpose, goals, plans and actions through resistance. As Pickering notes, resistance is liminal, existing at the boundary – always open and becoming. Similarly, a sociomateriality perspective also suggests that the contours of human agency in digital innovation are equally emergent and tentative. To quote Pickering (1993), "[n]o one could have foreseen in advance that this transformation would come about; no identifiable features of [innovator's] initial situation determines it. [Innovators] did not intent it at all. […] The social evolution of [innovator's work practice] itself was constitutively the product of maneuvers in the field of material agency" (p. 581)[1]. In this case, it is the maneuvers in the fields of two material agencies: physical and digital.

Conclusions

Digitalization and converging digital technologies have changed the innovation landscape in many industries. Generic platforms and dramatically reduced communication costs allow for integration of previously unconnected activities and artifacts (Yoo et al. 2008), opening a new source of creativity at the level of product designers. In contrast to many other areas, traditional product developing industries have not been particularly successful in conquering these architectural dimensions of digitization. Our study suggests that the threesome dance among physical material agency, digital material agency and human agency is an important aspect in explaining how embedded IT emerges in manufacturing. Digital architectures have to materialize in harmony with existing regimes, rather than replacing them. Essentially, the evolution of digital technologies in manufacturing is a result of a mangle of sociomaterial practices, resolving various resistance, subjection and accommodation among physical and digital materiality and human agency.

[1] In his original writing, instead of innovator, Pickering was referring to Glaser who invented bubble chamber.

References

Allen, P. *Service Orientation: Winning Strategies and Best Practices* Cambridge University Press, 2006.

Andersson, M., Lindgren, R., and Henfridsson, O. "Architectural Knowledge in Inter-Organizational IT Innovation," *Journal of Strategic Information Systems* (17:1) 2008, pp 19-38.

Andreasson, L., and Henfridsson, O. "Digital Differentiation, Software Product Lines, and the Challenge of Isomorphism in Innovation: A Case Study," in: *Proceedings of ECIS2009*, 2009.

Baldwin, C.Y., and Clark, K.B. "Managing in the age of modularity," *Harvard Business Review* (75:5) 1997, pp 84-93.

Baldwin, C.Y., and Clark, K.B. *Design Rules: The Power of Modularity Volume 1* MIT Press, Cambridge, MA, USA, 2000.

Barabba, V., Huber, C., Cooke, F., Pudar, N., Smith, J., and Paich, M. "A Multimetod Approach for Creating New Business Models: The General Motors OnStar Project," *Interfaces* (32:1) 2002, pp 20-34.

Barley, S. "Technology, power, and the social organization of work: Towards a pragmatic theory of skilling and deskilling," *Research in the Sociology of Organizations* (6) 1988, pp 33-80.

Boland, R.J., Lyytinen, K., and Yoo, Y. "Wakes of Innovation in Project Networks: The Case of Digital 3-D Representations in Architecture, Engineering, and Construction," *Organization Science* (18:4) 2007, pp 631-647.

Bijker, W. *Of bicycles, bakelites, and bulbs: Toward a theory of sociotechnical change* Cambridge, MA: MIT Press, 1995.

Boehm, B. "Software Engineering," *Transactions on Computers* (100:25) 1976, pp 1226-1241.

Broström, R., Engström, J., Agnvall, A., and Markkula, G. "Towards the Next Generation Intelligent Driver Information System (IDIS): The Volvo Car Interaction Manager Concept," in: *Proceedings of the 13th ITS World Congress*, 2006.

Callon, M. "Some elements of a sociology of translation: domestication of the scallops and the fishermen of St Brieux Bay," in: *Power, Action and Belief: A new sociology of Knowledge?*, J. Law (ed.), Routledge & Kegan Paul, London, 1986, pp. 196-229.

Fichman, R.G. "Going Beyond the Dominant Paradigm for Information Technology Innovation Research: Emerging Concepts and Methods," *Journal of Association for Information Systems* (5:8) 2004, pp 314-355.

Gamma, E., Helm, R., Johnson, R., and Vlissides, J. *Design Patterns: Elements of Object-Oriented Software Architecture* Addison-Wesley Reading, MA, 1995.

Garud, R., Kumaraswamy, A., and Langlois, R.N. (eds.) *Managing in the Modular Age: Architectures, Networks, and Organizations*. Blackwell Publishing, Malden, MA, 2003.

Godoe, H. "Innovation Regimes, R & D and Radical Innovations in Telecommunications," *Research Policy* (29) 2000, pp 1033-1046.

Henderson, R.M., and Clark, K.B. "Architectural Innovation: The Reconfiguration of Existing Product Technologies and the Failure of Established Firms," *Administrative Science Quarterly* (35:1) 1990.

Henfridsson, O., Yoo, Y., and Svahn, F. "Path Creation in Digital Innovation: A Multi-Layered Dialectics Perspective," *Sprouts: Working Papers on Information Systems* http://sprouts.aisnet.org/9-20 (9:20) 2009.

Heineman, G., and Councill, W. *Component-based software engineering: putting the pieces together* Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 2001.

Huhns, M., and Singh, M. "Service-oriented computing: Key concepts and principles," *IEEE Internet Computing* (9:1) 2005, pp 75-81.

Jones, M. "Information systems and the double mangle:steering a course between the Scylla of embedded structure and the charybdis of material agency.," in: *Information Systems: Current Issues and Future Challenges*, T. Larsen, L. Levine and J. DeGross (eds.), Laxenburg: International Federation forInformation Processing, 1998, pp. 287-302.

Jonsson, K., Westergren, U.H., and Holmström, J. "Technologies for Value Creation: An Exploration of Remote Diagnostics Systems in the Manufacturing Industry," *Information Systems Journal* (18:3) 2008, pp 227-245.

Kallinikos, J. *The consequences of information: institutional implications of technological change* Edward Elgar Publishing, Cheltenham, UK, 2006.

Klein, H.K., and Myers, M.D. "A Set of Principles for Conducting and Evaluating Interpretive Field Studies in Information Systems," *MIS Quarterly* (23:1) 1999, pp 67-93.

Langley, A. "Strategies for theorizing from process data," *Academy of Management Review*) 1999, pp 691-710.

Latham, R., and Sassen, S. *Digital formations: IT and new architectures in the global realm* Princeton University Press, Princeton, NJ, 2005.

Latour, B. *Science in Action* Harvard University Press, Cambridge, MA, 1987.

Law, J. *After method: Mess in social science research* Routledge, 2004.

Leonardi, P., and Barley, S. "Materiality and change: Challenges to building better theory about technology and organizing," *Information and Organization* (18:3) 2008, pp 159-176.

Leonardi, P.M. "Activating the informational capabilities of information technology for organizational change," *Organization Science* (18:5) 2007, pp 813-831.

Lyytinen, K., and Rose, G.M. "The Disruptive Nature of Information Technology Innovations: The Case of Internet Computing in Systems Development Organizations," *MIS Quarterly* (27:4) 2003, pp 557-595.

Markus, M., and Robey, D. "Information technology and organizational change: conceptions of causality in theory and research," *Management Science* (34:5) 1988, pp 583-598.

Mathiassen, L., Munk-Madsen, A., Nielsen, P.A., and Stage, J. *Object Oriented Analysis and Design* Marko Publishing, Aalborg, Denmark, 2000.

Monteiro, E., and Hanseth, O. "Social shaping of information infrastructure: on being specific about the technology," in: *Information technology and changes in organisational work.*, W. Orlikowski, G. Walsham, M.R. Jones and J.I. DeGross (eds.), Chapman & Hall, 1995, pp. 325-343.

Murmann, J.P., and Frenken, K. "Toward a systematic framework for research on dominant designs, technological innovations, and industrial change," *Research Policy* (35) 2006, pp 925-952.

Nelson, R.R., and Winter, S.G. *An Evolutionary Theory of Economic Change* Harvard University Press, Cambridge, MA, 1982.

Orlikowski, W.J. "The duality of technology: Rethinking the concept of technology in organizations," *Organization Science* (3:3) 1992, pp 398-427.

Orlikowski, W. "Using Technology and Constituting Structures: A Practice Lens for Studying Technology in Organizations," *Organization Science* (11:4) 2000, pp 404-428.

Orlikowski, W. "Sociomaterial practices: exploring technology at work," *Organization Studies* (28:9) 2007, p 1435.

Orlikowski, W., and Iacono, C. "Research commentary: desperately seeking the" IT" in IT research-A call to theorizing the IT artifact," *Information Systems Research* (12:2) 2001, pp 121-134.

Orlikowski, W., and Scott, S. "Sociomateriality: Challenging the Separation of Technology, Work and Organization," *The Academy of Management Annals* (2:1) 2008, pp 433-474.

Papazoglou, M., and Georgakopoulos, D. "Service-oriented computing " *Communications of the ACM* (46:10) 2003, pp 24-28.

Papazoglou, M., and van den Heuvel, W. "Service oriented architectures: approaches, technologies and research issues," *The VLDB Journal The International Journal on Very Large Data Bases* (16:3) 2007, pp 389-415.

Parnas, D.L. "On the criteria to be used in decomposing systems into modules," *Commun. ACM* (15:12) 1972, pp 1053-1058.

Pickering, A. "The mangle of practice: agency and emergence in the sociology of science," *American Journal of Sociology* (99:3) 1993, pp 559-589.

Pickering, A. *The Mangle of Practice: Time, Agency & Science* University of Chicago Press, 1995.

Rose, J., Jones, M., and Truex, D. "Socio-Theoretic Accounts of IS: The Problem of Agency," *Scandinavian Journal of Information Systems* (17:1) 2005, pp 133-152.

Royce, W. "Managing the development of large software systems," 1970.

Sanchez, R., and Mahoney, J.T. "Modularity, Flexibility, and Knowledge Management in Product and Organization Design," *Strategic Management Journal* (17) 1996, pp 63-76.

Schultze, U., and Orlikowski, W.J. "A Practice Perspective on Technology-Mediated Network Relations: The Use of Internet-Based Self-Serve Technologies," *Information Systems Research* (15:1) 2004, pp 87-106.

Slife, B. "Taking practice seriously: Toward a relational ontology," *Journal of Theoretical and Philosophical Psy. Vol* (24:2) 2004.

Sosa, M.E., Eppinger, S.D., and Rowles, C.M. "The Misalignment of Product Architecture and Organizational Structure in Complex Product Development," *Management Science* (50:12) 2004, pp 1674-1689.

Strauss, A., and Corbin, J. *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*, (2 ed.) SAGE, Thousands Oaks, CA, 1998.

Swanson, E.B. "Information Systems Innovation among Organizations," *Management Science* (40:9) 1994, pp 1069-1092.

Swanson, E.B., and Ramillier, N.C. "Innovating Mindfully with Information Technology," *MIS Quarterly* (28:4) 2004, pp 553-583.

Ulrich, K. "The role of product architecture in the manufacturing firm," *Research Policy* (24:3), May 1995, pp 419-440.

Ulrich, K.T., and Eppinger, S.D. *Product Design and Development* McGraw-Hill, 2003.

Walsham, G. *Interpreting Information Systems in Organizations* John Wiley & Sons, Chichester, 1993.

Walsham, G. "Doing interpretive research," *European Journal of Information Systems* (15:3) 2006, p 320.

Wolf, W. "What is embedded computing?," *Computer* (35:1) 2002, pp 136-137.

Yin, R.K. *Case Study Research: Design and Methods*, (4 ed.) SAGE Publications, Thousand Oaks, 2009.

Yoffie, D.B. *Competing in the Age of Digital Convergence* Harvard Business School Press, Boston, MA, USA, 1997.

Yoo, Y., Lyytinen, K., and Jr., R.J.B. "Distributed Innovation in Classes of Networks," in: *HICSS*, IEEE Computer Society, 2008, p. 58.

Zammuto, R., Griffith, T.L., Majchrzak, A., Dougherty, D.J., and Faraj, S. "Information technology and the changing fabric of organizaiton," *Organization Science* (18 5) 2007, pp 749-762.

Zittrain, J. "The Generative Internet," *Harvard Law Review* (119:7) 2006, pp 1974-2040.