Association for Information Systems

AIS Electronic Library (AISeL)

Selected Papers of the IRIS, Issue Nr 10 (2019)

Scandinavian (IRIS)

12-18-2019

Re-organizing for Digital Product Platforms: The Work of Vehicle Motion Engineers

Charlotte Shahlaei

Ulrika Lund Snis

Dick Stenmark

Rohit Tikekar

RE-ORGANIZING FOR DIGITAL PRODUCT PLATFORMS: THE WORK OF VEHICLE MOTION ENGINEERS

Research paper

Shahlaei, Charlotte A., University West, Trollhättan, Sweden, charlotte.aghavan.shahlaei@hv.se

Lund Snis, Ulrika, University West, Trollhättan, Sweden, ulrika.snis@hv.se

Stenmark, Dick, Gothenburg University, Gothenburg, Sweden, dick.stenmark@ait.gu.se

Tikekar, Rohit, National Electric Vehicle Sweden (NEVS), Trollhättan, Sweden, rohit.tikekar@nevs.com

Abstract

As flexibility and generativity of digitized information continuously afford new possibilities, a significant challenge for organizations becomes pinpointing forms and kinds of practice that are befitting from various aspects. Two overarching digitization eras have so far determined the greatness of the challenge for organizations; 'computerization', and 'the Internet'. Today, a third era of digitization is marked by the emergence of digitized products. As an increasing number of code lines and software are being incorporated in previously physical products such as cars, they can be used as complete products on one layer, and simultaneously turn into platforms enabling other firms to develop and integrate new components, content, or services on another layer. As digital product platform's multiple design layers need to be open to various applications and agendas, their development requires new justifications and approaches for organizing work. By looking into the characteristics of digital product platforms, we discuss the shifts in the work of engineers as they engage in developing digitized products along three main courses of action. We illustrate how these courses of action are formed based on the requirements of developing digital product platforms rather than managerial presuppositions.

Keywords: digital product platforms, digitization, organizing, work, engineers

1 Information Digitization Eras and the Emerging IS Research

With the advent of information digitization and its accompanying globalization course, the 'what's and 'how's of work are defined by logics and trends that extend beyond the managerial imperative in a single organization (Winter, Berente, Howison and Butler, 2014). As 'flexibility' and 'generativity' of digitized information continuously afford new possibilities, a significant challenge for organizations becomes pinpointing forms and kinds of practice that are befitting from an economic, ethical, safety and security stance (Brynjolfsson and McAfee, 2014). Information Systems scholarship is prolific with research on how information technologies have both supported and altered work, and how organizations have responded to these changes (Alter, 2008; Vessey, Ramesh and Glass, 2002). The challenge is not new then; it is the scope of the challenge that guides organizations and consequently IS research into a new direction.

So far, two overarching digitization eras have determined the greatness of the challenge for organizations; 'computerization', and 'the Internet' (Yoo, Henfridsson and Lyytinen, 2010; Tilson, Lyytinen and

Sorensen, 2010). With the emerging computing power, in the first era, organizational challenges were associated with improving the efficiency of internal operations and decision making. With the advent of net-enabled firms, in the second era, the focus was on how collaborative systems, knowledge management and e-business systems assisted competitive capability in a distributed network of firms (Yoo, et al., 2010; Tilson et al., 2010). The ongoing efforts for efficiency and flexibility brought by information digitization today is snowballing into the emergence of a third digitization era; the era of 'digital product platforms', with organizing challenges of its own (Yoo et al., 2010).

Digital product platforms such as iPads or more recently autonomous cars, can be used as complete products on one layer, and simultaneously enable other firms such as traffic or weather agencies, or app developers outside the industry to develop and integrate new components, content, or services on another layer (Yoo et al., 2010). A digital product platform's multiple design layers are thus open to new designs and applications and indicate multiple and conflicting goals and agendas in the process of their development. In the automotive industry, for instance, apart from the benefits of information digitization, such as big data simulations and virtual modelling in design and development processes, the industry will sow profits from facilitating users with digital services and autonomous driving assistance integrated in the car. (Khare, Stewart and Schatz, 2016).

To provide users with these services, extensive organizing is required for software design and development processes. Additionally, the car of the future needs to be able to not only monitor its own working parts and the safety of conditions around it in real time, but also to communicate with other vehicles and with an increasingly intelligent roadway infrastructure (Khare et al., 2016, Gao et al, 2014). These changes indicate that cars will be more like integrators of multiple technologies, productive data centers (Khare et al., 2016) and ultimately like apps-on-wheels rather than metal boxes (Khoushik and Mehl, 2015). The ability to quickly integrate multiple technologies such as traditional products with digital components and services will then have profound implications for organizing in the third digitization era.

A prominent implication relates to forming new research agendas within IS (Lyytinen et al., 2016). For instance, as we move from computerization and the age of net-enabled firms to the age of digital product platforms with increased innovation capabilities, rather than focusing on how the implementation and uses of information technology will affect the efficiency of organization' internal processes, scholars can ask what organizing processes can best support the development of new digital technologies. Like the earlier agendas, this new line of inquiry focuses on the relation between digital technologies and organizing processes. Unlike them, the focus is now on the *development* of these technologies rather than their *use*.

In this light, we have taken digital product platforms as an example of today's digital technology, and asked "how do the characteristics of work for developing digital product platforms impact processes of organizing". To provide a research context for investigating the effects of developing digital product platforms on organizing, we have selected the automotive industry as being involved in developing autonomous cars, i.e. a case in point for digital product platform. By looking at the engineers' work as they develop product lines for autonomous cars, we provide detailed real-case examples of how the technological characteristics of developing digital product platforms creates new work conditions. We have then discussed how these new work conditions are perceived to influence their related organizing processes.

In doing so, we address the practical implications of pervasive digitization for organizations involved in rapid product development contexts, such as digital platform owners. As could be anticipated, a practical challenge for these organizations is forming organizing processes that will support fast and adaptive product development in a thriving ecosystem of heterogenous firms. Our theoretical contributions include aligning the way we think about organizing with the current phenomena of interest in IS. This is arguably a complicated task as the phenomenon of interest in IS is a moving target, changings rapidly with changes in information technology (Gregor, 2018).

2 Digitization of Physical Products: Developing Digital Product Platforms

Today, the impact of cheaper, smaller, and more powerful computers has reached beyond corporate backrooms (Tilson et al, 2010, Yoo et al, 2010) into nearly all aspects of life (Sorensen, 2017). The continuously developing computation capabilities tied with the communication and connection potentials resulting from the emergence of the Internet (Yoo, 2010) has led to the digitization of many established physical products, such as books, music files, cameras or cars (Yoo et al, 2010). Such Digitization refers to the technical process of encoding analog information into digital format (Tilson et al., 2010, Yoo et al, 2010) in physical products which arguably is different from what is known as digitalization.

We distinguish digitization from digitalization i.e., the process of "applying digitizing techniques to broader social and institutional contexts" (Tilson et al, 2010: p.2) where digital technologies are used (Hukal and Henfridsson, 2017). In this study, we focus on digitization and when we refer to digital technologies, what we have in mind is digitized products (in this case, a digitized car) as an incorporator of various digital technologies. The reason for doing so is that we intend to tackle the technical particularities of development-digitizing physical products-rather than the processes of applying or using these products in broader social contexts.

Digitization of physical products relies on a continuous carrying out of new combinations of digital and physical components for developing novel products (Yoo, et al., 2010). As firms increasingly embed digital components into physical products, an ensemble of components from a set of heterogeneous firms are incorporated in the product. This way, a prominent characteristic of digitized products is that, they can be simultaneously a product and a platform, enabling other firms to invent novel components such as new applications and supplementary hardware accessories each of which can expand the basic functionality of the product. Yoo et al., (2010) argue that digital product platforms, such as iPads, Google Maps or autonomous cars have a few characteristics that are fundamental to their design and development, and that tend to affect organizing processes supporting their development. Below, we have summarized these characteristics based on Yoo et al.'s arguments, as we have understood and used them in our work to show the effects of digitization on organizing processes:

> ***Enabling*:** by allowing other firms to create applications and complementary components, digital products act as platforms enabling various heterogenous firms to develop their own components and products. To act as both a product in one layer, and an enabling platform on another layer, the architecture of the digitized product needs to be designed in ways that will allow its components to be connected to components developed by not only the firms inside the industry but also to those outside the industry.
>
> **Recombinable:** once the components of a product are intended for being connectable or integrable with/in other components developed by heterogenous firms, they offer generativity. Generativity refers to a technology's overall capacity to produce new contexts for development and usage that is unprompted, unplanned and usually driven by large, varied, and uncoordinated audiences. Every time the development or application context differs, the components need to have the capacity to be recombined in a new ensemble of components which result in a new development or use context. This means that their architecture needs to be designed so that changes in combination of components does not lead to changes in the core features of these components. This will increase efficiency and flexibility.
>
> ***Product agnostic*:** having the capacity to be recombined in unprompted ensembles implies that the designers of digital product platforms cannot fully know how the components will be used. This is because the product does not have a fixed boundary which will mark when the product is complete. An iPad or a smart phone, for instance, are never complete products. Their boundaries change as long as there are applications that could be integrated in them. Thus, the components in a digital

> product platform belong to different design hierarchies rather than the single design hierarchy of a given product. Hence, these components are product agnostic and their designers cannot anticipate all the possible ways in which these products will be recombined.
>
> ***Doubly distributed:*** digital product platforms are enablers for other firms for further development. Thus, the platform owner's ability lies in mobilizing a vibrant network of firms that can generate new contexts for design and use. In such an ecosystem, the value, control and the knowledge of developing digital product platforms is distributed across multiple firms. Since, the key here is attracting a large number of unexpected development scenarios by firms outside the industry as well as those inside the industry, the knowledge and value sources are doubly distributed.

The above-mentioned features are not exclusive; rather they are interrelated and overlapping. Our separation of them is only for analytical convenience. Yoo et al. (2010) explain that these features are consequential when it comes to the rationale for designing and evolving specific organizational arrangements. Our aim is to identify and describe how developing product—which are supposed to be enabling platforms for unknown future designs, with distributed knowledge and value bases— will affect various aspects of work inside firms. Next, we explain the research setting where we investigated how these features could affect organizing, followed by a description of our data gathering as well as data analysis methods.

3 Research Setting

The OEM (original equipment manufacturer) we have selected as the setting for our study has been focused on a fundamental reorganization process that aims at reducing the decision-making hierarchy and promoting an upward development structure. The company's emphasis on shaping processes and structures in a bottom-up way have coincided with its investments in developing autonomous cars. Since making autonomous cars is still at its early stages, even for OEMs with a long record in the automotive industry, the work guidelines and development requirements remain largely undefined to this date. The focus on minimizing decision-making hierarchies and an upward development structure then appears to be inevitably fitting. Besides the promised benefits of reduced hierarchy and distributed decision-making, there are also drawbacks. The younger engineers who are mostly in charge of developing product lines for autonomous cars find it increasingly challenging to be left to their own devices when it comes to setting product requirements and establishing work procedures. However, they are also aware that, given the newness of the products they are developing, it is practically impossible to have clear goals and structures before product lines are actually developed and, that only in retrospect, it would be possible to refine goals, requirements, models and procedures.

A smaller part of this challenge is thus motivated by the senior engineers assuming a more supervisory role than being immersed in product development. A larger share of the challenge is however prompted by the unconventionality of the work required for developing product lines for autonomous cars. Given the rather newness of the work requirements as well as the ongoing reorganization processes, we found little benefit in focusing on how organizing processes are inherited from management to the organizational teams in a bottom-down approach. Instead, we have studied the daily practices of the engineers to explore how organizing processes are shaped around and adapted to these daily practices. To do so, we have focused on the work of one particular squad; IVCcore. 'Squads' are names given to teams assigned to developing specific product lines. IVCcore squad is the abbreviation for Integrated Vehicle Control team which exists in the company's software development division and consists of 8 engineers. To put it simply, IVCcore squad has been in charge of two major development lines since its formation. Firstly, the squad has embarked on integrating the electric control units (ECUs) for three fundamental vehicle motions, including propulsion, steering, and braking in a single ECU. Secondly, the team has been modifying the ECUs to be eventually applied in autonomous driving (AD). Given this description, the squad's work is expected to illustrat an exemplary shift in practice and organizing necessary not just for developing products, but also for developing digital product platforms that are open to future modification and application. Following the squad's work has therefore been a useful strategy for capturing a shift from previous work and organizing arrangements.

4 Data Gathering

Given the exploratory nature of our study, we have followed the work of other researchers who have adopted an iterative approach[1] (Leonardi and Bailey, 2008; Henfridsson, Mathiassen and Svahn, 2014; Svahn, Mathiassen and Lindgren, 2017). Starting with initial interviews of mid-management and senior engineers, we first acquired a preliminary overview of the company's agendas related to the development of autonomous cars. The initial interviews were mainly aimed at understanding what divisions of the organization were specifically dedicated to research and development of autonomous product lines and how the human resources and competences were distributed across developer teams. Based on these initial interviews, we were able to identify the key informants in divisions of interest who then directed us to the IVCcore squad. The observation of the IVCcore squad started in October 2018 and is still ongoing. The observation sessions have included at least 3 complete work days weekly, and 1 day dedicated to an overarching analysis of data that could be used for devising research strategies for the week after.

The IVCcore squad consists of 8 developer engineers all seated in the same office area without any partitions dividing them. This spatial specification allows them to engage in conversations constantly to brainstorm, troubleshoot or discuss issues related to their work. The first author who has been conducting the observations, has been seated in the same area with the team. As there are no partitions separating the team members, it has been possible for the researcher to both see and hear the team members performing their work easily. During observations, we have been writing careful fieldnotes making sure to record not only the activities of the team, and the artifacts they use, but also the topics discussed by the team members as they engage in conversations to perform their work. These notes have assisted us to pose follow up questions and inquire about the tasks and challenges in the team. This strategy has usually resulted in exploring new topics and areas of the squad's work that could be further explored. Relevant conversations have been audio recorded and subsequently analyzed by the end of each week. This research design as well as the spatial arrangement during observations has allowed for witnessing things as they occur. Our observations thus have resembled what Hennik, Hutter and Bailey (2010) describe as "watching an unfolding drama unfold with characters, events and story lines" (p. 170).

By following the conversation topics among team member as well as their activities on a daily basis, we have been able to develop tentative tables that are intended for capturing phases of product development, activities, challenges, planning, and structure of work for the IVCcore squad (see table 2). To make sure that we have understood the observed activities and recorded all topics discussed in the group, we have conducted semi-structured interviews with the squad members where these tables were filled with the help of each interviewee confirming that both the observed topics were relevant and that the descriptions of technical issues in projects and tasks matched the engineers work.

What			How			
Phases to AD	**Goal**	**Task**	**Challenge**	**Planning phases**	**Order**	**Special to AD**

Table 1. The work of IVCcore based on phases of product development

As is clear in table 1, the phases of product development extend a storyline describing the product-related projects the IVCcore squad has and will be taking on. The left half of the table presents what the squad

---

[1] The data gathering phases are described in table 2 in appendix A.

does, establishing a narrative history for the team's work which can help outline the key properties of order and structure (Van de Ven and Huber, 1990) as the team's work proceeds. The right half of the table indicates how the projects and tasks are planned for. Each member of the squad has been scheduled for two in-depth interviews; one interview dedicated to *what* the work entailed and one dedicated to *how* the work was structured and planned for. The data in this paper is based on the above-mentioned interviews as well as the entire field investigation until April 2019.

5 Data Analysis

By focusing on forming the interview tables, we have intended to make the details of our observations and interviews transparent. This method has helped to, 1. summarize the most prominent themes of our observations, 2. make the interviewees' own classification of events clear, and 3. follow a "disciplined pursuit and analysis of the data" (Sarker, Xiao and Beaulieu, 2013). The interview durations ranged among 60- 75 minutes. Once the interviews were done, there were 8 tables -developed through 16 interviews- illustrating the work of IVCcore squad based on the way each interviewee had framed and phrased their work. In the first analysis phase, the 8 tables were compared and contrasted to formulate a single table that is illustrative and comprehensive of the IVCcore's work based on all 16 interviewees. This was the inductive phase of the analysis where we emphasized telling a story in detail and avoiding the risk of missing parts that lied outside the scope of a theory (Walsham, 1995).

The second phase of analysis included looking for a shift in the logic and strategies of structuring work as the team focused more on the development of functionalities for AD. As clear in table 2, an important probe during interviews was whether any part of what they were doing was necessarily related to making autonomous cars and not a traditional car. The purpose of this probe was to distinguish between activities or forms of practice which are accommodating towards future adaptations and applications by multiple users which are unclear and uncoordinated at the time of development. In a third analysis round, we looked for how these shifts were relevant to the characteristics of digital product platforms as described by Yoo et al. (2010). For instance, we looked at how being an "enabling" platform affects what engineers do and how they do it. By doing so, we hoped to move beyond a mere description of events and bind the empirical phenomenon to the cumulative research on organizing, and thus aim for a more compelling story of our digital age (Henfridsson, 2014).As might be expected, our analysis was an iterative process including several rounds of going back and forth between interviews, as well as trying to make sense out of sometimes conflicting issues between observations and interviews. We have presented the result of this iterative process in the next section.

6 The Work of the IVCcore squad

The events identified at the field during our time with the IVCcore squad, through both observations and interviews, can so far be categorized into 3 overarching courses; 1, developing an integrated vehicle control unit for the three main motions of the car, i.e., propulsion, steering and braking, and 2, modifying the architectural model for the integrated control unit, and 3. revamping organizational arrangements to support the current work. These three courses correspond to the main work of the IVC core, the architectural model of the product lines for autonomous cars, and 3, organizational arrangements for the development of these products. As mentioned previously, the subject of the interviews has been two very general questions of "what do you do", and "how do you plan or order your work". Below, we have presented these three major courses of action, illustrating the features of the work, the architectural model and the organizational design with excerpts from the interview transcripts.

6.1 Integrating Vehicle Control Units

The electric control units for the three main motions of the car have been separate components. For a faster, more seamless communication between these control units, as well as reducing extra weight of several hardware that each carry a separate control software, integration sounds like a smart idea. Especially in the case of electric and later autonomous cars, an integrated vehicle control (IVC) unit gives hope of a higher battery efficiency and a more fluent communication between propulsion, braking, and steering control units. Another issue is that, in traditional cars, where a driver still has had the control of the steering in the car, there has been no control unit for steering. In autonomous cars, there would be a need to replace human driver's control with an extra ECU for steering. This would mean adding to the complexities of having multiple components for control. Therefore, it is now favorable to put effort and time into improving the efficiency of the vehicle control.

The integration of the control units includes dedicating a considerable amount of time and human competence to research and development as this is a new trend. Yet, it is of high importance for OEMs to develop IVCs in-house. The in-house development does not so much target replacing the supplier-made components as it desires to gain control of the main vehicle motion software. Previously, the practice of developing IVC unit was not worth dedicating organizational resources such as time, money and human competence since it would only add to the efficiency of the car. Today, efficiency would be added value, the main catch is that as the car needs to be ready for being connected to external sources of command such as a remote controller, or the Internet of things, the control software needs to be modifiable on the spur of the moment. An in-house development would equip the OEMs with the knowledge of how the IVC runs at its core, and with the ability to customize the control according to the requirements of any future project in collaboration with other firms.

Integrating the control unit entails iterative rounds of analyzing the requirements, planning, prototyping, developing, testing and modifying. First, a prototype environment should be set up for producing a clone-like ECU like the ones made by a supplier. This is to make sure that the group has the capability of producing the exact same powerful ECU that was previously made by a supplier. Next, a model of the desirable IVC would be sketched and then tested in the simulation environment. Then the functionalities of the core ECUs would be integrated, an undertaking that would take months. The software would be tested in simulation in many iterations. Next the software would be flashed on the ECU hardware and this would also be tested in simulation. Once all these simulation tests are checked out, it would be the time to test the IVC in the car. This is a landmark for the squad to know that they have gained the core control of the car. Next step into making the car of the future would be to develop an integration layer that allows the control of the car to be controlled by any external source such as a remote controller or other partners. This task includes developing interfaces for connecting the IVC to various sources of command.

These partners are not just other OEMs focused on developing digital drivers. Rather, they can be firms outside the automotive industry with hands-on experience in neural networking, deep learning and developing digital drivers. The IVCcore squad has been involved in projects where a collection of firms, each with their own specific field of expertise, bring in their own components. But for the cars to be autonomous, these firms need the IVC squad to prepare the car as a platform for their components to grow. In the excerpt below, the engineer is explaining how their work entails turning the car into an enabling platform for other firms:

> *Yes. So, uh, how can I say this! They have some goals in this project for demonstrating some autonomous drive functions. And our part in that is not developing the autonomous driver. But we're delivering the car, and that with an interface to connect to the AD functionality. So, someone else will tell us steer this or that way, drive forward, or brake. So, someone else commands and we will just be the slaves in that [functionality demonstration], they [the outside partner] will request us to do that. So yeah, it's like we're delivering an enabler for them.*

To provide an enabling platform for other parties, the initial phases of developing the vehicle control software has focused on two different functional layers. The core layer belongs to the control of the car. The other layer belongs to developing automatic driving assistance (ADAS) functionalities such as adaptive cruise control (ACC), as well as fully autonomous drive (AD) functionalities such as automatic valet parking (AVP), which will use the core control functionalities in the bottom layer to work. Any future use case might require different functions in the car. For instance, for a budget version, the OEM and the service providers might decide to exclude certain functions in the car. As the engineers emphasized, the architecture for the vehicle control software needs to be prepared for such design variations:

> *You cannot say what components you need in different projects. For example, you know, Drivemode is just about having integrated drive train module. So, it is just about, combining the inverter, gearbox and of course the motor for one wheel. So you will have IEMs for two wheels in the front and for that you don't need the ADAS functions. In fact, none of the ADAS functions is required. If you include ADAS functions in that particular project, then it has to be tested with those things and that effort is basically not required because those features are not needed. So, if you can take all the existing model and just say "okay I don't need this or that component and the model is still the same, then you need less effort to provide the required software for the project.* (R5)

6.2 Modifying the Product Architecture Model

To put the importance of modifying the architectural model for autonomous cars into perspective, it is imperative to know how autonomous functionalities work. The core vehicle motion functions, propulsion, braking and steering each could be thought of as fundamental components that form the core layer of the vehicle control architecture. The integrated form of these core components is called IVCcore. These components will then be mixed in certain combinations to provide the car with various automatic driving assistance (ADAS) functions. ADAS functions include, for instance, adaptive cruise control (ACC) or, automatic emergency braking (AEB), or lane keep assist (LKA) which already exist in many versions of today's cars. These functions are called ADAScore and are placed in a second layer on top of the IVCcore, in the architecture model.

ADAS functions are not to be mixed with fully autonomous driving (AD) functionalities. An AD functionality such as automatic valet parking (AVP), or platooning might require a combination of ADAS functions such as AEB or ACC for the car to be able to park itself in the parking lot without human intervention. AD functionalities, as expected, will need a lot of sensor fusion information for detecting objects around and interpreting the information that comes from the environment. AD functions are thus not the core functions of a car, rather they are built on the core functions of the car which are IVC and ADAS core. Additionally, as mentioned in the previous section, developing these functionalities is undertaken mostly by firms with sturdy knowledge and experience in neural networking and deep learning areas. Such firms are usually outside the automotive industry. For any prospective project, these firms might need certain ensembles of the core vehicle control components in the car to be activated in specific combinations.

Thus, the enabling power of the OEMs to turn their car into a platform for other firms for developing their own products (AD functionalities) is heavily project-based. This all requires the architecture of the IVCcore and ADAScore to have the potential for allowing the components to be mixed and matched according to the requirements of each project. What affords the model with such a potential is two parameters. The components in the model need to be generic and modular. By generic, the engineers mean that the components should only address the core functionalities of the car motion. That is, the components need to have the capacity to be connected to various sources of control or re-combined in various constellations for each project, without the project imposing inherent changes to the core architecture. By modular, the engineers mean that the communication, adaptor and control blocks for each vehicle motion needs to be separated in the model. This will help detecting faults in the architecture or customizing these blocks based on emerging requirements.

In the previous vehicle motion architecture, these components were all integrated in each component, making modification or fault detection a difficult task. Any tiny change would also require changing the rest of the components as well. So, turning an integral IVCcore architecture with integrated and inseparable components—into a modular one—with separable and recombinable components—is an underlying pursuit for the IVC engineers. The engineer below emphasizes the importance of having a modular architecture as the industry moves towards an ever more integration of software on the way towards autonomous cars:

> *Even if there wasn't work being done for autonomous cars, the trend in the automotive industry is going more and more towards software. So, software in general is the biggest reason for changing the processes in the industry, I feel. Because, unlike a regular mechanical component—which takes a lot of time and money for development, which causes you not really being able to make many variants—in case of software, you just keep adding features, or removing them, or upgrading them. And when things are so dynamic, we should be able to have things ready prior, we cannot keep on addressing every small detail or every small update when we are already lacking in manpower and when the competition is so fast. We can't sit down and try to redo things we have done again and again; we have to mainly focus on how to improve our processes and methods right now. So developing autonomous cars is not the main reason. But I would say it makes it even more important.* (R5)

The respondent below explains how for different projects, various components such as sensor information would be fused in different parts of the vehicle motion architecture, and how these combinations can change in different projects:

> *The thing is like there are these two cores: the IVC-core and ADAS-core. The core functionality, say, for example, understanding sensor information will be done over here [referring to the presentation slide]. But this sensor information will also be fused, in ADAS-core and IVC-core. And depending on the project, for example, in this Autopilot project, the controller is from IoT. So, all the sensor information, the process to censor information will be given to IoT, and IoT will say, "now steer this way, or, accelerate that way"…So, it is actually someone else deciding on how the vehicle should be controlled based on all this sensor information. And based on the project, it will actually differ. So, in Autopilot, the controller is the IoT. For Remote Control, it is actually a person sitting in front of one monitor who will actually, you know, do some steering…* (R6)

6.3 Reorganization of Human Competence

A key event for this huge start-up company has been rearranging organizational processes and human resources. There has been a lot of emphasis on Agile ways of working which according to the software managers fit well into the requirements of the automotive industry as it is moving more and more into a software driven sector. Many of these managers come with a strong background in software development for the automotive industry and have long witnessed or been frustrated with the way the automotive industry has been relying majorly on mechanical engineering requirements. In their opinion, it is time for changing things around and start with software development when thinking about making a car. Therefore, organizational processes and arrangements which support software-driven practices are to be favored. For instance, fast iterations in software edition, less dependencies on suppliers, the need to invest more in human resources as opposed to expensive physical components, the possibility to try out new features which does not entail costly investments, and many other such instances separate the dynamics of the software world from that of the hardware world. Software development thus requires its own organizing rationale and flow.

Agile management which is based on fast work iterations, shorter development cycles and regular inter-team as well as intra-team communication seems to be a fitting style especially for software-oriented industries. Nevertheless, automobiles have since long included a large portion of software and this difference between the hardware world and the software world is not a new thing. Additionally, the Agile coach managers in the company seemed to agree that Agile is a method which can be advantageous to even hardware divisions in certain aspects, even though the software word would collect a bigger portion of the benefit. Thus, the whole company has been dedicating a considerable amount of time and effort to settle for Agile working processes. The interesting point however is that Agile processes have seemed to be necessary for the software division, but not sufficient. A more specific reorganizing initiative is taken by the software head manager to reorganize the competence allocation at least for his own division.

Instead of having separate departments each specializing in one line of software development, the software division-now called the software tribe- consists of four chapters which represent the four core competence areas required in automotive software development. Under all four chapters lies around 15-20 developer engineers with various competence areas that could all fit into the automotive software area. None of these developers are fundamentally classified under any specific chapter. This means that none of these developers specializes in a particular competence area. Based on any upcoming project that calls for the development of a specific product line, the relevant competence chapters are identified. Then the relevant chapter leaders form teams of 5-8 people-called squads- whose purpose would be to develop the requested product. Based on different requirements of each development phase, the developing engineers employ their skills and acquire new ones in order to finally deliver the product.

The competence allocation, development processes, and learning areas thus change based on various projects that fall into the software backyard. This arrangement gives a lot of mobility to who works with what, what is learnt and how the competences are employed or fostered. New squads are combined and then dissolved with projects. Each engineer might end up having worked in several constellation of competences during their working time. This fluid arrangement even affects the way various squads are spatially positioned in the company. The engineer below is explaining the pros and cons of the new arrangement which gives the engineers bits and pieces of experience in a wide competence area, as opposed to a more traditional way of specializing deeply in a specific competence area:

> *I can tell you what advantages it has at the team level. At the team level, it gives you the opportunity to try out different things basically. So, the selling point would then be that, since there are not specific duties assigned to you, you will have the freedom to drive your knowledge base in whichever way <u>you</u> want, and you can pick up the tasks which you want. So, it will be on you. You will discuss what you want within the team. So, in this specific number of weeks, you will be having a touch of this, and a touch of that. The conventional way of working is that, you assign something to a specific department and that department will keep doing the same thing. Then, there will be some experience shaping in the people in that department since you keep working on the same thing again and again, but also after a point of time, you don't enjoy your work anymore. So, the selling point is that, your work environment will be very heterogenous and this keeps you engaged in it…* (R7)

This arrangement has started to be favored by even the hardware divisions. After all, developing processes that allow for synergies of fluid competence combinations seem to have more to do with the fast changes in project-based requirements rather than the differences between the hardware and software divisions. The engineer below gives an example of how the loss of competence occurring as a result of engineers' transitioning in or out of the teams or even the company can be compensated by the new competence arrangement. This is true even for the hardware divisions, even though they usually work with hardware with long lasting life cycles or change intervals which would make it more sensible for engineers to specialize in one area more deeply:

> *I'm just thinking how it helped them [the hardware division] because they deal a lot with hardware, their simulations and their calculations are always about physical objects that are tangible. So, it would make more sense for them to be together [in traditional separate divisions] in the hardware tribe... For example, there was a case in the chassis and suspension team when they used this tool called Adams and it's a simulation environment, doing vehicle dynamics, kind of how the sprints move and how the damper site and tire site, etc move. They had almost three people at one time working on that tool. But, later, all of them moved out of the company. So, now the team at the chassis suspension is left with no one with that competence area. Maybe pulling everyone to work*

> *with simulations and with tangible objects into one group can help them split competences or get people to have wider competence areas*. (R6)

7 Discussing the Shift

Being both a product and a platform assigns a dynamic nature to digital product platforms and equips its developers to both compete with other firms on one layer, and peacefully co-exist with others on another layer (Yoo et al., 2010). Vibrant platforms require the participation of third-party developers (Ghazawneh, Henfridsson, 2013), and firms which shift their focus from product development to providing resources for third party development (Tiwana, 2015). The ability to act as a platform thus requires firms to design their products in a way that their present decisions do not affect future options for design and development. Our intention was to investigate how digital product platform developers manage to realize an open and enabling design and steer away from blocking their ability to be part of a thriving ecosystem in future. Our focus was on how various organizational arrangements were shaped to actualize this objective.

Our results showed three main courses of action taking place during our investigation time with the IVCcore squad. These three main action courses were 1. Development of integrated vehicle control, 2. Modifying the product architecture, and 3. Reorganization of competence. As might be evident, these three courses correspond to three aspects of the entire team's work, namely, the work practices, the product design, and the organizational arrangements. When analysed, all these three aspects reflect orders and measures which could accommodate the characteristics inherent to digital product platforms.

At the practice level, a strategic decision was made about the in-house development of ECUs, components that were previously bought from suppliers. Taking in mind what the future design and development would look like, the firm had been resolute to match the speed required for modifying the ECUs to be connected to external sources. Continuing to buy these components from suppliers would leave the firm without any knowledge of how these components work and could therefore spoil their ability to provide the car as a platform for third-part developers of AI and digital drivers. It had become clear to the firm that both the knowledge and the value of developing a digital product platform was distributed across double layers of expertise and competition including firms both within the industry and those outside of it. The value of their product platform would therefore be determined by their enabling capacity. This was essentially why the IVCcore squad was formed and a previously non-existent practice was added to the firm's operations.

At the product architecture level, the mindset was similar. If the firm was to enable third party developers, be they developers of digital drivers, remote control systems, or autonomous driving functionalities, it had to settle for changing projects at a rate previously non-existent. Sometimes projects could be ongoing simultaneously, and sometimes they could have conflicting agendas. The core functionalities of the car were thus to be at the service of multiple development undertakings. The previously integral architecture of the ECUs would require being modified at the core for each project; an impossible expectation and certainly not feasible at the desired speed. This meant that the architecture had to become modular and generic to allow for diverse recombination of components, as well as for the components to be adapted for various simultaneous agendas. Modifying the architecture had been one of the major concerns in the team and an ongoing activity for almost a year. The architecture editing was especially at the heart of product-agnostic designs that would make not knowing what the future product would be designed and applied for. Modifying the architecture was again a course of action that had emerged with the firm's determination to develop digital product platforms.

At the organizational level, departments with separate sub-divisions that would each specialize in one particular competence area were dissolving. Like the product it was developing, the firm was arranging for its human competence to be recombinable, distributed, future-agnostic, and enabling. Rather than separating and permanently dedicating human resources to specific competence areas, the firm's strategy was based on preparing the human resources to be able to respond to project-based requirements. Teams would be formed based on what backgrounds were needed to accomplish the current project. A constellation of human resources with various backgrounds would then set out to achieve the project-based goal. After the accomplishment of projects, teams would then dissolve, and the developers would be recombined into new teams put on the task of accomplishing another project. This layout could even allow for competences from software divisions to be combined into more hardware-oriented teams during the course of a specific project to enable them with knowledge areas that the hardware divisions usually lack. This layout would also allow for the blurring of clearly defined lines that separate hardware competence from software competence, rendering uncoordinated combination of software and hardware competence areas to form the future competences.

Thus, today's digitization trend, marked with the emergence of digital product platforms, requires its own arrangements and resources. Today's digitization trend operates on practices, product architectures and organizational layouts with dynamics that support designing for and working towards incompleteness (Garud, Jain and Tuertscher, 2008). These are all organizational arrangements that lack the level of specification needed for managerial instructions to be drawn (MacCormack, Rusnak and Baldwin, 2006). Excessive innovation level needed to prosper in ecosystems that take increasing advantage of digital product platforms calls for highly skilled, independent-minded employees eager to innovate. Although these employees do not respond well to tight control and traditional management style (Baldwin and Clark, 1997), yet they are challenged by the lack of specifications surrounding their work. Practice and organization of work for developing digital product platforms thus requires a change in the mindset that is historically been made convenient by pre-specified problems and solutions. Yet, this has proven a difficult task since much of the management thinking, education and research is designed around pre-identification of problems and looking for 'optimal solutions' (Garud et al., 2008).

An interesting point was the emphasis that was put in speech, during interviews and informal chats, on the company's need to shift its focus from a mechanical engineering mindset towards a software engineering mindset. In practice, however, the observed shift was more based on a shift from fundamentally coordinated arrangements for pre-specified products with single design frames towards arrangements that could accommodate incoordination and product agnostic designs. As respondents 6 had mentioned, these arrangements were even being formed in the hardware divisions. The shift from mechanical engineering to software engineering initiatives is of course an expected one with the increasing amount of software being integrated in the automobiles. A more delicate shift appears to be the one from software engineering for automobiles as a product to automobiles as both a product and a platform.

By investigating what practice and organizing orders allow for digitizing products into platforms, we have studied digitization as an operant resource; (Lyttinen et al., 2016), i.e., a resource that operates on other things rather than being operated on (Nambisan, 2013). Our approach in studying what organizing orders are needed for digitization of products is thus different than studying digitization as an operand resource, i.e. as a resource used for or operated on to enable the performance of certain tasks. We hope this approach allows for making integrative theoretical frameworks that bridge managerial scholarship with the scholarship on technological platforms which have so far been two separate streams with their own concerns and contributions (Gawer, 2014). As digital product platforms are coming to life, work systems need to be designed for uncoordinated development work and fluid team constellations, as well as employees who can learn to prosper in such systems. As illustrated in this study, working towards digital product platforms is a harbinger of change both in the arrangement and in the meaning of competence. A useful ploy for future studies is thus investigating the change in the meaning and methods of competence development for employees.

8 Conclusion

Digitization of previously established physical products is the hallmark of a new digitization era, with its own characteristics and mechanisms. Digitizing products into platforms that empower diverse product development by miscellaneous firms calls for work settings different than those for developing digital products. Advancement of digital product platforms requires the design to deliver an entity which is open to being plugged and played in various application contexts. Hence, unlike a product with fixed design frames, digital product platforms falls out of a single design hierarchy and is subordinate to many design requirements. Additionally, digitization of physical products requires immense amounts of integration between software and hardware without losing the main functionalities of the physical product. Therefore, understanding the mechanisms of today's digitization trend requires asking how the characteristics of work for developing digital product platforms impact processes of organizing.

We have provided an insider view on work and organizing for digital product platforms. By looking at the IVCcore squad's work we have identified various courses of action that present shift in three levels of the team's work; the work practices, the product's architecture model, and the arrangement of human competence. We have explained how these courses of action illustrated a shift in from arranging for accommodating the mechanisms of digital product platforms. We have also discussed how our findings reflect the change from organizing for products to organizing for digital product platforms contrary to the expectations of the management who were determined to see the shift in the light of the differences between software and hardware development.

References

Alter, S., 2008. Defining information systems as work systems: implications for the IS field. European Journal of Information Systems, 17(5), pp.448-469.

Baldwin, C. & Clark, K., 1997. Managing in an age of modularity. Harvard Business Review, 75(5), pp.84–93.

Brynjolfsson, E. and McAfee, A., 2014. The second machine age: Work, progress, and prosperity in a time of brilliant technologies. WW Norton & Company.

Gao, P., Hensley, R. and Zielke, A., 2014. A road map to the future for the auto industry. McKinsey Quarterly, Oct.

Garud, R., Jain, S. and Tuertscher, P., 2008. Incomplete by design and designing for incompleteness. Organization studies, 29(3), pp.351-371.

Gawer, A., 2014. Bridging differing perspectives on technological platforms: Toward an integrative framework. *Research policy*, *43*(7), 1239-1249.

Gregor, S., 2018. "On Theory". In Galliers R.D. and Stein M. (eds.) The Routledge Companion to Management Information Systems (1st ed.). Routledge Ltd.

Henfridsson, O., 2014. The power of an intellectual account: developing stories of the digital age. Journal of Information Technology, 29(4), pp.356–357.

Henfridsson, O., Mathiassen, L. & Svahn, F., 2014. Managing technological change in the digital age: the role of architectural frames. Journal of Information Technology, 29(1), pp.27–43.

Hennink, M., Hutter, I. and Bailey, A., 2010. Qualitative research methods. Sage.

Khare, A., Stewart, B., & Schatz, R. (Eds.). (2016). *Phantom ex machina: Digital disruption's role in business model transformation*. Springer.

Koushik, S., & Mehl, R., 2015. The automotive industry as a digital business. *Management Summary, NTT Data*.

Leonardi, P.M. & Bailey, Diane E, 2008. Transformational technologies and the creation of new work practices: making implicit knowledge explicit in task-based offshoring. Management information systems: MIS Quarterly, 32(2), pp.411–436.

Leonardi, P.M., 2011. When flexible routines meet flexible technologies: Affordance, constraint, and the imbrication of human and material agencies. MIS Quarterly, pp.147-167.

Lyytinen, K., Yoo, Y., & Boland Jr, R. J., 2016. Digital product innovation within four classes of innovation networks. *Information Systems Journal*, *26*(1), pp. 47-75.

MacCormack, A., Rusnak, J. & Baldwin, C., 2006. Exploring the Structure of Complex Software Designs: An Empirical Study of Open Source and Proprietary Code. Management Science, 52(7), pp.1015–1030.

Mumford, E., 2006. The story of socio-technical design: reflections on its successes, failures and potential. Information Systems Journal, 16(4), pp.317–342.

Nambisan, S. 2013. "Information Technology and Product/Service Innovation: A Brief Assessment and Some Suggestions for Future Research," Journal of the Association for Information Systems (14:Special Issue), pp. 215–226.

Sørensen, C., 2017. Beyond mobile IT: Ubiquitous digitality and work 1. In *The Routledge Companion to Management Information Systems*, pp. 466-480. Routledge.

Svahn, F., Mathiassen, L. & Lindgren, R., 2017. Embracing Digital Innovation in Incumbent Firms: How Volvo Cars Managed Competing Concerns. Management Information Systems Quarterly, 41(1), pp.239–253.

Van De Ven, A.H. & Huber, G.P., 1990. Longitudinal Field Research Methods for Studying Processes of Organizational Change. Organization Science, 1(3), pp.213–219.

Vessey, I., Ramesh, V. and Glass, R.L., 2002. Research in information systems: An empirical study of diversity in the discipline and its journals. Journal of Management Information Systems, 19(2), pp.129-174.

Tilson, D., Lyytinen, K. and Sørensen, C., 2010. Research commentary—Digital infrastructures: The missing IS research agenda. Information systems research, 21(4), pp.748-759.

Tiwana, A. (2015). Evolutionary competition in platform ecosystems. *Information Systems Research*, *26*(2), pp. 266-281.

Walsham, G., 1995. Interpretive case studies in IS research: nature and method. European Journal of information systems, 4(2), pp.74-81.

Winter, S., Berente, N., Howison, J. and Butler, B., 2014. Beyond the organizational 'container': Conceptualizing 21st century sociotechnical work. Information and Organization, 24(4), pp.250-269.

Yoo, Y., Henfridsson, O. and Lyytinen, K., 2010. Research commentary—the new organizing logic of digital innovation: an agenda for information systems research. Information systems research, 21(4), pp.724-735.

Yoo, Y., 2010. Computing in everyday life: A call for research on experiential computing. MIS quarterly, pp.213-231.

Appendix A

Stage	Method	Informant source	Topics	Purpose
Introductory	Open-ended interviews	Organizational management, organizational coaches, Software division mid-management	-Reorganization processes, -Division of work, -R&D projects for AD	Gain knowledge of divisions dedicated to developing AD product lines, & of distribution of human resources and competences across developer teams
IVC-core1	Observations Oct 2018-June 2019	IVCcore engineers	Daily tasks and routines, challenges	Find themes, formulate interview guides, develop a narrative map
IVC-core 2	8 Semi-structured interviews	IVCcore engineers	Project, phases, goals, tasks, challenges	To capture *what* the team does to prepare or develop AD product lines
IVC-core3	8 Semi-structured interviews	IVCcore engineers	Planning procedures & structure, decision-makers, specification of requirements for projects	To capture *how* the team prepared for working in different projects and phases

Table 2. Data gathering phases