A Case Study of Health Service Platforms as solution patterns: preliminary results

LYSANNE LESSARD, University of Ottawa
MARK DE REUVER, Technical University of Delft
RAPHAËL MÉNARD-GRENIER, University of Ottawa

Health service platforms (HSPs) are digital platforms that leverage the capabilities of digital technologies and infrastructures to innovate traditional care and research practices in a manner that increases patient participation, access to care, care efficiency, health outcomes and more. Despite relying on the same technologies as market-oriented platforms for product innovation or economic transactions, the purpose of HSPs is fundamentally different; HSPs focus on improving health outcomes for platform users, not on increasing revenue through economies of scale. Hence, traditional descriptions of digital platform architecture may not be sufficient to guide the design of HSPs. We propose to describe HSP architectures as solution patterns generating valuable resource configurations in response to actors' problems. This description should better reflect how these architectures address platform stakeholders' problems than the traditional articulation of digital platform architectures as layered modular components. This proposition will be examined through a multiple-case study aiming to identify architectural patterns in HSP platforms. We present preliminary results from the application of case study methods to the published description of a HSP platform. We also provide a template that could be used to identify solution patterns in other HSPs. The results of this study could help professionals designing or evolving HSPs as well as health providers to identify what should be part of their HSP structure and how it should be organized to meet desired health outcomes.

•Software and its engineering~Software notations and tools~System description languages~Architecture description languages

Additional Key Words and Phrases: Digital Platform, Health Service Platform, Case Study, Architectural Pattern

ACM Reference Format:

Lessard, L., de Reuver, M., and Ménard-Grenier, R. 2020. A Case Study of Health Service Platforms as solution patterns: preliminary results. HILLSIDE Proc. of Conf. on Pattern Lang. of Prog. 26 (October 2019), 12 pages.

1. BACKGROUND

The way in which digital platforms enable innovation and support market domination has been studied extensively in fields such as economics, management and information technology (de Reuver, Sørensen, and Basole 2018, Gawer 2014, Van Alstyne, Parker, and Choudary 2016). Part of this literature has focused on the architecture of digital platforms, thought to be key to their innovation capabilities. In particular, a seminal article on this topic argues that digital platforms are able to foster the development of novel products of services because their architecture combines the modular structure of physical products with four loosely coupled digital technology layers (devices, networks, services, and contents) (Yoo, Henfridsson, and Lyytinen 2010). A modular architecture reduces complexity and increases flexibility by decomposing a product into loosely coupled components interconnected through standardized interfaces (Baldwin and Woodard 2008). Components, or modules, are viewed as core or complementary, core modules providing generic, stable functions and peripheral modules providing varied and specialized applications. Standardized interfaces govern the behavior of the platform by determining how core and complementary modules interact (Baldwin and Woodard 2008). Instantiations of interfaces such as APIs, tools, and regulations act as distributed control mechanisms, allowing innovations within certain constraints (Mohagheghzadeh and Svahn 2016). The loosely coupled digital layers allow varied technologies to co-exist within a platform. Thus, developers can choose individual components within one layer without consideration for the other layers. Organizing information technology modules within digital layers results in a layered modular architecture that allows components to be combined through standards and protocols, within and across layers, in unpredictable ways. Thus, the layered modular architecture of digital platforms is taken to facilitate the creation in novel digital products and to be key to a platform's

This work is supported by a Telfer School of Management Research Grant.
Author's address: L. Lessard, 55 Ave. Laurier E. Ottawa, Ontario, Canada, K1N 6N5; email: lessard@telfer.uottawa.ca; M. de Reuver, Building 31, Jaffalaan 5, 2628 BX Delft, Netherlands; email: g.a.dereuver@tudelft.nl; R. Ménard-Grenier, 55 Ave. Laurier E. Ottawa, Ontario, Canada, K1N 6N5; email: rmena008@uottawa.ca.

success. The layered modular architectural framework has become the *de facto* approach to describing digital platform architectures in fields such as Information Systems, becoming an implicit basis for their design (e.g., Keijzer-Broers, Florez-Atehortua, and de Reuver 2016, Spagnoletti, Resca, and Lee 2015).

This approach to describing digital platform architecture has been labeled the "hierarchy-of-parts" frame, since it views an architecture as the functional *decomposition* of a complex system into *parts* arranged in a *hierarchy* that can be *aggregated* through *interfaces* (Henfridsson, Mathiassen, and Svahn 2014). While modularity increases design flexibility by allowing reusing and recombining components, the economies of scale resulting from this approach only accrue if design specifications are frozen for periods of time.It has thus been criticized for not truly reflecting the pace at which technological change happens in digital technology. Moreover, digital products can be easily reprogrammed and reproduced at virtually no cost, questioning the need for modular design as a means to generate economies of scale. An alternative perspective, referred to as the network-of-patterns frame, has thus been proposed (Henfridsson, Mathiassen, and Svahn 2014). This approach, which has its roots in the work of Christopher Alexander (Alexander 1979), views an architecture as a loosely-coupled *network* of patterns that address system complexity by providing *general* solutions that can be *specialized* to local problem contexts by selectively *inheriting* the properties of one or more generic solution. From the standpoint of technological change, the network-of-patterns approach thus allows capturing the (re)configurability of components within digital platforms. The car industry provides an interesting example of the motivations and benefits of moving from a hierarchy-of-parts frame to a network-of-patterns frame to manage technological change (Henfridsson, Mathiassen, and Svahn 2014). Indeed, most car manufacturers now approach the architecture of their car infotainment systems through a catalog describing available functional patterns with specifications on how to instantiate them in software and how they can be combined to generate more specific functionalities. This allows them to orchestrate shared systems resources at the systems level through rules for how to make use of hardware and software components such as speakers and displays. The use of network-of-pattern approach was also motivated by the need to more easily and rapidly relate software-enabled functionalities to market demands, which was difficult to achieve using more traditional architectures focusing on supplier-provided components that tightly coupled software functionality and hardware components. Such components could be combined as-is, but their functionalities could not be selectively used by other components.

In this work, we draw on the network-of patterns approach to operationalize the Service-Dominant Logic (S-D Logic) understanding of digital platforms and their architectural descriptions in the domain of health (Lusch and Nambisan 2015). S-D Logic, first proposed by management scholars (Ramirez 1999, Vargo and Lusch 2004) but now adopted in a number of fields including Computer Science, Software Engineering, and Information Systems (Lopes and Pineda 2013, Fragidis and Tarabanis 2011, Demirkan and Dolk 2013), has become a generally agreed-upon conceptual foundation for better understanding how value is created through service-for-service exchanges in modern economies. From the perspective of S-D Logic, digital platforms are actually service platforms that enable actors to interactively create and integrate new resources in a manner that creates value for them. Innovation then stems from the capabilities that actors gain from integrating new resources through the use of the platform, rather than from the ability to create novel digital products. This perspective builds on the understanding of digital platforms as modular structures that can be flexibly adapted (Gawer 2014, Yoo, Henfridsson, and Lyytinen 2010), but focuses on platforms' role in facilitating access to digitized resources, as well as their reconfiguration into novel solutions to actors' problems. As such, S-D Logic is highly compatible with the network-of-patterns approach. S-D Logic provides the core concepts to capture the way in which a digital platform may create value for its stakeholders, while the network-of-patterns approach helps to operationalize these concepts in a manner that allows empirical investigations of digital platforms' architectures. We combine these two conceptual pillars in order to arrive at better architectural descriptions of digital platforms in the domain of health, referred to as Health Service Platforms in the remainder of this paper.

2. RESEARCH OBJECTIVES

Health service platforms (HSPs) are increasingly being used in the health domain for a variety of purposes such as disease prevention, remote patient care and monitoring, and research (Sanchez et al. 2018, Aledavood et al. 2017). HSPs leverage the capabilities of digital technologies and infrastructures to innovate traditional care and research practices in a manner that increases patient participation, access to care, care efficiency, health outcomes and more. These objectives are achieved in part through their technological architecture, which follows the general principles of digital platforms in the sense that they adopt a layered modular architecture that enables digital product innovation and facilitates transactions (Gawer 2014, Yoo, Henfridsson, and Lyytinen 2010).

However, the purpose of HSPs goes beyond a product innovation or transaction logic (Thomas, Autio, and Gann 2014). Instead, they focus on enabling healthcare delivery in an effective and efficient manner in order to ultimately improve health outcomes. Hence, traditional descriptions of digital platform architecture, such as the layered modular architecture, may not be sufficient to capture the essential features of HSPs. Indeed, such an approach to architectural understanding does not capture the service elements that can fulfill the healthcare-related intentions of HSP stakeholders. For example, when health providers want to evolve an existing platform used for remote patient care and monitoring, they need to ensure that changes to the platform are made in a way that maintains or improves patient outcomes; to do so, they need to be able to relate patient-facing functionalities and system behaviors with core changes being made within the platform. The layered modular approach to describing a platform's architecture helps to understand how to enable the development of innovative digital products at large, but it does not help to differentiate between innovations (or combination of modules and layers) that are appropriate for patients' needs from those that may be useless or harmful. Developing an approach to describing HSP architectures in line with their purpose is a first step toward providing a systematic way to identify architectural requirements – which fundamentally shape the functioning and capabilities of a platform – for a new or evolving HSP.

We thus propose to undertake a case study aiming to answer the following research question: "How can HSP architectures be articulated in a manner that reflects their health-related purpose?" As a first step in this study, we present the methodology and preliminary results of a pilot case study, where the proposed approach is applied to published literature reporting on a HSP and its architecture.

The proposition being examined within the scope of this study is that HSP architectures can be articulated as series of solution patterns able to create value from the perspective of HSP stakeholders (see Figure 1). If the stated proposition is correct, it would entail that a solution patterns approach would better capture the purpose of a given HSP architecture at any given moment than the traditional layered modular approach. In other words, the solution patterns approach should be able to capture knowledge about how and why architectural components should be used to address stakeholders' problems, not only what these components can do as is the case for the hierarchy-of-parts approach. Moreover, it should be able to better capture how a given HSP architecture evolves through time. Indeed, from a S-D Logic perspective, the evolution of HSP architectures would be driven by changing stakeholder expectations of value and will to improve their offerings (Saarikko 2015). This perspective implies that the evolution of an HSP platform architecture would lead to modification in its solution patterns and in the problems they aim to address. From the traditional layered modular perspective, HSP architectures evolve through new combinations of digital and physical components, so as to enable the creation of innovative digital products that serve to attract a growing use base. This perspective implies that evolution in an HSP architecture could be adequately captured through the identification of changes in modules within and across layers.

3. METHODOLOGY

The case study is guided by a framework derived from S-D Logic (Lusch and Nambisan 2015) and the network-of-patterns approach to architecture (Henfridsson, Mathiassen, and Svahn 2014). The guiding framework illustrated in Figure 1 shows how the "network-of-patterns" approach to digital platform architectures anchors the S-D Logic view of service platforms. In the framework, the service ecosystem level where networks of actors engage to find solutions to their problems is taken to result in the definition of problems-in-context for these actors. The service platform level is where offerings composed of digital components (e.g., software) and resources (e.g., data) are defined and combined to form general solution patterns able to address actors' problems. From the perspective of S-D Logic, a successful solution pattern is one that offers a combination of useful resources from a given actor's perspective; in a platform, the ability to easily recombine resources in novel ways to address actors' problems is enabled by the digitalization of resources, thus their decoupling from software or devices. The value cocreation level is where new configurations of resource, created through the instantiation of general solution patterns, are integrated by network actors. However, the fundamental principles of S-D Logic imply that reconfigured resources are not valuable *per se*, but are rather part of a process that may lead to perceived value by the beneficiaries of these solutions if and when they are integrated within beneficiaries' existing resources and activities (Vargo and Lusch 2008). This is in line with S-D Logic's beneficiary-centric definition of value, and HSP's purpose of enabling healthcare delivery in a manner that may improve health outcomes.

The case itself is defined as the HSP architecture. The unit of analysis for the case is solution patterns, each understood as an arrangement of tangible and intangible components that enable the leveraging and recombination of digitized resources. Solution patterns thus stand as embedded units of analysis within the case (Yin 2014). The HSP architecture is taken to be part of a wider organizational form, the architecture's context, where agents shape a platform's interfaces, coordination mechanisms, and accessible capabilities (Gawer 2014, Van Alstyne, Parker, and Choudary 2016). For example, the medical program in which a platform is situated can serve as the context of a HSP case.

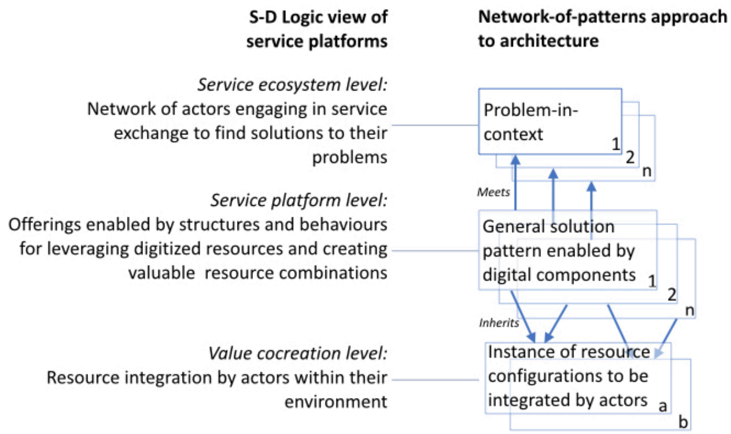

Fig. 1. Guiding framework of HSP architectures.

The guiding framework illustrated in Figure 1 serves as a starting point to investigate and represent HSP architectures as series of solution patterns able to create value from the perspective of HSP stakeholders. It is thus used to guide data collection and analysis. Hence, categories of data to be sought for each case include those related to platform stakeholders and their relationship with the platform; the resource bundles made available through the platform; the HSPs technical make-up, governance and rules; and, the evolution of the HPS over time. Data will be collected both as they relate to the HSP architecture, thus at the level of the case itself, and to solution patterns, thus at the level of embedded units. For example, data will be sought about the rules and regulations pertaining to the platform as a whole as well as to specific services offered by the platform. Within and across these categories, data collection will focus on uncovering relationships among various elements of the HSP in order to identify architectural patterns.

Table 1 Data to be collected for the pilot study

DATA CATEGORIES		Topics
CONTEXT		Purpose of the platform within the broader health program HSP stakeholders and their roles in relation to the platform Responsibilities and rights attributed to each role Coordination mechanisms allowing the platform's functioning and evolution Rules and regulations related to the platform Modifications to any of the above over time
STRUCTURE		Architectural structure, including layers, components, and modes of communication among layers and components and with other platforms or systems (e.g., Application Programming Interfaces) Modifications to any of the above over time
SOLUTION PATTERNS	PROBLEMS	Motivations for using or offering the platform (problems and goals) Constraints or issues that are related to problems and goals Modifications to the above over time
	OFFERINGS	Services offered by platform to varied stakeholders (e.g., health provider, patients) Service-specific rules and regulations, and their role Service-specific data or content, and their role

		Service-specific software components, and their role Service-specific devices, and their role Modifications to any of the above over time
	INTEGRATION	How each offering is used by relevant stakeholders Value derived by relevant stakeholders in relation to each offering Challenges to deriving value from each offering; impact of these challenges Modifications to any of the above over time

In this paper, we apply the methodology to published literature describing a HSP and its architecture (Ioanaa, Constantin Luciana, and Cosmin Septimiua 2018). In the full empirical study, data sources will include interviews with platform stakeholders (e.g., clinicians, administrators, patients, commercial platform provider), platform documentation (e.g., project plans, functional architecture description), and observation of the platform's functioning (e.g., through the creation of a user profile on the platform). Participants to be recruited for the case study include clinicians and administrators of the health provider using and offering the platform as part of a preventive and rehabilitation medical program, system architects and business analysts working for the company that has developed and is evolving the digital platform, and patients enrolled in the health provider's medical program using or having used the platform.

Data analysis relies mainly on deductive coding, thus on the application of pre-defined codes (the topics identified in Table 1) to collected data. However, specific solution patterns act as emergent codes, since they are identified in data rather than being derived from the guiding framework. Moreover, we remain open to additional emergent codes, thus to the identification of codes that were not anticipated at the time of data collection. As a result, the approach to architectural description developed through this study may refine the guiding framework or significantly modify it. Coding is to be done in two phases. In the first phase, each topic identified in Table 1 is used as a code that is applied to collected data. Emergent codes could be identified at this phase. In the second phase, codes are to be grouped as higher-level themes (e.g., types of stakeholder roles), patterns (e.g., a given solution pattern), or sequences (e.g., modifications made to rules and regulations over time). These higher-level codes are compared to the proposed framework and its rival theory in order to assess which one best described the studied HSP architecture and its evolution over time.

4. PRELIMINARY RESULTS

This section presents preliminary results from the pilot study, which applies the methodology described in the previous section to a published article describing a Health Service Platform, the Ambient Assisted Living (AAL) platform (Ioanaa, Constantin Luciana, and Cosmin Septimiua 2018). This article was chosen because it provides a detailed description of the platform and its architecture, enabling at least a partial application of the hierarchy-of-parts and networks-of-patterns perspectives. Since the purpose of the pilot study is to validate the methodology to be used when conducting empirical studies of Health Service Platforms, the results should not be taken as critiques of the article, but as a means to discuss the benefits and limitations of the proposed methodology. In the remainder of this section, we first describe the context of the AAL platform, followed by a brief overview of its architecture and functioning as presented in the article. We then describe the platform's architecture through the hierarchy-of-parts perspective, thus as layered modular components, and then through the network-of-patterns perspective, thus as a series of solution patterns. The ability of the proposed methodology to identify solution patterns is then be discussed, as well as some issues related to application of the guiding framework and proposed methodology.

The purpose of the AAL platform is to address the main requirements of the European Active and Assisted Living program, a funding program for information and communications technology (ICT) solutions that can improve the quality of life of older people and their caregivers. These requirements include maintaining the health and a pleasant living environment for the elderly; assisting them in their daily activities; and, connecting them with their caregivers. The AAL platform's main stakeholders are its primary users (elderly), secondary users (formal caregivers such as physicians and informal caregivers such as family), and tertiary users (application and service providers). Each type of user represents a role level with its associated permissions. For example, while primary users can visualize and manage medication alerts related to their treatment plan, secondary users can also create new treatment plans based on the analysis of primary users' health parameters and thus generate new alerts. Tertiary users have an administrative role in the platform, which allows them to manage user accounts and add custom services among others.

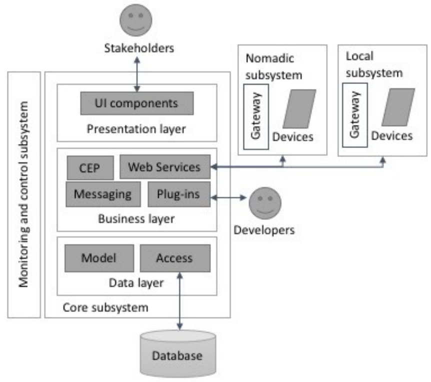

Fig. 2. AAL platform architecture (Adapted from Ioanaa, Constantin Luciana, and Cosmin Septimiua 2018)

The architecture of the AAL platform contains four sub-systems (see Fig. 2 for a simplified representation). The first two, the Local and Nomadic Subsystems, reflect the wired, wireless and wearable devices used to generate health and environmental data about primary users; each sub-system also contains a gateway that sends sensor measurements to other sub-systems using wireless or mobile networks. The framework is the name given to the two other subsystems, the Core Subsystem carrying system control and database functions, and the Monitoring and Control Subsystems offering functions such as service data management and data presentation to end users. The Monitoring and Control Subsystems is organized in three layers. The first one is the Data Layer containing a data model module as well as a data access module connecting to the platform's database. The second is the Business Layer containing workflows, business, complex event processing, messaging, Web services and OSGi plug-ins modules. The third layer is the Presentation Layer consisting of a Web module containing UI (user interface) components. This architecture aims at supporting uses cases identified as priorities for assisted living, namely in-home health monitoring, house monitoring, reminders, fall down detector, and indoor mobility patterns. It also allows OSGi (Open Service Gateway Initiative) plug-ins to be deployed on the platform without downtime; these service extensions can then be associated with a given elderly (primary user) by a caregiver (secondary user).

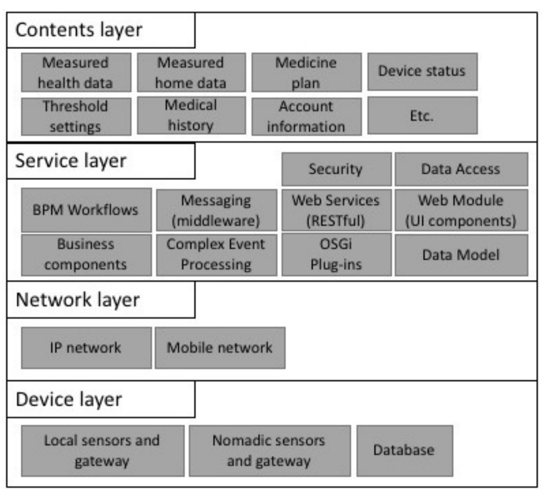

Fig. 3. The AAL platform as layered modules

The text in the article relating to the AAL platform's architectural structure, summarized in the previous paragraph, was analyzed in terms of the four layers identified within the layered modular framework of digital architecture (Yoo, Henfridsson, and Lyytinen 2010): devices, networks, services, and contents. The device layer includes physical machinery such as computer hardware and logical capabilities such as operating systems. This layer thus connects physical machinery to other layers. The network layer includes physical transport equipment such as cables and logical transmission capabilities such as network standards. The service layer contains functionalities that enable users to create, manipulate, store, and consume contents. The content layer includes any data stored and shared on the platform, such as texts, sounds, images and videos. Applying the layered modular framework to the AAL platform as described in the article allows capturing the modules described in the original paper, but it redefines the AAL platform's layers (see Fig. 3).

The article's text was also coded in terms of the problem, offerings, and integration topics presented in Table 1 in order to identify solution patterns. Extracted data were grouped in terms of two use cases directly identified in the article (health monitoring and home monitoring), and in terms of fall down detection and mobility patterns, which we refer to as human behavior monitoring. Within each use case, each type of identified problem was associated with the solution-specific functionalities, supporting functionalities, data, architectural components, and use of the solutions made by stakeholders. As a result, a number of solution patterns were identified for each use case. Table 2 presents the data extracted for solution patterns identified for health monitoring, while Table 3 and Table 4 present the data extracted for solution patterns for home and behavior monitoring respectively. Each pattern contains only partial data in terms of the framework guiding this study, since the article analyzed only presented assumed uses of the platform for the first two cases, and some patterns were not described in as much depth as the others (e.g., "Monitoring" pattern in Table 4). Nevertheless, Tables 2 to 4 demonstrate the ability of the framework to capture elements of a solution pattern.

Taking the example of health monitoring shown in Table 2, two core problems were identified: 1) keeping abreast of elderly's health status, and 2) quickly knowing if health status becomes abnormal. For the first problem, two solutions were identified: "Health Reminding", concerned with reminding elderly and caregivers of elderly's measurement and treatment schedules; and, "Health Monitoring", concerned with giving access to caregivers (and elderly, in a more limited manner) to data visualizations and reports of elderly's health status. The solution to the second problem was named "Health Alerting" and focuses on generating and transmitting alerts to caregivers when elderly's health measurements exceed pre-defined threshold. Within each use case, this approach allowed the identification of functionalities that directly help to solve identified problems and supporting functionalities that are used by multiple solution patterns, such as "Role management" and "Devices management". It also allowed to identify which data and architectural components were used by which patterns. For example, the component "Complex event processing" is used as part of "Alerting" patterns within each case, but not within the other patterns.

Table 2 Extracted data related to health monitoring solution patterns

Problems	**Keep abreast of elderly's health status**		**Quickly know if health status becomes abnormal**
Solutions	**HEALTH REMINDING**	**HEALTH MONITORING**	**HEALTH ALERTING**
Solution-specific functionalities	Elderly measurement management Elderly treatment management	Data visualization & report creation	Abnormal health measure alert
Supporting functionalities	Role management Medical profile Agenda Reminders generation Reminders management Reminders visualization	Role management Medical profile Device management Measures management	Role management Device management Measures management Agenda Alerts management Alert generation
Data used within solution	Elderly medical history Medicine plans	Measured health data Elderly medical history Medicine plans	Measured health data Hard-wired measurement thresholds Customized measurement thresholds

	Elderly account info	Elderly account info	Elderly account info
Architectural components used within solution	Web modules Database Business workflow Data access Gateway Devices Security Web services Business components	Web modules Database Business workflow Data access Gateway Devices Security Web services	Complex event processing Web modules Database Data access Gateway Devices Security Web services Business components
Assumed uses of solution by actors ("C" for caregivers, "E" for elderly)	C - Schedule health parameters monitoring plan C - Set and receive reminders for elderly care C - Insert new medicine plans E - Receive medicine taking reminders E - Receive health parameters monitoring reminders	C&E -Supervise health parameters C - Monitor progress based on plans	C - Take action based on health alert

Table 3 Extracted data related to home monitoring solution patterns

Problems	**Increase elderly safety**		
Solutions	**HOME MONITORING**	**HOME ALERTING**	**HOME CONTROLING**
Solution-specific functionalities	Data visualization and Report creation	Abnormal doors & Windows Status Abnormal home measure alert	Devices control
Supporting functionalities	Devices management Measures management	Alerts management Devices management Measures management	Devices management Measures management
Data used within solution	House sensor data Doors/windows status	House sensor data Measurement thresholds Customized Thresholds Doors/windows status	House sensor data Hard-wired measurement thresholds Customized measurement thresholds Doors/windows status
Architectural components used within solution	Web modules Data base Data access Gateway Devices Security Web services	Complex event processing Web modules Database Business components Data access Gateway Devices Security Web services	Web modules Database Data access Gateway Devices Security Web services
Assumed uses of solution by actors ("C" for caregivers, "E" for elderly)	C - Take action to assure and improve elderly's quality of life	E - Take action based on doors & windows status alert	C&E- Change windows & door status remotely

Table 4 Extracted data related to human behaviour monitoring solution patterns

Problems	**Keep abreast of elderly movement routine**	**Quickly know of abnormal movement** • Optimize response time by minimizing time needed to generate alerts
Solutions	**BEHAVIOUR MONITORING**	**BEHAVIOUR ALERTING**
Solution-specific functionalities	Indoor mobility pattern	Fall down detector Abnormal mobility pattern alert
Supporting functionalities	*No data available*	Alert generation
	No data available	Hard-wired measurement thresholds

Data used within solution		Customized measurement thresholds Sensors measures Measured values
Architectural components used within solution	*No data available*	Complex event processing Web modules Data base Devices Data access Gateway Web services Business components UWB positioning system Security
Assumed uses of solution by actors ("C" for caregivers, "E" for elderly)	*No data available*	*No data available*

Case-specific problems and solutions were then compared across cases. Recurrent solutions containing significant overlap were considered to demonstrate evidence of a solution pattern. As a result, four main solution patterns were identified for the AAL platform: Reminding, Monitoring, Alerting and Controlling. Other solution patterns were identified but were not presented in sufficient details in the article to be included here (e.g., a "plug-in" pattern allowing developers to deploy new services within the platform). Within each pattern, supporting functionalities, common data types and architectural components were taken to pertain to the solution pattern, while functionalities, data and components that were only used within case-specific solutions were taken to pertain to applications of the solution pattern (referred to as instances of resource configurations in the guiding framework). Fig. 4 maps the main elements of two solution patterns, Alerting and Monitoring, to two applications that inherit their properties to address the problem of quickly knowing about any issue related to the elderly. The illustration helps to understand that solution patterns and their applications are related through their use of system-wide functionalities, data and modules, resulting in a semilattice model characteristic of organized complexity (Gabriel and Quillien 2019). A full mapping would integrate actual instances of an application, thus who used a given solution, why, and with which consequences, but these data were not available in the article analyzed. Such data will however be sought in the larger case study.

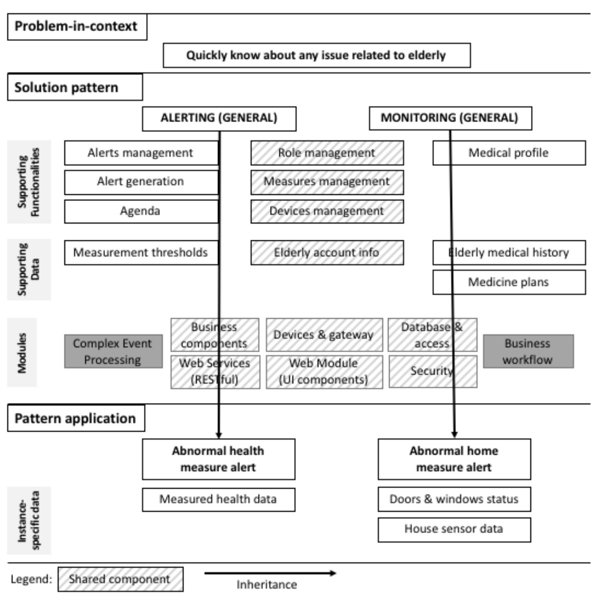

Fig. 4. Examples of applications of two solution patterns in the AAL platform

The solution patterns identified for the AAL platform can be described using a typical pattern format by analyzing the data extracted for case-specific solutions. As stated above, (generic) solutions can be identified through the common components, data, and functionalities across cases. The problem and the forces being addressed by the solution are also identified through cross-case comparison. The context of the solution is derived from data related to assumed uses of the solution, while known uses of the pattern are drawn from solution-specific functionalities. Related patterns are identified through the use, within a case-specific solution, of components, data, or supporting functionalities being shared with another solution. An example is given below, based on data related to the "Alerting" pattern (Ioanaa, Constantin Luciana, and Cosmin Septimiua 2018):

- *Context*. This pattern is to be used when caretakers want to be alerted about any abnormal status related to elderly living at home, in order to take appropriate actions.
- *Problem and forces*. Need to quickly be made aware of issues related to an elderly's abnormal health, home or behavioral status; need to minimize the time necessary to generate alerts.
- *Solution*. Generate alerts based on measurements thresholds, using a Complex Event Processing component to process in near real-time the data received from devices or sensors.
- *Consequences*. Benefits: Caretakers' response time is optimized. Disadvantages: *none stated in data.*
- *Known uses*. Abnormal health measure alert, abnormal home measure alert.
- *Related patterns*. Monitoring – Needed to set and monitor baseline measures and patterns.

The "Alerting" solution pattern demonstrates the ability of the guiding framework and its supporting methodology to lead to the identification of patterns that respond to problems-in-context; however, more data on the forces constraining or requiring trade-offs and their context would be needed to choose or design the right solution pattern for a given problem-in-context. Nevertheless, the identification of related patterns allows to understand the architecture of a HSP as a network of interrelated patterns. These preliminary results hence demonstrate that articulating HSP architectures as series of solution patterns enables the understanding of how a HSP's architecture addresses its health-related purpose. Moreover, these results can be contrasted with the application of the layered modular framework, which helps to capture the main components of the AAL platform architecture but not the way in which these components are used to address specific problems-in-context such as quickly knowing about any issues related to the elderly. Nevertheless, the two frameworks could be used in a complementary manner, for example by classifying modules in terms of service, network and device layers within general solution patterns.

5. DISCUSSION

While piloting the methodology by applying it to a published case of HSP mainly aimed at ensuring that its application resulted in relevant results, the template developed to support data analysis could be used as a tool to identify solution patterns in HSPs more generally (see Table 5). Since the template was developed as part of a multi-case study research protocol, it may be most useful when aiming to identify solution patterns from use cases. Moreover, the following heuristics, derived from the data collection and analysis methods used for the case study, could be used along with the template:

(1) Read all documents related to the HSP to gain an overall understanding of it.

(2) Write brief summaries of a) the platform being examined; b) the platform's context (e.g., the clinical program that it supports; and, c) solutions that are readily apparent in the documentation, such as services offered through the platform in relation to a given stakeholder need.

(3) From the template, create a table (or many tables) with one column per solution already identified.

(4) Go through each document and other source of information (e.g., notes from conversations with platform stakeholders, online platform demonstrations), extracting data related to each solution as specified in the template's rows. Additional solutions will likely be identified at this point. As such, it may be necessary to re-organize columns within or across tables, as relevant.

(5) Compare identified solutions and their elements to identify recurring or similar ones. Annotate table data to identify which elements (data, architectural components, etc.) are common across the latter, and which elements are specific to a given solution.

(6) Describe solution patterns present in the platform by analysing table data and their annotations in order to identify:

- Each generic solution, thus a label that identifies what is core to the solution, as well as its main functionalities, data types, and architectural components.
- The problems and forces that each solution addresses, including any specific rule or regulation that should be complied with.
- The context of use of each solution, thus what would motivate the choice of a solution, derived from the varied uses that have been made of each solution.
- The positive and negative consequences of each solution, derived from the perceived value and challenges of using them.
- Known uses of the solution, thus the solutions initially identified in the tables.
- Related solution patterns, thus other solution patterns that are used in combination with the one being described.

Table 5 Template for extracting data related to solution patterns in HSPs

Problems	*Description of problems being solved by solution, ideally from different actor perspectives. Any constraint or issue pertaining to the problem.*
Solutions	*SOLUTION NAME*
Solution-specific functionalities	*Functionalities that directly provide a solution to the problems stated above*
Supporting functionalities	*Functionalities that enable solution-specific functionalities*
Data used within solution	*Categories of data required to provide the solution*
Architectural components used within solution	*Architectural components required to provide the solution*
Uses of solution	*Description of how the solution is used, ideally from different actor perspectives*
Perceived value and challenges in using the solution	*Description of the way in which each solution creates value and challenges, ideally from different actor perspectives*

Despite the results of this pilot study and the template and accompanying heuristics that are provided above, a number of issues related to the application of the guiding framework remain. A first issue is related to the nature of identified patterns. While the network-of-pattern approach used to develop the guiding framework for this study referred to "architectural patterns" within digital platforms (Henfridsson, Mathiassen, and Svahn 2014), the integration of this approach with S-D Logic lead to the identification of socio-technical patterns rather than patterns of software architecture or organizational structure. This dimension is not emphasized in the results of this pilot study, since the article analyzed for this purpose did not provide information on the actual use or perceived value of the platform, which would help to understand how specific solution patterns are integrated in a beneficial manner by elderly and caregivers. Nevertheless, a solution pattern reflecting S-D Logic should contain both digital components and dimensions related to human perception of value and use. The meaning of "solution patterns" in this context may thus need to be formalized to improve their identification. A second issue is one of granularity, thus the appropriate level at which to define architectural patterns. For example, it could be argued that "devices management" is a pattern in itself within the AAL platform.

6. CONCLUSION

As Health Service Platforms (HSPs) are being increasingly used in the health domain, it is important to ensure that they are designed in a manner that leads to improved care efficiency and effectiveness. HSPs are more than facilitators of transactions and innovation; they should be designed and evaluated on how they ultimately enable new ways for actors to access and combine resources for desired health outcomes. As a first step towards that goal, the preliminary results of this study demonstrate an approach for describing HSP architectures in a manner that is aligned with their purpose, rather than their function. Moreover, the guiding framework of HSP architectures derived from S-D Logic and architectural solution patterns both helps to adapt the network-of-patterns approach for the service domain, and to operationalize S-D Logic for architectural analysis. In future work, this framework along with the proposed methodology will be applied to a number of empirical cases of HSP architectures. The results of this future study could extend current conceptualizations of digital platforms to the health domain, providing a systematic approach for designing or evolving HSP platform architectures in a manner that better meets patients' and other health care stakeholders' needs. Moreover, they could provide refined heuristics and tools to identify solution patterns in existing HSPs.

REFERENCES

Aledavood, Talayeh, Ana Maria Triana Hoyos, Tuomas Alakorkko, Kimmo Kaski, Jari Saramaki, Erkki Isometsa, and Richard K. Darst. 2017. "Data Collection for Mental Health Studies Through Digital Platforms: Requirements and Design of a Prototype." *JMIR research protocols* 6 (6):e110. doi: https://dx.doi.org/10.2196/resprot.6919.

Alexander, Christopher. 1979. *The timeless way of building*. New York: New York : Oxford University Press, 1979.

Baldwin, Carliss Y., and Jason C. Woodard. 2008. "The Architecture of Platforms: A Unified View." Harvard Business School Finance Working Paper No. 09-034.

de Reuver, Mark, Carsten Sørensen, and Rahul C. Basole. 2018. "The digital platform: a research agenda." *Journal of Information Technology* 33 (2):124-135. doi: 10.1057/s41265-016-0033-3.

Demirkan, H., and D. Dolk. 2013. "Analytical, computational and conceptual modeling in service science and systems." *Information Systems and E-Business Management* 11 (1):1-11. doi: https://doi.org/10.1007/s10257-012-0189-5.

Fragidis, Garyfallos , and Konstantinos Tarabanis. 2011. "Analyzing value co-creation in service systems: contribution from GORE." Proceedings of the 2011 ACM Symposium on Applied Computing, TaiChung, Taiwan.

Gabriel, Richard P., and Quillien, Jenny. 2019. "A Search for Beauty/A Struggle with Complexity: Christopher Alexander." *Urban Science*, 3 (2): 64; https://doi.org/10.3390/urbansci3020064

Gawer, Annabelle. 2014. "Bridging differing perspectives on technological platforms: Toward an integrative framework." *Research Policy* 43 (7):1239-1249. doi: https://doi.org/10.1016/j.respol.2014.03.006.

Henfridsson, Ola, Lars Mathiassen, and Fredrik Svahn. 2014. "Managing technological change in the digital age: the role of architectural frames." *Journal of Information Technology* 29 (1):27-43. doi: 10.1057/jit.2013.30.

Ioanaa, Stefan, Aldea Constantin Luciana, and Nechifor Cosmin Septimiua. 2018. "Web platform architecture for ambient assisted living." *Journal of Ambient Intelligence and Smart Environments* 10 (1):35-47.

Keijzer-Broers, Wally, L. Florez-Atehortua, and Mark de Reuver. 2016. "Prototyping a Health and Wellbeing Platform: An Action Design Research Approach." 49th Hawaii International Conference on System Sciences (HICSS), Koloa, HI, 5-8 Jan. 2016.

Lopes, Amit J., and Ricardo Pineda. 2013. "Service Systems Engineering Applications." *Procedia Computer Science* 16 (0):678-687. doi: http://dx.doi.org/10.1016/j.procs.2013.01.071.

Lusch, Robert F., and Satish Nambisan. 2015. "Service Innovation: A Service-Dominant Logic Perspective." *MIS Quarterly* 39 (1):155-176. doi: https://doi.org/10.25300/MISQ/2015/39.1.07.

Mohagheghzadeh, Amir, and Fredrik Svahn. 2016. "Transforming Organizational Resource into Platform Boundary Resources." 2016 European Conference on Information Systems, Istanbul, Turkey, Jun 12, 2016 – Jun 15, 2016.

Ramirez, Rafael. 1999. "Value co-production: Intellectual origins and implications for practice and research." *Strategic Management Journal* 20 (1):49-65. doi: https://doi.org/10.1002/(SICI)1097-0266(199901)20:1<49::AID-SMJ20>3.0.CO;2-2.

Saarikko, Ted. 2015. "Digital platform development: A service-oriented perspective." ECIS 2015.

Sanchez, Isabelle M., Lindsey Shankle, Marilyn T. Wan, Ladan Afifi, Jashin J. Wu, Frank Doris, Alisha Bridges, Marc Boas, Brian Lafoy, Sarah Truman, Ana-Maria Orbai, Junko Takeshita, Joel M. Gelfand, April W. Armstrong, Michael P. Siegel, and Wilson Liao. 2018. "Building a Citizen Pscientist: Advancing Patient-Centered Psoriasis Research by Empowering Patients as Contributors and Analysts." *Dermatology and therapy*. doi: https://dx.doi.org/10.1007/s13555-018-0242-5.

Spagnoletti, Paolo, Andrea Resca, and Gwanhoo Lee. 2015. "A design theory for digital platforms supporting online communities: a multiple case study." *Journal of Information Technology* 30 (4):364-380. doi: 10.1057/jit.2014.37.

Thomas, Llewellyn D. W., Erkko Autio, and David M. Gann. 2014. "Architectural Leverage: Putting Platforms in Context." *Academy of Management Perspectives* 28 (2):198-219. doi: 10.5465/amp.2011.0105.

Van Alstyne, Marshall, Geoffrey Parker, and Sangeet Choudary. 2016. "Pipelines, Platforms, and the New Rules of Strategy." *Harvard Business Review*:1.

Vargo, Stephen L., and Robert F. Lusch. 2004. "Evolving to a New Dominant Logic for Marketing." *Journal of Marketing* 68 (1):1-17. doi: https://doi.org/10.1509/jmkg.68.1.1.24036.

Vargo, Stephen L., and Robert F. Lusch. 2008. "Service dominant logic: Continuing the evolution." *Journal of the Academy of Marketing Science* 36 (1):1-10. doi: https:/doi.org/10.1007/s11747-007-0069-6.

Yin, Robert K. 2014. *Case study research: design and methods*. Fifth ed. Thousand Oaks: Sage.

Yoo, Youngjin, Ola Henfridsson, and Kalle Lyytinen. 2010. "Research Commentary—The New Organizing Logic of Digital Innovation: An Agenda for Information Systems Research." *Information Systems Research* 21 (4):724-735. doi: 10.1287/isre.1100.0322.

Received September 2019; revised October 2019; accepted February 2020