Artificial Intelligence for Engineering Design, Analysis and Manufacturing

cambridge.org/aie

Continuous cycles of data-enabled design: reimagining the IoT development process

Boyeun Lee, Rachel Cooper, David Hands and Paul Coulton

Lancaster Institute of Contemporary Art, Lancaster University, Bailrigg, Lancaster LA1 4ZA, UK

Research Article

Cite this article: Lee B, Cooper R, Hands D, Coulton P (2022). Continuous cycles of data-enabled design: reimagining the IoT development process. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **36**, e11, 1–15. https://doi.org/10.1017/S0890060421000299

Received: 31 October 2020
Revised: 2 August 2021
Accepted: 13 September 2021

Key words:
Conceptual model; data-enabled design; IoT; IoT design process; new product development

Author for correspondence:
Boyeun Lee,
E-mail: b.lee12@lancaster.ac.uk

Abstract

With the emergence of Internet of Things (IoT) as a new source of "big" data and value creation, businesses encounter novel opportunities as well as challenges in IoT design. Although recent research argues that digital technology can enable new kinds of development processes that are distinctive from their counterparts in the 20th century, minimal attention has been focused on the IoT design process. In order to contextualize New Product Development (NPD) processes for IoT, this paper comprehensively interrogates existing, and emerging development approaches for products, services, software, and integrated products, and several factors that affect designing IoT. This discussion includes the generic development process, the commonalities and differences of different development approaches, and processes. The paper demonstrates that only a few existing approaches reflect vital characteristics of networked artifacts or the integration of data science within the development model, which is one of the key attributes of IoT design. From these investigations, we propose "*The Mobius Strip Model of IoT Development ProcessI*," a conceptual process for IoT design, which is distinctive to others. The continuous loops of the IoT design integrate the attributes and phases of different processes and consist of two different development approaches and strategies. Understanding the particular attributes of the IoT NPD process can help novice and experienced researchers in both feeding and drawing insight from the broader design discourse.

Introduction

As contemporary competitive pressure and the pace of technological advancement increase, organizations encounter the challenges of increasing cost-efficiency, preempting competitors, and creating breakthroughs (Meyer and Utterback, 1995; Kessler and Bierly, 2002). In this context, New Product Development (NPD) is claimed as the principal determinant of competitive advantage (Clark and Fujimoto, 1991; Kleinschmidt and Cooper, 1991; Brown and Eisenhardt, 1995; Crawford, 1997; Alam, 2006) as well as the engine of renewal and survival (Andrews, 1975; Bowen *et al.*, 1994; Fairlie-Clarke and Muller, 2003) for many corporations. Accordingly, NPD has enjoyed remarkable attention in a diverse variety of fields for the past decades, including product development and innovation (Durisin *et al.*, 2010), business research and service innovation (Johne and Storey, 1998; Froehle *et al.*, 2000; Menor *et al.*, 2002; Blazevic and Lievens, 2004), and software engineering (Royce, 1970; MacConell, 1996). With the growing interest, the processes and methods of a physical product, service, and software development have evolved significantly over the late 20th century.

While the current literature adequately addresses valuable insights on different varieties of subject development, a new type of product has emerged associated with the Internet of Things (IoT). There have been vital opportunities for innovation by amalgamating sensors, actuators, and cloud computing with non-digital products and services (Xu, 2012; Yoo, 2013; Lasi *et al.*, 2014; Radziwon *et al.*, 2014). However, it has been revealed that nearly three-quarters of IoT developments are failing (Cisco, 2017) due to the lack of experience and understanding of IoT development (Reichert, 2017). There are extra layers of development complexity because IoT exists in larger network ecologies, bridging both the digital and physical worlds. IoT design is not simply the integration of IoT hardware-related features into software-based development. However, it requires creators to consider the complex ecologies and eternal data process holistically. If the complexity is effectively managed accordingly through the development process, in this case, IoT design may fully unlock value by selling physical products, providing customized services, and harnessing data arising from the product in use. The economic value of IoT is estimated to generate anywhere from \$2.7 to \$ 14.4 trillion in value by 2025 (Manyika *et al.*, 2013).

Nevertheless, the growing number of studies focused on IoT, and researchers from marketing and design argue that current NPD models are obsolete to be applied to IoT development (Speed and Maxwell, 2015; Ng and Wakenshaw, 2017). To move the field forward, an integrative understanding is required from the broader product development discourse. With this research, we aim to develop a conceptual process for IoT design that reconciles characteristics of NPD in current literature and the key attributes differentiating existing NPD to IoT design process and provide an integrative understanding of the field. This paper contributes to a body of existing knowledge on development processes. The first primary contribution of the research is to outline a comprehensive IoT design model to enable both practitioners and scholars to comprehend IoT development better. On the researcher side, access to a novel process model will allow researchers in different fields to develop a critical exploration in the subject area. On the other hand, practitioners would be able to create greater value by applying primary elements of the development model to their IoT design and development process.

To develop a conceptual model, a literature review was conducted in three stages. First, a structured literature review on the existing IoT NPD process was conducted to identify gaps in the literature. Then a comprehensive review of existing NPD, Systems Development Life Cycle or Software Development Life Cycle (SDLC), New Service Development (NSD), and integrated models was conducted as developing an integrative view of NPD models requires organizing the disparate literature into groups, and distinguishing, parsing, classifying, or categorizing an entity (Maclnnis, 2011). Finally, the relevant factors on the IoT NPD process were identified to be synthesized in the conceptual model.

As the first step of the review process, a structured literature review was conducted to explore existing NPD models for IoT systems. For the searching process, the following databases were used: Web of Science, ProQuest, ACM Digital Library, IEEE Xplore Digital Library, and Scopus. We included peer-reviewed studies that employed any type of research design but exceptionally included gray literature in the form of PhD theses. Studies that contain the model related to IoT development were included. We limited searches to papers published in English and between 2011 and 2021. The search terms used were as follows:

- "IoT" OR "Internet of Things" OR "internet-connected" OR "connected device*" OR "digitised product*" OR "connected product*" OR "digitised device*"

AND

- "NPD" OR "new product development" OR "design process" OR "development process" OR "NSD" OR "SDLC"

Keywords were searched for in the titles, abstracts, and indexed subject headings of articles. The initial electronic database search yielded a total of 69 published articles. Figure 1 shows a PRISMA chart that details how many studies were excluded at each stage of the review. After removing 17 duplicates, 52 were screened for eligibility, of which 39 met the inclusion criteria for the full-text review. Following the full-text review, two studies were included and using snowballing techniques, that is, a study of "references to references" (Wohlin, 2014), and two articles were added which ended up to the total number of four articles in the review.

Our review of existing literature on NPD began with an initial selection of critical review papers on innovation and development processes (Saren, 1984; Rothwell, 1994; Howard *et al.*, 2008; Durisin *et al.*, 2010; Eveleens, 2010; Matkovic and Tumbas, 2010; Gericke and Blessing, 2012; Papastathopoulou and Hultink, 2012). A comprehensive understanding of the existing NPD models was then supported by a manual investigation of abstracts and articles published in the selected journals: *Journal of Product Innovation Management*, *Research Technology Management*, *Strategic Management Journal*, *Journal of Service Marketing*, and *International Journal of Service Industry Management*. These are the leading journals in NPD publications in innovation management, management, service marketing, and service research (Page and Schirr, 2008; Papastathopoulou and Hultink, 2012). In order to encompass NPD models in the software and design domains,

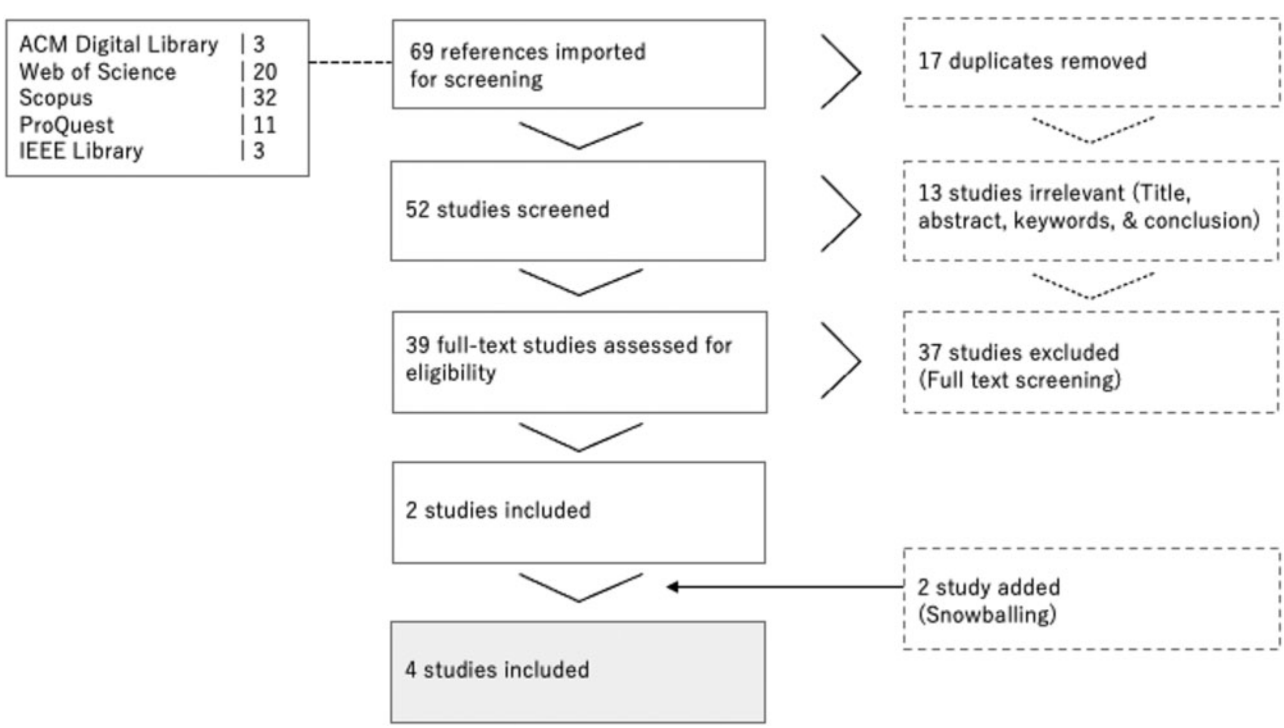

Fig. 1. Result from the structured review and the selection process.

https://doi.org/10.1017/S0890060421000299 Published online by Cambridge University Press

the relevant journals covering SDLC and design processes were also selected, such as the *Journal of Engineering Design* and *Journal of Software: Evolution and Process*.

The review of articles in the selected journals led to an additional search for a deeper understanding of mainstream NPD models. Thus, an additional literature search was conducted using the specific model names including “waterfall,” “v-model,” “agile,” “lean,” “concurrent,” “spiral,” “stage-gate,” and “PSS.” Primarily based on the critical review papers on the existing NPD models and the systematic reviews on the emergent NPD processes, the selected 36 models were identified (Table 1).

To extend our review, literature on relevant fields was more broadly searched by using another set of terms, including “digital innovation,” “big data,” “data science process,” “affordance,” and “material properties.” Using the key words as a starting point, the authors applied the snowball methods approach to critically examine relevant literatures. We refined our search to include journals, conference articles, academic texts, books, and white papers through electronic databases such as ProQuest Business Premium, Springer Journals Archive, and Wiley Online Library Journals. In terms of quality and relevance, a set of 135 articles were selected for analysis. Selected papers were grouped into two fields: (a) current theories, frameworks, and models on NPD processes ($n = 110$) and (b) the state of the art in the literature of data science processes and key factors of IoT development ($n = 25$). The first set of articles are employed to differentiate and integrate current theories and models of NPD across disciplines, and the rest is used to frame key factors and generic data science processes related to IoT design. Each text was thoroughly reviewed to contextualize NPD processes in IoT by drawing from the fields it intersects with.

We organized the paper as follows. First, the authors critically examined the established literature on the development processes across disciplines, discussing the commonalities and differences between the different development approaches, and the generic phases of the development process. Second, we identify the key factors and the generic data science practices that affect the IoT development process. Third, we link the relevance of the existing NPD to the theories of data science and digital innovation and propose the conceptual IoT development process. Finally, in the conclusion, we summarize the insights and the contributions of this study, as well as its limitations.

Established NPD processes

Definitions of different development models

Product is perceived distinctively depending on the fields. For instance, a product is regarded as a bundle of utilities, including intangible attributes in marketing theory (Kotler, 2000; Kahn, 2005; International Organisation for Standardization, 2015). In software engineering, Meyer (2001) proposed software as a product and a service. Sidi *et al.* (2012) and Huang *et al.* (2015) viewed data as a product, whereas Psomakelis *et al.* (2020) viewed data as a service. Whether they are a product or a service, in this paper, a product is defined as a broader concept, including physical and digital products/services/data regardless of their attributes.

Accordingly, the definitions of various development processes are compared and debated.

In innovation and product management theories, no single agreed definition of NPD exists. Krishan and Ulrich (2001) identify NPD as the transformation of a market opportunity and a set of assumptions about product technology into a product available for sale. Similarly, Ulrich and Eppinger (2011) define product development as a set of activities beginning with the perception of a market opportunity and ending in the production, sales, and delivery of a product. The definition of NPD by Bruce and Cooper (2001) is more inclusive, describing it to capture a range of disparate kinds of innovative activities leading to the production of a new service or product from radical innovations to simple modification and adaptations to existing products. NSD is defined as “the overall process of developing new service offerings” (Johnson *et al.*, 2000). This viewpoint is shared and extended by Edvardsson *et al.* (2000) to the extent that embraces strategy, culture, and service policy deployment and implementation.

In software engineering, NPD is referred to as SDLC. Matkovic and Tumbas (2010) define SDLC as a process of creating and adapting software products, as well as a basis for creating methodologies and models in software engineering. Similarly, the Department of Defense in the USA (1988) perceives it as the software development process for managing the development of the deliverable software including major activities, but with a focus on concurrent, recursive, and iterative development activities. Ruparelia (2010) identifies it as a conceptual framework or process that involves the development of an application from its initial feasibility study through to its deployment in the field and maintenance. This definition is distinguished from others by reflecting the recurring theme of software development within the process.

While examining definitions of New Product, Service, and Software Development, the commonality and difference of the definitions are identified. They are likely to be considered as a conceptual process encompassing required activities, but the extent of the activities seems to be different between the processes. Particularly, NPD and NSD are likely to involve the complete set of business activities from ideation to launch. Meanwhile, SDLC primarily focuses on software development activities from a technical perspective. Since IoT development is the hybrid of a physical product and software development, comprehending and comparing different terminology of different development processes enable us to revisit the definition of the development process of IoT. Subsequently, the next section explores current theories, frameworks, and models of development processes in order to identify the phases of the generic development process, and similarities and differences of different development approaches in relation to the IoT development.

Different approaches of NPD, NSD, SDLC, and integrated development

The waterfall model was highly influential (Boehm, 1995) until the traditional linear models were seriously being challenged in the 1980s (Berkhout *et al.*, 2010), and alternative approaches have emerged, including spiral, concurrent, and agile approaches (Fig. 2). Established NPD approaches are continuously evolving, supported by emergent trends of the increasing significance of NPD activities. The key attributes of different approaches are discussed in order to comprehend the emergent trend of NPD and value creation.

Sequential approach

Among the various types of product development models, the sequential approach is the most commonly used and conventional often referred to as “BAH model (Booz *et al.*, 1982),” “a stage-gate

https://doi.org/10.1017/S0890060421000299 Published online by Cambridge University Press

Table 1. List of the development processes across disciplines

Approach	Disciplines	Name of model	Researcher	Year	Source
Sequential Approach	Product Engineering	BAH NPD Model	Booz, Allen, & Hamilton	1982	Book: Booz, Allen & Hamilton
		Over the Wall Process	Walsh, V., Roy, R., Bruce, M., & Potter, S.	1992	Book: Blackwell Publishing
		Parallel Processing Model	Takeuchi, H. & Nonaka, I.	1986	Magazine: HBR
		Concurrent Engineering Model	Pennell, J., Winner, R., Bertrand, H., & Slusarszuk, M.	1989	Conference Proceedings: Automatic Testing Conference
		Stage-Gate System	Cooper, R.	1990	Magazine: Business Horizons
		Third-Generation Stage-Gate Process	Cooper, R.	1994	Journal: JPIM
		Activity Stage Model of NPD	Crawford, M.	1997	Book: McGraw-Hill
		Scalable Stage-Gate Systems	Cooper, R.	2008	Journal: JPIM
	Software Engineering	Program Production Visualization	Benlngton, H.	1956	Conference Proceeding: ON R Symposium on Advanced Programming Methods for Digital Computers
		The Traditional Waterfall Process	Royce	1970	Conference Proceeding: IEEE WESCON
		V-model	Forsberg, K. & Mooz, H.	1991	Conference Proceeding: NCOSE symposium
		The b-model	Birrell, N. & Ould, M.	1985	Book: Cambridge University Press
		Sashimi Waterfall Model	Mcconnel, S.	1995	Book: Microsoft Press
	Innovation	Open Innovation Model	Chesbrough, W.	2004	Conference presentation: 10th Annual Innovation Convergence
	Design	Double Diamond Process	Design Council	2007	Report: design council
Spiral	Product Engineering	Evans' Model of Ship Design Process	Evans, J,	1959	Journal: Naval Engineers Journal
	Software Engineering	Boehm's Spiral Model of Software Process	Boehm, B.	1986	Journal: IEEE Transactions on Computers
		Multiple Convergent Model	Baker, M. & Hart, S,	1999	Book: Financial Times Prentice Hall
		Spiral Model	Unger & Eppinger	2009	Journal: IJPD
		Network Model of NPD	Trott, P.	2012	Book: Pearson Education
		MSF Process model	Microsoft Team	2003	White Paper: Microsoft
	Service	The Service Design and Management Model	Ramaswamy, R.	1996	Book: Addison-Wesley Pub. Co
		NSD	Johnson, Menor, Chase & Roth	2000	Book: Sage Publication
	Innovation	Cyclical Innovation Model	Berkhout, G., Hartmann, D., & Trott, P.	2010	Journal: R&D management
Agile	Product Engineering	Agile-Stage-Gate Hybrids	Cooper, R.	2016	Journal : Research Technology Management
	Software Engineering	Scrum	Schwaber, K. & Beedle, M,	2002	Book: Prentice Hall
		Lean Development	Poppendieck, M. & Poppendieck, T.	2003	Book: Addison-Wesley Professional
		Agile Software Development	Kent, B., Grenning, J., Martin, R., *et al.*	2001	Website: http://agilemanifesto.org

(*Continued*)

https://doi.org/10.1017/S0890060421000299 Published online by Cambridge University Press

Table 1. (*Continued.*)

Approach	Disciplines	Name of model	Researcher	Year	Source
Integrated process	Software Engineering	Defense Systems Software Development	US Defense Standard	1983	White Paper: Report. DoD-STD-2167
	PSS	Integrated Product and Service Design Processes	Aurich, J., Fuchs, C., & Wagenknecht, C.	2006	Journal: Journal of Cleaner Production
		PSS Design	Maussang, N., Zwolinski, P., & Brissaud, D.	2009	Journal: Journal of Engineering Design
	Design	A New Design Process for IoT Products and Services	Jacobs, N. & Cooper. R.	2018	Book: Routledge
		The Eight-Shaped Model	Janne, v. & Bogers. S,	2019	PhD Thesis
	Management	NPD Model	Hartsell, C., Mahadevan, N., Nine, H., Bapty, T., Dubey, A., & Karsai, G.	2020	Conference Prooceeding: Design Automation for CPS and IoT
	Information Systems	Big Data Analysis Model for Mmarket Analysis	Yu, S. & Yang, D.	2016	Conference Proceeding: International Conference on Network and Information Systems for Computers

system (Cooper, 1990)," "over the wall process (Walsh *et al*., 1992; Trott, 2012)," "program production process (Benington, 1956)," "the waterfall process (Royce, 1970)," "the V-shape life cycle (Forsberg and Mooz, 1991)," and "the service design and management model (Ramaswamy, 1996)." The attributes of the traditional sequential models are (a) the linear continuation of the process to the next stage is resolved by a review and approval in order to minimize investment risks and (b) fully developed requirement documents work as completion criteria for early phases. Accordingly, the sequential approaches are regarded as a lengthy process, high costs, difficulty in adaptation to uncertainty, inflexibility between phases, and inability to react to changes.

Concurrent approach

The concurrent approach stems from several factors observed in industries such as faster innovation, increasing inter-company networking, and the appearance of new technologies (Rothwell, 1993). "Parallel processing models (Takeuchi and Nonaka, 1986)," "Concurrent Engineering (Pennell *et al*., 1989)," "Activity-stage models (Crawford, 1997)," "Sashimi model (Mcconnel, 1996)," "Unger and Eppinger's spiral model (2009)," and "NSD model (Johnson *et al*., 2000)" are categorized in this approach. With the emphasis on the iterative feedback loops and overlaps of the phases, it is enabled to increase the speed of the development process, and the flexibility to react to changes or errors. More importantly, new philosophies of design are emerging to respond to the flow of new information on customer needs and preferences, allowing offerings to be more tailored, adaptable, and desirable to the market. In the marketing and innovation theories, the emerging trends of having new strategic partners (international joint ventures) and establishing comprehensive networks are explained with the value constellation model (Normann and Ramírez, 1994), open innovation (Chesbrough, 2004), and co-creation with the customer (Royce, 1970).

Spiral approach

The representative spiral approaches include "Boehm's spiral life-cycle model (Boehm, 1986)," "Microsoft Solutions Framework (Microsoft Team, 2003)," and "Unger and Eppinger's Spiral model (2009)." The key aspects of these approaches are that (a) risks should be assessed and monitored at each milestone and (b) complex communication and workflow are enabled. With strong approval, control, and flexibility in the process, the approach enhances the opportunities for avoiding development risks. However, the development process potentially becomes increasingly costly, as this approach requires a highly specific risk analysis to be undertaken.

Agile approach

Stability and predictability have long been highlighted as the core elements of successful development within the traditional NPD process. However, irrespective of the development subject, the shift to more rapid and lightweight processes becomes a

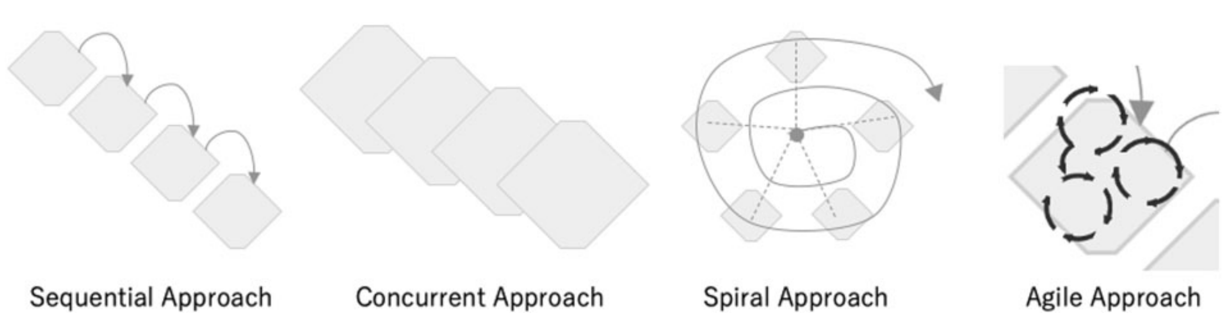

Fig. 2. Different approaches of development processes.

https://doi.org/10.1017/S0890060421000299 Published online by Cambridge University Press

dominant trend and consistent theme over time that is called the agile approach. Specifically, in software development, due to the fact that the continuous changes in system requirements occur over and during the process, this approach becomes popular. Overlaid on a traditional NPD process, they encompass the Agile development method (Martin, 2002), Lean Development (Poppendieck and Poppendieck, 2003), and the Scrum software development process (Schwaber and Beedle, 2002). These approaches (a) emphasize the small incremental releases for ongoing changes to the product/system specification and (b) de-emphasize on documentation and a formalized process-driven step. Although these approaches lack formal management, they enhance close communication and coordination activities, speed to market, and faster responses to changing customer requirements.

Integrated development approach

In an era of service-dominant logic (Vargo and Lusch, 2004*a*), the classical distinction between products and services is challenged (Lovelock and Gummesson, 2004; Vargo and Lusch, 2004*b*), and the boundary of digital and physical products are blurred. Consequently, the development approach for integrated offerings of product, software, and service has been an interesting subject among researchers. PSS methodologies, one of the widely recognized integrated development approaches, are proposed by Aurich *et al.* (2006) and Maussang *et al.* (2009), illustrating the systematic development of integrated product and service design. However, PSS methodologies are criticized as insufficient to describe the development details (Komoto and Tomiyama, 2009; Vasantha *et al.*, 2012) and not reflecting an entire life-cycle perspective (Mont, 2000; Welp *et al.*, 2008).

Although Jacobs and Cooper's new approach to IoT development (2018) and the eight-shaped model (Janne and Bogers, 2019) depict a continuous and never-ending IoT design process which reflects one of the main attributes of IoT development, they are limited in not integrating the whole NPD process nor fully reflecting the attributes of IoT design. The digitally-driven NPD model proposed by Yerpude and Rautela (2020) depicts the co-creation perspective of IoT NPD but it is not activity-based NPD model. The big data analysis model for market analysis (Yu and Yang, 2016) integrates big data analysis activities into the NPD process but does not illustrate the continuous feedback cycles.

Relevant differences between the approaches

Having compared the attributes of the different approaches, the relevant differences are thus identified: flexibility, a cross-functional approach, meaningful customer involvement, and rapid and lightweight processes. For example, the NPD processes with the agile approach are more lightweight than those adopting a waterfall approach. The Spiral approach combines the iterative characteristics with the waterfall model, in essence, allowing more flexibility and customer interactions. Although aspects of the models are distinctive, they are composed of a series of phases that must be followed and completed (Alshamrani and Bahattab, 2015). In the next section, the phases of different models are reviewed to identify the generic phases.

Generic phases of NPD processes

Over several decades, more than 600 diverse NPD models including all the variations of models have been developed by researchers with the aim of improving established NPD practices (Nijssen and Lieshout, 1995; Best, 2006). Several different development processes for physical products and engineering have been dominant within manufacturing economies, and then they started to be modified and developed specifically for service and software development. Through comprehensive interrogation of existing and emerging development processes for hardware, service, software, and combination of product and software (Table 2), the generic phases of the NPD process for this study are identified.

Interrogating the phases and activities of different processes, the generic phases of the development process are identified, which consists of six distinct phases. The brackets in each phase describe detailed activities of a physical product, service, and software development.

(1) Discovering users and business needs (Market research and analysis).
(2) Defining concepts of and strategies for business and technical solutions (System and Software requirements, Business model, and Concepts of software, product, and service).
(3) Testing feasibility of business and technical solutions (System and business analysis, Screening, and Evaluating the solution ideas).
(4) Designing, Prototyping, Integrating, and Testing solutions (Developing prototyping, Integrating and Testing plan, Coding, Debugging, and Casting),
(5) Manufacturing, Marketing, and Deploying solutions (Process development, Resetting organization, and Modifying product design for mass production).
(6) Maintaining, Evaluating, and Planning the next phase (Maintaining and customer support).

The generic phases of the development process are intended to cover the entire set of business activities from the transformation of market needs into a set of offerings to create organizational value, which is coherent with definitions of NPD and NSD. The last phase of the process, "Evaluating and planning the next phase," is additionally included reflecting the recurring theme of NSD and SDLC, since the term "products" in this research is not restricted to physical products but comprehensive of software, service, and data or the combination of all. This not only broadens the extent of the NPD process for IoT but also considers significant design aspects of the combination of physical and nonphysical products. The six generic phases will be used as a foundation for developing our conceptual IoT design process.

Commonalities and differences between development processes

Balancing between flexible iterations and structured review is one of the significant commonalities between development processes. Although all of the NPD, NSD, SDLC, and integrated development processes have managerial control and allow flexibility, the manner of reviews and iterations varies. Another determined commonality identified is that whether it is physical products, software, service, or integrated development processes, the initial stage of the process is characterized as fuzzy. These stages are often called the fuzzy front end (FFE) of an innovation process (Smith and Reinertsen, 1992) in which the corporations decide to build on an idea or not. Generally, including idea generation, opportunity validation, and concept development (Dewulf, 2013), FFE is widely recognized as critical, providing the foundations on which the overall development project is built (Stevens

Table 2. Phases, stages, or main activities of the NPD, NSD, SDLC, and PSS processes

	NPD						SDLC									NSD		PSS
Generic phases of NPD	BAH model	Over the wall	Stage-gate process	Activity stage model	Concurrent engineering	Double diamond process	Program production	Waterfall	b-model	V-shaped life cycle	Sashimi waterfall	Boehm's spiral lifecycle	MSF process model	Unger & Eppinger's spiral	Agile	NSD process	Service design & management model	PSS Design
1. Discovering users & business needs	New product strategy development	Market analysis Research		Strategic planning		Discover			Inception			Risk analysis				Formulation of service strategy	Defining design attributes	Customer needs External system analysis & usage scenarios
2. Defining concepts of and strategies for business & technical solutions	Idea generation	Product design	Idea	Concept generation	Requirement definition	Define	Operational plan Machine/ operational specifications Program Specifications	Requirements Software requirement	Definition	Define requirements	Software concept	Concept of operation Software requirement Requirements validation	Envisioning	Planning Concept design	Sprint planning	Idea generation Concept development & testing	Generating design concept	PSS function mapping approach PSS layout SI running scenarios
3. Testing feasibility of business & technical solutions	Screening & evaluation Business analysis		Preliminary assessment Business preparation					Analysis			Requirement analysis	Development plan	Planning			Screening ideas Business analysis	Evaluating design concept	External component analysis
4. Designing, prototyping, integrating, & testing solutions	Development Testing	Engineering	Development Testing 4 validation	Technical development	Product development Process development	Develop	Coding Parameter testing Assembly testing System evaluation	Program design Coding Testing	Design	System architecture design Detailed design Development coding Build & test	Architectural design Detailed design Coding & debugging Testing	Software product design Design validation & verification Detailed design & code Unit test	Developing	System level design Detailed design	Design Implementation test	Service design & testing Process & system design & testing	Developing design details	Component Development
5. Manufacturing, marketing, & deploying solutions	Commercialization	Manufacturing Sales	Full production & launch	Commercialization	Prototype & production	Deliver		Operation	Production Acceptance Operation	System Integration & test Deployments verification		Integration & test Acceptance test Implementation	Stabilizing Deploying	Integration & test Release	Release	Service testing and pilot run Test marketing & full-scale launch	Implementing the design	
6. Evaluating & planning the next phase									Evaluation			Plan next phase		Planning		Post-launch review	Measuring & Improving performance	

and Burley, 2003). Even though the SDLC is likely to cover technical aspects of activities, it still has fuzziness in the early stage of the processes, involving defining software requirements and software concepts. Accordingly, growing scholarly works indicate that the FFE shall be proactively managed and optimized to encourage successful innovations (Khurana and Rosenthal, 1997; Reinertsen, 1999; Kim and Wilemon, 2002).

Contrary to the commonality identified in the front end of the process, one of the differences between the processes is identified in the back end of the development. The NPD processes tend to have a beginning and an end of product development, as their final phases are *commercialization*, *sales*, or *delivering the products*. Meanwhile, the final phases of SDLC and NSD processes are *evaluation*, *post-launch review*, and *planning next*, which imply the continuous following loop of the next development process. A recurring theme of NSD and SDLC is attributed to the simultaneous provision and consumption of software and services, unlike traditional tangible products. This distinction stems from the characteristics of service, namely their intangibility, inseparability, heterogeneity, and perishability (Zeithaml *et al.*, 1985), and the characteristics of software, the ability to constantly being revised and improved while being provided to customers.

The different characteristics of physical products, software, and service have a significant influence on each development process, and they have been subject to the research of diverse scholars. Griffin (1997) argued that NSD and SDLC processes tend to be less formal than NPD processes. A conventional linear process is originated from designing physical artifacts underlining the hierarchy of design in which requirements of physical components are gradually broken down (Royce, 1970; Boehm, 1976). Thus, development risks over the NPD processes are managed and controlled through the decomposition of requirements. However, without physical constraints, software as a reusable unit of business-complete work (Papazoglou *et al.*, 2007) can easily be reassembled to deliver new features and user value. In this respect, development risks are controlled by agility with sense-and-respond capability over the software development process (Svahn *et al.*, 2009). Due to the different emphasis on the development between hardware and software, in information systems literature, Svahn *et al.* (2009) argued that there are tensions over the development process of networked artifacts.

Scholarly work on the existing development process adequately addresses valuable insights related to the IoT development process, including (a) the generic phases of NPD processes; (b) various attributes of different development approaches; and (c) commonalities and differences of tangible products, services, software, and integrated development processes. However, it has been identified that only a few existing processes "partially" reflect vital characteristics of networked products or integrate data science within the development model. As the data science process and the factors that influence IoT development are the key attributes of the IoT design process, the attention of this discussion focuses upon literature on the factors that affect the IoT development process.

Factors to be considered for IoT design

Different factors that affect IoT development

As the development process is affected by the attributes of the subject, it is required how a digitalized artifact differs from traditional products, which have a fixed, discrete set of boundaries and features. Researchers in information systems (Yoo *et al.*, 2010*b*, 2012; McAfee and Brynjolfsson, 2012; Henfridsson *et al.*, 2014; Johnson *et al.*, 2017) identified several factors that differentiate NPD processes of the 20th century from its counterpart within the digital economy (Table 3).

The material properties of digitalized artefacts can be matched to each layer of IoT architecture, as shown in Figure 2. *Sense-ability*, *Memorability*, and *Traceability* of digitalized artifacts linked to the layer of device hardware are closely related to *Homogenization* of data and *the dimensions of big data*. Embedded software leads to the Programmability of smart products interrelated to *reprogrammability*, *digital materiality*, and *the prevalence of combinatorial innovation*. *Addressability, Communicability*, and *associability* are enabled by the layer of communications and processing, which would then make IoT development more heterogeneous with the significance of the digital platform. There are no material properties matched to the application layer, as it is more about delivering value to the users. As illustrated in Figure 3, these material properties provide new ways of IoT development that can be explained with critical factors, including the dimensions of big data, the characteristics of digital technologies, the six dimensions of digital innovation, and the traits of innovations associated with pervasive digital technology.

Among the different factors, Yoo *et al.* (2010*b*) identified *data homogenization* and *reprogrammability* as the fundamental and unique design characteristics of digital technology. The homogenization of data means that a discrete representation of data in

Table 3. Factors affect the NPD process for IoT

Name	Dimension	Source
Seven material properties of digitalized artefacts	Programmability; addressability; sense-ability; communicability; memorability; traceability; and associability	(Yoo, 2010)
The dimensions of big data (3Vs)	Volume; variety; and velocity	(McAfee and Brynjolfsson, 2012; Johnson *et al.*, 2017)
The characteristics of digital technologies	Reprogrammability; homogenization of data; and self-referential nature of digital technology	(Yoo *et al.*, 2010*a*)
The six dimensions of digital innovation	Convergence; digital materiality; heterogeneity; generativity; locus of innovation; and pace	(Yoo *et al.*, 2010*b*)
Three traits of innovations associated with pervasive digital technology	The significance of digital technology platform; the emergence of distributed innovations; and the prevalence of combinatorial innovation	(Yoo *et al.*, 2012)
Impact of digital technology on new way of design and production	Design scalability and design flexibility	(Henfridsson *et al.*, 2014)

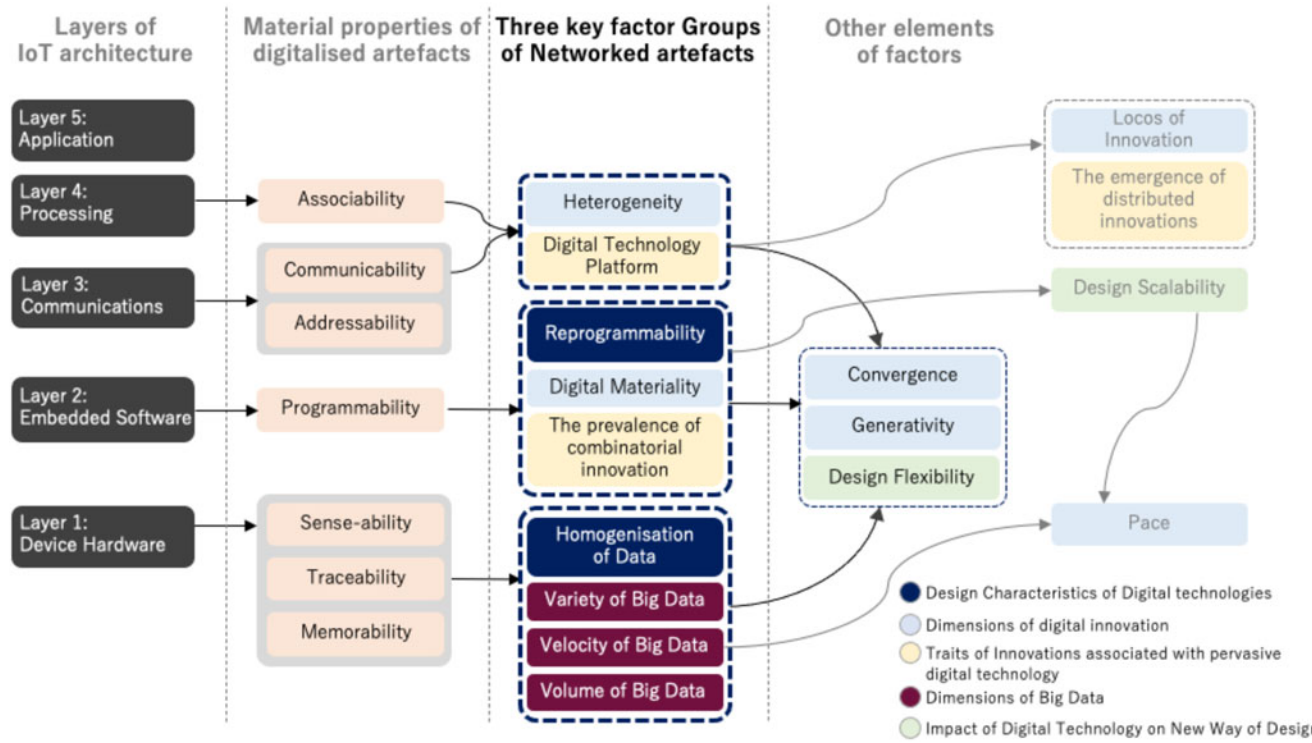

Fig. 3. Mapping the interconnectivities between the IoT architecture layers, material properties of digitalized artifacts, and three key factor groups of networked artifacts.

bits of 0 and 1 enables the homogenization of all data accessible by digital devices, whereas analogue data has a tight coupling between data and special devices (e.g., pictures and camera) (Yoo *et al.*, 2010*a*). *Data homogenization* empowers all different types of digital data collected from different sources that can be efficiently combined with other digital data (Yoo *et al.*, 2010*a*). With the homogenization of data, the dimensions of big data, volume, velocity, and variety (McAfee and Brynjolfsson, 2012; Johnson *et al.*, 2017) enables to create a diverse variety of services and dissolves product and industry boundaries.

Another critical aspect of a networked artefact is *reprogrammability*. In the digital realm, features of a product become malleable and flexible with *digital materiality* which is “what the software incorporated into an artefact can do by manipulating digital representations" (Yoo *et al.*, 2012). The affordances of pervasive digital technology lead to the *convergence* of multiple affordances with a single smart device (Yoo *et al.*, 2012). Due to *reprogrammability*, the development of digital artifacts has the aspects of *generativity* which empowers the continual reinterpretations, expansions, and refinements of products, contents, services, or processes (Yoo *et al.*, 2010*b*). The convergence of media and products increases the competition among heterogeneous markets and industries. The network and communication capabilities of IoT products are also a key attribute in IoT development. It results in IoT design not only *Heterogeneous* in which diverse forms of data, information, knowledge, and tools are integrated (Yoo *et al.*, 2010*b*), but also a platform-centered (Yoo *et al.*, 2012). Moreover, the heterogeneity of digitalized artifacts increasingly distributes the development activities, moving toward the periphery of organizations.

Along with the complex architecture and material properties of IoT, the unique properties and traits of digital technology and big data enable a new approach of design and development that are evidently distinctive from the development processes in the mid to late 20th century. The different elements of factors are independent and interdependent under the umbrella of IoT design. However, with a focus on IoT as networked artifacts, we identified three fundamental attributes of designing networked artifacts: 3Vs of big data, reprogrammability, and heterogeneity. These three key attributes will be revisited for discussing the conceptual IoT design process. In the next section, data science processes, which is another significant factor to be considered and reflected within the IoT NPD process, will be further discussed.

Data science processes and practices

Several recent studies have explored how modern data, often live, large, complicated, and/or messy, is analyzed. There are several different processes identified concerning different emphasis on the process, for example, exploratory approach or holistic approach. The study of Kandel *et al.* (2012) identified the data analysis process based on the interview of enterprise analysts in the context of the larger organization. The study focused on the comprehensive data science process containing five major phases: Discover data; Wrangle data; Profile data; Model data; and Report procedures and insights. Furthermore, based on Kandel *et al.*’s five phases of process, Alspaugh *et al.* (2018) devised a data science process that emphasizes the exploratory aspects of the data practice, which encompasses an additional phase, *EXPLORE*. The data science process proposed by O’Neil and Schutt (2013) and Baumer (2015) aims to teach data science to students. O’Neil and Schutt (2013) defined and developed the data science process which includes the phase of Exploratory Data Analysis. Baumer (2015) presents a holistic approach to

Table 4. Generic phases of data science process

Phases	Descriptions
Understanding business aim & identifying data	Understanding the business aim precedes data source investigation and collection, which enables identifying what data needs to be collected.
Collecting, cleaning, and wrangling data (preparing data)	Once the data source is evaluated and profiled, data is cleaned and wrangled into a desired format for analysis. This phase aims to integrate data from multiple sources into a single file or extract entities from documents and make accurate and consistent data.
Data profiling (diagnosing data quality & making assumptions)	Once data is cleaned, assembled, and integrated, analysts diagnose data to reveal data structure, null records, and potential data quality issues, as well as to understand what assumptions they can make about their data. This phase aims to verify its quality and its suitability for the analysis tasks, and to examine the distributions of values within fields.
Modeling (building machine learning algorithms statistical models)	After all the necessary data was assembled and comprehended, analysts begin modeling the data for summarization or prediction. This would include computing summary statistics, running regression models, or performing clustering and classification.
Identifying & Building Data Product	Last phase is to develop data products as the output of data science process. Reports could be one sort of data product that data is visualized or communicated to share insights from modeling with other business units. The other type of data product could be prediction algorithms or recommendation systems.

teaching data science processes, from framing a question to obtaining an answer through a variety of data practices. The pivot of the process is to form questions through data acquisition, processing, and exploring computational methods to discover answers in the form of inferences and presentations.

Even though each of the processes presents slightly different phases, similar to design and development processes, different data science processes share resembling phases of the process, which are further described in Table 4. Despite the appearance of their linear processes, they are described as a nonlinear and iterative process (Kandel *et al.*, 2012; Sands, 2018). In the next section, the theories, frameworks, and models are examined and analyzed, which will be used to establish the conceptual model of IoT design processes.

Conceptual model of IoT design processes

Our conceptual IoT design process consists of three layers of eight-shaped cycles (Fig. 4). Each layer represents physical product development, software development, and data practice process. Although the development activities for data represent the generic data science process identified through literature, the tasks can be done selectively depending on the smartness of IoT device, for example, monitoring, control, optimization, and autonomy (Porter and Heppelmann, 2014). We named this novel approach as *the Mobius strip model of IoT development*, as it implies an infinite loop. The IoT development process begins from the top of the left-hand cycle, consisting of six generic phases of NPD, which are identified through a literature review. IoT development involves three different types of subject matter; each phase has specific development activities depending on a physical product, software, and data (Table 5). Once the physical products and software are developed and deployed through the first cycle on the left hand, the process enters into the *Maintaining, Evaluating*, and *Planning next* phase.

At this point, the organization needs to devise a strategic decision to maintain and improve the existing value proposition or expand its value constellation, connecting new smart products to its system. If the organization decides to deliver a new service through more interconnected devices, they may revert to the left-hand cycle while analyzing data and maintaining new software. If they adopt the former strategy, they may enter the right NPD loop. Over this loop, the main NPD activities are related to software development and data process. The new users' and business needs identified in the right loop could trigger a new IoT development process and start other left-hand loops whilemaintaining the existing system.

The first phase of the first development cycle is "Discovering Users' & Business' needs" in which an organization conducts market & user research as well as strategic planning. The development activities for software and data are likely to be technically orientated that involves the second phase "defining concepts of & strategies for business/technical solutions." Once the concept and system requirements are identified, the organization tests feasibility of the business and technical solutions in which the data science process begins in earnest, identifying data source. The fourth phase "Designing, Prototyping, Integrating, & Testing Solutions' is the most condensed, incorporating many design activities and development risks. As one of the risks, Svahn *et al.* (2009) argued that different speeds and approaches of physical products and software development might occur tensions. On top of that, even greater tensions would be generated between data process, physical, and digital development if the IoT system encompasses AI elements. Because building annotated data sets and testing the algorithm are tedious, time-consuming (Kandel *et al.*, 2012; Deutsch, 2015), and costly, adding more complexity to the IoT design process. Alternatively, if a company decides to use annotated data sets, their design process would be simpler, but they might be challenged to differentiate their business from that of the competitors.

Developing physical and digital products challenges the designer with a continual pressure of never being able to finish design and integrate system due to reprogrammability (Yoo *et al.*, 2010*a*) and the pace of digital innovation (Yoo *et al.*, 2010*b*). Once the IoT system is manufactured and deployed, data starts to be collected and an organization enters the phase of "maintaining, evaluating, and planning next" in which the existing IoT system is monitored and maintained. Shifting from the left cycle to the right cycle is where the critical distinction between our conceptual design process and existing development processes is stressed. Over the right-hand side development, the organization endeavors to redefine the value proposition of IoT offerings based on real-time data on customers' experience. The

https://doi.org/10.1017/S0890060421000299 Published online by Cambridge University Press

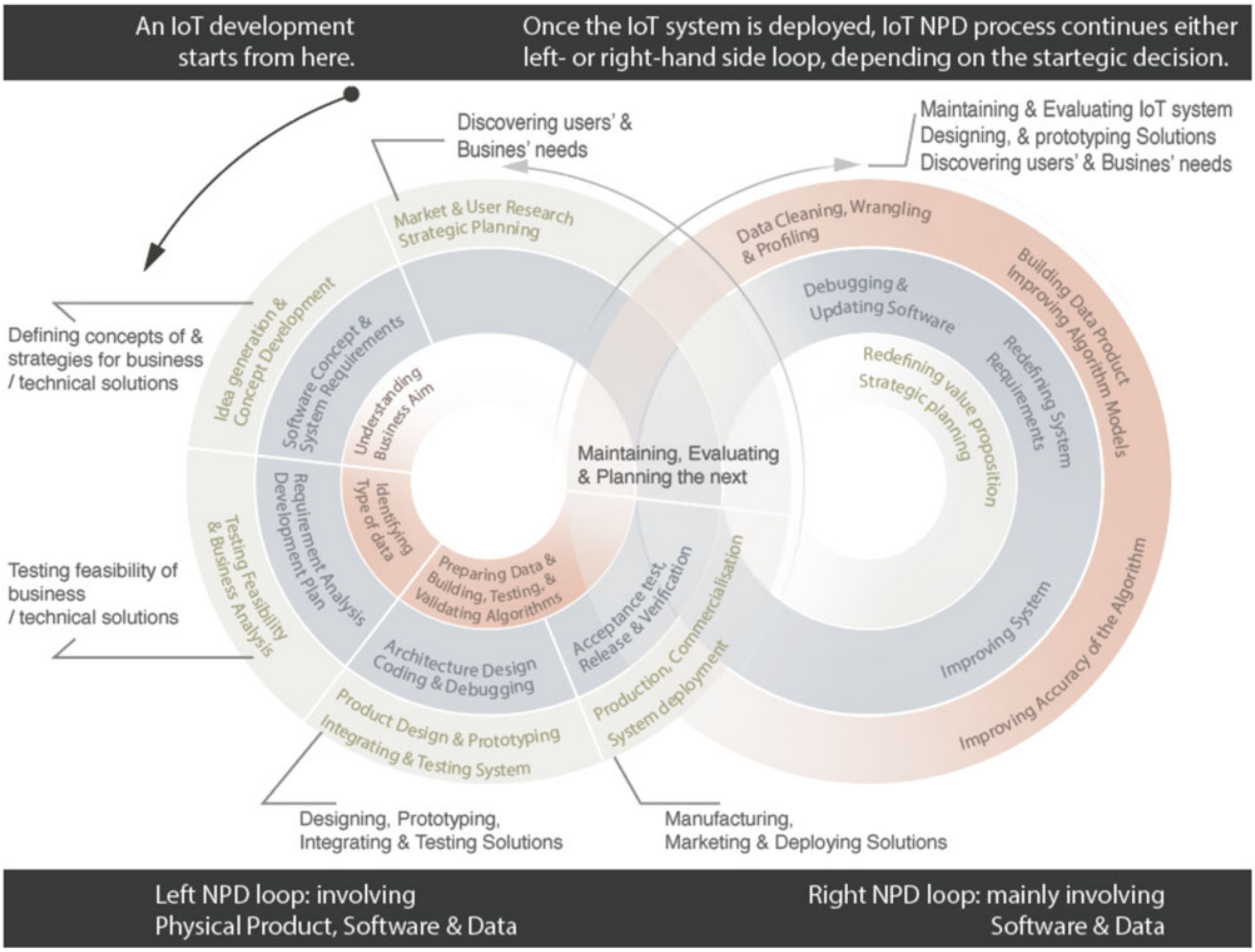

Fig. 4. Mobius strip model of IoT development processes.

Table 5. Development activities for physical product, software, and data in the first cycle IoT development

Phases	Physical Product	Software	Data
1. Discovering users' & business' needs	Market & user research; strategic planning	–	–
2. Defining concepts of & strategies for business/technical solutions	Idea generation & concept Development	Identifying software concept & system requirements	Understanding business aim
3. Testing feasibility of business/ technical solutions	Testing feasibility & business Analysis	Requirements analysis & development plan	Identifying data source & collecting data
4. Designing, prototyping, integrating, & testing solutions	Designing & prototyping product; integrating & testing system	Architecture design, coding, & debugging	Data wrangling, profiling, modeling, & building data product
5. Manufacturing, marketing, & deploying solutions	Production, commercialization, & system deployment	Acceptance test, release, & verification	Collecting data
6. Maintaining, evaluating, & planning next	–	Debugging & maintaining system	Data cleaning, wrangling, & profiling

redefined value could be provided through having new features released, connecting other IoT devices and services to the existing IoT system (Yoo *et al.*, 2012), or providing data to manage the customer relationship. While supporting customers and maintaining up-to-date software, if novel users and business needs can be met with new IoT development, it triggers the other left NPD loops. This point within the design process is where the true value of the IoT is thoroughly realized having more things connected and communicated with each other seamlessly (Lee *et al.*, 2019*b*). In this way, the IoT development process is continually re-designing IoT products and services in an iterative way.

Our conceptual IoT NPD process with two infinite loops is different from existing processes, as the two cycles need a distinctive strategy and approach to development and value creation.

https://doi.org/10.1017/S0890060421000299 Published online by Cambridge University Press

This is primarily caused by key considerations within each cycle. The left-hand side development loop heavily involves physical product development, whereas the right-hand side development loop mainly focuses on data process and software development. Consequently, the left loop is more influenced by physical product development, which requires more robust approval and control. In contrast, the development activities on the right-hand side are more flexible and faster, affected by software development and data analysis. Having stress on software and data practice results in the phase of manufacturing being omitted in the development cycle on the right-hand side. The separation between the phases of the left cycle represents stricter review gates, while over the cycle on the right-hand side occurred concurrently and more flexibly alongside IoT system consumption.

Unlike existing development processes that include FFE with the stages of idea generation, opportunity validation, and concept development, we argue that the fuzziness is no longer the characteristics of the initial stage but of the whole process. It is supported by the theories of the reprogrammability of the IoT system and big data. Once an IoT device is deployed, data is continuously collected and analyzed, which enables organizations to generate business opportunities and develop new product and service concepts. Due to a little time constraint in software development, ideas can be effectively and efficiently generated, and the speed of development is faster than the cycle on the left-hand side of the process. Each development cycle requires the organization to take a different business strategy and model. When involving physical product development, the organization's business model may be close to the manufacturer. However, once the IoT system is launched and implemented, the organization becomes more of a service provider maintaining, refining current offerings, and delivering new services.

Although the Mobius strip model of IoT development is conceptual in nature, the proposed model is based on bridging existing theories and linking works across disciplines, the constructs behind the conceptual model are drawn from the case studies of the IoT development. It is beyond the scope of this paper to discuss the case studies in detail. However, the individual IoT NPD processes can be found in the publications (Lee *et al.*, 2020, 2019*a*), which discuss how our case studies are related to conceptualizing the Mobius strip model.

Conclusions

As novel and challenging as today's IoT is, IoT offers fertile opportunities for organizations' long-term sustainable growth. Given its nascent status, there is still a lack of scholarly works on the development process of IoT products and services, which is arguably one of the most critical marketing planning and implementation process activities. In exploring this theme, the paper draws an extensive review of current theories, frameworks, and models connected to the research study area, critically interrogating extant accounts of NPD processes (in its broadest sense). Through contextualizing established literature, this paper provides attention to the core research questions at large: (1) What are the commonalities and differences between the development process for a physical product, software, service, and integrated product? (2) How do the factors and data science processes affect the IoT development process? (3) What is the conceptual process for IoT development?

Through the descriptive analysis, the paper identified: generic phases of the development process for an IoT design process; that process balance between strict reviews, controls, flexibility, and iteration; the FFE of an innovation process; a different backend between the development processes; and a distinctive development approach depending on the subject matter. Reflecting on the unique characteristics of IoT products, alongside insights from existing NPD and data science processes, the Mobius strip model of the IoT development process is proposed which has three layers of eight-shaped loops. Each cycle of the model needs different development approaches and strategies and enables the scope, feature, and value proposition of IoT systems to unceasingly evolve even after being launched and while in use.

Although this study has explored the fields that intersect with the design process for IoT, there are a couple of limitations that could be addressed in future studies. Given that there is no established body of literature for IoT design, our conceptual process is only based on related literature, and the model needs to be tested in the field to validate it. Moreover, a little reflection of the data-enabled design approach toward the Mobius strip model suggests that further work on the use of IoT data and the nuances of using data within the NPD, specifically within the design process, is necessary to ensure the validity of the model. Notwithstanding these limitations, however, the authors argue that the finding has some crucial contributions to ground our knowledge of the IoT NPD process. This research will enable industry practitioners to understand better, how the IoT system should be developed through a conceptual NPD process. For academics, this paper contributes to augmenting the body of literature regarding emergent innovation processes for IoT and serves as a starting point for future in-depth research on IoT NPD processes.

Acknowledgements. The research presented in this paper has been possible through the PETRAS IoT Hub, funded by the UK Engineering and Physical Sciences Research Council (EPSRC), and the Research and Development Management Association (RADMA) Doctoral Funding.

References

Alam I (2006) Removing the fuzziness from the fuzzy front-end of service innovations through customer interactions. *Industrial Marketing Management* **35**, 468–480.

Alshamrani A and Bahattab A (2015) A comparison between three SDLC models waterfall model, spiral model, and incremental/iterative model. *International Journal of Computer Science Issues* **12**, 106–111. Available at www.IJCSI.org

Alspaugh S, Zokaei N, Liu A, Jin C and Hearst M (2018) Futzing and moseying: interviews with professional data analysts on exploration practices. *IEEE Transaction on Visualisation and Computer Graphics* **25**, 22–31.

Andrews B (1975) *Creative Product Development: A Marketing Approach to New Product Innovation and Revitalisation*. London: Longman Group Limited.

Aurich JC, Fuchs C and Wagenknecht C (2006) Life cycle oriented design of technical product-service systems. *Journal of Cleaner Production* **14**, 15.

Baker M and Hart S (1999) *Product Strategy and Management*. Essex: Prentice Hall.

Baumer B (2015) A data science course for undergraduates: thinking with data. *American Statistician* **69**, 334–342.

Benington H (1956) Production of large computer programs. *ONR Symposium on Advanced Programming Methods for Digital Computers*, pp. 15–27.

Berkhout G, Hartmann D and Trott P (2010) Connecting technological capabilities with market needs using a cyclic innovation model. *R&D Management* **40**, 474–490. doi:10.1111/j.1467-9310.2010.00618.x.

Best K (2006) *Design Management: Managing Design Strategy, Process and Implementation*, 1st Edn. London: AVA Publishing.

Birrell N and Ould M (1985) *A Practical Handbook for Software Development*. New York: Cambridge University Press.

Blazevic V and Lievens A (2004) Learning during the new financial service innovation process – antecedents and performance effects. *Journal of Business Research* **57**, 374–391. doi:10.1016/S0148-2963(02)00272-2.

Boehm BW (1976) Software engineering. *IEEE Transactions on Computers* **25**, 1226–1241.

Boehm BW (1986) A spiral model of software development and enhancement. *ACM SigSoft Software Engineering Notes* **11**, 22–42. Available at https://ieeexplore-ieee-org.ezproxy.lancs.ac.uk/stamp/stamp.jsp?tp=&arnumber=59 (accessed September 7, 2020).

Boehm BW (1995) Spiral model of software development and enhancement. *IEEE Engineering Management Review* **23**, 69–81. doi:10.1145/12944.12948.

Booz E, Allen J and Hamilton C (1982) *NEW Product Management for the 1980s*. New York: Booz, Allen and Hamilton.

Bowen KH, Clark KB, Holloway CA and Wheelwright SC (1994) Developing project: the engine of renewal. *Harvard Business Review* 1994 September–October, 110–120. Available at https://hbr.org/1994/09/development-projects-the-engine-of-renewal.

Brown SL and Eisenhardt KM (1995) Product development: past research, present findings, and future directions. *The Academy of Management Review* **20**, 343–378.

Bruce M and Cooper R (2001) *Creative Product Design: A Practical Guide to Requirements Capture Management*. London, England: John Wiley & Sons, Ltd. Available at https://www.wiley.com/en-us/Creative+Product+Design%3A+A+Practical+Guide+to+Requirements+Capture+Management-p-9780471987208 (accessed June 28, 2020).

Chesbrough WH (2004) Open Innovation: Renewing Growth from Industrial R&D. *10th Annaul Innovation Convergence*, Minneapolis.

Cisco (2017) *The Journey to IoT Value: Challenges, Breakthroughs, and Best Practices*. Cisco.

Clark K and Fujimoto T (1991) *Product Development Performance: Strategy, Organisation, and Management in the World Auto Industry*. Boston, MA: Harvard Business School Press.

Cooper RG (1990) Stage-gate systems: a new tool for managing new products. *Business Horizons* **33**, 44–54. doi:10.1016/0007-6813(90)90040-I.

Cooper R (1994) Third-generation new product processes. *Journal of Product Innovation Management* **11**, 3–14.

Cooper R (2008) Perspective: The Stage-Gate® idea-to-launch process—update, what's new, and NexGen systems. *Journal of Product Innovation Management* **25**, 213–232.

Cooper R (2016) Agile–Stage-Gate hybrids. *Research Technology Management* **59**, 21–29.

Council Design (2007) *Eleven lessons: managing design in eleven global brands. A study of the design process*. London.

Crawford M (1997) *New Products Management*, 5th Edn. New York: McGraw-Hill, Inc.

Department of Defense (1988) *DoD-STD-2167*.

Deutsch R (2015) Leveraging data across the building lifecycle. *Procedia Engineering* **118**, 260–267. Available at https://doi.org/10.1016/j.proeng.2015.08.425 (accessed September 21, 2020).

Dewulf K (2013) Sustainable product innovation: the importance of the front-end stage in the innovation process. *Advances in Industrial Design Engineering* 139–166. Available at https://doi.org/10.5772/52461 (accessed October 23, 2020).

Durisin B, Calabretta G and Parmeggiani V (2010) The intellectual structure of product innovation research: a bibliometric study of the journal of product innovation management, 1984-2004. *Journal of Product Innovation Management* **27**, 437–451. doi:10.1111/j.1540-5885.2010.00726.x.

Edvardsson B, Johnson MD, Gustafsson A and Strandvik T (2000) The effects of satisfaction and loyalty on profits and growth: products versus services. *Total Quality Management* **11**, 917–927. doi:10.1080/09544120050135461.

Evans J (1959) Basic design concepts. *Journal of American Society of Naval Engineers* **71**, 671–678.

Eveleens C (2010) Innovation management; a literature review of innovation process models and their implications. *Science* **800**, 900–916.

Fairlie-Clarke T and Muller M (2003) An activity model of the product development process. *Journal of Engineering Design* **14**, 247–272.

Forsberg K and Mooz H (1991) The relationship of system engineering to the project cycle. *The First Annual Symposium of National Council on System Engineering*, pp. 1–12.

Froehle C, Roth AV, Chase R and Voss CA (2000) Antecedents of new service development effectiveness. *Journal of Service Research* **3**, 3–17.

Gericke K and Blessing L (2012) An analysis of design process models across disciplines. *International Design Conference*, Dubrovnik-Croatia, pp. 171–180.

Griffin A (1997) PDMA research on new product development practices: updating trends and benchmarking best practices. *Journal of Product Innovation Management* **14**, 429–458. doi:10.1016/S0737-6782(97)00061-1.

Hartsell C, Mahadevan N, Nine H, Bapty T, Dubey A and Karsai G (2020) Workflow automation for cyber physical system development processes. *Design Automation for CPS and IoT*.

Henfridsson O, Mathiassen L and Svahn F (2014) Managing technological change in the digital age: the role of architectural frames. *Journal of Information Technology* **29**, 27–43. doi:10.1057/jit.2013.30.

Howard TJ, Culley SJ and Dekoninck E (2008) Describing the creative design process by the integration of engineering design and cognitive psychology literature. *Design Studies* **29**, 160–180.

Huang G, He J, Chi C-H, Zhou W, *et al.* (2015) A data as a product model for future consumption of big stream data in clouds. *IEEE International Conference on Services Computing*, pp. 256–263. Available at https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7207361 (accessed September 1, 2020).

International Organisation for Standardization (2015) *ISO 9000: Quality Management Systems*. Available at https://asq.org/quality-press/display-item?item=T1039E (accessed September 1, 2020).

Jacobs N and Cooper R (2018) *Living in Digital Worlds: Designing the Digital Public Space*. London: Routledge.

Janne vK and Bogers S (2019) *Data-Enabled Design: A situated Design Approach That Uses Data as Creative Material When Designing for Intelligent Ecosystems*. Eindhoven, Netherland: Technische Universiteit Eindhoven.

Johne A and Storey C (1998) New service development: a review of the literature and annotated bibliography. *European Journal of Marketing* **32**, 184–251. doi:10.1108/03090569810204526.

Johnson SP, Menor LJ, Roth AV and Chase RB (2000) A critical evaluation of the new service development process: integrating service innovation and service design. In Fitzsimmmons JA and Fitzsimmons MJ (eds), *New Service Development: Creating Memorable Experiences*. Thousand Oaks, CA: SAGE Publications, Inc, pp. 1–32. doi:10.4135/9781452205564.n1.

Johnson JS, Friend SB and Lee HS (2017) Big data facilitation, utilization, and monetization: exploring the 3Vs in a new product development process. *Journal of Product Innovation Management* **34**, 640–658. doi:10.1111/jpim.12397.

Kahn KB (2005) In Kay S, Slotegraaf R and Uban S (eds), *The PDMA Handbook of New Product Development*, 2nd Edn. Hoboken, New Jersey: John Wiley & Sons, Inc.

Kandel S, Paepcke A, Hellerstein JM and Heer J (2012) Enterprise data analysis and visualization: an interview study. *IEEE Transactions on Visualization and Computer Graphics* **18**, 2917–2926.

Kent B, Grenning J, Martin R, Beedle M, Bennekum A, Cockburn A, Cunningham W, Fowler M, Highsmith J, Hunt A, Jeffries R, Kern J, Marick B, Mellor S, Schwaber K, Sutherland J and Thomas D (2001). *Manifesto for Agile Software Development*. Available at https://agilemanifesto.org/ (Accessed on 19 July 2020).

Kessler EH and Bierly PE (2002) Is faster really better? An empirical test of the implications of innovation speed. *IEEE Transactions on Engineering Management* **49**, 2–12. doi:10.1109/17.985742.

Khurana A and Rosenthal S (1997) Integrating the fuzzy front-end of new product development. *Sloan Management Review* **38**, 103–120.

Kim J and Wilemon D (2002) Focusing the fuzzy front-end in new product development. *R&D Management* **32**, 269–279.

Kleinschmidt EJ and Cooper RG (1991) The impact of product innovativeness on performance. *Journal of Product Innovation Management* **8**, 240–251.

Komoto H and Tomiyama T (2009) Design of competitive maintenance service for durable and capital goods using life cycle simulation. *International Journal of Automation Technology* **3**, 63–70. Available at https://www.fuji-press.jp/main/wp-content/themes/Fujipress/pdf_subscribed.php (accessed September 18, 2020).

Kotler P (2000) *Marketing Management: Millennium*. Upper Saddle River: Prentice Hall.

Krishnan V and Ulrich KT (2001) Product development decisions: a review of the literature. *Management Science* **47**, 1–21.

Lasi H, Fettke P, Kemper HG, Feld T, *et al.* (2014) Industry 4.0. *Business and Information Systems Engineering* **6**, 239–242. doi:10.1007/s12599-014-0334-4.

Lee B, Cooper R, Hands D and Coulton P (2019*a*) Design divers: a critical enabler to mediate value over the NPD process within Internet of Things. *4D Conference Proceedings: Meanings of Design in the Next Era*, Osaka, Japan. Available at https://eprints.lancs.ac.uk/id/eprint/139453/ (accessed September 3, 2020).

Lee B, Cooper R, Hands D and Coulton P (2019*b*) Value creation for IoT: challenges and opportunities within the design and development process. *Living in the Internet of Things (IoT 2019)*, London, England, p. 8.

Lee B, Cooper R, Hands D and Coulton P (2020) Designing for the Internet of Things: a critical interrogation of IoT processes and principles. *DRS – Design Research Society Conference*, Brisbane.

Lovelock C and Gummesson E (2004) Whither services marketing? In search of a new paradigm and fresh perspectives. *Journal of Service Research* **7**, 20–41. doi:10.1177/1094670504266131.

MacConell S (1996) *Rapid Development*. Redmond, WA, USA: Microsoft Press. Available at https://books.google.co.uk/books/about/Rapid_Development.html?id=-Up3BQAAQBAJ&source=kp_book_description&redir_esc=y (accessed September 1, 2020).

Maclnnis DJ (2011) A framework for conceptual contributions in marketing. *Journal of Marketing* **75**, 136–154.

Manyika J, Chui M, Bughin J, Dobbs R, *et al.* (2013) Disruptive technologies: advances that will transform life, business, and the global economy. *McKinsey*. Available at https://www.mckinsey.com/business-functions/mckinsey-digital/our-insights/disruptive-technologies#.

Martin RC (2002) *Agile Software Development, Principles, Patterns and Practices*, 1st Edn. New Jersey: Prentice Hall.

Matkovic P and Tumbas P (2010) A comparative overview of the evolution of software development models. *Journal of Industrial Engineering and Management* **1**, 163–172.

Maussang N, Zwolinski P and Brissaud D (2009) Product-service system design methodology: from the PSS architecture design to the products specifications. *Journal of Engineering Design* **20**, 349–366. doi:10.1080/09544820903149313.

McAfee A and Brynjolfsson E (2012) Big data: the management revolution. *Harvard Business Review* 2012 October, 60–68. Available at https://hbr.org/2012/10/big-data-the-management-revolution (accessed November 25, 2019).

Mcconnel S (1996) *Rapid Development: Taming Wild Software Schedules*. Redmond, WA, USA: Microsoft Press.

Mcconnel S (1996) *Rapid Development: Taming Wild Software Schedules*. Washington: Microsoft Press.

Menor LJ, Tatikonda MV and Sampson SE (2002) New service development: areas for exploitation and exploration. *Journal of Operations Management* **20**, 135–157. doi:10.1016/S0272-6963(01)00091-2.

Meyer B (2001) *Software: Product or Service?*

Meyer MH and Utterback JM (1995) Product development cycle time and commercial success. *IEEE Transactions on Engineering Management* **42**, 297–304.

Microsoft Team (2003) *Microsoft Solution Framework White Papers*.

Mont O (2000) *Product-Service Systems Final Report*.

Ng ICL and Wakenshaw SYL (2017) The internet-of-things: review and research directions. *International Journal of Research in Marketing* **34**, 3–21. doi:10.1016/j.ijresmar.2016.11.003.

Nijssen EJ and Lieshout KFM (1995) Awareness, use and effectiveness of models and methods for new product development. *European Journal of Marketing* **29**, 27–44. doi:10.1108/03090569510098483.

Normann R and Ramírez R (1994) *Designing Interactive Strategy: From Value Chain to Value Constellation*. Chichester, New York: John Wiley & Sons, Inc.

O'Neil C and Schutt R (2013) In Loukides M and Nash C (eds), *Doing Data Science*, 4th Edn. Sebastopol, CA: O'Reilly Media Inc. Available at https://semanticommunity.info/Data_Science/Doing_Data_Science (accessed August 27, 2020).

Page AL and Schirr GR (2008) Growth and development of a body of knowledge: 16 years of new product development research, 1989–2004. *Journal of Product Innovation Management* **25**, 233–248. Available at https://doi.org/10.1111/j.1540-5885.2008.00297.x (accessed September 16, 2020).

Papastathopoulou P and Hultink E (2012) New service development: an analysis of 27 years of research. *Journal of Product Innovation Management* **29**, 705–714.

Papazoglou MP, Van Den Heuvel W-J, Papazoglou MP and Van Den Heuvel W-J (2007) Service oriented architectures: approaches, technologies and research issues. *The International Journal on Very Large Data Bases* **16**, 389–415. doi:10.1007/s00778-007-0044-3.

Pennell J, Winner R, Bertrand H and Slusarszuk M (1989) Concurrent engineering: an overview for autotestcon. *The System Readiness Technology Conference*, Philadelphia, PA, pp. 88–99. Available at https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=64049 (accessed September 12, 2020).

Poppendieck M and Poppendieck T (2003) *Lean Software Development: An Agile Toolkit*. New Jersey: Addison-Wesley Professional.

Porter EM and Heppelmann EJ (2014) How smart, connected products are transforming competition. *Harvard Business Review*. Available at https://hbr.org/2014/11/how-smart-connected-products-are-transforming-competition (accessed February 11, 2020).

Psomakelis E, Nikolakopoulos A, Marinakis A, Psychas A, Moulos V, Varvarigou T and Christou A (2020) A scalable and semantic data as a service marketplace for enhancing cloud-based applications. *Future Internet* **12**, 77. doi:10.3390/FI12050077 (accessed September 1, 2020).

Radziwon A, Bilberg A, Bogers M and Madsen ES (2014) The smart factory: exploring adaptive and flexible manufacturing solutions. *Procedia Engineering* 1184–1190. doi:10.1016/j.proeng.2014.03.108.

Ramaswamy R (1996) *Design and Management of Service Processes*, 1st Edn. New York: Addison-Wesley Pub. Co.

Reichert C (2017) *Cisco: most IoT projects are failing due to lack of experience and security. ZDNet*. Available at https://www.zdnet.com/article/cisco-most-iot-projects-are-failing-due-to-lack-of-experience-and-security/ (accessed February 11, 2020).

Reinertsen DG (1999) Taking the fuzziness out of the fuzzy front-end. *Research Technology Management* **42**, 25–31.

Rothwell R (1993) Successful industrial innovation: critical factors for the 1990s. *R&D Management* **22**, 1. Available at https://www.jstor.org/stable/pdf/24128263.pdf (accessed September 11, 2020).

Rothwell R (1994) Towards the fifth-generation innovation process. *International Marketing Review* **11**, 7–31.

Royce WW (1970) Managing the development of large software systems. *IEEE WESCON*, pp. 1–9.

Ruparelia NB (2010) Software development lifecycle models. *ACM SIGSOFT Software Engineering Notes* **35**, 8. doi:10.1145/1764810.1764814 (accessed August 28, 2020).

Sands EG (2018) How to build great data products. *Harvard Business Review*. Available at https://hbr.org/2018/10/how-to-build-great-data-products.

Saren MA (1984) A classification and review of models of the intra-firm innovation process. *R&D Management* **14**, 11–24.

Schwaber K and Beedle M (2002) *Agile Software Development with Scrum*. New Jersey: Prentice Hall.

Sidi F, Panahy PHS, Affendey LS, Jabar MA, Ibrahim H and Mustapha A (2012) Data quality: a survey of data quality dimensions. *International Conference on Information Retrieval & Knowledge Management*, pp. 300–304. Available at https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6204995 (accessed September 1, 2020).

Smith PG and Reinertsen DG (1992) Shortening the product development cycle. *Research-Technology Management* **35**, 44–49.

Speed C and Maxwell D (2015) Designing through value constellations. *ACM Interactions Magazine* **22**, 38–43. doi:10.1145/2807293.

Stevens G and Burley J (2003) Piloting the rocket of radical innovation. *Research Technology Management* **46**, 16–25.

Svahn F, Henfridsson O, Yoo Y and Threesome Dance A (2009) A threesome dance of agency: mangling the sociomateriality of technological regimes in digital innovation. *International Conference on Information Systems*, pp. 1–18.

Takeuchi H and Nonaka I (1986) The new product development game. *Harvard Business Review*. Available at https://hbr.org/1986/01/the-new-new-product-development-game (accessed June 19, 2020).

Takeuchi H and Nonaka I (1986) *The new newproduct developmentgame. Harvard Business Review* January- February, 137–146.

Trott P (2012) *Innovation Management and New Product Development*, 5th Edn. London, England: Pearson Education.

Ulrich KT and Eppinger SD (2011) *Product Design and Development*. Boston, MA: McGraw-Hill.

Unger, D.W. & Eppinger, S.D. (2009) Comparing product development processes and managing risk. *International Journal of Product Development* **8** (4), 382–402. doi:10.1504/IJPD.2009.025253.

US Department of Defense (1983) *Military Standard: Defense System Software Development*. Washington.

Vargo SL and Lusch RF (2004*a*) Evolving to a new dominant logic for marketing. *Journal of Marketing* **68**, 1–17.

Vargo SL and Lusch RF (2004*b*) The four service marketing myths remnants of a goods-based, manufacturing model. *Journal of Service Research* **6**, 324–335. doi:10.1177/1094670503262946.

Vasantha G, Roy R, Lelah A and Brissaud D (2012) A review of product-service systems design methodologies. *Journal of Engineering Design* **23**, 635–659. Available at https://doi.org/10.1080/09544828.2011.639712 (accessed August 26, 2020).

Walsh V, Roy R, Bruce M and Potter S (1992) *Winning by Design: Technology Product Design and International Competitiveness*. Oxford, UK: Blackwell Publishing.

Welp EG, Meier H, Sadek T and Sadek K (2008) Modelling approach for the integrated development of industrial product–service systems. *The 41st CIRP Conference on Manufacturing Systems*.

Wohlin C (2014) Guidelines for snowballing in systematic literature studies and a replication in software engineering. *International Conference on Evaluation and ASSESSMENT in Software Engineering*.

Xu X (2012) From cloud computing to cloud manufacturing. *Robotics and Computer-Integrated Manufacturing* **28**, 75–86. doi:10.1016/j.rcim.2011.07.002.

Yerpude S and Rautela S (2020) Digitally driven new product development: an involved contemporary innovation case sonica rautela. *International Journal of Productivity and Performance Management*. Available at https://doi.org/10.1108/IJPPM-09-2019-0448 (accessed July 30, 2021).

Yoo Y (2010) Computing in everyday life: a call for research on experiential computing. *MIS Quarterly* **34**, 213–231.

Yoo Y (2013) The tables have turned: how can the information systems field contribute to technology and innovation management research? *Journal of the Association for Information Systems* **14**, 227–236.

Yoo Y, Henfridsson O and Lyytinen K (2010*a*) The new organizing logic of digital innovation: an agenda for information systems research. *Information Systems Research* **21**, 724–735. doi:10.1287/isre.1100.0322.

Yoo Y, Lyytinen KJ, Boland RJ and Berente N (2010*b*) The next wave of digital innovation: opportunities and challenges: a report on the research workshop "digital challenges in innovation research". *SSRN Electronic Journal*. doi:10.2139/ssrn.1622170.

Yoo Y, Boland RJ, Lyytinen K and Majchrzak A (2012) Organizing for innovation in the digitized world. *Organization Science* **23**, 1398–1408. doi:10.1287/orsc.1120.0771.

Yu S and Yang D (2016) The role of big data analysis in new product development. *International Conference on Network and Information Systems for Computers*, pp. 279–284.

Zeithaml VA, Parasuraman A and Berry LL (1985) Problems and strategies in services marketing. *Journal of Marketing* **49**, 33–46.

Boyeun Lee is a fifth-year PETRAS PhD candidate as well as a Post-Doctoral Research Associate in Prosperity at Imagination, Lancaster University. With a wide range of public and commercial experience across sectors as a service design researcher, her core research interests focus on IoT design process, data-enabled design, and design strategies in developing emerging technologies.

Professor Rachel Cooper is a Distinguished Professor of Design Management and Policy at Lancaster University. Her research interests cover design thinking; design management; design policy; and across all sectors of industry, a specific interest in design for well-being and socially responsible design.

Dr. David Hands is an Associate Director of Teaching, Imagination, Lancaster University. David has also written and published extensively on design management and design as a strategic resource for both profit and non-profit organizations alike.

Professor Paul Coulton is the Chair of Speculative and Game Design in the open and exploratory design-led research studio Imagination Lancaster. He uses research through a design approach to create fictional representations of future worlds in which emerging technologies have become mundane.