2016 IEEE 5th Global Conference on Consumer Electronics

Innovation Design and Its Verification & Validation in the Era of the Internet of Things

Naoshi Uchihira
School of Knowledge Science
Japan Advanced Institute of Science and Technology
Nomi, Ishikawa, Japan
uchihira@jaist.ac.jp

***Abstract*— In the era of cloud computing and the Internet of Things, fast product and service evolution has become a strong competitive advantage. Thus, the business model, business process, and system & software of the target product and service should be designed in an integrative and interactive way to satisfy key business goals (i.e., the creation of customer value and sustainable profit). We refer to this new integrative and interactive design approach that includes the business model, business process, and system & software design as "innovation design." Since business goals may change dynamically due to the speed of product/service evolution, how the innovation design should be verified and validated remains a key question. This paper proposes a new framework for the verification and validation of innovation design.**

***Keywords*— innovation design; IoT; V&V; creative adjustment; Dynamic Y-model; gap management; awareness**

I. Introduction

In the era of the Internet of Things (IoT), consumer electronics (CE) will gradually change from product-based systems to product-service systems (PSS) [1]. CE products are smarter and provide more value when they are connected to the Internet and cloud-based services. For example, a smart air conditioner can be controlled by smart phones from remote places, and a smart refrigerator can provide remote monitoring and management of freshness dates for foods in the refrigerator. The smart refrigerator can also connect with online shopping services.

For smart CE products, the method used to design the system and software should be changed. The design of a smart CE product must consider the speed of evolution and business ecosystem, which are the core factors that create competitive advantage in this era. Since a smart CE product is an open system that is openly connected with its dynamically changing environment through the Internet, the traditional verification and validation (V&V) methods for closed systems, which merely check for discrepancies between the specifications and actual implementation, are not sufficient. In a smart CE product, V&V must determine whether the smart CE product satisfies its business goal in the dynamically changing environment. In this context, the business goal means that the smart CE product is dependable, creates customer/business value, and achieves a sustainable profit. In today's competitive business environment, an innovative challenge is indispensable to achieve the business goal. We refer to the design for satisfying the business goal using innovation as "innovation design." The term "creative adjustment" is used to adjust the smart CE product to satisfy the business goal instead of the traditional V&V.

This paper defines and explains innovation design for smart CE product and then discusses the creative adjustment of the innovation design. Finally, we discuss a formal approach for innovation design and creative adjustment.

II. Why Is Innovation Design Required?

In this chapter, we briefly review the history of system design methods, including systems engineering, software engineering, requirements engineering, and service design. We then clarify why innovation design is required in the era of IoT.

A. Systems and Software Engineering

The advent of the computer has made many systems in business and society larger and more complicated. Analysis and design methods are required to construct these systems so that they satisfy their given goals in an optimal way. Therefore, "systems engineering," including analysis and design methods, has been actively investigated. System analysis provides theories, procedures, and tools for system modeling, simulation, and evaluation. System design methods support the analysis of requirements, generation of concepts, architectural & detail design, and planning of manufacturing, operation, and maintenance. Many design methods and techniques for mathematical optimization have been proposed during the long history of systems engineering.

Software is a key factor that can make systems large and complicated. Along with the growth of software, a shortage of software engineers became a serious issue (called the software crisis) in the 1960s. A new engineering field called software engineering was proposed in the early 1960s to solve the software productivity and quality problems. Since then, this field has continuously expanded. In software engineering, various techniques, methods, and tools for requirements analysis, functional design, programming, testing, and maintenance have been developed based on the waterfall model (sequential design process).

In recent years, the software (programming) productivity has dramatically improved due to the development and diffusion of software engineering, especially component-based

978-1-5090-2333-2/16/$31.00 ©2016 IEEE

software engineering. Several issues in software engineering remain, including how to acquire the customer's essential requirements (requirements engineering), how to manage a large and complex project (project management), and how to assure the dependability of open systems including smart CE products in dynamically changing environments. Requirements engineering becomes particularly important when discussing the business goals of smart CE products.

B. Requirements Engineering and Business Model Design

Requirements engineering provides processes, methods, and tools for defining, documenting, and maintaining customer requirements. Recently, the scope of requirements engineering has been expanded from requirements for an IT system to requirements for a business process using the IT system, where the goal-oriented approach is effective to extract IT system requirements from business goals [2]. In the case of PSS including smart CE products, the business process design includes service design [3]. However, traditional requirements engineering does not incorporate the design of competitive and collaborative strategies in business ecosystems.

On the other hand, many business modeling methods have been proposed. One of the most popular modeling methods is the Business Model Canvas (BMC) [4]. BMC is simple but involves all the essential elements of a business model, including a competitive and collaborative strategy. A smart CE product is often implemented in a business ecosystem. Uchihira et al. classified the opportunities and difficulties related to smart CE products [5] and proposed the open & closed strategy canvas, which complements BMC from the viewpoint of ecosystem design [6]. However, business modeling methods and system & software design methods are basically independent and used in different phases by different teams.

C. Innovation Design

Recently, various software, platforms, and infrastructures have been used on the cloud computing environment; these are respectively referred to as Software as a Service, Platform as a Service, and Infrastructure as a Service. These cloud services make it possible to start new businesses in the short term with low investment. Here the business model, business process, and system & software design are consistently handled by the same team. On the other hand, business goals change dynamically in the global, competitive, and collaborative business environment. In this context, the business model, business process, and system & software design should be integrated in a more lean and agile way as an alternative to the waterfall design process. DevOps (the collaborative process of software development and operation) and the cloud-native approach are typical design strategies that accomplish this integration. Lean and agile innovation requires this new type of integrative and interactive design approach that covers the business model, business process, and system & software, and which we refer to as innovation design. Although the importance and effectiveness of lean and agile innovation have been mentioned in prior studies [7, 8], an integrative and interactive design method that incorporates the business model, business process, and system & software along with the associated V&V has not been reported.

Innovation design consists of three interactive phases: idea generation, system design, and business model design (Fig. 1). Innovation design attaches considerable importance to the speed of evolution, which can create competitive advantage in the IoT era. Table I details the three phases and shows related design methods. Uchihira et al. proposed a PSS design method that is similar to innovation design and consists of service concept generation, service scheme design, transformation design, risk analysis, and evaluation & refinement [9].

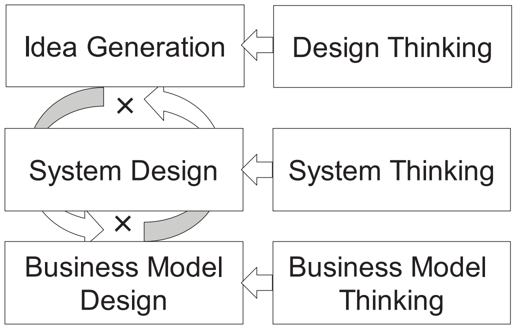

Fig. 1. Innovation Design Phases

TABLE I. INNOVATION DESIGN FUNCTIONS AND METHODS

Phase	Function	Related Methods
Idea Generation	Create ideas and concepts of innovative products and services based on customer observation.	Design Thinking, Requirements Engineering, Data & Text Mining
System Design	Make specification of the target products and services and implement them, then verify and validate them.	System Thinking, Systems Engineering, Software Engineering
Business Model Design	Construct a business model for the target products and services to make sustainable profit.	Business Model Thinking, Business Modeling Methods

III. VERIFICATION & VALIDATION OF INNOVATION DESIGN

In systems & software engineering, the V&V of the designed system are critical. Since the output of innovation design should change dynamically in the global, competitive, and collaborative business environment, traditional V&V for systems and software are not appropriate in innovation design. In this chapter, we discuss how the output of innovation design should be verified and validated.

A. Traditional V-Model and V&V

In the waterfall system & software development, the V-model is commonly used for V&V. Here, verification means the evaluation of whether or not the system & software comply with the given requirements and specifications. Validation means assuring that system & software meet the needs of the stakeholders, including customers. Recently, V-model covers an upper phase (business modeling phase; Fig.2). In this V-model, V&V can be classified into three levels:

- Business Model Validation

  Assure that the business model is working well and achieving the business goals in the global, competitive, and collaborative business environment.

- Business Process (Service) Validation

  Assure that the business process realized by the implemented system & software meets the needs of the stakeholders in the business process. To be more specific, a smart CE product satisfies customer needs.

- System & Software Verification

  Verify that the implemented system & software comply with given requirements and specifications.

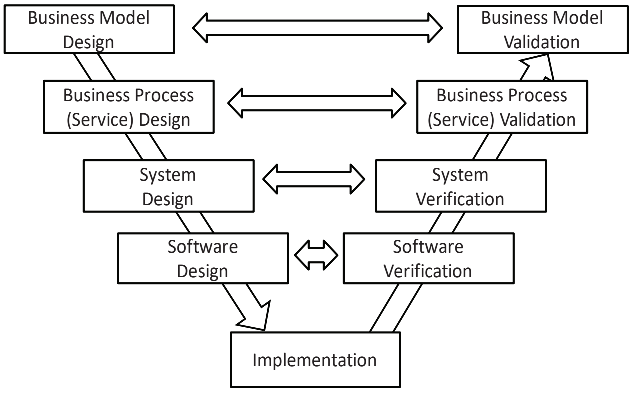

Fig. 2. V-model including Business Modeling

B. *Dynamic Y-model and Creative Adjustment*

Since the traditional V-model is intended for waterfall development, it is not appropriate for V&V in the proposed innovation design, which is integrative and interactive. In the case of a smart CE product, its functions and services can be updated online during use. Therefore, we propose a new V&V framework called creative adjustment for innovation design that is based on the dynamic Y-model [10]. The dynamic Y-model focuses on gaps among environment, specifications, and implementation, which are causes of undesirable situations (Fig. 3). The "Y" in Y-model stands for the shape of the gaps in Fig. 3. The business model, business process, and system & software are adjusted to satisfy the business goals via gap management, which consists of monitoring gaps, detecting and analyzing the gaps, and eliminating and minimizing the gaps as follows:

- Monitoring Gaps

  Monitor products, services, and businesses and gather information required for detecting and analyzing gaps among specifications, implementation, and environment.

- Detecting and Analyzing Gaps

  Detect gaps among specifications, implementation, and environment according to the monitoring information, and then analyze the causes of the gaps.

- Eliminating and Minimizing Gaps

  Eliminate the causes of the gaps to fill the gaps among specifications, implementation, and environment; alternatively, suppress the effects of the gaps.

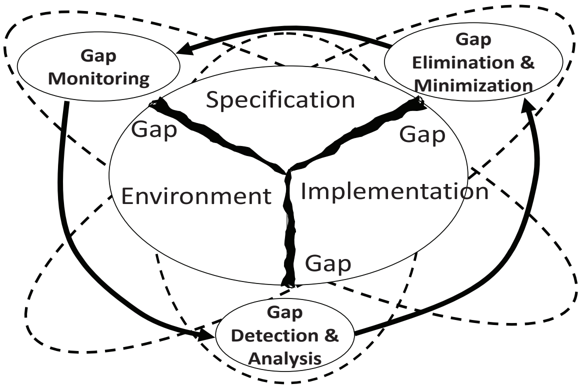

Fig. 3. Dynamic Y-model

The above gap management looks only adjustment of system & software to eliminate the causes of the gaps and reduce risks due to the gaps. However, gaps can become sources of innovation (new business opportunities). Therefore, we call it creative adjustment, in which gap management is used not only to reduce risk, but also to identify opportunities for innovation.

IV. 4. Technologies for Gap Management in Creative Adjustment

This chapter shows technologies including various formal methods for gap management and creative adjustment in the dynamic Y-model from three viewpoints (hardware, software, and human/business process; Table II).

TABLE II. TECHNOLOGIES OF GAP MANAGEMENT

Category	Hardware	Software	Human/Business
Gap Monitoring	- Ubiquitous Sensor Network - Edge Computing - Information Filtering	- Software Metrics - Assertion (Design by Contract)	- Business Intelligence - Situation Awareness
Gap Detection & Analysis	- Abnormal Detection - Fault Detection - Statistical Quality Control - Digital Twin	- Formal Verification (Model Checking, etc.) - Model-based Testing - Runtime Verification - Assurance Case	- Scenario Planning - Risk Analysis Methods (FTA, FMEA, STAMP/STPA, etc) - Failure Knowledge Database
Gap Elimination & Minimization	- Fault Tolerant System - Safety Circuit - Supervisor Control - Adaptive Control	- Resource Protection in OS - Adaptive Software - Program Adjustment	- Know-How & Know-Who Database - Lean and Agile Innovation - Minimum Viable Ecosystem

A. *Gap Monitoring Technologies*

For monitoring products, services, and businesses along with gathering the information required to detect and analyze gaps, ubiquitous sensor and powerful information storage and retrieval technologies are required in the IoT environment. Here, information filtering and edge computing are keys for handling huge amounts of sensor data. When monitoring software, checkpoints can be specified by assertions (pre/postcondition, and invariant). "Design by contract" is one of key technologies for monitoring software by assertions.

B. Gap Detection and Analysis Technologies

From the viewpoints of hardware and software, there are two types of gap detection: abnormal detection and violation detection. Abnormal detection can use machine-learning/data-mining techniques based on monitoring data during normal operation. Violation detection can use the runtime verification of software [11]. When the environment is changing dynamically, it is impossible to verify the specification and implementation beforehand; therefore, runtime verification is necessary. An approach of the open system dependability including D-Case based on the assurance case is also promising for runtime gap detection and analysis [12].

In gap detection and analysis, humans (managers, design and maintenance engineers) play an important role in addition to automated approaches such as runtime verification. STAMP (Systems Theoretic Accident Modeling and Processes) is a promising approach for risk analysis in dynamically changing environments [13]. We focus on human ability of gap detection especially for business model design.

C. Gap Eliminating and Minimizing Technologies

When detecting gaps, humans essentially have to eliminate and minimize the gaps. Although automatic restoration has been investigated for a long time as a fault-tolerant system, automatic gap-elimination and gap-minimizing technologies are required from the viewpoint of gap management in the dynamic Y-model. Hypersequential programming is one of the gap-minimizing technologies in runtime based on the formal approach [14].

D. Awareness Ability for Gap Detection in Business Model Design

In the global, competitive, and collaborative business environment, many unexpected events can occur at various levels (business model, business process, system & software). Thus, an integrative and interactive innovation design is required. Here, awareness ability becomes one of the most important factors in the early detection of gaps and identification of opportunities and risks associated with those gaps. On the other hand, automatic detection based on machine learning and formal methods has limitations related to unexpected events. Recently, development of awareness (mindfulness) ability in organizations is gathering attention as a tool of innovation management [15]. A case method is one of the effective ways of awareness ability development. Scenario planning can also develop awareness ability (awareness probe for future changes). We have proposed a knowledge-transfer method for R&D project management that can enhance awareness ability in project managers with the goal of identifying opportunities and risks based on past project cases [16].

Human awareness ability can be augmented by computer. Big-data visualization and mining are useful for humans to identify gaps. Formal approach can contribute the augmentation of human awareness ability. We have developed an awareness-support system using a smart voice-messaging system based on information supervisor control model [17].

V. CONCLUSION

This paper introduced innovation design as a new concept. Innovation design is required to design PSS including smart CE products in the era of IoT. We then discussed V&V for innovation design and proposed a new V&V framework called creative adjustment based on the dynamic Y-model instead of the traditional V-model. In creative adjustment, we believe that human awareness ability is a key factor for gap management. Formal methods can support gap management including augmentation of the human awareness ability. The integrative and interactive innovation design approach is expected to be further developed in the near future, and we believe that formal approach will play an important role in new design streams.

REFERENCES

[1] O. K. Mont, “Clarifying the concept of product–service system.” Journal of cleaner production, Vol.10, No.3, pp.237-245, 2002.

[2] A. Van Lamsweerde, Requirements engineering: from system goals to UML models to software specifications. Wiley Publishing, 2009.

[3] R. Ramaswamy, Design and management of service processes: keeping customers for life. Addison-Wesley, 1996.

[4] A. Osterwalder and P. Yves, Business model generation: a handbook for visionaries, game changers, and challengers. John Wiley & Sons, 2010.

[5] N. Uchihira, et al., "Service innovation structure analysis for recognizing opportunities and difficulties of M2M businesses." Technology in Society, Vol.43, pp.173-182, 2015.

[6] N. Uchihira, H. Ishimatsu, K. Inoue. "IoT Service Business Ecosystem Design in a Global, Competitive, and Collaborative Environment." 2016 Portland International Conference on Management of Engineering and Technology (PICMET), 2016.

[7] E. Ries, The lean startup: How today's entrepreneurs use continuous innovation to create radically successful businesses. Crown Books, 2011.

[8] L. Morris, M. Ma, and P. C. Wu, Agile Innovation: The Revolutionary Approach to Accelerate Success, Inspire Engagement, and Ignite Creativity. John Wiley & Sons, 2014.

[9] N. Uchihira, Y. Kyoya, S. K. Kim, K. Maeda, M. Ozawa, K. Ishii, “Analysis and Design Methodology for Recognizing Opportunities and Difficulties for Product-based Services.” Journal of Information Processing, Vol.16, pp.13-26, 2008.

[10] N. Uchihira, “Advanced Technologies for Dependable Systems through Product Life Cycle by Managing Gaps among Specification, Implementation, and Environment.” Toshiba Review (in Japanese), Vol.64, No.8, pp.2-7, 2009.

[11] J. Rushby, “Runtime Certification.” International Workshop on Runtime Verification (RV2008), Lecture Notes in Computer Science, 5289, Springer-Verlag, p.21-35, 2008.

[12] M. Tokoro (ed.), Open systems dependability: dependability engineering for ever-changing systems. CRC press, 2015.

[13] N. Leveson. Engineering a safer world: Systems thinking applied to safety. MIT Press, 2011.

[14] N. Uchihira, "Making Reactive Systems Highly Reliable by Hypersequential Programming.” IEICE Trans. Fundam. Electron. Commun. Comput. Sci., Vol.E88-A, No.4, p. 941-947, 2005.

[15] H. Breuer, and A. Gebauer, "Mindfulness for innovation. Future scenarios and high reliability organizing preparing for unforseeable." SKM Conference for Competence-based Strategic Management, 2011.

[16] N. Uchihira, “Knowledge Transfer Support Tool for R&D Project Management - Linking Awareness-on-Project and Awareness-in-Project.” 9th International Conference on Project Management (ProMAC2015), pp.37-42, 2015.

[17] N. Uchihira, et al, "Temporal–Spatial Collaboration Support for Nursing and Caregiving Services." In Global Perspectives on Service Science: Japan (J. C. Spohrer, S. K. Kwan, Y. Sawatani, eds.), Springer, pp.193-206, 2016.