International Journal of Service and Knowledge Management
International Institute of Applied Informatics
2022, Vol. 6, No. 2, IJSKM668

Continuous Modeling Supports from Business Analysis to Systems Engineering in IoT Development

Toshinori Takai [*] [†], Katsutoshi Shintani [‡],
Hideki Andoh [§], Hironori Washizaki [‡]

Abstract

Developing Internet of Things (IoT) systems is non-trivial due to the intrinsic nature of IoT. Diverse solution spaces, which are composed of various sensor-based and human operation-based contexts, must be satisfied simultaneously. This study proposes a chain approach method called the continuous modeling support process for business analysis and solution requirements in IoT development (COMP4BA-IoT). COMP4BA-IoT 1) captures contexts by natural language-based needs and requirements, 2) structures the identified contexts by a goal-oriented approach, and then 3) merges them to the system models. Using COMP4BA-IoT, evidence-supported communication media among stakeholders can be obtained, which can give traceability from organizational goals to IoT solutions. COMP4BA-IoT is the first trial to combine Business Analysis Body of Knowledge (BABOK), GQM$^+$Strategies, Goal Structuring Notation (GSN), and Systems Modeling Language (SysML) in the context of IoT development.

Keywords: BABOK, GQM+Strategies, GSN, SysML, Systems Engineering, MBSE, IoT

1 Introduction

The diversity in devices that comprise the *Internet of Things (IoT)* [1] makes developing IoT systems a challenge. Each device collects and shares information [2]. Motta et al. investigated issues in IoT development by reviewing the literature, surveying practitioners, and evaluating existing government reports [2]. They concluded that IoT systems have seven facets: (1) connectivity, (2) things, (3) behavior, (4) smartness, (5) problem domain, (6) interactivity, and (7) environment. Although things and connectivity are obvious characteristics of IoT systems, things are often mentioned at a very concrete level. Things often appear in the concept or business analysis phase of the system lifecycle. Behavior means that IoT can enhance the behavior of a *thing*. Smartness is also referred to as *intelligence*. The problem domain involves stakeholder needs or business requirements. Interactivity

[*] Change Vision, Inc., Tokyo, Japan
[†] Nara Institute of Science and Technology, Nara, Japan
[‡] Waseda University, Tokyo, Japan
[§] IBM Japan, Tokyo, Japan

and environment emphasize the human or machine actors. Seamlessly integrating all these aspects remains an engineering challenge.

Our research strives to realize a method that incorporates the above facets and supports decision making in the business requirement analysis or the stakeholder requirement definition process in IoT system development.

Although various techniques have been proposed in different domains, we focus on *Business Analysis Body of Knowledge* (*BABOK*) [8], *GQM*$^{+}$*Strategies* [7], and *Systems Modeling Language* (*SysML*) [4, 5]. BABOK is a collection of business analysis knowledge. GQM^{+}Strategies is based on goals, questions, and metrics designed to overcome omnipresent problems, while SysML is a modeling language that supports the implementation of *systems engineering* [3]. In this context, systems engineering is a methodology to develop a system involving multiple engineering fields and products or service domains. Systems engineering is often characterized by *front loading* and multidisciplinary. Front loading is also referred to as *executable requirements* that support early stage trade-off analysis. The multidisciplinary approach consistently deals with multiple engineering fields indispensable for IoT applications such as software, electronics, robotics, networks, etc.

We previously proposed a process for developing IoT systems [9]. Our process integrated BABOK, GQM^{+}Strategies, and SysML with a translation between an artifact obtained by GQM^{+} Strategies and a model in SysML. The translation provided effective communication between business analysts, who oversee GQM^{+}Strategies, and system engineers, who perform systems engineering with SysML. The process was used as a foundation for a class in an educational program for professional engineers. Although the class was well received, two issues were noted. First, the representation of an artifact obtained by GQM^{+}Strategies in an SysML model is complicated and redundant. Second, GQM^{+}Strategies must be translated into SysML and vice versa. To overcome these issues, we adopted a relatively simple notation specialized for goal structures called *Goal Structuring Notation* (*GSN*) [6] and refined the process [10] (Fig. 1). Herein we refine our previously proposed process and develop the COntinuous Modeling support Process for Business Analysis and solution requirements in IoT development (COMP4BA-IoT) and evaluate it via a case study.

The rest of this paper is organized as follows. Section 2 introduces background techniques. Section 3 explains the refined method. Section 4 describes our case study. Section 5 mentions related works. Finally, Section 6 summarizes this study and considers future directions.

2 Background

GQM^{+}Strategies [7] assesses the achievement of *goals*, clarifies appropriate *strategies*, and identifies the associated *metrics*. This method requires explicit management and recording of information for the *contexts* and *assumptions* of the defined goals and strategies. Such artifacts are often summarized as a diagram representation called a *GQM*$^{+}$*Strategies grid*. Fig. 2 shows an example of a GQM^{+}Strategies grid.

GSN [6] is typically used in safety assurance activities. It has a tree structure with different types of nodes: *goal*, *strategy*, *solution*, *context*, *assumption*, and *justification*. A goal, strategy, and solution represent a claim, part of an argument, and reference to evidence, respectively. Fig. 3 denotes a goal, strategy and solution as `G1`, `S1`, and `Sn1`, respectively.

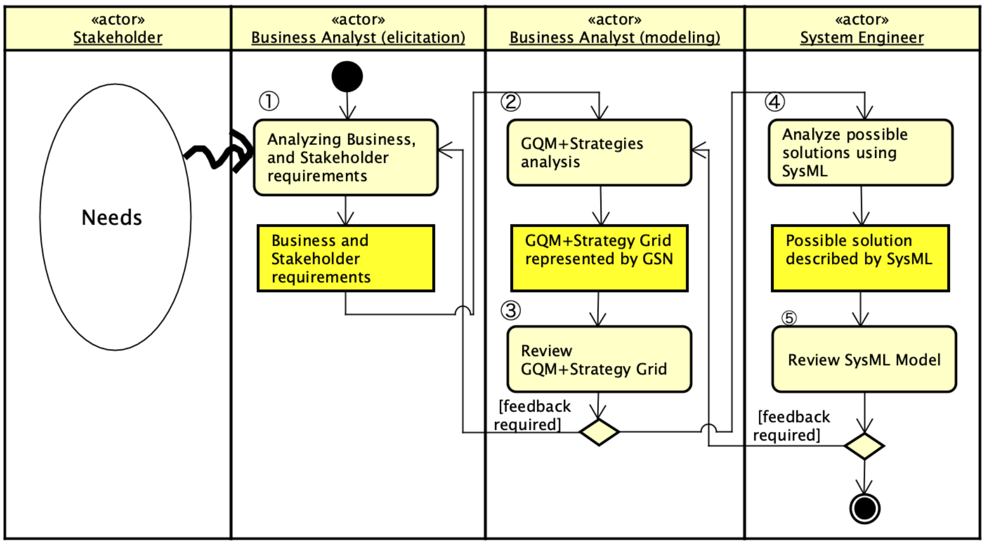

Figure 1: Proposed process to integrate BABOK, GQM$^+$Strategies, GSN, and SysML.

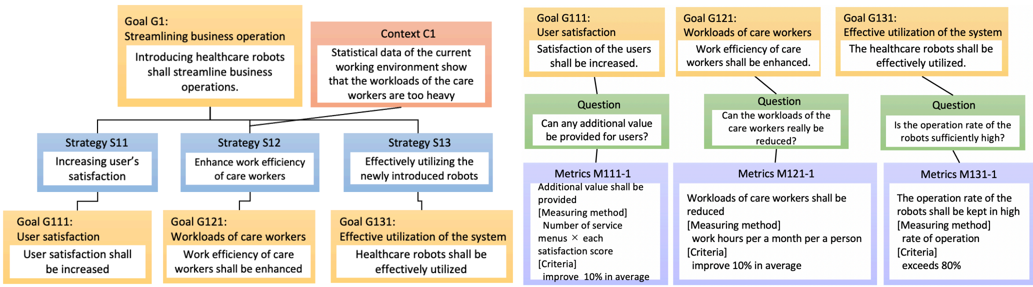

Figure 2: Example of a GQM$^+$Strategies grid

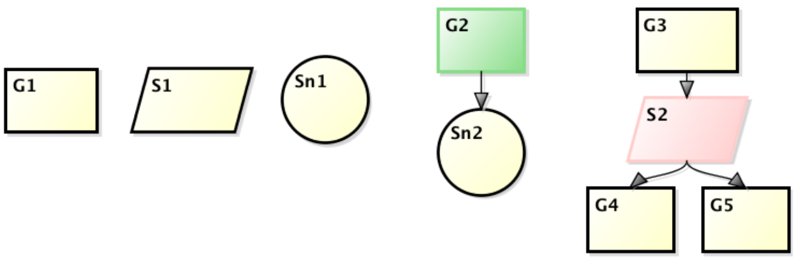

Figure 3: Elements and fragments of GSN

The root of a GSN argument is a goal. A goal is supported by a solution (e.g., G2 in Fig. 3) or a strategy (e.g., G3 in Fig. 3). A strategy may have one or multiple goals. Leaves represent solutions.

3 COMP4BA-IoT Approach

We previously proposed a process to manage business and stakeholder requirements with BABOK, GQM+Strategies, and SysML, and discussed its benefits [9]. Although GQM$^+$ Strategies is a useful tool for managing business and stakeholder requirements in IoT system development, it has the following shortcomings:

1. Representing the goal structures in SysML requirement diagrams is complicated and redundant.

2. GQM$^+$Strategies must be translated to SysML requirement diagrams and vice versa.

To overcome the above issues, COMP4BA-IoT describes a GQM$^+$Strategies grid in the GSN representation(Fig. 4). Specifically, we consider the GSN representation and its advantages for a GQM$^+$Strategies grid.

Goals and strategies in a GQM$^+$Strategies grid can be transcribed since GSN represents a goal with a rectangle and a strategy with a diamond (Fig. 4, S1). The barrel in GSN (Fig. 4, C1), ellipse, and circle indicate a context, assumption, and solution, respectively (Fig. 5).

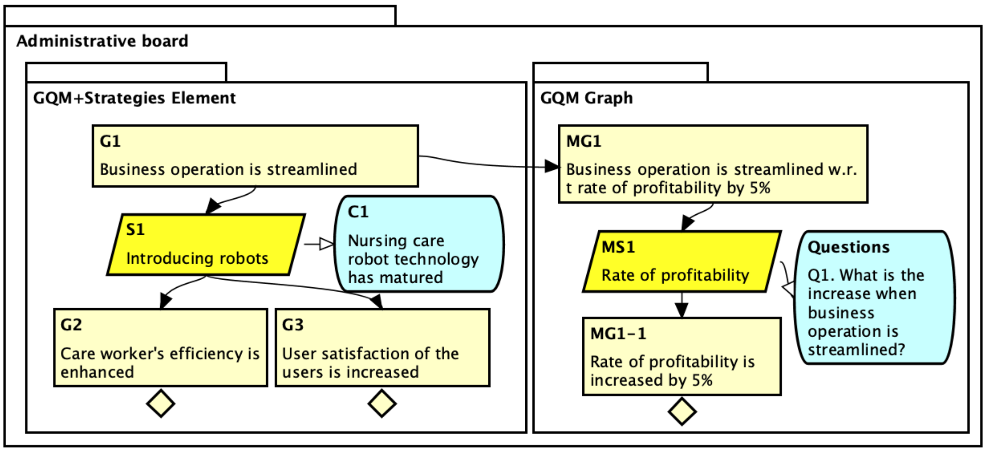

Figure 4: Example of a GQM$^+$ Strategies grid written in GSN.

Table 1 shows the correspondence mapping between GQM$^+$Strategies and GSN element types, and Fig. 5 shows the general form representing a GQM$^+$Strategies grid.

A GSN diagram can effectively represent all information in GQM$^+$Strategies. Additionally, the solution nodes in a GSN diagram provide evidence that the associated goal can be implemented. We previously reported that blocks associated with the requirement model elements via a “satisfy” relation in a requirement diagram in SysML play a similar role [9]. The solution nodes in a GSN diagram correspond to two types of evidence in a GQM Graph: (1) evidence or data showing that the corresponding criterion of the metric has been met and (2) evidence showing that the data for the corresponding metric can be

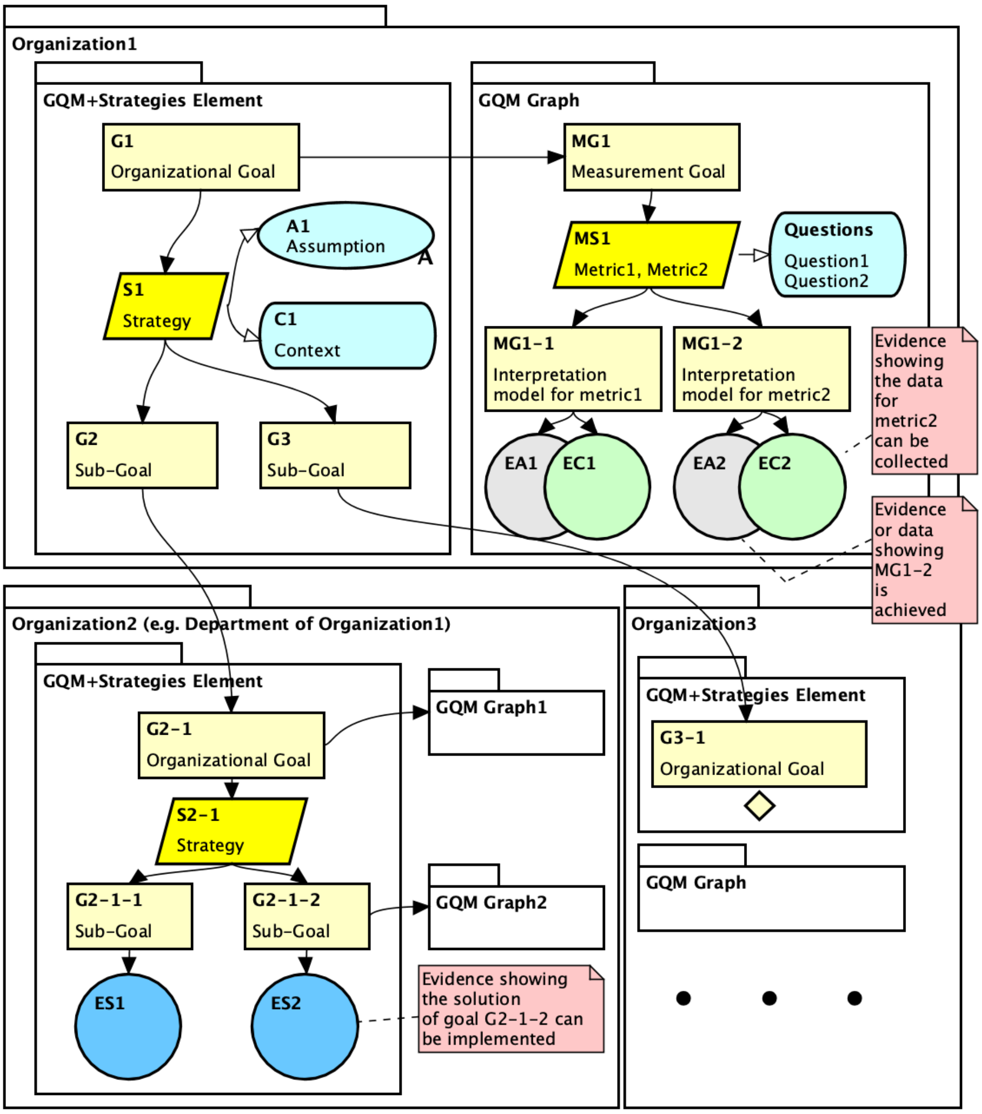

Figure 5: General form of GSN representing a GQM$^+$Strategies grid

Table 1: Mapping from GQM$^+$Strategies to GSN

GQM$^+$Strategies grid element		**GSN element**
Organization		Module
GQM$^+$ Strategies element	Organizational goal	Goal
	Strategy	Strategy
	Assumption	Assumption
	Context	Context
GQM Graph	Measurement goal	Goal
	Question	Context(*1)
	Interpretation model	Goal(*2)
	Metrics	Strategy
Related information		
Evidence that the solution of a goal can be implemented		Solution(*3)
Evidence or data that the interpretation model of a metric is achieved		Solution(*4)
Evidence that the data for a metric can be collected (i.e., measurement method)		Solution(*4)

(*1) Context of a strategy representing a metric
(*2) Sub-goal of a strategy representing a metric
(*3) Solution of an atomic goal in GQM$^+$Strategies Element
(*4) Solution of a goal representing the interpretation model of a metric

collected. Type (1) is used in the evaluation phase (i.e., in the basic GQM$^+$Strategies process, to execute plans and analyze outcomes steps [7]). A typical example of type (2) is an IoT service that collects data for a corresponding metric.

After implementing the GQM$^+$Strategies steps, the obtained GQM$^+$Strategies grid depicted with GSN is reviewed (Fig. 1 ③). We assume that the cycle of the GSN representing the GQM$^+$ Strategies grid does not have solutions. If the GSN is accepted, then a system engineer considers possible *solution requirements* [8] using SysML (Fig. 1 ④).

In practice, the activities with SysML are unrestricted. The next section demonstrates our revised method using an example in which SysML can model and capture the characteristics of IoT systems. A typical first step is to consider use cases of the solution. After specifying the use case descriptions, a system engineer constructs a structural model of the entities. Each entity in the use case descriptions is employed in the behavior models. It should be noted that in the case of IoT systems, the description may contain device-level entities such as sensors, actuators, and any entity identification technologies (e.g., RFID tags) even in the top-level use case. Thus, we adopt the IoT conceptual model in ISO/IEC30131:2018 *Internet of Things (IoT) Reference Architecture* [11].

4 Case Study

COMP4BA-IoT is evaluated following the process explained in Fig. 1.
① (Step 1) A business analyst elicits a list of business and stakeholder requirements. These requirements are written in a natural language and are obtained by the appropriate elicitation techniques explained in the *requirement analysis* part of BABOK [8]. The corresponding step in Fig. 1 is ①. Below is an example of such a list for a healthcare facility.

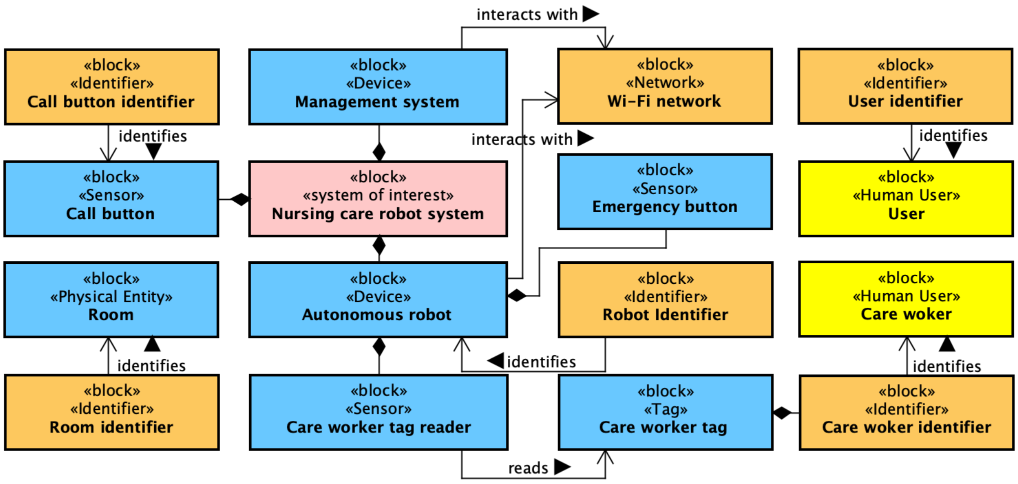

Figure 6: Part of an example description of a solution requirement: structural model of the entities in the usage viewpoint represented by a SysML block definition diagram.

- **Business requirements**
  - Administrative director
    - Business operations shall be streamlined.
    - Care worker's efficiency shall be enhanced.
    - User satisfaction shall be increased.
- **Stakeholder requirements**
  - Care worker
    - Overtime for care workers shall be reduced.
    - Burden for moving users shall be decreased.
    - Paperwork shall be minimized.
  - User
    - User satisfaction shall be increased.
    - Physical activity shall be increased.
    - Users shall safely use all equipment in the nursing facility.

② (Step 2) A business analyst in charge of modeling executes the initial GQM$^+$Strategies trial, as shown in Fig. 1.
③ (Step 3) The GQM$^+$Strategies trial is represented in GSN (Fig. 4)
④ (Step 4) A system engineer considers possible solution requirements with SysML. Fig. 7 shows an example of a top-level use case diagram, while Table 2 provides its use case description.

After specifying the use case descriptions, the system engineer constructs a structural model of the entities (Fig. 6).

This model is used to elaborate the processes for the use case scenarios. For example, the use case "A user with difficulty walking moves from his/her room to a rehabilitation room by riding an autonomous robot" can be elaborated as the activity diagram shown in Fig. 9.

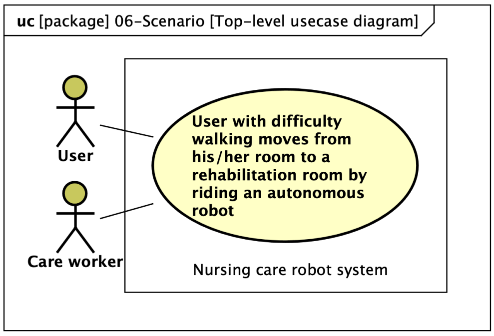

Figure 7: Part of an example description of a solution requirement: interaction scenario in a usage viewpoint model represented by a SysML use case diagram.

Table 2: Part of an example description of a solution requirement: interaction scenario in a usage viewpoint model represented by the use case description

Item	**Content**
Use case	A user with difficulty walking moves from his/her room to a rehabilitation room by riding an autonomous robot
Actor	- User - Care worker
Pre-conditions	- The user is lying in his/her bed. - The bed is equipped with a call button. - The management system is launched. - At least one care worker is waiting who has an identification tag(e.g., RFID). - At least one robot is available.
Post-conditions	- The user arrives at the destination (e.g., rehabilitation room). - The care worker is available for other tasks. - The history of the user's move is recorded in the management system. - The working time of the care worker is recorded in the management system.
Basic flow	1. The user pushes the call button. 2. One robot and one care worker come to the room. 3. The robot reads the tag. 4. The user rides the robot with the help of the care worker. 5. The care worker is released 6. The robot sends the working time of the care worker to the management system, and the system records it. 7. The user provides his/her destination to the robot. 8. The robot autonomously moves to the destination. 9. The robot indicates the arrival to the user. 10. The robot sends the history, which indicates that the move has successfully finished to the management system, and the system records it.
Exceptional flow	During step 8, if an emergency event that requires additional support occurs, the user pushes the emergency button on the robot. Any care worker would assist.

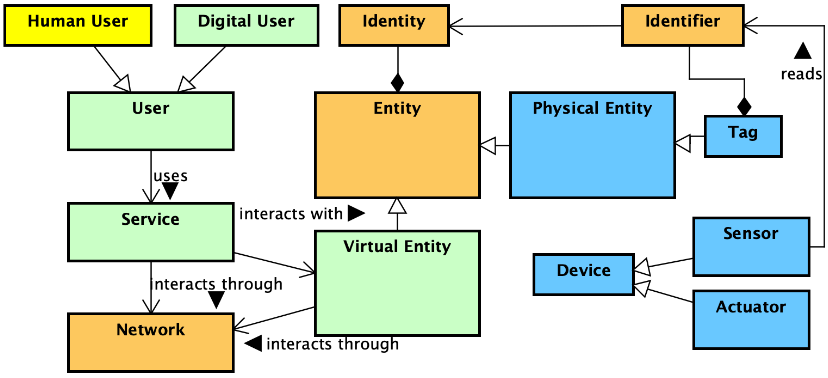

Figure 8: Part of an IoT conceptual model [11]

The IoT conceptual model shown in Fig. 8 indicates the partition type of the subjects for the contained actions. The actions associated with stereotype Service indicate the services, which are characterized in the IoT conceptual model (Fig. 8). First, a user pushes the call button located on his/her bed to request a robot. Because the call button is associated with a specific user and room, the management system can transfer the message with the appropriate user and room information. Then the message is sent to a care worker and an autonomous robot. Both proceed to the room. The action "move to the room on foot" is modeled as a send signal action since it is also a trigger for another activity defined in Fig. 10, which is explained later. Once the care worker and the robot arrive to the user's room, the user tries to ride the robot with the assistance of the care worker. If the user successfully rides the robot, the care worker is available for another task. The user inputs the desired destination into the robot. This operation can abstractly be seen as transferring the identifier of the destination room to the robot.

Another activity diagram is used to satisfy the post-conditions "the history of the user's move is recorded in the management system" and "the working time of the care worker is recorded in the management system" (Fig. 10). It should be noted that Figs. 9 and 10 have some common actions. For example, the send signal actions "Move to the room on foot", "Released", and "Arrival notification" in Fig. 10 are associated with those in Fig. 9. In Fig. 9, the roles of services "Record working time" and "Record user activity" can be understood through their relation to other services and their triggers.

⑤ (Step 5) The GQM$^+$Strategies grid (Fig. 4) is elaborated using the identified services to deal with the remaining stakeholder requirements. Fig. 11 shows a GSN diagram to describe the GQM$^+$Strategies grid for care workers as an example.

Solution EC2-1-1-1 refers to the service "record working time" identified in the process model of usage in Fig. 10. Both ES2-1-1-1 and ES2-1-1-2 represent the "autonomous robot" identified in the structural model of usage in Fig. 6 using the service definitions in Figs. 9 and 10.

After several cycles of steps ② to ⑤ in Fig. 1, the final GQM$^+$Strategies grid is generated (Fig. 12). That is, the grid shows that the goals of the administrative board, care workers, and users are consistently aligned with appropriate evidence.

The case study reveals the following:

- Business requirements elicited by business analysts (Section 3) can be associated with the elements of structured solution requirements with explicit traceability. For example, analyzing the stakeholder requirement "Care burden for moving users shall

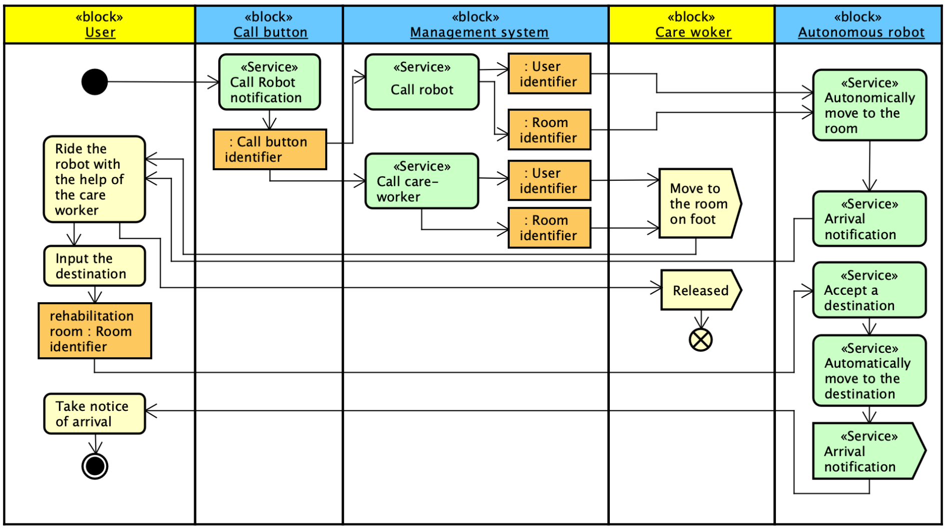

Figure 9: Part of an example description of the solution requirement: process model of the main use case represented by a SysML activity diagram.

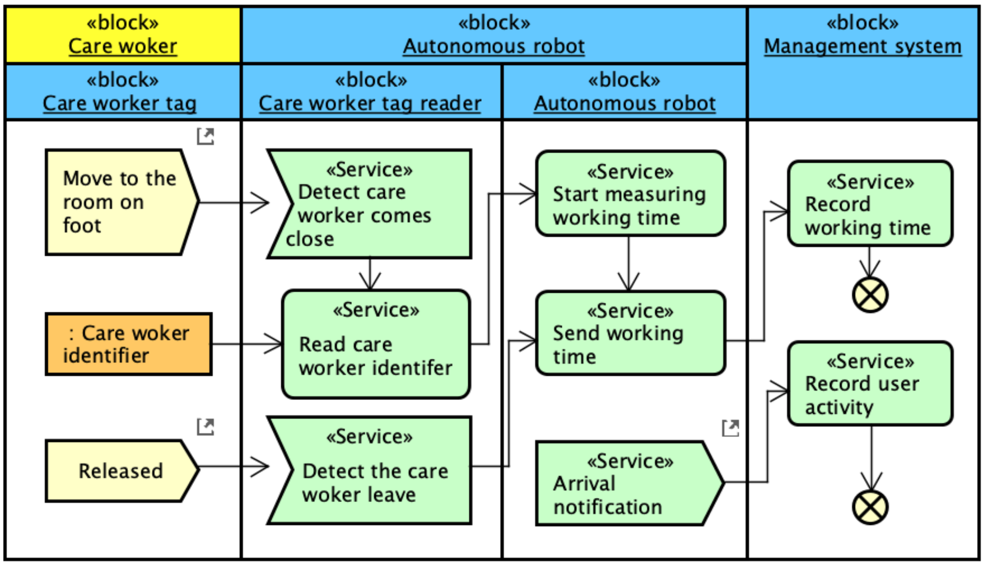

Figure 10: Part of an example description of a solution requirement: process model of the recording related activities represented by a SysML activity diagram.

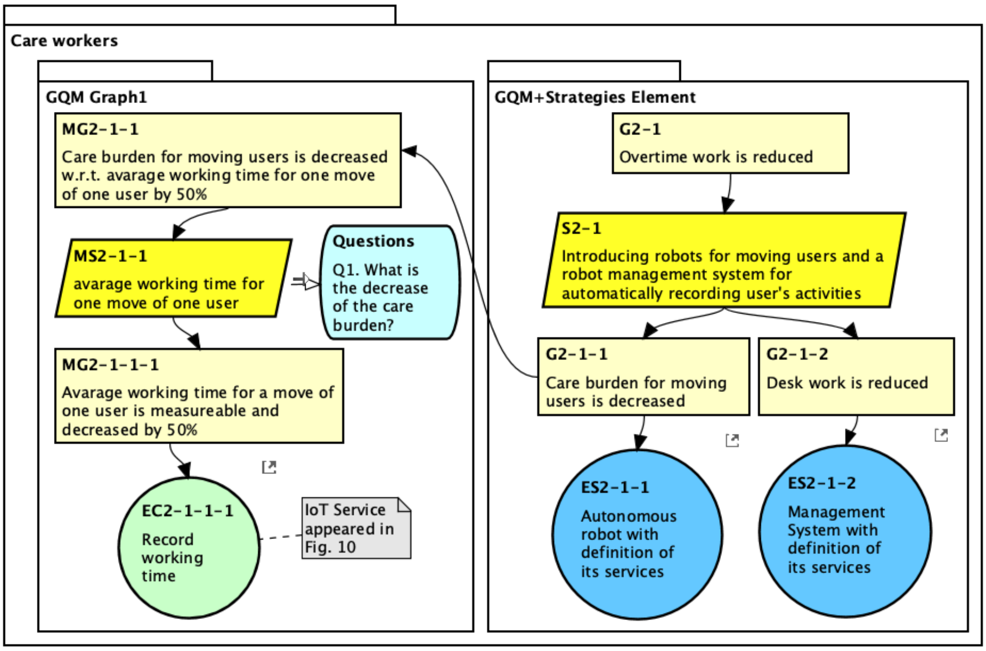

Figure 11: GQM$^+$Strategies grid for care workers represented by GSN

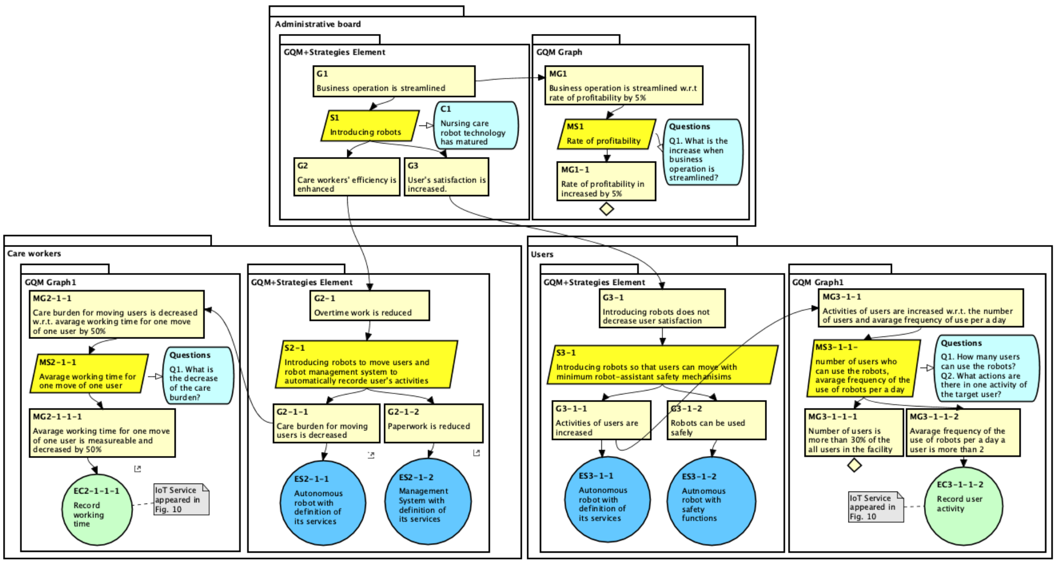

Figure 12: Image of a GQM$^+$Strategies grid of several stakeholders represented by GSN.

be decreased" indicates it is a sub-goal of another stakeholder requirement "Overtime for care workers shall be reduced", which has a goal description of G2-1 "Overtime work is reduced" and an explicit strategy S2-1. Moreover, the sub-goal is supported by ES2-1-1, which refers to the element Autonomous robot in the model. The behavior requirement, which is an element defined in Fig. 9, can verify whether the system provides a measure to reduce the burden of care workers.

- A measurement metric defined through the activities of GQM$^+$Strategies can be measured when the measurement is executed via the functions provided by the target system. For example, metrics MG2-1-1 for the sub-goal G2-1-1 is associated with another sub-goal MG2-1-1-1, indicating that the metrics can be measured. Here, sub-goal MG2-1-1-1 is supported by evidence EC2-1-1-1, which refers to the service "Record working time" defined in Fig. 10 provided by the Management system.

5 Related Works

5.1 Processes and methods for IoT system development

Silva et al. [13] considered a process for requirement engineering activities in IoT systems. Their process consists of three sub-processes: project scope definition, IoT system definition, and IoT system requirements definitions. The project scope definition includes the steps necessary to analyze the problem or opportunity. COMP4BA-IoT can be regarded as a concrete method for these steps. Although several proposals have focused on generating codes from models in IoT system development (i.e., model-driven development [14–18]), we focus on continuous supports from business and stakeholder requirements to systems engineering.

5.2 Integrated multi-level modeling for business and system alignment

N. Mimura et al. proposed a method to systematically align business requirements and system functions by linking GQM$^+$Strategies and SysML models at the metamodel level [22]. Their method ensures traceability from business requirements to system functions and fills the gap. They use requirement diagrams in SysML to integrate business requirements and system functions. In contrast, our method utilizes GSN, which can be used by both system engineers and business analysts. M. E. Hamlaoui et al. proposed an approach to link different models using a metamodel and various relationships, including aggregation and generalization among model elements [24]. Their approach uses a specialized business-level model for the systems design model and business processes. Here, we suppose the metamodel can be applied as a basis of our method. X. Cui et al. proposed a framework that integrates the development of motivation and requirements models at organization, business, product, and system/software levels [20]. Their framework contains the motivations described by Business Motivation Models and requirements by Requirements Models of SysML from the top-level organization motivation to the low-level product requirements. The method only concerns qualitative statements to represent requirements whereas our method can deal with quantitative goals and their relations. ArchiMate is an enterprise architecture modeling standard, which consistently expresses both business-level and system-level models [25]. It has six layers (strategy, business, application, technology, physical, and implementation & migration) and four elements (passive structure, behavior, active structure, and motivation).

ArchiMate provides a wide variety of notions and notations for a business-level architecture. However, it cannot systematically manage quantitative metrics. Using our proposed method with ArchiMate to represent business architectures should realize a better effect.

5.3 Other model enhancements and integration approaches

Reggio [12] proposed a method to describe IoT system requirements based on UML. Their method employs a use case diagram to decompose goals. COMP4BA-IoT can be seen as a concrete procedure to decompose goals with metrics. J. Gardan et al. proposed an approach to provide conceptual models that enhance a Model-Based Systems Engineering (MBSE) and consider knowledge management (KM) through SysML [23]. Their approach is intended to provide an organizational solution integrating cooperative work and KM in Systems Engineering. We expect our method can be extended by integrating a KM process with their method. N. B. Wilson et al. proposed an agile methodology named MADAIKE, which promotes the integration of various international standards and justifies the value to the organization [21]. It is an integrated methodology composed of layers. Each layer represents a project phase with a broad perspective across phases. Their methodology covers the processes and roles that are beyond the scope our proposal such as implementation and deployment processes. However, they do not mention metrics associated with requirements or architectures. J. V. Brocke et al. proposed an approach to model both factual and value-oriented potentials and integrated a value-oriented perspective into information modeling [19]. Their approach provides a means to evaluate information systems design alternatives in a specific organizational context to analyze and assess the economic value of IT investments. Their methodology is applicable to service-oriented architecture. In contrast,our method can effectively apply IoT Systems. In their method, the concepts “fact” and “value” play significant roles, while these are implicitly assumed in our method. We expect the process of COMP4BA-IoT can be refined based on these concepts.

6 Summary

This study proposes a tool chain to support and manage requirements called the COMP4BA-IoT approach. COMP4BA-IoT is a refinement of our previous approach, which adopted SysML parametric diagrams to express the relationship between measurement goals and a measure of effectiveness (MoE) for the system of interest [9].

GSN diagrams, which represent a GQM$^+$Strategies grid in IoT system development and solution nodes, are beneficial to connect measurement goals and their measurement means realized by IoT functionalities. Using SysML parametric diagrams in COMP4BA-IoT can describe solution requirements in which decision making is supported by accountable evidence.

Acknowledgement

This research is partially supported by the *enPiT-Pro Smart SE: Smart Systems and Services Innovative Professional Education Program*, and JSPS KAKENHI (Grant Number 19K11914).

References

[1] CASAGRAS. RFID and inclusive Model for the Internet of Things: Technical Report. EU Framework 7 Project. 2008.

[2] Rebeca C. Motta, Káthia M. de Oliveira, and Guilherme H. Travassos. 2018. On challenges in engineering IoT software systems. In Proceedings of the XXXII Brazilian Symposium on Software Engineering (SBES'18). Association for Computing Machinery, New York, NY, USA, 42–51. DOI: https://doi.org/10.1145/3266237.3266263

[3] Shortell, Thomas M., ed. INCOSE Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities. John Wiley & Sons, 2015.

[4] OMG. OMG Systems Modeling Language (OMG SysML) v.1.4, 2015.

[5] Friedenthal, Sanford, Alan Moore, and Rick Steiner. A practical guide to SysML: the systems modeling language. Morgan Kaufmann, 2014.

[6] Goal Structuring Notation Working Group, GSN Community Standard Version 2, 2018.

[7] Basili, V., Heidrich, J., Lindvall, M., Munch, J., Regardie, M., and Trendowicz, A. "GQM^{+}Strategies–aligning business strategies with software measurement." First International Symposium on Empirical Software Engineering and Measurement (ESEM 2007). IEEE, 2007.

[8] IIBA. A Guide to the Business Analysis Body of Knowledge (BABOK Guide). International Institute of Business Analysis, 2015.

[9] Takai, T., Shintani, K., Andoh, H., and Washizaki, H. "Case Study Applying GQM+Strategies with SysML for IoT Application System Development." 2019 8th International Congress on Advanced Applied Informatics (IIAI-AAI). IEEE, 2019.

[10] Toshinori Takai, Katsutoshi Shintani, Hideki Andoh and Hironori Washizaki, "Continuous modeling supports from business analysis to systems engineering in IoT development," 5th International Conference on Enterprise Architecture and Information Systems (EAIS 2020), September 1-15, 2020.

[11] ISO/IEC, ISO/IEC 30141:2018, Internet of Things (IoT) – Reference architecture, ISO/IEC, 2018.

[12] G. Reggio, A UML-Based Proposal for IoT System Requirements Specification, IEEE/ACM 10th International Workshop on Modelling in Software Engineering (MiSE), Gothenburg, Sweden, 2018, pp. 9-16, 2018.

[13] D. Silva, T. G. Gonçlves, and Ana Regina C. da Rocha, A Requirements Engineering Process for IoT Systems. In Proceedings of the XVIII Brazilian Symposium on Software Quality (SBQS'19). Association for Computing Machinery, New York, NY, USA, 204–209, 2019.

[14] P. Patel and D. Cassou. Enabling high-level application development for the Internet of Things. *Journal of Systems and Software*, 103:62–84, 2015.

[15] X. T. Nguyen, H. T. Tran, H. Baraki and K. Geihs, FRASAD: A framework for model-driven IoT Application Development, 2015 IEEE 2nd World Forum on Internet of Things (WF-IoT), Milan, 2015, pp. 387-392.

[16] Ciccozzi F., and Spalazzese R. “MDE4IoT: Supporting the Internet of Things with Model-Driven Engineering.” In: Badica C. et al. (eds) Intelligent Distributed Computing X. IDC 2016. Studies in Computational Intelligence, vol. 678. Springer, 2017.

[17] M. Hussein, S. Li, A. Radermacher. “Model-driven development of adaptive IoT systems”, 2017 MODELS Satellite Event, Austin, United States, pp.17-23, Sep 2017.

[18] Brambilla, M., Umuhoza, E. and Acerbis, R. Model-driven development of user interfaces for IoT systems via domain-specific components and patterns. J Internet Serv Appl 8, 14 (2017).

[19] J. V. Brocke, C. Sonnenberg, A. Simons, “Value-Oriented Information Systems Design: The Concept of Potentials Modeling and its Application to Service-oriented Architectures,” Business & Information Systems Engineering, June 2009, pp. 223-233.

[20] X. Cui, R. Paige, “An Integrated Framework for System/Software Requirements Development Aligning with Business Motivations,” IEEE/ACIS 11th International Conference on Computer and Information Science, June 2012, pp. 547-552.

[21] N. B. Wilson, L. A. Carmenza, “Agile Methodology to Develop Architecture of Information and Knowledge in Organizations (MADAIKE),” International Journal of Machine Learning and Computing, Vol. 5, No. 2, April 2015, pp. 153-159.

[22] N. Mimura, S. Okuda, H. Washizaki, K. Shintani, Y. Fukazawa, “Systematical Alignment of Business Requirements and System Functions by Linking GQM+Strategies and SysML,” International Journal of Service and Knowledge Management (IJSKM), Vol. 5, No. 1, pp. 15-35, 2021.

[23] J. Gardan, N. Matta, “Enhancing Knowledge Management into Systems Engineering through New Models in SysML,” Proceedings of the 27th CIRP Design Conference, Procedia CIRP, Vol. 60, pp. 169-174, 2017.

[24] M. E. Hamlaoui, S. Bennani, M. Nassar, S. Ebersold, B. Coulette, “A MDE Approach for Heterogeneous Models Consistency,” Proceedings of the 13th International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE), pp. 180-191, 2018.

[25] C. Atkinson and T. Kühneb, “A Deep Perspective on the ArchiMate Modeling Language and Standard,” Enterprise Modelling and Information Systems Architectures, Vol. 15, No. 2, pp. 1-25, 2020.