International Journal of Service and Knowledge Management
International Institute of Applied Informatics
2021, Vol. 5, No. 1, 15 – 35

Systematical Alignment of Business Requirements and System Functions by Linking GQM+Strategies and SysML[1]

Natsuki Mimura *, Shuji Okuda *, Hironori Washizaki *, Katsutoshi Shintani *, Yoshiaki Fukazawa *

Abstract

Many businesses strive to align their business goals, strategies, and systems before and during system development as the studies of GORE and alignment of business/IT show. However, the level of success derived from their interrelationships may be ambiguous because the relationships between goals, strategies, and systems are often vague. As agile development becomes more prevalent, it is becoming more difficult to adjust a system to accommodate changing goals and strategies. By linking GQM+Strategies and SysML, we propose a method to systematically align business requirements and system functions. Then we evaluate this method from the viewpoint of traceability from business requirements to system functions and the solution to fill the gap between them. This proposed method not only helps maintain consistency from business requirements to system functions but also solves the gap between customers' needs and the developed system while simultaneously dealing with changing strategies and requirements.

Keywords: Alignment of business goals and system requirements, GQM+Strategies, link of models, SysML

1 Introduction

Businesses are becoming increasingly aware of the importance of software and IT in present and future business strategies [2]. Many businesses align their business goals and strategies to validate business processes, but these relationships are often vague. Thus, the level of success derived from their interrelationships may be ambiguous.

---

[1] This paper is an extension of our previous work, "Linking business strategies and system demands using GQM+Strategies and Systems Modeling Language," EAIS 2019 [1]. Here, we add a supplemental explanation for model linkage and model deployment. In addition, explanations about problems in actual system development that our method can help solve and related works are expanded.

* Waseda University, Tokyo, Japan

GQM+Strategies [3] is one method to solve this issue. It is an integrated approach that creates a hierarchical model to more ideally align goals and strategies at different levels from high-stake business strategies to individual development projects. On the other hand, modeling is becoming increasingly important in system development. The demand for system product development processes is becoming increasingly diversified and complicated. Consequently, addressing challenges related to system integration due to vague specifications and increasing complexity is crucial.

Systems Modeling Language (SysML) [4] is a modeling language to solve these issues. SysML visualizes and organizes complex demands to grasp system boundaries and to elucidate system requirements. However, adjusting the system to accommodate changing goals and strategies is difficult due to the increased prevalence of agile development [5].

By linking GQM+Strategies and SysML, we propose a framework to systematically align business requirements and system functions. Through a case study, the proposed method is validated using the relationships between business strategies and system functions. This paper contributes to the alignment of business goals and IT systems. The proposed method is useful to ensure that business strategies are aligned with the IT systems introduced in the organization. This method can also maintain the consistency of business strategies and system requirements by linking GQM+Strategies and SysML at the meta-model level.

2 Background and Challenges

2.1 GQM+Strategies

GQM+Strategies [3] introduces the GQM approach [6] to measure goal orientation and align goals and strategies across all levels of an organization using the goal orientation requirement approach. The elements of GQM+Strategies indicate the inter-relationships among a goal, strategy, and rationale (context and assumption). A goal is defined as a measurable and attainable objective within an organization. A strategy is defined to achieve this goal. Furthermore, factors influencing the goal and strategy definitions are the contexts and assumptions that provide the rationale in relation to an organizational environment. Based on a series of goals and strategies, lower-level goals are defined in a hierarchy. Applying this approach provides a hierarchical model of goals and strategies, which is similar to the organizational structure [7].

Organizational goals correspond to the GQM graph, allowing the results of goal attainment and a strategy to be evaluated. The graph has a tree structure comprised of a goal, question, and metric (together called GQM). In the GQM approach, questions, which evaluate goal attainment, are divided into organizational goals to measure the target characteristics. Then metrics provide the most relevant information to answer the questions [3][6].

2.2 SysML

SysML [4] is a modeling method to support a wide range of complex specifications, analyses, designs, verifications, and tests in hardware and software systems [8]. There are several modeling patterns. SysML is an extended UML specific to systems modeling. In this research, we use a Requirement Diagram specifically modified for systems modeling to link with GQM+Strategies. By linking SysML and GQM+Strategies, the validity of the SysML model can be complemented based on data and assumptions. The elements called "requirements" describe the functions and conditions that the system must satisfy. The requirements, which are linked in a tree structure, can be divided into functional and non-functional requirements. In addition, a related element called a "block" shows the concrete steps to satisfy the detailed breakdown of the individual requirements. A "rationale" describes the reason for a requirement. The linkage between elements has a stereotype to define each relationship.

2.3 Method to Link Multiple Models

Because complex systems are often designed using different models, previous works have proposed a method to maintain consistency between business models and system models during complex system design [9][10]. However, to understand the overall system, all models must be linked. Figure 2.1 shows part of the Metamodel of Correspondences (MMC), which was proposed by M.E. Hamlaoui et al. In the MMC model, a "relationship" defines the elements between models, and the overview can be expressed by automatically changing the elements of the model by "relationship" type. Their research used a specialized business-level model for the systems design model and business processes. Similarly, our linkage method is based on these relationship types, but unlike their model, we use a business-level model specialized in business goals and strategies.

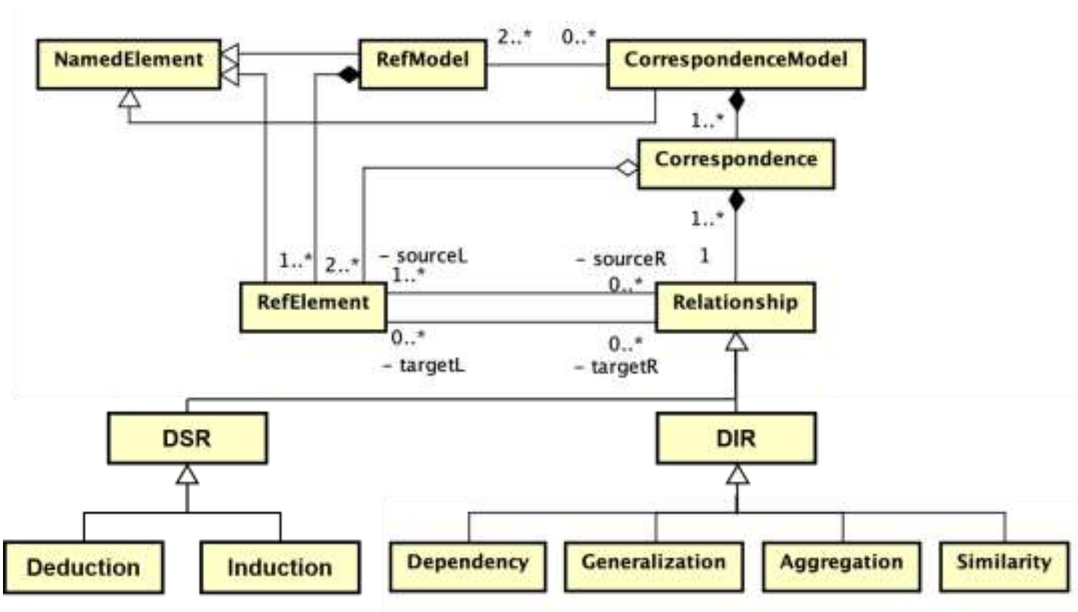


Figure 2.1: Part of MMC by UML

2.4 Motivating Example

Business goals and strategies are formulated by business owners or strategy decision makers. Figure 2.2 overviews system development processes for a business. Based on the overarching goals and strategies, each department such as the sales or production department manages tasks to achieve business goals. When a department plans to systematize tasks, the system department receives the departmental requests and works on system development in collaboration with vendor businesses. On the other hand, tasks that are not systematize are sometimes executed by the business department itself.

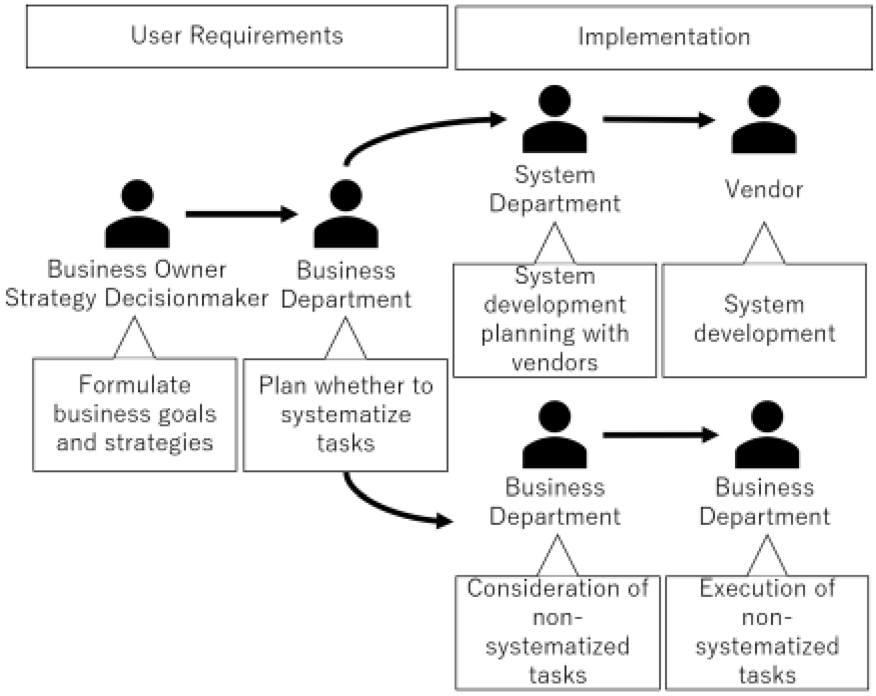


Figure 2.2: Business system development processes

In an actual system development, an effective requirement process is necessary to realize a successful project. A major problem is the gap between the needs of the customers and the system developed [11]. Figure 2.3 depicts the flow in the requirement definitions related to the system gap. Often the original requirements for systematization by a department are overlooked during the requirement definitions. Additionally, ambiguous requirement definitions often result in misunderstandings or exaggerated interpretations. In the present systems engineering method, the cooperative aspect is often not considered even though systems engineering is a cooperative activity [12]. The lack of a cooperative aspect causes these problems.

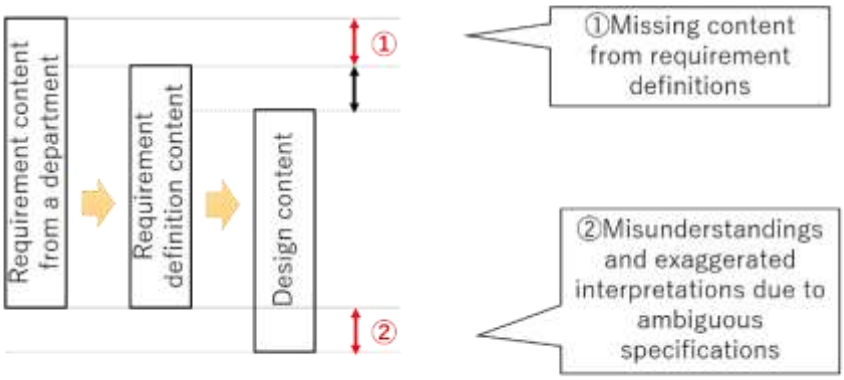


Figure 2.3: Gaps in the system related to the requirement definition

In addition, goals, business processes, and requirements typically change in an actual system development. However, many organizations cannot effectively respond to evolving goals and requirements [5]. For project success, decision making must reflect the adjustments across strategical, tactical, and operational levels [13]. First, when strategies are changed, which part of the system to change needs to be understood. For example, if a strategy decision maker decides to change a strategy, as shown in Figure 2.4, the system designer may be unsure about which requirements to change.

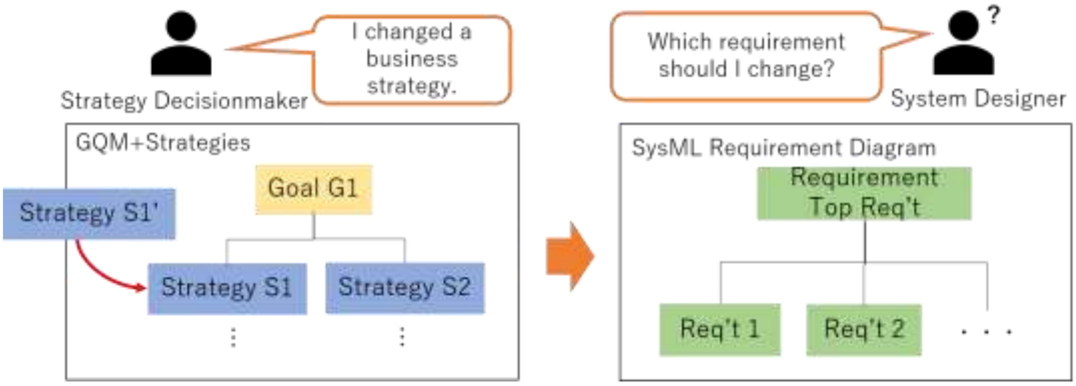


Figure 2.4: Issues in system development (business-to-system)

Second, whether business goals and strategies can be achieved after changing the system or whether business goals or strategies need to be changed must both be confirmed. For example, when a system designer changes a system requirement, the strategy decision maker may be unsure if the business objective is achievable or if the strategies must be changed (Figure 2.5).

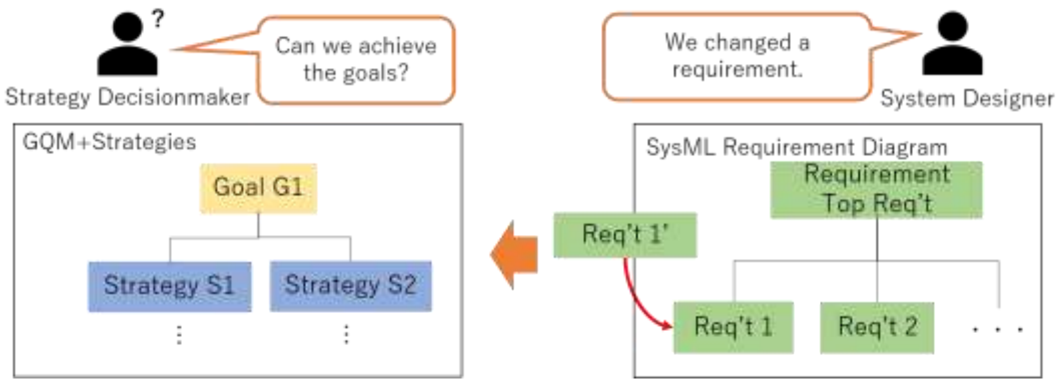


Figure 2.5: Issues in system development (system-to-business)

To align system requirements and business strategies, our research integrates two models: GQM+Strategies and SysML. The former is used to select business strategies and tactics, while the latter is a system development model. Our proposal has three main benefits. First, adequate requirement definitions that incorporate business goals are created, providing a solution to the gap problem explained in Figure 2.3. Second, the requirements corresponding to the goal or strategy that needs to be changed are clear, addressing the problem in Figure 2.4. Finally, the level of goal achievement can be confirmed by the system requirement changes, to solve the problem in Figure 2.5. In addition, it is easier to understand the model for both the user and vendor businesses by modifying the GQM+Strategies graph and requirement diagrams of SysML. These improvements can prevent misunderstandings and exaggerated interpretations shown in Figure 2.3.

This paper addresses the following research questions:

- RQ1 (Validity): Can this model trace the relationship from business goals and strategies to the system requirements?

It is necessary to confirm the consistency from business goals and strategies to system requirements. Satisfying this question means our method aligns business strategies and system requirements correctly. To answer this, we conducted a case study and created an alignment diagram of business goals and strategies of GQM+Strategies and system requirements of SysML.

- RQ2 (Effectiveness): Can our method solve the gap between customers' needs and the developed system, while dealing with changing strategies and requirements?

It is necessary to confirm effectiveness of our method for the problems identified in Figures 2.3–2.5. If this question is satisfied, it means the developed system is consistent with customers' needs. In addition, the system development process can handle changes in strategies and requirements. To answer this, we asked experts to evaluate whether our method helps solve these problems.

3 Model Linkage and Deployment

By linking GQM+Strategies and SysML, we can maintain consistency among business goals, strategies, and system requirements. This is achieved by defining a framework that aligns business requirements and system requirements as well as defining model deployment for each newly defined linkage type. A similar model has been proposed by Takai et al. [16].

In this chapter, we first show the process of our method utilization. Then we propose a model linkage of GQM+Strategies and SysML. Finally, we discuss model deployment in each defined linkage.

3.1 Method Utilization

Figure 3.1 shows the process of our method. There are three scenarios: one main and two additional scenarios. The main scenario is always used in the system development process. The business owner decides the business goal, business analysts align business goals and strategies, and system engineers derive system requirements. Then business analysts and system engineers confirm that each business strategy and system requirement correspond and align them by our model linkage method (Figure 3.1, No. 1). Using the linked model, system developers develop the system. First, the additional scenario is used when a business strategy must be changed under the system development. System engineers refer the linked model, change the requirement, and realign the model by our model deployment method

(Figure 3.1, No. 2). Then the second additional scenario is used when system requirement must be changed under the system development. Business analysts refer to the linked model, change the business strategy, and realign the model by our model deployment method (Figure 3.1, No. 3).

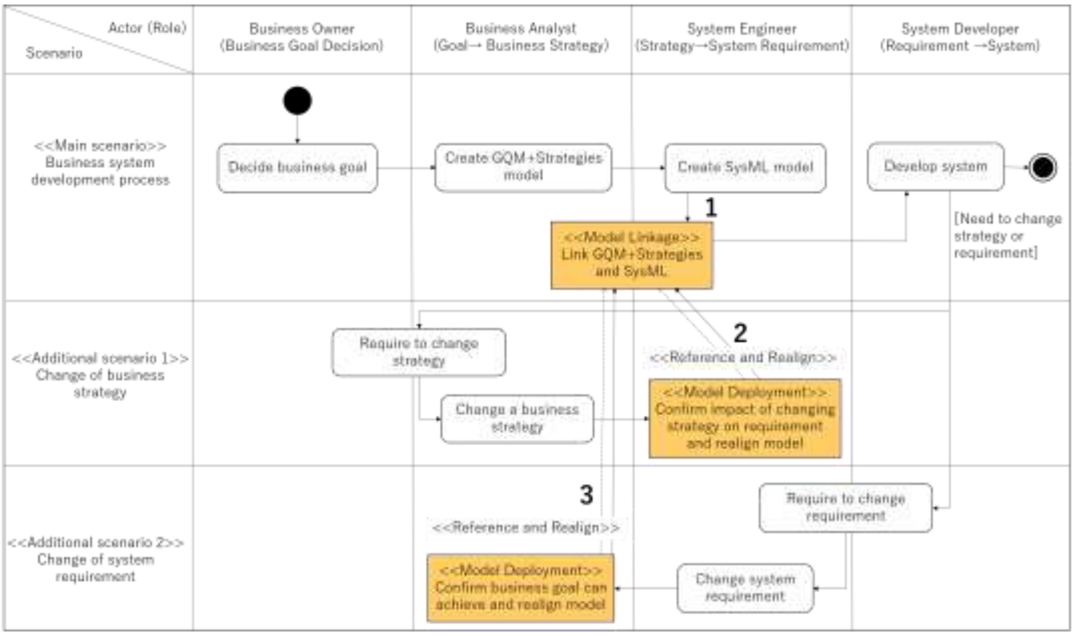


Figure 3.1: Process of method utilization

3.2 Model linkage of GQM+Strategies and SysML

Figure 3.2 shows the metamodel linking GQM+Strategies and SysML. Each metamodel of GQM+Strategies and SysML is created by referencing C. Shimura et al. [14]. When the two models (GQM+Strategies and SysML) are linked, new vertical linkages between the models are defined. To describe these linkages in more detail, we use a case study called "visualizing the health system" of the Health Promotion Plan by Kashiwa City, Chiba [15]. Figure 3.3 shows a simple diagram of the six possible linkages, which are based on MMC [9][10].

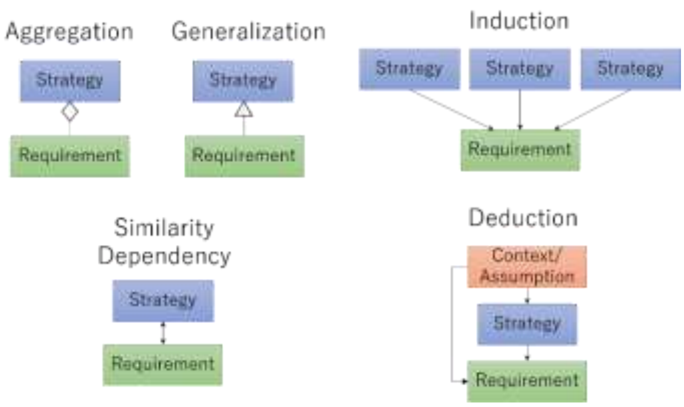


Figure 3.3: Simple diagram of the relationships among elements in each linkage

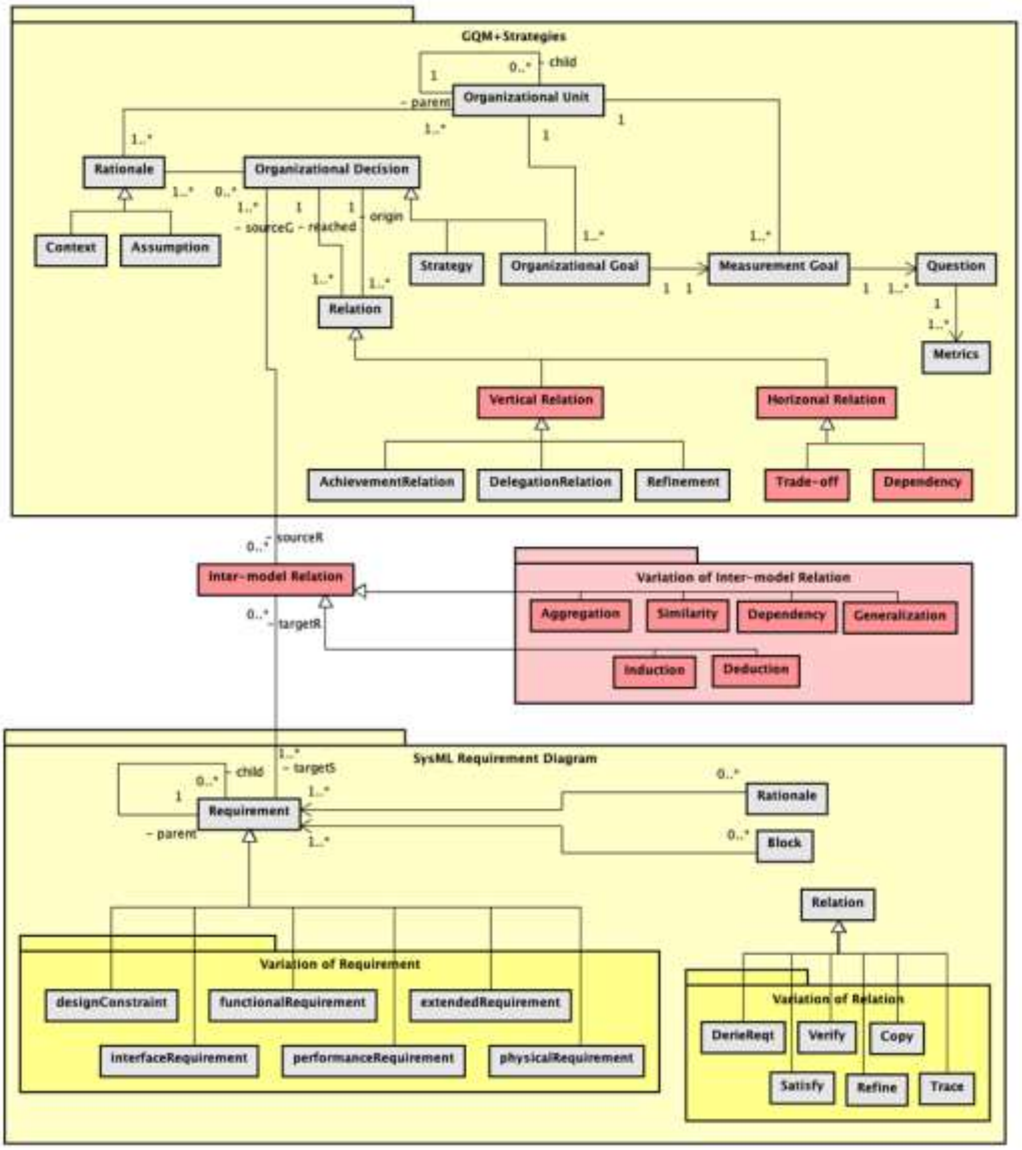


Figure 3.2: UML metamodel of GQM+Strategies and SysML requirement diagram

(1) Aggregation

An aggregation, which is based on the Domain Independent Relationship (DIR) of MMC [9][10], is a relationship composed of a single strategy for multiple requirements. When the “whole – part” relationship is established, this model is used. For example, Figure 3.4 shows the health conditions, which consist of data such as step count, calorie consumption, and weight. These conditions can clearly be aggregated.

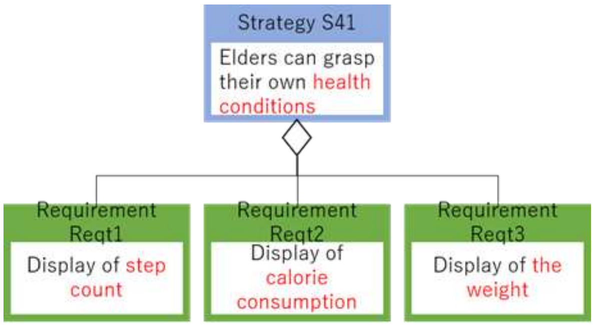


Figure 3.4: Example of aggregation

(2) Generalization

Generalization, which is based on DIR [9][10], is a relationship created by linking an element without a parent in a requirement diagram to an abstract goal or strategy as its parent. This defines the visualization range of the strategy and the system domain. For example, Figure 3.5 shows the created system from strategy S22.

Figure 3.5: Example of generalization

(3) Similarity

Similarity, which is based on DIR [9][10], is a relationship where a strategy can be replaced by a requirement. For example, Figure 3.6 shows the strategy “use a physical activity meter”, which has a similar meaning as the requirement “obtain data from a physical activity meter”.

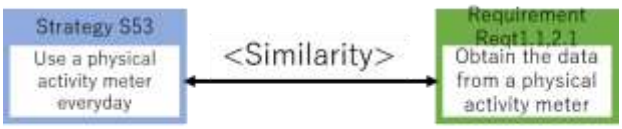
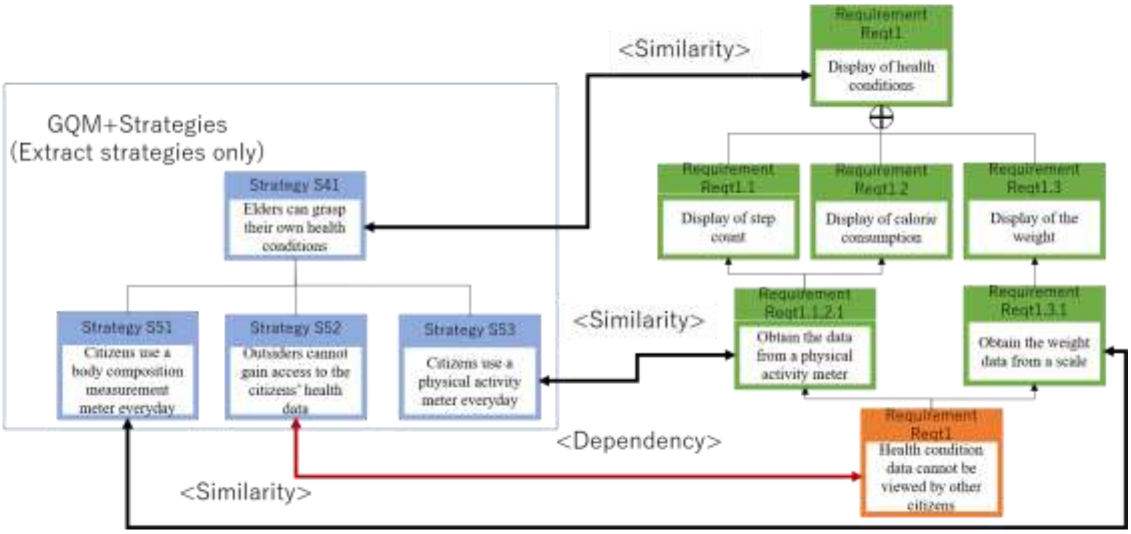


Figure 3.6: Example of similarity

(4) Dependency

Dependency, which is based on DIR [9][10], is like the similarity relationship in a sense that a strategy can be replaced by a requirement. The difference is depicted using the example in Figure 3.7. While strategies S51, S52, and S53 in the GQM+Strategies model are aligned horizontally, the requirement of strategy S52 links the requirements of strategies S51 and S53 in the SysML requirement diagram vertically. If such parent-child relationships of strategies in GQM+Strategies and that in SysML-requirement-diagram are aligned, then use similarity. Otherwise, use dependency.

Figure 3.7: Example of similarity and dependency



(5) Induction

Induction, which is based on the Domain Specific Relationship (DSR) of MMC [9][10], is used when a single requirement is derived from multiple strategies that do not have parent-child relationships. For example, Figure 3.8 shows a single requirement satisfying multiple strategies.

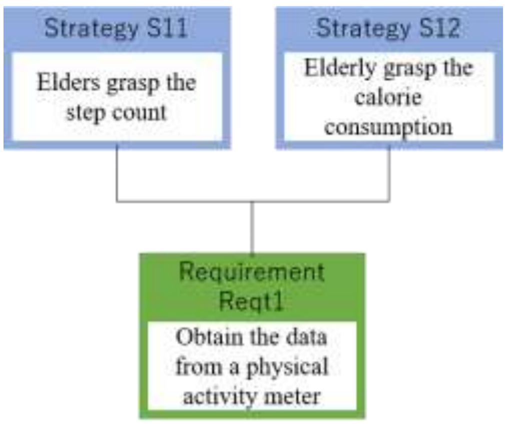


Figure 3.8: Example of induction

(6) Deduction

Deduction, which is based on DSR [9][10], is used when a single requirement is derived from multiple strategies, contexts, and assumptions that have parent-child relationships among themselves. For example, Figure 3.9 shows the strategy “grasp health conditions” with the background assumption of “motivation for health will increase by understanding health conditions and history of physical activities”, which derives the system requirement “graph display”.

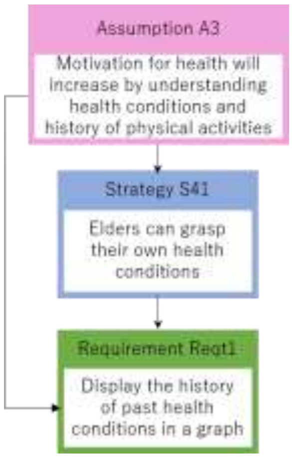


Figure 3.9: Example of deduction

3.3 Model deployment by linkage

The model deployment for each linkage type is described when addition, removal, or other changes are implemented in strategies or requirements.

(1) Aggregation

An aggregation relationship deals with the addition of a single requirement between a strategy and the requirements. Figure 3.10 depicts the addition of a single requirement to Figure 3.4 with an aggregation relationship, where the additional requirement is linked to a strategy in a similarity relationship.

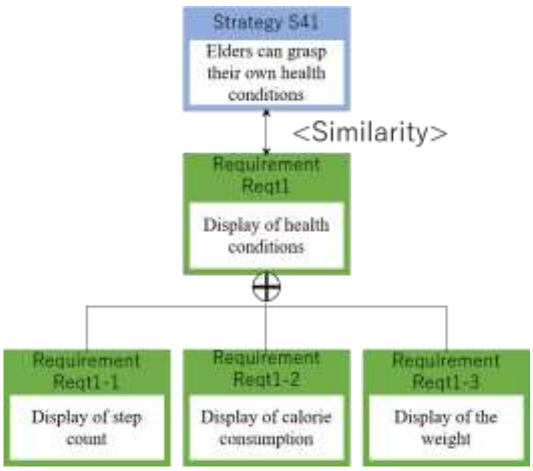


Figure 3.10: Model deployment diagram of aggregation

(2) Generalization

When the position of a linked strategy changes and a child strategy is added in generalization, the requirements for the strategies must be considered. If a child strategy is removed, the linked requirements must also be removed.

(3) Similarity

When a strategy is added, the requirements must be changed, and the relationship is induction. Figure 3.11 shows the impact of changing the similarity relationship in Figure 3.6 to an induction relationship. Relationship aggregation must be verified when adding a requirement. If aggregation does not occur, the content of a strategy must be changed or added for each added requirement and linked as a similarity relationship. If it is an aggregation relationship, the aggregation approach is used. For example, Figure 3.6 becomes Figure 3.12. If one of the strategies or requirements must be changed (or removed), the other content must also be changed (or removed).

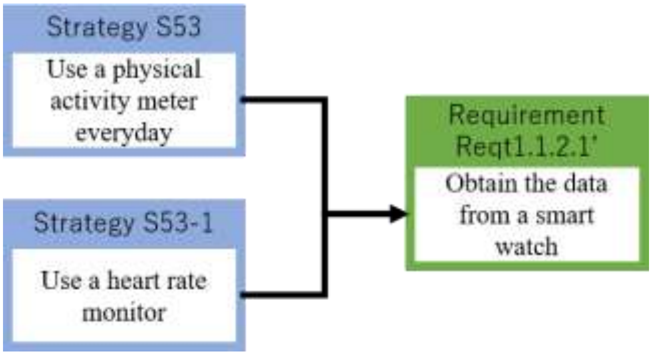


Figure 3.11: Model deployment diagram of a similarity relationship when a strategy is added

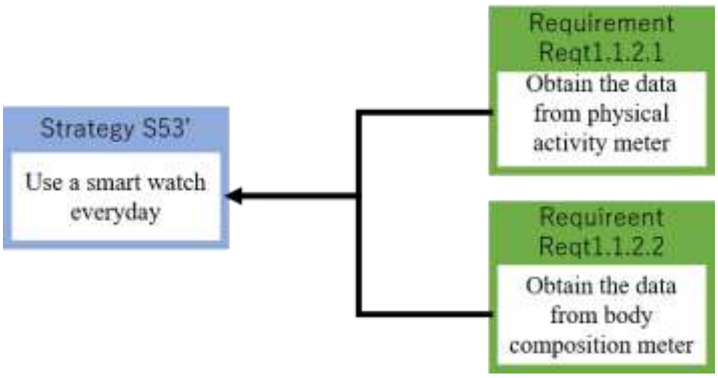


Figure 3.12: Model deployment diagram of a similarity relationship when a requirement is added

(4) Dependency

When a strategy is added, it changes from a dependency relationship to an induction relationship. In addition, the requirement must be changed. When a requirement is added, relationship aggregation must be verified. If aggregation does not occur, the content of the strategy must be changed or added for each linked requirement. Additionally, the requirement must be linked as a dependency relationship. If the content of one of the strategies or requirements needs to be changed (or removed), the other content must also be changed (not removed).

(5) Induction

When a strategy is added, the content of the requirement must be changed. When a requirement is added, whether the strategy and requirement can be changed to either a similarity or dependency relationship must be verified. If the content of one strategy or requirement is changed, the other one must also be changed. If a strategy is removed and two or more strategies remain for a requirement, the content of the requirement must be changed. If only one strategy is linked to a requirement, the relationship must be changed to similarity or dependency and the content changed. If a requirement is removed, the linked strategies must be revised, and a requirement and a linkage added.

(6) Deduction

A parent strategy, context, and assumption in GQM+Strategies must be added as a rationale to link to a parent of a requirement in the SysML-requirement-diagram. For example, Figure 3.9 can be expressed as Figure 3.13. When the parent strategy, context, and assumption must be changed, whether the content of the child's strategy and requirement or linkage need to be changed must be verified. If a child strategy is added, the relationship should be switched to induction. When the content of a strategy is changed, the content of a linked requirement must be changed. Similarly, when a strategy is removed, the linked requirement must also be removed. If a requirement is added, whether it is an aggregation relationship must be verified. If the relationship is aggregation, the linkage must be changed. If not, each requirement must be revised such as changing the content of a linked strategy or adding a strategy.

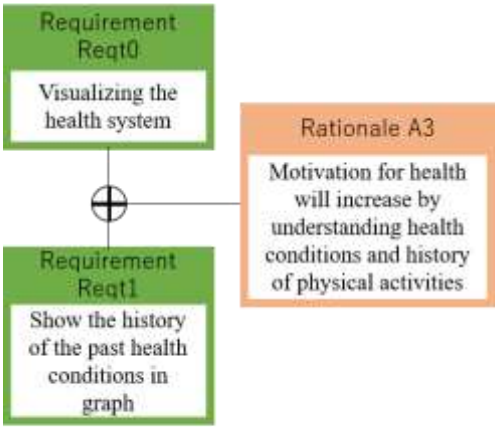


Figure 3.13: Model deployment diagram of a deduction

4 Evaluation

To verify the effectiveness of the metamodel of GQM+Strategies and SysML requirement diagram and the model deployment by each linkage type, we conducted a case study to address the two RQs. Section 4.1 shows the results for a nursing-care robot system. Section 4.2 discusses the results with respect to RQs.

- RQ1 (Validity): Can this model trace the relationship from business goals and strategies to the system requirements?
- RQ2 (Effectiveness): Can our method solve the gap between customers' needs and the developed system, while dealing with changing strategies and requirements?

4.1 Case Verification

4.1.1 Overview

We conducted a case verification of the relationship diagram of the elements based on an actual case involving a nursing-care robot system. In this verification, a plan to introduce an IT system was devised in a nursing-care business to achieve the goal of "resolving the shortage of workers in a nursing-care business". Part of the project plan was developed into a GQM+Strategies grid, and a SysML requirement diagram was generated for the system configuration. Then the models were linked by our method. Experts evaluated the models before and after linking.

4.1.2 Results

The alignment diagram using GQM+Strategies and SysML describes the business strategies and system requirements. Figure 4.1 shows part of the business goals and strategies using GQM+Strategies, while Figure 4.2 shows part of the system requirement diagram using SysML. For the diagrams, the linkage types and model deployment methods described in Chapter 3 are applied. Figure 4.3 is the alignment diagram using the proposed method. The expert evaluation reveals that the proposed method considers the necessary system requirements based on business goals.

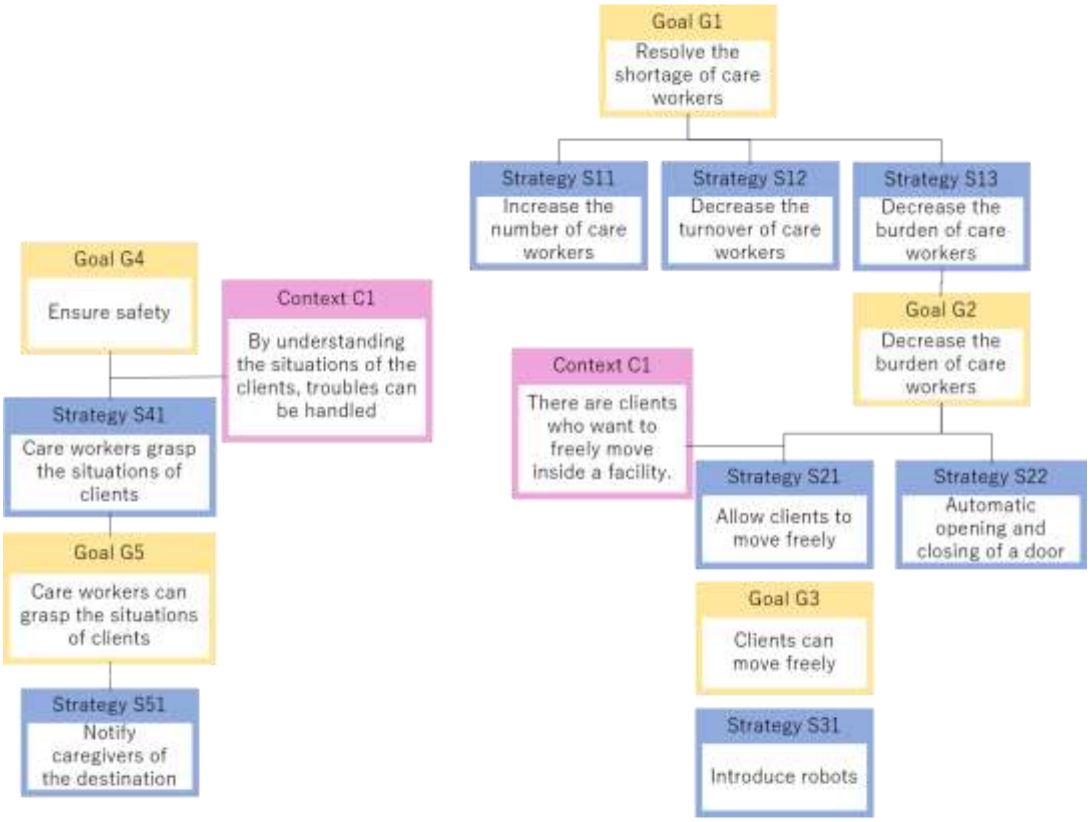


Figure 4.1: Part of GQM+Strategies grid before linkage

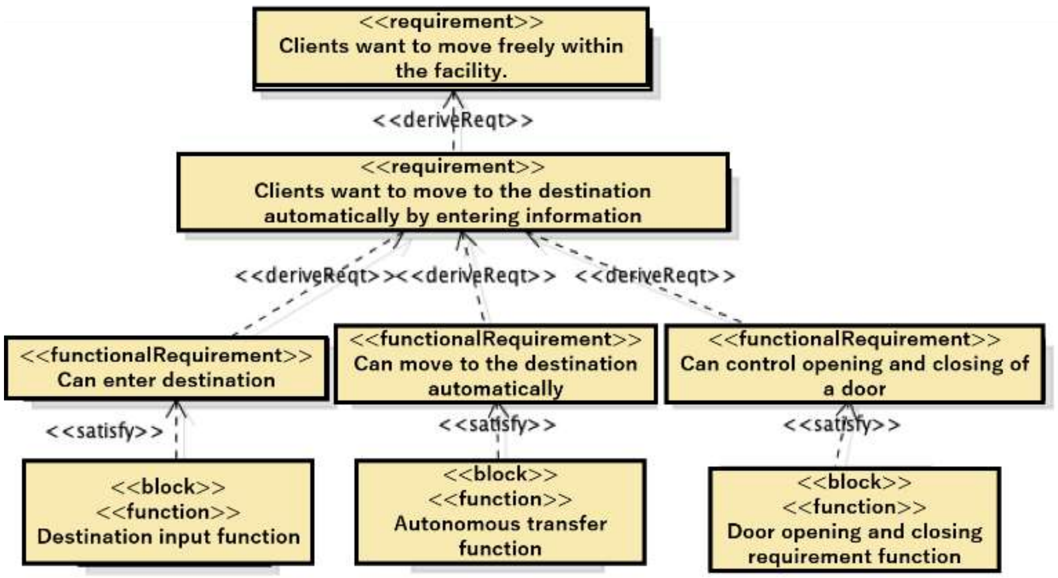


Figure 4.2: Part of SysML requirement diagram before linkage

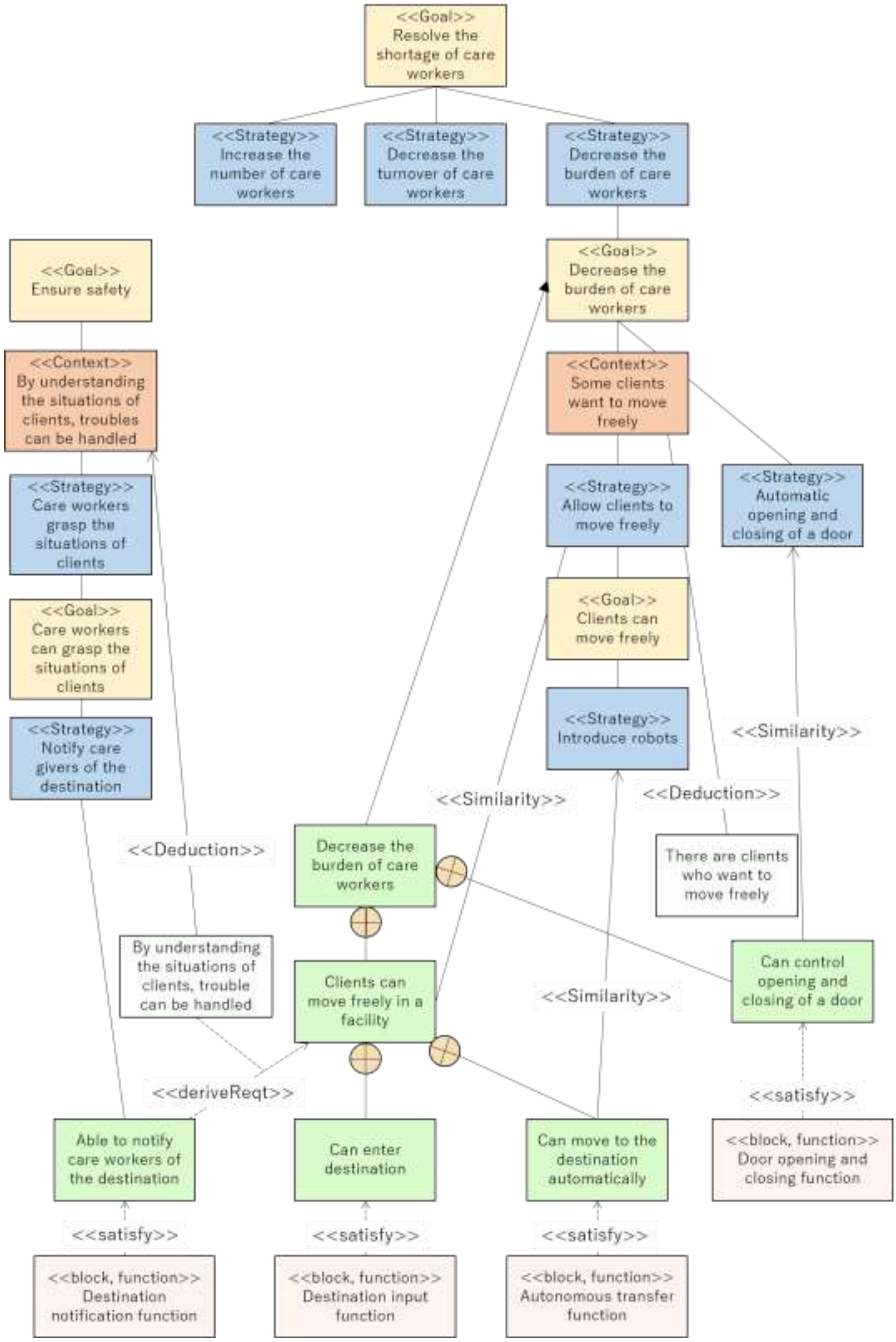


Figure 4.3: Alignment diagram after linkage

4.2 Discussion

We defined the linkage between two models: GQM+Strategies and a SysML requirement diagram. We also proposed a model deployment method for each linkage type, conducted a case verification, and confirmed the effectiveness of the method by expert evaluations.

4.2.1 RQ1

In this section, we discuss RQ1, "Can this model trace the relationship from business goals and strategies to the system requirements?" The background of the derived system requirements can be understood by aligning elements of business goals, strategies, and system requirements (Figure 4.3). Hence, business goals and strategies can be traced to the system requirements.

However, not all business strategies are reflected as system requirements. For example, requirements are unnecessary in the system development for strategies related to the organization and people but not an IT system. Moreover, non-functional requirements such as usability are not focused business strategies. Hence, new strategies do not need to be created. Thus, the actual content must be evaluated to determine which business strategies should be linked to which system.

4.2.2 RQ2

In this section, we discuss RQ2, "Can our method solve the gap between customers' needs and developed system, while dealing with changing strategies and requirements?" By applying our method to this case, hidden goals, strategies, and requirements are defined explicitly. For example, the strategy, "automatic opening and closing of doors" was not defined before applying our method. It was added because we are able to consider whether the strategies are appropriate from the perspective of system development. In addition, the requirement, "Clients can move to the destination automatically" is also defined explicitly. It corresponds to the strategy, "introduce robots" and the rationale, "some clients want to move freely".

The experts in the case verification stated that the business-goal oriented consideration of system requirements is realized. That is, the gap shown in Figure 2.3 is reduced. Hence, linking these two models can generate system requirements aligned with business goals and strategies. Furthermore, the addition of business strategies is dealt with in reference to the diagram. Therefore, our method can help solve the gap problem and deal with changes.

4.3 Threats to Validity

The following are identified as threats to the internal validity. This verification only applied the model to add business strategies. Strategies were not modified or removed. In addition, system requirement additions, changes, or removals were not performed. The linkage types were mostly the similarity relationship, and some other linkage types were not involved. These may affect the validity. In the future, the effects of these factors should be confirmed by increasing the verification case studies.

With regard to threats to external validity, the following is noted. The case verification was limited to a single business and a single domain. Because the proposed method is not designed for a particular domain, this method should be similarly effective in other domains where business and systems have a different relationship. We plan to apply the proposed method to other domains and businesses to verify its effectiveness.

5 Related Work

5.1 Metamodel of Correspondences

As shown in Section 2.3 and Figure 2.1, M.E. Hamlaoui et al. proposed MMC [9][10]. Model linkage types in our method are based on MMC. In particular, the relationships of aggregation, generalization, similarity, and dependency are based on DIR of MMC, while those of induction and deduction are based on DSR. Their research used a business process-specific model as a business-level model and system design model. However, in this study, we use a business-level model specialized in business goals and strategies.

5.2 Integration of GQM+Strategies and SysML

T. Takai et al. proposed an approach that integrates the GQM+Strategies method and the modeling language SysML for system engineering [16]. By translating the elements of the GQM+Strategies grid to SysML and the SysML requirement diagram to GQM+Strategies systematically, GQM+Strategies and SysML can be aligned. As a result, the gap of misunderstanding between business analysts and system engineers can be reduced, and well aligned models can aid in decision-making because effective analysis of system are possible. Our method aligns GQM+Strategies and SysML to confirm business strategies and system requirements are corresponding properly. However, their method aligns them to find solutions to satisfy the needs of stakeholders because there are wide variations in solution space to satisfy the given needs in an IoT system.

5.3 Alignment of Software Requirements with Business Objectives

Various research has analyzed the value of aligning business and IT development [17]. X. Cui et al. proposed a framework that integrates the development of motivation and requirements models at the organization, business, product, and system/software levels [18]. Their model is constructed based on the OMG standardized Business Motivation Model (BMM) and Systems Modeling Language (SysML). This framework contains the motivations shown by Motivation Models and requirements shown by Requirements Models from the top-level organization motivation to the low-level product requirements. Their method uses BMM for business analysis, which can express business elements from various aspects by Vision, Mission, Goal, etc. On the other hand, our method uses GQM+Strategies for business analysis, which allows hierarchical and recursive descriptions, and expresses comprehensive business parts across multiple business departments.

5.4 Agile Methodology MADAIKE

N. B. Wilson et al. proposed an agile methodology called MADAIKE. MADAIKE promotes the integration of various standards and justifies the value to the organization [19]. It is an integrated methodology of layers where each layer represents a project phase. Each phase defines the development process and the role of developers. Their methodology aims to integrate various standards and has a broad perspective across phases. On the other hand, our method aims for traceability and consistency of business goals, strategies, and system requirements.

5.5 ArchiMate

ArchiMate [20] is a model that expresses both business-level and system-level models consistently. ArchiMate, which is a standardized enterprise architectural modeling language from The Open Group, has six layers (strategy, business, application, technology, physical, and implementation & migration) and four elements (passive structure, behavior, active structure, and motivation) (Figure 5.1). GQM+Strategies is specialized for the business strategy layer. In addition, goal, question, and measurement (GQM) related to business strategies are included. Consequently, business strategy measurement analysis is possible using GQM.

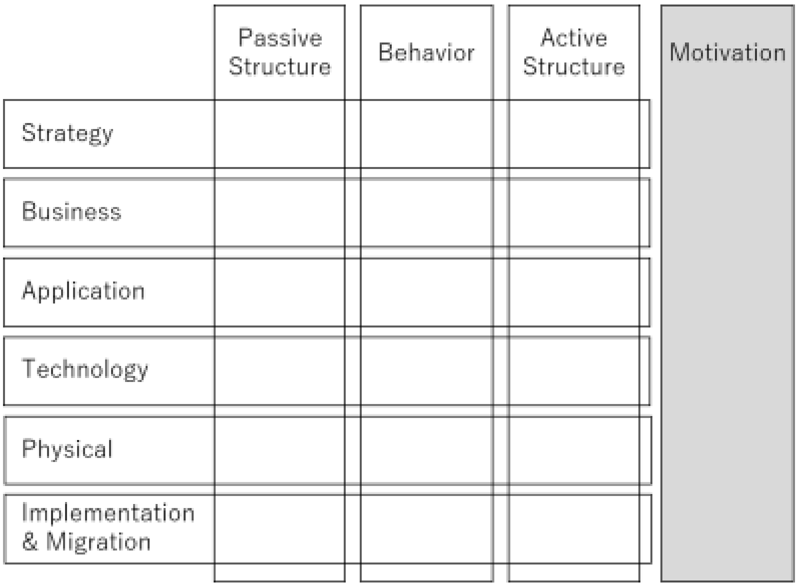


Figure 5.1: ArchiMate framework

6 Conclusion

As system product developments evolve, the practice of creating products solely in-house must be replaced by protocols where a business integrates its products with those of other businesses. The demand for system product development processes is diversifying and becoming more complicated. Additionally, the increased prevalence of agile development makes it difficult to adjust a system to changing goals and requirements. To solve these issues, we proposed a method to consistently manage business goals, business strategies, and system requirements by aligning elements of GQM+Strategies and SysML. Then we elucidated model deployment about the linkage types and conducted a case study to verify its effectiveness. The proposed method not only helps maintain the traceability and consistency from business requirements to system functions but also solves the gap between customers' needs and the developed system while simultaneously dealing with changing strategies and requirements under system development.

Acknowledgement

This research is partially supported by enPiT-Pro Smart SE: Smart Systems and Services innovative professional Education program, and JST-Mirai Program Grant Numbers JP18077318 and JP20319852.

References

[1] N. Mimura, H. Washizaki, K. Shintani, S. Okuda, “Linking business strategies and system demands using GQM+Strategies and Systems Modeling Language,” 4th International Conference on Enterprise Architecture and Information Systems(EAIS), Japan, 2019.

[2] A. Trendowicz, J. Heidrich, K. Shintani, “Aligning Software Projects with Business Objectives,” 21st International Workshop on Software Measurement and the 6th International Conference on Software Process and Product Measurement, Japan, 2011.

[3] V. Basili, et al., “Aligning Organizations Through Measurement,” Springer International Publishing, 2014, p.36- 44.

[4] S. Friedenthal, A. Moore, R. Steiner, "OMG systems modeling language (OMG SysMLTM) tutorial," INCOSE international symposium, Vol.18, No.1, 2008.

[5] I. Inayat, S. Salim, S. Marczak, M. Daneva, S. Shamshirband, “A systematic literature review on agile requirements engineering practices and challenges,” Computers in Human Behavior, Vol. 51, No.B, October 2015, pp.915-929.

[6] V. Basili, “Software modeling and measurement: the Goal / Question / Metric paradigm,” Maryland Univ, USA, September 1992.

[7] V. Basili, et al., “GQM+Strategies in a Nutshell,” in The Fraunhofer IESE Series on Software and Systems Engineering, Springer, March 2014, pp. 36-44.

[8] L. Balmelli, “An Overview of the Systems Modeling Language for Products and Systems Development,” Journal of Object Technology, vol. 6, no. 6, 2007, pp. 149–177.

[9] M.E. Hamlaoui, et al., “Alignment of viewpoint heterogeneous design models: Emergency Department Case Study,” in 4th International Workshop On the Globalization of Modeling Languages (GEMOC 2016) co-located with ACM/IEEE MODELS 2016, Oct 2016, pp. 18-27.

[10] M.E. Hamlaoui, S. Bennani, M. Nassar, S. Ebersold, B. Coulette, “A MDE Approach for Heterogeneous Models Consistency,” ENASE2018, MARCH 2018.

[11] D.M. Fernández, et al., “Naming the Pain in Requirements Engineering: Contemporary Problems, Causes, and Effects in Practice,” Empirical Software Engineering, Nov 2016.

[12] J. Gardan, N. Matta, "Enhancing Knowledge Management into Systems Engineering through New Models in SysML," Procedia CIRP, Volume 60, 2017, pp. 169-174.

[13] N.B. Moe, A. Aurum, T. Dyba, "Challenges of shared decision-making: A multiple case study of agile software development," Information and Software Technology, Vol. 54, No.8, Aug 2012, pp.853-865.

[14] C. Shimura, H. Washizaki, Y. Aoki, T. Kobori, K. Honda, Y. Fukazawa, K. Shintani, T. Nonomura, "Identifying Potential Problems and Risks in GQM+Strategies Models Using Metamodel and Design Principles," Proceedings of 50th The Hawaii International Conference on System Sciences (HICSS 2017), January 2017, pp. 4857-4866.

[15] Kashiwa City Health Center Community Health Promotion Section (Kashiwa-shi hokenjo chiiki-kenkouzukuri-ka) "Kashiwa City Health Promotion Plan (Kashiwa-shi kenko-zoshin keikaku)" Kashiwa City, April, 2013.

[16] T. Takai, K. Shintani, H. Andoh, H. Washizaki, "Case study applying GQM+Strategies with SysML for IoT application system development," EAIS 2019.

[17] 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.

[18] 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.

[19] 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.

[20] The Open Group, "Open Group Standard ArchiMate® 3.0.1 Specification" August 2017.