Journal of Health Informatics in Developing Countries

www.jhidc.org
Vol. 9 No. 1, 2015

Submitted: September 1, 2014 *Accepted: January 7, 2015*

Health Information Systems' Architectural Evaluation with Architecture Trade-off Analysis Method (ATAM): a case study in Nigeria

Ishaya GAMBO[a,a], Abimbola SORIYAN[a] and Rhoda IKONO[a]
[a] *Obafemi Awolowo University, Computer Science and Engineering Department, Ile-Ife, Nigeria*

Abstract. This paper identified what should be architectural quality attributes for health information system (HIS) architectural evaluation. It raises awareness of the importance of stakeholders in the architecture process. This was with a view to ensuring that HIS follow a specific architecture for the quality to be determined exactly at all times. ATAM was used to validate the analysis using the architecture of the Made in Nigeria Primary Healthcare Information System (MINPHIS). The paper demonstrates the applicability and usefulness of architectural evaluation in HIS context. The analysis with ATAM showed the generation of quality attributes utility tree, priorities, the scenarios, sensitivity points, risks involved and its benefits towards healthcare system. Evaluating the system showed that the database system needs to be migrated from hierarchical to relational structure. By implication, the evaluation is useful for quality improvement in the earlier stages of HIS development, which supposes an indirect impact on patient care.

Keywords. ATAM, MINPHIS, Software architecture, Health information system.

1. Introduction

Health information system (HIS) is a service-oriented system that comprises multiple types of organisation. An inside view of this reveals a multitude of different types of actors, information, and information systems that are highly regulated and governed by legislation. In [1], the authors observed that in the last two decades HIS has undergone tremendous progress in terms of development. For the Nigerian healthcare system, HIS has gone through a well-defined manual structure and is being provided by collaborative teams that involve multiple healthcare providers at different levels in terms of delivery. In the work of [2], three tiers of healthcare delivery in terms of system implementation were identified to include: the federal, state and the local government. The development thus far has always been through a process that includes activities for analysis, design, implementation, evaluation, and sustainability supports. The design aspect

[a] Ishaya Gambo, Computer Science & Engineering Department, OAU, Nigeria. ipgambo@gmail.com, ipgambo@oauife.edu.ng.

serves as the blue-print for the entire system implementation that needs thorough evaluation for quality improvement.

Going further, quite a lot of the hospital information systems in Nigeria are robust, non-commercial and developed by the local universities and corresponding teaching hospitals. Specifically, the Made in Nigeria Primary Healthcare Information System (MINPHIS) is one of the appropriate software products for African hospitals and health centres developed with the peculiar nature of Nigeria and Africa in mind. This system is the first in Nigeria that is locally developed since 1989. What necessitate the development of MINPHIS was as a result of the information management needs of a hospital accounting, and medical records that further engenders a joint project with Nigerian and Finnish software developers [3]. The objective was to develop a low-cost application with sustainable, endogenous capabilities. The public-domain software developed by the United State Department of Veterans Affairs (VA), and its' Admission-Discharge-Transfer package was used as the basis and requirements of local hospitals [3. 4]. These requirements are programmed in to suit the information management practices [3, 4, 5, 6].

Consequently, the technical architecture of the first version was originally based on MUMPS language, FileMan-database, and Kernel-package [5, 7]. Since then the software architecture and database structure has been modernised several times but architectural evaluation has not been carried since inception.

Technically, the accuracy and suitability of an architectural design in software engineering practice requires thorough evaluation while considering several quality attributes. [8, 9] proposed the following quality attributes suitable for architectural evaluation: Usability, Performance, Reliability, Availability, Security, Functionality, Modifiability, Portability, Variability, Testability, Conceptual Integrity, Building simplicity, Cost and Time to market. This was supported by the work in [10] where several quality attributes were justified according to the SEI-ATAM evaluations. As such, software architecture main study area with respect to quality attributes is on how programs or application components are internally built. The architecture is the foundation for predicting information system quality attributes.

For the healthcare system, an important goal of architectural evaluation is to analyse and ensure the architectural potential for implementing any healthcare system capable of supporting major business requirements. In addition, the identification of major risks and trade-offs is a major concern during architectural evaluation. As observed in [11], "the main aim of an evaluation is to analyse the architecture in order to identify potential risks and to verify that the quality requirements have been addressed in the design". As opined also in [12], the aim of analysing the architecture of a software system is to predict the quality of a system before it has been built and not to establish precise estimates, but the principal effects of the architecture. Therefore, quality issues should be the fundamental focal target in the development of HIS. This is because most stakeholders using the system are particular about quality in terms of functionality, ease of use, security, portability, efficiency, effectiveness and satisfaction etc.

In view of the above, this paper acknowledges that quality attributes of HIS products can be highly constrained by the architecture. Consequently, being able to evaluate the quality of software product is very important, not only from the perspective of a software engineer, but also from a business point of view for information system in order to determine the level of the provided quality [12]. Some of the analysis and evaluation techniques used to achieve this includes: the SAAM (Software Architecture Analysis Method) [14], the QAW (Quality Attribute Workshop) [15], the ADD (Attribute Driven Design) method [9], ATAM (Architecture Trade-off Analysis Method) [16] and the CBAM (Cost Benefit Analysis Method) [17] among others.

This paper reports the outcome of the architectural evaluation of an healthcare system called MINPHIS with ATAM. The architecture of this system was reported in [18, 13]. The author in [18] applied the ATAM method to help detect real flaws in a design and identify potential risks. ATAM as a whole is organized around the idea that architectural styles are the main determinants of architectural quality attributes [16]. ATAM has 4 main phases, each of which consists of several intermediate steps. Totally, the method contains nine steps [15, 20]. In this paper, the evaluation with ATAM considered the Software Architecture Scenario-Based Performance Quality Model (SASPUM) in [21]. ATAM is considered a mature approach, as it has been validated in different domains [13]. The purpose of ATAM to MINPHIS is to assess the consequences of MINPHIS architectural decision alternatives in light of the performance quality attributes.

2. Methods

Extensive review of current and related literatures on architectural evaluation was carried out. Relevant literatures supporting architectural evaluation of information systems were harvested from Journals and conference proceedings. Software quality models were studied with the aim of identifying their weaknesses. Various architectural evaluation methods were studied. In addition, the different architectural styles, views and design pattern and tactics were studied and analysed with the aim of providing the needed software architectural quality for evaluation. In addition, a systematic literature review was conducted to consider the relationship between architectural design and healthcare systems' qualities; requirements of HIS and its corresponding architectures. These requirements have been reported as the major business goals in the first phase of the ATAM evaluation process.

The various architectural views, tactics and design patterns studied prompted the adoption of the Krutchen‟s 4+1 Views model with the different views to provide the basis for reasoning about the appropriateness and quality of the architecture in achieving the system quality goals on the developed quality model in [21]. These views, tactics and design patterns were broken down into their structural parts with the aim of providing the needed architectural quality evaluation [13].

By using qualitative research method, the scenario-based approach to analyse and evaluate software architecture was used after a thorough review of various analysis and evaluation methods. In this sense, the Architecture Trade-off Analysis Method (ATAM) was chosen and used because it provides insight into the way quality attributes are mapped onto architecture and also shows the trade-offs existing between the identified quality and others. Table 1 shows the methodological steps of ATAM and corresponding activities that were followed.

Going further, 3 teaching hospitals where MINPHIS was deployed and in use were contacted. In particular, the Obafemi Awolowo University Teaching Hospital Complex (OAUTHC) (i.e. one of the teaching hospitals where MINPHIS is currently running) was visited to make investigation into how the system functions and behave during execution. In addition, 4 users of MINPHIS from two other teaching hospital: Ahmadu Bello University Teaching Hospital (ABUTH) and Ladoke Akintola University Teaching Hospital (LAUTECH) each were contacted through phone calls and email due to distance to make the investigation about the system functionality. Questions related to the way the system behaves and function at run time were texted and emailed to these identified users. The investigation at the OAUTHC was carried out using interview and direct observation of the system at run time. From the behavior of the system and how it functions as elicitated and discovered from the three (3) teaching hospitals, scenarios were generated. Other scenarios were generated by other stakeholders (like the system administrator, system analyst, programmer, requirement engineer, and medical record officers) based on the business requirements of the system. All these were carried out to support the evaluation processes for quality improvement.

Table 1. Methodological steps of ATAM and corresponding activities

ATAM Phases	Steps	Activities	Descriptions
Phase 1: Presentation	1	Present the ATAM.	ATAM was explained to those involved in evaluation for the purpose of understanding.
	2	Present the business drivers	An overview of the system (MINPHIS), it requirements, business goals, and context, and the architectural quality attribute drivers were presented.
	3	Present the architecture	MINPHIS architectural design was presented.
Phase 2: Investigation and Analysis	4	Catalog the architectural approaches	MINPHIS Architectural Approaches were identified. Here, the evaluation process captures a list and adds to it any approaches observed during Step 3 or learned during the pre-exercise review of the architecture documentation. The approaches will help carry out the analysis steps.
	5	Generate a quality attribute utility tree	Participants in the evaluation process will build a utility tree, have a prioritized set of detailed statements about which quality attributes are most important for the architecture to carry out alongside the specific scenarios that express those attributes.
	6	Analyze the architectural approaches	The utility tree scenarios were map to the architecture to see how it responds to each important scenario. This was carried out with the MINPHIS architects.
Phase 3: Testing	7	Brainstorm and prioritize the scenarios	The stakeholders brainstorm additional scenarios that express specific quality concerns.
	8	Analyze the architectural approaches.	As in Step 6, the high-priority brainstormed scenarios were mapped to the architecture.
Phase 4: Reporting	9	Present the results	A presentation and final report was produced that capture the results of the process and summarize the key findings.

3. Results

The results of the evaluation are as presented in figure 1 to 8.

Steps in Brief	Techniques	Expected Results
1	Utility tree generation	Utility Tree
2	Architecture elicitation and analysis	Architectural Approaches
3	Scenario brainstorming/mapping	Scenarios
		Risk & Non-Risk, Sensitivity points and Tradeoffs

Figure 1: Step 1 of ATAM Method as presented and explained.

Figure 1 indicates the various 2 steps that were presented alongside the techniques to be used, and what results should be expected. The stakeholders present during first phase meeting were introduced to ATAM as a method. After clarification of questions asked, consent to go ahead with the evaluation process was received and endorsed.

General Focus	Business Goals	Business Requirements	Business Constraints (performance Related)
Information Backbone for Healthcare Institutions	Use for Keeping patient records	-	Transaction response time
	Use for Answering ad hoc queries from medical Researchers	-	System throughput
	Use to Provide performance information	-	-
	Use for Resource Management Decisions	An improved web-based system with web services	-
	Produce on-demand & Periodic reports (based on specific needs of Inst. Or Hospital)	-	Response time & System throughput

Figure 2: Step 2: Business perspective of MINPHIS as presented

The various business goals were captured as synthesised in figure 2 alongside the constraints relating to the performance of the system. Step 3 entails the presentation of MINPHIS Architecture as presented in Module View [21].

Approaches		Description	Architecture Layer
Layering	I	FileMan (a data-centric architectural pattern) and M Software	Database Layer
	II	RPC Broker Server	
	III	Ready Made Components	
	IV	Actual Application	
API		MINPHIS API comprise the M Procedures defined in RP file	Database Layer
Client Server Architecture		Client-Server transaction processing	Database Layer, RPC Broker Server Layer

Figure 3: Step 4 showing MINPHIS Architectural Approaches as identified

Stakeholder	Scenario	Identified Quality attributes Concern
User	No unauthorized access to the system	Security
	User errors in using system should be handled	Reliability
	All operations are processed in fastest possible speed	Performance
Architect	Proper data encapsulation and secure data structures	Security
	Flexible implementation using other programming languages	Portability
	Overall consistent behavior expected from the architecture	Conceptual Integrity
Application Developer	Framework should be complete, clear and perform exactly as required.	Functionality

Figure 4: Step 5 showing MINPHIS Scenario generated based on the different views of Stakeholders and the identified, prioritized, and refined performance quality attribute.

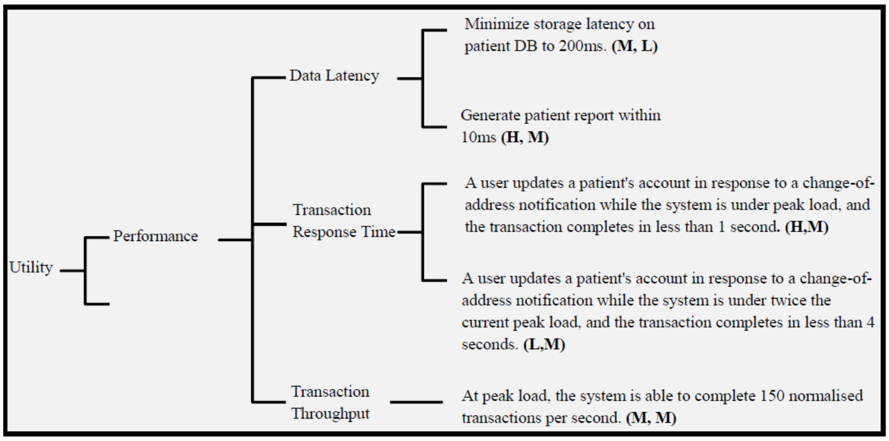

Figure 5: Performance utility tree for MINPHIS architectural evaluation

High priority scenarios extracted	Performance QA Refinement	Scenario	Outcome
(H,M)	Data Latency	Generate patient report within 10 ms	Performance is important to the system
(H,M)	Transaction Response Time	A user updates a patient's account in response to a change record while the system is under peak load, and the transaction complete in less than 0.75 second	Performance is important to the system

Figure 6: Step 6 showing analysed output of the utility tree generated and discovered architectural approaches that deals with performance quality attribute.

S/N	Scenario	Priority
1	A hospital needs to centralize the record maintenance process across multiple affiliates; associated business process is re-engineered.	(H,M)
2	A manager wants a report on historical payment delinquency rates for people who were treated for some specified diseases.	(M,M)
3	A defect corrupts data and is not detected until the next reporting cycle.	(H,H)
4	MINPHIS is installed in a hospital, and the hospital's existing database must be converted.	(H,M)
5	Introduce a new work flow process for patient check-in and check-out.	(M,L)
6	A report needs to be generated using information from two hospitals that use different configurations.	(H,M)
7	Change the rules for generating a warning about conflicting medications.	(M,L)

Figure 7: Step 7 showing additional scenarios examined and prioritized with stakeholders

Risk	Non Risk	Sensitivity Point	Tradeoffs Points
The decision to keep backup is a risk if the performance cost is excessive	The decision to keep backup is a non-risk if the performance cost is not excessive	The average speed at which the task is performed is sensitive to the number of components involved in processing the task	Keeping the backup database affects performance also so it is a trade-off between reliability and performance

Figure 8: Step 8 showing the determined risk, non-risk, sensitivity and trade-off points

4. Discussion

It was discovered that addressing performance quality attribute among others is crucial for the system (MINPHIS), and this justifies the fact that it is the most common quality attributes according to the SEI-ATAM evaluations [10]. Going further, a module view of MINPHIS architecture was shown, the refinement process for performance quality characteristics was shown as reported in [21], and the performance tactics were generated.

After the presentation of ATAM in step 1, it was discovered that keeping patient records and providing performance information relevant to particular health care professionals, such as the mortality rates for patients treated by a particular staff member, as well as number of patients attended to by particular medical staff among others are the business drivers. These business drivers provided a clear understanding and expectation of MINPHIS.

In step 3, several views of the architecture and the architectural approaches emerged. After identifying the architectural approaches specific to MINPHIS, it becomes possible to conceptual the system, which in turn allows for asking probing questions at the scenario analysis stage. As such, scenarios were classified according to their priority on importance and difficulty. These scenarios are generated for the stakeholders who include: the end users (e.g. Doctors, Nurses, Pharmacist, Medical Record Officers, Radiologist, Lab Technicians, Management staff etc.), the architect and the application developers. The scenarios are annotated with the priority rankings assigned by the decision makers of the system present during the evaluation process. The ranking is done in order of pairs indicating the importance of the capability and the architect's estimation of the difficulty in achieving it. This was captured in the utility tree that contains the 'utility' as the root node, with performance quality attribute forming the secondary level of the utility tree as shown in figure 5. The prioritization in the utility tree was based on relative rankings: High (H), Medium (M) and Low (L).

Performance is broken down into "data latency", "transaction throughput" and "transaction response time". This is a step towards refining the attribute goals to be concrete enough for prioritization. Latency and throughput are two of the types of response measures noted in the attribute characterization as described in

[9]. The analyzed architectural approaches were related to the scenarios and their ranking. Hence, the utility tree exercise actually produced no scenario ranked (H, H), which indicates high-importance, high-difficulty scenarios that merit high analytical priority. The (H,M) scenarios was targeted, a cluster of which appeared under "Data Latency" and "Transaction Response Time" hypothesizing the generation of patient report within 10ms (H, M) and updating a patient's account in response to a change record while the system is under peak load, and the transaction complete in less than 1 second. (H, M). From the prioritization of these scenarios, shown by the (M, L), (H, M), (H, M), (L, M) and (M, M) beside the scenarios, it was decided that the architectural quality attribute - performance is important to the system.

The contributed scenarios from other stakeholders are the ones at the leaves of the utility tree, but were not actually analyzed. At this point, the stakeholders expressed their views on the fact that some scenarios deserved more attention. As indicated in the results, the scenario produces a collection of sensitivity and trade-off points (figure 8). The sensitivity point is an architectural decision that affects the achievement of a particular quality, while the trade-off point is an architectural decision that affects more than one quality attributes [19].

Secondly, a collection of risks, which is an architectural decision that is problematic in light of the quality attributes that it affects; and non-risks that defines an architectural decision that is appropriate in the context of the quality attributes that it affects were discovered. Obviously from the evaluation process, evaluating the system showed that the database system need to be migrated from hierarchical to relational structure. This convinced the evaluator that a well-thought-out procedure was in place, with known strengths and reasonable limitations.

4.1. Importance of Architectural Evaluation to Health Informatics Community

Healthcare system development in developing countries is yet to harness all the possible potentials inherent in various development models of a typical software engineering process. During and after the architectural design, it is necessary to analyse the structures and various components that defines the architecture for quality assurance. The main point for such evaluation is to predict the quality of the system considering it's critically as an intensive-life care system for patient care and safety.

For the developing countries, healthcare system architectural evaluation is expected to facilitate continuous quality of system in providing quality care in the hospitals and other healthcare facilities. For the Health Informatics Community (HIC), HIS architectural evaluation can be beneficial in the area of quality improvement. Specifically, the evaluation is a medium through which HIC identifies insufficiencies in the early design phases. By this, the HIC will be able to know whether the architecture does not meet their expectation regarding one or more quality attributes. Secondly, HIS architectural evaluation enables HIC to compare between alternative architectural suggestions regarding one or more quality attributes in the care process. Another importance is the platform to make investigation whether an architecture holds risks for certain quality attributes of the system considering HIS as critical life system. At this point, it will be possible to know if the system has helped to improve the quality of patient data, which in turn, has been used through reporting to improve the quality of decision-making.

5. Conclusion

Conclusively, quality control and management must be carried out through the whole development process of any HIS to ensure the implementation of required quality characteristics. The evaluation explanations emphasize the SEI's advice that software architecture evaluations be performed before code is developed so that surfaced risks can be mitigated when it is least costly to do so [19]. This also supports the opinion of [22] in adapting 'open system architecture' for the healthcare system in order to benefit in terms of cost, efficiency, effectiveness and other quality attributes during implementation. In the future, it will be necessary to postulate an increased quality of final HIS through quality assurance of the system architecture in earlier stages of development, which will definitely have an impact on patient care.

References

[1] GAMBO, I., OLUWAGBEMI, O., & ACHIMUGU, P. (2011). Lack of Interoperable Health Information Systems in Developing Countries: An Impact Analysis. *Journal of Health Informatics in Developing Countries*, **5**(1).

[2] Ijadunola KT, Bello MO, Korpela M, and Koskivirta M. (1998). A strategy for the use of information systems to support community-oriented primary healthcare in a deprived economy: A case study of Ife Central LGA, Nigeria. In: Richards B, ed. *Current Perspectives in Healthcare Computing 1998, Conference Proceedings*. Weybridge, BJHC Books, pp 3-9.

[3] Makanjuola ROA, Daini OA, Soriyan HA & Korpela M 1991.Low-Cost Hospital Informatics in Africa: The Ile-Ife Experience. MIE 1991 Conference Proceedings. In: Adlassing et al (Eds.) *Medical Informatics Europe 1991 Proceedings*, Springer-Verlag p. 111-115.

[4] Daini OA, Korpela M, Ojo JO & Soriyan HA 1993. The Computer in a Nigerian Teaching Hospital: First-Year Experiences. *HELINA conference 1993. Ile-Ife. Nigeria.*

[5] Korpela M. 1999. Moving a valuable heritage to the network computing architecture. *Proceedings of the 1999 Annual HIMSS Conference, Atlanta, Georgia*. Vol. **3**, pp. 405-419.

[6] Soriyan HA 2004. A Conceptual Framework for information System Development Methodology for Educational and Industrial sectors in Nigeria. Doctoral dissertation. Obafemi Awolowo University. Ile-Ife.

[7] Päkkilä T 2010. Re-engineering of software architectures. A Case-study on a Nigerian Hospital Information System. M. Sc. Thesis. School of Computing. University of Eastern Finland. Kuopio.

[8] Bass, L., Clements, P., and Kazman, R. (1998). *Software Architecture in Practice*. Addison-Wesley: Reading MA.

[9] Clements,P., R. Kazman, and M.Klein, *Evaluating Software Architectures: Methods and Case Studies*, Addison-Wesley Professional, ISBN 02017048, 2002.

[10] Ozkaya, I., Bass, L., Sangwan, R., and Nord, R. 2008. Making Practical Use of Quality Attribute Information. *IEEE Software*, vol **25**, no.2, March/April 2008, pp. 28-31.

[11] Dobrica, L., and Niemela, E., A Survey on Software Architecture Analysis Methods, *IEEE Tansactions on Software Engineering*, vol. **28**, no. 7, 2002.

[12] Kazman, R., Bass, Len., and Klein, M. 2006. The essential component of software architecture design and analysis. *The Journal of Systems and Software* **79** (2006) 1207-1216.

[13] Gambo, I., Soriyan, A., & Achimugu, P. (2012). Software Performance Quality Evaluation of MINPHIS Architecture using ATAM. *International Journal of Computer Applications*, Volume **46**– No.23, (pp 0975 – 8887).

[14] Kazman, R., Abowd, G., Bass, Len., Webb, M. 1994. SAAM: a method for analyzing the properties of software architectures. In: *Proceedings of the 16th International Conference on Software Engineering. Sorrento, Italy, May 16-12, 1994*. IEEE Computer Society, Los Alamitos, CA pp.81-90.

[15] Barbacci, M., Ellison, R., Lattanze, A., Stafford, J., Weinstock, C., and Wood, W. (2003). *Quality Attribute Workshops (QAW)*, third ed. (CMU/SEI-2003-TR-016), third ed. Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA.

[16] Kazman, R., Barbacci, M., Klein, M., Carriere, S., 1999. Experience with performing architecture tradeoff analysis. In: *Proceedings of the 21st International Conference on Software Engineering, May 1999, Los Angeles, CA*, pp. 54-63.

[17] Kazman, R., Asundi, J., Klein, M., 2001. Quantifying the costs and benefits of architectural decisions. In: *Proceedings of the 23rd International Conference on Software Engineering, May 2001, Toronto, Canada*, pp. 297-306.

[18] Korpela, M. 1990. The Ife Project: Report 1989. Health care informatics in the context of a developing country. *University of Koupio*, 1990.

[19] Barbacci, M., Clements, P., Lattanze, A., Northrop, L., & Wood, W. (2003).Using the Architecture Tradeoff Analysis MethodSM (ATAMSM) to evaluate the software architecture for a product line of avionics systems: A case study (No. CMU/SEI-2003-TN-012). CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST.

[20] Kazman, R., Klein, M., and Clements, P. 2000. ATAM: A Method for Architecture Evaluation. Technical Report CMU/SEI-2000-TR-004, Software Engineering Institute, Carnegie Mellon University, Pittsburgh PA.

[21] Gambo, I., Soriyan, A., and Achimugu, P., 2011. Software Architecture Performance Quality Model: Qualitative Approach. *ARPN Journal of Systems and Software*, Vol.**1**. April 2011, pp. 28-33.

[22] Somu, G., & BHASKAR, R. K. (2011). Adapting Information Technology (IT) in healthcare for Quality patient care-Study conducted in a Hospital in South India. *Journal of Health Informatics in Developing Countries*, **5**(2).