

CBAM과 AHP를 이용한 아키텍처 평가

이 지 현[†] · 강 성 원^{**} · 조 진 희^{***} · 김 진 삼^{****}

요 약

CBAM(Cost Benefit Analysis Method)은 아키텍처 전략을 평가할 때 전략으로 인하여 얻게 될 이익(Benefit)뿐만 아니라 비용(Cost), 스케줄, 위험을 함께 고려하여 최적의 아키텍처 전략을 결정하도록 하기 위한 방법이다. 그러나 CBAM은 아키텍처 전략을 평가하는 과정이 이해당사자들의 합의나 투표, 경험적 직관에 의존하고 있어 결정에 대한 명시적인 정량적 증거를 제시하고 있지 못하다. 본 연구에서는 CBAM에서 제시한 이해당사자들의 합의나 투표를 수행하는 과정에 계층분석적 의사 결정 방법인 AHP(Analytic Hierarchy Process)를 도입하여 아키텍처 전략 결정의 정량적 증거를 명시적으로 제시할 뿐만 아니라 주관적 판단 오류의 가능성을 줄이고자 한다.

키워드 : CBAM, AHP, 소프트웨어 아키텍처

Architecture Evaluation Utilizing CBAM and AHP

Jihyun Lee[†] · Sungwon Kang^{**} · Jinhee Cho^{***} · Jinsam Kim^{****}

ABSTRACT

The CBAM (Cost Benefit Analysis Method) provides a method for deciding the best architectural strategy by considering cost, schedule, and risks as well as the benefits of the architectural strategy. But many parts of the CBAM do not present quantitative evidence explicitly on whether it is the best architectural strategy among others because it depends on the stakeholders' consensus, vote, and/or intuition. In this study, we apply the AHP (Analytic Hierarchy Process) to CBAM to provide explicit quantitative evidence and to reduce the possibility of subjective decision making errors that may occur in the CBAM.

Key Words : CBAM, AHP, Software Architecture

1. 서 론

소프트웨어 아키텍처의 설계와 평가는 서로 밀접히 관련된 활동이다. 소프트웨어 아키텍처 설계방법은 비즈니스 목표와 성능, 보안, 변경용이성, 가용성 등의 품질속성을 아키텍처에 반영하도록 하며, 아키텍처 평가방법은 이러한 속성들이 얼마나 잘 아키텍처에 반영되어 있는가를 평가해준다. 아키텍처를 결정할 때 아키텍트(Architect)들이 직면하게 되는 이러한 품질속성들 사이의 트레이드오프(tradeoff)를 식별하고 위험을 추정할 수 있도록 도움을 주는 방법이 ATAM(Architectural Tradeoff Analysis Method)[1]이다. 반면 CBAM(Cost Benefit Architecture Method)[2]은 어떤 아키텍처 전략(Architectural Strategy)이 갖는 이익, 비용, 스케줄, 위험을 함께 고려하도록 하여 아키텍처 전략에 내재되어 있는 경제적 트레이드오프를 계산할 수 있도록 해준다.

그러나 ATAM과 CBAM은 아키텍처 결정 과정이 이해당

사자들의 직관에 매우 의존적이어서 많은 부분이 명시적인 정량적 근거를 제공하고 있지 못하다. 또한 여러 아키텍처 전략을 동시에 비교함으로써 인하여 판단 오류의 가능성도 내재하고 있다.

예를 들면, ATAM의 경우 트레이드오프나 민감점(Sensitivity Point) 결정을 수행하는 명시적인 방법은 제공하고 있지 않다. 이러한 문제를 해결하기 위하여 Al-naeem과 Zhu는 [3, 4]에서 정성적 요소를 평가할 때 복잡한 의사 결정 절차를 보다 간결한 방식으로 정량화하여 분석할 수 있도록 해주는 방법인 계층분석적 의사결정(Analytic Hierarchy Process, 이하 AHP)[5] 방법론을 이용하여 ATAM의 품질속성들이 갖는 상대적 중요도를 결정하는 방법을 제시하였다.

CBAM의 경우에는 품질속성(예, 성능, 보안, 변경용이성, 가용성 등) 대신 사용가능도(Utility Value)를 도입하고 이해당사자들이 투표나 합의를 통하여 추출한 시나리오에 중요도 값을 부여하도록 하는 방식으로 이익을 정량화하고 있다. 그러나 CBAM은 이해당사자들이 사용가능도 점수를 결정하고 비용을 정량화하기까지의 과정에 대한 논리적 근거(Rationale)를 제시하고 있지 못하다. 따라서 본 논문에서는 유지보수 시나리오들의 아키텍처 전략들이 갖는 상대적 중

* 본 연구는 한국전자통신연구원의 위탁과제 "실시간 임베디드소프트웨어를 위한 계사용기반의 구조설계 기술연구"의 일부임.

† 정 회 원 : 한국정보통신대학교 소프트웨어공학연구소 선임연구원

** 정 회 원 : 한국정보통신대학교 부교수

*** 정 회 원 : 한국전자통신연구원 SW공학연구소

**** 정 회 원 : 한국전자통신연구원 책임연구원

논문접수: 2006년 4월 17일, 심사완료: 2006년 9월 6일

요도인 사용가능도와 각 전략의 비용 및 위험 요인들을 분석하는데 AHP를 적용하여, CBAM에 필요한 의사결정을 보다 간결한 방법으로 수행하게 하고 의사결정방법에 동반된 불확실성을 줄이고자 한다.

본 논문의 구성은 다음과 같다. 먼저 2장에서는 배경 지식으로서 CBAM과 AHP에 관하여 기술하고, 3장에서는 AHP를 적용하고자 하는 CBAM 단계와 적용 방법에 관하여 설명한다. 4장에서는 사례를 중심으로 제시한 방법을 설명하고, 5장에서는 제시한 방법의 이점과 한계에 관하여 기술한다. 마지막으로 결론 및 향후 연구과제에 관하여 기술한다.

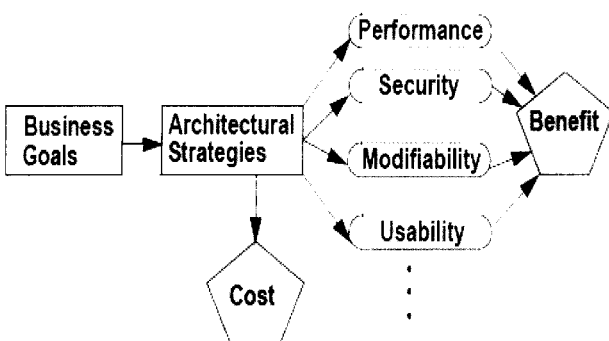
2. 관련 연구

2.1 CBAM²⁾(Cost Benefit Analysis Method)

조직은 이득을 최대화하고 위험을 최소화하기 위하여 자원을 어떻게 투자해야 하는지를 알아야 한다. 과거에는 이와 같은 경제적 문제가 제기될 경우 비용만을 고려하였다. 특히 시스템 유지보수나 업그레이드에 소요되는 장기적인 관점에서의 비용이 아닌 시스템 구축에 소요되는 비용만이 포함되었다. 그러나 아키텍처 결정이 조직에게 줄 수 있는 이익(Benefits)은 매우 중요한 고려사항이며 경우에 따라서는 비용보다 더 중요할 수도 있다.

시스템 구축 및 유지보수를 제한된 비용으로 효과적으로 수행하기 위하여는, 초기 설계 단계와 업그레이드가 진행되는 동안 여러 아키텍처 선택사항들 중 선택하기 위한 합리적인 과정을 거쳐야 한다. 이들 선택사항들은 비용, 자원, 특성(Features)이 모두 다르며 조직에 가져다 줄 이익이 다르다. 그리고 이들 선택사항들에는 위험 및 불확실성이 내재되어 있다.

CBAM은 아키텍처 전략이 시스템의 품질속성에 영향을 끼치고 그 결과로서 이해당사자가 이익을 얻게 되므로 전략을 제대로 선택해야 한다는 것을 기본 개념으로 한다. 소프트웨어 아키텍처 혹은 결정권자는 시스템으로부터 얻게 될 이익과 설계를 구현하는 비용 사이의 차이를 최대화하는 방향으로 아키텍처 전략을 선택한다(그림 1).

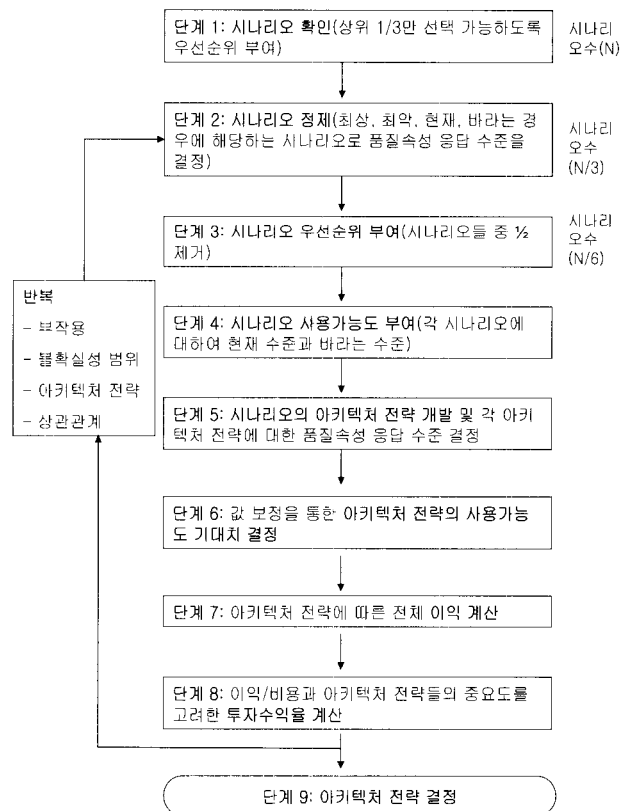


(그림 1) CBAM의 컨텍스트

CBAM의 기본 아이디어는 시스템의 아키텍처 전략들이 성능, 보안, 변경용이성, 사용성 등과 같은 품질속성들에 영향을 끼친다는 것이며, 이러한 특성들은 이해당사자들이 결정한 사용가능도 수준에 반영된다. 그렇지만 각 아키텍처 전략은 그와 관련하여 소모되는 비용을 가지며 구현하는데 시간이 소요되며 구현으로 인한 부작용, 잘못된 추정으로 인한 추가적인 비용 초과 등의 위험요인들을 내재하고 있다. CBAM은 이러한 정보들을 고려하여 이해당사자들이 각 아키텍처 전략에 대한 투자수익률(Return On Invest, ROI)을 계산할 수 있도록 해주며, 이를 바탕으로 아키텍처 전략들을 선택하는데 도움을 준다. 그러나 CBAM에서 투자수익률은 이익 대비 비용을 표현한 용어로서 사용되었을 뿐이며 실제의 투자수익률과는 다른 개념이다. 투자수익률 값이 높다는 것은 아키텍처 구현에 소요되는 비용에 비하여 아키텍처가 주는 이익이 크다는 것을 의미한다.

2.2 CBAM의 프로세스

CBAM의 절차는 (그림 2)에서와 같이 9단계로 구성되어 있으며, 첫 번째 반복에서는 아키텍처 전략의 상대적 이익과 비용을 고려하여 투자수익률을 계산하여 투자수익률이 가장 높은 아키텍처 전략을 결정한다. 그리고 두 번째 반복을 거치면서 위험, 부작용과 같은 불확실성을 반영한다. CBAM은 상대적 중요도를 고려한 이익을 계산하고, 비용을 고려하여 투자수익률을 계산한다. 이 과정은 다음의 9단계로 구성된다.



(그림 2) CBAM 프로세스 흐름 다이어그램

2) 2.1절과 2.2절의 내용은 [2]를 토대로 작성되었다.

단계1-단계3에서 시스템 유지보수를 위한 시나리오가 비즈니스 목표를 만족하느냐에 따라 이해당사자들이 투표를 통하여 시나리오의 우선순위를 결정한다.

CBAM에서 사용가능도는 아키텍처 전략의 이익을 결정하는데 중요한 변수이다. 그러나 CBAM은 사용가능도 값을 결정하기 위한 측도(Measure)나 척도(Scale)를 제공하고 있지 않다. 투자수익률은 단계7에서 계산한 이익을 단계8에서 결정한 비용 또는 일정 추정결과로 나누어 계산한다. 단계8에서 수행하는 비용 또는 일정 추정은 이해당사자들이 결정하며 CBAM은 결정 방법을 제시하고 있지 않다. 이와 같이 CBAM은 아키텍처를 평가하기 위한 체계적인 방법을 제공하고 있지만, 대부분의 단계가 이해당사자들의 투표와 합의, 경험적 직관에 의하여 결과를 도출한다는 문제점이 있다.

2.3 계층분석적 의사결정(AHP)

일반적으로 의사결정문제는 서로 상반된 기준과 불완전한 정보 하에서 최적의 대안을 선택하는 문제를 내포하고 있다. 상이한 척도를 통합하거나 정량적 요소와 정성적 요소를 통합해야 하는 경우 또는 그룹의사결정을 도출해야 하는 경우에 합리적 의사결정을 내리는 과정에 어려움이 있다. AHP³⁾는 Tomas L. Saaty[5]가 개발한 방법으로 의사결정을 내리고자 하는 항목을 필요한 만큼 작은 구성요소로 나누되 그것들을 중요도 및 상호관계에 따라 계층구조(Hierarchy)로 정리하여 분석하는 방법이다. AHP는 의사결정의 목표 또는 평가기준이 다수이며 복잡한 경우 상호배반적인 대안들에 대한 체계적인 평가를 지원하는 다기준 의사결정 방법의 하나로서 다음과 같은 주요 기능을 갖는다.

- 복잡한 상황의 계층화
- 쌍대비교를 통하여 중요도 도출
- 통합 및 논리적 일관성 검증

AHP는 각각의 평가기준들을 계층구조로 재구성한 후 해당 상위기준들에 대한 하위레벨 기준의 요소들을 쌍대비교(pair-wise comparison)하여 상대적 중요도를 결정하는 방법이다. 그런 다음 i 를 j 보다 x 배 중요하게 생각하고 j 를 k 보다 y 배 중요하게 생각한다면 i 를 k 보다 xy 배 중요하게 평가한다는 가정을 통해 두 요소간의 상대적 중요도의 측정 결과를 종합한다. AHP 방법을 이용한 분석 절차는 다음과 같다.

- 브레인스토밍을 통한 계층적 구조 설정: 평가항목을 중요도에 따라 상위 계층과 상위 계층에 영향을 미치는 평가 항목들을 차하위 계층으로 구성하는 방식으로 평가항목을 계층화
- 쌍대비교를 통해 평가요소의 상대적 중요도 설정: 평가항목들 간 쌍대비교를 통하여 가중치를 산정
- 평가요소에 따른 대안의 평가: 논리적 일관성 확인

$$A = [a_{ij}] = \begin{bmatrix} w_1/w_1 & w_1/w_2 & \cdots & w_1/w_n \\ w_2/w_1 & w_2/w_2 & \cdots & w_2/w_n \\ \vdots & \vdots & \ddots & \vdots \\ w_n/w_1 & w_n/w_2 & \cdots & w_n/w_n \end{bmatrix}$$

(그림 3) 쌍대비교 매트릭스

AHP에서 중요도를 결정할 경우 쌍대비교 결과를 도대로 가중치를 산정하기 위하여 다음과 같은 쌍대비교 매트릭스를 구성한다. 의사결정자가 한 수준에서 n 개의 평가항목에 대해 nC_2 회의 쌍대비교를 수행하여 쌍대비교 매트릭스 $A_{n \times n}$ 을 구성한다.

이때 매트릭스 A 를 구성하는 a_{ij} 는 요소 j 에 대한 요소 i 의 상대적 가중치 $\frac{w_i}{w_j}$ 의 추정치이다. 예를 들어 요소 i 와 요소 j 를 쌍대비교 했을 경우 i 가 j 보다 3배 중요하다면 w_i 는 3이고 w_j 는 1이 되므로 a_{ij} 는 3이고 a_{ji} 는 $1/3$ 이 된다.

매트릭스 A 에 평가항목간 상대적 중요도를 나타내는 가중치인 열벡터 $w = (w_1, w_2, \dots, w_n)^T$ 는

$$A \cdot w = \lambda \cdot w$$

관계를 이용하여 계산한다(단, λ 은 행렬 A 의 고유치이다).

AHP 방법은 복잡한 의사결정문제를 전문가 그룹의 객관적인 판단과 수리적인 분석을 통하여 해결하는 방법론 중의 하나로, 특히 여러 전문가들이 참여하는 집단 의사결정에 있어서 나타날 수 있는 특성인의 영향력을 배제하고 합의 도출을 위한 시간 및 비용낭비 등의 현실적인 문제점을 완화하는 장점을 가지고 있다. AHP는 정성적 요소를 평가할 때 전문가의 지식을 적절히 활용하여 복잡한 의사결정의 문제를 체계적으로 분석 가능하도록 해준다. 상대적 중요도를 비율척도를 이용하여 정량화하고 실증 분석과 엄밀한 수리적 검증을 수행함으로써 사회적 합의 도출 및 과정을 명확하게 해준다.

3. CBAM에서 AHP의 활용 방안

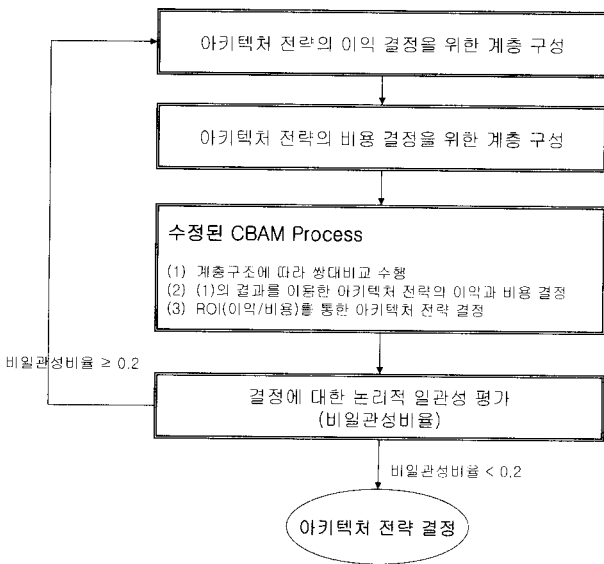
CBAM은 2.1절에서 기술한 바와 같이 많은 부분이 이해당사자들의 경험과 직관에 의존해서 결정된다.

반복 I에서는 시나리오 우선순위를 결정하고 아키텍처 전략을 개발하며 각 아키텍처 전략이 해결할 수 있는 시나리오들을 식별한다. 이 때 이해당사자들은 사용가능도를 통하여 각 시나리오들과 아키텍처 전략이 갖는 유용성을 정량적으로 기술한다. 그러나 CBAM에서 사용가능도 점수를 결정하는 과정에 대한 합리적인 근거를 제시하지 못하고 있다. 더욱이 여러 요소들을 대상으로 하고 있어서 인간의 정보처리능력을 감안할 때 판단 오류의 가능성이 매우 높다.

3) 2.2절의 AHP 내용은 많은 부분이 Expertchoice.co.kr의 연구자료를 참조하였다.

이들 과정에 AHP를 적용하면 이해당사자는 판단 대상의 다른 측면이나 다른 요소를 고려하지 않고 오직 한 가지 성질을 기준으로 두 개의 요소만을 상호 비교하여 상대적 중요도를 결정하므로 보다 단순화할 수 있다. 또한 아키텍처 전략의 이익이 결정되는 과정을 합리적으로 제시할 수 있다.

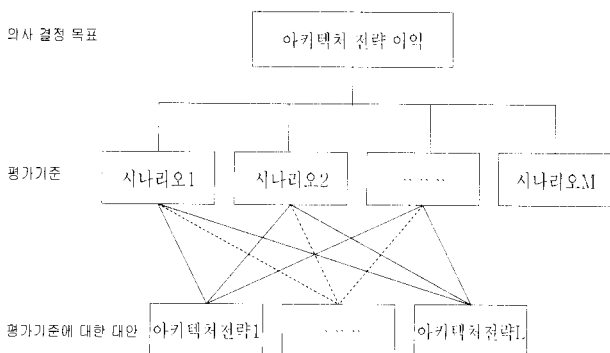
본 장에서는 반복 1의 이들 과정에 AHP를 적용하여 이해당사자들의 경험, 직관 등을 합리적으로 구조화하여 시나리오들 상호간의 중요도를 평가하고 아키텍처 전략을 선별하는 방안을 제시하고자 한다. 본 논문에서 제안한 방법을 적용하기 위한 절차는 다음 그림과 같다.



(그림 4) CBAM에 AHP 적용을 위한 절차

3.1 아키텍처 전략의 이익과 비용을 결정하기 위한 계층 구성

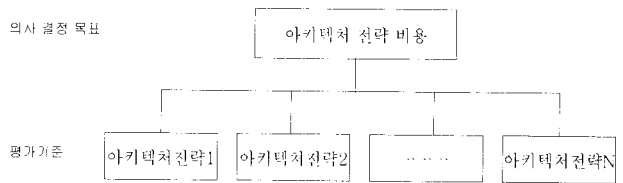
AHP에서 계층은 요소를 상호관계에 따라 동질적인 집단으로 군집화하는 것이다. 어떤 한 레벨에 있는 요소 모두가 인접한 상위레벨에 있는 일부 요소와 관련되어야 하며 요소들의 최종 결과에 대한 상대적 중요도를 파악하여야 한다. 상대적 중요도는 우선순위 설정과정을 통하여 이루어진다. 이 과정에서 전체 목적에 미치는 영향이 약한 요소는 제거될 수 있다.



(그림 5) 아키텍처전략 이익 결정을 위한 계층 구성

CBAM에 AHP를 적용하기 위하여 계층을 구성할 때에는 최상위 수준에는 CBAM의 궁극적 목표인 아키텍처 전략 이익 결정을 나타내고 제 1수준에는 아키텍처 전략의 이익 결정에 1차적인 영향을 미치는 평가 기준인 시나리오를, 제 2수준에는 시나리오들을 구현하기 위한 아키텍처 전략들을 나타낸다(그림 5).

(그림 5)에서 아키텍처 전략의 이익은 이해당사자들이 결정한 시나리오에 대한 상대적 중요도를 가중치로 하여 이들 시나리오들을 구현하기 위한 아키텍처 전략들과의 관계에 의해서 다시 중요도가 결정된다. 이 두 과정의 통합을 통해서 각 아키텍처 전략에 대한 이익이 결정된다. 반면 아키텍처 전략의 비용 결정에 영향을 주는 요소는 아키텍처 전략 뿐이므로 (그림 6)과 같은 계층구조에 의하여 의사 결정 목표를 달성한다.



(그림 6) 아키텍처전략 비용 결정을 위한 계층 구성

3.2 CBAM에 AHP 적용

3.1절과 같이 계층구조를 구성한 후 아키텍처 전략을 평가하기 위하여 다음과 같은 단계를 거친다.

- (1) 쌍대비교를 통한 시나리오들의 상대적 중요도⁴⁾ 가중치 결정
- (2) 아키텍처 전략들을 쌍대비교하여 대안 평가를 위한 매트릭스 작성
- (3) 중요도 순위 결정((1)의 결과 * (2)의 결과)
- (4) 쌍대비교를 통한 아키텍처 전략들의 비용 우선순위 결정
- (5) ROI 계산((3)의 결과 / (4)의 결과)
- (6) 아키텍처 전략 선택

본래 CBAM은 2.1절에서 설명한 바와 같이 9단계를 통하여 아키텍처 전략을 선택한다. CBAM에 AHP를 도입하면 6단계로 단축할 수 있다.

3.3 결정에 대한 논리적 일관성 평가

논리적 일관성은 유사한 개념이나 사물들이 동질성이나 관련성의 기준에 따라 그룹핑되고 어떤 기준에 대한 사물이나 개념 간의 관계도 논리적 방법으로 정당화되었음을 의미한다. 논리적으로 일관된다는 것은 A가 B보다 5배 중요하고 B가 C보다 2배 중요하다면, A가 C보다 10배 중요한 것으로 판단해야 함을 의미한다. 그렇지 않은 경우는 논리적

4) 상대적 중요도 평가에는 AHP의 9점 척도를 사용하였다[6].

으로 비일관적이라고 할 수 있다.

AHP는 계층 구조를 결정하고 요소들의 상대적 중요도를 측정하며 이러한 판단이 일관적인지를 판단하는 논리적 일관성 평가를 통하여 이를 검증한다. 논리적 일관성 판단은 비교 매트릭스의 고유 벡터를 활용한 쌍대비교 결과의 통합 과정에서 비일관성 비율(CR)을 도출함으로써 이루어진다.

$$CR = CI / RI$$

$$CI = (\lambda_{\max} - n) / (n-1), \text{ 단 } n\text{은 요소의 수}$$

RI(Random Index, 무작위 지수)는 순수하게 무작위로 응답한 경우의 CI(Consistency Index, 일관성 지수)를 의미한다. Saaty[7]는 비일관성 비율(CR)이 0.1 미만이면 쌍대비교는 합리적인 일관성을 갖는 것으로 판단하고, 0.2 이내일 경우 용납할 수 있는 수준의 비일관성을 가지나, 0.2 이상이면 의사결정자의 논리성과 합리성에 대한 재검토가 필요하다고 제안한다.

CBAM에서는 아키텍트나 전문가들의 경험적 직관에 의한 판단이 합리적이었는지 검증할 만한 방법을 제시하고 있지 못하다. 그러나 CBAM에 AHP를 적용하면 논리적 일관성 평가 방법을 이용하여 판단이 논리적으로 일관되었는지를 평가할 수 있다.

4. 적용 사례

본 절의 사례는 Glass Box(GB) project에 기초한다[3]. GB project에서 아키텍처(ARCH)를 구현하기 위한 아키텍처 전략은 J2EE를 이용한 3 tier(THTJ), '.Net'을 이용한 3-tier(THTD), 2-tier(TWOT) 그리고 COABS(COAB)이다. GB project에서는 ARCH와 관련된 품질 속성들을 제시하고 있지만 본 논문에서는 비용을 제외한 나머지 품질 속성을 시나리오로 대체하여 사용하였다.

<표 1>은 각 시나리오들을 쌍대비교한 후 2.2절에서 설명한 방법을 이용하여 계산한 결과이다. Aggregated 값은 본래 고유치 계산을 통하여 도출되나 본 연구에서는 근사법을 이용하여 기하평균으로 계산하였다. Aggregated 값은 각

<표 1> 시나리오들에 대한 상대적 중요도⁵⁾

시나리오	이해당사자			Aggregated
	Dev. Team	Research Team	Funding Agency	
시나리오 1	0.216	0.294	0.184	0.227
시나리오 2	0.087	0.092	0.038	0.067
시나리오 3	0.052	0.117	0.087	0.081
시나리오 4	0.245	0.019	0.272	0.108
시나리오 5	0.050	0.155	0.053	0.074
시나리오 6	0.106	0.304	0.093	0.144

5) [3]의 Table 4의 내용이다.

<표 2> 시나리오 별 아키텍처 전략의 상대적 중요도 가중치

시나리오	아키텍처 전략			
	THTJ	THTD	TWOT	COAB
시나리오 1	0.521	0.182	0.046	0.250
시나리오 2	0.402	0.402	0.054	0.143
시나리오 3	0.204	0.204	0.347	0.246
시나리오 4	0.152	0.110	0.515	0.223
시나리오 5	0.450	0.050	0.050	0.450
시나리오 6	0.168	0.368	0.256	0.208

시나리오들에 대하여 이해당사자들이 생각하는 상대적 중요도 값을 통합한 것으로 다음 계층의 요소들의 가중치 벡터 w 가 된다.

다음은 각 시나리오들에 대하여 아키텍처 전략들이 갖는 상대적 관련성을 쌍대비교를 이용하여 동일한 방법으로 계산한 결과이다. 이 결과를 통하여 각 시나리오를 지원하는 아키텍처 전략들의 상대적 중요도를 판단할 수 있다.

다음으로 <표 1>과 <표 2>의 결과를 통합하여 아키텍처 전략의 이익을 계산한다. 아키텍처 이익은 이전 계층에서 결정된 각 시나리오들의 상대적 중요도를 나타내는 가중치 벡터 w 를 이용하여 다음과 같은 식에 의하여 계산할 수 있다.

$$\text{이익} = \begin{bmatrix} 0.521 & 0.042 & 0.204 & 0.152 & 0.450 & 0.168 \\ 0.182 & 0.402 & 0.204 & 0.110 & 0.050 & 0.368 \\ 0.046 & 0.054 & 0.347 & 0.515 & 0.050 & 0.256 \\ 0.250 & 0.143 & 0.246 & 0.223 & 0.450 & 0.208 \end{bmatrix} \times \begin{bmatrix} 0.227 \\ 0.067 \\ 0.080 \\ 0.108 \\ 0.074 \\ 0.144 \end{bmatrix}$$

그리고 각 아키텍처 전략에 대한 비용 특성⁶⁾은 (그림 6)과 같은 계층구조에 따라 쌍대비교를 통하여 계산하였으며 이익/비용으로 계산 ROI 결과는 <표 3>과 같다.

CBAM의 반복 1의 과정에서 많은 불확실성을 내포하고 있는 단계는 '아키텍처 전략이 영향을 주는 시나리오'를 결정하고 사용가능도 응답곡선의 특성에 따라 사용가능도를 보정하는 과정이다. 특히 이해당사자들로부터 사용가능도 특성을 추출하여 사용가능도-응답곡선을 결정하는 과정은 상당한 시간을 소요하는 과정이다. 무엇보다도 향후 사용가능도를 예측한다는 것은 불확실한 것이다. 따라서 이 과정을 AHP를 이용하면 보다 정량적인 방법으로 결정이 가능하다.

<표 3> 아키텍처 전략의 이익 결정

	아키텍처 전략			
	THTJ	THTD	TWOT	COAB
이익	0.236	0.154	0.138	0.174
비용	0.166	0.120	0.487	0.227
ROI	1.421687	1.283333	0.283368	0.76652

6) 비용에 대한 쌍대비교 결과는 [3]의 품질 속성 중 비용의 가중치 결정 결과를 그대로 사용하였다.

<표 4> Funding Agency의 시나리오에 대한 쌍대비교

	시나리오 1	시나리오 2	시나리오 3	시나리오 4	시나리오 5	시나리오 6
시나리오1	1	5	2	1	3	1
시나리오2	1/5	1	1/4	1/5	1/2	1/2
시나리오3	1/2	4	1	1/5	2	1
시나리오4	1	5	5	1	5	4
시나리오5	1/3	2	1/2	1/5	1	1/2
시나리오6	1	2	1	1/4	2	1

마지막으로 논리적 일관성을 측정하였다. <표 4>는 이해 당사자들 중 Funding Agency의 쌍대비교 결과이다. <표 4>를 토대로 AHP의 논리적 일관성 측정 방법의 근사법으로 계산하였다.

<표 1>에 제시되어 있는 Funding Agency의 시나리오 별 가중치를 각 요소에 곱한 후 행의 평균을 계산한다.

$$0.184 \times \begin{bmatrix} 1 \\ 0.2 \\ 0.5 \\ 1 \\ 0.33 \\ 1 \end{bmatrix} + 0.038 \times \begin{bmatrix} 5 \\ 1 \\ 4 \\ 5 \\ 2 \\ 2 \end{bmatrix} + 0.087 \times \begin{bmatrix} 2 \\ 0.25 \\ 1 \\ 5 \\ 0.5 \\ 1 \end{bmatrix} + 0.272 \times \begin{bmatrix} 1 \\ 0.5 \\ 0.2 \\ 1 \\ 0.2 \\ 0.25 \end{bmatrix} + 0.053 \times \begin{bmatrix} 3 \\ 0.5 \\ 2 \\ 5 \\ 1 \\ 2 \end{bmatrix} + 0.093 \times \begin{bmatrix} 1 \\ 0.5 \\ 1 \\ 4 \\ 0.5 \\ 1 \end{bmatrix}$$

이 평균을 다시 시나리오 별 가중치 벡터로 나누면 다음과 같은 고유치 λ 를 얻을 수 있다.

$$\lambda = \begin{bmatrix} 5.8261 \\ 8.0407 \\ 6.7172 \\ 5.8975 \\ 6.3157 \\ 6.6022 \end{bmatrix}$$

그런 다음 이들의 평균을 계산하여 최대 고유치 λ_{max} 를 계산한다. 일관성 지수 CI는 $(\lambda_{max} - n)/(n-1)$ 로 구할 수 있다.

$$\lambda_{max} = 6.5665$$

$$CI = \frac{(6.5665 - 6)}{(6 - 1)} = 0.11$$

RI⁷⁾ 값은 가중치 벡터 λ 의 요소 수가 6이므로 $n=6$ 인 경우 RI 값이 1.24이므로

$$CR = 0.11/1.24=0.089$$

그 결과 Funding Agency 이해당사자의 비밀관성비율은 0.089이었으며 Saaty[7]의 연구 결과에 따르면 논리적인 의

사결정이었다고 판단할 수 있다. 같은 방법으로 다른 팀들에 대한 일관성 지수도 계산할 수 있다.

다음으로 아키텍처 전략의 이익을 결정하기 위한 계층구조가 2계층이므로(그림 4) 계층별 CI, CR 값을 종합하여 비밀관성비율을 산정하는 종합 비밀관성비율을 계산한다. 제2계층의 CI값들은 제1계층의 요소들(시나리오1~시나리오6)을 기준으로 제2계층의 행렬들을 이용하여 가중치 벡터 w 를 계산하고 고유치 λ 를 도출하는 등 Funding Agency와 같은 방법으로 계산할 수 있다.

쌍대비교 결과를 이용하여 제2계층에서 시나리오1에 해당하는 각 아키텍처 전략의 CI를 Funding Agency와 같은 방법으로 계산한 결과를 간단히 기술하면 다음과 같다.

$$\text{가중치 벡터 } w = (0.450, 0.050, 0.050, 0.450)$$

$$\lambda = (4, 4, 4, 4)$$

$$CI = 0$$

나머지 시나리오에 대하여서도 같은 방법으로 CI값을 계산할 수 있으며 그 결과는 다음과 같다.

$$\text{제1계층의 가중치 벡터} = (0.227, 0.067, 0.081, 0.108, 0.074, 0.144)$$

$$\text{제1계층의 CI} = 0.11$$

$$\text{제2계층의 CI} = (0, 0.25, 0, 0, 0.105, 0.25)$$

$$\begin{aligned} \text{전체 CI} &= \text{제1계층의 CI} + \text{제1계층의 가중치벡터} \times \text{제2계층의 CI} \\ &= 0.222 \end{aligned}$$

$$\begin{aligned} \text{전체 RI} &= \text{제1계층의 RI} + \text{제1계층의 가중치벡터} \times \text{제2계층의 RI}(n=4\text{이면 } 0.9\text{임}) \\ &= 1.87 \end{aligned}$$

$$\text{종합 CR} = \text{전체 CI} / \text{전체 RI} = 0.222/1.87 = 0.12$$

따라서 아키텍처 전략의 이익의 계층구조 또한 논리적으로 일관성을 갖는다고 볼 수 있다.

결론적으로 <표 3>의 결과는 여러 아키텍처 전략들 중 THTJ를 선택한 정량적 근거를 제시해 주며, 더불어 이러한 정량적 근거가 어느 정도의 정확성을 갖는지는 비밀관성비율 값을 통하여 검증할 수 있었다. CBAM은 아키텍처 전략 결정이 올바른 것이었는지 여부를 구현에 앞서 평가할 수 있는 방법을 제공하고 있지 않다. 이에 AHP를 도입함으로써 아키텍처 전략 결정의 정확성 여부를 판단하고, 추가적인 검토와 같은 행동을 취할 수 있다.

또한 CBAM은 아키텍처 전략 선택에 참여하는 이해당사자들이 해당 도메인에 대한 상당한 경험과 지식을 가지고

7) RI 값은 Wharton School에서 표본수 500개를 이용하여 계산된 값을 이용하였다.

있다는 것을 전제로 하고 있다. 이는 조직의 다양성이나 현실을 고려하지 않은 방법론의 일방적인 전제라고 볼 수 있다. 반면 AHP는 의사 결정자가 다수인 경우, 의사결정에 참여하는 개인들이 전문성이 높은 집단인지 아닌지에 따라 적절한 방법을 선택할 수 있도록 해준다. 따라서 CBAM에 AHP를 도입하면 의사결정에 대한 정량적 근거 제공 이외에도, 조직의 특성을 반영한 의사결정이 가능해진다.

5. 결론 및 향후 연구

AHP는 국가기관, 공공, 기업 등에서 분석 및 계획 수립을 위하여 사용하고 있다. 본 연구에서는 비용평가 모델인 CBAM에서 아키텍트 혹은 그 외 이해당사자들의 경험이나 직관에 의한 판단이 필요한 단계에 AHP를 도입하여 의사결정을 도출하는 방안을 기술하였다.

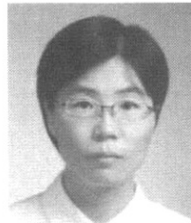
CBAM은 기본적으로 많은 단계와 단계별로 수행해야 할 일을 제시하고 있으며 이해당사자들의 투표를 통한 합의를 기반으로 하고 있어 실용화하기 어려운 부거운 프로세스(heavy process)로 알려져 있다. 본 연구에서는 이해당사자들의 투표를 통한 합의 방식을 AHP의 쌍대비교를 사용하여 이익 및 비용이 정량적으로 결정되는 과정을 제시할 수 있도록 하였으며 더불어 사람들의 주관적 판단 오류의 영향을 최소화하고자 하였다. 또한 사용가능도를 보정할 때 CBAM에서 사용하는 사용가능도 응답곡선을 사용하는 대신 평가 기준에 대한 대안별 가중치 매트릭스와 평가 기준별 가중치 매트릭스를 곱한 의사결정 매트릭스를 이용하여 이해당사자의 참여를 최소화하였다. 본 논문에서는 언급하지 않았지만 CBAM의 반복 II에서 불확실성 및 위험 분석에서도 같은 방법으로 AHP를 적용할 수 있다.

그러나 AHP의 쌍대비교를 통하여 타 의사결정 방법들보다 경험이나 직관 등의 주관적 판단의 오류를 최소화하고 구조적 추상화를 가능하게 하는 단순 명료한 방법이지만[8], 과도한 쌍대비교를 수행하면서 발생할 수 있는 한계를 내재하고 있다. 또한 아키텍처 전략의 최종 이익 평가에서 CBAM이 제안한 사용가능도 응답곡선을 논문에서 사용한 의사 결정 매트릭스가 대체할 수 있는지 여부는 추가적인 검증이 필요한 부분이다.

참 고 문 헌

- [1] P. Clements, R. Kazman, and M. Klein. Evaluating Software Architectures Methods and Case Studies, Addison Wesley, 2002.
- [2] R. Kazman, J. Asundi, and M. Klein. "Making Architecture Design Decisions: An Economic Approach (CMU/SEI 2002-TR-035, ESC-TR-2002-035)". Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2002.
- [3] T. Al-Naeem, I. Gorton, M.A. Rabhi, and B. Benatallah, A Quality-Driven Systematic Approach for Architecting Deistributed Software Applications, In Proceeding of the 27th International Conference on Software Engineering (ICSE), St. Louis, USA.
- [4] L. Zho, A. Aurum, I. Gorton, and R. Jeffery, "Tradeoff and Sensitivity Analysis in Software Architecture Evaluation Using Analytic Hierarchy Process", Software Quality Journal, 13, pp.357-375, 2005.
- [5] T.L. Saaty, The Analytic Hierarchical Process. McGraw-Hill, 1980.
- [6] T.L. Saaty and L.G. Vargas, The Logic of Priorities, Kluwer-Nijhoff Publishing, London, 1982.
- [7] T.L. Saaty, Multicriteria Decision Making: The Analytic Hierarchical Process, I, AHP series, RWS Publication, 1990.
- [8] K. Peniwati, Evaluation of Group Decision Making Method, Ph.D. Dissertation, 2001.

이 지 현



e-mail : puduli@icu.ac.kr

1993년 전북대학교 정보통신공학과 (공학사)

2000년 전북대학교 교육대학원 전자계산교육(교육학석사)

2005년 전북대학교 대학원 전산통계학과 (이학박사)

2005년~현재 한국정보통신대학교 소프트웨어공학연구소 선임연구원

관심분야: Software 테스트, Software Product Line

강 성 원



e-mail : kangsw@icu.ac.kr

1982년 서울대학교 외교학과(정치학사)

1989년 Univ. of Iowa 전산학 (전산학석사)

1992년 Univ. of Iowa 전산학 (전산학박사)

1993년~2001년 한국통신 연구개발본부 선임연구원

1995년~1996년 미국 국립표준기술연구소(NIST) 객원연구원

2001년~현재 한국정보통신대학교 부교수

2003년~현재 미국 Carnegie-Mellon University

소프트웨어공학석사과정 겸임교수

관심분야: 소프트웨어구조, 소프트웨어시험, 형식기법

조 진 희



e-mail : chojh@etri.re.kr
1993년 경북대학교 컴퓨터공학(학사)
1996년 경북대학교 컴퓨터공학(석사)
1995년~1998년 KIST 부설
시스템공학연구소
1998년~현재 한국전자통신연구원
SW공학연구팀

관심분야 : 시스템 개발방법론, SW아키텍처, 제품계열, UML,
SW테스팅

김 진 삼



e-mail : jinsam@etri.re.kr
1984년 중앙대학교 전자계산학과(이학사)
1986년 중앙대학교 대학원
전자계산학과(이학석사)
2006년 대전대학교 대학원
컴퓨터공학과(공학박사)

1987년~1994년 시스템공학센터 연구원
1994년~1996년 시스템공학연구소 선임연구원
1996년~2001년 한국전자통신연구원 선임연구원
2001년~현재 한국전자통신연구원 책임연구원
관심분야 : 소프트웨어 개발방법론, 소프트웨어 공학 표준