

컴포넌트 기반 소프트웨어 유지보수 의사결정에 관한 연구 A Study on Decision Making for the Maintenance of Component-Based Software

윤민석

여수대학교 멀티미디어학부

김계영

고려대학교 경영대학

Abstract

최근 재사용성에 근간을 둔 컴포넌트 기반 소프트웨어(Component-Based Software : CBS) 기술의 실현에 따라 이와 관련된 유지보수 의사결정의 내용도 변화하여 새로운 연구 과제로 등장하고 있다. 이에 본 연구는 컴포넌트 기반 소프트웨어 유지보수와 관련된 의사결정의 이론적 근거를 탐색하고 컴포넌트 평가 및 선택을 위한 모형 개발을 목적으로 한다. 모형 설계에는 평가 기준의 체계를 수립하고 적절한 평가 기법의 선택 및 적용이 포함된다. 평가 기준 구성을 위하여 CBS 유지보수의 특성을 고려한 다차원적 접근법에 의한 계층적 체계화를 시도하였으며, 일부 상호 종속적 관계를 나타내는 요소들에 대하여는 네트워크 관계를 나타내는 모듈로 포함시켰다. 체계화된 평가 기준에 따른 의사결정 대안의 평가 기법으로 주관적 판단의 계량화에 적절하다고 판단되는 계층분석과정(Analytic Hierarchy Process) 기법 및 네트워크분석과정(Analytic Network Process) 기법을 선정하고 적절한 예를 들어 모형 적용의 이해를 돕고자 하였다.

1. 서론

최근 소프트웨어와 관련한 이슈 중 중요하게 논의되는 두 가지는 관리적 측면에서의 소프트웨어 유지보수 문제와 기술적 측면에서 컴포넌트 기반 소프트웨어(Component-Based Software : CBS)이다. 다양한 형태의 업무처리 욕구를 수용하게 되면서 대규모화된 소프트웨어는 시스템의 구축 또는 도입 이후, 새로운 기능의 추가 및 성능의 향상 또는 새로운 환경으로의 적용 활동을 포함하는 유지보수의 어려움과 이의 해결을 위한 막대한 비용의 증가를 야기 시키고 있다 [1, 21].

한편 객체지향(object-oriented) 기법의 발달과 소프트웨어 재사용을 위한 노력은 소프트웨어의 컴포넌트화를 실현시키고 있다. 이를 통하여 독립적으로 컴파일된 소프트웨어 컴포넌트들의 결합으로 전체 소프트웨어(CBS)를 구성하고 필요한 컴포넌트의 대체 및 추가를 통한 전체 소프트웨어의 개선 및 비용절감을 추구하고자 한다. 최근 이들을 위해 발표된 아키텍처(architecture) 중 SUN사의 Java Beans, Microsoft사의 DCOM(Distributed Component Object Modeling), OMG(Object Management Group)의 CORBA(Common Object Request Broker Architecture) 등을 예로 들 수 있다[4]. 나

아가 ERP(Enterprise Resource Planning)와 같은 대규모 정보시스템도 컴포넌트 기반으로 구축하는 방안이 제시되고 있다 [12].

CBS의 유지보수는 원시코드(source code)를 수정하는 기존 형태와 달리 기존 컴포넌트의 대체 또는 추가를 통하여 수행되므로 최적의 컴포넌트 선택이 핵심적인 의사결정 문제로 부각된다. 그러나 지금까지 국내외의 관련 연구들은 컴포넌트의 설계 및 개발에 관련한 공학적 접근에만 치우쳐, 관리적 측면에서 접근해야 할 유지보수 문제는 도외시되고 있다. 이로 인하여 그 중요성에도 불구하고 컴포넌트의 평가 및 선택과 관련한 적절한 모형이 제시되어 있지 않다.

이에 본 연구는 CBS 유지보수 단계에서 활용할 수 있는 체계적인 컴포넌트 평가 모형의 개발을 목적으로 한다.

2. CBS 유지보수 고찰

2.1 CBS 유지보수의 특성

소프트웨어 컴포넌트란 형태적으로 볼 때, 원시코드로부터 컴파일 되어 독립적으로 존재하며 다른 소프트웨어와 결합되어 사용되어지는 소프트웨어로 plug&play 방식으로 작동되는 완성된 형태를

통상적으로 의미한다[3]. 기술적으로 컴포넌트는 오브젝트(object=attributes+operations)에 인터페이스(interface)와 관련된 제약/순서/상호작용(constraints/sequencing/interactions)이 결합된 것으로 파악된다[8]. 컴포넌트들로 소프트웨어를 구성할 때 이들간의 관계와 구조를 정의하는 컴포넌트 아키텍처 설정이 선행되어야 한다. 이 때 서비스 수준, 유연성, 예외처리, 통합 환경 등이 결정되므로 아키텍처는 유지보수 활동과도 밀접하게 관련된다[8, 17].

유지보수 단계에서 CBS는 기존의 맞춤형(custom-build) 소프트웨어와 비교할 때, 다음과 같은 차이가 있다. 우선, 시스템 엔지니어는 원시 코드를 접근할 필요가 없다. 둘째, 유지보수는 컴포넌트 대체 및 추가를 통한 컴포넌트 결합으로 수행된다. 마지막으로 컴포넌트 자체의 유지보수 또는 개선은 컴포넌트 개발자의 몫이다[17]. 이는 유지보수와 관련된 의사결정 과제의 근본적인 변화를 초래한다. 전통적 유지보수는 원시코드를 접근해야 하기 때문에 유지보수의 외주(outsourcing) 여부가 중요한 결정 사항 중 하나였다. 그러나 CBS 환경에서는 주어진 여건에서 최적의 시스템을 구성하기 위한 적절한 컴포넌트의 선택이 유지보수를 위한 핵심적인 과제로 대두된다.

2.2 평가 기준의 체계화

일반적으로 소프트웨어 평가는 평가 목적과 제품 특성에 따라 상이한 평가 기준이 제시되어 왔다[7]. 상용 소프트웨어 평가와 관련한 선행 연구들은 기능, 성능, 품질, 사용자 인터페이스, 가격, 조직 적합성, 공급자 지원 등을 주요한 평가 요인으로 제시하고 있다[5, 16, 20].

한편, 소프트웨어 수용 과정에서 사용자 태도에 영향을 미치는 주요 요인으로 유용성과 사용의 용이성이 제시된 바 있다[6]. 신뢰성, 성능, 사용성, 설치성, 유지보수성, 성과, 문서화 등은 소프트웨어 사용자 만족에 영향을 미치는 주요 요인으로 제시되어 있다[11]. 또한 사용자 만족을 목표로 하는 품질 접근법은 기능성, 신뢰성, 사용성, 효율성, 유지보수성, 이식성을 주요한 소프트웨어 품질 특성으로 제시하고 있다[9, 10]. 이들 중 기능성, 신뢰성, 사용성, 효율성이 패키지 최종 사용자를 위한 주요한 품질 특성임이 확인되었다[2, 19].

소프트웨어 재사용을 위한 평가 기준으로 기능적 요구사항, 품질, 전략적 고려사항 그리고 아키텍처 호환성(compatibility)이 제안된 바 있다[13]. Yacoub 등[18]은 소프트웨어 컴포넌트의 특성을 외부/내부(external/internal) 특성으로 구분하였는데 이중 유지보수 단계에 관련되는 것은 상호운용성(interoperability)과 이식성(portability)이다.

본 연구에서는 CBS 유지보수 단계에서 컴포넌트 평가에 요구되는 기준의 체계화를 위하여 단계적 접근법을 이용하였다. 우선, 소프트웨어 종합적 품질을 위주로 하여 전략적 측면의 공급자 평가요인 및 비용요인을 함께 고려하였다. 품질은 컴포넌트의 최종 사용자를 위한 품질과 유지보수를 담당하는 사내의 시스템 엔지니어를 위한 품질로 구분하여 1단계 4개의 평가 요인을 식별한 후 각 요인을 다시 세분하여 다음 <표 1>과 같다.

<표 1> 컴포넌트 평가 기준

최종 사용자 품질	기능성	적절성(f1), 정밀성(f2), 상호운용성(f3), 순용성(f4), 보안성(f5)
	신뢰성	성숙성(r1), 고장허용(r2), 회복성(r3)
	사용성	이해성(u1), 학습성(u2), 운용성(u3)
	효율성	시간행동(e1), 자원행동(e2)
시스템 엔지니어 품질	유지보수성	분석성(m1), 변경성(m2), 안정성(m3), 시험성(m4)
	아키텍처	소프트웨어 아키텍처(ac1), 호환성
	도메인 아키텍처(ac2)	
공급자 평가요인	공급자 신뢰성	
	발전 가능성	
	지원 사항	
비용	직접비 : 제품 가격	
	간접비 : 교육훈련비, 기타 운영관련비	

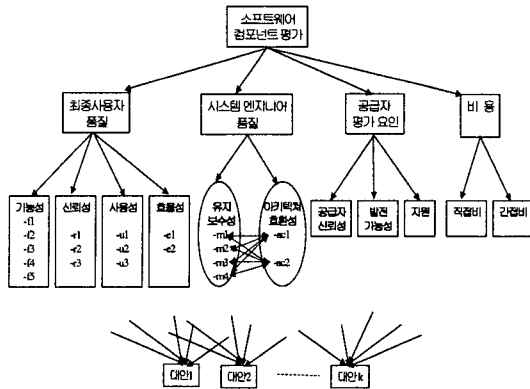
3. 컴포넌트 평가 모형 설계

3.1 계층분석과정(Analytic Hierarchy Process)

기법을 통한 평가 모델

본 연구에서 제시한 유지보수 단계에서의 컴포넌트 평가를 위한 평가 기준은 [그림 1]과 같이 계층적으로 분해할 수 있다. 계층적 분해는 문제를 구조화하여 평가 과정에서 소요되는 요소들에 대한 이해를 보다 명확히 할 수 있다. 평가 구조의 특징적인 면만을 약속하면 첫째, 본 연구의 평가 기준이 대체적으로 정성적(qualitative) 요인이라는 점과 둘째, 대체적으로 각 요소는 차상위 단계의 요인에 의해서만 영향을 받고 동일 그룹내 요소들 간에는 독립성이 존재한다는 점, 마지막으로 시스템 엔지니어 품질의 하위 요소인 유지보수성과 아키텍처 호환성간에는 상호 종속적 관계가 존재할 것으로 판단된다는 점이다. 상호 종속관계의 존재는 아키텍처 구성이 유지보수성에 영향을 미칠 수밖에 없다는 기술적 고려로부터 연유한다. [그림 1]에 동일 그룹내 독립성이 존재하는 요소는 박스로 표시하였으며 종속적 관계가 있는 요소는 타원으로 표시하였다.

3.2 네트워크분석과정 (Analytic Network Process) 기법을 통한 평가 모델



[그림 1] 컴포넌트 평가 모형도

본 연구의 컴포넌트 대안 평가는 각 기준에 대하여 대안을 평가하고 이를 종합하는 다기준 의사결정(Multi-Criteria Decision Making)이다. 다기준 의사결정의 주요 과정은 각 기준의 중요도를 산출하는 단계와 각 기준에 대한 각 대안들의 우수성을 평가하여 대안별 복합 가중치를 산출함으로써 최적 대안을 선택하는 단계로 구분된다. 이 때 각 기준의 중요도는 계층을 따라 순차적으로 하위 기준의 중요도에 전가된다. 이를 본 연구의 모형에 적용하면 대안의 우수성은 계층 구조의 최하위 단계 기준에 대하여만 평가된다. 이를 수식으로 표현하면 다음과 같다.

$$Q_k = \sum_{i=1}^{I_{\max}} w_i \cdot q_{ik}, \text{ 여기서 } \sum_{i=1}^{I_{\max}} w_i = 1 \quad \text{식(1)}$$

- Q_k : k번째 컴포넌트 대안의 종합적 우수성
- w_i : i번째 기준의 상대적 중요도
- q_{ik} : i 번째 기준에 대한 k번째 대안의 우수성
- I_{\max} : 대안을 평가할 최하위 단계 기준의 총 수

본 연구의 평가 모형에 적용될 평가 기법으로는 주관적 판단의 객관화와 계층적 분화의 이점을 비교적 잘 반영할 수 있다고 판단되는 계층분석과정 (Analytic Hierarchy Process : AHP) 기법을 근간으로 하고, 상호 종속적 관계를 반영하여 중요도를 산출할 수 있는 네트워크분석과정 (Analytic Network Process : ANP) 기법을 병용하고자 한다.

AHP는 지난 20여년간 광범위한 분야에 적용되어 계층적 분화를 이용한 연산절차는 이미 널리 알려져 있고 [14] 소프트웨어 평가 분야에서도 여러 차례 적용되어 온 바 [1, 2, 10, 19], 본 연구에서는 이에 관한 기술은 생략하고 ANP를 통한 평가 모델 부분만을 기술하고자 한다.

ANP는 평가 구조의 목표, 기준 그리고 대안 상호간의 종속성이나 피드백을 포함하는 네트워크 구조의 의사결정 시스템이다. 이 때 종속성은 구성 요소들간 존재하는 영향력의 종속성을 의미하며 집단간 요소들 사이에 작용하는 외적 종속성 (inter-dependence)과 집단 내 요소들 사이에 작용하는 내적 종속성 (inner-dependence)으로 구분된다.

본 연구에서 기술적 내용을 고려하여 파악한 유지보수성과 아키텍처 호환성의 하위 요소들간 상호 영향력의 관계는 [그림 1]과 같이 외적 종속성이 있는 것으로 파악된다. 이들 관계 속에서 각 하위 요소의 중요도 산출은 네트워크라는 그래픽 표현을 수퍼매트릭스 (Supermatrix : Z)라는 대수적 표현으로 전환하여 수행된다. 최초 (initial) 수퍼매트릭스의 작성은 각 열별로 우선, 현재 열에 영향을 받는 요소들의 쌍대비교를 통한 고유벡터를 구하고 합이 1이 되도록 정규화 한다. 산출된 중요도를 현재 열과 해당 요소를 나타내는 행과의 교차점에 기록하고 나머지 행과의 교차점에는 0을 기록한다.

이렇게 구성된 수퍼매트릭스는 각 열별로 그 합이 1이 되는 확률 행렬 (stochastic matrix)이며 이 행렬의 고유치 중 최대값은 1이다. 최초 수퍼매트릭스의 극한 (Z^∞)을 취하여 얻어지는 결과가 각 요소의 중요도가 된다. 이 때 구성된 수퍼매트릭스의 대수적 특성에 따라 Z^∞ 가 안정화 상태 (stationary state)에 도달하지 않을 수도 있으며 Saaty는 이러한 경우 대체식을 제안하였다 ([15], pp. 108-127 참조). 본 연구의 수퍼매트릭스도 특성상 다음과 같은 식이 적용된다.

$$\frac{1}{c} (I - Z^c) (I - Z)^{-1} (Z^c)^\infty \quad c \geq 2 \quad \text{식(2)}$$

여기서, I 는 단위행렬이며 c 는 수퍼매트릭스가 극한을 취할 때 동일한 결과가 반복되는 순환 주기 (cyclicality)이다.

4. 평가 모형의 예시

[그림 1]의 네트워크에 포함되는 유지보수성의 분석성(m1), 변경성(m2), 안정성(m3), 시험성(m4)과 아키텍처 호환성의 소프트웨어 아키텍처(ac1) 및 도메인 아키텍처(ac2)의 요소간 영향력 정도를 <표 2>와 같이 예로 제시하여 최초 수퍼매트릭스를 구성하였다. 이에 극한을 취하면 순환 주기 2인 수퍼매트릭스가 됨을 알 수 있다.

<표 2> 최초 수퍼매트릭스

구분	유지보수성				아키텍처 호환성	
	m ₁	m ₂	m ₃	m ₄	ac ₁	ac ₂
m ₁	.0000	.0000	.0000	.0000	.4000	.3000
m ₂	.0000	.0000	.0000	.0000	.3000	.5000
m ₃	.0000	.0000	.0000	.0000	.2000	.1000
m ₄	.0000	.0000	.0000	.0000	.1000	.1000
ac ₁	.4000	.6000	.8000	.7000	.0000	.0000
ac ₂	.6000	.4000	.2000	.3000	.0000	.0000

예를 들어 유지보수성과 아키텍처 호환성의 중요도가 각각 0.3과 0.2이라 하자. 이를 식(2)의 적용 결과와 함께 이용하면 계층 내에서 각 요소의 상대적 중요도를 다음과 같이 간단히 구할 수 있다.

$$\begin{pmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \end{pmatrix} = 0.3 \times \begin{pmatrix} 0.3570 \\ 0.3860 \\ 0.1570 \\ 0.1000 \end{pmatrix} = \begin{pmatrix} 0.1071 \\ 0.1158 \\ 0.0471 \\ 0.0300 \end{pmatrix}$$

$$\begin{pmatrix} ac_1 \\ ac_2 \end{pmatrix} = 0.2 \times \begin{pmatrix} 0.5700 \\ 0.4300 \end{pmatrix} = \begin{pmatrix} 0.1140 \\ 0.0860 \end{pmatrix}$$

AHP를 통한 모듈과 ANP를 통한 모듈로 평가 구조도내의 모든 요소에 대한 상대적 중요도가 산출된다. 구조도 최하위 각 요소에 대하여 컴포넌트 대안의 우수성 (q_{ik})을 측정하여 식(1)을 적용하면 각 대안의 종합적 우수성 (Q_k)이 산출되어 최적의 대안을 선택할 수 있다.

5. 결론 및 향후 연구방향

본 연구에서는 최근 실현되고 있는 CBS 유지보수와 관련하여 중요성 높은 컴포넌트 평가 및 선택 모형을 제시하였다. 이를 위하여 평가 기준을 계층적으로 체계화하였고 계층 내부에 기술적 사항을 고려하여 네트워크 모형이 포함되도록 하였다. 이를 위한 적절한 분석 기법으로 AHP 및 ANP를 병용하였다.

본 연구의 의의 및 향후 연구방향은 다음과 같이 사료된다. 첫째, 소프트웨어 유지보수의 패러다임(paradigm) 전환의 필요성을 제시하였다. 이는 CBS의 특성에 따라 소프트웨어 유지보수 활동 및 의사결정의 내용에 근본적인 변화가 요구됨을 의미한다. 둘째, 이러한 변화에 대한 대책의 일부를 본 연구가 제시하였다. CBS 유지보수의 핵심사항인 컴포넌트 평가를 위한 체계적인 기준과 함께 적절한 기법을 선택하고 기법의 활용 예를 제시하여 활용 가능성을 높였다. 셋째, 향후 연구 방향과 관련하여 본 연구의 프로세스를 정형화하면 유지보수 프로세스 표준화에 기여할 것으로 기대된다. 이를 통하여 프로세스의 지속적 개선 및 비용 절감에 기여할 것으로 판단된다.

참고 문헌

- [1] 윤민석, 이영, 성삼경, "유지보수성 목표하의 소프트웨어 개발방법 평가에 관한 실증 연구 : ANP 기법을 중심으로," 「한국경영과학회지」, 제24권 4호, 1999.
- [2] 이상석, 윤민석, "통계처리용 소프트웨어 패키지의 품질 비교에 관한 연구," 「품질경영학회지」, 제27권 1호, 1999.
- [3] Aoyama, M., "New Age of Software Development : How CBSE Change the Way of Software Development," *Proc. of 1st International Workshop on CBSE*, 1998.
- [4] Brown, A.W. and K.C. Wallnau, "The Current State of CBSE," *IEEE Software*, September/October, 1998.
- [5] Brownstein, I. and N.B. Lerner, *Guidelines for Evaluation and Selecting Software Packages*, Elsevier Science Pub., New York, 1982.
- [6] Davis, F.D., "Perceived Usefulness, Ease of Use, and User Acceptance of Information Technology," *MIS Quarterly* Vol. 13, No. 3, 1989.
- [7] Fritz, C.A. and B.D. Carter, *A Classification, and Summary of Software Evaluation and Selection Methodologies*, Computer Science Technical Report No. 940823, Mississippi State University, 1994.
- [8] Han, J., "Characterization of Components," *Proc. of 1st International Workshop on CBSE*, 1998.
- [9] ISO 9126, *Information Technology - Software Product Evaluation-Quality Characteristics and Guidelines for Their Use*, 1991
- [10] Jung H.W. and M.S. Yoon, "A Software Product Quality Evaluation and Resource Allocation Model," *Proc. 5th European Conference on Software Quality*, 1996.
- [11] Kekre, S., M.S. Krishnan and K. Srinivasan, "Drivers of Customer Satisfaction for Software Products," *Management Science*, Vol. 41, No. 9, 1995.
- [12] Kim, B. O., "Component-Based ERP Design in a Distributed Object Environment," *Proc. of 2nd International Workshop on CBSE*, 1999.
- [13] Kontio, J. "A Case Study in Applying a Systematic Method for COTS Selection," *Proc. of the 18th International Conference on Software Engineering*, 1996.
- [14] Saaty, T. L., *The Analytic Hierarchy Process* McGraw-Hill, 1980.
- [15] Saaty, T. L., *The Analytic Network Process* RWS Publications, 1996.
- [16] Subramanian, G.H. and M. Gershon, "The Selection of CASE tools: A MCDM Approach," *Decision Science*, Vol. 22, No. 5, 1991.
- [17] Vigder, M.R., "Building Maintainable CBS," *Proc. of 2nd International Workshop on CBSE*, 1998.
- [18] Yacoub, S., H. Ammar, and A. Mili "Characterizing a Software Component," *Proc. of 2nd International Workshop on CBSE*, 1998.
- [19] Yoon, M. S., *Software Quality Evaluation Model Using The AHP - Developing a New Judgment Aggregation Method*, Ph. D. Dissertation, Graduate School of Korea University, 1997.
- [20] Zhedi, F., "DBMS Evaluation and Selection Decisions," *IEEE Transactions on Software Engineering*, Vol. 14, No. 9, 1985.
- [21] Zsyperski, C., *Component Software*, Addison-wesley, New York, 1999.