



한국정보과학회

제1권 제1호  
Vol. 1 No. 1



2005

# 한국 소프트웨어공학 학술대회 논문집

Proceedings of 2005 Korean Conference on  
Software Engineering

- 일시 : 2005년 2월 21일(월)~23일(수)
- 장소 : 무주리조트

주최 : 한국정보과학회, 한국정보처리학회

주관 : 한국정보과학회 소프트웨어공학연구회  
한국정보처리학회 소프트웨어공학연구회  
한국전자통신연구원

후원 : 한국소프트웨어진흥원  
KAIST 소프트웨어프로세스 개선센터  
건국대 소프트웨어 연구센터  
(주)n3soff

◆ 논문 목차 ◆

<b>튜토리얼 (2월 21일)</b>	
· 온돌로지 공학의 연구 동향 및 적용 사례 김홍기 교수 (서울대 치과대학)	3
· 정보보증을 위한 소프트웨어 공학 적용 사례 및 연구 필요성 김태훈 사무관 (국군기무사령부)	26
<b>주제 1: 산업체 I, II (2월 21일)</b>	
· 기능 관점의 임베디드 소프트웨어 아키텍처 프레임워크 설계 방법 오광근, 문건일 (LG산전연구소), 박수용 (서강대)	43
· Embedded SW에서 SW Architecture 검증 방법에 관한 연구 박복남 (삼성전자)	56
· 국방 정보 시스템 간의 상호운용을 위한 LSI 기반 체크리스트 연구 류동국 (국방과학연구소), 김우열, 김영철 (홍익대)	67
· 초소형 저전력 센서 모듈을 위한 운영체제의 구조 및 설계 황대호, 송병철 (전자부품연구원), 조위덕 (아주대)	75
· 전자정부참여광장을 통한 전자민주주의 실현 신영진, 이종현 (행정자치부)	84
· J2EE 환경에서 DB2 for z/OS 성능 향상 방안 연구 최병석 (고려대)	92
<b>주제 2: 테스트 I, II (2월 22일)</b>	
· Function Block Diagrams에 대한 제어 데이터 흐름 테스트 지은경, 유준범, 박수현, 차성덕 (한국과학기술원)	105
· 소프트웨어 신뢰도 평가 대상의 파라미터 연구 최규식 (건양대학교)	115
· 개방형 서버의 상호운용성 테스트를 위한 테스트 케이스 생성 이지현, 노혜민, 유철중, 장옥배 (전북대), 이준욱 (한국전자통신연구원)	122
· 임베디드 소프트웨어 벤치마킹을 위한 BMT 모델 최현미, 성아영, 최병주 (이화여대), 신석규 (한국정보통신기술협회)	136
· 패킷 필터링 기능 테스트를 위한 테스트베드 구축 박영대, 국순학, 김현수 (충남대)	145
· 테스트 프로세스 향상을 통한 테스트 성숙도 모델(TMM) 개선에 관한 연구 김기두, 김영철 (홍익대), 박병호 (국군의무사령부)	155
<b>주제 3: 품질관리 I, II (2월 22일)</b>	
· 국내 중소 SW 기업을 위한 품질관리 현황 파악 모델 최순황, 이창병, 박수용 (서강대), 황만수 (신홍대), 신재식, 문장원 (소프트웨어진흥원)	165
· BlueJ 기반 객체지향프로그래밍 교육 및 품질평가 도구에 관한 연구 강유경, 황석형 (선문대), 양해술 (호서대)	180
· ISO/IEC 9126기반의 모바일 게임 소프트웨어 시험을 위한 품질 특성 및 평가항목 최재현, 이우진, 정기원 (숭실대)	186
· Weakest Precondition 함수를 이용한 형상관리 프로세스 활동 수행의 타당성 검증 방법 제안 장옥현, 이병걸 (서울여대)	195
· 영상처리 어플리케이션 자동 생성에 필요한 개발자 지식 추출을 위한 소프트웨어 형상관리 방법 이정현 (자르테크), 채옥삼 (경희대)	207
· 대규모 시스템을 위한 ISO/IEC 12207 적용성 및 확장성에 대한 연구 성민숙, 강태현, 백인섭 (아주대)	220

## 국방 정보 시스템 간의 상호운용을 위한 LISI 기반 체크리스트 연구

류동국\*, 김우열, 김영철

국방과학연구소\*

서울 송파구 가여동 산 25번지

홍익대학교 컴퓨터정보통신

충남 연기군 조치원읍 신안리 300

dkryu@lycos.co.kr\*, (johu, bob)@hongik.ac.kr

**요약** : 시스템간의 상호운용은 국방, 금융, 행정 등 임무 중심적인 시스템의 개발 및 운용에 있어서 많은 관심과 연구가 되고 있는 분야이다. 근래에 들어 국방 정보 시스템도 컴포넌트를 기반으로 개발되고 있으며 이들 시스템간의 상호운용성을 확보하기 위하여 많은 연구가 진행되고 있다. 본 논문에서는 상호운용성 수준을 측정하기 위한 LISI 모델을 이용하여 국방 정보 시스템간의 상호운용성 향상을 위한 체크리스트를 제시한다. 상호운용 체크리스트는 국방 컴포넌트 개발 방법론상에서 개발의 각 단계마다 상호운용을 위하여 고려하여야 할 요소들을 도출하여 평가항목으로 작성한 것이다. 본 논문에서는 체크리스트를 검증하기 위하여 웹서비스 방식의 컴포넌트 시스템을 개발하여 상호운용성 수준을 평가 하였다.

**핵심어** : 상호운용성(Interoperability), LISI(Level of Information System Interoperability), 체크리스트(Checklist)

### 1. 서 론

정보 시스템이 유기적으로 상호 연동됨에 따라 시스템간의 상호운용성의 확보가 매우 중요한 부분이 되고 있다. 특히 국방, 행정, 금융 등 이질적이고 일부중심적인 시스템들 간의 상호 연동이 필요한 분야에서는 시스템의 개발에 있어서 상호운용성은 매우 중요한 고려 요소이다. 예를 들어 미군은 걸프 전 및 이라크 전을 수행하면서 컴퓨터 게임을 하듯이 전쟁을 수행하였다. 전투 지휘관은 컴퓨터 화면을 보면서 전 전장의 상황을 파악하고 컴퓨터 통신을 통하여 명령을 전달하였다. 이러한 전투 환경이 가능하게 된 것은 미군이 사용하는 여러 정보 시스템들이 유기적인 상호운용을 통하여 효율적인 정보 공

유 및 교환이 가능하기 때문이다. 만약 정확하고 효율적인 상호운용이 가능하지 않고 상호운용 과정에 오류가 발생하여 잘못된 정보가 전달되게 된다면 무기 시스템의 오작동 등 심각한 문제점이 발생할 수 있다. 따라서 임무 중심적이고 여러 복합체계가 유기적으로 운용되는 환경에서는 정보 시스템의 상호운용성은 아주 중요한 고려 요소가 된다 [1].

그러나 정보 시스템의 개발 기술 및 운용 환경은 복잡하고 다양하여 이질적인 환경에서 시스템간의 상호운용은 점점 어려워지고 있다. 이러한 상호운용의 문제점을 인식하고 이를 해결하기 위해 많은 연구비를 지원하여 연구하고 있는 곳은 미국 국방성이다. 미군은 DoD AF(Architecture Framework)와 LISI(Level of Information System Interoperability)를 개발하여 정보 구조적 측면에서 상호운용성을 측정하는 평가 모델을 개발하여 사용하고 있다 [1,2]. LISI는 CMM(Capability Maturity Model)을 개발한 Carnegie Mellon 대학의 SEI 연구소에서 개발하였으며, CMM의 성숙도 측정 모델을 참고하여 정보시스템의 상호운용성 수준을 측정하는 평가 모델이다 [3].

본 논문에서는 국방 정보 시스템의 상호운용성을 높이기 위한 LISI 기반의 체크리스트를 제시한다 (이하 상호운용 체크리스트). 상호운용 체크리스트는 LISI를 이용한 상호운용 수준의 향상을 위하여 국방 컴포넌트 개발 방법론(ADDME, Advanced Defense Component Development Methodology)의 각 단계마다 적용하여 상호운용성 수준 향상을 위한 점검사항을 제시한다 [4]. 상호운용 체크리스트는 국방 컴포넌트 개발 방법론을 이용

하여 개발하는 과정에서 어떠한 상호운용 고려요소가 있는지를 개발자 및 관리자에게 제공하여 최종적으로 개발되는 시스템의 상호운용성을 향상시키는 것을 목표로 하고 있다. 상호운용 체크리스트의 검증은 위하여 본 논문에서는 웹 서비스 방식의 컴포넌트 기반 시스템을 개발하여 상호운용성을 평가하고 결과를 비교 분석하였다.

의서를 통하여 수집된 정보는 LISI 평가 기반에서 정의된 상호운용 성숙도 모델, 참조 모델, 능력 모델, 구현 옵션 테이블을 이용하여 LISI 평가 프로세스를 통하여 상호운용 프로파일, 상호운용 매트릭스, 상호운용 행렬, 비교 테이블, 구조 산출물 등의 LISI 평가 산출물을 생성하게 된다. 구성요소에 대한 자세한 설명은 다음과 같다.

본 논문의 구성은 다음과 같다. 2장 관련연구에서는 상호운용 평가모델인 LISI와 본 연구에서 적용한 국방컴포넌트 개발 방법론을 상호운용성 관점에서 살펴본다. 3장에서는 컴포넌트 기반 시스템의 LISI 기반 평가를 위하여 확장된 LISI 능력모델을 제시한다. 4장에서는 상호운용 체크리스트를 설명하고, 5장에서는 프로토타입을 이용한 체크리스트 사례 분석 결과를 기술하였다. 마지막으로 6장에서 본 연구의 결론과 향후과제에 대하여 논하였다.

## 2. 관련 연구

정보 시스템의 상호운용성은 미 국방부의 DISA(Defense Information System Agency)에서 가장 연구가 활발히 진행되고 있다. 이 장에서는 미 국방부에서 상호운용성 평가를 위해 사용하는 LISI(Level of Information Interoperability)를 소개한다. 그리고 본 연구에서 적용한 ADDME 방법론의 개발 과정 및 산출물을 상호운용 관점에서 살펴본다.

### 2.1 LISI

LISI는 1998년에 CMM(Capability Maturity Model)을 개발한 카네기 멜론대학의 SEI 연구소에서 정보 시스템의 상호운용성을 측정하기 위한 평가 모델로 개발되었다. LISI는 성숙도 모델에서의 수준 개념을 이용하여 정보시스템의 상호운용 능력을 평가 한다.

#### 2.1.1 LISI 구성요소

그림 1은 LISI의 구성요소를 나타낸다. LISI는 상호운용성을 평가하는 기반환경이라 할 수 있는 LISI 평가 기반과 이를 활용하여 평가 하는 평가 프로세스 그리고 평가 결과에 해당하는 LISI 평가 산출물로 구성된다. LISI에 의한 상호운용 평가는 먼저 구조화된 상호운용 질의서를 이용하여 평가 대상 시스템의 정보를 수집한다. 질

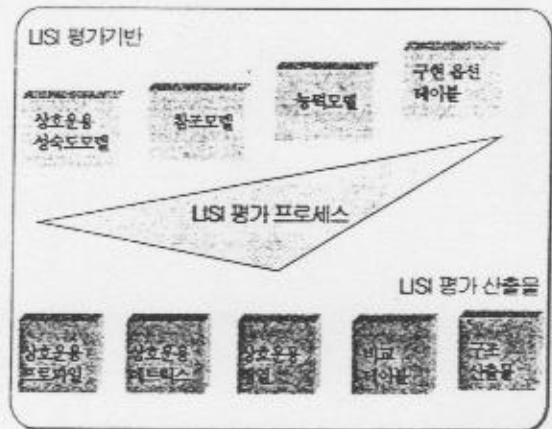


그림 1 LISI 구성요소

평가지준	정보교환	상호협력	자료&응용관계	컴퓨터환경
5 전사적	전사적 공유자료	협력기성공간	자료: 공유 응용: 공유	
4 도메인	공유된 데이터베이스	진보된 상호협동	자료: 공유 응용: 개별	
3 개능적	어종의 자료	복잡한 상호협동	자료: 개별 응용: 개별	
2 연결	동종의 자료	기본	자료: 개별 응용: 개별	
1 실연전	수동적 교환	인간	자료: 개별 응용: 개별	
0 크리	없음	없음	자료: 개별 응용: 개별	

그림 2 상호운용 성숙도 모델

• 상호운용 성숙도 모델 : LISI 프로세스는 시스템 간 상호작용의 복잡도 수준에 대하여 시스템을 정의하기 위하여 시스템의 상호운용 능력의 경계를 정의하였다 (그림 2 참조). 이 경계가 시스템의 생명주기 전반에 걸쳐 측정되는 상호운용성 수준이 된다.

서정의  
구현  
동하여  
행열,  
확생성  
과 같다.

Description		Environment Level		P		A		I		D	
Enterprise	External	4	Enterprise Level	Interactive	Multi-Dimensional Topologies	Enterprise Model					
Domain	Integrated	3	Domain Level	Groupware	Web-based Networks	Domain Model					
Functional	Distributed	2	Program Level	Desktop Automation	Local Networks	Program Model					
Operational	Peer-to-Peer	1	Local/Intra Level	Standard System Definition	Simple Connections	Local Model					
Isolated	Stand-alone	0	Appar. Class	N/A	Independent	Primitive					

그림 3 LISI 참조 모델

• 참조 모델 : LISI 참조 모델은 시스템 혹은 시스템간의 상호운용성 수준을 결정하기 위한 공통적인 평가 기반이다 (그림 3 참조). 참조 모델을 이용하여 시스템의 정보를 수집, 분석, 측정 및 비교할 수 있다. 참조 모델에서는 특정한 설계를 위한 일련의 공통적인 서비스와 인터페이스를 정의한다.

• 능력 모델 : LISI 참조 모델은 일반적이고 단순화된 비교로 상호운용성 수준 및 특성에 관련된 세부사항은 제공하지 않는다. 따라서 각 수준별 세부적인 특성을 PAID 항목으로 표현하기 위한 세부적인 모델이 필요한데 이러한 확장을 LISI 능력 모델이라 한다 [5].

• 구현 옵션 테이블 : LISI 능력 모델의 능력 사항을 구현하기 위하여 가용한 구현 선택사항을 포함한다 [5].

### 2.1.2 LISI 측정 프로세스

LISI의 상호운용 측정 프로세스는 그림 4와 같다. 그림에서 1단계와 2단계를 상호운용성 평가를 위한 지침, 표준 평가 모델을 만드는 단계이고, 일반적인 상호운용성 평가는 3단계에서 6단계까지를 포함한다. 평가 단계별 내용은 다음과 같다.

- ① 상호운용에 대한 정의, 표준, 지침 제정하여 상호운용 측정 환경을 정립한다.
- ② LISI 성숙도 모델, 참조 모델, 구현 옵션을 개발하여 상호운용성 측정을 가능하게 한다.
- ③ 상호운용 질의서에 개발 시스템에 대한 구현 정보(구현 옵션)를 입력한다.
- ④ 질의서에 입력된 정보를 바탕으로 평가 대상 시스템의 상호운용 능력에 관련된 정보를 종합한 상호운용 프로파일을 작성한다.

⑤ 상호운용 프로파일을 바탕으로 상호운용성을 평가하고 평가 결과를 저장한다.

⑥ 더욱 향상된 상호운용 수준을 확보하기 위하여 시스템의 구현 옵션을 조정하고 시스템 개발에 반영한다.

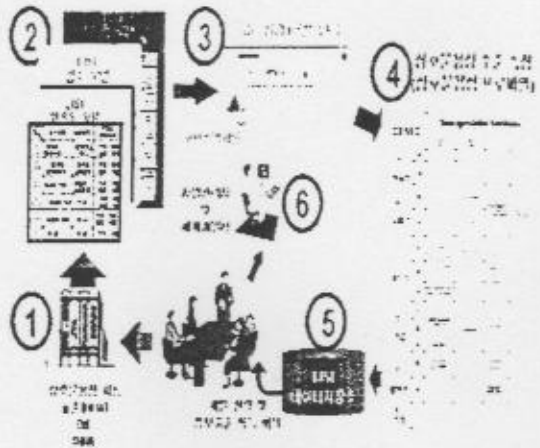


그림 4 상호운용성 측정 프로세스

### 2.1.3 상호운용 체크리스트 도출

상호운용 체크리스트를 작성하는데 있어서 LISI 능력 모델과 구현 옵션 등을 참조하였다. LISI 능력 모델은 절차, 응용 프로그램, 기반 구조, 데이터로 평가 분야를 세분하고 각각의 분야별로 상호운용 수준을 정의한다. 본 연구에서는 상호운용의 분야와 평가 결과(상호운용 수준)에 LISI 모델을 활용하였다. 구현 옵션은 상호운용에 관련된 표준, 제품 등을 목록화한 것이다. 구현 옵션 또한 체크리스트의 구체적인 질의 내용을 작성하는데 활용하였다.

## 2.2 국방 컴포넌트 개발 방법론

### 2.2.1 개요

국방 컴포넌트 개발 방법론은 국방 분야에서 사용되는 컴포넌트를 획득하기 위한 표준 방법론으로 2004년에 제정되었다. 산업 표준인 UML(Unified Modeling Language) 기반의 CBD 방법론에 기초하여, 분석, 설계 구현 과정의 산출물을 표준화하였다. 국방 컴포넌트 개발 방법론은 카탈리시스, 어드바이저, RUP(Rational Unified Process) 그리고 마르미-III 등 기존의 CBD 방법

본들의 장점을 종합하여 국방 환경에 적합한 방법론으로 개발하였다.

### 2.2.2 구성요소 및 관련 문서 구조

국방 컴포넌트 개발 방법론은 그림 5와 같이 단계(Phase), 활동(Activity), 작업(Task)으로 구성된다. 단계, 활동, 작업은 단계 설명서, 활동 설명서, 작업설명서 및 기법서에서 구체적으로 정의된다. 또한 각 작업의 결과는 산출물이 생성된다. 국방 컴포넌트 개발 방법론은 6 단계, 20활동, 50작업 그리고 총 43종의 산출물로 구성된다. 각 문서에 대한 설명은 다음과 같다.

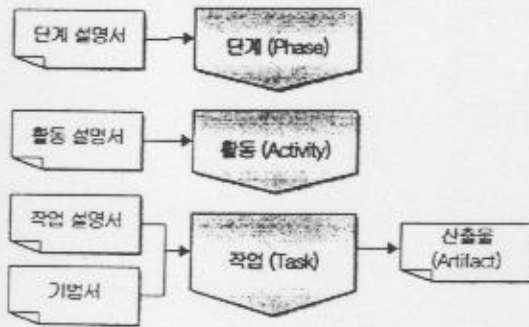


그림 5 관련 문서 구조

- 단계 설명서 : 해당 단계에 대한 개요 설명과 활동 흐름도를 나타낸다. 단계에 속한 활동에 대한 설명과 활동간의 관계 정보가 포함된다.
- 활동 설명서 : 해당 활동에 대한 개요 설명과 작업 흐름도를 나타낸다. 활동에 속한 작업에 대한 설명과, 활동을 수행하기 위한 입력력 목록을 보여준다.
- 작업 설명서 : 해당 작업에 대한 설명과 작업을 수행하기 위한 세부 절차의 흐름도 및 설명을 나타낸다.
- 산출물 : 작업의 결과로 생성된다. 작성자 및 작성 이력, 산출물을 작성하는데 필요한 입력물 관련 기법이 표기된다. 산출물별로 정해진 양식에 따라 작성한다.
- 기법서 : 산출물 작성에 참고하는 문서로 특별한 기법에 대한 설명이다. 예를 들어 UML 다이어그램을 작성하는 방법을 설명한 UML 모델링 기법이나 데이터베이스 설계 시 컴포넌트와의 연동을 위한 기법 등을 설명한다.

### 2.2.3 개발 절차

국방 컴포넌트의 개발 절차는 그림 6과 같다. 총 6개의 단계로 이루어져 있으며, 분석, 설계, 구현의 점진적인 반복을 통하여 개발된다. 분석 위주의 반복에서는 도메인/비즈니스 분석에 중점을 두어 수행한다. 설계 위주의 반복에서는 모델 주도형 아키텍처 기반 컴포넌트 아키텍처 설계, 컴포넌트 내부 설계, 시스템 설계, 테스트 및 구현 단계를 중점적으로 수행한다. 그리고 분석 단계에서의 도메인/비즈니스 분석이나 요구사항 분석을 변경 보증한다. 마지막으로 구현 위주의 반복에서는 이전 단계에서 수행한 단계들을 정제하여 완성하는 작업을 수행한다.

	분석	설계	구현
도메인/비즈니스 분석	S(Start)	R(Refine)	
요구사항 분석	S	R	
모델 주도형 아키텍처 기반 컴포넌트 아키텍처 설계		S	R
컴포넌트 내부 설계		S	R
시스템 설계		S	R
테스트 및 구현		S	R

그림 6 개발 절차 구조

### 2.2.4 상호운용 체크리스트 도출

본 연구에서는 국방 컴포넌트 개발 방법론에서 정의한 구성 요소 및 문서를 바탕으로 상호운용 체크리스트를 도출하였다. 국방 컴포넌트 개발 방법론에서 정의한 각각의 단계, 활동, 작업에 있어서 산출물을 생성하는 과정에서 이중 컴포넌트의 상호운용에 필요한 고려 요소를 체크리스트로 작성하였다.

### 3. LISI 능력 모델의 확장

상호운용 체크리스트는 국방 컴포넌트 기반 시스템의 상호운용성 수준 평가 및 향상을 주 목적으로 한다. 상호운용성 평가의 대상이 국방 컴포넌트이므로 본 논문에서는 기존의 LISI 모델을 확장하여 컴포넌트의 상호운용성 평가를 위한 LISI 능력모델을 제시한다 [6].

### 3.1 기존 LISI의 문제점

LISI 평가 모델은 미군이 사용하는 상호운용성 평가 모델이다. 현재 공개된 LISI 모델은 1998년에 개발되었다. LISI 평가 기반인 능력모델, 구현옵션에는 컴포넌트 개발 방식으로 구현되는 시스템의 평가에 대한 고려가 없다 [1]. 우리나라에서는 국방부에서 이 LISI 모델을 개정한 국방 LISI 01 능력 모델을 사용하고 있으나 역시 최신 컴포넌트 기반 시스템에 대한 고려는 적은 실정이다 [5].

### 3.2 새로운 능력모델의 필요성

상호운용성은 다양한 정보체계를 개발하여 운영하는 조직에서 중요하게 고려되어야 하는 요소이다. 현재 많은 조직에서는 CBD 기반의 소프트웨어를 개발하여 사용하고 있다. 컴포넌트는 재사용이 가능하므로 개발 비용의 절감, 적시성의 확보 등의 장점이 있다. 그러나 기존의 LISI로는 컴포넌트 방식으로 개발되는 시스템에 대한 고려가 없으므로 CBD 방식의 특성을 고려한 평가를 할 수 없다. 컴포넌트 기반 시스템의 상호운용 능력을 평가하기 위해서는 제일 먼저 컴포넌트 기반 시스템에 적용 가능한 상호운용 능력모델을 개발하여야 한다.

### 3.3 컴포넌트 상호운용 능력 모델

표 1 컴포넌트 상호운용 능력 모델

상호운용수준 범위	상호운용환경	C (Component)		설명
		상호운용 (Interoperable)	독립 (Independent)	
Enterprise	Universal	b	전 세계적 상호운용	-통합 환경의 컴포넌트 상호운용 가능
		a	국가적 상호운용	
Domain	Integrated	b	도메인 간 상호운용	-도메인 컴포넌트 상호운용 가능
		a	도메인 내부 상호운용	
Functional	Distributed	b	컴포넌트 상호운용 기술적 방안 제시	-컴포넌트 간 상호운용 가능 -상호운용 규약 및 절차 부재

Connected	Peer to Peer	독립 (independent)	a	컴포넌트 규약 및 절차 정의 -운용환경	컴포넌트 모델 종속
			b	독립	
			a	개발 언어, 운용환경 의존	
			b	운용환경 의존	
Isolated	Manual	의존 (dependent)	b	운용환경 의존	-개발환경 및 운용환경에 의존
			a	개발환경 의존	

컴포넌트 기반 시스템 상호운용 능력 모델은 기존의 LISI 능력모델의 PAID 특성 필드에 C필드, 즉 컴포넌트 필드를 추가한 것이다. 표 1과 같이 C필드의 단계는 5단계로 정의된다. C필드는 컴포넌트를 컴포넌트의 특성에 따라 상호운용 능력을 LISI 성숙도 모델의 능력 단계에 따라 정의한 것이다.

본 논문에서는 컴포넌트의 성숙도를 의존, 독립, 상호운용, 도메인, 통합 컴포넌트로 분류 하였다. 각 단계는 세부적인(a,b) 단계로 분류할 수 있으나 본 논문에서는 5개의 대분류에 대해서만 기술한다.

- 의존 컴포넌트 : 가장 상호운용 수준이 낮은 단계이다. 의존 컴포넌트는 개발 언어, 운용환경 등에 의존적인 컴포넌트로 특정 개발언어, 운영체제, 통신 인프라에서만 동작하는 단계이다.
- 독립 컴포넌트 : 독립 컴포넌트는 EJB나 .Net과 같이 특정 컴포넌트 모델이 정의되어 있어 모델에서 정의하는 규격을 만족하는 컴포넌트 간에 상호운용이 가능한 컴포넌트이다. 그러나 서로 다른 컴포넌트 모델 간에는 상호운용이 불가능하다.
- 상호운용 컴포넌트 : 상호운용 컴포넌트 단계에서는 컴포넌트 모델에 제약이 없이 모든 컴포넌트가 컴포넌트 기술적으로는 상호운용이 가능한 단계이다. 그러나 각 컴포넌트간의 상호운용에 대한 세부적인 규약이나 절차는 정의되어 있지 않는 단계이다.
- 도메인 컴포넌트 : 도메인 컴포넌트는 컴포넌트를 공유하는 도메인에서 컴포넌트 표준을 개발하여 제공하므로 도메인에서 상호운용이 가능한 단계이다.
- 통합 컴포넌트 : 통합 컴포넌트는 컴포넌트의 표준이 정의되고 모든 컴포넌트 사용자에게 표준으로 개발되어 범 국가적 또는 전 세계적으로 공통 사용이 가능한 단계이다.

다. 총 6개의 점진적에서는 도 설계 위주 포넌트 아 네, 테스트 분석 단계 석을 변경 이진 단 령을 수행

구현
R
R
R
R

정 의 한 리스 트를 의 한 각 는 과 장 요 소 들

이 스 템 의 간 다. 상 논 문 에 호 운 용

4. 상호운용 체크리스트

이 장에서는 상호운용 국방 컴포넌트 개발 방법론 상에서 체크리스트 도출과정 및 체크리스트의 구조에 대하여 설명한다.

4.1 국방 컴포넌트 개발 방법론 단계별 상호운용 요소 식별

상호운용 체크리스트는 국방 컴포넌트 개발 방법론의 각 단계를 분석하여 필요한 상호운용 고려요소를 도출하였다. 예를 들어 국방 컴포넌트 개발 방법론의 첫 번째 단계인 도메인/비즈니스 분석 단계는 명확한 시스템 개발을 위하여 도메인 및 비즈니스를 도식화하는 단계이다. 자세한 활동 흐름은 그림 7과 같다. 이 단계에서 상호운용에 중요한 문서는 비즈니스 조직 흐름 모델 명세서와 비즈니스 프로세스 흐름 명세서이다. 이 단계에서는 이들 문서에서 상호운용이 요구되는 외부 체계가 식별된다. 또한 어떠한 비즈니스 흐름이 존재하는지 알 수 있다.

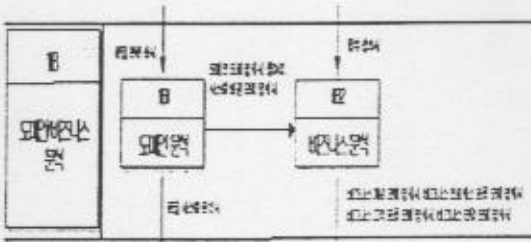


그림 7 도메인 비즈니스 분석 단계 활동 흐름

4.2 상호운용 체크리스트의 구조

표 2는 설계 단계에서의 상호운용 체크리스트 예이다. 체크리스트는 개발단계, 관련문서, 개발 절차 흐름, 관련 문서 세부 항목 그리고 질의로 구성되어 있다. 이러한 세부 항목은 국방 컴포넌트 개발 방법론의 구성 요소를 참조하여 질의서 응답하는 구조이다. 그리고 해당하는 LISI 특성 값을 참조하여 LISI 평가를 하게 된다. 상호운용 체크리스트는 상호운용의 평가 방식에 따라 개별 시스템의 상호운용 능력을 평가하는 질의와 시스템간의 상호운용성을 평가하는 질의를 포함하여 모두 1,179개의 질의를 작성하였다 [7].

5. 상호운용 체크리스트 측정 사례 분석

이 장에서는 본 논문에서 제시한 컴포넌트 기반 시스템 상호운용 측정 기법의 적합성을 시험하기 위하여 먼저 웹 서비스를 설명하고 프로토타입 형태의 컴포넌트 기반 시스템을 개발하여 평가하고 결과를 분석하였다. 상호운용 체크리스트는 컴포넌트 개발 과정에 상호운용 고려요소를 제시한다. 이때 측정값을 LISI 평가에 적용하여 상호운용 수준 평가를 수행한다.

5.1 웹서비스를 이용한 상호운용

본 절에서는 웹 서비스 기술을 이용한 컴포넌트 기반 시스템 간의 상호운용 기술을 설명한다. 웹 서비스는 웹 기술의 발전된 형태이다. 웹 서비스는 웹 기술에 XML을 이용한 분산 컴퓨팅이라 할 수 있다. 웹 서비스에서 제공하는 서비스는 어플리케이션, 프로세스 또는 컴포넌트 등이

표 2 상호운용 체크리스트

개발 단계	관련 문서	개발 절차 흐름	관련 문서 세부항목	질의	LISI 특성	측정값
설계 (분석 단계)	컴포넌트 명세서	인터페이스 명세	인터페이스 명세서	-식별된 컴포넌트와 인터페이스의 표준 명세화가 되었는가?	P(1)	
			컴포넌트 명세서	-국가 정보 기반 체계개발에서의 표준 컴포넌트 명세화가 되었는가?	P(5b)	
			컴포넌트 명세서	-전서적 표준과 국방 정보 기술 구조 개발에서의 표준 컴포넌트 명세화가 되었는가?	P(26/c)	
			컴포넌트 명세서	-DBMS와 같은 응용 어플리케이션을 통해 공통의 데이터 접근에 대한 컴포넌트 인터페이스를 명세화 하였는가?	A(5)	
		컴포넌트 명세	컴포넌트 명세서	-공통의 데이터도구를 통해 다른 응용과 데이터를 공유에 대한 컴포넌트 인터페이스를 명세화 하였는가?	A(4b)	



트 예이다. 흐름, 관련성, 이러한 상호요소를 담당하는 상호운용 개별 시스템간의 상 1,179개의

기반 시스템을 위하여 먼 컴포넌트 석하였다. 상호운용 세 적용하

기반 시스템은 웹 기술이 제공하 동이

속경관

계약 없이 될 수 있다. 그러나 서비스를 이용하는 측면에서는 서비스의 내부가 어떠한 기술로 구현되었는지는 고려대상이 아니다. CORBA, RMI, DCOM과 같은 기존의 분산 컴퓨팅 기술 대신에 웹 서비스를 사용하는 이유는 웹이 이용하기 편리하고 어디에서나 손쉽게 사용할 수 있기 때문이다. 또한 웹 서비스는 이종 플랫폼간의 연동을 지원하고 운용 플랫폼 및 개발 언어에 독립적이다. 웹 서비스의 세부 요소 기술은 다음과 같다.

- SOAP : XML에 기반을 둔 플랫폼 독립적이며 간단한 통신 프로토콜
- WSDL : 웹 서비스의 인터페이스 정보와 연결점 정보를 기술하는 언어
- UDDI : 서비스 검색을 위한 등록 디렉토리

### 5.1.1 웹 서비스 운용 절차

웹 서비스의 운용 절차는 그림 8과 같다. 우선 서비스를 제공하는 컴포넌트를 UDDI에 등록한다. 그리고 서비스를 요청하는 컴포넌트는 UDDI에서 서비스를 검색하여 WSDL로 서비스 컴포넌트의 수행에 필요한 정보를 제공받는다. 제공받은 WSDL을 수행하면 SOAP을 통하여 서비스 컴포넌트를 수행하고 결과를 받는다.

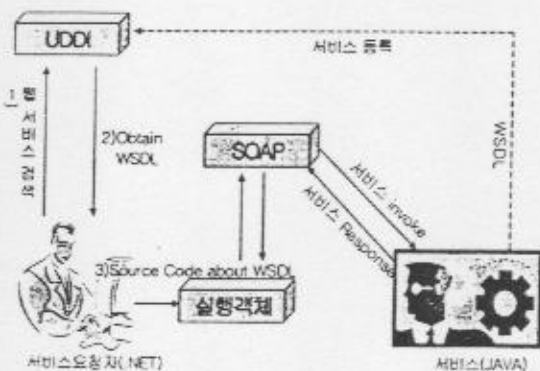


그림 8 웹 서비스 운용 절차

### 5.2 프로토타입 개요

본 논문에서는 각각 EJB와 .Net으로 구성된 두개의 컴포넌트를 개발하였다. 그리고 개발된 컴포넌트를 조합하여 웹 서비스를 이용한 컴포넌트 기반 시스템을 개발하였다.

### 5.3 프로토타입 구조

EJB와 .Net 컴포넌트간의 상호운용성을 웹 서비스 기술을 이용하여 구현하였다. 서비스를 제공하는 역할을 하는 서버 컴포넌트와 서비스를 제공 받는 클라이언트 컴포넌트의 구현기술은 다음과 같다.

- 서버 컴포넌트 : EJB 기반
- 클라이언트 컴포넌트 : .Net 기반

### 5.4 웹 서비스를 이용한 구현

웹 서비스를 이용해서 서버 컴포넌트와 클라이언트 컴포넌트가 파일을 전송하는 시나리오로 구현하였다. 그림 9는 웹 서비스를 이용한 EJB 서버 컴포넌트와 .Net 클라이언트 컴포넌트의 소스코드와 실행화면이다.

```
// 서버 EJB 소스의 일부분
// 클라이언트가 서버에게 파일을 전송한다.
// 전달되는 파일의 타입은 DataHandler이다.
public void setFile(DataHandler dh)
{
    try {
        fos = new FileOutputStream("FromClientFile.txt");
        dh.writeTo(fos);
        fos.flush();
    } catch
}
}
```

```
'클라이언트 .Net 소스의 일부분
WScript.Echo ("Create Client Succeeded ---")
'wsdl과 wsml을 이용하여 SoapClient 객체를 초기화한다.
call client.mssoapinit(wsdl, "", "", wsml)

WScript.Echo ("Init Client Succeeded -----")
'서비스를 호출한다.
Set ReceivedAttach = Client.getFile()
FileSaveName = "FromServerFile.txt"
'호출결과 반환받은 객체를 파일로 저장한다.
ReceivedAttach.SaveToFile(FileSaveName, True)
```

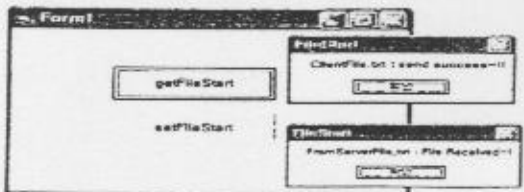


그림 9 웹 서비스를 이용한 코드 및 실행 화면

5.5 측정 결과

	응용 (Application)	중요 요소 (Requirement)	기반구조 (Infrastructure)	데이터 (Data)	컴포넌트 (Component)
응용	1. 사용자 인터페이스 2. 데이터베이스 3. 보안 4. 성능	1. 데이터베이스 2. 사용자 인터페이스 3. 보안 4. 성능	1. 데이터베이스 2. 사용자 인터페이스 3. 보안 4. 성능	1. 데이터베이스 2. 사용자 인터페이스 3. 보안 4. 성능	1. 데이터베이스 2. 사용자 인터페이스 3. 보안 4. 성능
중요 요소	1. 데이터베이스 2. 사용자 인터페이스 3. 보안 4. 성능	1. 데이터베이스 2. 사용자 인터페이스 3. 보안 4. 성능	1. 데이터베이스 2. 사용자 인터페이스 3. 보안 4. 성능	1. 데이터베이스 2. 사용자 인터페이스 3. 보안 4. 성능	1. 데이터베이스 2. 사용자 인터페이스 3. 보안 4. 성능
기반구조	1. 데이터베이스 2. 사용자 인터페이스 3. 보안 4. 성능	1. 데이터베이스 2. 사용자 인터페이스 3. 보안 4. 성능	1. 데이터베이스 2. 사용자 인터페이스 3. 보안 4. 성능	1. 데이터베이스 2. 사용자 인터페이스 3. 보안 4. 성능	1. 데이터베이스 2. 사용자 인터페이스 3. 보안 4. 성능
데이터	1. 데이터베이스 2. 사용자 인터페이스 3. 보안 4. 성능	1. 데이터베이스 2. 사용자 인터페이스 3. 보안 4. 성능	1. 데이터베이스 2. 사용자 인터페이스 3. 보안 4. 성능	1. 데이터베이스 2. 사용자 인터페이스 3. 보안 4. 성능	1. 데이터베이스 2. 사용자 인터페이스 3. 보안 4. 성능
컴포넌트	1. 데이터베이스 2. 사용자 인터페이스 3. 보안 4. 성능	1. 데이터베이스 2. 사용자 인터페이스 3. 보안 4. 성능	1. 데이터베이스 2. 사용자 인터페이스 3. 보안 4. 성능	1. 데이터베이스 2. 사용자 인터페이스 3. 보안 4. 성능	1. 데이터베이스 2. 사용자 인터페이스 3. 보안 4. 성능

그림 10 측정 결과

그림 10은 웹 서비스 방식을 이용하여 개발한 컴포넌트 기반 시스템에 대하여 본 논문에서 개발한 상호운용 체크리스트를 적용한 결과이다. 상호운용 체크리스트는 개발과정에서 개발자나 관리자로부터 하여금 상호운용성을 향상시키도록 할 수 있는 여러 점검사항을 제시하는 것을 주목적으로 하고 있다. 또한 상호운용 체크리스트는 체크리스트의 점검항목의 LISI 상호운용 수준을 제시하여 주므로 상호운용성 수준 평가 또한 가능하다.

웹 서비스 기반 시스템의 상호운용성 측정 결과들 요소 별로 살펴보면 절차 3c, 응용체계 5b, 기반구조 4b, 데이터 4a, 컴포넌트 3c이다. 상호운용 수준의 측정 결과 값은 가장 낮은 부분이 시스템의 상호운용 능력을 대표하므로 웹 서비스 기반 시스템은 3c에 해당한다 [1]. 따라서 전체적인 상호운용 수준은 3c가 된다. 결과로 살펴볼 때 보다 높은 상호운용 수준에 도달하기 위해서는 상호

운용 수준이 낮은 절차나 컴포넌트 부분의 상호운용 수준을 높이는 것이 효과적이다.

6. 결론

본 연구에서는 국방 컴포넌트 개발 방법론에 따른 컴포넌트 개발에 있어서 컴포넌트 기반 시스템의 상호운용성 향상을 위한 체크리스트를 개발하였다. 상호운용 체크리스트는 국방 컴포넌트 개발 방법론의 개발 과정의 각 단계에 적용하여 개발 단계마다 고려하여야 할 상호운용 요소를 지적하여 개발자로 하여금 상호운용성이 향상된 시스템의 개발을 가능하도록 도와준다.

향후연구로는 국방 컴포넌트뿐만 아니라 모든 컴포넌트 기반 방법론에 적용될 수 있도록 체크리스트의 보완과 상호운용 체크리스트의 적용을 손쉽게 지원하는 도구 개발이 필요하다.

참고 문헌

- [1] "Level of Information System Interoperability (LISI)", CAISR Architecture Working Group, 1998.
- [2] "DoD Architecture Framework version 2.1", DoD Architecture Framework Working Group, 2000
- [3] Paulk, M. C., Weber C. V., Curtis, B. and Chrissis M. B. "The Capability Maturity Model Guidelines for Improving the Software Process", Carnegie Mellon University, Software Engineering Institute, Addison-Wesley Publishing Company, 1995.
- [4] "국방 컴포넌트 기반 방법론 지침(ADDME)", 국방과학연구소, 2004.
- [5] "국방정보체계 상호운용성 수준(LISI) 업무편람", 국방부, 2002.
- [6] 류동국 외, "컴포넌트 기반 시스템 상호운용성 측정 및 평가를 위한 상호운용 능력 모델 개발", 정보과학회 2004 춘계학술대회 발표논문집, 2004.
- [7] 김영철, "컴포넌트 기반 체계 상호운용 적합성 평가 및 인증 기술 연구", 국방과학연구소 위탁과제, 홍익대학교, 2004.
- [8] 문소영 외, "성숙도 모델을 이용한 상호운용 측정 사례 연구", 정보과학회 2004 춘계학술대회 발표논문집, 2004.