

분산 컴포넌트 명세에 기반한 비즈니스 컴포넌트 구축에 관한 연구

김 행 곤*, 최 하 정*, 한 은 주*
* 대구가톨릭대학교 컴퓨터공학과

A Study on Business Component Construction based on Distributed Component Specification

Haeng-Kon Kim*, Ha-Jung Choi*, Eun-Ju Han*
* Software Engineering Lab., Dept. of Computer Engineering,
Catholic University of Daegu
<http://selab.cataegu.ac.kr>

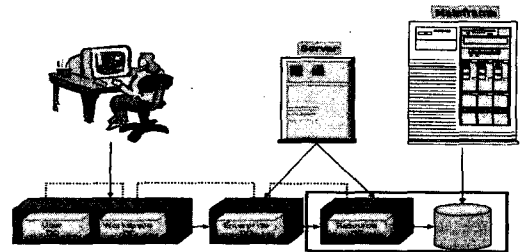
요 약

인터넷을 기반으로 한 디지털 정보의 생성과 운영의 중요성이 더욱 강조되어 가고 있는 현재, 네트워크 상에서 효과적인 인터넷 서비스로 기업의 모든 운영과 관리가 인트라넷 및 인터넷 환경에서의 프로세싱으로 전개되어 빠르고 정확한 정보와 서비스를 지원한다. 컴포넌트기반의 시스템은 분산 컴포넌트(Distributed Component : DC)로 이루어져 있으며, 이는 네트워크 접근 가능한 인터페이스를 제공하는 가장 작은 단위의 컴포넌트로, 사용자(User), 워크스페이스(Workspace), 엔터프라이즈(Enterprise), 그리고 리소스(Resource) 계층으로 나뉘어 진다. 분산 컴포넌트 명세 개발 프로세서를 통한 4계층으로의 분석으로 비즈니스 컴포넌트(Business Component : BC)를 구축하고 재사용 가능하며 비용 절감 및 새로운 비즈니스 컴포넌트들의 접근을 용이하게 할 뿐만 아니라 인터넷 기반의 비즈니스 서비스를 폭넓게 확장시킬 수 있도록 한다.

1. 서론

네트워크 상에서 효과적인 인터넷 서비스의 전문화에 따라 기업의 모든 운영과 관리가 인터넷 및 인트라넷 환경에서의 프로세싱으로 전개되어 빠르고 정확한 정보와 서비스를 지원한다. 그러나 재사용 대상이 되는 컴포넌트들이 대부분 원시 코드 수준의 클래스 라이브러리에 국한되어 있으며, 정규화 된 모델의 미 정립에 의해 재사용에 의한 비용이 스크래치에 의한 비용보다 적지 않다[1].

본 논문에서는 네트워크 접근 가능한 인터페이스를 제공하는 가장 작은 단위의 컴포넌트인 분산 컴포넌트의 요소들, 즉 사용자, 워크스페이스, 엔터프라이즈, 리소스 영역으로 나뉘어 이들이 비즈니스 컴포넌트들로 구성되기 위한 각각의 역할과 기능들을 정확하게 요구 및 분석명세로 정의하여 기업에서 원하는 최종의 시스템 구현에 용이하고자 한다. 분산 컴포넌트 명세 개발 프로세서를 통해 얻어진 명세를 기반으로 비즈니스 컴포넌트를 구축함으로써 재사용이 용이하고 비용 절감, 그리고 기존의 컴포넌트 접근을 용이하게 할 뿐만 아니라 인터넷 기반의 비즈니스 업무 처리 서비스를 폭넓게 할 수 있다.



(그림 1) 분산 컴포넌트 구성

- UDC(User DC) : 비즈니스 컴포넌트를 위한 사용자 인터페이스를 표현하기 위해 특별히 구축되는 분산 컴포넌트들로 윈도우 크기와 사이즈 등의 정보를 제공한다.
- WDC(Workspace DC) : 단일 사용자와 관련 있는 응용논리의 필요성에 의해 제공되는 분산 컴포넌트로, 실제 컴포넌트의 작업공간을 이룬다.
- EDC(Enterprise DC) : 비즈니스 컴포넌트간의 상호동작과 검증, 비즈니스 규칙을 적용하는 계층으로 데이터무결성을 관리한다.
- RDC(Resource DC) : 데이터베이스 액세스를 제공하는 실제 물리적인 처리부분의 컴포넌트 저장소이다.

2. 관련연구

2.1 분산 컴포넌트(Distributed Component : DC)

분산 컴포넌트는 네트워크 접근 가능한 인터페이스를 제공하는 가장 작은 단위의 컴포넌트로서 상업적으로 유용한 컴포넌트 구현기술을 사용하여 구현되는 설계패턴의 응용으로, 전 개발단계 동안 사용되어진다. 분산 컴포넌트의 4가지 카테고리는 (그림 1)과 같이 구분할 수 있다[2].

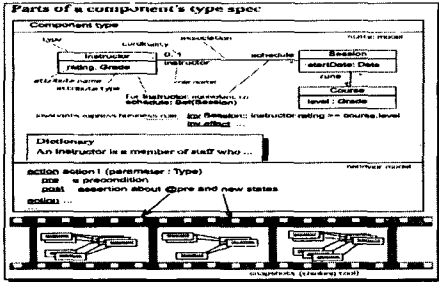


(그림 2) 분산 컴포넌트의 조합을 통한 여러 가지 비즈니스 컴포넌트의 예

(그림 2)에서는 분산 컴포넌트의 여러 가지 조합 방법에 따른 비즈니스 컴포넌트의 종류를 제시하였다[3].

2.2 기존 컴포넌트 명세

카타르시스 방법론에서 컴포넌트를 설계 시, 먼저 컴포넌트의 명세를 정의하여 비즈니스 혹은 도메인을 분석하게 되는데, 이때 비즈니스 타입 모델로의 명세가 제시된다. 또한, 행위 중심의 명세를 제시하여 객체 그룹에서의 이전과 이후 조건에 따른 상태변화를 살펴볼 수 있다. (그림 3)에서는 타입 명세와 행위명세를 표현한 예이다[4].



(그림 3) 타입과 행위 명세

3. 분산 컴포넌트 명세에 기반 한 비즈니스 컴포넌트

3.1 분산 컴포넌트 명세 개발 프로세스

- **요구 명세 단계(요구사항 분석 단계)**

주요 비즈니스 개념을 식별하여 최종 비즈니스 산물 및 서비스를 결정하기 용이하도록 요구 명세를 얻는다.

- **분석 명세 단계(분산 컴포넌트 이해 단계)**

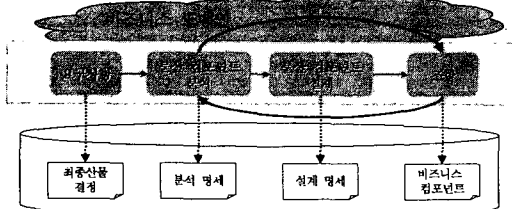
요구 명세를 통해 사용자가 원하는 시스템을 조합할 수 있도록 기본 요소인 사용자, 워크스페이스, 엔터프라이즈, 리소스 측면에서의 고려사항들을 분석하여 내부, 외부, 그리고 의존성 명세로 정확하게 작성하고, 비즈니스 컴포넌트로 조합 가능한 명세를 제공한다.

- **설계 명세단계(모델링 단계)**

분석을 통해 얻어진 명세들을 통해 분산 컴포넌트들간의 관계들을 시각적으로 나타내어 이들간의 상호동작을 쉽게 파악하고 분산 컴포넌트들이 비즈니스 논리에 맞게 조합되는데 도움을 준다.

- **구현 단계(조합 단계)**

비즈니스 컴포넌트는 컴포넌트의 역할과 재사용성을 고려하여 하나이상의 분산 컴포넌트들로 구성되며 결국, 독립된 하나의 비즈니스 컴포넌트를 생성한다. 분석과 설계, 그리고 조합과정은 반복적으로 수행되어진다(그림 4).



(그림 4) 분산 컴포넌트 명세 개발 프로세스

3.2 내부 명세(Internal Specification)

내부적인 설계 명세는 화이트박스 형태로 분산 컴포넌트의 각 역할을 구분할 수 있도록 일관된 형태를 지닌다.

- **작업영역 분산 컴포넌트 명세**

기능	세부기능	명세
workspace	single-user mode	
	multiple-user model	

- **사용자 인터페이스 분산 컴포넌트 명세**

기능	세부기능	명세	
display	User Interface Component	VBX JavaBeans	
	component-based user interface		
	Pluggable user interface	pull	
		push	

- **엔터프라이즈 분산 컴포넌트 명세**

기능	세부기능	명세	
state	Database	get	
		put	
	component accounting		
	introspection		

- **리소스 분산 컴포넌트 명세**

기능	세부기능	명세
Database model	single integrated DB	
	multiple autonomous islands of data	
	single integrated physical DB	
Granularity	RDB	
	OODB	
foreign key	primary key	
	logical foreign key	
	physical foreign key	
relationship	normalization	
	one to many	
	many to many	
access	Direct SQL access	
	layered access	

3.3 외부 명세(External Specification)

외부명세에서는 다른 분산 컴포넌트들간의 상호관계를 위한 인터페이스 표현으로, 최종 시스템의 특정 행위에 대한 형식적인 의존성을 가진 행위라고 할 수 있다.

- **작업 영역 분산 컴포넌트**

기능	세부기능	명세	
interface categories	business entity life cycle		
	collection management		
	functional interface		
	component lifecycle		
	socket operation		
	configurability		
	introspection		
	dispatch		
	time-type	run-time interface	
		build-time interface	
design-time interface			
interface design patterns	naming conventions		
	one operation for each kind		
	a group operation for single attribute		
	a tagged data string as the operation parameter		
	a mix of tagged data parameter and typed parameter		
Business Data Type	dependent object		
	network class		

● 엔터프라이즈 분산 컴포넌트

기능	세부기능	명세
interface categories	위와 동일	
naming conventions		
time-type	위와 동일	
interface design patterns	위와 동일	
Business Data Type	primary key inclusion	
	consistency with database	
	nesting minimization	
	self-containment	
	system wide consistency	
	boundary values	

3.4 의존성 명세(Dependency Specification)

의존성 명세는 하나의 비즈니스 컴포넌트내의 분산 컴포넌트들간의 관계와 다른 비즈니스 컴포넌트들내의 분산 컴포넌트들과의 관계를 나타낼 수 있다.

● 의존성 컴포넌트

기능	명세
internal	
external	

4. 사례 연구

4.1 요구사항 분석

● 요구 사항

고객 요구사항
사용자가 원하는 물건을 구매할 때 사용자의 웹 브라우저를 통해 사업자의 사이트에 접속하여, 물건을 구매하기 위해 신규등록을 하여 자신의 개인 정보를 서버에 저장시켜 놓는다. 또한 원하는 물건에 대한 정보를 얻고자 할 때 서버로부터 물건에 대한 정보를 얻을 수 있으며, 서버는 구매 정보를 전용 데이터베이스를 통해 탐색하여 사용자에게 정보를 전달해 준다. 구매 요구가 들어오면 서버는 계약된 납품업자에게 연락을 하여 사용자가 원하는 물품을 제작하도록 연락하며, 납품업자는 물건에 대한 명세를 받아서, 제작하도록 한다.

● 구분된 요구사항 명세

전자상거래			단계 : 요구사항
번호	비즈니스 컴포넌트	의존성 컴포넌트	설명
1	고객		전자상거래를 이용하여 물품 구매
2	정보		서버에서 얻은 정보 사용자 전달
3	서버	정보	고객 정보, 물품정보등 관리
4	납품업자	물품	물품을 공급

4.2 분석 명세

● 사용자 인터페이스 명세

기능	세부기능		명세
display	User Interface Component	JavaBeans	웹 상에서 사용자가 정보 획득, 검색
	browser		netscape

● 엔터프라이즈 명세

기능	세부기능		명세
state	Database	get	데이터베이스로부터 정보 획득, 저장
		put	
	component accounting		컴포넌트의 관리

● 리소스 명세

기능	세부기능		명세
Database model	single integrated DB		모든 명세란 한 곳에 저장
Granularity	RDB		컴포넌트들간의 관계 표현
foreign key	primary key		고객, 납품업자 번호, 물품 번호
	logical foreign key		서버 정보
	physical foreign key		서버번호, 데이터베이스 번호

normalization		2NF
relationship	one to many	서버 와 고객
	many to many	고객 과 물품
access	Direct SQL access	정보 요청, 신규등록, 정보 검색

4.3 조합된 비즈니스 컴포넌트

서버 비즈니스 컴포넌트 명세				
명세	기능	세부기능	명세	
내부	state	Database	get put	서버를 통해 개인정보 및 상품 정보 접근기능
	Database model	single integrated DB		모든 명세 저장
	Granularity	RDB		고객, 서버, 데이터베이스 관계 표현
	primary key			서버명
	foreign key	logical foreign key		고객번호, 상품 번호
		physical foreign key		데이터베이스 번호
	normalization			2NF
	relationship	one to many		서버 와 고객
	access	Direct SQL access		정보 검색 요청
	외부	interface categories	functional interface	
naming conventions			상품과의 연계	
time-type		run-time interface		고객의 요청에 따른 인터페이스 수행
interface design patterns		a mix of tagged data parameter and typed parameter		쿠폰부분
Business Data Type		consistency with database		등록된 고객과 상품에 대한 정보
		primary key inclusion		고객 번호, 상품번호 등
의존성	external	정보 비즈니스 컴포넌트와의 연계		

5. 결론 및 향후연구

현재 인터넷을 이용한 비즈니스 거래와 분산 업무 처리가 확대되면서 인터넷을 기반으로 한 컴퓨터 환경의 변화로 정확한 정보와 서비스를 지원하기 위해 컴포넌트기반 기술이 많이 확대되고 있다.

본 논문에서는 분산 컴포넌트 명세 개발 프로세서를 통해 네트워크 가능한 인터페이스를 제공하는 분산 컴포넌트를 4가지 영역으로 나누어 분석하고, 내부, 외부, 그리고 의존성 명세로 구분 정의하여 얻어진 명세를 통해 비즈니스 컴포넌트로 구축한다. 향후 연구과제로는 비즈니스 컴포넌트 및 공통 컴포넌트와 영역별 컴포넌트의 분석 및 설계 명세 접근과 이러한 명세 정보의 확장 및 컴포넌트 명세 저장소 구축에 있다.

【참고 문헌】

[1] Peter Eccles, Oliver Sims, Building Business Object, OMG Press, 1998.
 [2] 김행곤, 최하정 외 2인, “웹에서 비즈니스 모델을 기반으로 한 컴포넌트 관리”, 1999 한국 정보처리학회 추계 학술논문 발표회, 포스터 발표, 1999.10.
 [3] Peter Herzum, Oliver Sims, Business Component Factory, OMG Press, 2000.
 [4] Desmond D'souza & Alan Wills, Object, Component and Framework with UML, Addison-Wesley, 1998.
 [5] Chris Marshall, Enterprise Modeling with UML, Addison-Wesley, 2000.
 [6] Jeff Sutherland, "The Emergence of a Business Object Component Architecture", Available by web server from <http://jeffsutherland.org/ooopsla99/>, 1999.