

기업 자재관리시스템에서 효율적인 검색을 위한 순환적 방법의 적용

김영준^{1*}

¹백석문화대학 경상학부

An Application of Recursive Method for Efficient Retrieval in Business Material Management

Young-Jun Kim^{1*}

¹Economics & Commerce Division, Baekseok Culture University

요약 기업의 자재관리 영역에서 자재소요계획은 효과적인 의사결정을 위한 중요한 활동 중 하나이다. 수요가 점차 늘어남에 따라 실용성 있는 데이터 모델이 필수적으로 요구되고 있다. 본 연구는 다양한 형태로 표현되는 데이터를 적절한 구조로 모델링하여 기업 자재관리 응용 분야에서 효율적으로 사용할 수 있도록 연역 모델 릴레이션을 설계한다. 이를 이용하여 순환적 데이터 검색을 가능하게 하는 새로운 질의 방법을 제시하며, 그 효율성을 비교한다.

Abstract In business material management, as the number of products and their parts increases, the employment of databases becomes inevitable. Many researchers have tried to incorporate semantics in traditional models using logic programming, resulting in the deductive method. However, since the designer of this method does not overcome the traditional design concept of relational method, there have been not much successful implementations in deductive method. Therefore, this paper propose the new way designing the retrieval methods for deductive databases and using those relations the query that makes recursive retrievals possible and hypotheses test in business material management.

Key Words : Business modeling, Material management, Recursive process, System analysis

1. 서론

정보화 및 지식산업사회가 점차 발전할수록 기업의 어느 조직에서나 처리하고 분석해야 하는 정보의 양은 기하급수적으로 늘어나고 있으며, 이를 조직 전체의 효율성 (efficiency)과 실효성(effectiveness) 관점에서 접근하고자 노력들이 시도되고 있다. 기업 관리의 성공 여부는 얼마나 유용한 정보를 얻어 정확히 분석해 적시에 제공하느냐에 따라 좌우된다고 해도 과언은 아닐 것이다. 기업의 자재소요계획은 자재관리 영역에서 중요한 의사결정 활동 중 하나이며, 수요가 점차 늘어남에 따라 실용성 있는 데이터 모델이 필수적으로 요구되고 있다. 관계형 데이터 모델은 이러한 요구에 능률적으로 부합하기 위하여 SQL,

QBE, Quel 등의 질의어(query language)를 채용하고 있다. 그 중 SQL(structured query language)은 대표적인 질의어로서 현재는 인공지능 기법이 정립됨에 따라 데이터 검색에 있어 이들 질의에 논리(logic) 프로그래밍 기법의 적용에 대한 관심이 고조되고 있으며 그와 관련된 연구도 활발히 진행되고 있다 [3,4]. 그러나 SQL은 논리 기법을 적용하는 데 있어 다소의 문제점을 내포하고 있다. 논리 언어에서 아주 빈번이 사용되고 쉽게 표현되는 순환적 질의(recursive query)를 처리할 수 있는 반복적 연산 기능이 없다는 것이다[1]. 따라서 논리 기법을 적용하려면 순환적 개념을 처리할 수 있는 별도의 알고리즘이 필요하다.

본 연구에서는 자재관리 영역에서 이러한 문제점을 해

*교신저자 : 김영준(yjkim@bscu.ac.kr)

접수일 10년 07월 21일

수정일 10년 08월 05일

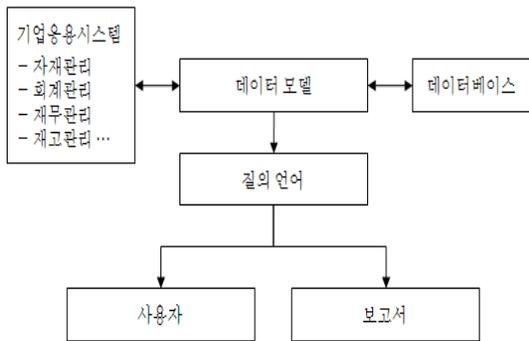
게재확정일 10년 09월 08일

결하고 순환적 질의를 검색하기 위하여 논리 기법을 이용한 연역 모델(deductive model)을 설계하고, 기존의 데이터 모델과의 비교·분석을 통해 효율적인 측면을 고찰한다. 본 모델은 다양한 형태로 표현되는 데이터들을 적절한 구조로 모델링하여 여러 응용분야에서 효율적으로 사용할 수 있도록 구성된 체계적인 모델이다. 즉, 연역 기법을 적용하여 데이터 모델과 추론 기능을 결합함으로써 추론을 통해 새로운 데이터를 추출하고 생성할 수 있는 기반을 제공하며 연역 모델을 위한 릴레이션과 새로운 질의를 설계한다.

2. 데이터 모델 연구

2.1 연역 모델

일반적인 데이터 모델의 질의처리 환경은 다음의 [그림 1]과 같다.



[그림 1] 데이터 모델의 질의처리 환경

관계형 모델로부터 발전된 연역 모델은 논리를 기반으로 하고 있으며, 데이터 모델의 조작 기능과 논리시스템의 추론 기능이 결합된 모델이다[7]. 이는 추론을 통해 데이터베이스에 명백하게 저장되어 있지 않은 데이터를 생성할 수 있다[8]. 연역적 추론 기법을 이용하여 이미 존재하는 데이터로부터 새로운 데이터 즉, 명시적으로 저장되어 있지 않은 데이터를 검색한다[9]. 연역 모델은 데이터를 릴레이션 뿐만 아니라 일반 규칙의 형태로도 저장할 수 있기 때문에 특정 데이터들의 관계를 표현할 경우에 불필요하게 장소를 차지하면서 그 관계들을 일일이 저장할 필요없이 필요한 추론에 의해 데이터를 검색할 수 있어 기존 모델보다 검색능력이 더욱 강력하다. 관계형 모델은 데이터를 명시적으로 저장하기 때문에 데이터 관리는 용이하나 규칙을 표현할 수 없으며 데이터의 표현을 위한 유용성이 배제되어 있는 것이 사실이다[10]. 반면,

연역 모델은 데이터를 릴레이션 뿐만 아니라 일반 규칙의 형태로도 표현할 수 있어 필요한 추론에 의해 데이터를 적시에 추출할 수 있다.

[표 1] 관계형 모델과 연역 모델의 비교

명시적인 데이터	연역적 데이터
use (02,301) use (15,301) use (21,503) use (32,602) use (03,301)	use (x,301) :- x<20.

[표 1]은 ‘어떤 제품(x)은 특정한 부품(y)을 사용한다’의 예를 통해 관계형 모델의 명시적인 데이터와 연역적 데이터를 비교한 것이다. 명시적인 데이터에서는 제품번호가 20 미만에서는 부품 301을 사용한다는 것이 릴레이션 내에 명시적으로 저장되어 있는 반면에 연역 모델은 부품번호가 입력되었을 때 $x < 20$ 의 규칙에 따라 x 와 20을 비교하여 부품 301을 사용하게 됨을 알 수 있게 한다. 순환(recursion)은 논리식의 헤드(head)와 바디 프레디카트(body predicate)에 동일한 관계 기호가 나타나는 것을 의미하며, 자신이 자신을 되부름 하는 표현이다[5]. 논리식의 순환 규칙에 대한 SQL 질의는 관계형 모델에는 반복적 연산 기능이 없기 때문에 응용 프로그램을 사용하여 해결책을 얻을 수 있는 형태로의 전환이 불가피하다. 순환적 질의의 논리식 표현은 $a :- b, a$. 이며, need(buyer,product)의 기본 관계로부터 유도되는 buy(buyer,product) 관계의 선형적 순환으로 표현할 수 있다[6]. 이 관계를 이용해 ‘x가 y를 필요로 하면 x는 y를 구입한다’의 순환적 개념을 표현하면 다음과 같다.

$buy(x,y) :- need(x,y).$

$buy(x,y) :- need(x,z), buy(z,y).$

2.2 자재관리 모델

자재관리는 여러 겹으로 얽혀진 생산 조직에 대해 언제 어떠한 행동을 취해야 하는가를 정확하게 제시하고, 그것에 따라 원자재, 부자재로 부터 완제품에 이르기까지 자재의 흐름을 효율적이고 합리적으로 관리하는 방법이다[2]. 조직내에서 자재의 전체적인 흐름은 완제품에 대한 하위시스템으로 형성되어 있고 그 하위시스템은 조직의 기능이나 규모에 따라 각각 구조를 달리하고 있으며 일반적으로는 원자재, 재공품, 완제품 등의 관리부서로 구분하고 있다. 기업의 자재관리는 전사적인 차원에서 관리하기도 하고 각 부문별로 구분하여 각각의 주관부서에서 관리하기도 한다. 본 연구에서 사용할 관계형 릴레이

선은 [표 2]에 표현되어 있으며, 자재들의 소요 수준 즉, 완제품에 쓰이는가 또는 부품에 쓰이는가에 따라 데이터를 4단계로 구분하여 사용한다.

[표 2] 자재관리 모델 릴레이션

product			sub_part1		
product#	product_name	part#	sub_part1#	sub_part1_name	sub_part2#
101	monitor	201	301	VGA cable	401
101	monitor	202	302	DVI cable	402
101	monitor	203	303	condenser	401
101	monitor	204	304	adapter	403
			305	backlight	-
			306	antenna	404
			307	plug	-
			308	speaker	405
			309	sound cable	403

part		
part#	part_name	sub_part1#
201	AD board	301
201	AD board	302
201	AD board	303
201	AD board	304
202	inverter	303
202	inverter	305
202	inverter	306
203	SMPS	305
203	SMPS	307
203	SMPS	308
204	LIPS	301
204	LIPS	302
204	LIPS	309

sub_part2	
sub_part2#	sub_part2_name
401	cab1
402	con1
403	sou1
404	ada1
405	bac1

모델 릴레이션은 product, part, sub_part1, sub_part2 등 네 개로 구성되어 있다. 릴레이션 product와 part는 제품 번호 101인 완제품 monitor를 생산하는데 AD board, inverter, SMPS, LIPS 등의 부품을 필요로 하며 이들의 부품번호는 201, 202, 203, 204 임을 보여주고 있다. 이 릴레이션은 부품 AD board에는 하위 부품으로 301, 302, 303, 304의 부품들이 소요된다는 내용도 명시적으로 포함하고 있다. 릴레이션 sub_part1, sub_part2도 이와 동일한 기준으로 데이터 상호간의 관계를 해석할 수 있다.

일반적으로 기업의 부품 구성표는 대부분 3~4단계 이상이고, 산업이 점차 발전함에 따라 생산 공정의 성격도 매우 정밀화, 전문화되어 다른 기업의 완제품을 원자재로 사용하는 기업이 점차 늘고 있다. 심지어 전혀 제품은 생산하지 않고 타사의 완제품만을 조립하는 기업까지 등장하고 있다. 따라서 기존의 관계형 모델 릴레이션으로 데이터를 설계한 경우 부품의 구성이 바뀌고 부품 구성의 단계가 추가될 때 새로운 릴레이션을 추가하고, 그에 따른 새로운 SQL 질의 표현을 사용해야만 사용자가 원하는 정보를 얻을 수 있다. 그리고 표현 자체도 부품 구성이 4단계 이상으로 구성되면 매우 복잡하게 되기 때문에 그 처리는 매우 비효율적이 된다. 이러한 비효율성을 극복하기 위한 본 연구의 수행방향은 크게 두 가지로 나

누어진다. 첫째는 [표 2]의 전통적 릴레이션 설계에서 벗어나 자재관리에 적합한 새로운 릴레이션을 설계하는 것이고, 둘째는 새롭게 설계된 릴레이션을 사용하여 간결한 질의를 통해 원하는 정보를 얻을 수 있도록 순환적 질의를 설계하는 것이다.

3. 질의의 표현

[표 2] 릴레이션을 기반으로 하는 질의 ‘릴레이션 sub_part1에서 sub_part2#가 403인 sub_part1#와 sub_part1_name을 검색하라’와 ‘릴레이션 part에서 sub_part1#가 305보다 작은 part#와 part_name을 검색하라’는 질의를 통해 명시적으로 표현되어 있는 데이터를 검색할 수 있다.

```

질의1 : SELECT sub_part1#, sub_part1_name
        FROM sub_part1
        WHERE sub_part2# = 403

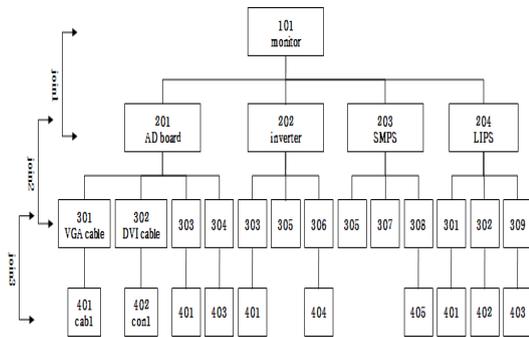
결과 : sub_part1#      sub_part1_name
        304              adapter
        309              sound cable
    
```

```

질의2 : SELECT part#, part_name
        FROM part
        WHERE sub_part1# < 305

결과 : part#          part_name
        201            AD board
        202            inverter
        204            LIPS
    
```

[표 2]의 관계형 모델 릴레이션에는 완제품 monitor의 최종 부품이 어떤 것인지에 관한 명시된 데이터가 포함되어 있지 않다. 또한 부품 AD board에 어떤 최종 부품이 필요한지에 관한 명시된 데이터도 없다. 즉, 완제품 monitor의 최종 부품을 찾으려면 부품번호 101 → 201 → 301 → 401의 순으로 검색할 뿐만 아니라 101 → 201 → 302 → 402 등의 순으로 검색하는 일련의 과정을 거쳐야만 원하는 결과를 얻을 수 있다. 그림 2는 본 연구의 부품 구성을 단순화시킨 것으로 최종 완제품에 소요되는 각 단계별 부품들의 번호와 이름을 표현하였다.



[그림 2] 자재 모델의 구성

현재, 관계형 모델에서 명시적으로 저장되어 있지 않은 데이터를 처리하는 방법으로는 조인 오퍼레이션(join operation)을 이용하는 방법이 있다. 질의 ‘완제품 monitor의 최종 부품을 검색하라’를 처리하기 위해서는 먼저 조인 오퍼레이션을 이용하여 필요한 데이터들이 포함된 릴레이션들을 결합해 새로운 제 3의 릴레이션을 만들고 그 새로운 릴레이션에 데이터들을 명시적으로 저장하여 처리해야 한다. 그러나 조인은 한 번에 단지 서로 다른 두 개의 릴레이션만을 결합하기 때문에 관계형 릴레이션에서 제품에 소요되는 최종 단계들 또는 하위 단계들의 부품을 알고자 할 때에는 현 모델의 릴레이션이 네 개이므로 조인을 이용한 세 번에 걸친 결합 과정이 필요하다. 즉 [그림 2]의 좌측에 표기되어 있듯이 릴레이션 product와 part를 이용해 릴레이션 join1을 만들고, part와 sub_part1을 이용해 join2를 만들고, join3을 만드는 과정을 거치게 된다. 이를 이용하여 ‘새로이 생성된 릴레이션 join3에서 완제품 monitor에 소요되는 최종 단계의 부품을 검색하라’를 처리하면 다음과 같다.

질의 : *SELECT product_name, sub_part2_name*
FROM join3
WHERE product_name = monitor

결과 : *product_name sub_part2_name*
monitor cabl
monitor con1
monitor soul
monitor adal
monitor bac1

이 질의는 릴레이션 join3을 사용자 원하는 형태로 성공적으로 형성된 경우에 한하여 올바른 정보를 제공하게 된다. 그러나 위 결과에서 볼 수 있듯이 검색 결과는 단계 4의 부품만 검색되었다. 결국 이것은 완제품 monitor

에 소요되는 최종 단계의 부품 전부가 아님을 알 수 있다. [그림 2]에서 완제품 monitor에 소요되는 최종 부품으로는 단계 4의 부품만 있는 것이 아니라 부품번호 305, 307 등 단계 3에도 최종 부품이 있기 때문이다. 결국 관계형 모델을 기반으로 한 SQL은 사용자가 원하는 올바른 정보를 제공하여 주지 못하고 있는 것이다. 그 이유는 관계형 모델에서 지원해 주는 조인 오퍼레이션이 단지 릴레이션간의 결합을 행할 뿐 각 릴레이션 상호간의 논리적인 관계를 연결해 주지 못하고 있기 때문이다. 이러한 문제점을 효율적으로 해결하고자 본 연구에서는 자재관리 영역에 순환적 질의 방법을 적용하고, 그 순환적 질의에 맞는 새로운 데이터 모델과 릴레이션을 설계한다.

4. 순환적 질의의 설계

4.1 연역 릴레이션 설계

순환이 가능하기 위해서는 부품과 그 부품에 사용되는 다른 부품 또는 완제품간의 관계가 부품 단계와 무관하게 동일한 관계로 표현되어야 한다. 본 연구에서 설계하여 자재관리에서의 순환적 표현에 사용한 연역 모델의 릴레이션은 [표 3]이며, 두 개의 릴레이션 pro_name, pro_use로 구성되어 있다.

[표 3] 자재관리의 연역 모델 릴레이션

pro_name		pro_use	
product#	product_name	product#	part#
101	monitor	101	201
201	AD board	101	202
202	inverter	101	203
203	SMPS	101	204
204	LIPS	201	301
301	VGA cable	201	302
302	DVI cable	201	303
303	condenser	201	304
304	adapter	202	303
305	backlight	202	305
306	antenna	202	306
307	plug	203	305
308	speaker	203	307
309	sound cable	203	308
401	cabl	204	301
402	con1	204	302
403	soul	204	309
404	adal	301	401
405	bac1	302	402
		303	401
		304	403
		306	404
		308	405
		309	403

[표 3]에 설계된 연역 모델 릴레이션은 [표 2]의 관계형 모델 릴레이션과 동일한 의미의 데이터이며, 네 개의 관계형 모델을 두 개의 연역 모델로 표현한 것이다. 릴레이션 `pro_name`은 완제품 또는 부품번호와 이에 해당하는 완제품 또는 부품이름을 가지고 있는 릴레이션이며, 릴레이션 `pro_use`는 완제품 또는 부품번호들로만 구성된 연역적 데이터를 표현하고 있는 릴레이션이다.

이 방법으로 설계된 릴레이션에서는 완제품 또는 부품을 명시적으로 나타낼 필요가 없다. 왜냐하면, 어떤 질의 검색과정에서 `product#`에 있는 완제품 또는 부품이 `part#`에 나타나지 않으면 그것은 완제품이며, 또 `part#`에 있는 부품이 `product#`에 나타나지 않으면 그것은 최종 부품이라는 것을 알 수 있기 때문이다. 따라서 본 연구에서의 연역 모델에서는 완제품과 부품의 구별하는 것은 의미가 없다.

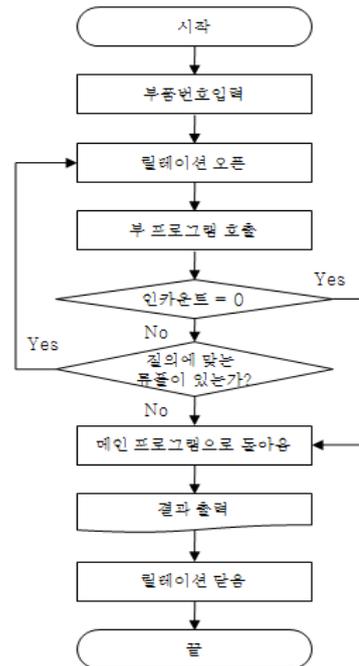
4.2 순환적 질의 프로세스

[표 2]에 표기된 관계형 모델 릴레이션만을 이용해서는 순환적 개념을 적용할 수 없고 그 릴레이션을 기반으로 응용 프로그램을 작성한다고 할지라도 순환적 개념을 표현할 수는 없었다. 따라서 본 연구에서는 연역 모델 릴레이션을 사용하여 순환적 개념을 적용해 결과를 도출하고자 임베디드 SQL을 이용해 질의를 설계하였다. 본 연구에서 설계한 방법은 연역 모델 릴레이션에서 순환적 개념을 이용해 부품에 소요되는 최종 부품의 리스트를 출력하는 것으로, 이 연역 모델을 이용하면 ‘부품번호 101인 monitor에 소요되는 최종 부품을 모두 검색하라’의 결과가 정확하게 도출된다(결과: 305 307, 401, 402, 403, 404, 405). 이와 같은 결과는 프로그램 내에 순환 프로시저가 다른 매개변수 값을 가지고 반복해서 수행되는 것이 가능하기 때문에 나타났다. 즉, 모든 변수 값이 순환 프로시저에 의해 다시 생성되고, 그 순환이 가능한 이유는 부품간의 종속관계가 하나의 릴레이션으로 나타나기 때문이며 [그림 3]은 이러한 본 연구의 연역 모델 릴레이션을 위한 순환적 질의의 흐름을 보여주고 있다.

사용자에 의해 검색하고자 하는 부품번호가 입력되면 비교를 위해 릴레이션이 오픈되고 부프로그램을 호출하는 기능으로 이어진다. 질의에 맞는 튜플이 있는지 검색하는 과정을 거쳐 조건에 적합한 결과가 도출될 때까지 반복과정으로 진행된다. 이러한 과정의 연역 모델 릴레이션을 기반으로 하는 순환적 질의의 임베디드 SQL을 관계형 모델 릴레이션을 이용한 방법과 비교하면 매우 간결하고 효율적이다. 그 이유는 순환적 질의 검색에는 순서 제어에 관한 내용이 삽입되지 않아도 되기 때문이며, 프로그램의 의미를 보다 명확하게 반영한다고 할 수 있

다. 또한 연역 모델을 이용한 방법은 관계형 릴레이션을 사용한 방법과는 달리 부품 구성의 단계가 증가해도 릴레이션이나 프로그램의 수정없이 기존의 `pro_use` 릴레이션에 새로운 튜플만을 추가하여 그대로 사용할 수 있으며, 정형화된 형태이기 때문에 예로 든 질의 이외에 다음과 같은 형태의 질의도 프로그램의 수정없이 가능하다. ‘부품 403이 쓰이는 최종 부품(완제품)을 검색하라’는 연역 모델 릴레이션을 기반으로 한 순환적 질의의 순환 프로시저 내에서 SELECT 문장을 약간 수정함으로써 정확한 결과를 도출할 수 있다. 즉, 이 질의를 검색하기 위해서는 다음과 같이 기존의 애트리뷰트 `product#`를 `part#`로 이름을 변경해 주면 가능하게 된다.

```
EXEC SQL SELECT product#, part#
FROM pro_use
WHERE part# = num;
```



[그림 3] 순환적 질의 처리

따라서 본 연구에서 설계한 모델은 부품에 소요되는 최종 부품을 효율적으로 검색할 뿐만 아니라 응용 프로그램에서 SELECT 문장의 수정을 통하여 특정 부품이 어떤 상위 부품에 필요한지도 검색할 수 있는 등 다양한 형태의 순환적 개념이 포함된 질의들을 검색할 수 있음을 알 수 있다.

5. 시스템 평가

5.1 가설 분석

연역 모델을 이용해 기업 자재관리시스템에서 효율적인 검색을 위한 순환적 모델을 설계하는 과정에서 얻었던 것 중 의미 있었던 것은 실제적으로 기업내 자재관리 영역의 본질을 파악할 수 있었다는 것이다. 특히, 연역 모델링 과정에서 표현한 자재관리 분야에서의 검색 방법은 순환적 질의에 대한 어려움을 해결하고 있다. 본 연구에서 설계한 시스템의 타당성을 검토하기 위해 다음의 가설을 설정하여 평가한다.

가설 1 : 기존 데이터 모델보다 연역 모델 릴레이션 사용이 더욱 효율적일 것이다.
 가설 2 : 효율성은 초보사용자 그룹보다 전문가사용자 그룹에서 높게 나타날 것이다.

본 연구에서 설계한 시스템을 평가하기 위해 기존의 데이터 모델링 방법과 본 연구에서 제시한 연역 모델을 이용하는 방법을 상호 비교하였다. 이를 위해 사용자 그룹을 초보사용자와 전문가사용자 그룹으로 나누어 통계적 기법을 이용하여 분석하였다. 각 그룹에게 문제 명세와 연역 모델을 제시하고, 각 그룹별로 사용자가 연역 모델을 생성하고 수정하는 일련의 과정을 통해 최종 연역 모델을 도출할 수 있도록 하였다. 측정 결과를 종합해 본 시스템의 평가 가설을 분석하면 다음과 같다.

[표 4] 기존 모델과 연역 모델 릴레이션 비교

측정	기존 시스템 모델	본 시스템 연역 모델
평균소요	38.3	10.4
표준편차	4.21	1.20
표준기준	100	27.1

[표 4]와 같이 전체적인 측면에서 기존 데이터 모델링 방법보다는 본 시스템에서 설계한 연역 모델의 이용에 소요되는 시간이 현저하게 감소되었음을 알 수 있다. 따라서 가설 1은 채택되어 기존 데이터 모델보다 본 시스템 연역 모델 릴레이션 사용이 더욱 효율적이라 할 수 있다. 즉, 초보사용자와 전문가사용자를 대상으로 측정한 결과 본 시스템은 두 사용자 그룹에서 모두 연역 모델링 과정에 대한 적절한 지원이 가능하였다.

[표 5] 초보사용자와 전문가사용자 그룹간 비교

측정	초보사용자 그룹		전문사용자 그룹	
	기존 모델	본 연역 모델	기존 모델	본 연역 모델
평균소요	46.2	14.7	18.4	6.5
표준편차	3.26	1.35	3.11	1.04
표준기준	100	31.8	100	35.3

두 그룹간의 차이를 분석하면 기존 데이터 모델링 방법보다 본 시스템을 사용할 경우 기업의 자재관리 부문 초보사용자 그룹보다는 전문가사용자 그룹에서 연역 모델 이용에 소요되는 시간이 현저하게 감소되었음을 알 수 있다. 이는 초보사용자 그룹보다 모델의 작성 경험이 있어 그 개념을 잘 이해하고 있는 전문가사용자 그룹에서 본 시스템의 지원은 더욱 효율적이라는 것을 의미한다. 따라서 데이터 모델링에 대한 적절한 경험과 지식이 있는 사용자가 본 시스템을 사용할 경우 연역 모델링 과정에서의 상당한 생산성을 기대할 수 있을 것으로 판단된다. 가설 2는 채택되어 연역 모델링 효율성은 초보사용자 그룹보다 전문가사용자 그룹에서 높게 나타난다고 할 수 있다.

5.2 의의

이상의 가설 분석 결과로부터 얻을 수 있는 이점은 다음과 같다. 첫째, 연역 모델을 이용함으로써 데이터 모델링에 소요되는 시간을 단축할 수 있으며 이로 인해 실제적으로 기업에서는 자재관리시스템의 개발 노력과 비용 절감의 효과를 얻을 수 있다. 둘째, 단순히 데이터 모델이 아닌 순환적 표현이 가능한 연역 모델의 그래픽 사용자 인터페이스를 제공함으로써 기존 데이터 모델링 방법보다 사용하기 쉽다. 셋째 순환적 질의에 대한 효율적인 검색이 가능해 짐으로써 새로운 연역 모델의 데이터를 쉽게 생성할 수 있다. 즉, 본 시스템을 이용한 기업 자재관리시스템에서 효율적인 검색을 위한 순환적 방법의 적용은 시스템 모델링에 소요되는 시간과 노력을 감소시키고 이로 인해 기업의 생산성을 향상시킬 수 있게 된다. 따라서 본 시스템의 순환적 질의 검색은 소프트웨어의 품질을 향상시키고 기업 자재시스템 관리의 효율성을 가져다 줄 수 있을 것으로 기대된다.

6. 결론

지금까지 기업의 자재관리 데이터를 이용하여 관계형 질의의 문제점을 제시하였고, 그 문제점들을 해결하기 위한 순환적 질의의 표현을 중심으로 고찰하였다. 또한, 관

계형 모델과 연역 모델을 비교·기술하면서 연역 모델에서 순환적 질의를 가능하게 하는 자재관리 릴레이션과 임베디드 SQL을 설계하였다. 본 연구에서 설계한 연역 모델의 릴레이션 및 순환적 질의의 이점은 다음과 같다.

첫째, 관계형 릴레이션을 사용하는 방법에 비해 본 연구의 연역 모델 릴레이션을 이용하는 방법은 순서 제어에 관한 내용이 삽입되지 않아도 되기 때문에 프로그램이 간결해지게 됨으로써 이해하기가 쉽고 프로그램의 의미를 보다 명확하게 반영할 수 있다. 둘째, 하위 단계의 부품 구성이 추가되었을 때 관계형 모델에서는 새로운 릴레이션 작성해야 하지만 본 연구에서 설계한 자재관리 연역 모델 릴레이션에서는 새로운 데이터 릴레이션을 만들 필요없이 기존의 데이터 릴레이션에 새로운 튜플만을 추가하여 사용할 수 있다. 셋째, 본 연구에서 설계한 연역 모델은 매우 정형화된 형태이기 때문에 다른 기업의 자재관리에 적용할 경우에도 프로그램의 변경없이 적용하고자 하는 기업의 자재관리에 맞는 데이터와 그 애트리뷰트만을 수정하여 사용할 수 있는 프로그램의 유연성(flexibility)이 있다. 넷째, SQL 표현식의 형태로 나타나는 다른 질의의 검색에도 프로그램 전부를 수정하는 것이 아니라, 그 표현식만을 변경하여 사용할 수 있으므로 본 연구의 연역 모델 릴레이션을 기반으로 하는 방법은 다른 질의의 검색에 쉽게 응용될 수 있는 이점이 있다. 그러나 본 연구는 릴레이션의 애트리뷰트를 부품과 하위 부품으로 구성하여 그 관계만을 이용해 질의를 검색하였기 때문에 실제 업무에서 사용하는 자재관리 데이터로는 부족하다 할 수 있다. 자재관리에 는 부품 소요량이 매우 중요한 정보이기 때문에 부품과 하위 부품과의 관계뿐만 아니라 부품 소요량, 공급자 등 데이터의 범위를 확장하여 순환적 질의를 검색하면 더욱 효율적일 것이다.

Quantification in SQL," *ACM SIGMOD RECORD*, Vol. 20, No. 3, Sep., 2001, pp.304-325.

- [5] Gallaire, H., J. Minker, and J. Nicolas, "Logic and Database : A Deductive Approach," *ACM Computing Surveys*, Vol. 16, No. 5, 2008, pp.145-156.
- [6] Naughton, F., "Minimizing Function-Free Recursive Inference Rules," *Journal of the ACM*, Vol. 36, No. 1, Jan., 2009, pp.69-91
- [7] Parsaye, K., M. Chignell, S. Khoshafian, and H. Wong, *Intelligence Database*, John Wiley & Sons Inc., 2000.
- [8] Saraiya, P., "Hard Problems for Simple Logic Programs," *ACM SIGMOD*, 2004, PP.64-80.
- [9] Sarda, L., "Extensions to SQL for Historical Database," *IEEE Transactions on Knowledge and Data Engineering*, Vol 23, No. 5, June, 2006, pp.220-235.
- [10] Thur, S., and C. Zaniolo, "LDL: A Logic-Based Data Language," *Proceedings of the 12th International Conference on Very Large Database*, Kyoto, Aug., 2008, pp.302-320.

김 영 준(Young-Jun Kim)

[정회원]



- 1992년 8월 : 아주대학교 대학원 경영정보학과 (경영학석사)
- 2001년 2월 : 한국외국어대학교 대학원 경영정보학과 (경영학박사)
- 1998년 3월 ~ 현재 : 백석문화 대학 경상학부 교수

<관심분야>

경영정보시스템, 시스템 분석, 비즈니스 모델링

참고문헌

- [1] Chakravarthy, U., J. Grant, and J. Minker, "Logic-Based Approach to Semantic Query Optimization," *ACM Transactions on Database Systems*, Vol. 15, No. 2, 2005, pp102-121.
- [2] Chase, B., and N. Aquilano, *Production and Operations Management : A Life Cycle Approach*, 5th ed., Richard D. Irwin Inc., 2009.
- [3] Emerson, L., M. Darnovsky, and J. Bowman, *The Practice SQL Handbook using Structures Query Language*, Addison-Wesley Publishing Company, 2007.
- [4] Fratarcangeli, C., "Technique for Universal