한국항해항만학회지 제29권 제5호, pp. 457~464, 2005 (ISSN-1598-5725)
Journal of Korean Navigation and Port Research

데이터베이스의 효과적인 통합방안에 관한 연구
- Name Conflict의 식별을 중심으로-

이홍걸* · 比嘉邦彦** · 富士川孝之***

*한국해양대학교 물류시스템공학과 강사, **동경공업대학교 CRAFT 교수, ***NTT Data 연구원

A Study on the Effective Database Integration Methodology
- The Identification of Name Conflict -

Hong-Girl Lee · Kunihiko Higa** · Takayuki Fujikawa****

** Department of Logistics Engineering, Korea Maritime University, Busan 606-791, Korea*
*** Center for Research in Advanced Financial Technology, Tokyo Institute of Technology, Tokyo 152-8550, Japan*
**** Department of Systems Development, NTT Data Corporation, Tokyo 104-6591, Japan*

요 약 : 물류환경에 있어서, 데이터베이스의 연계와 데이터베이스 통합의 문제는 매우 중요한 과제로 인식되어 왔다. 그러나, 여기에 대한 빈번한 문제제기에 비해 합리적인 데이터베이스 통합방안에 관한 학술적 측면의 연구는 아직까지 매우 미흡한 실정이다. 본 연구는 효과적인 DB통합법과 관련하여 개체 및 속성 간의 유사도 측정에 기반을 둔 계량화된 충돌 식별법을 제안하는 것을 연구의 목적으로 한다. 구체적으로, DB 통합 시 빈번히 발생하는 의미적 충돌(Semantic Conflict)현상인 이른바 "Name Conflict"의 식별을 위한 하나의 해결법으로서 개체 및 속성 간 종합적인 유사도를 측정하는 계량화된 식별법을 제안하고자 한다. 그리고, 간단한 예제를 통해 제안한 방안의 유효성과 식별방안을 가늠해 보고자 한다.

핵심용어 : 물류정보시스템, 데이터베이스 통합, 스키마 통합, 의미적 충돌

Abstract : *Database integration has been recognized as a critical issue for effective logistics service in logistics environment. However, research related to effective methodology for this have been little studied, and also, prominent achievements have yet to be suggested. The aim of this paper is to present a quantitative methodology for the identification of conflict that is a representative problem on database integration. To achieve this aim, we suggested a quantitative methodology that can efficiently fine troubles such as name conflicts when schema integration, based on the level of semantic similarity between attributes and entities. And, in order to measure these semantic similarities, we used a thesaurus dictionary that proposed previous research. Finally, we presented effectiveness of the proposed methodology through some typical examples.*

Key words : *Logistics information system, Database integration, Schema integration, Semantic conflict*

1. 서 론

효율적인 물류서비스를 제공하기 위한 핵심과제로서, 종합물류망의 재정비와 함께 정보시스템의 통합에 관한 문제는 이미 오래 전부터 제기되어 왔다. 특히, 이는 동북아 물류중심을 표방하는 우리나라의 현 상황에서 시급히 개선되어야 할 사항으로서, 대부분의 관련보고서나 각종회의(IBC포럼, 2004)에서 항상 제기되고 있는 문제이다. 그러나, 거듭되는 지적에도 불구하고, 이와 관련한 구체적인 해결방안에 대한 학술적 측면의 연구는 지극히 소수인 것 또한 현실이다(한국컨테이너부두공단, 2004).

한편, 통합 물류망을 새롭게 구성하는 것과 데이터베이스(이하 DB)의 연계문제의 일환으로 현재 추진되고 있는 물류관련 DB의 통합방안은 물류 관련 업체 및 기관에서 독립적으로 사용하고 있는 DB의 구성을 파악하여, 범용성있는 통합 DB를 새롭게 구축하여, 정보의 일관성과 통합적 관리를 도모하는 것을 골자로 하고 있다. 그러나, 이는 기존의 DB를 통합하는 것이기 보다는 새로운 DB로 기존의 DB를 교체하는 것에 초점을 맞추고 있는 것으로써, 엄밀한 의미의 DB 통합이라 말하기 곤란하다. 즉, 각각의 조직에 있어 필요에 따라 구축된 기존 DB의 구조를 유지시키면서 합리적으로 통합할 수 있는 방안에 대한 접근은 모색되지 않았다. 따라서, 업계 전체적인 관점에서 설계된 통합 DB가 모두가 만족할 만한 수준의 유용한 정보를 포괄할 수 있을 지도 미지수이며, 또한 이를 위해서

* 대표저자: 이홍걸(정회원), hglee@hhu.ac.kr, 051)410-4911
** khiga@craft.titech.ac.jp, +81-3-5734-3571
*** fujikawat@nttdata.co.jp, +81-3-5546-8051

는 업계전체를 대상으로 한, 장기간에 걸친 발생 정보의 실태 파악이 무엇보다 중요하며, 결과적으로 막대한 시간과 재원이 필요하게 될 것이다.

게다가, 새롭게 통합 DB를 구축하는 방안은 지극히 현재지향적인 방편으로, 앞으로 유비쿼터스 시대가 도래함에 따라, 이른바 취급정보의 경계가 거의 없어질 전망이어서, DB가 담아 두고 있는 객체량은 현재에 비해 현저히 증가될 것으로 예측되고 있기 때문이다. 따라서, 현재의 추진방안과 같이 필요에 따라 DB를 새롭게 구축하는 방식으로 문제에 접근한다면, 가까운 미래에 또다시 새로운 구조의 DB를 구축해야할 가능성이 매우 크며, 이는 막대한 개발비용과 시간비용을 필요로 한다. 결과적으로, DB통합을 위한 사전 준비단계로써 대대적인 데이터 표준화에 대한 연구와 현행 DB의 구조를 유지시키면서 단계별로 합리적으로 통합해 나가는 방안의 수립이 효과적이고 융통성 있는 접근방법이라 할 수 있다(山室 등, 1995).

이 논문은 이러한 현 상황에 주목하여, 합리적인 DB통합법과 관련하여 개체 및 속성간의 유사도 측정에 기반을 둔 계량화된 충돌식별법을 제안하는 것을 연구의 목적으로 한다. 구체적으로, DB통합 시 빈번히 발생하는 의미적 충돌(Semantic Conflict)현상인 이른바 "Name Conflict"의 식별을 위한 하나의 해결법으로서 개체 및 속성간 종합적인 유사도를 측정하는 계량화된 식별방안을 제안하고자 한다. 그리고, 간단한 예제를 통해 제안한 방안의 유효성과 식별방안을 가늠해 보고자 한다.

2. 데이터베이스 통합상의 문제점과 기존연구

2.1. 데이터베이스 통합상의 문제점

일반적으로 다양한 조직 및 기업에서 운용하고 있는 DB는 개별 조직과 사용자의 요구에 맞게끔, 서로 다른 구조로써 정의되고 설계되어 있다. 따라서, 이러한 DB들을 통합하는 것은 상이한 데이터 정의와 구조로 인해 결코 쉬운 일이 아니다 (Batini *et al.*, 1986). 효율적인 DB통합을 저해하는 대표적인 문제점은 개체 및 속성의 의미에 있어서 충돌이 발생하는 Name Conflict 문제와 스키마 구조상에서 동일한 의미임에도 각기 다른 구조를 보이고 있는 Structural Conflict 문제이다 (Batini *et al.*, 1997). 또한, Structural Conflict에 속하는 문제로서는 동일한 실세계의 개념이 다른 데이터 구조로 표현되는 문제인 "Type Conflict", 관계형 DB에 있어 동일한 개체와 속성임에도 정의된 키(Key)가 다른 경우를 의미하는 Key Conflict가 대표적이다(Batini *et al.*, 1997). 이 이외에도 강성개체(Strong Entity)와 약성개체(Weak Entity)정의의 차이에서 비롯되는 "Weak Entity Conflict" 및 일종의 관계형(Relationship Type) 정의의 차이에서 발생되는 "Dependency Conflict" 등이 있다 (Song *et al.*, 1996).

특히, Name Conflict 문제는 개체 및 속성의 의미에 있어서의 동의성과 다의성(多義性)의 문제로써, 통합문제에 있어 해결해야할 대표적인 과제에 해당된다(Larson *et al.*,1989). 이른바, 동음이의어(Homonym), 동의어(Synonym)/유사어가 각기 다른 DB에 존재하고 있을 때, 이를 개념스키마를 바탕으로 효율적으로 식별하여 통합하지 않으면, 단순히 두개의 DB를 연계시킨 것에 불과한 통합 DB가 탄생할 수도 있고, 그와 반대로 두 가지 DB 중 일부 개체가 설계미숙으로 소실되는 문제를 야기하게 된다. 결과적으로 이를 방지하기 위해서는 무엇보다 설계자의 역량과 주의가 요구되나, 대규모 DB의 경우 이를 하나하나 대조해서 파악하기에는 막대한 시간을 필요로 하는 문제이다. 따라서, 이를 위해서는 개체 및 속성의 의미적 충돌을 효과적으로 식별하여, 통합 설계에 효율을 도모할 수 있는 방안의 마련이 필요하다 (Fong *et al.*, 1999).

2.2. 기존연구

Batini *et al.*(1986)은 일찍이 DB통합과 관련한 문제에 주목하여, DB의 개념설계에 있어, 합리적인 뷰 통합과 DB통합을 곤란하게 하는 여러 가지 요인들을 일목요연하게 정리하였다. 이러한 요인에는 '동일 대상을 바라보는 설계자의 관점의 차이', '모델에서 구성요소사이의 동치성문제' 등이 속한다. 한편, 이 연구에서는 이러한 DB간의 의미적/구조적 충돌을 식별해 내기 위한 일련의 방안을 가이드라인 형식으로 서술적으로 제시하고 있는데, Fong *et al.*(1999)의 연구에서도 이러한 접근방법을 발견할 수 있다. Fong *et al.*(1999)는 기존 DB를 재구축하는 문제에 주목하여, Batini *et al.*(1986, 1997)과 거의 흡사한 방안을 제안하였다. 또한, Larson *et al.*(1989)은 DB통합 시 발생하는 명칭, 데이터길이, 데이터구조, 데이터추상화 등과 관련한 충돌현상은 속성의 일치성, 객체 클래스의 일치성, 릴레이션십(Relationship)의 일치성의 검토를 통해 식별할 수 있다는 점을 지적하고 Batini *et al.*(1986)과 비슷한 방식의 식별법을 제안하였다. 그러나, 상기의 연구들에서 제안한 통합법은 결과적으로 통합에 임하는 설계자가 제안된 가이드라인을 충분히 숙지한 후, 각 DB내에 정의된 객체들을 하나하나 비교/검토해야 하는 과정을 수반하기 때문에, 비효율적인 접근방법이라는 지적을 받아 왔다.

한편, 일본에서는 DB 통합이 사회적으로 매우 필요한 과제로 대두됨에 따라, DB 통합의 효율성을 도모하기 위한 선결과제로서 데이터 표준화의 필요성에 주목하였다. 黑川 등(1993), 關根 등(1993), 川下 등 (1992), 山室 등 (1995) 은 데이터 표준화와 더불어 용어사전과 유사한 데이터 사전을 만들어 통합시 발생되는 충돌현상을 사전에 예방하고, 용어사전을 참고하여 통합시에 문제점을 완화시키는 방안을 제안하였다. 또한, 이와 유사하게 통합을 위한 Metadata를 사전에 구축하여 통합 시 효율을 도모하는 방안도 고려되고 있다(Tseng, 1998). Song *et al.*, (1996)의 연구도 이와 유사한데, 특히 정량적인 수치로 개체간의 의미적 충돌정도를 파악할 수 있다는 점에서 기존연구에 비해 진전된 방법론이라 말할 수 있다. 이 연구에 의하면, 계층적 분류법을 기반으로 데이터 의미사전(Semantic Dictionary)을 만들어, 설계자가 마련된 사전을 통해 데이터의

유사도를 쉽게 파악할 수 있게 하였다. 이와 같은 접근방법은 앞서 언급한 Batini를 필두로 한 방안에 비해 용어사전을 통해 정량적인 유사성을 참고할 수 있어 통합 시 효율적이나, 범용적인 용어사전을 얼마나 완벽하게 구축하느냐가 중요한 관건이 된다. 게다가, 어떤 용어를 계층적으로 상위와 하위계층으로 구분하여 분류한다는 것은 결코 쉽지 않은 것으로써 전문성을 요하는 작업이다.

3. 속성 및 개체간 유사도 측정방법의 제안

3.1. 형태소 분석과 시소러스사전의 구축

시소러스 사전이란 어휘의 의미에 따라 분류/배열한 일종의 어휘집이다. DB의 데이터가 지닌 "의미"의 차이를 식별하기 위해, 이러한 형태의 용어사전은 사전에 수립되어야 할 필수적인 과정이다. 특히, 시소러스 사전의 분류방식에 따라, 유사성 식별의 효율성이 좌우되며, 게다가 영어권 국가처럼 우리와 언어체계가 다른 나라에서 구축된 시소러스 사전은 분류방식이 달라, 이를 그대로 활용할 수 없다. 따라서, 효율성 높은 시소러스 사전을 구축하기 위해서는 무엇보다 우리나라의 언어체계를 파악하여, 시소러스 사전의 분류체계에 반영할 필요가 있다.

한편, DB내에 정의된 개체와 속성의 유사성 판단에 있어, 효율성을 도모하기 위해서는 용어자체의 분류체계의 구조에 주목할 필요가 있다. 우리말과 동일한 언어구조를 지니고 있는 일본에 있어, 관련연구(黒川 등, 1993; 關根 등, 1993;, 川下 등, 1992)의 결과를 참조해 보면, 일반적으로 우리말과 같은 언어체계에서는 데이터 항목명칭의 용어는 단일어인 경우와 복합어인 경우로 크게 구분할 수 있다. 여기서, 복합어 형식으로 정의된 데이터 항목인 경우가 문제인데, 이 경우 기존연구(關根 등, 1993; 川下 등, 1992;, 山室 등, 1995) 에 의하면, 「수식어+주요어+구분어」의 순서로 용어가 배열될 때, 데이터 항목을 명확히 정의할 수 있는 표준적인 어순이 되며, 용어들은 이러한 형식으로 구분될 수 있다. 즉, 이는 일종의 형태소분석에 해당되는 것으로써, 예를 들어 「주문고객코드」라는 데이터 항목이 있다면, 형태소로 분리해 보면 주문/고객/번호가 된다. 여기서 「주문」은 데이터 항목의 주요어에 해당하는 「고객」을 보충설명해주는 일종의 수식어의 역할을 하며, 「코드」라는 것은 고객을 구분해 주는 구분어의 역할을 하게 된다.

黒川 등(1993)의 연구에 의하면, 용어는 앞서 언급한 「수식어+주요어+구분어」(이하 "수주구"로 명명함), 「수식어+주요어」(이하 "수주"), 「수식어」(이하 "수")의 세 종류에 해당하는 역할을 한다. 즉, 용어는 언어체계상 일반적으로 상기의 세 가지 형태로 그 역할을 구분할 수 있다는 것을 의미하는 것으로써, 이렇게 구분된 용어들은 유사도 식별 시 해당 형태소 종별만을 체크하면 되므로, 대규모 시소러스사전을 통한 유사어 및 유사성 검색에 효율을 도모할 수 있게 된다. 한편, 기존연구에서는 이러한 체계에 부합하지 않는 데이터 항목이 실제 구축된 DB내에 많이 존재하고 있다는 점을 지적하고 있다. 특히, 이는 DB통합 시 가장 문제시되는 요인으로써 DB 설계자가 정의한 데이터의 애매성에 기인한다. 따라서 이러한 문제를 사전에 예방하고 DB 통합에 효율성을 높이기 위해서는 세 가지 분류유형에 대해 DB 설계자가 숙지한 후, DB내 데이터 항목을 수정하는 것이 바람직하다.

사실, 형태소 분해에 기초한 시소러스사전의 구축은 등록 시 데이터량을 현저히 줄일 수 있게 한다(黒川 등, 1993). 예를 들어, 동의어에 해당하는 「종업원번호」, 「종업원코드」, 「사원번호」, 「사원코드」라는 속성이 있고, 이를 종래의 방식인 데이터항목에 의거하여 시소러스사전을 작성한다면, 각각의 데이터항목과 동의어를 하나하나 등록해 두어야 하므로, 6가지 동의어 정보의 등록이 필요하게 된다. 그러나, 여기서 제안한 형태소 분석에 따른 시소러스사전 작성법으로는 형태소의 종별로 2가지 '종업원과 사원', '번호와 코드'만을 중심으로 입력하면 되므로, 훨씬 효율적이다(黒川 등, 1993).

따라서, 본 연구에서는 유사성 측정을 위해 상기의 기존연구(黒川 등, 1993)에서 제안한 형태소 분석에 의거한 시소러스사전을 참고 한다.

3.2. 속성(Attribute)간의 유사도 정의

3.2.1. 속성간 의미관계 정의

3.1절에서 제시한 형태소별 구분과 그 분류체계가 구축되면, 시소러스사전에 등록된 구분어와 주요어의 유사성 정보를 토대로 속성간의 의미관계를 다음과 같이 정량적으로 표현할 수 있다. 여기서 *dic* 을 속성간 의미관계를 나타내는 변수로 정의한다.

dic = 1 : 속성명을 구성하는 구분어와 주요어가 각각 동일 혹은 동의어 관계일 경우

dic = 0.75 : 속성명을 구성하는 주요어가 유사어이며, 그리고 구분어가 동일 혹은 동의어 관계일 경우

dic = 0.25 : 속성명을 구성하는 구분어가 동일 혹은 유사어이며, 그리고 그 외의 용어는 동일하거나, 동의어/유사어의 관계가 아닐 경우

dic = 0 : 상기 이외의 경우

사실, 본 연구에서는 식별치를 간단하고 명확하게 하기 위해서 네 가지 경우로 한정하여 의미관계 값을 설정하고 있는데, 보다 정밀하게 식별치를 파악하기 위해서는 시소러스 사전에 유사어의 유사정도를 미리 파악하여 등록해 놓으면 된다. 그러나, 의미관계 값을 지나치게 세분화시키는 것은 그만큼 통합 시 충돌되는 항목이 그 정도에 따라 많아지게 되어, 정밀성은 높일 수 있으나 효율성 측면에서의 효과는 낮아진다.

3.2.2. 속성간 유사도 정의

속성간 유사도 Sim_{atr} 은 다음과 같이 정의된다.

$$\sim_{atr} = \begin{cases} 1 & (dic = 1) \\ \dfrac{dic \times w_d + type \times w_t + length \times w_l}{w_d + w_t + w_l} & (0 \le dic < 1) \end{cases} \quad (1)$$

단, *type* : 각 속성에 해당되는 데이터형의 일치정도
(*type = 0 or 0.5 or 1*)
length : 각 속성에 정의된 데이터 길이의 일치 정도
(*length = 0 or 1*)
w_d, w_t, w_l : 각각 *dic*, *type*, *length*의 가중치
$w_d + w_t + w_l = 1$

여기서, *type* 은 각 속성에서 정의된 데이터 형의 일치정도를 나타내는 변수이다. 실제, DB 통합 시 반드시 고려되어야 할 사항이라고 볼 수는 없지만, 데이터 형의 대분류가 다른 경우 (예: 속성 A가 수치 데이터로 정의된 반면, A'는 날짜/시각형으로 정의된 경우), 통합시 문제를 일으킬 수 있는 여지가 있다. 따라서, 본 연구에서는 데이터형의 완전히 일치하는 경우는 1로 두고, 대분류 카테고리가 같은 경우 0.5, 대분류 카테고리조차 일치하지 않는 경우 0으로 일치 정도를 구분하였다. *length* 의 경우도 실제 유사성에 미치는 정도는 그다지 크지 않으나, 그렇다고 완전히 배제할 정도는 아니므로, 일치하는 경우와 일치하지 않는 경우로 구분하여 각각 1, 0의 값을 가지게 하였다.

결과적으로, 속성간 유사도 Sim_{atr} 은 1이거나 1에 가까워질수록 유사성이 높다는 것을 의미하는 변수로써, 속성간 의미관계의 정도를 나타낸 *dic* 의 비중이 앞서 언급한 바와 같이 *type* 과 *length*에 비해 매우 높으며, 상황에 따라서는 *type* 과 *length*는 유사성 판단의 고려대상에서 제외될 수도 있다. 따라서, 각각의 변수에 가중치 파라메터를 추가하여, 이를 조절할 수 있게끔 하였으며, 최종적으로는 설계자가 탄력적으로 해당 상황에 적합한 가중치를 부여하면 된다.

3.3. 개체(Entity)간 유사도 정의

3.3.1. 개체 명(Entity Name)의 일치성

각 개체명의 일치성을 나타내는 변수 *name* 은 다음과 같이 정의된다.

name = 1 : 개체명이 일치하거나 동의어인 경우
name = 0 : 그 외의 경우

개체명의 경우도 속성명의 의미관계를 나타내는 *dic* 과 같이, [0, 1]사이의 다양한 값으로 나타낼 수 있다. 그러나, 개체는 속성들로 표현되는 것으로써, 개체명 자체가 가지는 유사성의 정도보다, 각 개체와 거기에 속하는 속성들이 지닌 유사도와의 관련성이 더욱 중요하다. 이러한 관련성은 최종적으로 Name Conflict를 판정하는 수단이 된다. 즉, *name*의 값을 0과 1로 둠으로써, 보다 명확히 식별할 수 있는 수단이 되는데, 거기에 대해서는 4장에서 예제와 함께 자세히 설명된다.

3.3.2. 각 개체의 공통속성의 비율

각 DB가 지닌 공통속성의 비율은 개체의 최종적인 유사성을 식별하는 데 주요한 영향을 미친다. 따라서, 기존 연구(Song, 1996)에서도 공통속성 비율을 Name Conflict 판단의 중요 기준으로 두고 있다. 그러나, 기존 연구에서는 공통속성을 각 개체에 귀속된 속성들이 완전히 동일 한 경우에만 공통속성으로 하고 있어, 동일하지는 않지만 유사성이 높은 속성이 존재하는 경우 식별이 불가능하였다.

따라서, 본 연구에서는 공통속성 비율을 판정하기 위한 변수로, 식 (1)을 활용하고자 한다. 즉, 본 연구에서의 공통속성은 각 개체에 속하는 속성들 사이에서 동일하거나 유사한 속성집합을 의미하며, 그 값은 Sim_{atr} 을 통해 구해진다. 한편, Sim_{atr} 은 유사한 정도에 따라 [0, 1]의 값으로 표현되므로, 이 중에서 유사하다고 판단되는 속성들을 추출하여 공통속성집합을 구성하기 위해서는 어떤 기준이 필요하다. 본 연구에서는 편의상 유사성 임계값을 0.5로 두며, 이는 DB 통합 시 상황에 따라, DB설계자가 임의로 조절할 수 있다. 다만, 이러한 임계값이 낮아질수록 공통속성의 비율이 그 만큼 많아지게 되므로 유의할 필요가 있다.

공통속성 비율 com_{atr} 은 다음과 같이 정의된다.

$$com_{atr} = \frac{2 \sum_{(x,y)} \sim_{atr}(x,y)}{n(A) + n(B)} \quad (\text{if } \sim_{atr}(x,y) \ge 0.5) \quad (2)$$

단, *n(A)* : A라는 개체의 속성 수
n(B) : B라는 개체의 속성 수
x : 개체 A의 속성
y : 개체 B의 속성
$Sim_{atr}(x, y)$: 속성 *x* 와 *y*의 유사도

3.3.3. 개체간 유사도 정의

최종적인 개체간 유사도 Sim_{ent} 는 다음과 같이 정의된다.

$$\sim_{ent} = \frac{name \times w_n + com_{atr} \times w_c}{w_n + w_c} \quad (3)$$

단, w_n, w_c : 각각 *name*, com_{atr}의 가중치
$w_n + w_c = 1$

개체간 유사도 Sim_{ent} 는 3.2.2에서 제시한 식(1)과 유사한 형태를 보이고 있으며, 가중치의 존재이유와 부여방법 역시 3.2.2에서 설명한 것과 동일하다. 결과적으로 최종적인 Name Conflict의 식별은 식(3)을 통해서 판별되나, 식(3)을 구하기 위해서는 앞서 언급된 모든 계산과정을 거쳐야만 한다.

이상, Name Conflict 식별을 위한 전체적인 산출과정을 정리하면, 다음의 Fig. 1과 같다.

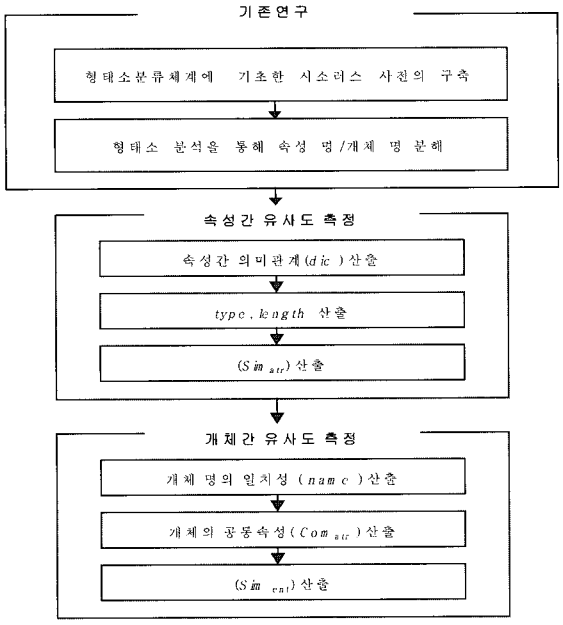


Fig. 1 Flowchart of Quantitative Methodology

4. 적용 예제와 식별방안

전형적인 Name Conflict 문제를 일으키는 간단한 예제를 통해, 제안한 방법론을 이용한 식별방안과 그 유효성을 가늠해 보고자 한다. 단, 본 예제에 해당하는 시소러스사전은 기존연구(關根 등,1993)를 참조한 것이며, 다음과 같이 사전에 마련되어 있는 것으로 가정한다.

참고로, Table 1의 시소러스사전은 어디까지나 다음 절부터의 예제에 적용하기 위해 간단히 구성된 것으로써, 실제 제대로 된 시소러스사전의 정확성을 가지고 있는 것은 아니다.

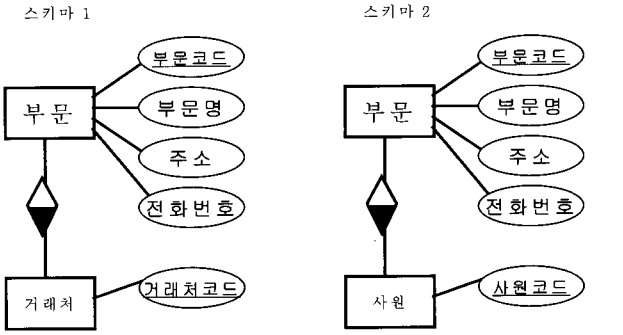
Table 1 Thesaurus for Examples

구 분	동의어	유사어	용어종별
부문	부서,부,계열	과,계,지역	수주
코드	번호,기호,넘버	암호,국번지역	수주구
명	명칭,이름,성명	별명,약칭,상호	수주구
주소	번지,현주소,근무처	소재지,장소	수주구
전화	TEL,폰,다이얼	내선,휴대폰	수주
종업원	사원,직원,피고용인	사무원,점원	수주
FAX	팩시밀리		수주
우편	편지,메일	소포,택배	수주
번호	코드,넘버,#,No	국번	수주구
거래처	고객	업체,협력업체	수주

4.1. Name Conflict의 식별

4.1.1. 동음이의((Homonym)의 식별

Fig. 2는 DB통합 시 전형적인 동음이의 문제를 일으키는 예제이다.



스키마	개체	속성명	데이터형	데이터길이
1	부문	부문코드(PK)	문자형	4
		부문명(AK)	문자형	40
		주소(AK)	문자형	40
		전화번호	문자형	12
	거래처	거래처코드(PK)	수치형	15
2	부문	부문코드(PK)	수치형	4
		부문명(AK)	문자형	30
		주소(AK)	문자형	30
		전화번호	문자형	30
	사원	사원코드(PK)	수치형	20

(단, PK: Primary Key, AK: Alternative Key)

Fig. 2 Conceptual Schema 1 and 2 with Homonym

각 스키마의 부문은 동일한 속성을 가지고 있으나, 스키마1의 경우 「부문」은 거래처라는 개체와 릴레이션십으로 연계되어 있고, 스키마2의 경우 사원이라는 릴레이션십과 연계되어 있다. 즉, 스키카1의 부문이라는 것은 거래처와 관계된 부문이고, 스키마2의 경우 자사부문이라는 것을 알 수 있다. 따라서, 통합 시 주의를 기울이지 않으면 서로 같은 개체로 착각하여 서로 다른 성격의 개체인 「부문」을 합쳐 버리게 되는 경우가 발생하게 된다.

Name Conflict 식별을 위해 속성간의 의미관계인 *dic* 값을 산출하기 위해서는 기존연구(黑川 등, 1993)의 제안에 따라, 우선 형태소 분석을 통해 각 속성을 분해해야 한다. 그 결과는 Table 2 와 같다.

Table 2 Decomposition of Attributes

속성명	수식어	주요어	구분어
부문코드	-	부문	코드
부문명	-	부문	명
주소	-	(부문)	주소
전화번호	(부문)	전화	번호
거래처코드	(부문)	거래처	코드
사원코드	(부문)	사원	코드

Table 3 Calculation of *dic, type,* and *length* between attributes

스키마1 / 스키마2	부문코드			부문명			주소			전화번호		
부문코드	*d*	*t*	*l*	*d*	*t*	*l*	*d*	*t*	*l*	*d*	*t*	*l*
	1	0	1	0	0	0	0	0	0	0.25	0	0
부 문 명	*d*	*t*	*l*	*d*	*t*	*l*	*d*	*t*	*l*	*d*	*t*	*l*
	0	1	0	1	1	0	0	1	0	0	1	0
주 소	*d*	*t*	*l*	*d*	*t*	*l*	*d*	*t*	*l*	*d*	*t*	*l*
	0	1	0	0	1	0	1	1	0	0	1	0
전화번호	*d*	*t*	*l*	*d*	*t*	*l*	*d*	*t*	*l*	*d*	*t*	*l*
	0.25	1	0	0	1	0	0	1	0	1	1	0

(단, *d: dic, t: type, l: length*)

Sim_{atr} 을 산출하기 위해, 각 변수들을 계산한 결과는 Table 3과 같으며, 그 결과를 토대로 식(1)에 의거 각 스키마의 「부문」에 속하는 Sim_{atr}(가중치: w_d =0.7, w_t =0.2, w_l =0.1)을 산출한 결과는 Table 4와 같다. 여기서, 식(3)에 따라 최종적인 개체 간 유사도를 산출하면, Sim_{ent} = 1(가중치: w_n =0.2, w_c =0.8)이며, 개체 명 일치성 *name* 은 1.00, 공통속성 비율 com_{atr} 은 1.00이 된다.

Table 4 Sim_{atr} between attributes in entities with homonym conflict

스키마2 / 스키마1	부문코드	부문명	주소	전화번호
부문코드	1.00	0.00	0.00	0.18
부 문 명	0.20	1.00	0.20	0.20
주 소	0.20	0.20	1.00	0.20
전화번호	0.18	0.20	0.20	1.00

덧붙여, 동일한 방식으로 각 스키마의 개체 「부문」과 릴레이션십으로 연계된 「거래처」와 「사원」을 계산하면, *name* = 0.00, com_{atr} = 0.00, Sim_{ent} = 0.00이 된다.

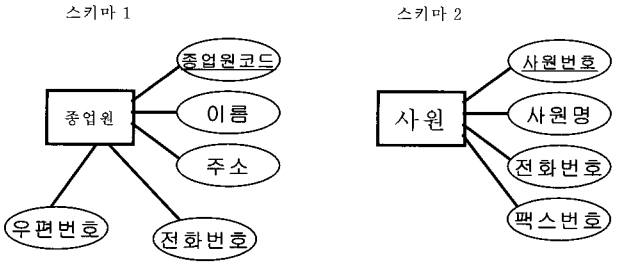
한편, 예제와 같은 형태를 취하는 동음이의 문제의 식별방안은 동음이의의 경우, 우선 개체명이 같으므로, *name* = 1.00 이 된다. 그러나, 예제와 같이 릴레이션십으로 연계된 개체는 *name* = 0.00 이면서, 개체사이의 유사도인 Sim_{ent} 의 값도 그다지 크지 않게 나타나게 된다. 그리고, 당연히 공통속성 비율인 com_{atr}도 크지 않다. 이런 경우 동음이의 형태의 Name Conflict를 의심해 볼 필요가 있다. 참고로, 예제와 같은 경우 DB 설계자가 이러한 방법으로 동음이의 문제를 식별했다면, 스키마1의 「부문」을 「거래처 부문」으로, 스키마 2의 「부문」은 「자사부문」으로 개체의 명칭을 변경하면 간단히 해결될 수 있다.

덧붙여, 예제와 다르게 릴레이션십으로 연계된 개체가 아니라, 단일개체사이에서도 동음이의 문제가 야기될 수 있는데, 이 경우의 식별은 더욱 간단해진다. 우선, 개체명이 같으므로, *name* = 1.00이지만, 정작 개체를 설명하는 속성들 사이는 유사도가 높지 않을 것이므로 Sim_{ent} 값은 낮을 가능성이 크다. 그러나, Sim_{ent} 의 *name* 변수에 가중치를 예외적으로 상당히 높게 부여한 경우 Sim_{en} 값만으로 식별이 곤란할 수 있다. 따라서, 이 경우 보다 확실한 것은 com_{atr} 의 값을 참조해 보면 알 수 있는데, 공통속성 비율인 com_{atr} 값이 낮게 나타나는 것이 일반적이다. 이상, 동음이의 문제의 식별방안을 요약하면 다음과 같다.

> 〈Name Conflict: 개체간 동음이의 식별〉
> *name* = 1.00이지만 해당 개체(혹은 해당 개체와 릴레이션십으로 연계된 개체)의 Sim_{ent} 특히 com_{atr} 가 그다지 크지 않은 경우

4.1.2. 이음동의(Synonym)의 식별

다음의 Fig. 3은 DB통합 시 이음동의의 문제를 일으키는 전형적인 예제이다. 이 경우, 개체의 속성내용이 동일함에도 불구하고, 통합 시 설계자의 부주의로 별개의 개체로 취급될 수 있다.



스키마	개체	속성명	데이터형	데이터길이
1	종업원	종업원코드(PK)	문자형	4
		이름(AK)	문자형	40
		주소(AK)	문자형	40
		전화번호	문자형	7
		우편번호	문자형	12
2	사원	사원번호(PK)	수치형	4
		사원명(AK)	문자형	30
		전화번호	문자형	30
		팩스번호	문자형	30

(단, PK: Primary Key, AK: Alternative Key)

Fig. 3 Conceptual Schema 1 and 2 with Synonym

4.1.1절의 예제와 같은 산출과정을 통해 Sim_{atr} 를 계산한 결과를 정리하면 Table 5와 같다.

계산과정에서 사용되는 가중치는 4.1.1 절과 동일한 가중치를 부여하였고, 또한 구체적인 산출과정 역시 4.1.1절과 같으므로 중간 계산과정 및 결과에 대한 설명은 생략해도 무방할 것으로 사료된다.

따라서, 이 경우 최종적인 계산 결과는, Sim_{ent}= 0.664, *name* = 0.00, com_{atr} = 0.83으로 나온다. 한편, 이음동의 문제의 식별

방안은 동음이의 충돌문제와 정반대의 특성을 가지고 있는데, 우선, *name* = 0 이지만, Sim_{ent} 가 적어도 0.5 이상의 값을 가지게 되거나, 특히 com_{atr} 값이 높게 나타날 때 이음동의의 충돌이 발생했을 가능성이 높다.

Table 5 Sim_{atr} between attributes in entities with synonym conflict

스키마2 / 스키마1	사원번호	사원명	전화번호	팩스번호
종업원코드	1.00	0.00	0.38	0.38
이 름	0.00	1.00	0.20	0.20
주 소	0.00	0.20	0.20	0.20
우편번호	0.18	0.20	0.38	0.38
전화번호	0.18	0.20	1.00	0.73

다만, 이 경우도, Sim_{ent} 내의 변수에 부여한 가중치에 따라 Sim_{ent} 의 크기가 달라질 수 있는데, 이러한 경우 com_{atr} 값을 참조해 보면, 이러한 충돌현상을 쉽게 파악할 수 있게 된다.

〈Name Conflict: 개체간 이음동의의 식별〉

name = 0.00이지만 해당 개체의 Sim_{ent} 가 적어도 0.5 이상이고 특히 com_{atr} 가 큰 경우

4.2. 활용방안

4.1절에서는 본 연구에서 제안한 계량적 식별방안을 전형적인 간단한 예제를 토대로 제시하였다. 따라서, 제안한 계량적인 방법은 실제 통합 DB설계 시 Name Conflict 식별에 있어 효율을 도모할 수 있을 것으로 기대된다. 특히, 본 예제에서와 같은 충돌개체를 포함한 수십개 이상의 개체로 구성된 실질적인 복잡한 스키마간 통합문제일수록 제안된 계량적 식별법은 그 효과를 발휘할 수 있을 것으로 사료된다.

한편, 제시한 예제는 어디까지나 식별여부와 식별방안 자체를 제시하기 위해 만들어진 전형적인 예제이다. 즉, 계량적 식별방안의 실질적인 효율성을 도모하기 위해서는 식별법을 적용함에 있어 수작업이 아닌 기계적으로 소위 "충돌우려 개체"를 간단히 추출해 낼 수 있는 방안의 도입이 필요하다. 여기에는 여러 가지 방안이 있을 수 있으나, 본 연구에서는 하나의 방편으로써 Fig. 4와 같은 절차의 클러스터링 기법을 제안하고자 한다.

즉, 이는 제안된 유사도 측정과정을 토대로 최종적인 Sim_{ent} 가 구해지면, 이러한 개체간 유사도를 토대로 군집화하는 것을 의미한다. Sim_{ent}은 유사성이 높을수록 1에 가까운 값을 가지게 된다. 따라서, Fig. 4와 같이, 전체 개체 수에 대한 Sim_{ent} 의 평균을 구해, 그것을 임계값으로 하여 군집분석을 행하면, 각 스키마 사이에서 충돌우려 개체가 하나의 군집으로 묶이게 된다. 결과적으로 통합 설계 시 설계자는 개체 하나하나를 대조할 필요없이 군집화된 개체들만 참고하여 충돌여부를 식별하면 되므로 통합의 효율성을 높일 수 있을 것으로 기대된다.

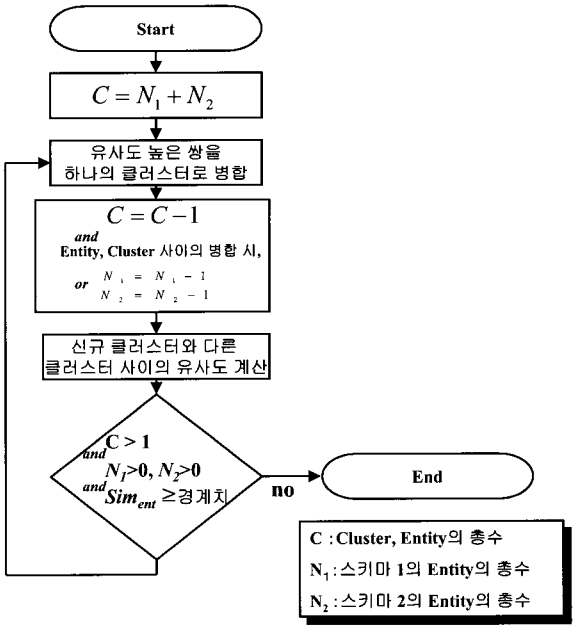


Fig. 4 Flowchart of Clustering

5. 결 론

본 연구는 현재 물류분야에 있어, 물류정보시스템의 통합화의 중요한 과제로서 빈번히 지적되고 있는 DB 통합의 문제를 연구의 대상으로 하였다. 그리고, 이러한 DB 통합과 관련하여 합리적이고 효율적인 통합을 저해하는 대표적인 요인으로 알려져 있는 Name Conflict 문제에 주목하여 그것을 효율적으로 식별할 수 있는 하나의 계량적인 방안을 제시하였다. 특히, 이는 비단 물류분야뿐만이 아니라, 정보화의 진전과 유비쿼터스 개념의 등장으로 인해, 정보량과 정보의 경계가 비약적으로 확대되어감에 따라, 이러한 정보들을 통합적으로 관리하고 정보간의 일관성을 도모하기 위해, 우리사회전반에 걸쳐 인식을 같이 하고 있는 하나의 절실한 과제이기도 하다. 따라서, 본 연구의 결과는 효율적인 DB통합과 관련한 문제에 있어, 부분적이나마 하나의 참고자료로서 역할을 할 수 있으리라 기대된다.

한편, 본 연구는 몇 가지 한계점을 가지고 있다. 우선, 본 연구에서 제안한 유사도 측정법은 Song *et al.* (1996)의 계량적 유사도 측정방안의 일부와 접근법을 개선한 형태를 취하고 있으며, 여기에 덧붙여, 여러 번의 반복 실험을 통해 충돌개체의 명확한 분류가 가능하게끔 수식을 수립하였다. 그러나, 이러한 접근은 대수학적 측면의 타당성을 확보하는데 문제점이 될 수 있다. 또한, 수립된 기존연구에서의 의미적 유사도 측정방안과의 비교를 통해, 본 연구에서 제안한 식별법의 차별성 및 유효성을 제시하지 못하여, 이러한 과정을 통해 제안한 방법의 재검토와 보완의 절차가 필요할 것으로 사료된다. 다음으로, 본 연구에서는 유사도 식별을 위한 계량적 방안을 제안하는데 주안점을 둔 나머지, 유사도 측정을 위해 필요한 실질적인 시소러스사전의 구축은 이루어지지 않았으며, 기존 연구를 참조하는 것을 전제로 하고 있다. 따라서, 실질적인 시소러스 사전의

구축을 통해 제안한 방안의 실질적인 유효성을 제시할 필요가 있다. 덧붙여, 본 연구는 어디까지나 DB통합의 여러 가지 과제들 중 일부분에 해당하는 것으로써, 실제 DB통합문제를 실무적인 차원에서 효율적으로 처리하기 위해서는 또한 다음과 같은 다양한 과제가 아직까지 남아있다. 첫째, 본 연구에서는 Name Conflict 문제를 중심으로 DB통합의 충돌식별방안을 수립하였으나, 합리적인 스키마통합을 하기 위해서는 각 스키마 사이의 구조적인 측면의 충돌문제(Structural Conflict)도 고려되어야 한다. 특히, 각 스키마의 주키(혹은 대체 키)사이에서 충돌이 발생하는 Key Conflict 문제와 같은 경우 합리적인 통합을 저해하는 주요한 원인으로 작용하게 되므로, 여기에 대한 식별방안의 수립이 필요하다. 둘째, 전반적인 충돌식별방안이 수립되면, 이를 실제 문제에 적용하여 실질적인 스키마통합을 수행해 봄으로써, 그 효과와 문제점을 파악하고 보완할 필요가 있다. 마지막으로, DB통합을 보다 효과적으로 지원하기 위해서는 제안한 계량화된 식별법을 실제 DB에 실장하는 방안이나 통합설계자를 지원하는 응용S/W의 구축이 필요하다. 특히, 제안한 방법론에 있어 임계값 및 유사도의 높고 낮음의 문제는 다소 통합설계자의 주관적 판단과 식견을 필요로 한다. 즉, DB 통합의 편의성과 효율성을 더욱 높이기 위해서는 이러한 부분까지 기계적으로 해석할 수 있는 계량적 방안의 수립이 요구된다.

따라서, 상기의 여러 가지 문제들이 향후 추진되어야 할 본 연구의 중요한 과제로 남아있다고 말할 수 있다.

참고문헌

[1] 한국컨테이너 부두공단 (2004), 상해(대소양산) 및 북중국 항만의 발전이 미치는 영향과 대응방안 연구.

[2] 川下滿, 關根純, 中川優, 黑川淸 (1992), 大規模DBのためのデータ標準化手法, NTT R&D, Vol.41, No. 12, pp.1425-1432.

[3] 關根純, 川下滿, 町愿宏毅 中川優(1993), 體系的DB構築のための用語辭典を用いたデータ標準化手法, 情報處理學會論文誌, Vol. 34, No. 3, pp.457-467.

[4] 關根純, 川下滿, 中川優 (1992), DB設計を支援する情報資源辭典システムの操作機能と實現法 情報處理學會論文誌, Vol. 33, No. 4, pp.532-542.

[5] 黑川淸, 中川優, 關根純 (1993), 複合語解析技術を用いたデータ項目名稱の標準化手法, 情報處理學會論文誌, Vol. 34, No. 3, pp.447-456.

[6] 山室雅司, 川下滿, 中川優 (1995), データベースエンジニアリングへの知識處理技術の適用, 人工知能學會誌, Vol.10, No. 1, pp.31-37.

[7] Batini, C., Ceri, S., and Navathe, S.B.(1997), "Conceptual Database Design: An Entity-Relationship Approach", Benjamin/Cummings Publishing Company Inc.

[8] Batini, C., Lenzerini, M., and Navathe, S.B.(1986), "A Comparative Analysis of Methodologies for Database Schema Integration", ACM Computing Surveys, Vol. 18, No. 4, pp.650-663.

[9] Fong, J., Karlapalem, K., Li, A., and Kwan, I. (1999), "Methodology of Schema Integration for New Database Application: A Practitioner's Approach", Journal of Database Management, Vol. 10. No. 1, pp.3-18.

[10] IBC포럼 (2004), 동북아 물류중심지 개발전략: 순차적 실천방안 및 실행체계 구축.

[11] Larson, J.A., Navathe, S.B., and Elmasri, R.(1989), "A Theory of Attribute Equivalence in Databases with Application to Schema Integration", IEEE transaction on Software Engineering, Vol. 15, No. 4, pp.449-463.

[12] Song, W.W., Johannesson, P., and Bubenko, J.A. (1996), "Semantic Similarity Relations and Computation in Schema Integration", Data & Knowledge Engineering, 19, pp.65-97.

[13] Tseng, F.S.C., Chiang, J.J., and Yang, W.P. (1998), "Integration of Relations with Conflicting Schema Structures in Heterogeneous Database Systems", Data & Knowledge Engineering, 27, pp.231-248

원고접수일 : 2005년 5월 3일
원고채택일 : 2005년 6월 28일