Journal of The Korea Society of Computer and Information
Vol. 23 No. 3, pp. 41-48, March 2018
www.ksci.re.kr
https://doi.org/10.9708/jksci.2018.23.03.041

Development of the Unified Database Design Methodology for Big Data Applications - based on MongoDB -

Junho Lee*, Kyungsoo Joo**

Abstract

The recent sudden increase of big data has characteristics such as continuous generation of data, large amount, and unstructured format. The existing relational database technologies are inadequate to handle such big data due to the limited processing speed and the significant storage expansion cost. Current implemented solutions are mainly based on relational database that are no longer adapted to these data volume. NoSQL solutions allow us to consider new approaches for data warehousing, especially from the multidimensional data management point of view. In this paper, we develop and propose the integrated design methodology based on MongoDB for big data applications. The proposed methodology is more scalable than the existing methodology, so it is easy to handle big data.

▸Keyword: Database, NoSQL, MongoDB, Design Methodology

I. Introduction

최근 데이터의 급증으로 생겨난 분석 데이터는 기존의 관계형 데이터베이스로는 처리하기 어려운 크기로 확대되었다[1]. 또한, 기존의 관계형 데이터베이스는 확장성이 좋지 않아 빅 데이터를 처리하기 위한 데이터베이스로 적당하지 않다[2, 3, 4].

그러한 문제점을 해결하기 위해 NoSQL(Not Only SQL)이 등장하였고, 이는 새로운 가능성을 제시한다[5]. NoSQL 데이터베이스 중 MongoDB는 분산 확장을 지원하여 빅 데이터 처리에 용이한 데이터베이스이다. 따라서 빅 데이터를 위한 OLTP(On-Line Transaction Processing) 및 OLAP(On-Line Analytical Processing)는 관계형 데이터베이스가 아닌 MongoDB 기반으로 구축되어야 바람직하다.

본 논문에서는 NoSQL 중 하나인 MongoDB를 기반으로 빅 데이터 응용을 위한 MongoDB 기반의 대규모 OLTP 및 OLAP 구축을 위한, 정보 요구사항 분석부터 OLTP 및 OLAP 구축까지 일관된 설계 방법론을 개발하여 통합 데이터베이스 설계 방법론을 제안한다. 제안한 방법론으로 MongoDB 기반의 대규모 데이터베이스를 구축하기 위한 OLTP 데이터 셋은 통계청에서 제공하는 교사 및 학교 관리자를 대상으로 한 기초학력 진단-보정 시스템 설문조사 응답 내용을, OLAP 데이터 셋은 판매 정보 데이터를 사용하였다.

본 논문의 구성은 다음과 같다. 2장에서는 OLTP 및 OLAP 그리고 MongoDB에 대해 기술하고, 3장에서는 본 논문에서 제안하는 빅 데이터 응용을 위한 통합 데이터베이스 설계 방법론을 설명한다. 그리고 제안한 방법론에 실제 데이터를 적용하여 데이터베이스를 구축하고, 구축된 MongoDB 컬렉션을 보여준다. 4장에서는 결론 및 향후 연구에 대해 기술한다.

II. Related works

2.1 OLTP(On-Line Transaction Processing)

OLTP(On-Line Transaction Processing)는 네트워크상 다수의

• First Author: JunHo Lee, Corresponding Author: KyungSoo Joo
*Junho Lee (wnsgh461@naver.com), Dept. of Computer Science, Soonchunhyang University
**Kyungsoo Joo (gsoojoo@naver.com), Dept. of Computer Software Engineering, Soonchunhyang University
• Received: 2018. 01. 25, Revised: 2018. 02. 01, Accepted: 2018. 02. 28.
• This work was supported by the Soonchunhyang University.

사용자가 실시간으로 데이터베이스의 조회, 갱신처리 등의 트랜잭션을 처리하는 것으로 정의 할 수 있다. 사용자가 최신 데이터를 실시간으로 접근하는 것이 가능하고, 즉시 수정이 가능하다. 즉, 트랜잭션을 온라인으로 처리하는 것을 OLTP 시스템 이라고 한다[6, 7].

2.2 OLAP(On-Line Analytical Processing)

OLAP(On-Line Analytical Processing)는 OLTP에 상대되는 개념으로 볼 수 있다. 데이터 웨어하우스에서 주로 사용되고 있으며 사용자가 다차원 정보에 직접 접근하여 정보를 분석하고 의사결정에 활용 할 수 있다[7, 8].

OLAP 시스템의 핵심은 사용자의 의사결정을 지원하는 다차원 정보 분석이라 할 수 있다. 일반적인 다차원 정보 분석을 위해 스타 스키마를 사용한다.

스타 스키마는 정보를 사실과 차원으로 분류하는데, 사실은 실제 데이터 요소로 분석을 요하는 항목들을 말한다. 차원은 사실을 보는 관점을 나타내며 각각의 차원은 별도의 차원 테이블로 분류한다.

2.3 MongoDB

MongoDB는 문서 지향형 데이터베이스이다. 관계형 모델을 사용하지 않고 문서 모델을 채택하는 이유는 분산 확장을 쉽게 하기 위한 것이지만, 다른 이점도 가지고 있다. 내장문서와 배열을 허용함으로써 문서 지향 모델은 복잡한 계층 관계를 하나의 레코드로 표현할 수 있다. 또한 MongoDB에서는 문서의 키와 값을 미리 정의하지 않는다. 따라서 고정된 형태의 스키마가 존재하지 않는다. 고정된 스키마가 존재하지 않기 때문에 필요에 따라 필드를 추가하고 삭제하는 것이 쉬워졌고, 개발과정 또한 빠르게 이루어질 수 있다[9].

MongoDB는 분산 확장을 염두에 두고 설계된 데이터베이스로, 여러 장비에 데이터를 분산 확장을 지원한다. 문서를 자동적으로 재분배하고 사용자 요청을 올바른 장비에 라우팅 함으로써 클러스터 내 데이터양과 부하를 조절할 수 있다. 이러한 분산 확장은 기존의 관계형 데이터베이스의 성능 확장 보다 경제적이고 확장에 용이하다.

또한, MongoDB는 범용 데이터베이스 목적으로 만들어져 데이터의 생성, 읽기, 변경, 삭제 외에도 특별한 기능을 제공한다. 그러나 MongoDB에서는 관계형 데이터베이스에서 일반적으로 제공하는 기능 중 몇몇 기능이 없는데 그 중 대표적인 기능은 조인과 다중 트랜잭션이다. 이런 기능들은 분산 시스템에서 효율적으로 제공하기 어렵기 때문에 제외되었고, 높은 확장성을 제공하는 아키텍처를 위한 결정이다[9, 10].

III. Development of the Unified Database Design Methodology for Big Data Applications

본 논문에서는 대규모 OLTP 및 OLAP 구축을 위해 MongoDB 기반의 데이터베이스를 설계함에 있어 다음과 같은 설계 방법론을 제안한다. 개념적 데이터 모델로 시작하여 논리적 모델, 물리적 모델을 거쳐 MongoDB 데이터 웨어하우스 컬렉션을 정의하는 것으로 끝난다[11].

각 단계는 Fig.1과 같이 표현 할 수 있다.

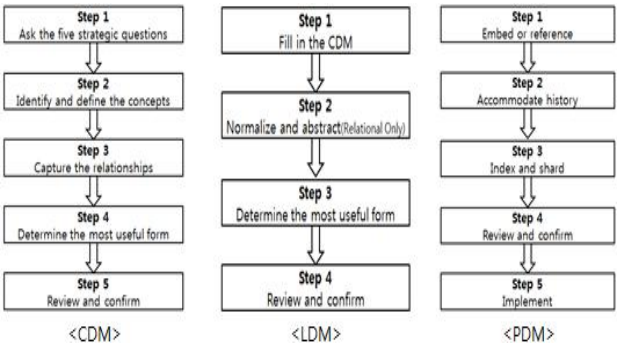

Fig. 1. Flowchart of Modeling

CDM(Conceptional Data Modeling) 단계에서는 본 논문의 핵심인 시스템의 목적을 파악하여 OLTP를 적용할 것인지, OLAP를 적용할 것인지 파악하는 역할을 추가로 수행하게 된다.

Table 1의 5가지 전략적인 질문들을 통해 시스템의 범위 및 목적을 파악한다.

Table 1. Five strategic questions

Q1. Waht is the application going to do?
Q2. “As is” or “to be”?
Q3. Is analytics a requirement?
Q4. Who is the audience?
Q5. Flexibility or simplicity?

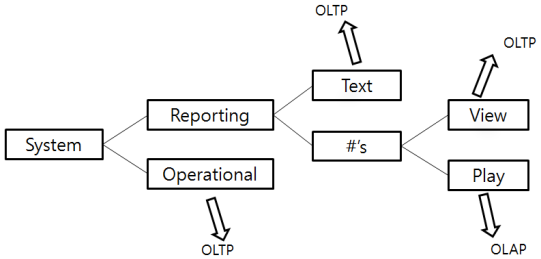

Fig. 2. Condition of Choosing a Data Model

다섯 가지 질문에 대한 답변을 얻은 후, Fig.2와 같이 목적에 따라서 분류를 수행한다. 목적에 따라 운영과 보고 두 가지로 분류 된다. 목적이 운영인 경우 데이터베이스를 구축한다. 그러나 목적이 보고인 경우, 단순 텍스트냐 또는 분석이 필요한 자료냐에 따라 분류를 한 번 더 수행한다. 단순 텍스트만 보고하는 경우 데이터베이스를 구축하며, 분석이 필요한 자료를 보고한다면 다시 한 번 분류를 수행한다. 시스템이 데이터를 단순히 보여주기만 한다면 OLTP를 위한 데이터베이스를 구축하고, 분석이 필요하다면 OLAP를 위한 데이터웨어하우스를 구축한다[11].

3.1 OLTP Database

본 논문에서 OLTP를 위해 사용한 데이터 셋은 기초학력 진단-보

정 시스템 설문조사 내용을 사용하였다. 해당 데이터셋을 CDM의 첫 번째 단계인 Table 1의 5가지 전략적인 질문들을 통해 시스템의 범위 및 목적을 파악한다.

다섯 가지 질문에 대한 답변으로는 첫 번째, 응용 프로그램은 모든 조직의 모든 유형의 설문조사 결과를 저장하고 결과를 분석할 수 있는 토대를 제공해야한다. 설문조사 저장 및 분석이 어려운 많은 조직에 유용할 것 이다. 두 번째, 구축해야 하는 응용프로그램은 새로운 시스템이다. 세 번째, 아직 이것은 데이터 입력 응용프로그램으로 분석 또는 보고가 필요한 단계가 아니다. 네 번째, 데이터 모델링 그룹이 사용자가 되어야 한다. 다섯 번째, 시간이 지남에 따라 설문조사 질문이 추가/삭제 될 가능성이 있고, 어떤 유형의 설문조사 질문에도 시스템에서 받아들일 수 있어야 하기 때문에 유용성 보다는 유연성이 중요하다. 따라서, 해당 시스템의 목적은 OLTP가 되어야 한다. OLTP 시스템에 맞는 데이터베이스를 구축하기 위해 CDM부터 PDM(Physical Data Modeling)까지의 과정을 진행한다.

다음 단계로 개념 템플릿을 정의한다. 개념 템플릿이란, 데이터 모델링에 필요한 각각의 개념들을 정리하는 것이다. 모든 개념을 확인하고 정의하는 단계에서 Table 2와 같은 개념 템플릿을 완성할 수 있다. Table 2에서 결정한 각각의 개념들에 대한 정의는 다음 Table 3과 같은 형식으로 정리한다.

Table 2. Concept Template for OLTP

Who	What	When	Where	Why	How
Org.	Survey				
Industry	Survey Category	Survey Completion Date			Completed Survey
Survey Respondent	Survey Section				
	Survey Question				

각각의 개념들에 대한 정의는 다음 Table 3과 같다.

Table 3. Definitions for each of these concepts for OLTP

Completed Survey	One filled in survey that contains a collection of opinions from a survey respondent in reaction to service.
Industry	The general sector in which an organization operates.
Organization	The company or government agency that needs the survey.
Survey	A questions designed to be completed by an single for improvement.
Survey Category	Category of Survey.
Survey Completion Date	The date that an individual filled in the survey.
Survey Question	The inquiry an organization uses to seek feedback.
Survey Respondent	The respondent who completes the survey.
Survey Section	A logical grouping within the survey.

세 번째 단계에서는 각 개념들 간의 관계를 파악하는 단계로서, 간단한 그림으로 각 개념들 간의 관계를 표현 할 수 있다.

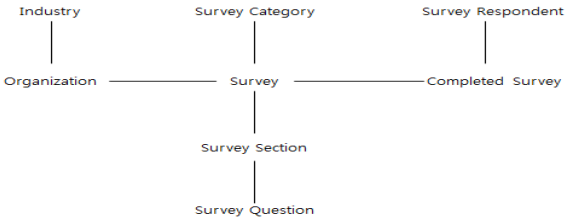

Fig. 3. Relationship of concepts

Fig.3의 간단한 그림을 가지고 다음 단계인 데이터 모델을 표현하기 위한 표기법을 결정한다. 이 경우, 정보공학 방법론에 의해 표기할 수 있고, Axis기법 이라고 하는 비즈니스 친화적 모델링 형식을 사용 할 수도 있다[11].

본 논문에서는 데이터베이스를 구축하기 위해 정보공학 방법론에 의한 표기법을 기본 표기법으로 채택한다. 따라서 다음 Fig.4와 같은 결과를 얻을 수 있다.

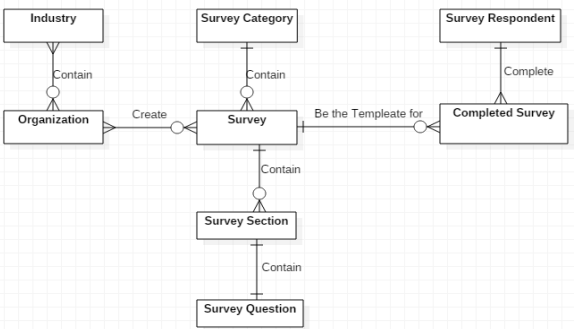

Fig. 4. Relationship Conceptual Data Model

네 번째 확인 및 검토 단계에서는 도출된 모델이 올바른지 확인하는 단계이다. 올바른 모델이라고 판단 될 때까지 위의 모델링 방법을 반복 적용하여 수정할 수 있고, 올바르다고 판단된다면 논리적 모델링 단계로 넘어간다.

논리적 데이터 모델의 첫 번째 단계는 개념의 속성을 식별하고 정의하는 것이다. CDM에서 정의한 개념은 LDM(Logical Data Modeling))의 엔티티가 되며, 이 단계를 마치게 되면 속성 템플릿과 속성 특성 템플릿을 얻을 수 있다. 각 엔티티의 속성 또는 데이터요소를 파악하여 속성 템플릿을 작성한다. 작성된 템플릿은 각각 Table 4, Table 5와 같다.

두 번째 단계로는 정규화 및 추상화 작업을 실행한다. 추상화는 모델내의 속성과 엔티티 그리고 관계를 재정의하고 결합하는 과정으로써, 설계에 유연성을 줄 수 있다[11].

각 컬렉션에 필요한 옵션을 결정해 Table 6의 컬렉션 기록 템플릿을 만든다[11].

세 번째 단계는 인덱싱(Indexing)과 샤딩(Sharding)이다. 인덱싱은 논리적 데이터 모델의 기본키와 대체키를 고유 인덱스(Unique Index)로 변환하는 것이고, 성능을 위한 인덱스를 추

Table 4. Properties Template for OLTP

	Industry	Organization	Survey	Survey Category	Survey Section	Survey Question	Survey Respondent	Completed Survey
Name	Id_name	Org_name		Sc_name	Ss_name	Sq_l_name	Sr_name	
Text					Ss_description	Sq_description, Poss_An_valeue, Poss_An_discription		Comp_S_Free_Text, Comp_S_Fiexed_answer
Date		Org_First_S_date						Comp_S_date
Code	SIC_code		S_code	Sc_code				
Number		Ogr_DUNS_num				Sq_num		
Identifier							Sr_id	
Indicator						Sq_singular_Respon_indicator		

Table 5. Properties Characteristics Template for OLTP

Property	Definition	Sample Value	Format	Length
Id_name	The common term used to describe the general sector an organization operations within. This is the standard description for the SIC code.	Computer programming services	Char	50
SIC_code	The Standard Industry Classification (SIC) is a system for classifying industries by a four-digit code. it is used by government agencies to classify industry areas.	62010 [Computer programming services]	Char	6
Org_name	The common term used to describe the company or government agency that needs the survey.	Google Naver	Char	50
Org_First_S_date	The date when the organization first started using survey.	sep-08-2015	Date	
Ogr_DUNS_num	Dun & Bradstreet(D&B) provides a DUNS Number, a unique nine digit identification number, for each organization.	123456789	Char	9
S_code	The unique and required short term referring to a survey.	A001-A	Char	6
Sc_name	The common term used to describe the driver for the survey.	Consumer Feedback	Char	50
Sc_code	The unique and required short term to describe the driver for the survey such as employee satisfaction.	AA BB	Char	2
Ss_name	The common term used to describe the logical grouping within the survey.	General	Char	50
Ss_description	The detailed text explaining the logical grouping within the survey. This is not displayed on the survey form.	Contains those questions pertaining to overall using experience.	Char	255
Sq_l_name	What the survey respondent sees on the form. That is, the question that appears.	What about the overall user interface?	Char	255
Sq_description	An explanation or background on the question, which is not displayed on the survey form.	This question lets the respondent rate user interface.	Char	255
Poss_An_valeue	Certain questions have a fixed response such as the Gender question for “Male” or “Female” or the “From 1 to 5. This field stores all of the possible fixed responses.	Male Female	Char	50
Poss_An_discription	This field stores the meaning for each of the Possible Answer Values.	1 means poor 3 means Average	Char	100
Sq_num	A required number assigned to each question. This number is unique within a survey.	1 2	Integer	3
Sq_singular_Respon_indicator	Some questions allow for more than one response.	Y N	Boolean	
Sr_name	The name the survey respondent writes on the completed survey.	JunHo Lee KyungSoo Joo	Char	100
Sr_id	A unique and required value for each survey respondent.	000001	Integer	6
Comp_S_Free_Text	Captures the responses to those questions that do not require a fixed response.	management of students	Char	255
Comp_S_Fiexed_answer	Captures the responses to those questions that require a fixed response.	Male Female	Char	100
Comp_S_date	The date the survey respondent completed the survey.	Sep-09-2017	Date	

가 할 수 있다. 샤딩은 컬렉션이 두 개 이상의 부분으로 분리되 는 경우를 말하며 수평, 수직 두 가지의 분할 방식이 있다[11].

Table 6. Collection History Template for the Case for OLTP

	Type 0	Type 1	Type 2	Type 3
Organization		✔		
SurveyCreation		✔		
Survey			✔	
Completed Survey	✔			
Survey Question		✔		

네 번째 단계는 확인 및 검토 단계로 해당 모델이 올바르게 작성된 모델인지 검토한다. 이러한 단계를 통해 최종 완성된 데이터 모델은 다음 Fig.5와 같다.

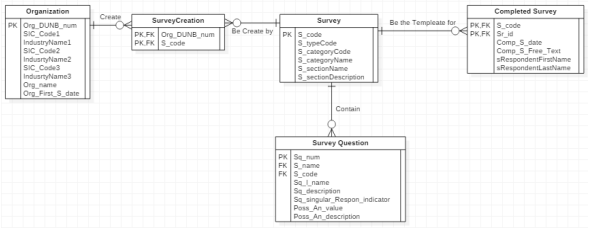

Fig 5. Final Data Model

다섯 번째 단계는 설계된 Fig.5의 데이터 모델을 토대로 MongoDB 컬렉션을 정의한다[11].

3.2 OLAP Data warehouse

본 논문에서 OLTP를 위해 사용한 데이터 셋은 판매정보 데이터를 사용하였다. 해당 데이터셋을 CDM의 첫 번째 단계인 Table 1의 5가지 전략적인 질문들을 통해 시스템의 범위 및 목적을 파악한다.

다섯 가지 질문에 대한 핵심 답변으로 본 시스템은 판매 정보에 대한 분석이 필요한 시스템이다. 따라서, 해당 시스템의 목적은 OLAP가 되어야 한다. OLAP 시스템에 맞는 데이터 웨어하우스를 구축하기 위해 CDM부터 PDM까지의 과정을 진행한다.

개념적 모델링의 두 번째 단계는 개념을 식별하는 단계로서, Dimensional면에서 대답해야하는 구체적인 질문들을 결정한다. 이 질문들은 실제로 구축할 시스템이 분석해야하는 것에 대한 질문들로 구성된다.

구체적인 질문들은 다음과 같다.
① 제품별 기간별 매출현황
② 제품별 판매지역별 매출현황
③ 프로모션별 제품별 매출현황

세 번째 단계에서는 각 개념들 간의 관계를 파악하는 단계로서, 위의 구체적인 질문을 토대로 Grain 행렬을 만들게 된다. Grain 행렬은 질문의 측정값이 열이 되고 질문의 번호가 행이 되는 스프레드시트이다[11].

구체적인 질문들은 매출 현황이라는 것에 중점을 둔다. 따라서 Table 7과 같은 Grain 행렬을 만들 수 있다.

Table 7. Grain Matrix

	Sales Status
Product	1, 2, 3
Term	1
sell-area	2
Promotion	3

네 번째 단계는 파악한 개념들의 관계를 토대로 해당 데이터 모델을 표현하기 위한 표기법을 결정해야 한다. 정보공학 방법론에 의해 표기할 수 있고, Axis기법 이라고 하는 비즈니스 친화적 모델링 형식을 사용 할 수도 있다[11].

본 논문에서는 데이터 웨어하우스를 구축하기 위해 기본 표기법으로 스타 스키마[12]를 채택한다. 개념 스타 스키마는 아래 Fig.6과 같다.

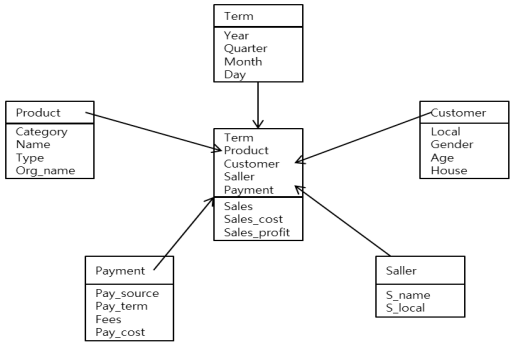

Fig. 6. Conceptual Star Schema

논리적 모델링의 첫 번째 단계는 개념의 속성을 식별하고 정의하는 것이다. CDM에서 정의한 개념은 LDM의 엔티티가 되며, 이 단계를 마치게 되면 속성 템플릿과 속성 특성 템플릿을 작성할 수 있다. 각 엔티티의 속성 또는 데이터요소를 파악하여 속성 템플릿을 작성한다.

CDM에서 얻은 데이터 모델을 기반으로 Table 8과 같은 속성 템플릿을 작성한 후, Table 9와 같은 속성 특성 템플릿을 작성한다.

아래의 Table 8, 9를 작성 한 후 스타 스키마로 변환시키게 되면 다음 Fig.7과 같은 스타 스키마를 얻을 수 있다. 두 번째 단계는 관계형 데이터베이스를 사용할 경우 진행하는 정규화 단계이기 때문에 본 논문에서는 진행하지 않는다.

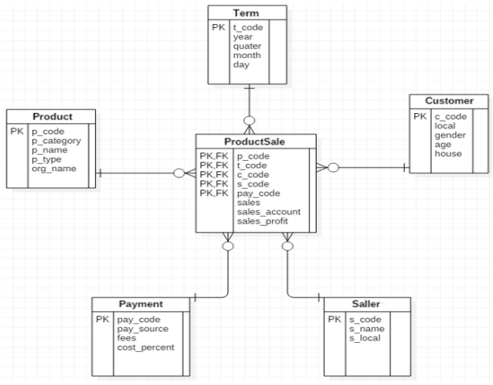

Fig. 7. Logical Star Schema

Table 8. Properties Template for OLAP

	Product	Term	Customer	Payment	Seller
Name	p_name, org_name		u_name		s_name
Text	p_type, p_category		local, gender, house	pay_source	s_local
Code	p_code	t_code	c_code	pay_code	s_code
Date		month, day, year			
Quater		quater			
Number			age	fees, cost_percent	

Table 9. Properties Characteristics Template for OLAP

Property	Definition	Sample Value	Format	Length
p_name	production name.	PC	char	100
org_name	supply company name.	SCH univ.	char	100
u_name	customer name.	Junho Lee	char	30
s_name	seller name.	Kildong Hong	char	30
p_type	product specification.	AP_0001	char	50
p_category	product category.	0001A	char	50
local	customer's purchase area.	Asan	char	100
gender	customer's gender.	M or F	char	1
house	type of the customer's house.	APT	char	100
pay_source	payment method.	Card or Cash	char	20
s_local	sales area of the product.	Asan	char	100
p_code	product code.	Product0001	char	50
c_code	customer code.	C0001	char	50
t_code	term code.	A2017M4	char	50
pay_code	billing code.	P_0001	char	50
s_code	seller code	A0001	char	50
month	sales period (month).	10	integer	2
day	sales period (day).	5	integer	2
year	fiscal year.	2017	integer	4
quater	sales period (quarter).	1	integer	2
age	customer's age.	45	integer	10
fees	commission rate.	3.5	float	20
cost_percent	payment rate.	90	float	20

세 번째, 네 번째 단계는 CDM 단계와 동일하게 유용한 표기법을 결정하고 데이터 모델을 확인하는 단계이다.

다음은 물리적 모델링의 첫 번째 단계로 LDM 단계에서 나온 논리적 데이터 모델을 물리 스타 스키마로 변경한다. 변경된 물리 스타 스키마는 Fig. 8과 같다.

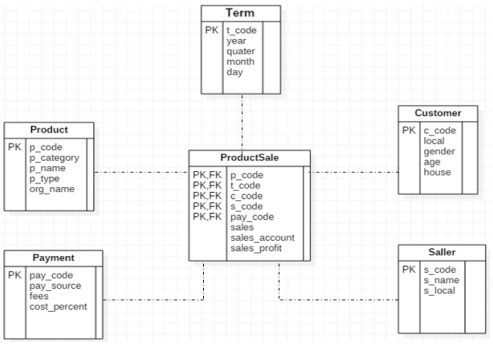

Fig. 8. Physical Star Schema

두 번째 단계는 Accommodate History로 필드 값이 시간에 따라 어떻게 변하는지에 대한 옵션을 설정한다. 필드 값의 변화를 다루기 위한 옵션은 SCD(Slowly Changing Dimension)로 불리며, 숫자(0, 1, 2, 3)로 구성된다[11].

① SCD 0 : 원 상태를 저장하고 변경사항은 저장하지 않는다.

② SCD 1 : 가장 최근의 상태를 저장한다.

③ SCD 2 : 데이터의 변경사항이 있을 때마다, 모든 변경사항을 컬렉션에 저장한다.

④ SCD 3 : 일부 기록에 대해 요구사항(최신보기, 이전보기 등...)이 있다.

각 컬렉션에 필요한 옵션을 결정해 컬렉션 기록 템플릿을 만든다[11].

해당 물리 스타 스키마에 적합한 컬렉션 기록 템플릿을 정의하면 Table 10과 같이 정의 할 수 있다.

Table 10. Collection History Template for OLAP

	Type 0	Type 1	Type 2	Type 3
Term	◎			
Product		◎		
Customer		◎		
Payment			◎	
Seller		◎		
Sales			◎	

세 번째 단계는 인덱싱(Indexing)과 샤딩(Sharding)이다. 인덱싱은 논리적 데이터 모델의 기본키와 대체키를 고유 인덱스(Unique Index)로 변환하는 것이다. 샤딩은 컬렉션이 두 개 이상의 부분으로 분리되는 경우를 말하며 수평, 수직 두 가지의 분할 방식이 있다[11].

네 번째 단계는 확인 및 검토 단계로 해당 모델이 올바르게 작성된 모델인지 검토한다.

다섯 번째 단계는 설계된 데이터 모델을 토대로 MongoDB 데이터 웨어하우스 컬렉션을 정의한다.

3.3 MongoDB Collection

3.1, 3.2절의 Fig.5, 8의 PDM을 통해 정의된 MongoDB 컬렉션은 다음 Table 11, 12, 13, 14, 15, 16, 17, 18, 19, 20과 같다.

Table 11. Organization Collection

```
Organization:
{
    Org_DUNS_num : “123456789”,
    Industry : [
    {
        SIC_code : “000013”,
        id_name : “Elementary School” ,
        SIC_code : “000014”,
        id_name : “Middle School”
    ] },
    Org_name : “Seoul Middle School”,
    Org_First_S_date : ISODate(“2017-08-01”)
}
```

Table 12. Survey Collection

```
Survey:
{
    S_code: “A001-A”,
    S_typeCode : “AA”,
    S_categoryCode : “CF”,
    S_categoryName : “Consumer Feedback”,
    S_sectionName : “Program Experience”,
    S_sectionDescription : “Contains those questions pertaining to overall using experience.”
}
```

Table 13. SurveyCreation Collection

```
SurveyCreation:
{
    Org_DUNS_num : “123456789”,
    S_Code : “A001-A”,
}
```

Table 14. Survey Question Collection

```
Survey Question:
{
    Sq_num : “1”,
    S_code : “A001-A”,
    Sq_l_name : “What about the overall user interface?”,
    Sq_Description : “This question lets the respondent rate user interface.”,
    Sq_Singular_Respons_Indicator : “Y”
}
```

Table 15. Completed Survey Collection

```
Completed Survey:
{
    S_code : “A001-A”,
    Sr_id : “123456”,
    Comp_S_date : ISODate(“2017-09-05”),
    sRespondentFirstName : “JunHo”,
    sRespondentLastName : “Lee”
}
```

Table 16. Product Collection

```
Product:
{
    p_code : “Product0001”,
    p_category : “00001A”,
    p_name : “PC”,
    p_type : “APC_0001”,
    org_name : “Soonchunhyung Univercity”
}
```

Table 17. Period Collection

```
Period:
{
    t_code : “A2017M4”,
    year : “2017”,
    month : “10”,
    day : “30”,
    quater : “1”
}
```

Table 18. Customer Collection

```
Customer:
{
    c_code : “C0001”,
    local : “Asan”,
    gender : “M”,
    age : “35”,
    home : “APT”
}
```

Table 19. Seller Collection

```
Seller:
{
    s_code : “S0001”,
    s_name : “Kildong Hong”,
    s_local : “Asan”
}
```

Table 20. Payment Collection

```
Payment:
{
    pay_code : “P_0001”,
    pay_source : “Card”,
    fees : “3.5”,
    cost_percent : “90”
}
```

V. Conclusions

본 논문에서는 빅 데이터 응용을 위한 MongoDB 기반의 대규모 OLTP 및 OLAP 구축을 위한, 정보 요구사항 분석부터 OLTP 및 OLAP 구축까지 일관된 설계 방법론을 개발하여 제안하였다. 본 방법론에 따르면 개념, 논리, 물리 각 단계별 질문을 통해 시스템의 목적 및 필요한 개념과 관계를 질문을 통하여 식별한다. 목적 분류를 수행하여 OLAP 시스템인지 OLTP 시스템인지 판별하여 OLTP 라면, 개념 템플릿을 통해 각 개념들을 정의하고 식별하여 스키마를 도출한다. 또한 OLAP 라면, 구체적인 질문을 통해 Grain 행렬을 찾아내고 찾은 Grain 행렬을 통해 모델을 만들어 최종 단계까지 스키마를 도출할 수 있다. 각 단계별 검토 과정을 통해 수정이 간략하게 이루어 질 수 있다.

본 논문에서는 MongoDB를 기반으로 한 대규모 OLTP 및 OLAP를 위한 일관적인 설계 방법론을 제안함으로써, 기존의 관계형 데이터베이스를 기반으로 한 OLTP 및 OLAP 시스템의 한계인 대용량 데이터 저장 및 처리의 어려움을 극복하는, 빅 데이터를 위한 통합 데이터베이스 설계 방법론을 제안하였다.

REFERENCES

[1] Jacobs, A., “The pathologies of big data,” Communications of the ACM, 52(8), pp. 36-44, 2009.

[2] Stonebraker, M., “New Opportunities for New SQL,” Communications of the ACM, Vol.55, issue11 pp. 10-11, 2012.

[3] Cuzzocrea, Alfredo, Ladjel Bellatreche, and Il-Yeol Song., "Data warehousing and OLAP over big data: current challenges and future research directions," Proceedings of the sixteenth international workshop on Data warehousing and OLAP. ACM, 2013.

[4] Dehdouh, Khaled, Omar Boussaid, and Fadila Bentayeb., "Columnar nosql star schema benchmark," International Conference on Model and Data Engineering. Springer, Cham, 2014.

[5] Chevalier M, El Malki M, Kopliku A, Teste O, Tournier R., "Implementing Multidimensional Data Warehouses into NoSQL," 17th International Conference on Enterprise Information Systems, Vol.1, pp. 172-183, 2015.

[6] W.H.Inmon., "Building the Data Warehouse." John Wiley & Sons, 2002.

[7] Conn, Samuel S. "OLTP and OLAP data integration: a review of feasible implementation methods and architectures for real time data analysis." SoutheastCon, 2005. IEEE, pp. 515~520, 2005.

[8] Chaudhuri, Surajit, and Umeshwar Dayal. "An overview of data warehousing and OLAP technology." ACM Sigmod record 26.1, pp. 65-74, 1997.

[9] Michael Dirolf, "MongoDB The Definitive Guide." O'ReillyMedia, 2013.

[10] Mamenko, J. "Introduction to data modeling and msaccess," Lecture Notes on Information Resources, 2004.

[11] Hoberman, S., “DataModeling for MongoDB,” Technics Publications, 2014.

[12] Christopher Adamson, “Star Schema The Complete Reference,” McGraw-Hill Osborne, 2010.

Authors

Junho Lee received the B.S. degrees in Computer Software Engineering from Soonchunhyang University, Korea, in 2015 respectivelySoftware Engineering at Soonchunhyang University, Asan, Korea, in 2015. He is currently a M. S. course in the Department of Computer Science, Soonchunhyang University. He is interested in database and big data database.

Kyungsoo Joo received the Ph.D. degrees in Computer Science from Korea University, Korea, in 1993 respectively Korea University, Seoul, Korea, in 1993. He is currently a Professor in the Department of Computer Software Engineering, Soonchunhyang University. He is interested in database and big data database.