

---

# NoSQL 기반의 SNS 데이터베이스 설계

장성호\* · 김수희\*

\*호서대학교

NoSQL-based SNS Data Model Design

Seongho Jang\* · Suhee Kim\*

\*Hoseo University

j2000moom@nate.com, shkim@hoseo.edu

## 요 약

SNS(Social Networking Service)는 사용자 간의 자유로운 의사소통과 정보 공유, 그리고 인맥 확대 등을 통해 사회적 관계를 생성하고 강화시켜주는 온라인 플랫폼을 의미한다. 이 연구에서는 SNS에서 주요 개체들을 발견하고 그들간의 관계를 도출하고, 이들을 기반으로 ERD를 그린다. 작성한 ERD를 NoSQL 데이터베이스인 MongoDB 데이터 모델의 컬렉션들로 변환함으로써, SNS 데이터베이스의 주요 스키마를 설계한다.

## ABSTRACT

A SNS(Social Networking Service) is an online platform to build social networks or social relations among people who, for example, share free communication, information, and make more personal connections. In this paper, we find representative entities, develop relationships among them, and draw an ERD based on the entities and their relationships. And then we design a SNS database schema by converting the ERD into collections according to data model of MongoDB, which is an NoSQL database.

## 키워드

SNS, NoSQL, MongoDB, DB Modeling

## I. 서 론

현재 스마트폰의 보급과 SNS(Social Networking Service)의 급격한 확산, 비정형 데이터의 폭증으로 인하여 빅데이터에 대한 관심이 증가하고 있다. 이것은 기존의 RDBMS 중심의 데이터 저장 기술에서 새로운 데이터 저장 기술인 NoSQL이 등장하게 된 계기이다. NoSQL은 Not Only SQL의 약자로 기존 RDBMS 형태의 데이터베이스가 아닌 다른 형태의 데이터 저장 기술을 의미한다. RDBMS는 획일화된 구조의 데이터 저장 방식(table)을 취하지만, NoSQL은 이와는 매우 다른 형태의 데이터 저장 구조를 가진다. NoSQL의 등장으로 소셜 서비스 분야에서 RDBMS에서 NoSQL

로의 전향이 이루어지고 있다. 소셜 네트워크를 통하여 생성되는 비정형/반정형/정형 형태의 데이터를 더 효율적으로 저장하고 사용하기 위해서는 대상 데이터들을 세밀히 분석하고 그들의 관계를 파악하여 이를 통한 데이터 모델을 설계해 보는 것이 필요하다. 본 논문에서는 open-source NoSQL 중 상당히 대중적으로 많이 사용되고 있는 MongoDB를 사용하여 SNS 데이터베이스를 설계한다.

## II. NoSQL

NoSQL은 대부분 non-relational, distributed, open-source, horizontally scalable 등의 특징을 갖는 차세대 데이터베이스이다[1]. NoSQL 데이터베이스는 전통적인 관계형 데이터베이스보다는 일관성을 위한 제약이 더 적은 모델로 데이터의 저장과 검색 메커니즘을 제공한다. NoSQL 데이터베이스들은 최고의 성능향상을 목적으로 단순한 검색과 데이터 추가 연산들을 위해 설계된 매우 최적화된 stores이다[2].

### 1. NoSQL 데이터 모델

NoSQL은 column, document, key-value, graph 등의 다양한 데이터 모델들을 가지고 있다. 현재 NoSQL 데이터베이스는 150종 이상 발표되었으며, <표 1>은 각 데이터모델로 분류되는 몇 개의 데이터베이스를 예시한다.

표 1 데이터 모델 기반 데이터베이스 분류

데이터 모델	데이터베이스
Column	Hbase, Apache Cassandra, Google Big Table
Document	MongoDB, Couchbase
Key-value	DynamoDB, Riak
Graph	Allegro, Virtuoso

### 2. MongoDB

이 절에서는 문서 지향 데이터베이스로 NoSQL 사용자들로부터 상당히 인기가 있는 MongoDB의 특징을 살펴본다.

MongoDB(“humongous”)는 크로스 플랫폼 문서지향 데이터베이스이다. 데이터의 기본 단위는 문서(document)로 키와 대응되는 값의 집합으로 이루어진다. 이러한 문서들의 모임을 collection이라 한다. Document는 관계데이터베이스의 row(행)과 대응되며, collection은 table과 대응된다. 그리고 document 내에 또 다른 document가 subdocument 형태로 포함될 수 있으며, 하나의 키 값으로 배열 데이터가 허용된다.

MongoDB는 고성능에 주안점을 두어 모든 것을 설계하였다. MongoDB의 특징을 다음과 같이 요약할 수 있다[3].

- 문서 지향 데이터베이스 : JSON 스타일 문서
- Full 인덱스 지원 : 모든 속성 대상으로 다양한 형태의 인덱싱 가능
- Replication, High Availability
- 자동 샤딩 : 자동 분산 확장 (scale-out)
- 질의 : 문서 기반 다양한 질의 지원
- 빠른 In-Place Updates

- Map/Reduce : 다양한 집계기능 및 데이터 분산 처리
- GridFS : 대용량 파일과 메타데이터의 편리한 저장 기능

## III. SNS 데이터베이스 설계

SNS는 사회적 관계 개념을 인터넷 공간으로 가져온 것으로 사람과 사람, 콘텐츠와 콘텐츠간의 관계 맺기를 통해 네트워크 형성을 지원하는 서비스이다[4].

이 장에서는 SNS 스키마를 개념적으로 파악하기 위해 ERD(entity-relationship diagram)를 그려보고, MongoDB 데이터 모델을 이용한 데이터베이스를 설계한다.

### 1. SNS ERD

이 절에서는 SNS에서 가장 중심이 되는 개체들과 그들의 관계를 파악하며 ERD를 그려본다.

#### 1.1 개체타입

SNS에서 가장 대표가 되는 개체타입은 사용자들(users), 이들이 작성하는 소식들(News)이라 할 수 있다. 그리고 각 사용자가 News에 댓글들(Comments)을 명시할 수 있다.

- Users
- News
- Comments

#### 1.2 관계 타입

사용자들 간의 대표적인 관계 타입 (Friend)와 사용자가 뉴스를 포스팅하는 관계(Posts-News)가 있다. 각 뉴스에 사용자들이 “추천” 멘트를 명시할 수 있는데, 이를 User-Likes 관계 타입이라 하자. 각 뉴스에는 사용자들이 많은 댓글(Comments)들을 작성할 수 있는데, Users, News 간의 2진 관계와 Users, Comments간의 2진 관계, News, Comments간의 2진 관계가 있다.

- Friend : Users간의 다대다의 관계
- Posts-News : Users와 News간에 1대다 관계
- User-Likes : Users와 News간에 다대다 관계
- Posts-Comments : Users와 Comments간의 관계,  
: Comments는 전체 참여,  
키계약조건
- Responses\_Of : News와 Comments간의 관계,  
Comments는 약개체, 전체참여,  
키계약조건

### 1.3 ERD

지금까지 서술한 개체들과 관계들을 ERD로 표현하면 <그림 1>와 같다.

SNS 사용자는 다른 사용자들과 친구가 되어 각종 정보를 공유할 수 있으며, 친구의 친구 정보도 조회할 수 있다. 사용자는 친구들이 생성한 여러 가지 소식들을 모두 조회할 수 있다. 또한, 각 SNS 사용자는 자신의 News(글, 문서)를 포스팅할 수 있으며, 친구들은 이들을 읽을 수 있고 댓글을 달 수 있다.

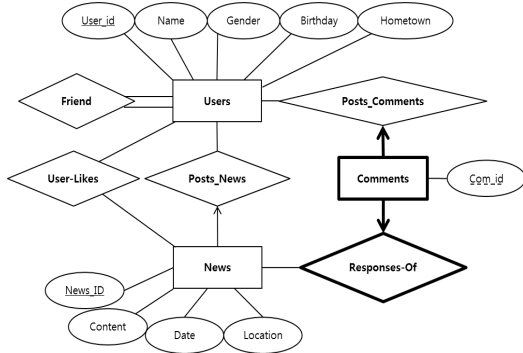


그림 1 SNS 주요 개체와 관계 ERD

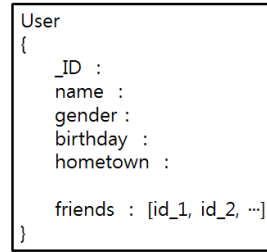


그림 2 Users Collection

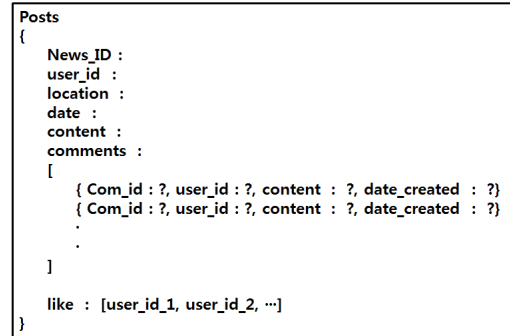


그림 3 Posts Collection

## 2. MongoDB 데이터모델을 이용한 SNS 데이터베이스 설계

이 절에서는 <그림 1>에서 제시한 ERD를 토대로 하여, MongoDB 데이터 모델에 기반을 둔 SNS 데이터베이스를 설계한다.

MongoDB 스키마 디자인에서 객체가 자신만의 컬렉션을 가질 것인가, 다른 컬렉션 안에 내장되어야 하는가를 결정하는 것이 중요하다. 일반적인 RDBMS에서는 정규화를 통해서 테이블을 분리하지만, MongoDB에서는 객체를 내장할 경우 디스크에서 같은 곳에 위치하기 때문에 더 효율적이기 때문이다[5].

앞서 제시한 <그림 2>의 ERD에서 Users 개체 타입과 News는 최상위 컬렉션에 속한다. Users 컬렉션에 friends 관계를 배열 형태로 추가하였다. 그리고 Posts 컬렉션에 News 컬렉션 전체와 누가 포스팅한 것인지를 식별할 수 있는 Users ID를 추가하였다. 또한, comments와 각 comment를 포스팅한 사용자를 식별하기 위해 subdocument들로 이루어진 배열로 설계하였다. <그림 3>과 <그림 4>는 이들을 표현한 스키마이다. 앞서 언급하였듯이 MongoDB는 컬렉션간의 조인 연산을 지원하지 않는데, 이를 감안하여 ERD상의 여러 개의 개체 타입과 관계타입들을 2개의 컬렉션으로 모두 표현하였다. 이렇게 함으로써 데이터의 저장과 검색에 대한 성능을 최적화할 수 있다.

## IV. 결 론

본 논문에서는 SNS 데이터베이스를 설계하기 위해 주요 객체들을 도출하고 이들의 관계들을 분석하여 ERD로 표현해 보았다. 이를 토대로 하여 MongoDB를 이용하여 SNS 데이터베이스의 대표적인 개체 및 관계들을 설계하였다. MongoDB에서는 최대의 성능을 실현하기 위해 조인 연산을 지원하지 않지만, 키의 값이 subdocument 형태가 가능하고 배열을 허용하므로 이러한 특징을 이용해 컬렉션의 스키마를 도출할 수 있었다. 본 논문에서 설계한 내용을 토대로 SNS 전체에 대한 데이터베이스 설계를 설계하고, 구현할 수 있을 것이라 기대한다.

### 참고문헌

- [1] <http://nosql-database.org>, October 3. 2013
- [2] Mimul, SNS 개요, MIMUL'S DEVELOPER WORLD, October 3. 2013, June 30. 2009
- [3] <http://en.wikipedia.org/wiki/NoSQL>, October 3. 2013
- [4] <http://www.mongodb.org/>, October 3. 2013
- [5] OUTSIDER, MongoDB Schema 디자인 하기, October 3. 2013, June 13. 2011