Using Static Db for Social Networking

Let's take an Example

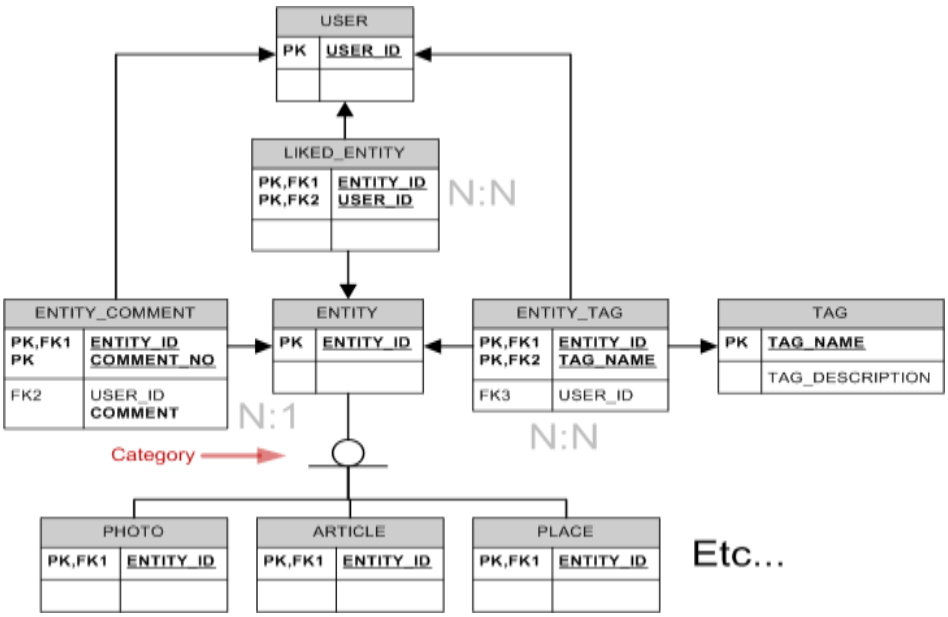
The picture below is a snapshot from a comment section in Facebook.

It is observed in the above picture that a hierarchy of comments and replies are accumulated on all posts.

Now if we look into the Facebook site we are left to wondered that how these comments are stored inside a database so that efficient retrieval can be performed.

At first, we may think about a RDBMS i.e. is mapping all these comments and replies to a table structed datastore where all comments and replies are stored with proper user id and no of likes.

- **Let's see how Using RDBMS we can Store Facebook Comments.**



Now that's a lot of tables thus this very inefficient as most of the time will be spend of joining the tables.

- Let's see how Static Db does it

```
CREATE sportcenter_post1 = {
    likes: 5851,
    comments:
    [
        {
            user_name : "Steve",
            comment: "Steve comment to post",
            likes : 5,
            replies :
            [
                {
                    user_name : "Harry",
                    comment : "Harry replies to Steve",
                    likes : 2,
                    replies : []
                }
            ]
        },
        {
            user_name : "Sport Center",
            comment : "Sport Center comment to post",
            likes : 5,
            replies :
            [
                {
                    user_name : "Steve",
                    comment : " Steve replies to Sport Center",
                    likes : 2,
                    replies : []
                }
            ]
        }
    ]
}
```

The entire post is stored Way more easily in Static Db than the RDBMS implementation.
The advantages are: -

1. The entire data is stored in one collection (RDBMS uses many table and JOINs)
2. No overhead of table joining.
3. Efficiently queries data that is only needed.
4. Reduced redundancy as foreign key and primary key are not used.

Even static Db support querying making it easily retrievable.

```
>>> FIND { "user_name" } IN sportcenter_post1.comments WHERE { likes : { $gt : 1 } }
RETURN MSG:
[
    {
        user_name : "Steve"
    } ,
    {
        user_name : "Sport Center"
    }
]

>>> FIND { "comment" } IN sportcenter_post1.comments WHERE { user_name: "Sport Center" }
RETURN MSG:
[
  {
    comment : 'Sport Center comment to post'
  }
]

>>> FIND {"comment","user_name" }
    IN (FIND {"replies"} IN sportcenter_post1.comments WHERE{user_name: "Sport Center"})
    WHERE {user_id: 0}

RETURN MSG:
[
  {
    user_name: "Steve"
    comment: 'Steve replies to Sport Center'
  }
]
```