



Andrew Risse HW1 Report

My design is an Extended Entity Relationship Model (EERM) that models a Forum. This report will explain the relationships and design decisions made that are depicted on the EERM. It will not describe each and every relationship, but instead discuss the important ones that significantly affected the overall design and its strengths/weaknesses.

The EERM has three specialization hierarchies that are used to help define the relationships between USERS/ADMINISTRATOR, THREAD/ANNOUNCEMENT, and LIKE/SUPERLIKE.

- To take advantage of the properties of inheritance, ADMINISTRATORS are subtypes of USERS, because they have the same attributes of normal users, plus one additional date attribute. They can also post normal threads, but needed to be a separate entity so that they could post ANNOUNCEMENTS as well. I gave both surrogate keys as primary keys to uniquely identify each USER or ADMINISTRATOR. The relationship has partial completeness because not all USERS have to be ADMINISTRATORS; the subtype is optional. The Disjoint/Overlapping constraint (d/o) is omitted because there is only one subtype. A subtype discriminator called USR_IS_ADMIN indicates if a specific instance is a member of the subtype or supertype.
- A THREAD can be one of two subtypes- NORMALTHREAD or ANNOUNCEMENT so they have total completeness and a disjoint relationship. A subtype discriminator called THREAD_TYPE indicates which subtype an instance is a member of. All threads have a surrogate key TID that uniquely identifies them and serves as a primary key. ANNOUNCEMENT had to be a different subtype because of the “one to zero or one” relationship to VIDEO as well as only having a “post” relationship with ADMINISTRATOR and not USER. ANNOUNCEMENT and NORMALTHREAD needed to share the THREAD relationships with likes and tags, so that is why they share a supertype.
- LIKE and SUPERLIKE are subtypes of MARKPOST and have a disjoint and total completeness relationship. This design has a weakness in that MARKPOST will likely have many NULL values because if the MARKPOST is associated with a thread, the RID (reply ID) will be NULL, and vice versa. Because a REPLY is not a type of THREAD, REPLY is not part of the THREAD supertype and is a separate entity type. This also allows REPLY to maintain a different (“one to zero or one”) relationship to PICTURES. A subtype discriminator called LIKE_TYPE indicates which subtype the instance is a member of.

There is one bridge table in the design that serves to represent the many to many relationship between threads and tags. A THREAD can have many TAGs, and a TAG can belong to many THREADs. The table POSTTAG is the bridge table that combines primary keys from THREAD and TAG. These tables incorporate strong relationships because TID and TAG_ID are primary keys while also being components of the primary key for POSTTAG.

The overall design has several strengths and weaknesses, a few of which have been mentioned, but will be included again.

- Strengths
 - There are no multi-valued attributes, which reduces complexity.

- “Tag” and “like” relationships can be shared and used amongst different entities which reduces the need for redundant tables (ex. A REPLY_LIKE table and THREAD_LIKE table)
 - Only two strong relationships are in the design, most relationships are weak
 - Inheritance allows specialization hierarchies to simplify the diagram by allowing entities to share attributes and relationships
- Weaknesses
 - Excessive NULL values in MARKPOST, REPLY, and PICTURES tables due to TIDs and RIDs
 - Greater memory requirement to store many surrogate keys

Although some tables will still have many NULL values, utilizing specialization hierarchies enabled this design to mitigate that weakness while visually and conceptually simplifying the distribution of the many relationships in the EERM.