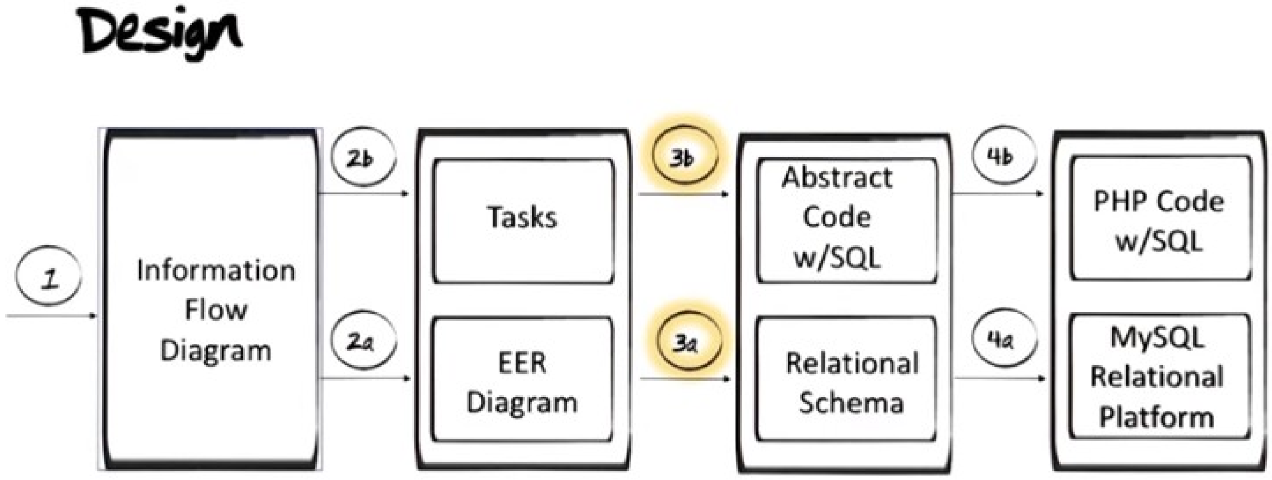
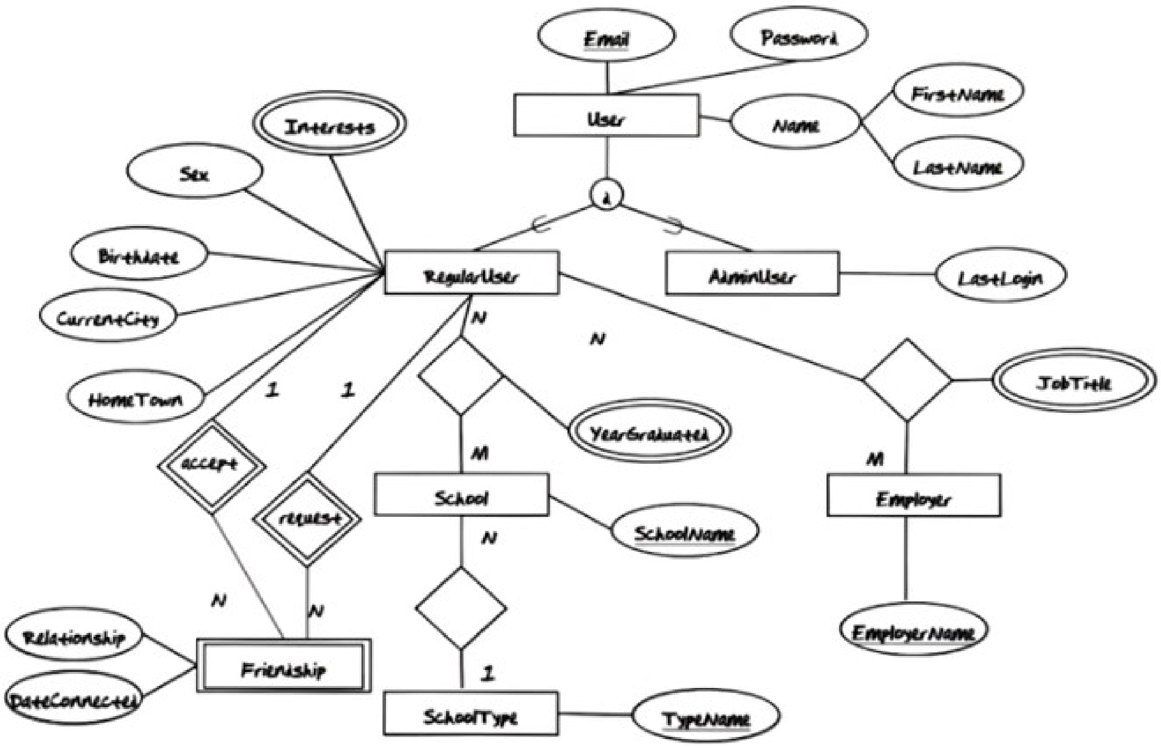
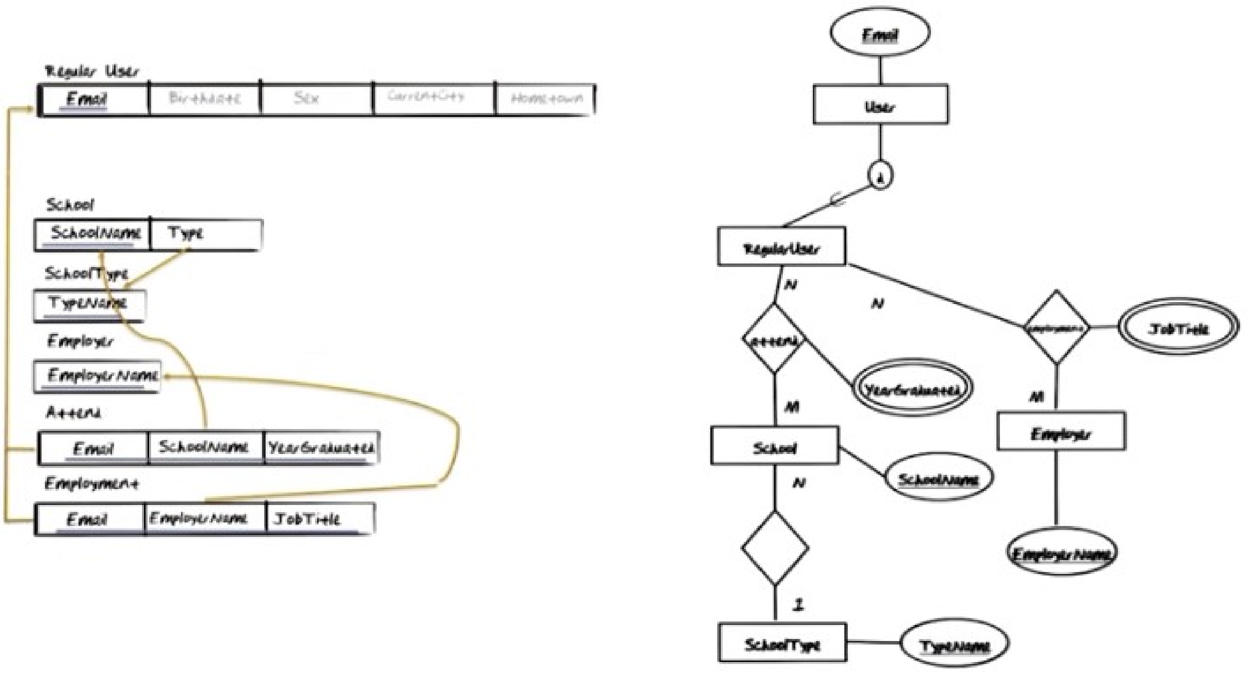


別忘了上圖中的四個分別為:

1 Analysis
2 Specification
3 Design
4 Implementation

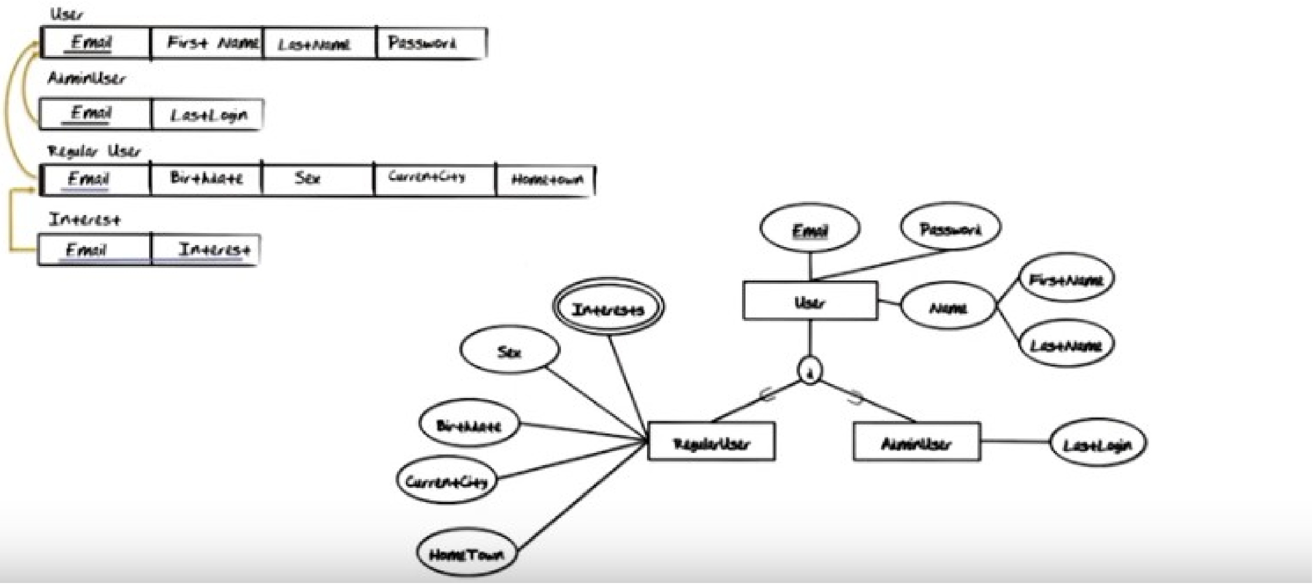


The above figure: Here's the extended entity-relationship diagram we ended up with. …. So our job now is to take this extended entity-relationship diagram and translate it into a relational database. We will do that a little step at a time. 注意上圖的之前出現時, accept 和 Friendship 之間是粗線, request 和 Friendship 之間也是粗線, 這裡是劃錯了, 日他大爺.



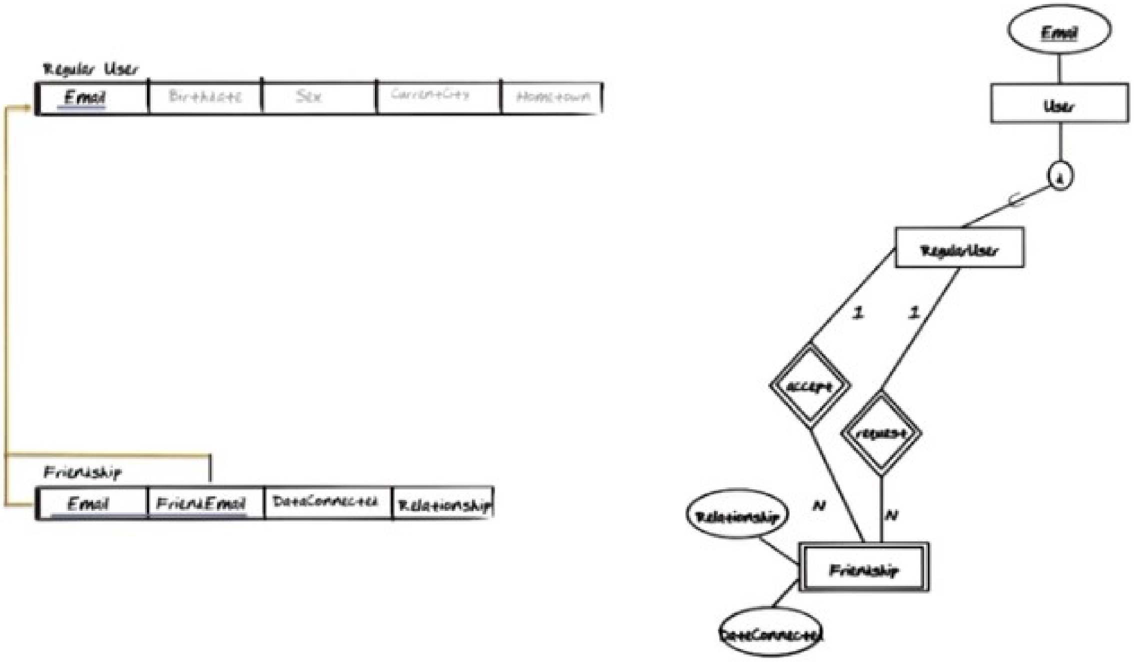
The above figure: School 這個表中的 Type: the Type you see here is basically a foreign key pointing to the primary key of SchoolType.

The Attend relationship type is a many-to-many relationship type as you can see. In order to map a many-to-many relationship types we need the identifying property type of each one of the participating entity types. So we need Email to identify RegularUser and we need SchoolName to identify the School. What is a little bit different about this mapping is that for each one of these instances, there is a YearGraudated associated with it. One reason for that could be that you might attend, for example, Georgia Tech twice, once for your bachelors and once for your masters. In each one of those, there will of course be a unique YearGraduated (這就是 EER 圖中 relationship 帶 property(應該都是 multi-valued property)時的意思, 己 copy 到我的 note 3. Extended-Entity Relationship Model 中). 注意 Attend 的 identifier 是 Email-SchoolName-YearGraduated 一起組成的.



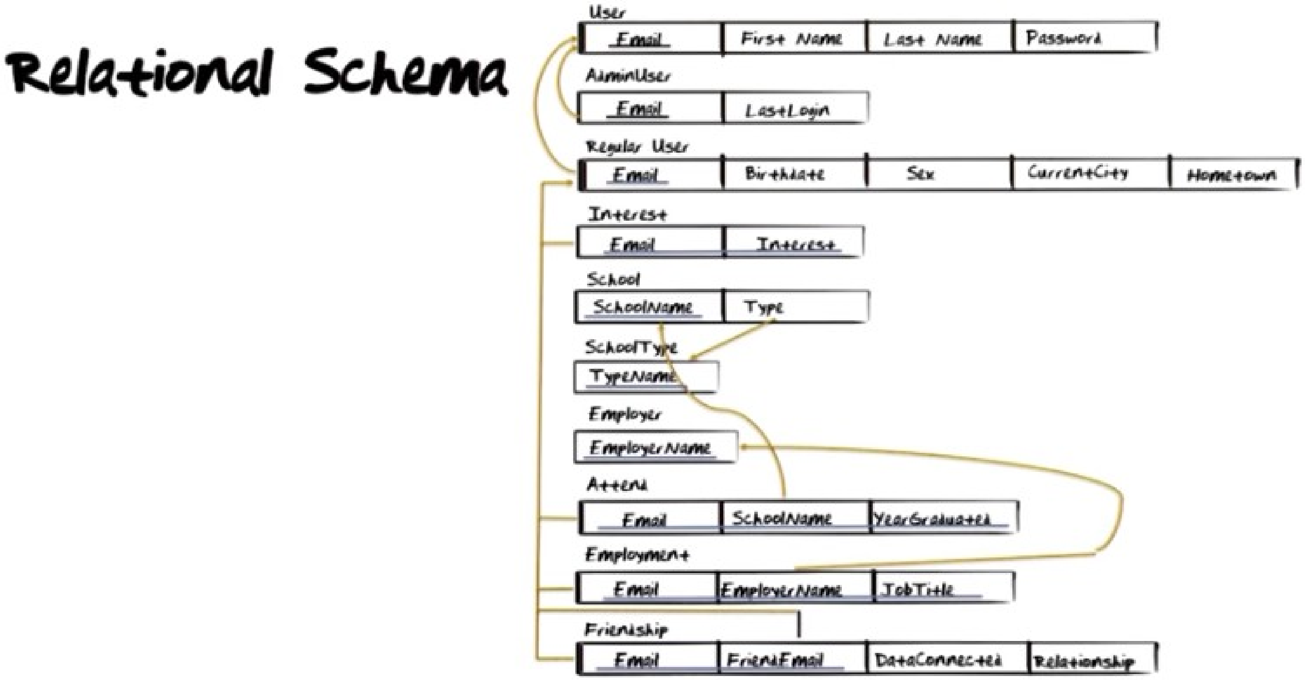
The above figure: 注意原圖中 User 和 d 之間是粗線. 弄 d 時, there were couple of options in the mapping. We have chosen the option where we create a relation for each one of the entity types.

Good: 對於 RegularUser and AdminUser, both of those will inherit Email, Passowrd, FirstName, and LastName. However, in the mapping of each one of these entity types, the only one of these that will appear is the identitying property type Email, which would become the primary key in the AdminUser relation and in the RegularUser relation. 當然 RegularUser 和 AdminUser 自己的 extra 的 property 還是要加上.



上圖複雜的地方在於 RegularUser 和 Friendship 之間有兩個 identification relationship. 所以對應的 Friendship 表中, it will have email twice. 其中的 Email 即 email of the one making the request(即對應 request 這個 relationship), FriendEmail 即 email of the friend that the request is being made to(即對 accept 這個 relationship).

Notice that there are two 1-to-many relationships into friendship. And typically when we have a many-to-many relationship type, that can actually be modeled by two 1-to-many relationships into the same entity type. When that's the case, then the combination(即 Email 和 FriendEmail) is a key. That corresponds very much all the cases we've seen of many-to-many relationship type where you take a key from each end of the relationship.



The above figure: Before we go any further, we need a small detour to study an additional database tool or technique.