





Comments on Swimming ERD

- Attributes – Appropriate attributes for Event, Heat, Participant, Org, and Univ were not specified. At a minimum, each needs a key attribute, so a simple ID attribute is used here. Providing additional attributes that are obvious from context (e.g., swimmer name) is acceptable.
- Event – As shown here, Event represents a particular stroke & distance combination independent of any Meet. Alternatively, Event could be related to Meet, and Heat then be related to and dependent on only Event.
- Heat – Heat is considered a weak entity because heats are normally only referred to as, e.g., “Heat 7 of the 100m Butterfly at the Rice Invitational”, i.e., dependent on the Event and Meet. (Acceptable to not be weak, if you argue such.)
- Org – We could have a second disjoint subclass of non-university organizations. However, it needlessly adds another entity set.
- Meet attributes – A meet could have an end date instead of num_days. However, then you’d want a constraint that the end date was after the start date. It’s arguably simpler to store the duration, instead, and constrain it to be positive.
- Stroke, Distance – These are just lookup tables. An alternative would be to have one lookup table of appropriate stroke-distance combinations, which would then be related to the Stroke and Distance lookup tables.
- Team/Univ – Since there is only one team per university, there’s no need to distinguish a separate Team entity set.
- Host – While it would be realistic to allow multiple Orgs to host a single Meet, this was not allowed by the specification.
- A swimmer “specialization” is a nebulous concept that aren’t ever defined or used, so trying to explicitly represent it is a bad idea. With the given information, it would be at most a calculated attribute, with its value calculated based upon what a swimmer has swum in.
- Attend – The possibility of a swimmer transferring schools isn’t mentioned, and so isn’t allowed here.
- Gender – Since each Event has a gender, then we also need to know the gender of each Swimmer. With this representation, the consistency of gender in an event could be checked by a query. Adding a gender attribute to Heat provides no benefit, but provides a chance for a Heat and Event to be inconsistent in gender. Subclassing Event, Heat, and Participant into separate genders (similar to the addition of relays) is also acceptable, but whether there is significant conceptual benefit for this complexity is debatable.

Comments on Swimming ERD – COMP 533

I thought of three basic approaches:

- Barely modifying the original diagram – Keep the diagram simple, as we did for gender. Adapting to relay races, however, will generally abuse some of the diagram semantics.
- Add individual- and relay-based subclasses to Participant, and possibly Heat and Event. Most accurate, but most complex. (Remember that the complexity can disappear during implementation if we only implement the superclasses.)
- Change the semantics of Event, Heat, and Participant so that every race is a relay, and that an individual race is just a relay with one leg. This allows future flexibility if relays of, say, 3 legs are used. Also simple. But allows inconsistencies such as 2 people in a 4-person event that we would want to check for.

I chose the last version due to a preference for extensibility and simplicity.

- More extensible means that the system design can handle some kinds of changes in system requirements. It is generally a good principle of software engineering.
- Added Leg to Swim relation to show the order of a swimmer in the relay.
- Added Leg to StrokeOf relation to show allow each leg of a relay to have a different stroke.