

is not having to look at the tables in order to find out which record contains the latest event for the `User_id` (for the fields). It is still more expensive than a query on the `but it's not too bad.` What can you do with an **Event Sourcing** design, that you couldn't with a traditional one? Locking. With no updates, no locking is required, and instead we can move from prohibiting clashes to resolving them later. This works well with sharding and high availability. What banks do. They don't use transactions for everything. It's important that a cash machine works even if its link is down. History. By adding `timestamp < X` clauses to the Current State Query we can see the state of an object at any point in the past. Undo. Because we store every event, we can undo changes by simply adding a new event. We can undo any change, not just the latest change. Debugging. We now know about every event in the system, which makes debugging a breeze. See Martin Fowler's page if you are unfamiliar with **Event Sourcing**.

Youth Sports Training with Weights. At what age should a young person begin lifting weights or using Kettlebells? The question I