

Week 4: Discussion Question

Under what circumstances are composite primary keys appropriate?

I have found some schools of thought online that consider composite primary keys a bad practice always. In fact, the Ruby on Rails and django applications frameworks do not support their use at all, forcing developers working with tables that have composite primary keys to find work-arounds and plugins to be productive (Dr. Nic, 2009). The argument appears to be that composite keys make object-relational mapping, pulling data from a relational database into an object-oriented application, overly complex (Walker, 2007).

The problem with the above is that it argues for less-efficient database design in certain instances in order to satisfy the shortcomings of an OO application architecture. Last week we talked about recursive relationships, with some examples of a table where each row has parent-child relationships. These relationships can only be rendered recursively when they are 1:M or 1:1. In the case where they are M:N, such as an environment where each child may have many parents and each parent many children, a separate table is useful to define these relationships (Rob and Coronel, 2009):

Parent	Child
1	3
2	3
3	2
3	4
4	1

In the above example the Parent and Child columns contain foreign primary keys to entities in another table. The composite primary key is the combination of PARENT and CHILD. In order to eliminate this composite key, we would need to create a third column with a unique identifier, but this would allow for duplicate PARENT – CHILD entries, which would require a constraint to prevent duplicates. This is more effort and more complex than simply using the composite primary key, which would constrain the data (Smith, 2007).

Id	Parent	Child
1	1	3
2	2	3
3	3	2
4	3	4
5	4	1
6	4	1

References:

Dr. Nic (2009). *Composite Primary Keys->Ruby on Rails->ActiveRecords*. Retrieved from rubyforge.org Aug 1, 2009 at: <http://compositekeys.rubyforge.org/>

Rob, Peter and Coronel, Carlos (2009). *Database Systems: Design, Implementation, and Management, Eighth Edition*. Course Technology, Boston MA.

Smith, Jeff (2007). *Composite Primary Keys*. Jeff's SQL Server Blog. Retrieved from sqlteam.com Aug 1, 2009 at: http://weblogs.sqlteam.com/jeffs/archive/2007/08/23/composite_primary_keys.aspx

Walker, Robert (2007). *Re: composite primary key doesn't work*. Retrieved from ruby-forum.com Aug 1, 2009 at: <http://www.ruby-forum.com/topic/131389#586540>