Subject: Re: Batch Script connecting two databases in an Extract Tranform Load (ETL) Design Pattern

Posted by AJM on Fri, 19 Jun 2015 09:20:29 GMT

View Forum Message <> Reply to Message

(1) Relationships are used as follows:

1.1 You cannot create a transaction from a transaction pattern with two or more entities (tables) unless thozse tables are related.
1.2 If a table has a parent, and you have defined one or more columns to be retrieved from that parent, when the default SQL statement is created the framework will automatically insert a JOIN to that parent and include the nominated columns in the SELECT list.
1.3 When deleting a record the validateDelete() method will check the status of any records on child tables to see what action needs to be taken.
1.4 If you have a many-to-many relationship you create two one-to-many relationships which relate Parent1 and Parent2 to a Child table (which is somethimes known as a LINK table, an XREF table or a INTERSECTION table). When you create transactions to deal with this relationship will need one version with Parent1 as the "outer" entity and a second version with Parent2 as the "outer" entity.

(2) What is the difference between putting my where clause in sql_where, vs into $where?

In your code you should ONLY be using $where. $object->sql_where is used in some controllers to specify additions to the $where clause which cannot be changed.

(3) Can the parent or child table class handle these inserts into the linking tables for me?

No. The default behaviour for any transaction pattern is that it will only write to a table that is specified in the component script. If you wish to update other tables then you will have to insert custom code in the relevant _cm_post_insertRecord(() or _cm_post_updateRecord() method.