**What Currently Happens**

1. The Competitor Codes (CompetitorCode\_quote) are kept in the div\_competitor\_adj data table.
2. The Competitor Code can be set at two times:
   1. Add Line Items (the first next)
      1. Via the “Pre Pricing Formulas” commerce library
   2. Finalize Quote
      1. Via the setTransactionCode commerce library
3. For case a:
   1. If the Sales Activity does not equal “New/New,” then we pull the Competitor Code from the div\_competitor\_adj data table based on the division #.
   2. If the division # is not in the table, the Competitor Code is set to blank.
4. For case b:
   1. The Competitor Code is set to blank.
   2. If the Price Adjustment Reason is “Roll back: competitive bid,” then we query the div\_competitor\_adj table based on the division and competitor name.
   3. If the division # - competitor name combo is not in the table or the Price Adjustment Reason is not “Roll back: competitive bid,” then the Competitor Code will be blank.

**Why do we Have a Problem**

1. There are only 6 divisions in the div\_competitor\_adj data table. So, even if we query the data table, we may not return a result.
2. The division is in the div\_competitor\_adj data table, but the Price Adjustment Reason is not correct, so the Competitor Code is set to blank.

**Proposed Solutions**

*Need to do both parts.*

1. In the setTransactionCode, do NOT initialize the Competitor Code to a blank value, instead initialize it as the current value. This will prevent the CC from being wiped out when the Price Adjustment Reason is NOT “Roll back: competitive bid.”

competitorCode = competitorCode\_quote;

1. Add all of the divisions and their competitors to the div\_competitor\_adj data table. This will ensure that whenever we query the table, we return a Competitor Code.