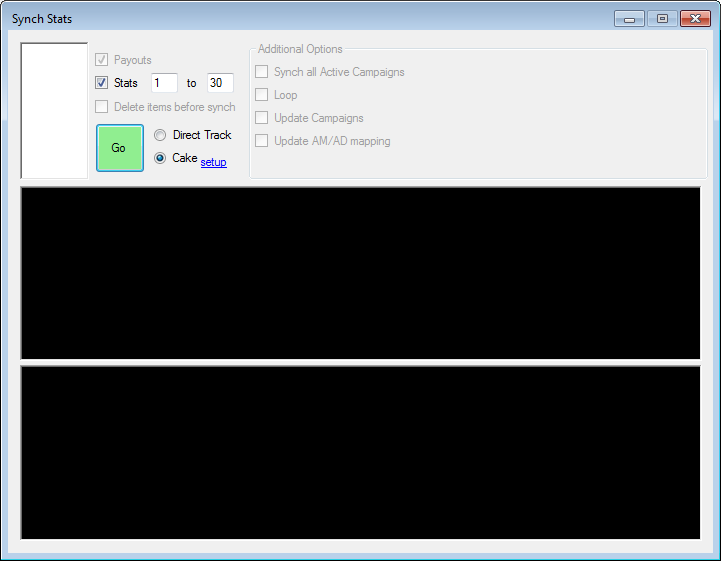
# Cake Syncher

The Cake Syncher is launched from the main menu by selecting **Campaigns->Synch**.

Then a Cake Offer ID is placed in the text box and the Go button is clicked.



Screen 1 - Synch (Campaigns->Synch)

## The following code executes when the Go button is clicked.

|  |
| --- |
| private void SynchStatsFromCake(int offerID, int startDay, int endDay)  {  int campaignID = offerID;  try  {  using (var db = EomDatabaseEntities.Create())  {  var campaignQuery = from c in db.Campaigns  where c.external\_id == offerID && c.TrackingSystem.name == "Cake Marketing"  select c;  var campaign = campaignQuery.FirstOrDefault();  if (campaign != null)  {  campaignID = campaign.pid;  }  }  }  catch  {  }  var parameters = new CakeSyncher.Parameters  {  CampaignId = campaignID,  CampaignExternalId = offerID,  Year = Properties.Settings.Default.StatsYear,  Month = Properties.Settings.Default.StatsMonth,  FromDay = startDay,  ToDay = endDay  };  var cakeSyncher = new CakeSyncher(this.Logger, parameters);  cakeSyncher.SynchStatsForOfferId();  } |

Code Listing - Synching to Cake

The inputs are:

* A Cake Offer ID.
* The number of the day in the current month to begin synching at.
* The number of the day in the current month to synch to.

The following logical steps are performed:

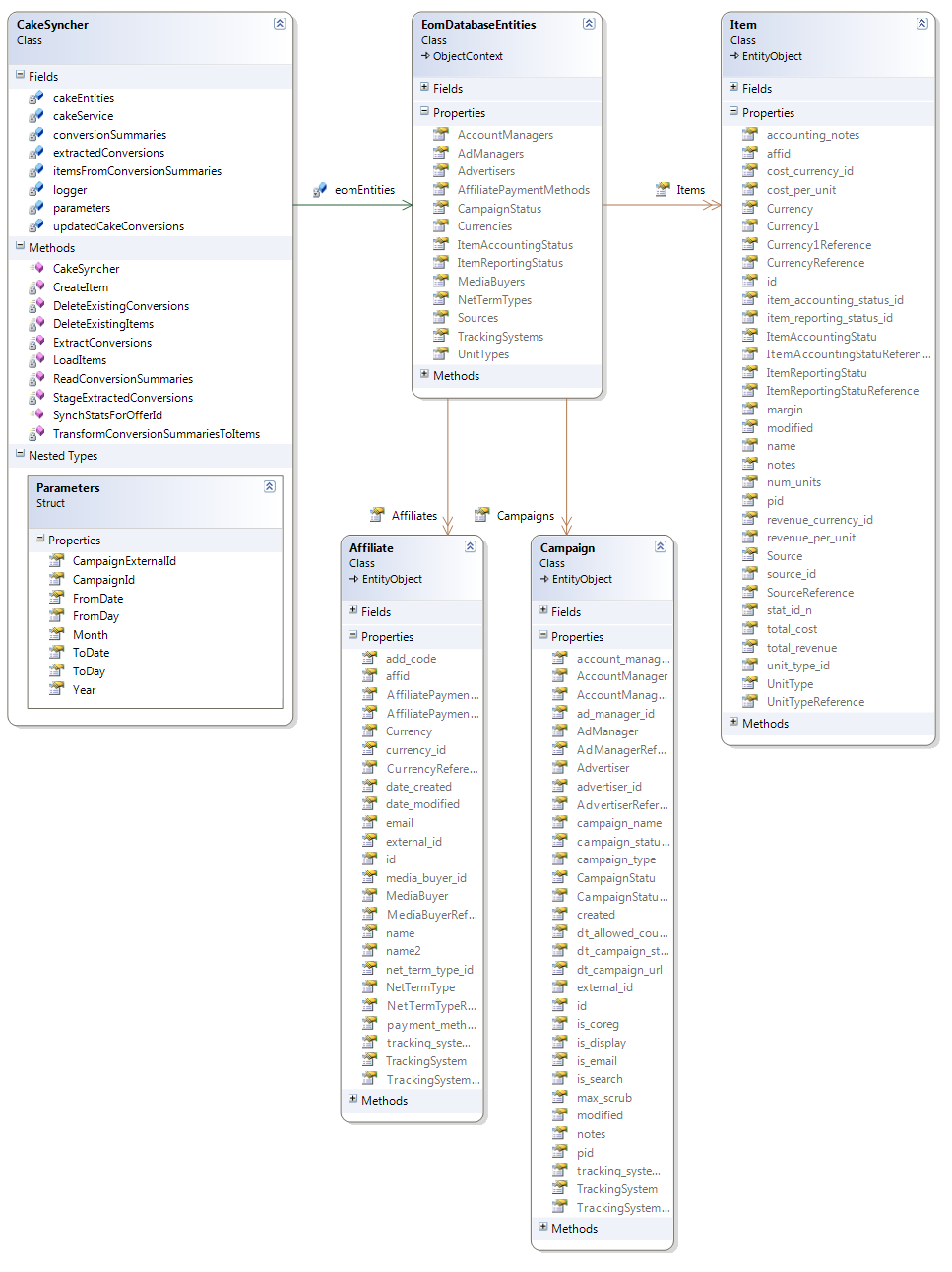
* The Campaign table is queried to determine if a row exists with the tracking system set to Cake and the external ID matching the Cake Offer ID. Otherwise, the Campaign with a PID matching the Cake Offer ID is selected.
* An empty catch clause is used to silently fail if the tool is being used in a month prior to the schema changes made for the Cake Syncher.
* A new Parameters object is created which is passed into a new CakeSyncher object.
* The properly configured CakeSyncher is invoked to do the work of synchronizing the stats.

## The primary database tables involved are as follows:

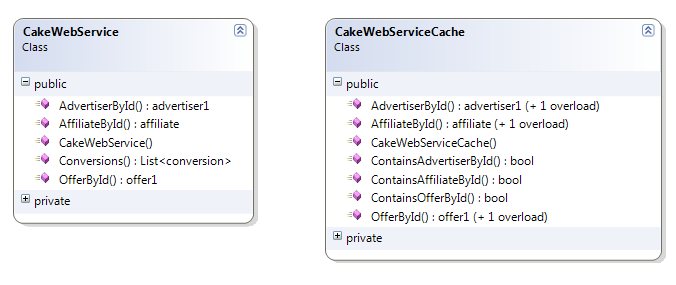


Database Diagram

## The primary classes are:



Class Diagram



Class Diagram

## The following code implements an ETL workflow to accomplish Cake synchronization:

|  |
| --- |
| public void SynchStatsForOfferId()  {  using (this.cakeEntities = new CakeEntities())  {  this.DeleteExistingConversions();  this.cakeEntities.SaveChanges();  }  using (this.eomEntities = EomDatabaseEntities.Create())  {  this.DeleteExistingItems();  this.eomEntities.SaveChanges();  }  using (this.cakeEntities = new CakeEntities())  {  this.ExtractConversions();  this.StageExtractedConversions();  this.cakeEntities.SaveChanges();  using (this.eomEntities = EomDatabaseEntities.Create())  {  this.ReadConversionSummaries();  this.TransformConversionSummariesToItems();  this.LoadItems();  this.eomEntities.SaveChanges();  }  }  } |

Code Listing