

Problems with DBT Files

There are several problems with **.DBT** files, which are:

1. An attached table is missing;
2. An index, just added, doesn't work;
3. The **.DBT** file is corrupt, resulting in inability to login to the application

The first problem, a missing table, has one of two probable causes:

- After creating/copying/attaching the table, the database definition was not updated (using [Define_Database](#), for example);
- Somehow the volume and table information for this table was out of sync, and the auto-synchronization code (see below) removed the table from the database.

Almost always, it is the first problem. To solve this, attach the table and use [Define_Database](#) to save the database definition. The fastest way to run [Define_Database](#) in OpenInsight is to open **Database Manager** and choose **Save** from the **Database** menu.

The second problem is similar to the first, but it shows itself in a different manner – a non-working index. When an index is added to a table that has no indexes, the system creates a new table to store the index information. This table is called a "bang table", because its name is the same as the data table but has an exclamation point (a "bang") glued on the front. The solution is to attach the data table (which will automatically attach the dictionary and index tables, if they exist) and then use [Define_Database](#) to save the database definition.

The last problem happens rarely, but it has happened. The easiest way to fix this is to pull the affected **.DBT** from a backup.



If the backup **.DBT** is lost, use the following procedure to recover:

1. Assuming your database name is **EXAMPLES**, so your **.DBT** file is named **EXAMPLES.DBT**;
2. Rename **EXAMPLES.DBT** to **EXAMPLES.BAK**, making a temporary backup;
3. Make a copy of **SYSPROG.DBT**; name the copy **EXAMPLES.DBT**;
4. Log into the **EXAMPLES** application.

What this does is start the **EXAMPLES** database out with just the system tables attached. Now you have to go back into Database Manager (or use the command line) to attach the tables necessary for your application. Once you have done that, use [Define_Database](#) to save the database definition. After you are happy that **EXAMPLES** is working well again, make a back up.

You should update your **.DBT** (using [Define_Database](#)) whenever the following occur:

- A table has been created that will be part of the application's database;
- A table has been deleted that was part of the application's database;
- A table has been attached that will be part of the application's database;
- A table has been detached that was and will no longer be part of the application's database;
- An index has been added to a table that is part of the application's database and that previously did not have an index;
- The last index on a table that is part of the application's database has been removed;
- One or more tables that are part of the application's database have been moved.



When using the system tools (as opposed to the command line), the **.DBT** file is usually updated in each of these cases except in the case of adding and removing indexes.