

# Data Structure Simplified

- [Tables](#)
- [Links](#)

The information here is mostly metaphorical examples of real life to help explain complex database concepts in regards to BBGM data model to serve more general audience.

## Tables

Imagine the database being made out of different spreadsheets which we call Tables. Every row on that table is representing one record of a type represented by that table. For example, if we are adding a Jane Doe as a contact to our database, here is how it would look like:

Table would be Contacts. IT would look like this:

ID	First_Name	Middle_Name	Last_Name	Birth_Date	Organization_ID
1111	Jane	Melanie	Doe	19690401000000	11

ID is a unique number that has no meaning and it identifies the record in that table/spreadsheet in our database.

In Blackbaud database, the column names are not the same as the labels we see in our views. The only time we would see the database name would be in our Advanced Search. Everywhere else, we will see the human readable labels.

## Links

Let's start this section with a story:

When we invite a guest to our house, we prepare for them. We make sure that we make space for them to come and we expect them to come. The other way around is not the case; they do not expect us to go to their house. When they show up to our house, we know what their name is and other information related to them. So, if anyone asks us who is your guest, we are able to tell them the name of our guest and their relationship to us and also their food allergies, because we were expecting them and we were cooking for them.

Imagine at the same time some birds flying and sitting on our roof. We don't expect them and we don't even know they exist. So, if anyone asks us about those birds, we would probably say "what birds?!". The birds however, know about our house. They were expecting to sit on a roof and that's what they did. There is only one way relationship between our house and the birds. Our house is going to be our house with or without the birds. But the birds need a roof to sit on.

The same idea is happening in our database. As we mentioned in Tables section, we can imagine tables as spreadsheets. So, like spreadsheets, a Request spreadsheet would not know anything about Organization spreadsheets unless we connect them together somehow. In databases, we can **Link** the spreadsheets that have relationship together. For example, a Request would like to know which Organization it belongs to. So, there is an empty seat in Requests spreadsheets that refers to the related Organization:

Organizations spreadsheet:

ID	Name	Tax_ID	EMail_Address	Primary_Contact_ID
<b>11</b>	Test Organization	111111111	test@org.com	111

Requests spreadsheet:

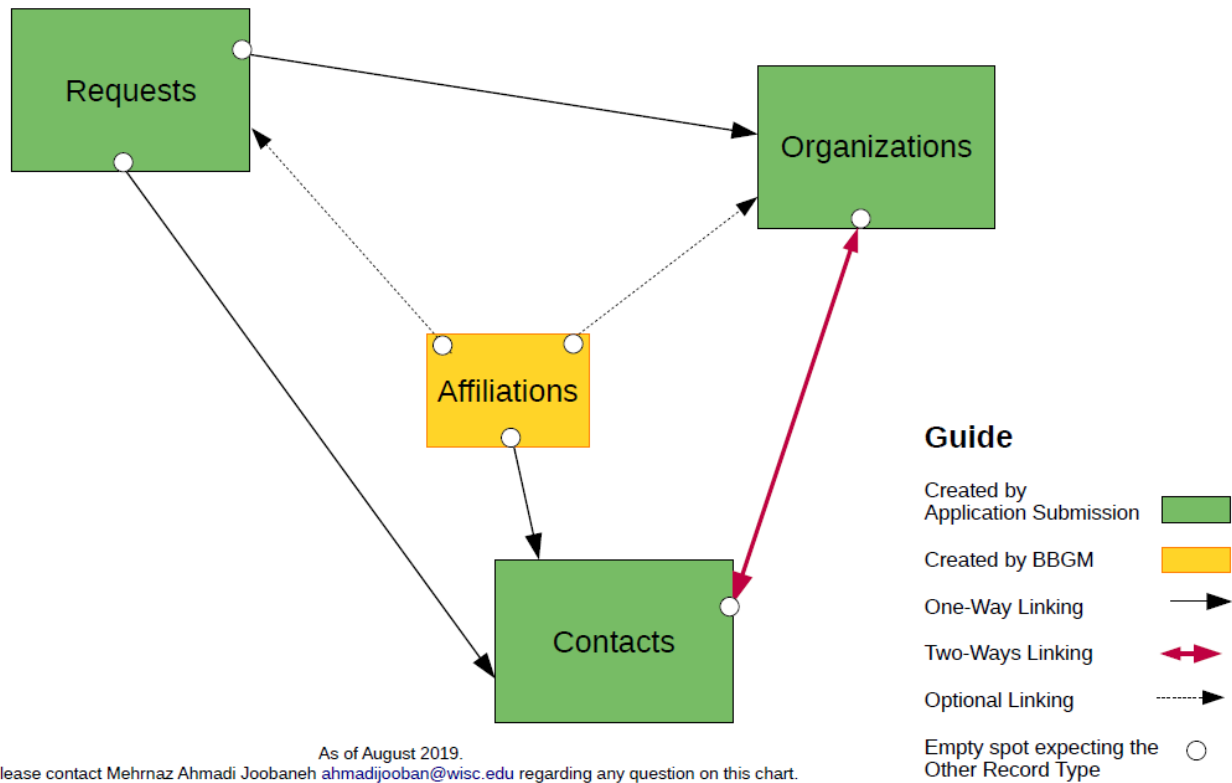
ID	First_Name	Middle_Name	Last_Name	Birth_Date	Organization_ID
1111	Jane	Melanie	Doe	19690401000000	<b>11</b>

In other words, there is one empty space for an organization in a certain request. That is why we can only have one organization for a request.

Organization however, is independent from the request. Requests are like birds on its roof. It doesn't even know that they are there (we can retrieve the information when we do a query).

The following chart is showing the main linking that is in place in Blackbaud Grantmaking Database:

## Main Data Links in BB GM



Downloadable version of the chart (PDF):



Links.pdf