# Create a Query containing Custom Fields
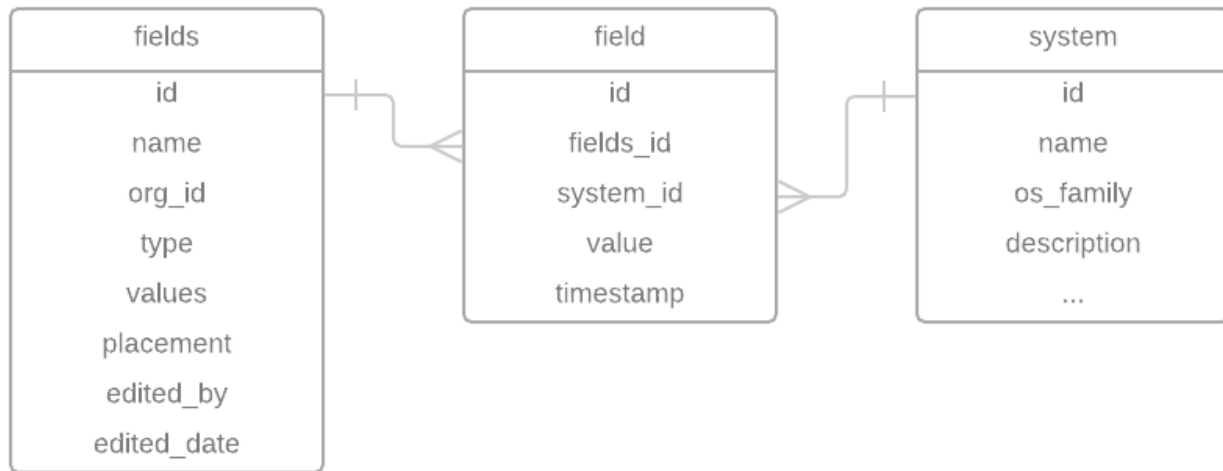
Creating custom queries to report on exactly the fields you need is what Open-AudIT excels at. But what if you need a field that Open-AudIT doesn't have out of the box? That's what custom fields are for. OK, so you have created some fields and now you wish to use those in a query (as well as in the device details). How would you do that? Some background information would be useful here.

Fields are defined by the user (ie, you) and stored in the database in a table called (surprise!) 'fields'. That table contains the definition for the field - its type, name, last modified, etc. When you assign a value for a device to a given field, that data is stored in another table called 'field' (note - no 's'). That table contains a link to the fields table, as well as the system (devices) table. You can see the relationships in the graphic below.



Say you have a field named "my field". You have devices that you wish to query and report on those that contain the value of "my value" in that custom field. How to do so?

You could approach this in several ways, but the simplest way is to know the ID of the field you wish to query. You need to join the system to the field table and specify the fields_id and the value you require. Say our "my field" has an id of 8 (you can see the fields.id in the web interface on the details page of the field in question). An example query SQL statement is below.

```
SELECT system.id AS `system.id`, system.icon AS `system.icon`, system.type AS `system.type`, system.name AS
`system.name`, field.value AS `field.value` FROM system LEFT JOIN field ON (system.id = field.system_id) WHERE
@filter AND field.fields_id = '8' AND field.value = 'my value'
```

If you don't care about the value and want all values reported, you could use the below.

```
SELECT system.id AS `system.id`, system.icon AS `system.icon`, system.type AS `system.type`, system.name AS
`system.name`, field.value AS `field.value` FROM system LEFT JOIN field ON (system.id = field.system_id) WHERE
@filter AND field.fields_id = '8'
```