

DBRefs

Suppose we have a database, named “a” having a collection “embedded”.
Here “embedded” collection contains one document.

```
> use a
switched to db a
>
> db.embedded.insert<<"name" : "mary", "age" : 22, "address" : {"home" : "delhi",
"residence" : "bangalore"}>>
WriteResult<< "nInserted" : 1 >>
>
> db.embedded.find<>.pretty<>
{
  "_id" : ObjectId<"54cc6b2b4bffd3788aac730">,
  "name" : "mary",
  "age" : 22,
  "address" : {
    "home" : "delhi",
    "residence" : "bangalore"
  }
}
```

Then we have another database, named “b” having “reference” collection.
Here reference collection again contain one document, which takes the \$ref, \$id, \$db fields for database references.

\$ref: This field specifies the collection of the referenced document

\$id: This field specifies the _id field of the referenced document

\$db: This is an optional field and contains name of the database in which the referenced document lies.

```
> use b
switched to db b
>
>
> db.reference.insert<<"name" : "sam", "dept" : "accounts", "links" : {"$ref" : "
embedded", "$id" : ObjectId<"54cc6b2b4bffd3788aac730">, "$db": "a"}>>
WriteResult<< "nInserted" : 1 >>
```

↑ here id shows the id of document in another collection

↑ db for the database

↑ ref for the reference of another collection

Then if we see the document in the reference collection, the document looks something like this. That in the links field it displays you the collection name and the id of that particular document.

And as you can see here that DBRef does not contain the \$db field, as you have entered at the time of insertion. Here DBRef displays you only the collection name "embedded" and the id of the document because this is the standard DBRef syntax.

```
> use b
switched to db b
> db.reference.find().pretty()
{
  "_id" : ObjectId("54cc6c104bffd3788aac731"),
  "name" : "sam",
  "dept" : "accounts",
  "links" : DBRef<"embedded", ObjectId<"54cc6b2b4bffd3788aac730">>
}
```

← go to another database, where reference is given for different collection.

↑ shows the reference of another collection and its id.

This is just to show you that we can do referencing of documents in the collection. But referencing within the same database is preferred instead of referencing between different databases. This is because mongoDB does not support inter database commands.