How to deserialize a BsonDocument object back to class

Asked 8 years ago   Active 3 years, 11 months ago   Viewed 41k times

▲

33

▼

★

15

🕐

How do I deserialize a BsonDocument object back to the class after getting it from the server?

```
QueryDocument _document = new QueryDocument("key", "value");
MongoCursor<BsonDocument> _documentsReturned = _collection.FindAs<BsonDocument>
(_document);

foreach (BsonDocument _document1 in _documentsReturned)
{
    //deserialize _document1
    //?
}
```

Do I deserialize using a BsonReader?

c#   mongodb   mongodb-.net-driver

edited Mar 24 '16 at 14:44   asked Feb 28 '12 at 8:29

Revious   iefpw
6,337 ● 25 ● 79 ● 125   **5,508** ● 14 ● 42 ● 70

1 Answer

▲

76

▼

✔

🕐

There are three ways actually:

1.Specify type you want to load directly in `FindAs<>`

```
var docs = _collection.FindAs<MyType>(_document);
```

2.Deserialize document via `BsonSerializer` :

```
BsonSerializer.Deserialize<MyType>(doc);
```

3.Map bson document manually to your class:

```
var myClass = new Mytype();
myClass.Name = bsonDoc["name"].AsString;
```

For most cases you are okay with first approach. But sometimes, when your documents is

An error occurred while deserializing the Property1 property of class Class1: ReadString can only be called when CurrentBsonType is String, not when CurrentBsonType is ObjectId. I put the bson objectid to guid of the class and mapped the class. How should I fix this? – iefpw Feb 28 '12 at 19:27

@iefpw: Error say that you have different types of Property1 in database and in class (objectId in database and string in class). – Andrew Orsich Feb 28 '12 at 20:33

I was trying to deserialize to the wrong class. Add "ObjectId Id { get; set; } to the class and basically everything works without class mapping. – iefpw Feb 29 '12 at 0:27