In this lecture, we will discuss…

✧ M:1 Embedded Linked

✧ `annotated` Link

✧ Demo

Relationships – Embedded Linked

✧ M:1 Embedded Link

✧ Many actors in a movie, but each play a specific role

✧ Movie Role ←→ Actor

```
class MovieRole
  include Mongoid::Document
  field :character, type: String
  field :actorName, as: :actor_name, type: String
  field :main, type: Mongoid::Boolean
  embedded_in :movie
  ...
  belongs_to :actor, :foreign_key => :_id
  ...
end
```

M:1 Embedded Linked (MovieRole <-> Actor)

```ruby
class Actor
  include Mongoid::Document
  field :name, type: String
  def roles


  #not supported
  #has_many roles:, class_name: 'MovieRole`
  #replaced with
  def roles
    Movie.where(:"roles._id"=>self.id)
         .map {|m| m.roles.where(:_id=>self.id).first}
  end
end
```

DEMO

Summary

✧ annotated link

✧ "1" side has a primary key and typically has no reference to the child within the document.

✧ "M" side will typically host the foreign key

What's Next?

✧ `1:1` – Linked