

# Documentos Embebidos



2024-03 —



10 min.

# Que es un documento embebido?

Un "documento" u "objeto" dentro de otro.

# Ejemplo

```
{
  "_id": 1,
  "name": "Ashley Peacock",
  "address": {
    "address_line_1": "10 Downing Street",
    "address_line_2": "Westminster",
    "city": "London",
    "postal_code": "SW1A 2AA"
  }
}
```

# Alternativa?

Documento referenciado.

# Documento referenciado

```
// en la collection de usuarios
{
  "_id": 1,
  "name": "Ashley Peacock",
  "address": 1000
}
```

```
// en la collection de direcciones
{
  "_id": 1000,
  "address_line_1": "10 Downing Street",
  "address_line_2": "Westminster",
  "city": "London",
  "postal_code": "SW1A 2AA"
}
```

# Documento referenciado (Opcion 2)

```
// en la collection de usuarios
{
  "_id": 1,
  "name": "Ashley Peacock",
  // sin campo address
}
```

```
// en la collection de direcciones
{
  "id": 1000,
  "user_id": 1, // se agrega el user_id
  "address_line_1": "10 Downing Street",
  "address_line_2": "Westminster",
  "city": "London",
  "postal_code": "SW1A 2AA"
}
```

# Documentos referenciados

- Es el "modus operandi" de las BDs relacionales.

# Referenciados vs Embebidos

- SQL tiene *joins* con buen performance, MongoDB no.
- Por eso, referenciados puede traer problemas de performance.



# Segun el ejemplo,

```
// en la collection de users
{
  "_id": 1,
  "name": "Ashley Peacock",
  "address": 1000
}
```

```
// en la collection de addresses
{
  "_id": 1000,
  "address_line_1": "10 Downing
Street",
  "address_line_2": "Westminster",
  "city": "London",
  "postal_code": "SW1A 2AA"
}
```

# Segun el ejemplo,

Para tener la info de un usuario (incluidas sus direcciones), habría que hacer 2 queries distintas y separadas.

# Referenciados vs Embebidos

Cuando uso uno o el otro?

Se considera bueno usar embebidos cuando:

- Ambos datos son leídos/modificados al mismo tiempo siempre o casi siempre

# Referenciados vs Embebidos

Si son, o serían, muchos documentos embebidos, mejor es que sean referenciados.

Esto es para reducir el tamaño del documento, y el tiempo de ejecución de una query.

# Referenciados vs Embebidos

Un tamaño de documentos agrandado haría modificar el documento mas costoso (en recursos y tiempo).

Esto es debido a que en MongoDB se lee todo el documento, y se re-escribe todo el documento.

# Casos

Para una aplicacion como Rappi,  
Los usuarios ↔ sus direcciones

# Casos

Para una aplicacion como Rappi,  
Los usuarios  $\leftrightarrow$  sus pedidos

# Casos

Para una aplicacion como Twitter,  
Los usuarios  $\leftrightarrow$  sus tweets



# Casos

Para una aplicacion como Rappi,  
Los restaurantes ↔ sus productos

# Casos

Para una aplicacion como Instagram,  
Los usuarios  $\leftrightarrow$  sus seguidores + seguidos