

# MongoDB Schema

---

# One-to-Few

```
{  
  name: 'Kate Monster',  
  ssn: '123-456-7890',  
  addresses : [  
    { street: '123 Sesame St', city: 'Anytown', cc: 'USA' },  
    { street: '123 Avenue Q', city: 'New York', cc: 'USA' }  
  ]  
}
```

# One-to-Many

```
{  
  _id : ObjectID('AAAA'),  
  partno : '123-aff-456',  
  name : '#4 grommet',  
  qty: 94,  
  cost: 0.94,  
  price: 3.99  
}
```

# One-to-Many

```
{  
  name : 'left-handed smoke shifter',  
  manufacturer : 'Acme Corp',  
  catalog_number: 1234,  
  parts : [    // array of references to Part documents  
    ObjectID('AAAA'),    // reference to the #4 grommet  
    above  
    ObjectID('F17C'),    // reference to a different Part  
    ObjectID('D2AA'),  
    // etc  
  ]  
}
```

# One-to-Many

// Fetch the Product document identified by this catalog number

```
product = db.products.findOne({catalog_number: 1234});
```

// Fetch all the Parts that are linked to this Product

```
product_parts = db.parts.find({_id: { $in :  
product.parts } } ).toArray() ;
```

# One-to-Squillions

```
> db.hosts.findOne()  
{  
  _id : ObjectId('AAAB'),  
  name : 'goofy.example.com',  
  ipaddr : '127.66.66.66'  
}
```

```
> db.logmsg.findOne()  
{  
  time : ISODate("2014-03-28T09:42:41.382Z"),  
  message : 'cpu is on fire!',  
  host: ObjectId('AAAB')    // Reference to the Host  
document  
}
```

# One-to-Squillions

```
// find the parent 'host' document
> host = db.hosts.findOne({ipaddr : '127.66.66.66'});
// assumes unique index

// find the most recent 5000 log message documents linked
to that host
> last_5k_msg = db.logmsg.find({host:
host._id}).sort({time : -1}).limit(5000).toArray()
```

# MondoDB Schema Design

- Will the entities on the “N” side of the One-to-N ever need to stand alone?
- What is the cardinality of the relationship: is it one-to-few; one-to-many; or one-to-squillions?