MongoDB at Visibiz

Why and how we're using MongoDB in our application

Mike Brocious
Tech Lead
Visibiz, Inc.

mongoDB

Visibiz
KNOW YOUR NETWORK. SELL BETTER.

Why We're Here

- Discuss why we chose MongoDB at Visibiz
- Show how we're using it
- ~~Made mistakes~~ Learned along the way
- Not a sales pitch

mongoDB

Visibiz
KNOW YOUR NETWORK. SELL BETTER.

About Us

- Startup
- Founded April, 2010
- 8 employees
- Located just outside of Philadelphia, PA
- Social CRM
    - 'know your network...sell better'
- Currently in limited beta
    - Sign up at www.visibiz.com

mongoDB

Visibiz
KNOW YOUR NETWORK. SELL BETTER.

Physical Components

Apache

Tomcat
-
Grails application

MongoDB

Elastic Search

Unix

Amazon EC2

How Did We Get To MongoDB?

Application Requirements

- Application requirements for extensibility
    - customer extensible objects
    - customer definable objects
- Scalability

Extensible Objects

- We provide core objects
    - person, company, prospect, relationship, etc.
- Customer can add their own attributes
    - including relationships with other objects
- A 'person' object for customer #1 may not look like a 'person' object for customer #2

CUSTOMER #1

- Person
  - name
  - address
  - date of birth
  - employment history
    - list:
      - company
      - begin date
      - end date
      - job title

<= Core Attributes =>

<= Customer Defined Attributes =>

CUSTOMER #2

- Person
  - name
  - address
  - date of birth
  - gender
  - hobbies
    - list:
      - name

mongoDB

Visibiz
KNOW YOUR NETWORK. SELL BETTER.

Customer Definable Objects

- Give customers ability to define their own objects
  - collection of attribute names, types
  - relationships with other objects

Scalability

- Will eventually have large amount of data
  - social networks, blogs, articles, etc.
  - email
- Scaling should be (relatively) easy

What We Liked About MongoDB

What We Liked About MongoDB

- Dynamic schemas ("schema-free")
    - fit well with our extensibility requirements

What We Liked About MongoDB

- Dynamic schemas ("schema-free")
    - fit well with our extensibility requirements

- Document datastore
    - easy to understand, visualize objects
    - cmd shell, log files, debuggers, viewers

mongoDB

Visibiz
KNOW YOUR NETWORK. SELL BETTER.

Schema Flexibility Comparison

- Relational vs. document-based

- Support extensible 'person' object

Relational Database Example

Person Id	Name	Address	Date of Birth
1	Mike Brocious	Malvern, PA	6/10/1984
2	Doug Smith	Philadelphia, PA	2/24/1980

Person Id	User Defined 1	User Defined 2	User Defined 3	User Defined 4	...
1	Good Burger	3/20/1998	3/31/2010	Flipper	
1	Visibiz	4/1/2010	<null>	Tech Lead	
2	10gen	7/9/2006	3/18/2009	Engineer	

Column	Name	Type
User Defined 1	Company	String
User Defined 2	Begin Date	Date
User Defined 3	End Date	Date
User Defined 4	Job Title	String

mongoDB

Visibiz
KNOW YOUR NETWORK. SELL BETTER.

Document Datastore Example


```
{
    "_id" : ObjectId("4d87a4d32739a23b3c834b67"),
    "name" : "Mike Brocious",
    "address" : "Malvern, PA",
    "dateOfBirth" : "Sun Jun 10 1984 00:00:00 GMT-0400 (EDT)",
    "employmentHistory" : [
        {
            "company" : "Good Burger",
            "beginDate" : "Sun Mar 20 1998 00:00:00 GMT-0400 (EDT)",
            "endDate" : "Thu Mar 31 2010 00:00:00 GMT-0400 (EDT)",
            "jobTitle" : "Flipper"
        },
        {
            "company" : "Visibiz",
            "beginDate" : "Fri Apr 01 2010 00:00:00 GMT-0400 (EDT)",
            "jobTitle" : "Engineer"
        }
    ]
}
```

Document Datastore Example


```
{
    "_id" : ObjectId("4d87a4d32739a23b3c834b67"),
    "name" : "Mike Brocious",
    "address" : "Malvern, PA",
    "dateOfBirth" : "Sun Jun 10 1984 00:00:00 GMT-0400 (EDT)",
    "employmentHistory" : [
        {
            "company" : "Good Burger",
            "beginDate" : "Sun Mar 20 1998 00:00:00 GMT-0400 (EDT)",
            "endDate" : "Thu Mar 31 2010 00:00:00 GMT-0400 (EDT)",
            "jobTitle" : "Flipper"
        },
        {
            "company" : "Visibiz",
            "beginDate" : "Fri Apr 01 2010 00:00:00 GMT-0400 (EDT)",
            "jobTitle" : "Engineer"
        }
    ]
}

{
  "_id": ObjectId("4d87a9732739a23b3c834b6d"),
  "name": "Julie Harper",
  "address": "Ocean City, NJ",
  "dateOfBirth": "Thu Sep 22 1994 00:00:00 GMT-0400 (EDT)",
  "gender": "F",
  "hobbies": [
      "painting",
      "skateboarding",
      "reading",
      "cooking"
  ]
}
```

More MongoDB Goodness

More MongoDB Goodness

- Scalability
  - replication - easy to setup
  - sharding

More MongoDB Goodness

- Scalability
  - replication - easy to setup
  - sharding

- Uses JSON
  - "the new XML"
  - easy to build, parse and read
  - great support from tools, languages and services

mongoDB

Visibiz
KNOW YOUR NETWORK. SELL BETTER.

JSON

- JSON * (MongoDB + Groovy + Grails + JavaScript) == LOVE

- MongoDB == BSON/JSON

- Groovy == Map

- Grails == JSON converter/builder

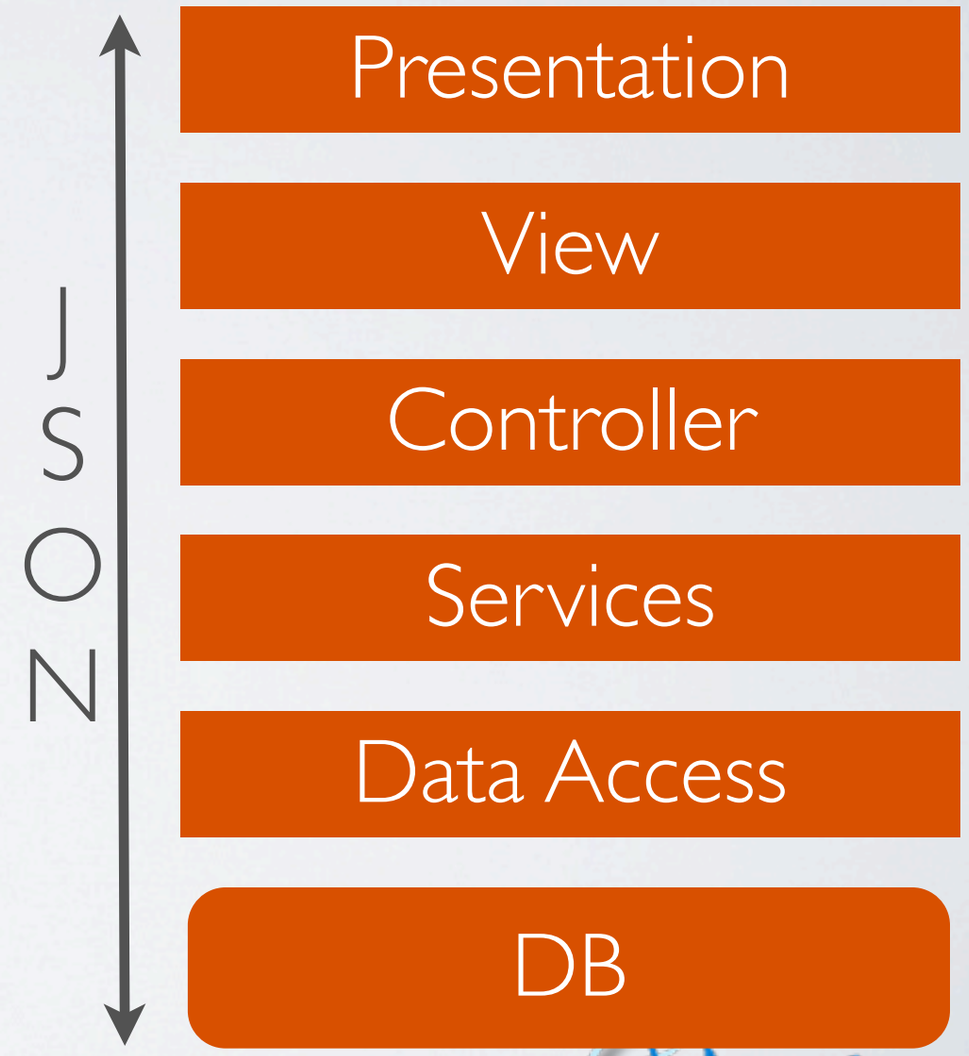
- JavaScript = Well...it's the JS in JSON

mongoDB

Visibiz
KNOW YOUR NETWORK. SELL BETTER.

JSON-eze

- Every layer of the application understands common format
- Not forced into transforming data
  - DB result sets <--> DO/DTO
  - DO/DTO <--> view
  - view <--> presentation
- Enables rapid development



mongoDB

Visibiz
KNOW YOUR NETWORK. SELL BETTER.

What Else We Liked About MongoDB

- Active product development
- Community support
  - plugins, drivers, viewers
- 10gen support
  - developers, CEO very active on forum, JIRA
  - MongoDB conferences, webcasts
- Deep list of sites already using MongoDB

Be Aware Of...

- No transactions
  - mitigation: schema design + atomic document updates
- No really complex queries (JOINs, nested SELECTs)
  - mitigation: schema design
- Felt we could minimize the impact

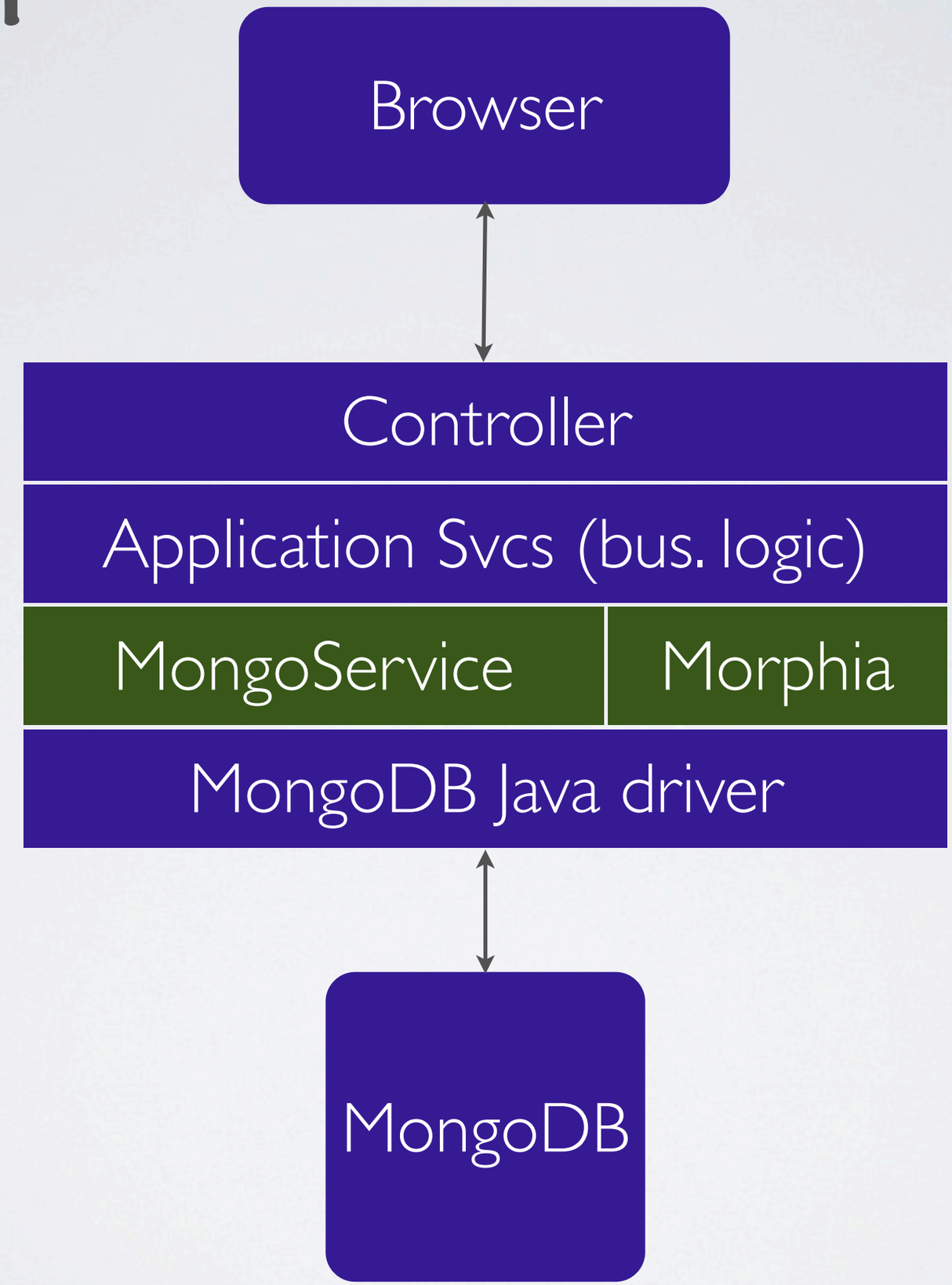
mongoDB

Visibiz
KNOW YOUR NETWORK. SELL BETTER.

- OK, so that's WHY

- Let's get into the HOW

Application Layers



Home-grown convenience wrapper around Java driver

Lightweight object-mapping library

mongoDB

Visibiz
KNOW YOUR NETWORK. SELL BETTER.

Our Primary Collections

- Things

- Rels

mongoDB

Visibiz
KNOW YOUR NETWORK. SELL BETTER.

Things

- Our main collection: "things"
- Domain objects (person, company, note, event, etc.)
- Customer-defined objects
- Takes advantage of 'schema-free' nature of MongoDB
  - able to easily query across all types

mongoDB

Visibiz
KNOW YOUR NETWORK. SELL BETTER.

'Person' Thing


```
{   "_id" : ObjectId("4d10f60f39fe153be3316478"),
    "thingType" : "person",
    "name" : {
        "firstName" : "Gail",
        "lastName" : "Staudt"
    },
    "owner" : ObjectId("4d10f47e39fe153b552e6478"),
    "createdDate" : "Tue Dec 21 2010 13:46:39 GMT-0500 (EST)",
    "tags" : [
        {
            "tag" : "java",
            "score" : 2
        },
        {
            "tag" : "clojure",
            "score" : 2
        }
    ],
    "addresses" : [
        {
            "addr1" : "40 Lloyd Ave",
            "city" : "Malvern",
            "state" : "PA"
        }
    ],
    "emailAddresses" : [
        {
            "type" : "work",
            "value" : "staudt@foo.com"
        }
    ]
}
```

'Company' Thing


```
{
    "_id" : ObjectId("4d10ffe939fe153b193b6478"),
    "thingType" : "company",
    "name" : "We Be Coders, Inc."
    "owner" : ObjectId("4d10f60e39fe153b20316478"),
    "createdDate" : "Tue Dec 21 2010 14:28:41 GMT-0500 (EST)",
    "primaryBusiness" : "Software development consulting"
    "tags" : [
        {
            "tag" : "software",
            "score" : 2
        },
        {
            "tag" : "development",
            "score" : 7
        },
        {
            "tag" : "clojure",
            "score" : 3
        },
        {
            "tag" : "java",
            "score" : 5
        }
    ]
}
```

'Company' Thing

```
{
    "_id" : ObjectId("4d10ffe939fe153b193b6478"),
    "thingType" : "company",
    "name" : "We Be Coders, Inc."
    "owner" : ObjectId("4d10f60e39fe153b20316478"),
    "createdDate" : "Tue Dec 21 2010 14:28:41 GMT-0500 (EST)",
    "primaryBusiness" : "Software development consulting"
    "tags" : [
        {
            "tag" : "software",
            "score" : 2
        },
        {
            "tag" : "development",
            "score" : 7
        },
        {
            "tag" : "clojure",
            "score" : 3
        },
        {
            "tag" : "java",
            "score" : 5
        }
    ]
}
```

- Find all things I own tagged with 'java'

```
> db.things.find({owner: ObjectId("4d10f60e39fe153b20316478"),
                  tags.tag: "java"})
```

Relationships - Act One

- Connections between things
  - employment history
  - co-workers, friends

mongoDB

Visibiz
KNOW YOUR NETWORK. SELL BETTER.

Relationships - Act One

- Connections between things
  - employment history
  - co-workers, friends

- First cut: separate documents in the things collection
  - essentially a mapping/join table

Relationships - Act One

mongoDB

Visibiz
KNOW YOUR NETWORK. SELL BETTER.

Relationships - Act One

- Single atomic insert (good! no transaction concerns)

- Made retrieval inefficient (no JOINs, nested SELECTs)

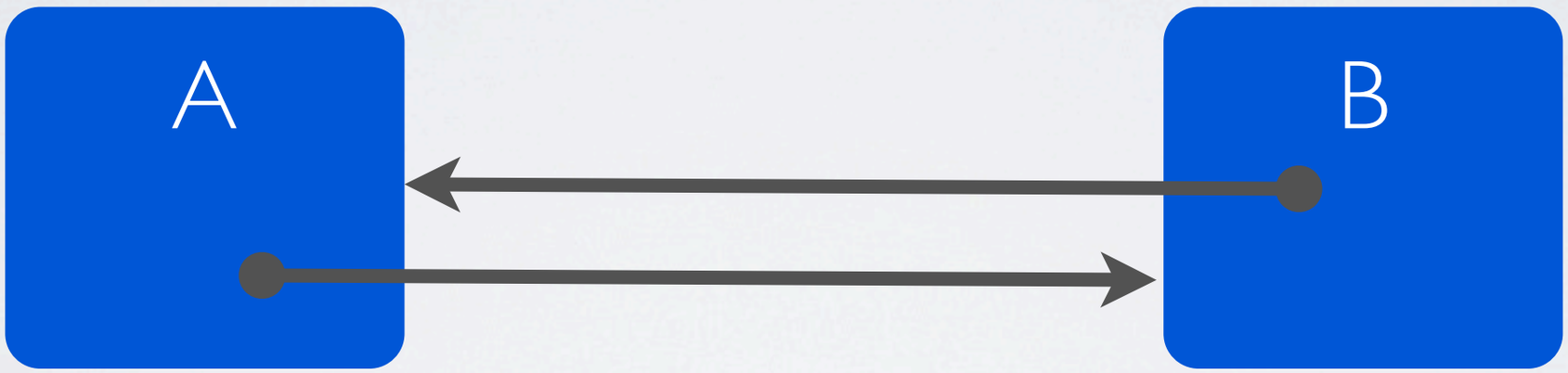
Relationships - Act One

Find all things I'm related to

```
me = ObjectId("4d10f60e39fe153b20316478")

// Get all the relationships I'm involved in
related = db.things.find({$or: [left.id: me,
                                right.id:me]})

// Build up a list of ids I'm related to
i = 0
relatedId = new Array()
for(relationship in related)
   if (relationship.left.id == me) {
       relatedIds[i++] = relationship.right.id
   } else {
      relatedIds[i++] = relationship.left.id
   }
)

// Now get the things
relatedThings = db.things.find({_id: { $in : relatedIds } })
```

Relationships - Act Two

- Moved relationships to nested document inside thing
  - natural approach for document-based datastores



Relationships - Act Two

- Queries easy and fast - awesome!

```
// Get all the things I'm related to
relatedThings = db.things.find({$or: [rels.left.id: me,
                                       rels.right.id:me]})
```

Relationships - Act Two

- Queries easy and fast - awesome!

```
// Get all the things I'm related to
relatedThings = db.things.find({$or: [rels.left.id: me,
                                       rels.right.id:me]})
```

- Two updates required to insert new relationship
  - relationship stored in both things
  - bad! - transaction concerns

mongoDB

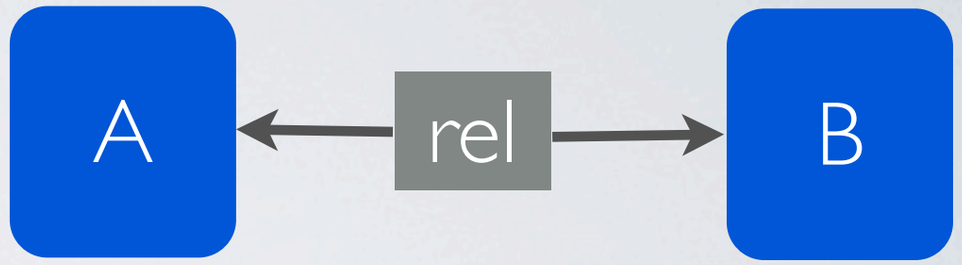
Visibiz
KNOW YOUR NETWORK. SELL BETTER.

Relationships - Act Three (final?)

- Best of both worlds

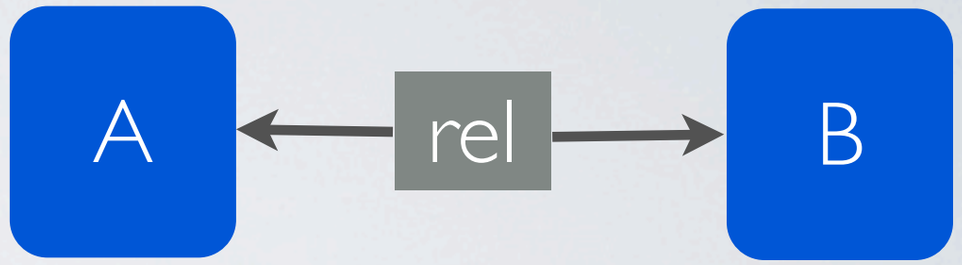
Relationships - Act Three (final?)

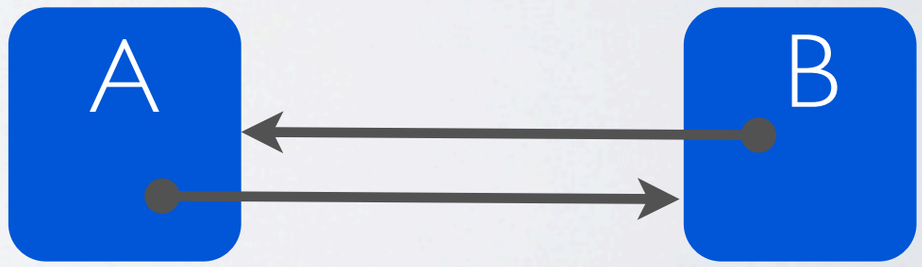
- Best of both worlds
- Separate "rels" collection
  - master source of relationship details
  - atomic insert (good!)
  - unique index (good!)



Relationships - Act Three (final?)

- Best of both worlds
- Separate "rels" collection
  - master source of relationship details
  - atomic insert (good!)
  - unique index (good!)
- Nested "rels" documents in things
  - easy, fast queries (good!)



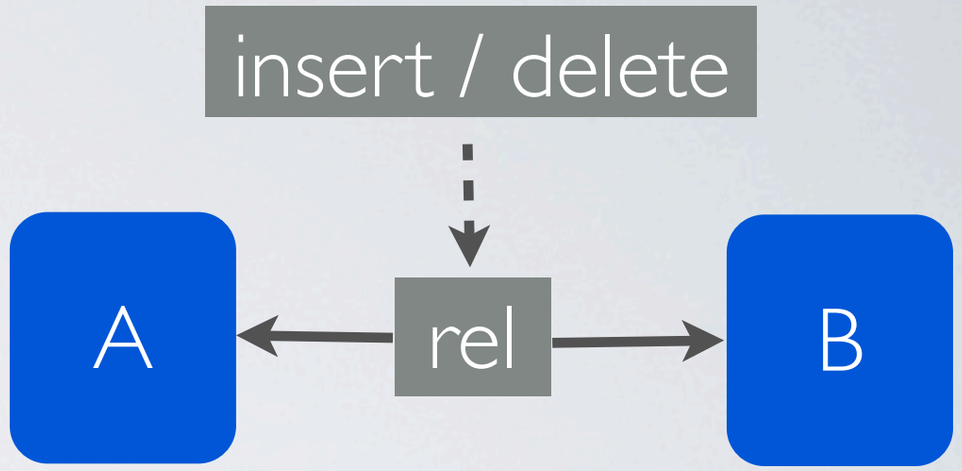


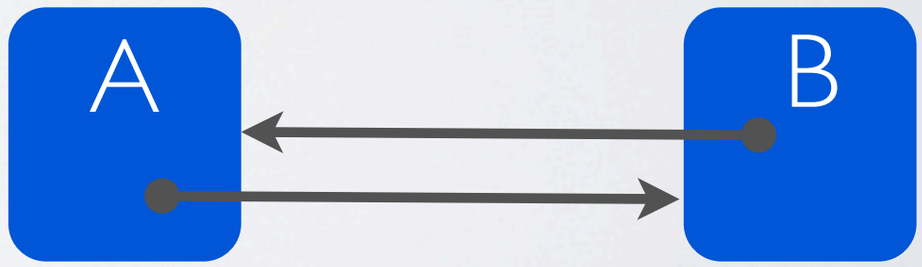
mongoDB

Visibiz
KNOW YOUR NETWORK. SELL BETTER.

Relationships - Act Three (final?)

- Best of both worlds
- Separate "rels" collection
  - master source of relationship details
  - atomic insert (good!)
  - unique index (good!)
- Nested "rels" documents in things
  - easy, fast queries (good!)



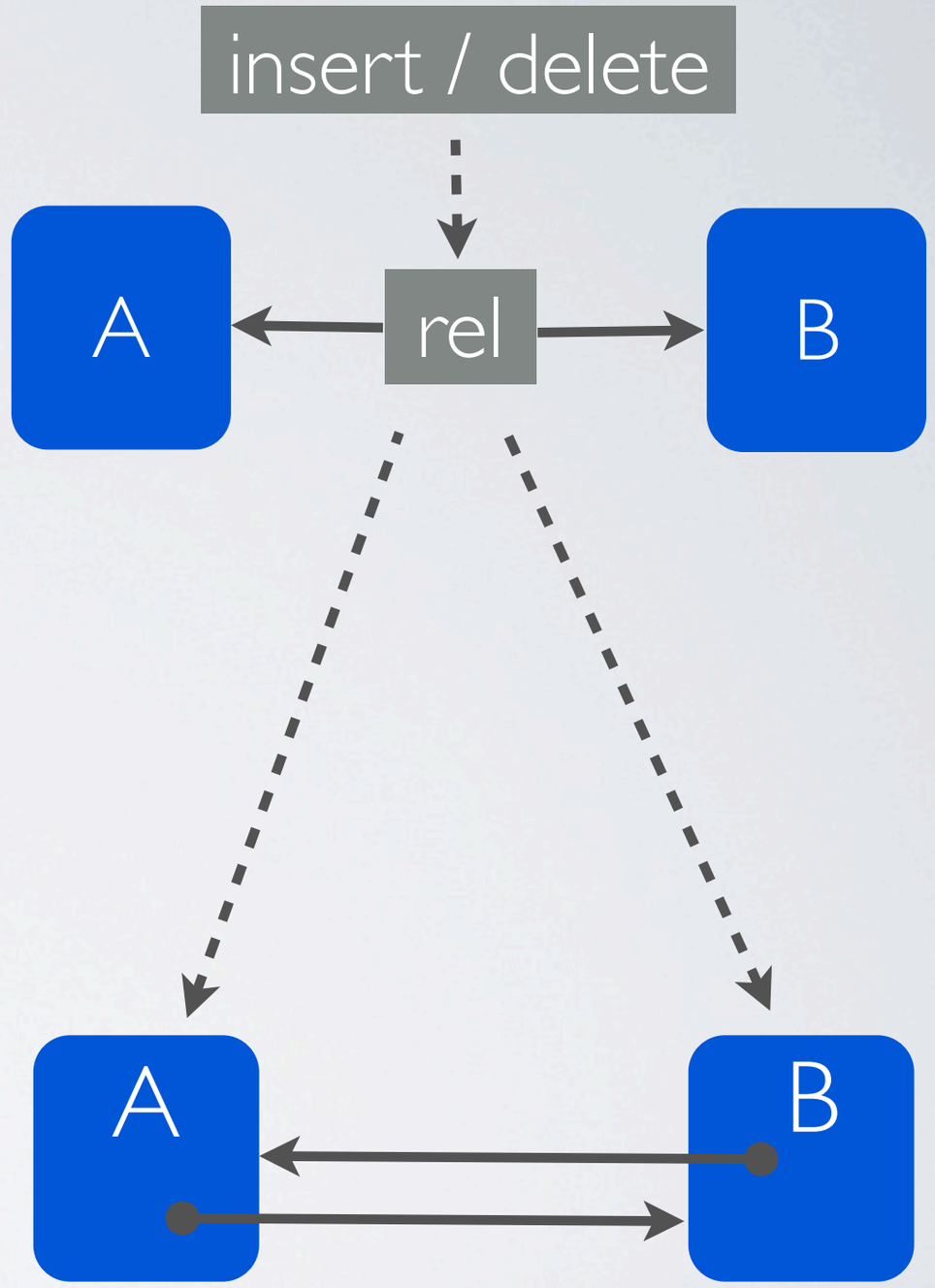


mongoDB

Visibiz
KNOW YOUR NETWORK. SELL BETTER.

Relationships - Act Three (final?)

- Best of both worlds
- Separate "rels" collection
  - master source of relationship details
  - atomic insert (good!)
  - unique index (good!)
- Nested "rels" documents in things
  - easy, fast queries (good!)



mongoDB

Visibiz
KNOW YOUR NETWORK. SELL BETTER.

Other MongoDB Collections

- Workflow status log
- Query log
- Staging area for imported data
- Users
- Duplicates scan log

mongoDB

Visibiz
KNOW YOUR NETWORK. SELL BETTER.

REST

- RESTful interface on top of services
- Expose services for internal and external development (API)

mongoDB

Visibiz
KNOW YOUR NETWORK. SELL BETTER.

REST

- RESTful interface on top of services
- Expose services for internal and external development (API)

- MongoDB doesn't enforce datatypes, object shape
- Need a way to validate data to prevent "garbage in"
- JSON schema
  - http://json-schema.org/

mongoDB

Visibiz
KNOW YOUR NETWORK. SELL BETTER.

Person Schema

```
person = [
    name: "person",
    type: "object",
    extends: "contact",

    properties: [
        name: [type: "object",
               title: "Name",
               properties: [
                   firstName:  [type: "string",
                                title: "First Name",
                                optional:true],
               middleName: [type: "string",
                            title: "Middle Name",
                            optional: true],
               lastName:   [type: "string",
                            title: "Last Name",
                            optional: true],
               suffix: [type: "string",
                        title: "Suffix",
                        optional: true]]]]]
```

mongoDB

Visibiz
KNOW YOUR NETWORK. SELL BETTER.

Thing Schemas

- Separate "schemas" MongoDB collection
- Every "thing" passes through validation before being stored

- Uses:
  - validate incoming data
  - track customer-specific schema extensions
  - generate UI to display things

Summary

- Why MongoDB works for us
  - Schema-free
  - Document-oriented
  - JSON
  - Scalable
  - Active product
  - Free

mongoDB

Visibiz
KNOW YOUR NETWORK. SELL BETTER.

Questions?