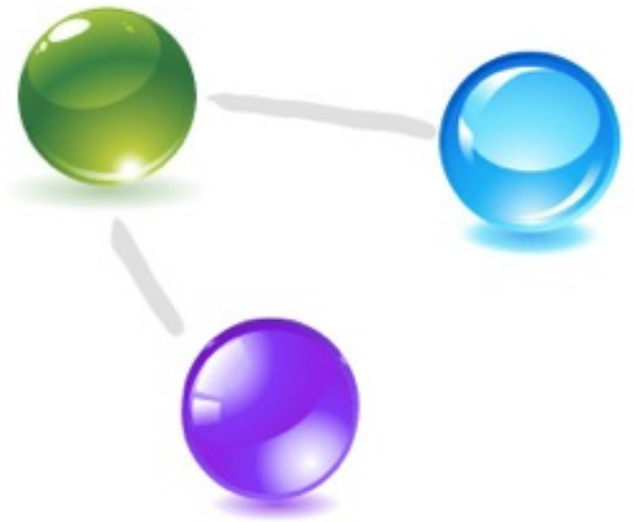
Build a SPA in 30min

by
Michael Hackstein
@mchacki

Michael Hackstein
@mchacki

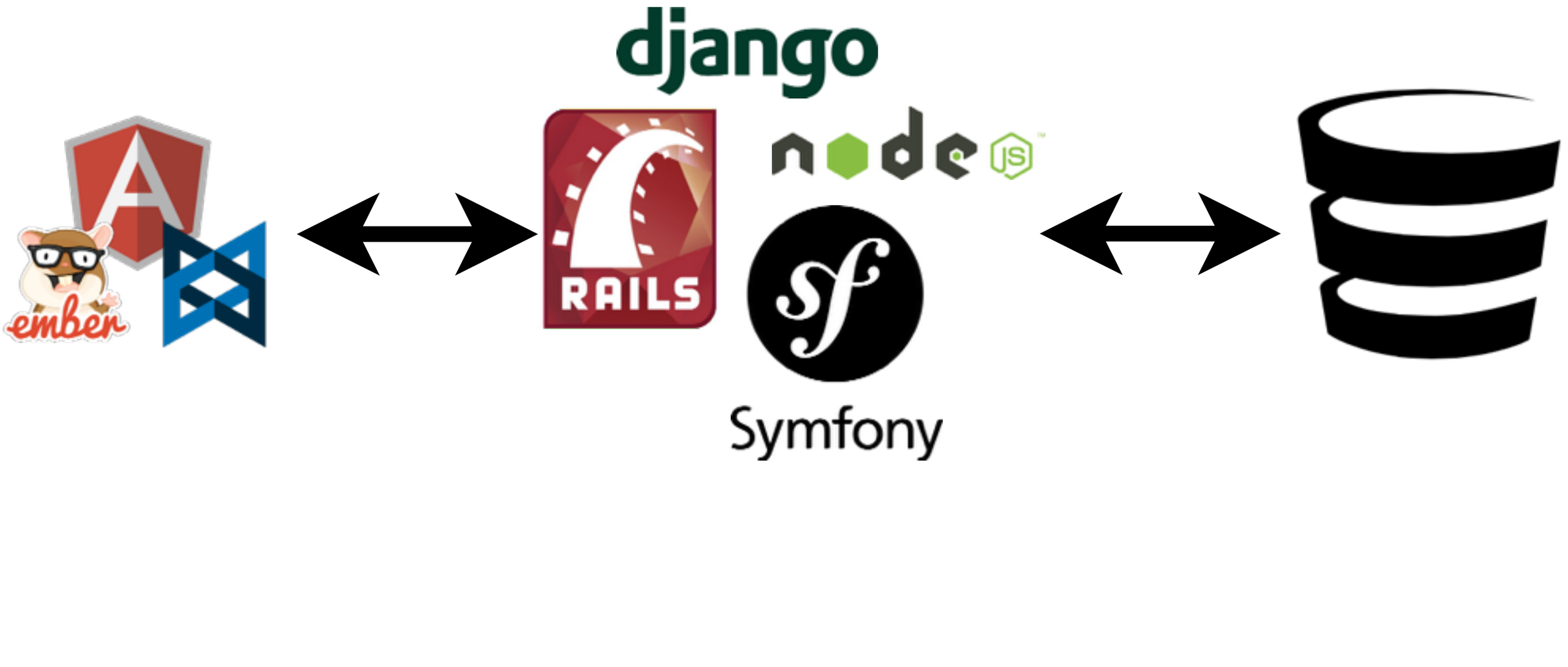
- Member of ArangoDB Core Team
  - web front-end
  - graph features
- Organizer of cologne.js
- Master's Degree (spec. Databases and Information Systems)



RWTH AACHEN UNIVERSITY

RWTH i5

Single Page Web Applications

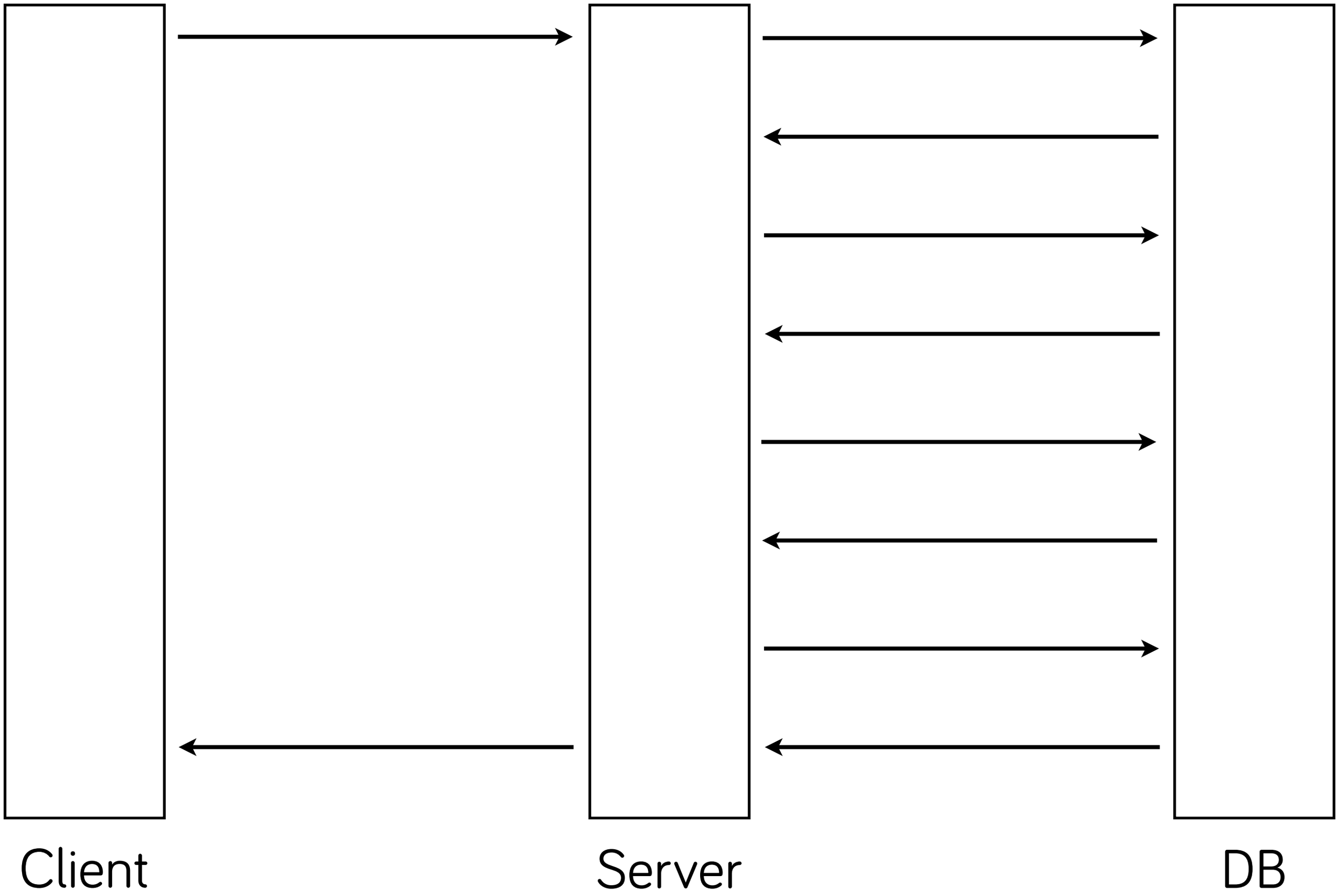


The Idea

- What if we could talk to the database directly?
- It would only need an API
- What if we could define this API in JavaScript?

Single Page Web Applications

This doesn‘t mean its a Rails/… Killer


Client
Server
DB

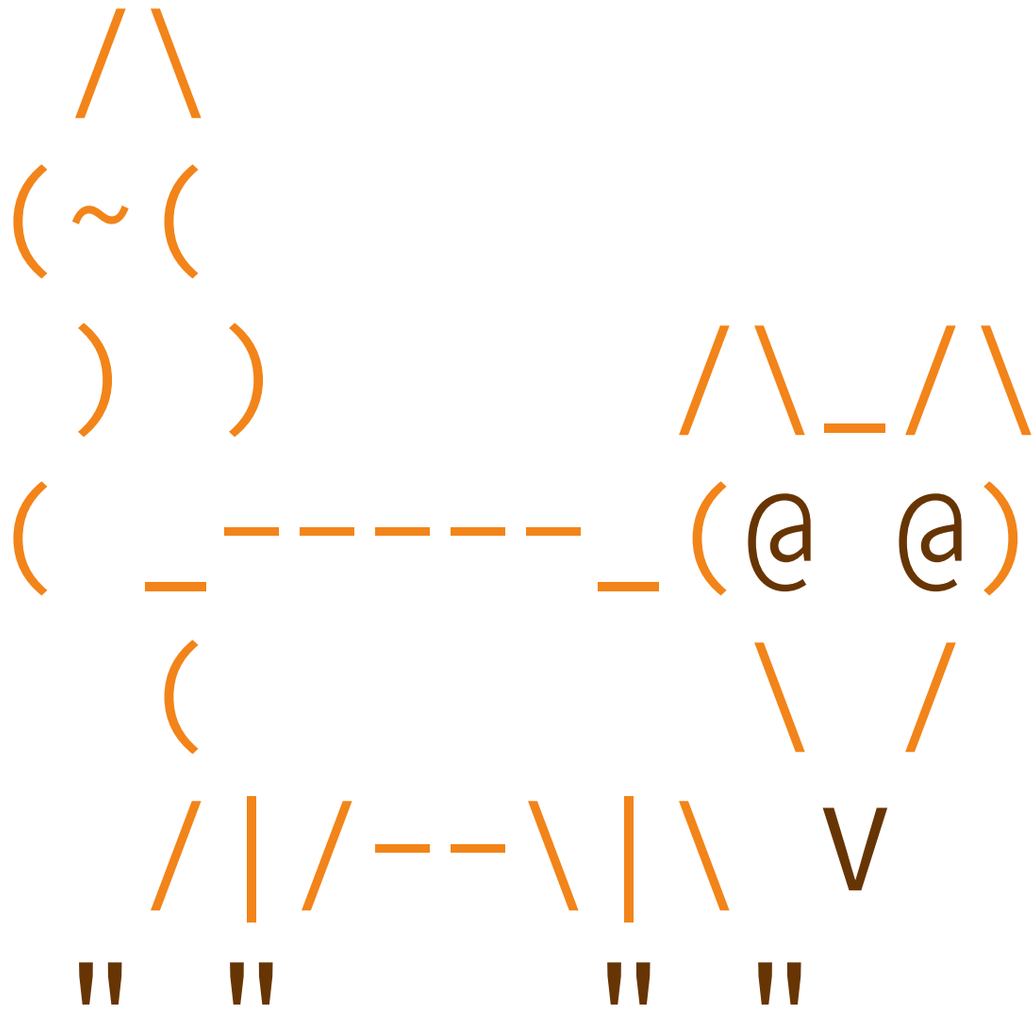
Client

Server
(optional)

DB
with Foxx

What is  ArangoDB ?

- Free and Open Source...
- ... Document and Graph Store...
- ... with embedded JavaScript...
- ... and an amazing query language



Foxx is...

- ... a feature of ArangoDB since 1.4
- ... an easy way to define REST APIs on top of ArangoDB
- ... a toolset for developing your single page web application
- ... not requiring any special code on the client side – use it with Angular, Backbone, Ember,...

Why another solution?

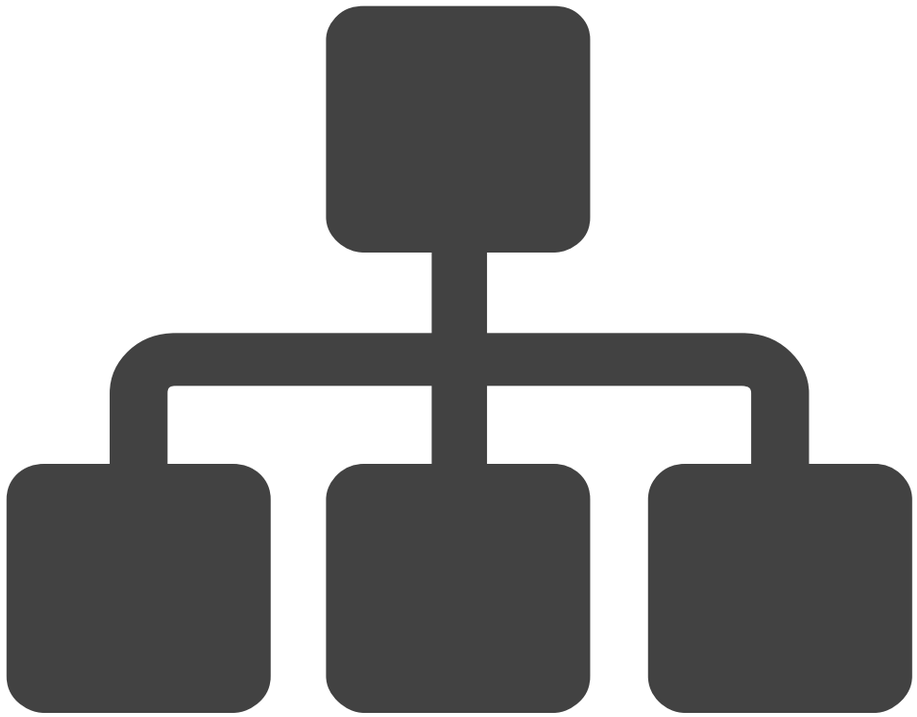
- ArangoDB Foxx is streamlined for API creation – not a Jack of all trades
- There‘s no communication overhead between (serverside) application and database
- It is designed for front end developers: Use JavaScript, you already know that

Live Coding Controller

```
var FoxxApplication = require("org/arangodb/foxx").Controller;
var controller = new FoxxApplication(applicationContext);
var db = require("internal").db;
var Heroes = require("repositories/heroes").Repository;
var Hero = require("models/hero").Model;
var heroes = new Heroes(db.heroes, Hero);
/** Load random
 *
 * Load a random hero
 */
controller.get("random", function(req, res) {
  res.json(heroes.random());
});
/** Load hero
 *
 * Load a specific hero
 */
controller.get("superheroes/:id", function(req, res) {
  res.json(heroes.byId(req.params("id")).forClient());
}).pathParam("id", {
  type: "string",
  description: "The key value of a hero"
});
/** Replace a hero
 *
 * replace a hero with new values
 */
controller.put("superheroes/:id", function(req, res) {
  var hero = req.params("hero");
  if(hero.get("_key") !== req.params("id")) {
    throw new Error("Id missmatch");
  }
  heroes.replace(hero);
  res.json(hero);
}).pathParam("id", {
  type: "string",
  description: "The key value of a hero"
}).bodyParam("hero", "The new values for a hero", Hero)
.errorResponse(Error, 400, "Id missmatch");
```

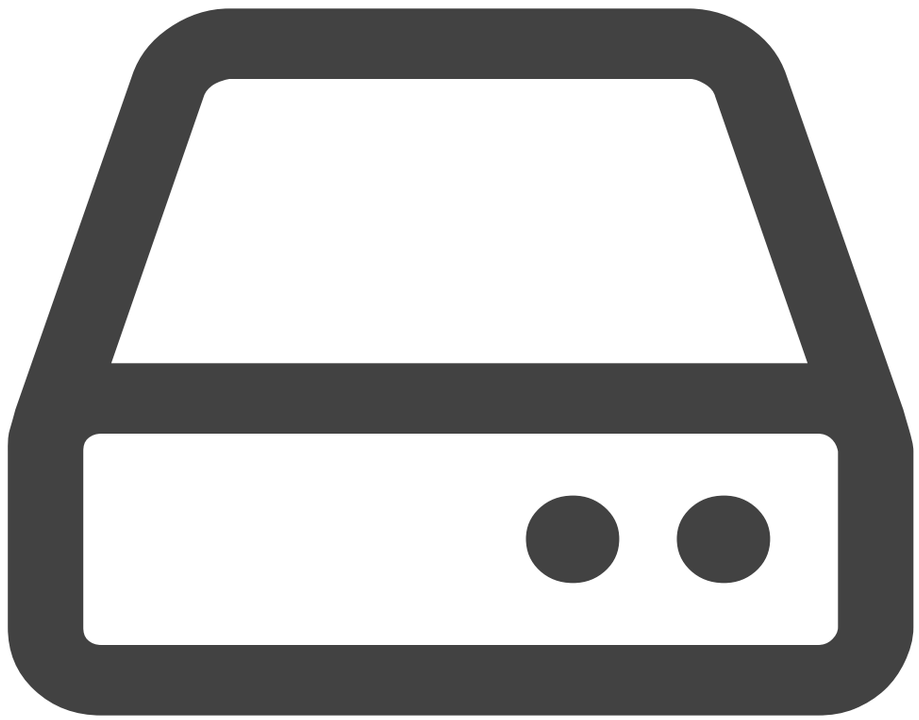
Foxx.Repository and Foxx.Model

Domain Models



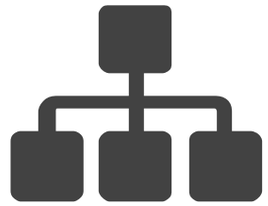
Foxx.Model

Persistence



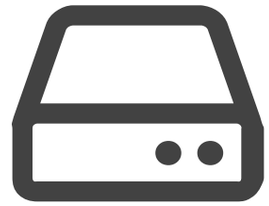
Foxx.Repository

Foxx.Model



- Representation of the data
- Convenience Methods
- Validation

Foxx.Repository



- Save and Retrieve Data
- Simple Queries
- Define your own queries

Why this separation?

- It doesn‘t violate the SRP like ActiveRecord
- In a lot of cases you can use the standard Repository or Model and don‘t need your own
- It‘s great for testing
  - You can mock the collection and the model prototype to test your Repository
  - You don‘t need to mock anything to test your model

Live Coding

Model & Repository

Model

```
var Foxx = require("org/arangodb/foxx");
var Hero = Foxx.Model.extend({
  // instance properties
},{
  attributes: {
    "_id": "string",
    "_key": "string",
    "name": "string",
    "comment": "string"
  }
});
exports.Model = Hero;
```

Repository

```
var Foxx = require("org/arangodb/foxx");
var Heroes = Foxx.Repository.extend({
  random: function() {
    return this.collection.any()._key;
  }
});
exports.Repository = Heroes;
```

Thanks

- Please try ArangoDB Foxx
- www.arangodb.org
- We ❤ to get feedback

Contact

- mchacki@arangodb.org
- @mchacki on Twitter

Thanks

- First Slide version by Lucas Dohmen
- Andreas Streichardt and Julian Steiner for giving an AngularJS & Foxx Training with me and elaborating the Idea for the App
- Database icon designed by Romeo Barreto from The Noun Project
- Logos from Node.js, Ruby on Rails, Django, AngularJS and Symfony from the respective projects
- All other icons are from Font Awesome