Storing Data

Every web app's dream

Allen Pike, Steam Clock Software
allenpike.com

Allen Pike

- SFU
- Apple
- Steam Clock Software
- VanJS

Plan

- Relational Data and SQL
- Object-Relational Mappers
- NoSQL Data Stores
- HTML5 Data Storage

RDBMS

Relational Data

- Schema
- Tabular data
- Queries
- Optimized query engine
- Results

SQL

```
CREATE TABLE test_subjects;

INSERT INTO test_subjects
 SET subject_name = "Chell", facilityID = 23;

INSERT INTO test_subjects
 ET subject_name = "Freeman", facilityID = 17;
```

```
SELECT subject_name, facility_name
 FROM test_subjects
  JOIN facilities ON facilityID;

 -------------------------------
 subject_name | facility_name
 -------------------------------
 Chell        | Aperture Science
 Freeman      | Black Mesa
 -------------------------------
```

Strengths of RDBMS

- ACID
- Well supported
- Insanely featureful
- Crazy fast (usually)

Options

- MySQL
- Postgres
- SQLite
- MSSQL
- Oracle

Admin

- mysql command line
- mysqldump
- phpMyAdmin
- Sequel Pro

Demo

Connecting

- Server-side only
- Sessions
- Libraries

ORM

Relational Mappers

- Map data into objects and back
- OO interface to SQL
- Less time writing CRUD

Avoiding SQL

- Makes easier tasks easier
- Makes harder tasks harder
- Good for getting started

OO Conventions

- Maps pretty well
- Less mental overhead
- Impedance mismatch

Performance Issues

- Lots of data
- Hidden costs
- Hard to optimize

Choices

- Ruby - Rails' ActiveRecord
- Java - Hibernate
- Python - SQLAlchemy or Django
- PHP, .NET, etc.

Rails Example

```
mesa = Facility.find(17)
aperture = Facility.find(23)

chell = aperture.testSubjects.create("Chell")
gordon = mesa.testSubjects.create("Freeman")

@subjects = TestSubjects.all
```

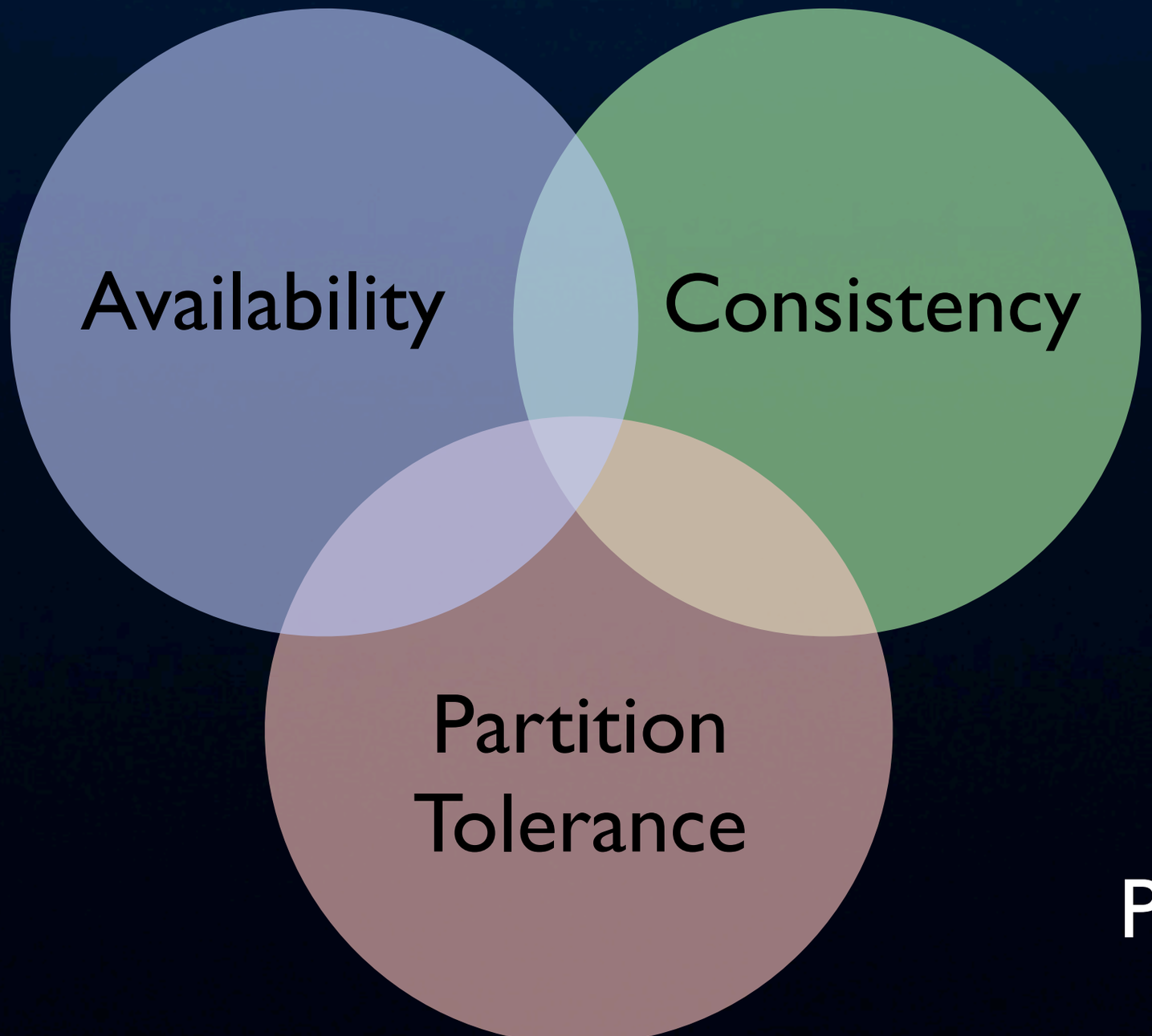
```
<!-- Later, in a template... -->

<table>
<% @subjects.each do |subject| %>
  <tr>
    <td><%= subject.name %></td>
    <td><%= subject.facility %></td>
  </tr>
<% end %>
</table>
```

Break

NoSQL

CAP Theorem



Pick two.

Hence, NoSQL

- Schemaless data stores
- Little ceremony
- Very fast
- Easier to replicate and scale

Weaknesses

- Not ACID
- Susceptible to data loss and error
- Less mature
- No fancy features

Document Stores

- CouchDB
- SimpleDB
- MongoDB

Key Value Stores

- Cassandra
- Riak
- Redis
- BigTable
- MySQL!

Browser Storage

Client Side

- Cookies
- Flash
- HTML5 indexedDB
- HTML5 localStorage
- HTML5 AppCache

Cookies

- Key-value
- Sent with every request
- Very limited in size

Flash

- Larger in size
- Not always available
- Evil

LocalStorage

IE 8, Fx 3.5, Cr 4, Mobile

- Local key value store
- More storage
- Not sent with every connection

IndexedDB

IE 10, Fx 4, Cr 11

- Not SQL (anymore)
- Schemas
- Can be overkill

AppCache

IE 10, Fx 3.5, Cr 4, Mobile

- Full file storage and caching
- Pre-fetching assets
- Hard to debug

Thanks!

Allen Pike, Steam Clock Software
allenpike.com