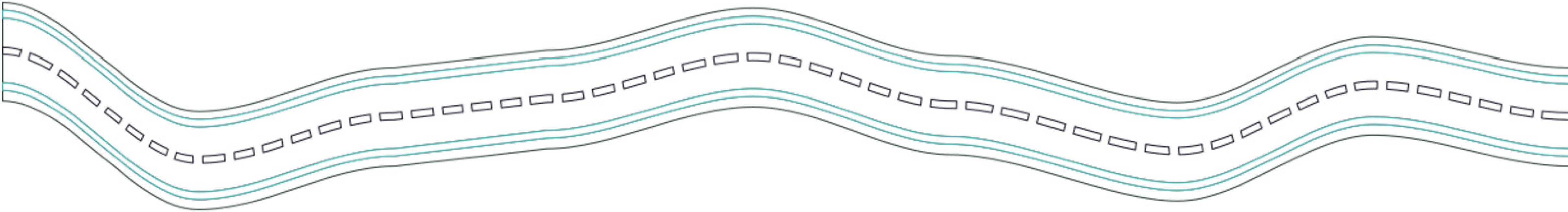
Road Trip Through Database Country

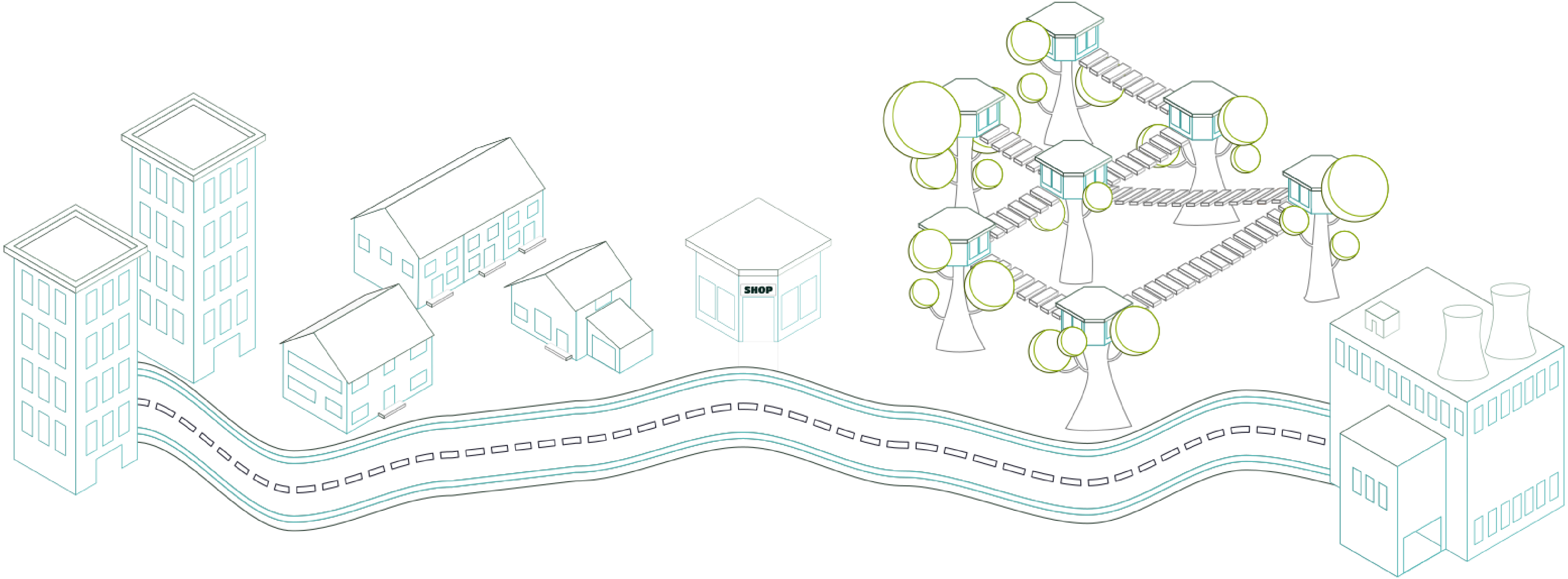


Lorna Mitchell, IBM

Data is the lifeblood of our applications, but we rarely study it. Why?

We rely on databases

Database Country

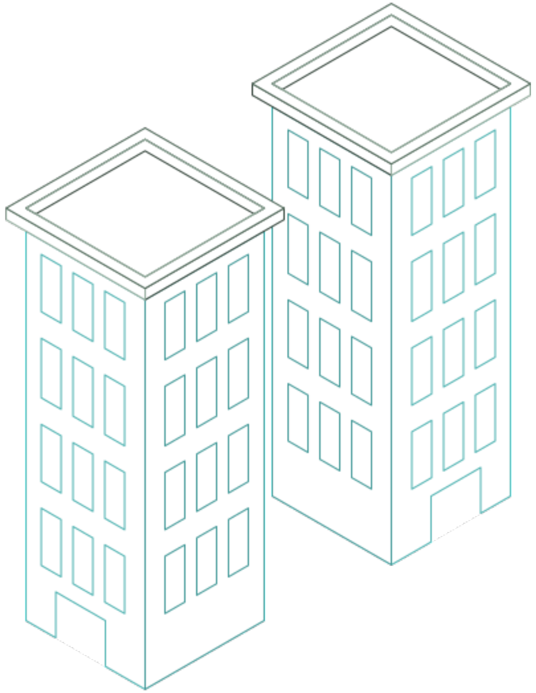


Relational Databases

Relational databases are brilliant if you need to relate different bits of data to each other.

For example: Order data

They are also reliable places to put things, implementing ACID compliance.



ACID Compliance

- Atomicity
- Consistency
- Isolation
- Durability

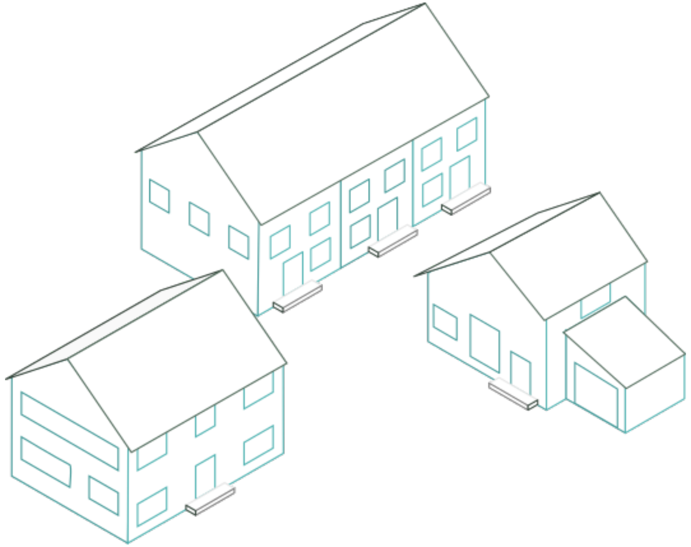
Upfront schema planning is required. Changing structure can be painful.

PHP and MySQL: BFFs

Life Advice: Learn SQL

Document Databases

- Schemaless, just add any JSON document
- Good to excellent performance
- Not usually ACID-compliant



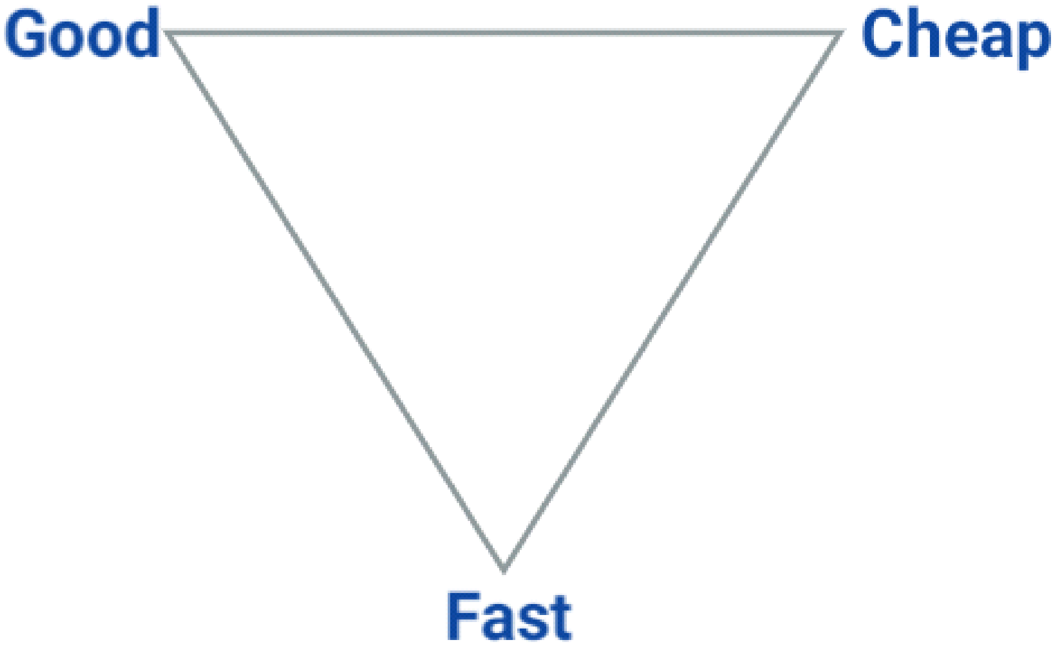
For example: product catalog, CMS data

Speedy and distributed

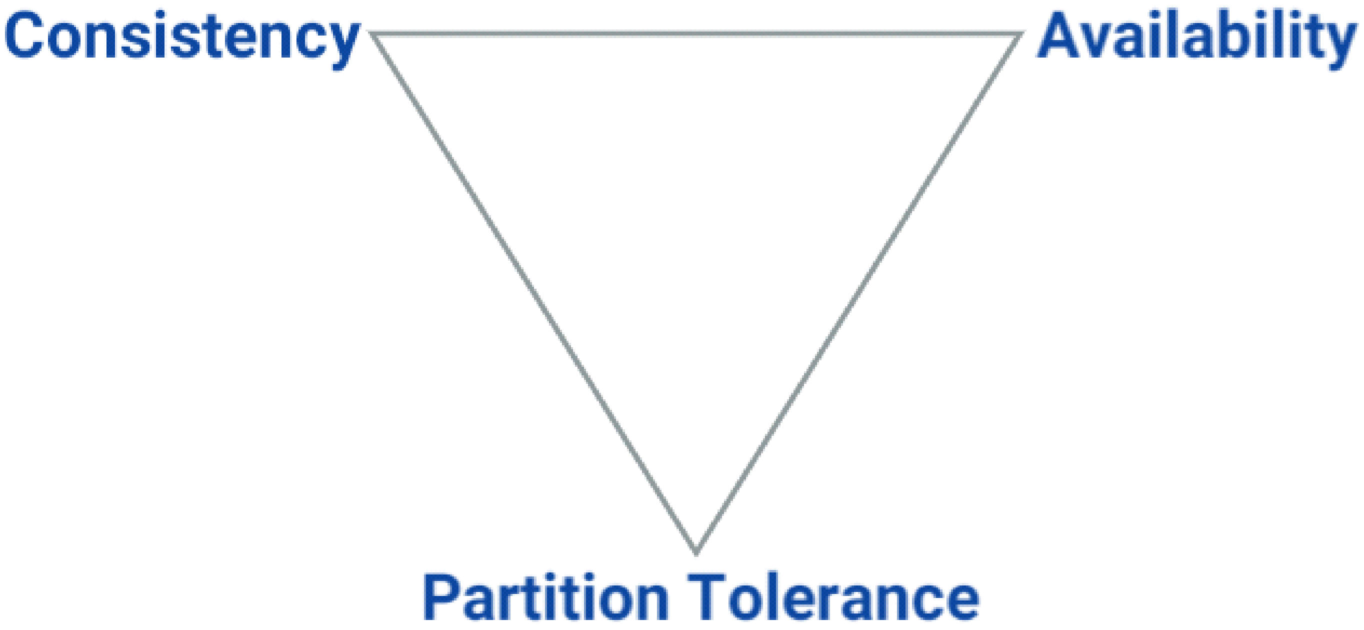
BASE

- Basic Availability
- Soft-state
- Eventually consistent

CAP Theorem for Distributed DBs

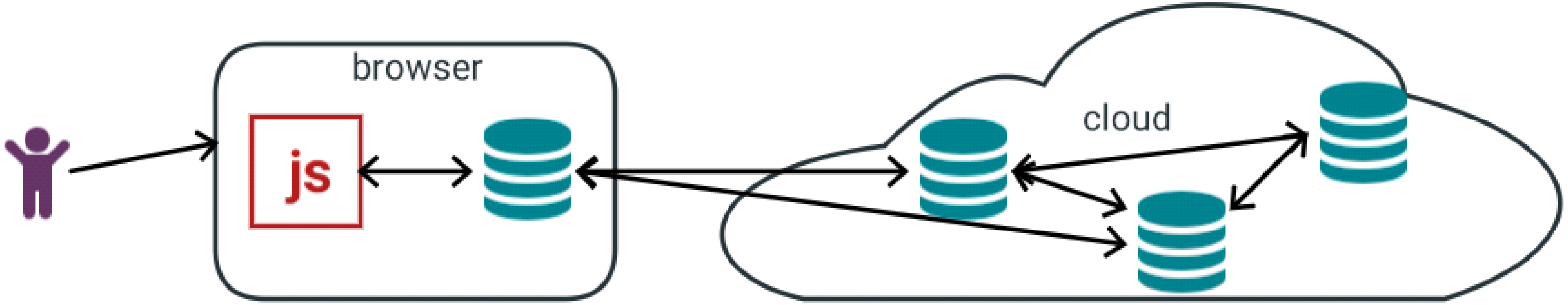


CAP Theorem for Distributed DBs



Offline First

Common to see CouchDB in Progressive Web Apps because it can replicate to PouchDB on the client side.



PHP and Document Databases

Document databases are well-supported in PHP:

- MongoDB needs an extension and a Composer library
- CouchDB and RethinkDB can use Composer libraries

Special Mention: ElasticSearch

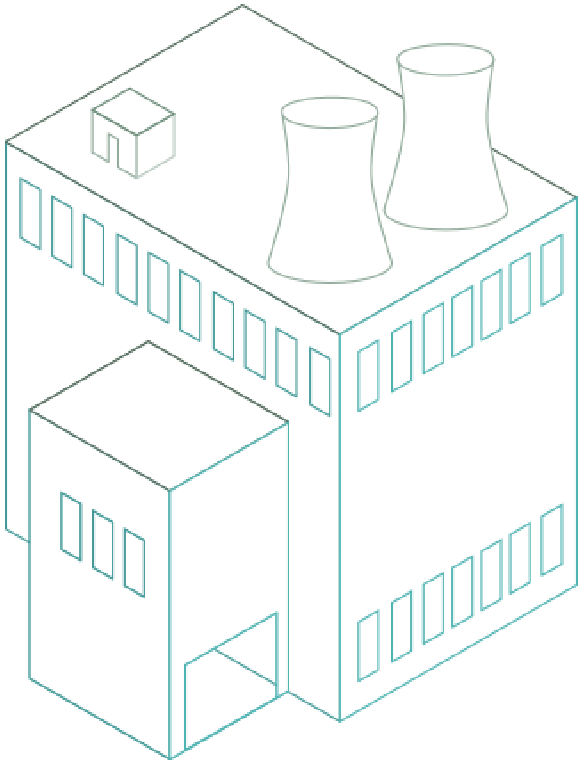
ElasticSearch is a Document Database

"You Know, for Search"

Duplicate data to it, use it for search

Data Warehouses

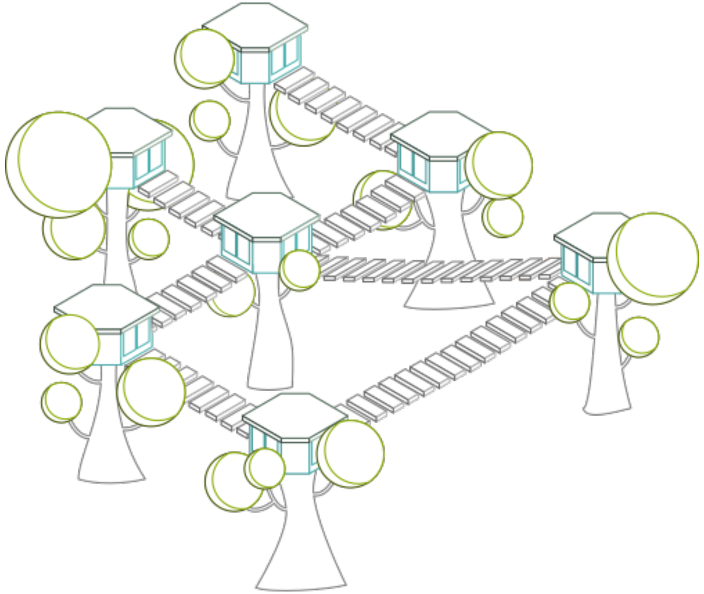
- As simple as a read-only database copy to report against.
- May use specific tech, e.g. Hadoop, Apache Spark
- Can serve as an archive to reduce load on the production system.



Graph Databases

Represent nodes and edges, with data attached.

For example: recommendations, actual route planning



Think of your data as nodes and edges, with properties. What questions will you need to answer?

Redis

In-memory key/value store, with an excellent grasp of data types.

For example: sessions, tracking the most-viewed article today, caching (especially calculated) stuff

Redis cluster is available for larger use cases

For Redis, performance and persistence are inversely correlated.

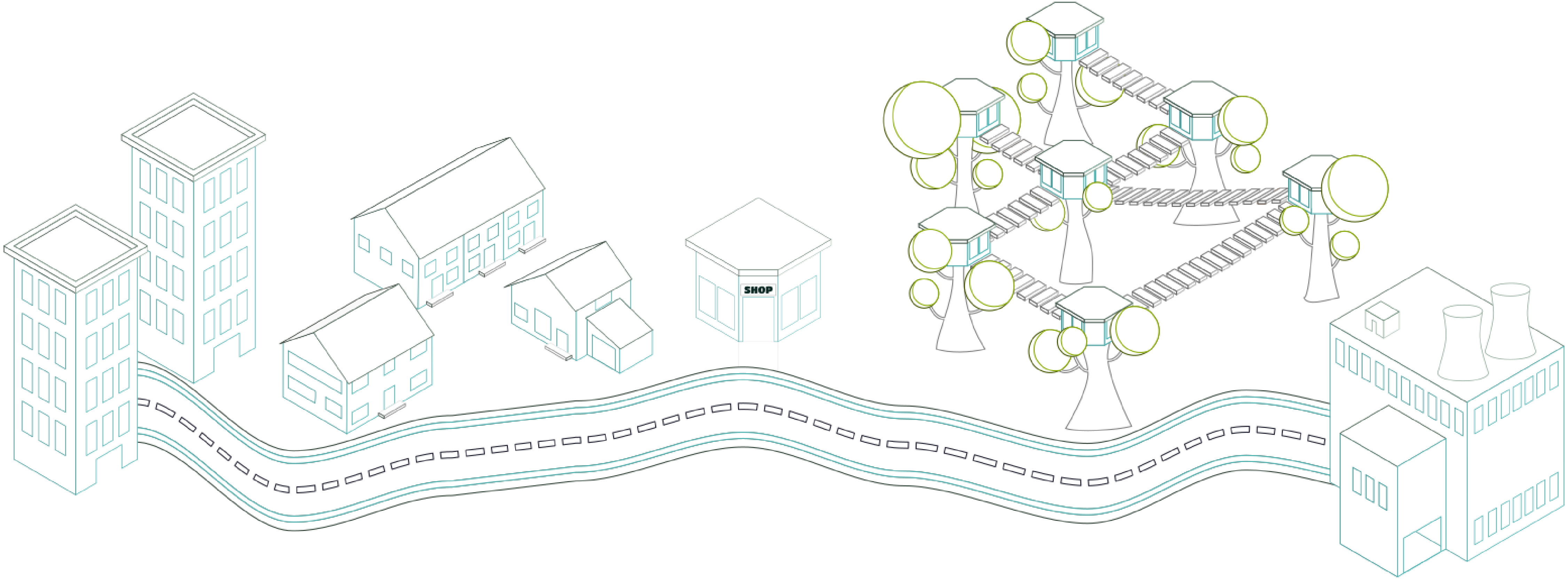
Redis Data Types

Redis supports (these and more):

- strings and numbers
- lists
- hashes
- sets and sorted sets

Also: simple Pub/Sub

Database Country



TL;DR Use PostgreSQL with Redis

Resources

https://www.ibm.com/cloud/data-management

https://en.wikipedia.org/wiki/CAP_theorem

http://lornajane.net

"7 Databases in 7 Weeks" Eric Redmond and Jim R Wilson

https://insights.stackoverflow.com/survey/2018/#technology-most-loved-dreaded-and-wanted-databases