

# Connecting to Databases

Norbert Hartl  
2denker



Not so fast!



Mostly  
Immense  
Schema  
Migrations  
And  
Terrible  
Coding  
Habits



Good  
Object  
Oriented  
Design  
and  
Fine  
Iterative  
Techniques



# Why a database?

Everything that can store data is a database



# Why a database?

Everything that can store data is a database

- ✦ persist



# Why a database?

Everything that can store data is a database

- ✦ persist
- ✦ share



# Why a database?

Everything that can store data is a database

- ✦ persist
- ✦ share
- ✦ data does not fit in memory



# Why a database?

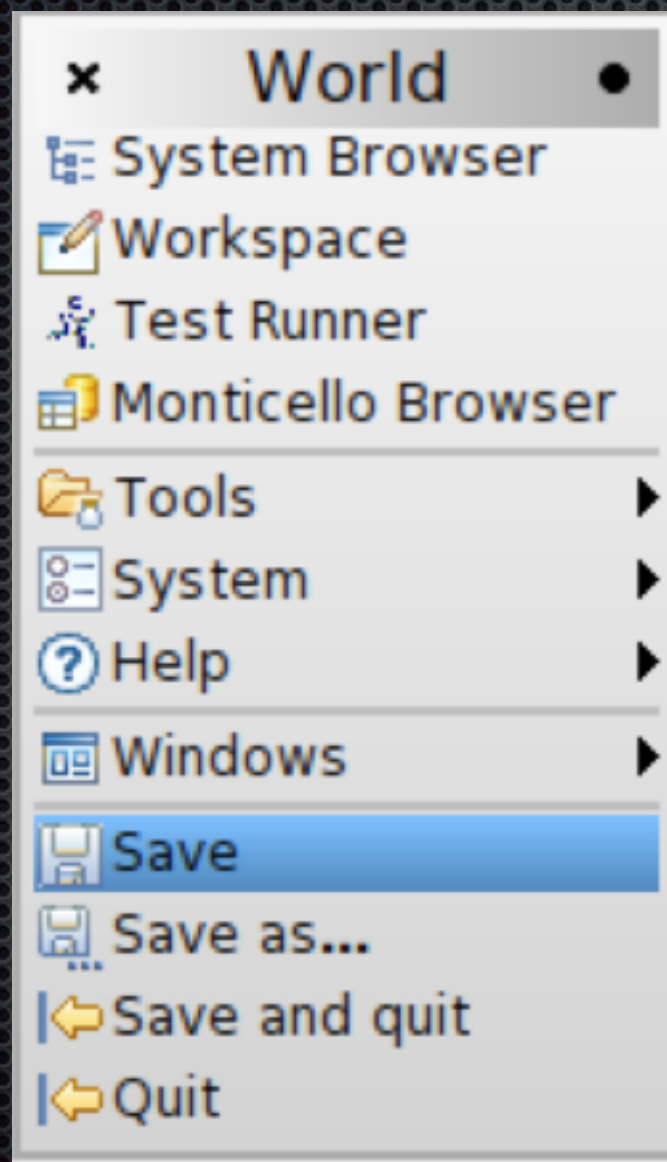
Everything that can store data is a database

- ✧ persist
- ✧ share
- ✧ data does not fit in memory
- ✧ concurrent access



# Pharo database

if you just need it persisted



Smalltalk image  
snapshot: **true**  
andQuit: **false**



# Persist outside of image

- ✦ image might break
- ✦ move data between images
- ✦ share data



# Fuel

if you want to store data outside the image

- ✦ persists whole graph
- ✦ is quite fast

FLSerializer

serialize: `myModel`

toFileNamed: `,model.fl'`



# PUnQlite

if there is more data than memory

- ✦ just a library, no server
- ✦ is a key-value store
- ✦ supports transactions

```
(PqDatabase open: ,model.db')  
  at: ,foo' put: ,bar';  
  close
```



# Voyage/Mongo

- ✦ Serializes objects to JSON
- ✦ can be tuned using magritte descriptions

```
(VOMongoRepository database: 'foo') enableSingleton
```

```
MyClass new property: ,value'; save.
```

```
MyClass selectMany: [ :each | each property = ,value' ]
```



# Plenty of options

- ✦ Phriak
- ✦ Voyage2 with new backends (Riak, anyone? )
- ✦ SQLite, SandstoneDB, ....



# RDBMS

Guille?