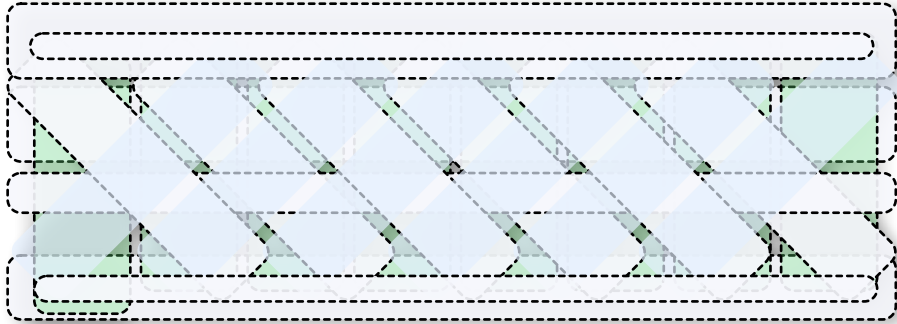# The Object Hotel

nothing special, just a place to store things

get a room!
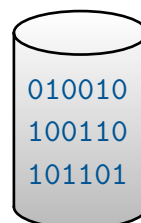
CONSEQUENCE HOTEL

```
{ "task": "AirportStatistics"
, "date-range": "2015-01"
, "^epsilon": 0.3
}
```

info

0x3b78c6ff2e0a002bcdfex746ab0a

010010
100110
101101

The concept

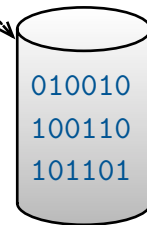nothing special, just a place to store things

you have some
"document",
the result of
a computation,
maybe.

info

*image, table,
graph, GIS, ...*

it's stored in a file
using bytes,
made of bits,
this is called **data**

this file has a <u>file type</u>
or <u>MIME-type</u>,
declaring its
format, how to
interpret it

we label the document,
with an information card,
this is called **meta-data**

```
010010
100110
101101
```

`.csv .tsv .geojson`
`.yaml .png .xml`

```
{ "task": "AirportStatistics"
, "date-range": "2015-01"
, "^epsilon": 0.3
}
```

the document is given an address,
the *hash* of it's contents

the file is uploaded
to the address

`0x3b78c6ff2e0a002bcdfex746ab0a`

```
010010
100110
101101
```

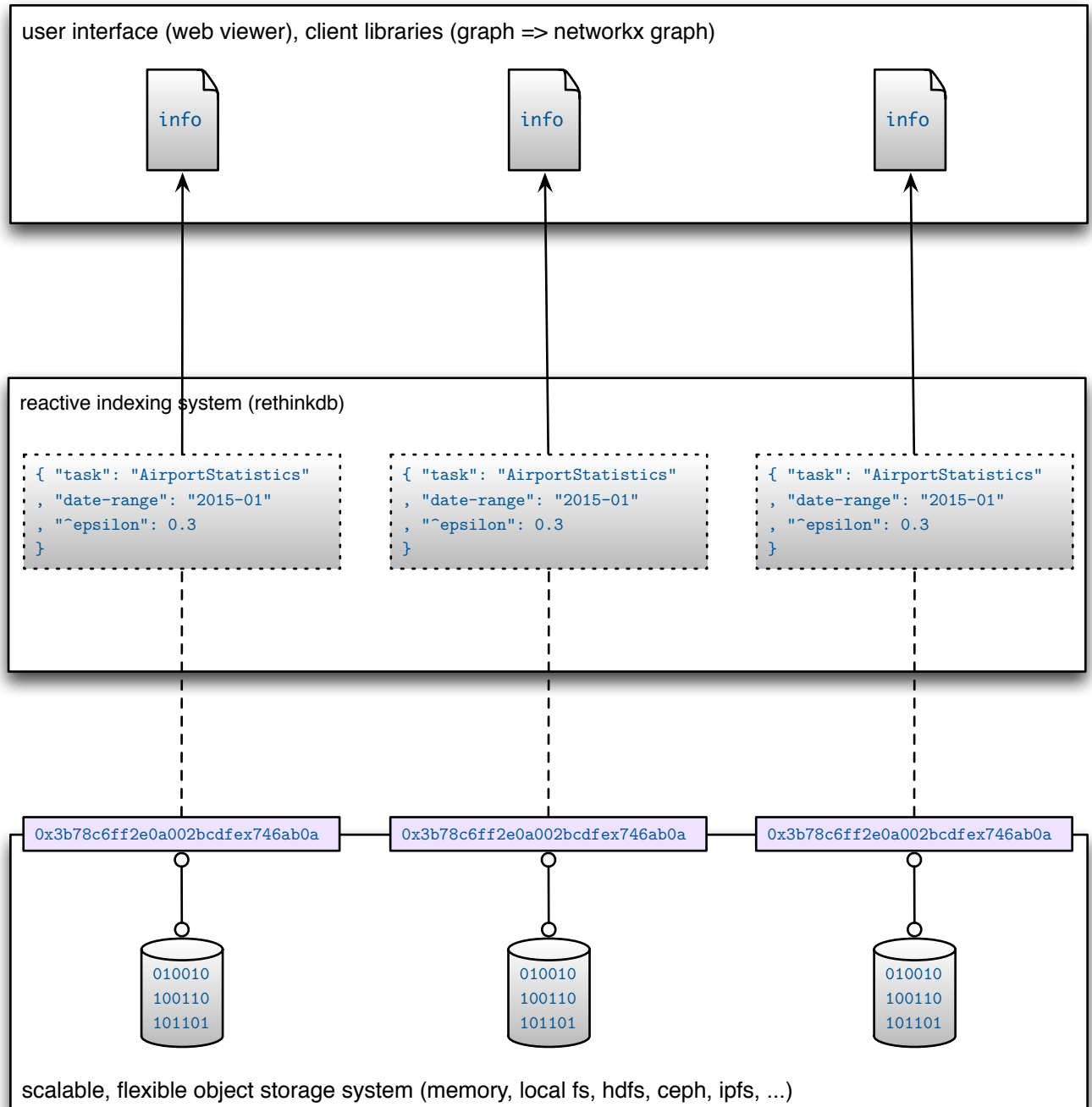*equivalent labels get the same address,*

conversely,

each different file needs a different label
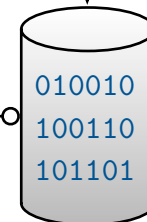
you can freely decide how to label and
adorn each file. you can use this to find it again

The result

a library of things, fruits of labour, rooms with information

user interface (web viewer), client libraries (graph => networkx graph)

info

info

info

reactive indexing system (rethinkdb)

```
{ "task": "AirportStatistics"
, "date-range": "2015-01"
, "^epsilon": 0.3
}
```

```
{ "task": "AirportStatistics"
, "date-range": "2015-01"
, "^epsilon": 0.3
}
```

```
{ "task": "AirportStatistics"
, "date-range": "2015-01"
, "^epsilon": 0.3
}
```

0x3b78c6ff2e0a002bcdfex746ab0a

0x3b78c6ff2e0a002bcdfex746ab0a

0x3b78c6ff2e0a002bcdfex746ab0a

```
010010
100110
101101
```

```
010010
100110
101101
```

```
010010
100110
101101
```

scalable, flexible object storage system (memory, local fs, hdfs, ceph, ipfs, ...)
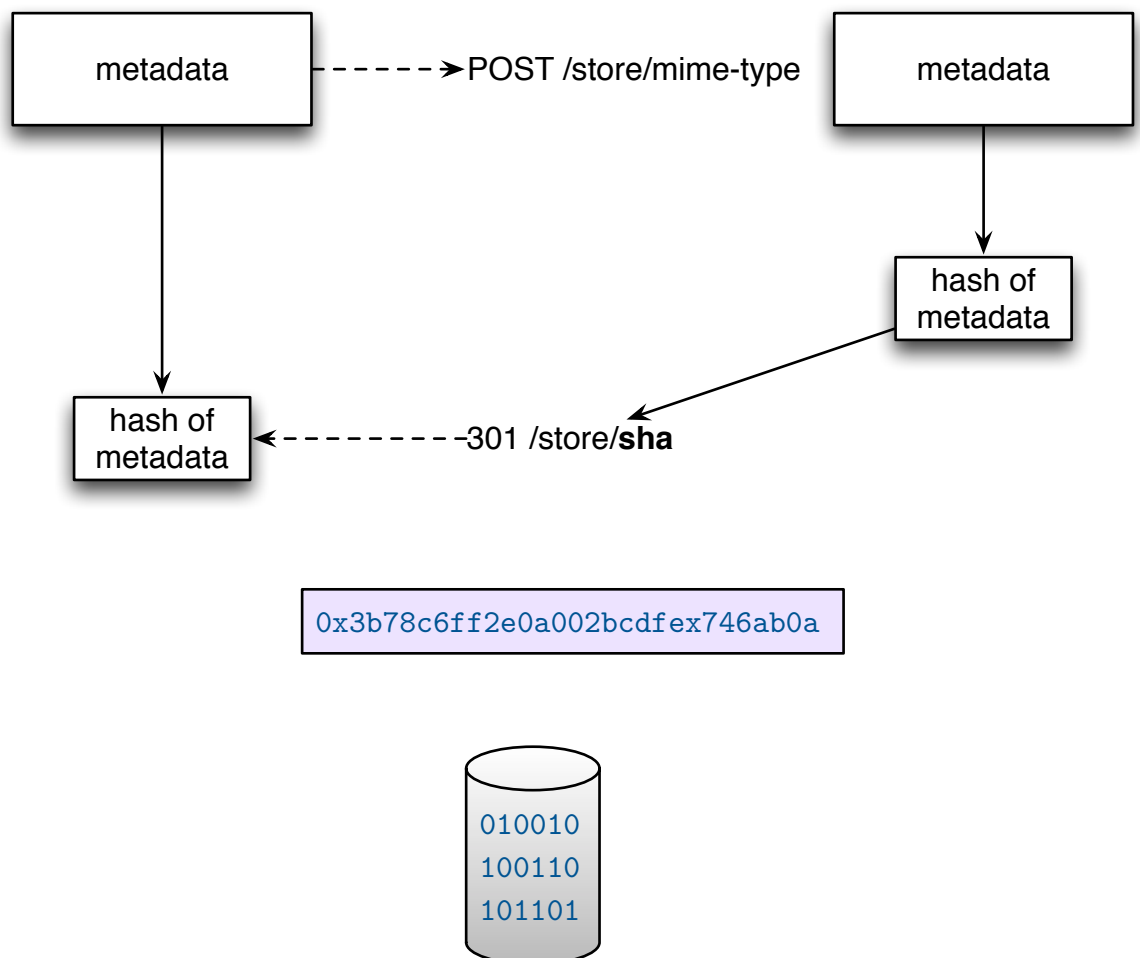
Requesting a room

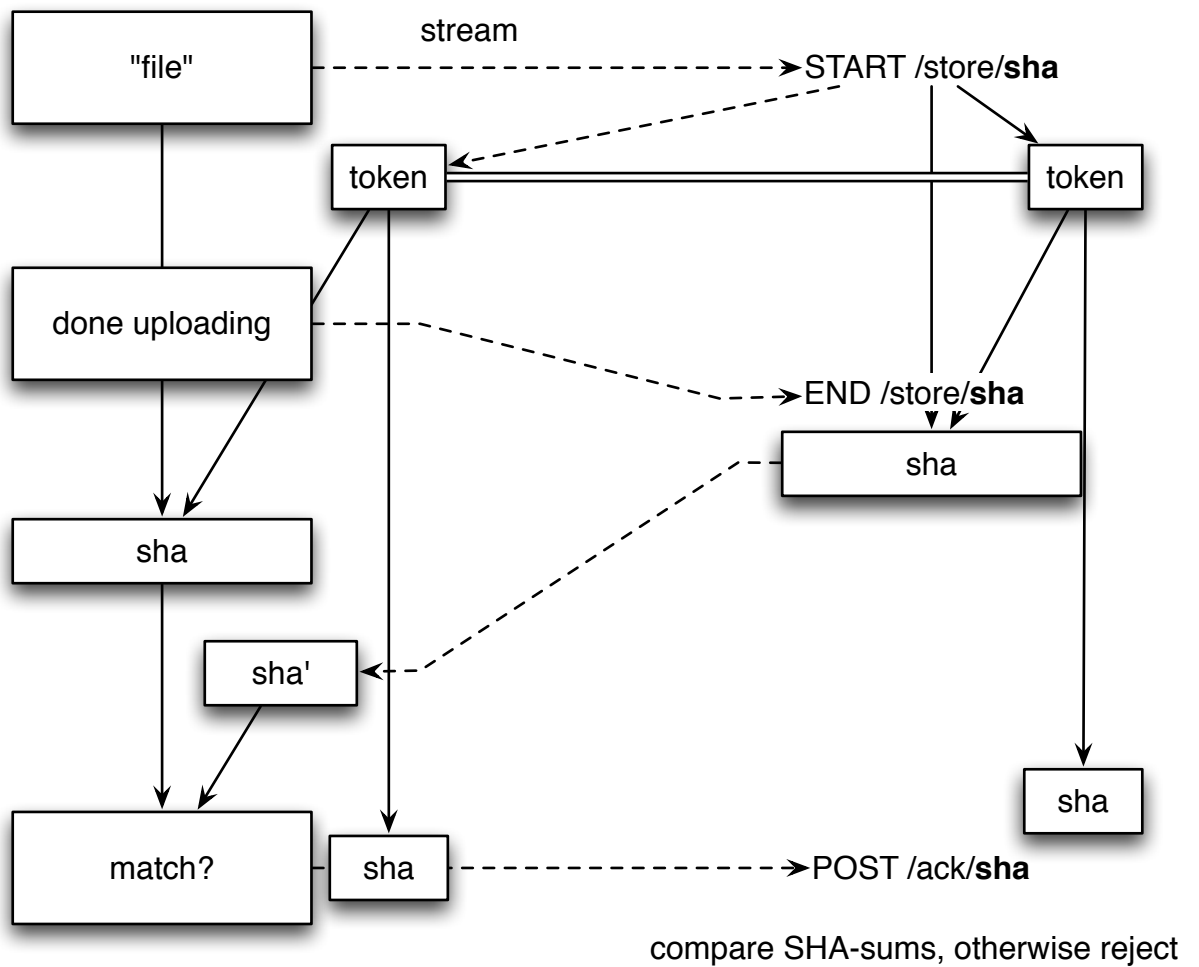before you can put a file into a room, you need to request it.

this is simple, a meta-data dictionary is passed as a room-coordinate

```
{ "task": "AirportStatistics"
, "date-range": "2015-01"
, "^epsilon": 0.3
}
```

| metadata | - - - - - - ➤ POST /store/mime-type | metadata |

metadata → hash of metadata

hash of metadata

hash of metadata ← - - - - - - - 301 /store/**sha** ← hash of metadata

0x3b78c6ff2e0a002bcdfex746ab0a

```
010010
100110
101101
```

Streaming file upload

A simple protocol ensures that files are uploaded complete and intact

stream

| "file" | START /store/**sha** |

| token | token |

| done uploading | END /store/**sha** |

| sha | sha |

| sha' |

| sha |

| match? | sha | POST /ack/**sha** |

compare SHA-sums, otherwise reject

once transmission is acknowledged, the file is locked and the meta data updated