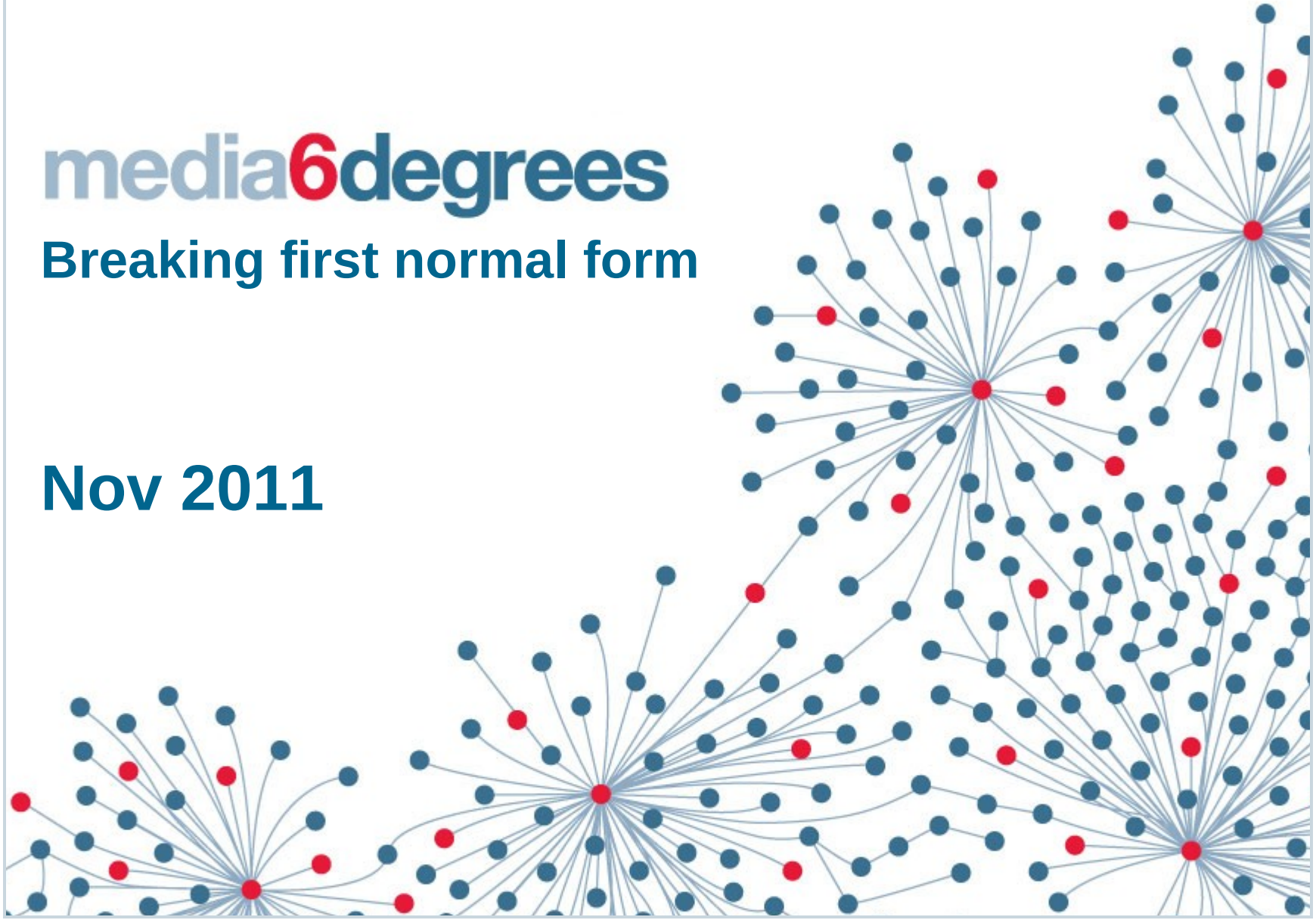
media6degrees

Breaking first normal form

Nov 2011



1st Normal Form

In a nutshell, to be 1NF do not:

- userid firstname, lastname , nicknames
- 1,ed,capriolo,killer;spike;iceman
- (these are not my nicknames btw)

- Why do this in typically relational databases?
  - Relational designs wants this in a one-to-many or many/many relationship across 2 or three tables
  - Easier to build ordered indexes
  - Most relational databases are design for narrow columns

But Hive ain't your grandmas datastore

- Normalizing and 3nf is in not usually a good idea in map reduce and hive
- Joins can be done map side and reduce side
- There is now index support way to go sichi et all
- **But STIL,L hive is not your grandmothers datastore, if you design thinking about joins and indexes your probably not modeling the design correctly**

A quick note about m6d (media6degrees)

Online advertising

A prospect engine for brands

We have to use cookies in many places

Cookies have limited size

Cookies need to have binary values encoded

Hacking data to make it smaller

LastSeen: long (64 bits)

Segment: int (32 bits)

Literal ','

Segment: int (32 bits)

Zipcode (32bits)

1 chose a relevant epoc and use byte

Use a byte for # of segments

Use a 4 byte radix encoded number

... and so on

Nice its a smaller cookie by now it looks like: abe34zfjtowsafsgsg34

So parsing this value could be pita

We could make upstream log hive friendly

But I never go upstream, i work in my box

Solution 1: Lot's o UDFs: Rejected

Write N UDFS for each object like:

getLastSeenForCookie(String)

getZipcodeForCookie(String)

...

But this would have made a huge toolkit

Solution 2: Structs

Hive has a struct like a c struct

Struct is list of name value pair

Structs can contain other structs!!!!

This gives us the serious ability to do object mapping!!!

UDFs can return structs!!!

Solution in action

Add jar myjar.jar;

Create temporary function parseCookie as 'com.md6.ParseCookieIntoStruct' ;

Select parseCookie(encodedColumn).lastSeen from my data;

Sweet! now we have access to scalar members in side encoded object with hive

Hive lateral view and explode

In my mind the coolest feature since dynamic partitions

Lateral view and explode allows us to convert an embedded list into rows.

This is very powerful for our nested objects

Sample query

```
SELECT
client_id,entry.spendcreativeid
FROM datatable
LATERAL VIEW explode
    (AdHistoryAsStruct(ad_history).adEntrylist) entryList as
    entry
where hit_date=20110321 AND mid=001406;
```

```
3214498023360851706     215286
3214498023360851706     195785
3214498023360851706     128640
```

Questions?