Inheritance and Core Data Objects

Core Data supports the concept of object inheritance

Entities inherit properties, relationships and fetched properties from their superentity.

Useful technique for representing slightly different object types

Base entity `Item`, child properties `File` and `Folder`.

Major caveat: all these objects will end up in one huge table.

Both `File` and `Folder` will share one table and they will all have columns from one another.

This might greatly affect your performance when working with huge data sets

Underlying database queries will also search through other entities entries

Core Data and String Searches

[cd]

Case insensitive search `c`

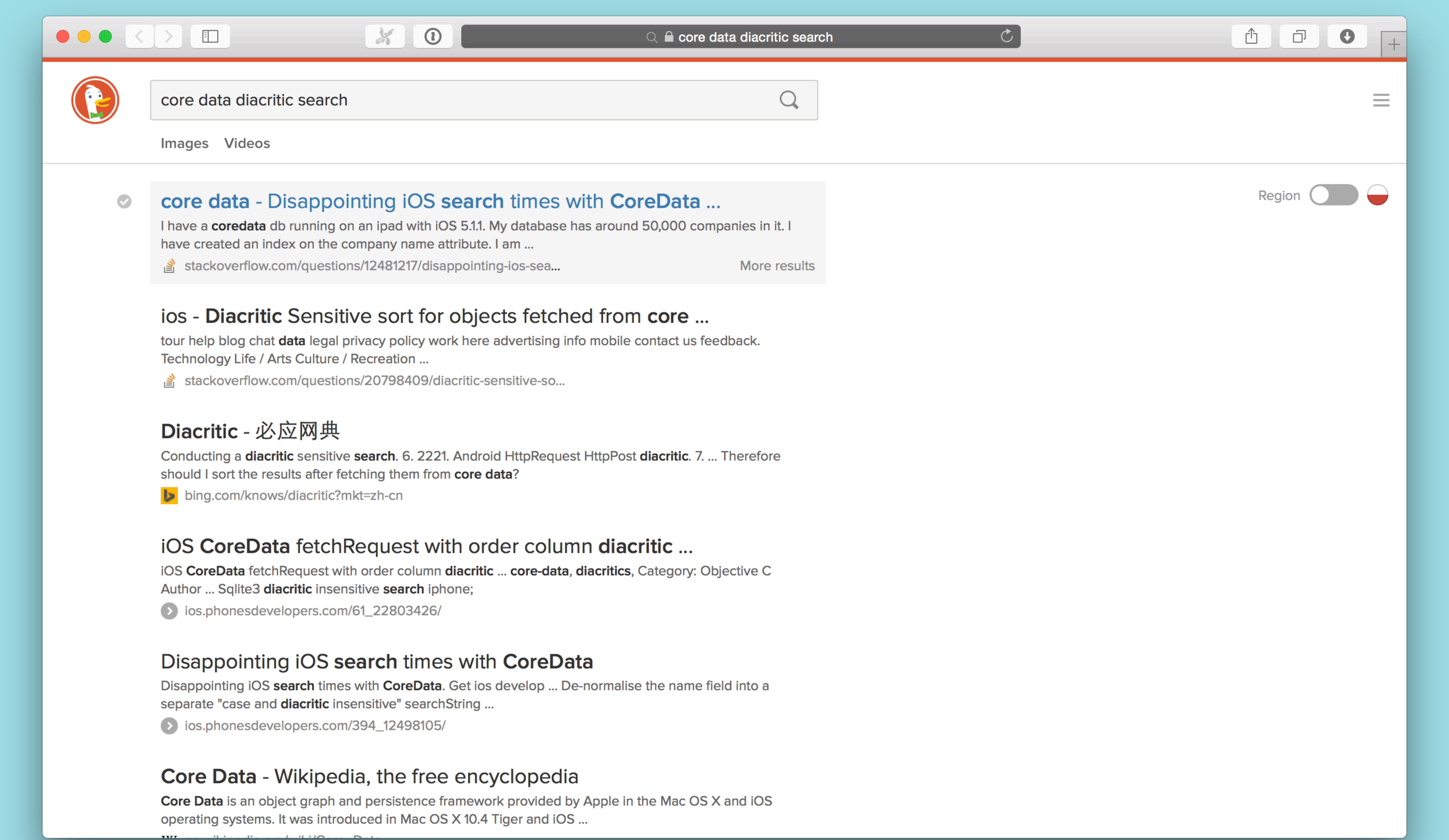
```
name = Pawel

name CONTAINS pawel // false
name CONTAINS[c] pawel // true
```

Diacritic search `d`

```
name = Paweł

name CONTAINS Pawel // false
name CONTAINS[d] Pawel // true
```



core data diacritic search

core data - Disappointing iOS search times with CoreData ...
I have a **coredata** db running on an ipad with iOS 5.1.1. My database has around 50,000 companies in it. I have created an index on the company name attribute. I am ...
stackoverflow.com/questions/12481217/disappointing-ios-sea... More results

ios - **Diacritic** Sensitive sort for objects fetched from **core** ...
tour help blog chat **data** legal privacy policy work here advertising info mobile contact us feedback. Technology Life / Arts Culture / Recreation ...
stackoverflow.com/questions/20798409/diacritic-sensitive-so...

Diacritic - 必应网典
Conducting a **diacritic** sensitive **search**. 6. 2221. Android HttpRequest HttpPost **diacritic**. 7. ... Therefore should I sort the results after fetching them from **core data**?
bing.com/knows/diacritic?mkt=zh-cn

iOS **CoreData** fetchRequest with order column **diacritic** ...
iOS **CoreData** fetchRequest with order column **diacritic** ... **core-data**, **diacritics**, Category: Objective C Author ... Sqlite3 **diacritic** insensitive **search** iphone;
ios.phonesdevelopers.com/61_22803426/

Disappointing iOS **search** times with **CoreData**
Disappointing iOS **search** times with **CoreData**. Get ios develop ... De-normalise the name field into a separate "case and **diacritic** insensitive" searchString ...
ios.phonesdevelopers.com/394_12498105/

Core Data - Wikipedia, the free encyclopedia
Core Data is an object graph and persistence framework provided by Apple in the Mac OS X and iOS operating systems. It was introduced in Mac OS X 10.4 Tiger and iOS ...

The reason?

`NSCoreDataStringSearch`

`NSCoreDataStringSearch` loads all the data *into memory*.

Core Data will load all objects from given table into memory and perform the search there.

This will always be a major performance hit. Your app will have to load all the data from given table into memory and parse it there. This is bad for both CPU and memory and disk usage.

Moreover `CONTAINS` is superbly slow for string searches

Solution

Let's be clever

We don't really need a `CONTAINS` search

We can just use

```
firstName >= "pawel" AND firstName < "pawem"
```

Assignment I

Normalizing data before saving

Goals

- Create new `normalizedFirstName` and `normalizedLastName` properties for `Employee`
- Use `willSave` method to create a normalized version of searched properties
- Refactor search fetch to leverage new normalized first and last name
- Refactor search to use the `>=/<` pattern instead of `CONTAINS`

But I really want a full text search…

Building blazing fast search with Core Data

External SQLite database

FTS enabled

Use `NSManagedObjectContext` notifications to determine when save occured

Update your SQLite database accordingly

Alternatively you can use `NSManagedObject` callbacks

Or any custom solution that knows when data gets inserted/deleted/updated

Performance improvements?

80.000 objects (Employees)

Down from 14 seconds to 0.4 second

Thank you very much for attending!

And have a great second day of the conference!

Got questions?

@eldudi
hello@dudek.mobi
github.com/paweldudek