Introduction to Core Data

Daniel Tull

Core Data is...

...a way to store an object graph

...fine-tuned for memory and performance

...less code than SQLite

...better database code than I could write (hopefully)

Core Data Stack

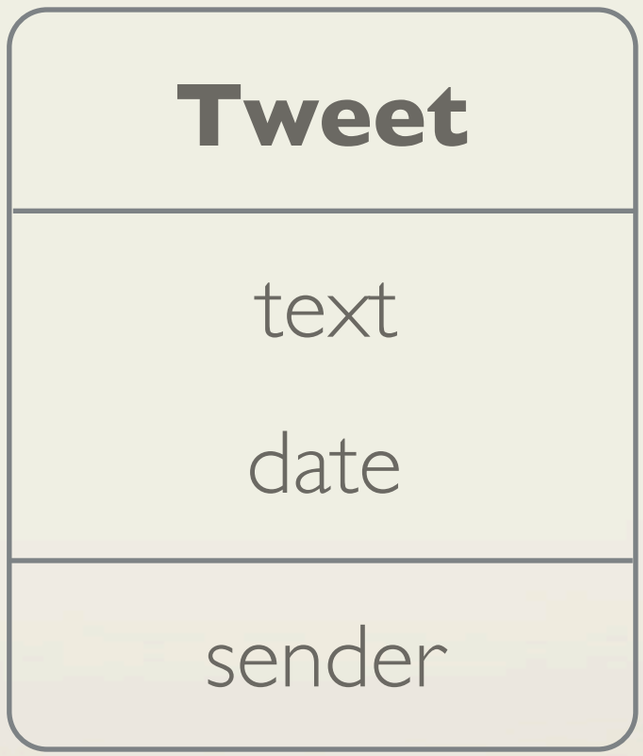
NSManagedObjectModel

Managed Object Model

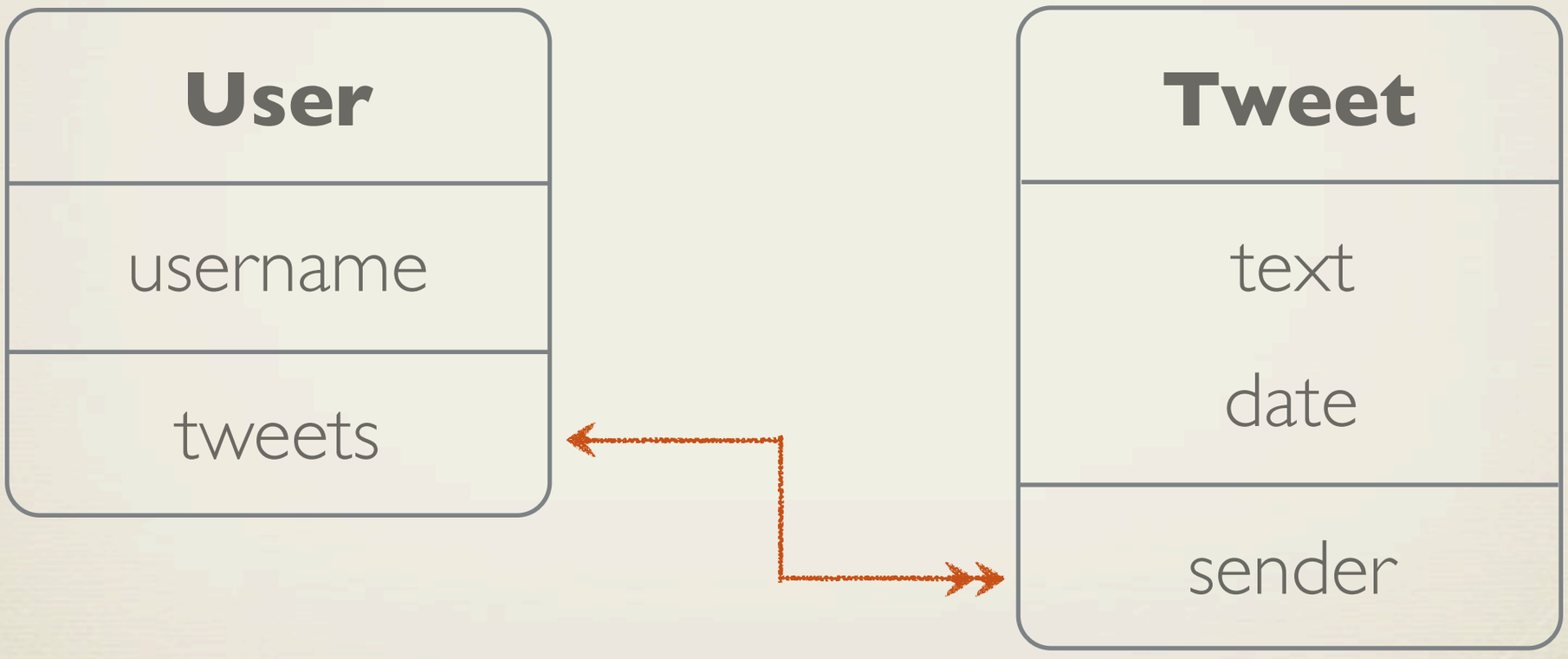
Entities are like Core Data's idea of a class

Core Data objects are referred to as managed objects

Managed Object Model

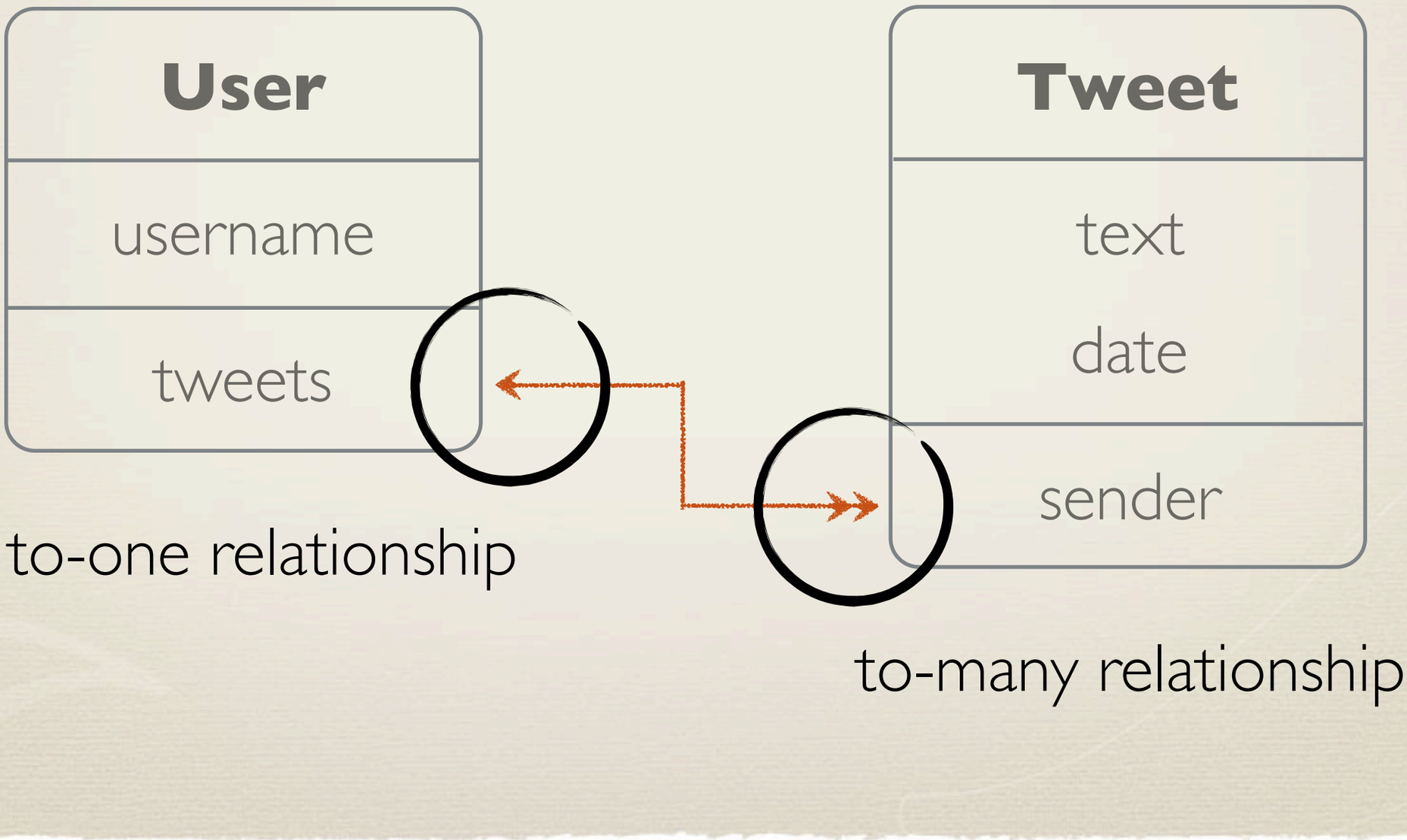


Managed Object Model



Relationships need inverses for performance reasons

Managed Object Model



TwitterTest – TwitterTest.xcdatamodel

Xcode

T.. 〉 iPhone 4.3...

TwitterTest 〉 TwitterTest 〉 TwitterTest.xcda... 〉 TwitterTest.xcdatamodel 〉 No Selection

ENTITIES

FETCH REQUESTS

CONFIGURATIONS

Attributes

Attribute	Type

Relationships

Relationship	Destination	Inverse

Fetched Properties

Fetched Property	Predicate

Identity and Type

File Name: TwitterTest.xcdatamodel

File Type: Default – Core Data Model

Location: Relative to Group

TwitterTest.xcdatamodel

Full Path: /Users/danielctull/Desktop/TwitterTest/TwitterTest/TwitterTest.xcdatamodeld/TwitterTest.xcdatamodel

Core Data Model

Identifier: Model Version Identifier

Tools Version

Minimum: Automatic (Xcode 4.1)

Target Membership

TwitterTest

Outline Style

Add Entity

Add Attribute

Editor Style

TwitterTest – TwitterTest.xcdatamodel

Xcode

T.. 〉 iPhone 4.3...

TwitterTest 〉 TwitterTest 〉 TwitterTest.xcda... 〉 TwitterTest.xcdatamodel 〉 No Selection

ENTITIES

FETCH REQUESTS

CONFIGURATIONS

Outline Style

Add Entity

Add Attribute

Editor Style

Identity and Type

File Name: TwitterTest.xcdatamodel

File Type: Default – Core Data Model

Location: Relative to Group

TwitterTest.xcdatamodel

Full Path: /Users/danielctull/Desktop/TwitterTest/TwitterTest/TwitterTest.xcdatamodeld/TwitterTest.xcdatamodel

Core Data Model

Identifier: Model Version Identifier

Tools Version

Minimum: Automatic (Xcode 4.1)

Target Membership

TwitterTest

TwitterTest – TwitterTest.xcdatamodel

Xcode

T.. 〉 iPhone 4.3...

TwitterTest 〉 TwitterTest 〉 TwitterTest.xcdata... 〉 TwitterTest.xcdatamodel 〉 Entity

ENTITIES

Entity

FETCH REQUESTS

CONFIGURATIONS

Default

Entity
Attributes
Relationships

Identity and Type

File Name: TwitterTest.xcdatamodel

File Type: Default – Core Data Model

Location: Relative to Group

TwitterTest.xcdatamodel

Full Path: /Users/danielctull/Desktop/TwitterTest/TwitterTest/TwitterTest.xcdatamodeld/TwitterTest.xcdatamodel

Core Data Model

Identifier: Model Version Identifier

Tools Version

Minimum: Automatic (Xcode 4.1)

Target Membership

TwitterTest

Source Control

Version --

Status No changes

Outline Style | Add Entity | Add Attribute | Editor Style

TwitterTest – TwitterTest.xcdatamodel

Xcode

Project 2

T.. iPhone 4.3...

TwitterTest › TwitterT... › TwitterT... › TwitterTest.xcdatamodel › No Selection

ENTITIES

Tweet

FETCH REQUESTS

CONFIGURATIONS

Default

Tweet

Attributes

date

text

Relationships

Identity and Type

File Name TwitterTest.xcdatamodel

File Type Default – Core Data Model

Location Relative to Group

TwitterTest.xcdatamodel

Full Path /Users/danielctull/Desktop/TwitterTest/TwitterTest/TwitterTest.xcdatamodeld/TwitterTest.xcdatamodel

Core Data Model

Identifier Model Version Identifier

Tools Version

Minimum Automatic (Xcode 4.1)

Target Membership

TwitterTest

Source Control

Version --

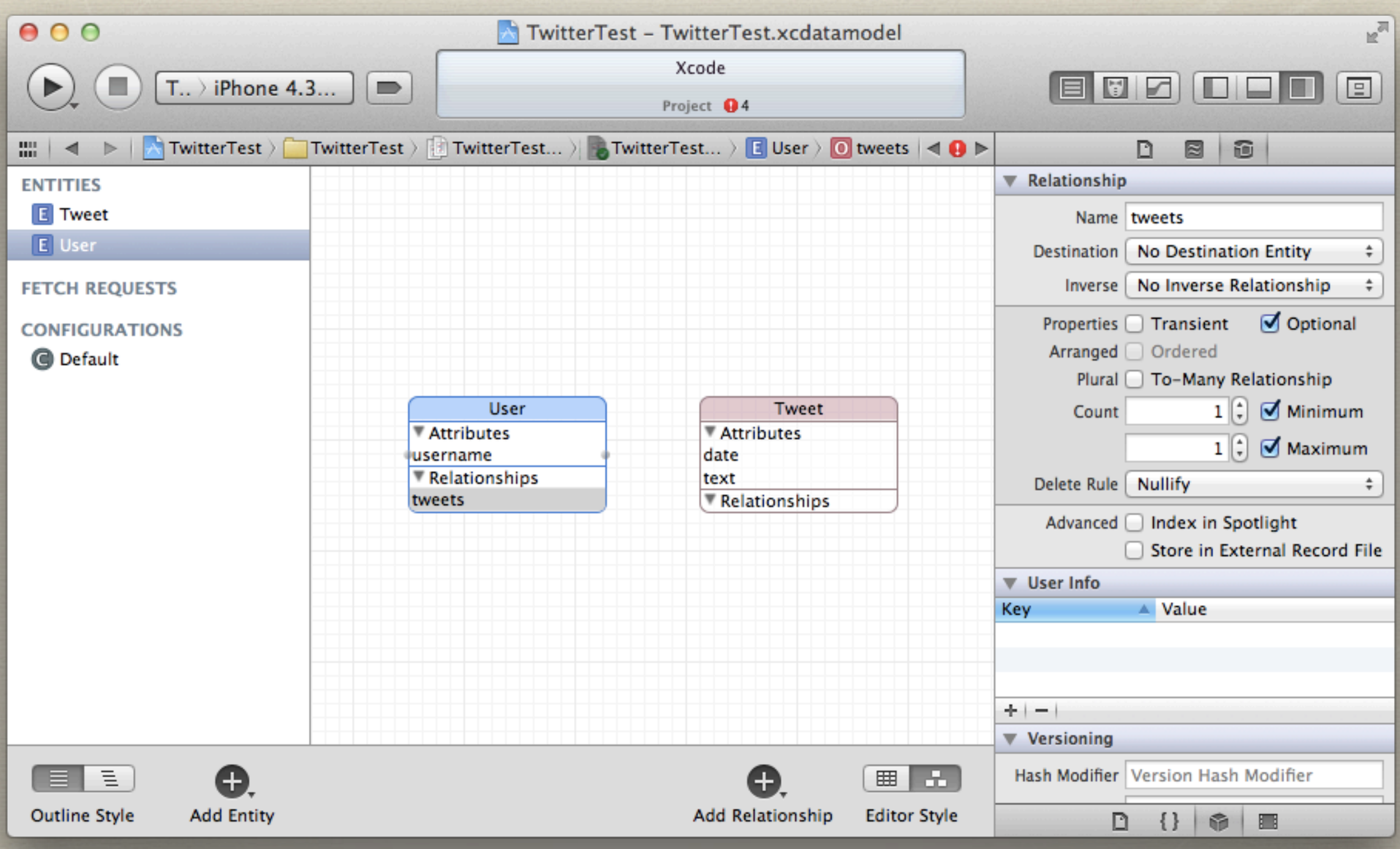
Status No changes

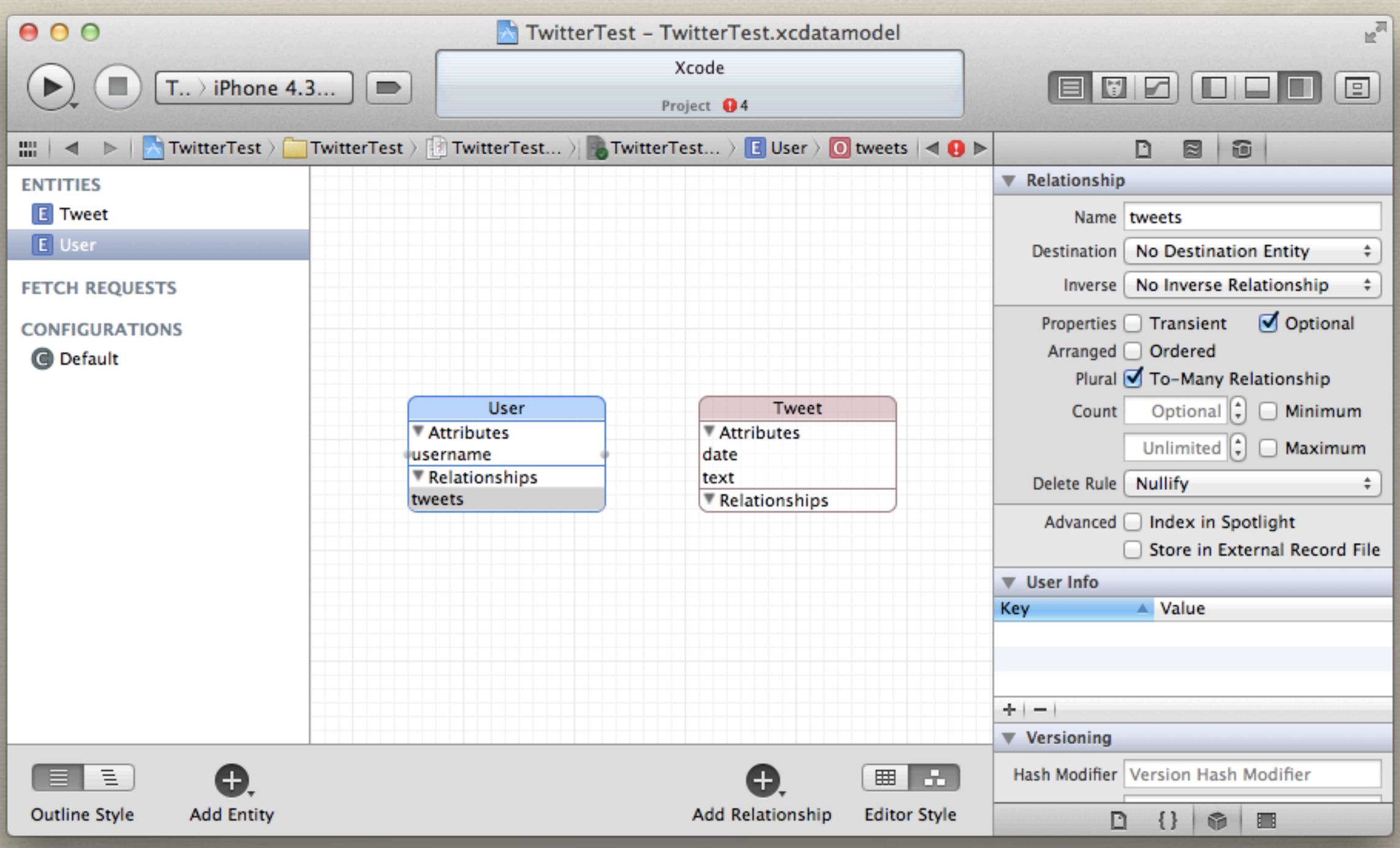
Outline Style

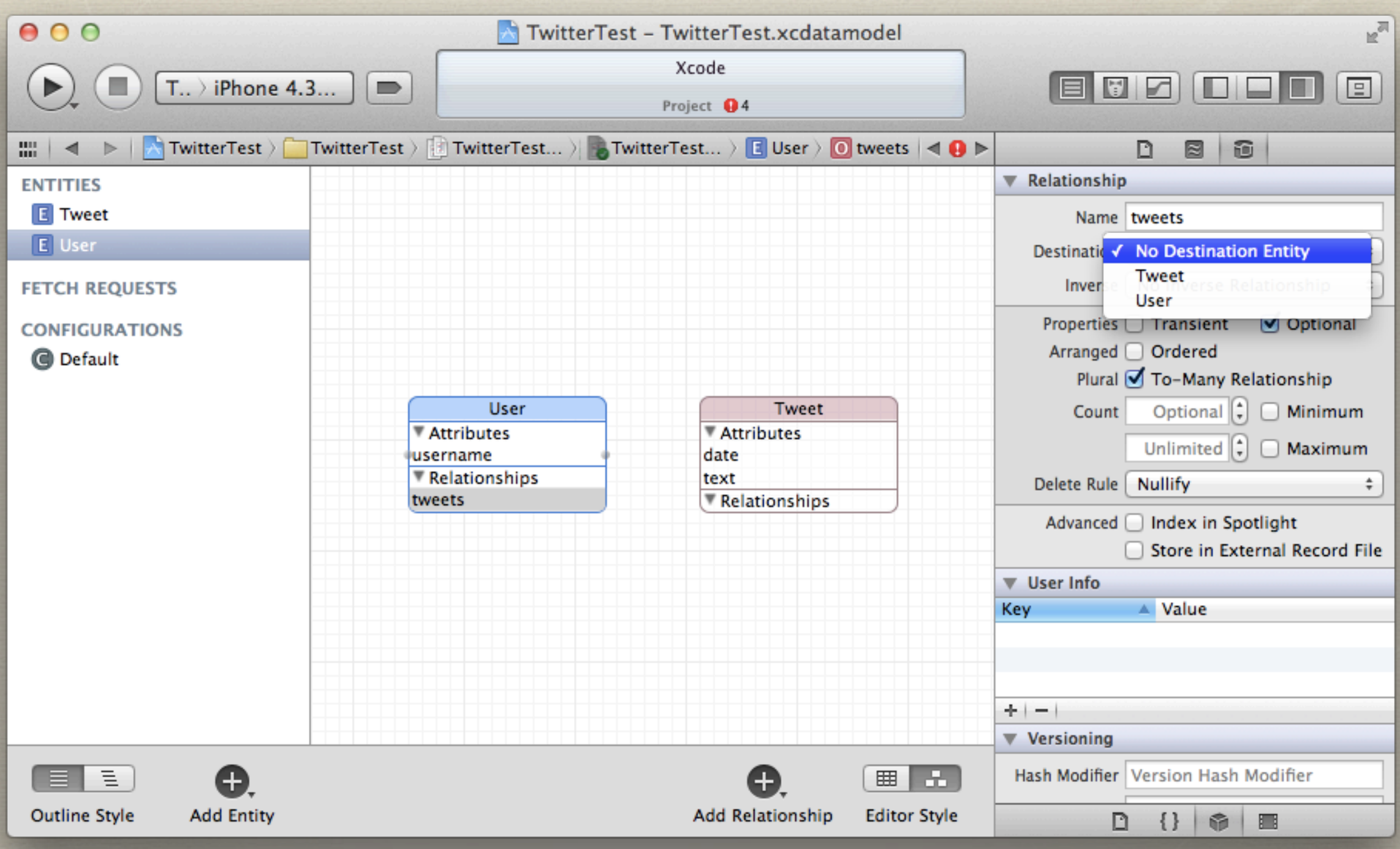
Add Entity

Add Attribute

Editor Style


TwitterTest – TwitterTest.xcdatamodel
Xcode
Project 4
T.. iPhone 4.3...
TwitterTest
TwitterTest
TwitterTest...
TwitterTest...
User
tweets
ENTITIES
Tweet
User
FETCH REQUESTS
CONFIGURATIONS
Default
User
Attributes
username
Relationships
tweets
Tweet
Attributes
date
text
Relationships
Relationship
Name tweets
Destination No Destination Entity
Inverse No Inverse Relationship
Properties Transient Optional
Arranged Ordered
Plural To-Many Relationship
Count 1 Minimum
1 Maximum
Delete Rule Nullify
Advanced Index in Spotlight
Store in External Record File
User Info
Key Value
Versioning
Hash Modifier Version Hash Modifier
Outline Style
Add Entity
Add Relationship
Editor Style


TwitterTest – TwitterTest.xcdatamodel
Xcode
Project 4
T.. iPhone 4.3...
TwitterTest
TwitterTest
TwitterTest...
TwitterTest...
User
tweets
ENTITIES
Tweet
User
FETCH REQUESTS
CONFIGURATIONS
Default
User
Attributes
username
Relationships
tweets
Tweet
Attributes
date
text
Relationships
Relationship
Name tweets
Destination No Destination Entity
Inverse No Inverse Relationship
Properties Transient Optional
Arranged Ordered
Plural To-Many Relationship
Count Optional Minimum
Unlimited Maximum
Delete Rule Nullify
Advanced Index in Spotlight
Store in External Record File
User Info
Key Value
Versioning
Hash Modifier Version Hash Modifier
Outline Style
Add Entity
Add Relationship
Editor Style


TwitterTest – TwitterTest.xcdatamodel
Xcode
Project 4
T.. iPhone 4.3...
TwitterTest
TwitterTest
TwitterTest...
TwitterTest...
User
tweets
ENTITIES
Tweet
User
FETCH REQUESTS
CONFIGURATIONS
Default
User
Attributes
username
Relationships
tweets
Tweet
Attributes
date
text
Relationships
Relationship
Name tweets
No Destination Entity
Tweet
User
Properties Transient Optional
Arranged Ordered
Plural To-Many Relationship
Count Optional Minimum
Unlimited Maximum
Delete Rule Nullify
Advanced Index in Spotlight
Store in External Record File
User Info
Key Value
Versioning
Hash Modifier Version Hash Modifier
Outline Style
Add Entity
Add Relationship
Editor Style

TwitterTest – TwitterTest.xcdatamodel

Xcode

Project 3

T.. iPhone 4.3...

TwitterTest TwitterTest TwitterTest... TwitterTest... User tweets

ENTITIES

Tweet

User

FETCH REQUESTS

CONFIGURATIONS

Default

User
Attributes
username
Relationships
tweets

Tweet
Attributes
date
text
Relationships

Relationship

Name tweets

Destination Tweet

Inverse No Inverse Relationship

Properties Transient Optional

Arranged Ordered

Plural To-Many Relationship

Count Optional Minimum

Unlimited Maximum

Delete Rule Nullify

Advanced Index in Spotlight

Store in External Record File

User Info

Key Value

Versioning

Hash Modifier Version Hash Modifier

Outline Style

Add Entity

Add Relationship

Editor Style

TwitterTest – TwitterTest.xcdatamodel

Xcode

Project 4

T.. iPhone 4.3...

TwitterTest TwitterTest TwitterTest... TwitterTest... Tweet user

ENTITIES

Tweet

User

FETCH REQUESTS

CONFIGURATIONS

Default

User
Attributes
username
Relationships
tweets

Tweet
Attributes
date
text
Relationships
user

Relationship

Name user

Destination No Destination Entity

Inverse No Inverse Relationship

Properties Transient Optional

Arranged Ordered

Plural To-Many Relationship

Count 1 Minimum

1 Maximum

Delete Rule Nullify

Advanced Index in Spotlight

Store in External Record File

User Info

Key Value

Versioning

Hash Modifier Version Hash Modifier

Outline Style

Add Entity

Add Relationship

Editor Style

TwitterTest – TwitterTest.xcdatamodel

Xcode

Project 4

T.. iPhone 4.3...

TwitterTest TwitterTest TwitterTest... TwitterTest... Tweet user

ENTITIES

Tweet

User

FETCH REQUESTS

CONFIGURATIONS

Default

User
Attributes
username
Relationships
tweets

Tweet
Attributes
date
text
Relationships
user

Relationship

Name user

Destinati No Destination Entity
Tweet
User

Inver

Properties Transient Optional

Arranged Ordered

Plural To-Many Relationship

Count 1 Minimum

1 Maximum

Delete Rule Nullify

Advanced Index in Spotlight

Store in External Record File

User Info

Key Value

Versioning

Hash Modifier Version Hash Modifier

Outline Style

Add Entity

Add Relationship

Editor Style

TwitterTest – TwitterTest.xcdatamodel

Xcode

Project 3

T.. iPhone 4.3...

TwitterTest TwitterTest TwitterTest... TwitterTest... Tweet user

ENTITIES

Tweet

User

FETCH REQUESTS

CONFIGURATIONS

Default

User
Attributes
username
Relationships
tweets

Tweet
Attributes
date
text
Relationships
user

Relationship

Name user

Destination User

Inverse No Inverse Relationship

Properties Transient Optional

Arranged Ordered

Plural To-Many Relationship

Count 1 Minimum

1 Maximum

Delete Rule Nullify

Advanced Index in Spotlight

Store in External Record File

User Info

Key Value

Versioning

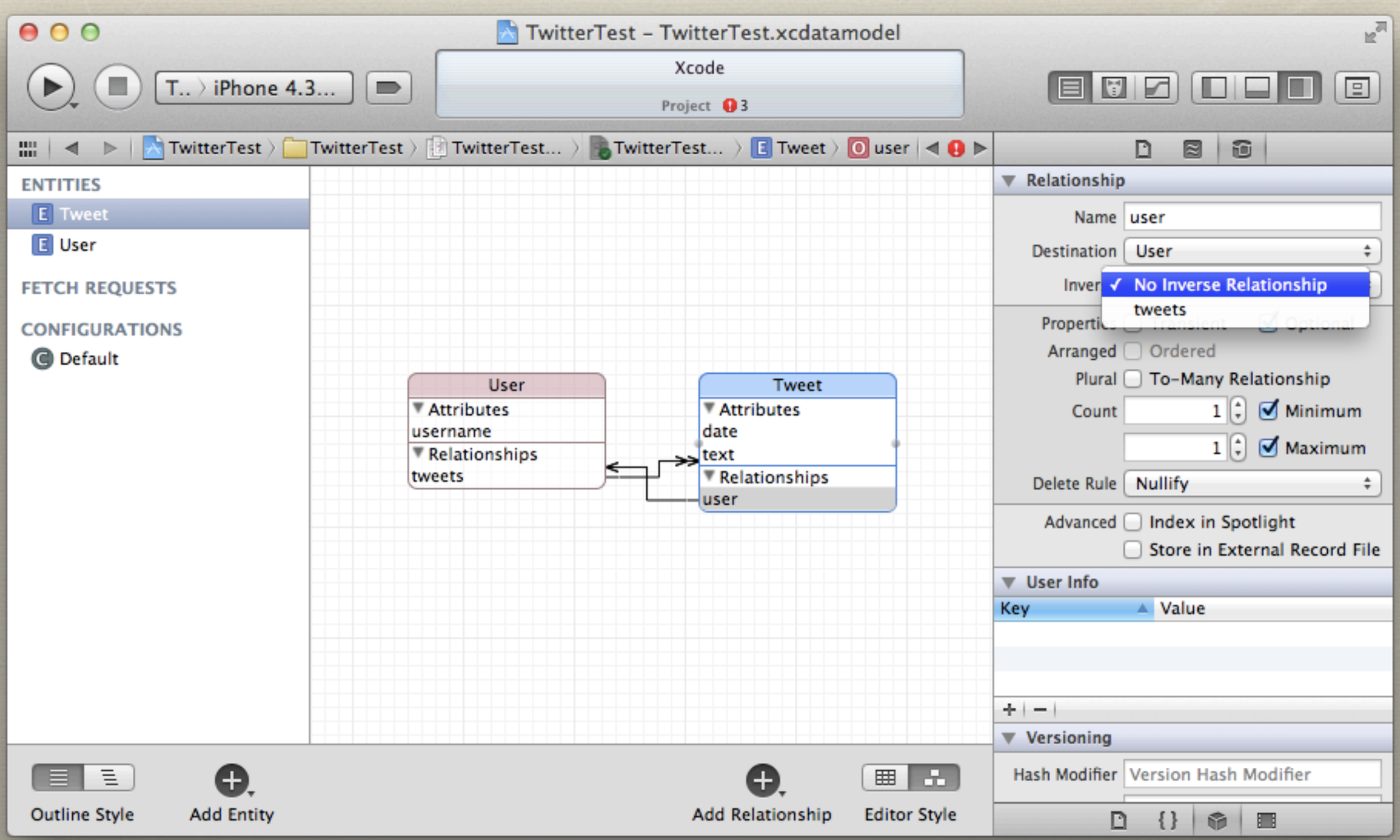
Hash Modifier Version Hash Modifier

Outline Style

Add Entity

Add Relationship

Editor Style


TwitterTest – TwitterTest.xcdatamodel
Xcode
Project 3
T.. iPhone 4.3...
TwitterTest
TwitterTest
TwitterTest...
TwitterTest...
Tweet
user
ENTITIES
Tweet
User
FETCH REQUESTS
CONFIGURATIONS
Default
User
Attributes
username
Relationships
tweets
Tweet
Attributes
date
text
Relationships
user
Relationship
Name user
Destination User
No Inverse Relationship
tweets
Arranged Ordered
Plural To-Many Relationship
Count 1 Minimum
1 Maximum
Delete Rule Nullify
Advanced Index in Spotlight
Store in External Record File
User Info
Key
Value
Versioning
Hash Modifier Version Hash Modifier
Outline Style
Add Entity
Add Relationship
Editor Style

TwitterTest – TwitterTest.xcdatamodel

Xcode

Project 3

T.. iPhone 4.3...

TwitterTest TwitterTest TwitterTest... TwitterTest... Tweet user

ENTITIES

Tweet

User

FETCH REQUESTS

CONFIGURATIONS

Default

User
Attributes
username
Relationships
tweets

Tweet
Attributes
date
text
Relationships
user

Relationship

Name user

Destination User

Inverse tweets

Properties Transient Optional

Arranged Ordered

Plural To-Many Relationship

Count 1 Minimum

1 Maximum

Delete Rule Nullify

Advanced Index in Spotlight

Store in External Record File

User Info

Key Value

Versioning

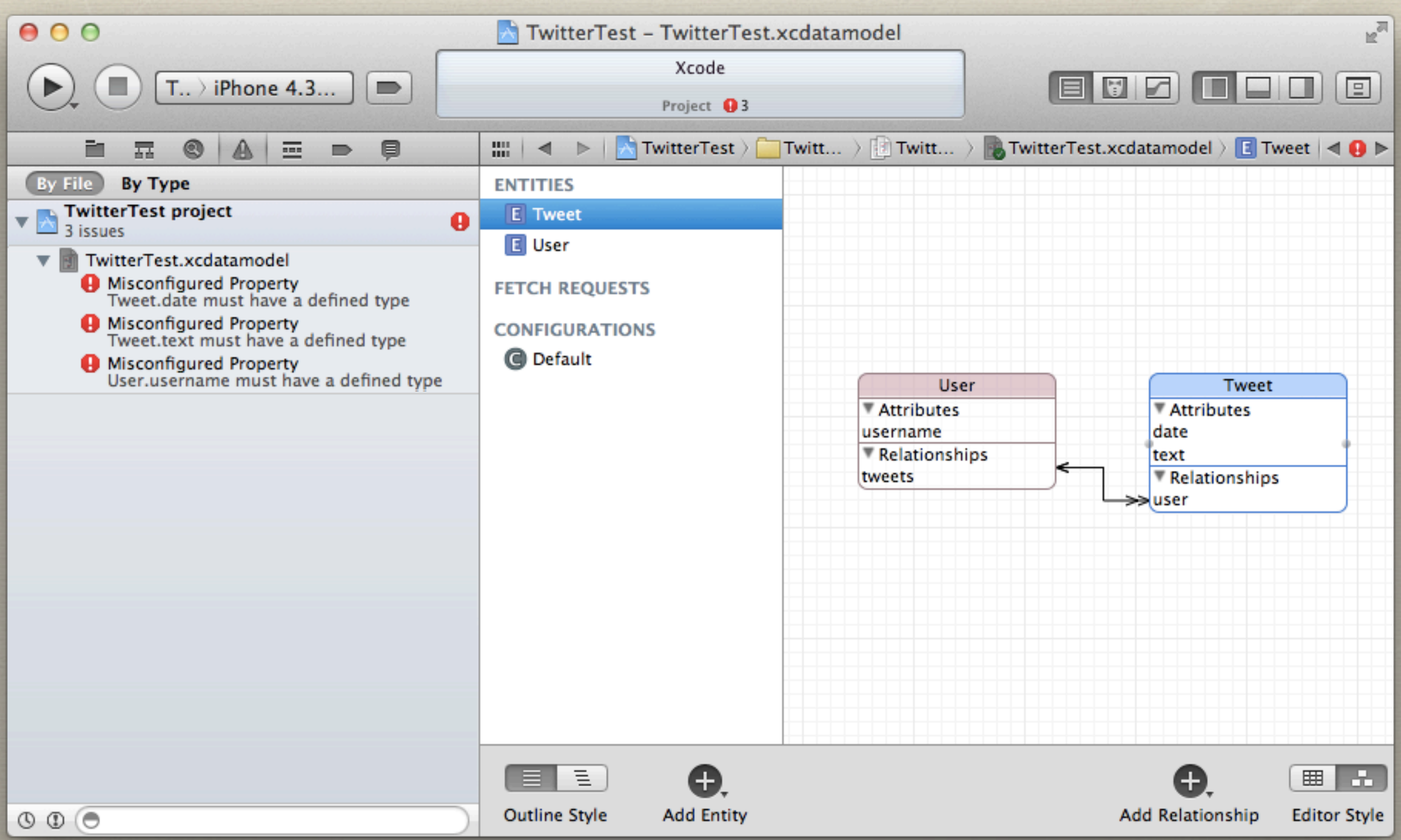
Hash Modifier Version Hash Modifier

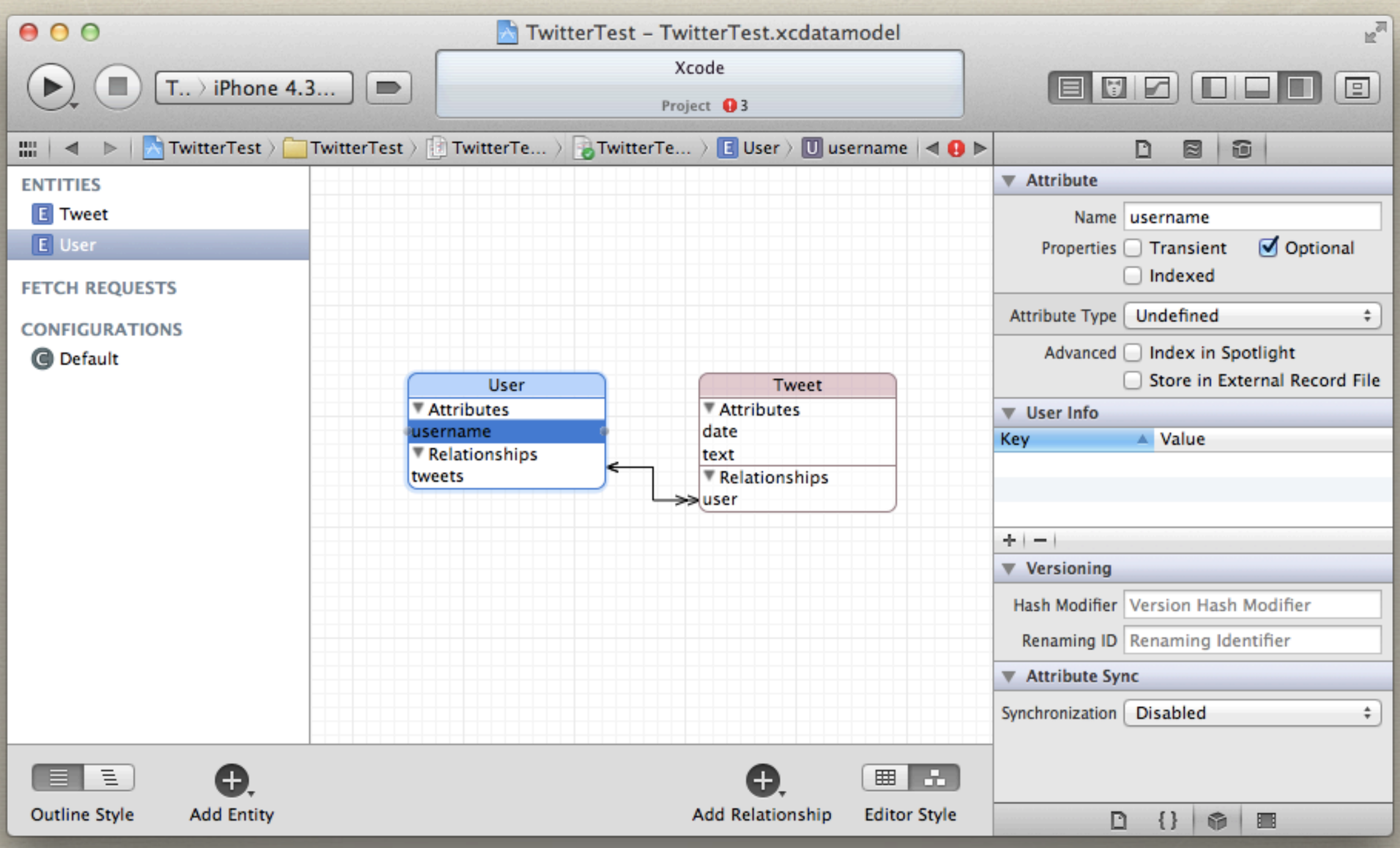
Outline Style

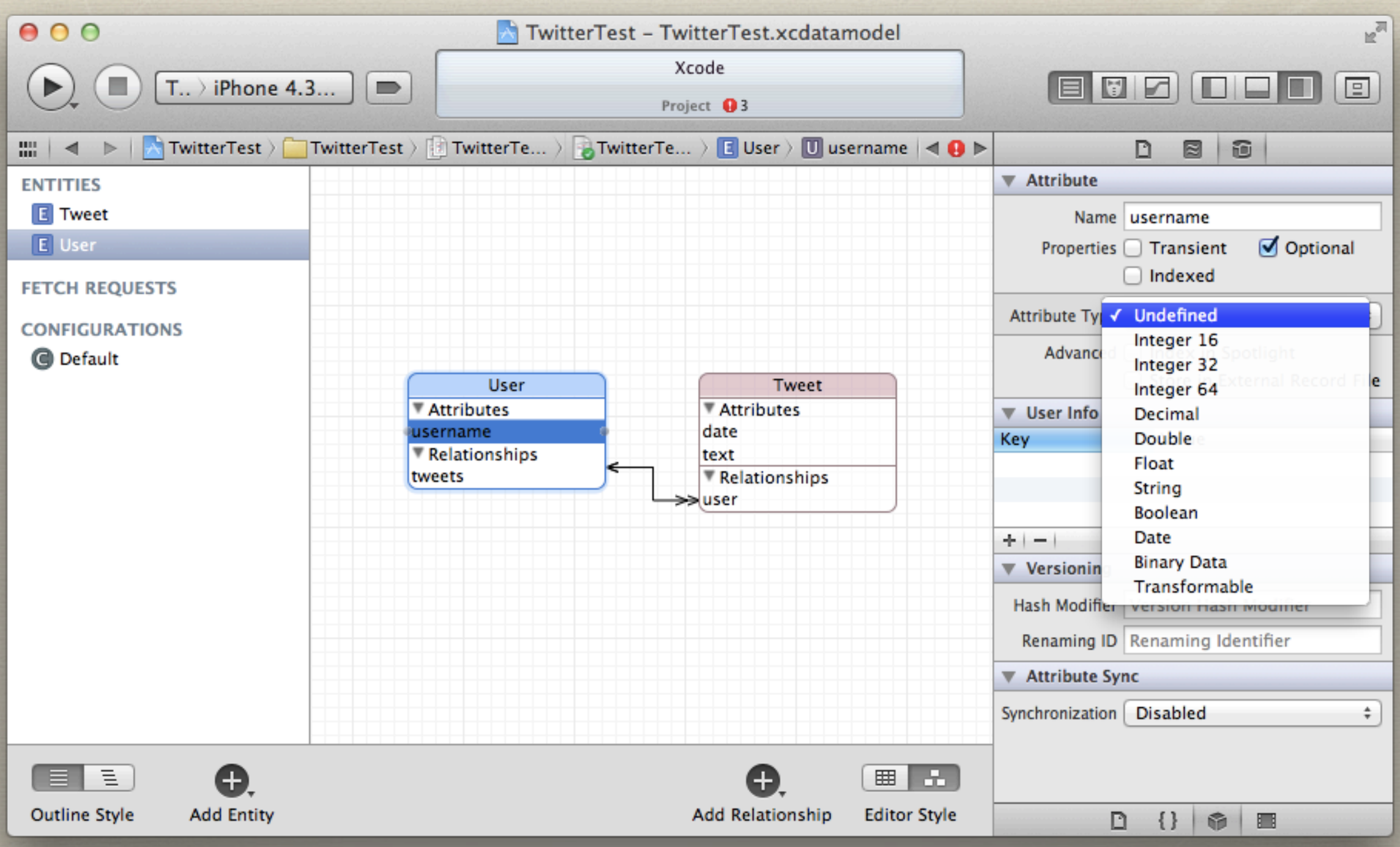
Add Entity

Add Relationship

Editor Style


TwitterTest – TwitterTest.xcdatamodel
Xcode
Project 3
T.. iPhone 4.3...
TwitterTest
Twitt...
Twitt...
TwitterTest.xcdatamodel
Tweet
By File
By Type
TwitterTest project
3 issues
TwitterTest.xcdatamodel
Misconfigured Property
Tweet.date must have a defined type
Misconfigured Property
Tweet.text must have a defined type
Misconfigured Property
User.username must have a defined type
ENTITIES
Tweet
User
FETCH REQUESTS
CONFIGURATIONS
Default
User
Attributes
username
Relationships
tweets
Tweet
Attributes
date
text
Relationships
user
Outline Style
Add Entity
Add Relationship
Editor Style


TwitterTest – TwitterTest.xcdatamodel
Xcode
Project 3
T.. iPhone 4.3...
TwitterTest
TwitterTest
TwitterTe...
TwitterTe...
User
username
ENTITIES
Tweet
User
FETCH REQUESTS
CONFIGURATIONS
Default
User
Attributes
username
Relationships
tweets
Tweet
Attributes
date
text
Relationships
user
Attribute
Name username
Properties Transient Optional
Indexed
Attribute Type Undefined
Advanced Index in Spotlight
Store in External Record File
User Info
Key
Value
Versioning
Hash Modifier Version Hash Modifier
Renaming ID Renaming Identifier
Attribute Sync
Synchronization Disabled
Outline Style
Add Entity
Add Relationship
Editor Style


TwitterTest – TwitterTest.xcdatamodel
Xcode
Project 3
T.. iPhone 4.3...
TwitterTest TwitterTest TwitterTe... TwitterTe... User username
ENTITIES
Tweet
User
FETCH REQUESTS
CONFIGURATIONS
Default
User
Attributes
username
Relationships
tweets
Tweet
Attributes
date
text
Relationships
user
Attribute
Name username
Properties Transient Optional
Indexed
Attribute Ty
Undefined
Integer 16
Integer 32
Integer 64
Decimal
Double
Float
String
Boolean
Date
Binary Data
Transformable
User Info
Key
Versionin
Hash Modifier
Renaming ID Renaming Identifier
Attribute Sync
Synchronization Disabled
Outline Style
Add Entity
Add Relationship
Editor Style

TwitterTest – TwitterTest.xcdatamodel

Xcode

Project 2

T.. iPhone 4.3...

TwitterTest › TwitterTest › TwitterTe... › TwitterTe... › User › username

ENTITIES

Tweet

User

FETCH REQUESTS

CONFIGURATIONS

Default

User
Attributes
username
Relationships
tweets

Tweet
Attributes
date
text
Relationships
user

Attribute

Name: username

Properties: Transient, Optional, Indexed

Attribute Type: String

Validation: No Value, Min Length

No Value, Max Length

Default Value: Default Value

Reg. Ex.: Regular Expression

Advanced: Index in Spotlight

Store in External Record File

User Info

Key | Value

Versioning

Hash Modifier: Version Hash Modifier

Renaming ID: Renaming Identifier

Outline Style

Add Entity

Add Relationship

Editor Style

TwitterTest – TwitterTest.xcdatamodel

Xcode

Project 2

T.. iPhone 4.3...

TwitterTest TwitterTest TwitterTest... TwitterTest... Tweet date

ENTITIES

Tweet

User

FETCH REQUESTS

CONFIGURATIONS

Default

User
Attributes
username
Relationships
tweets

Tweet
Attributes
date
text
Relationships
user

Attribute

Name date

Properties Transient Optional Indexed

Attribute Type Undefined

Advanced Index in Spotlight Store in External Record File

User Info

Key Value

Versioning

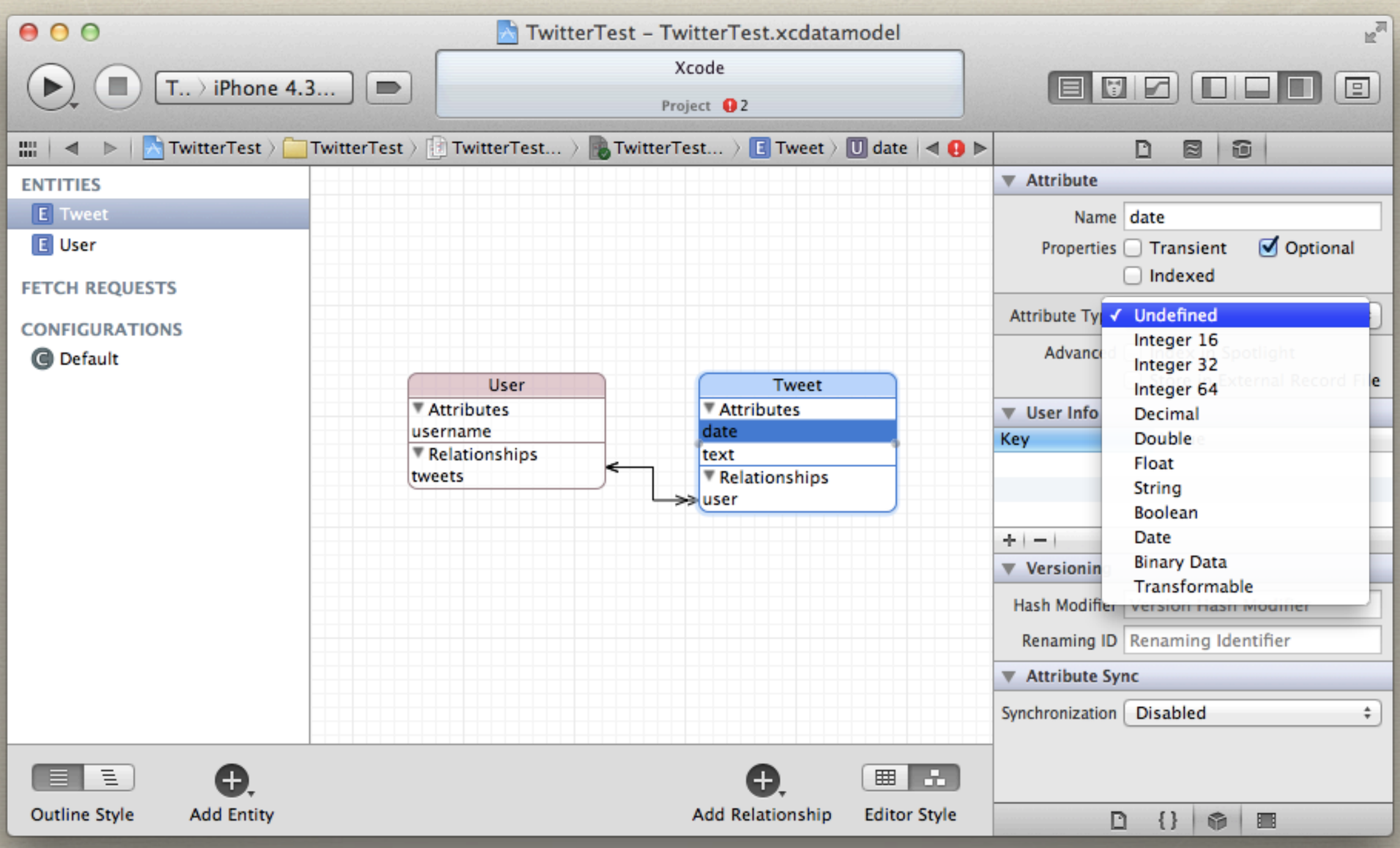
Hash Modifier Version Hash Modifier

Renaming ID Renaming Identifier

Attribute Sync

Synchronization Disabled

Outline Style Add Entity Add Relationship Editor Style


TwitterTest – TwitterTest.xcdatamodel
Xcode
Project 2
T.. iPhone 4.3...
TwitterTest
TwitterTest
TwitterTest...
TwitterTest...
Tweet
date
ENTITIES
Tweet
User
FETCH REQUESTS
CONFIGURATIONS
Default
User
Attributes
username
Relationships
tweets
Tweet
Attributes
date
text
Relationships
user
Attribute
Name date
Properties Transient Optional
Indexed
Attribute Ty
Undefined
Integer 16
Integer 32
Integer 64
Decimal
Double
Float
String
Boolean
Date
Binary Data
Transformable
User Info
Key
Versionin
Hash Modifier
Renaming ID Renaming Identifier
Attribute Sync
Synchronization Disabled
Outline Style
Add Entity
Add Relationship
Editor Style

TwitterTest – TwitterTest.xcdatamodel

Xcode

Project 1

T.. iPhone 4.3...

TwitterTest TwitterTest TwitterTest... TwitterTest... Tweet date

ENTITIES

Tweet

User

FETCH REQUESTS

CONFIGURATIONS

Default

User
Attributes
username
Relationships
tweets

Tweet
Attributes
date
text
Relationships
user

Attribute

Name date

Properties Transient Optional Indexed

Attribute Type Date

Minimum Minimum Value

Maximum

Default Default Value

Advanced Index in Spotlight Store in External Record File

User Info

Key Value

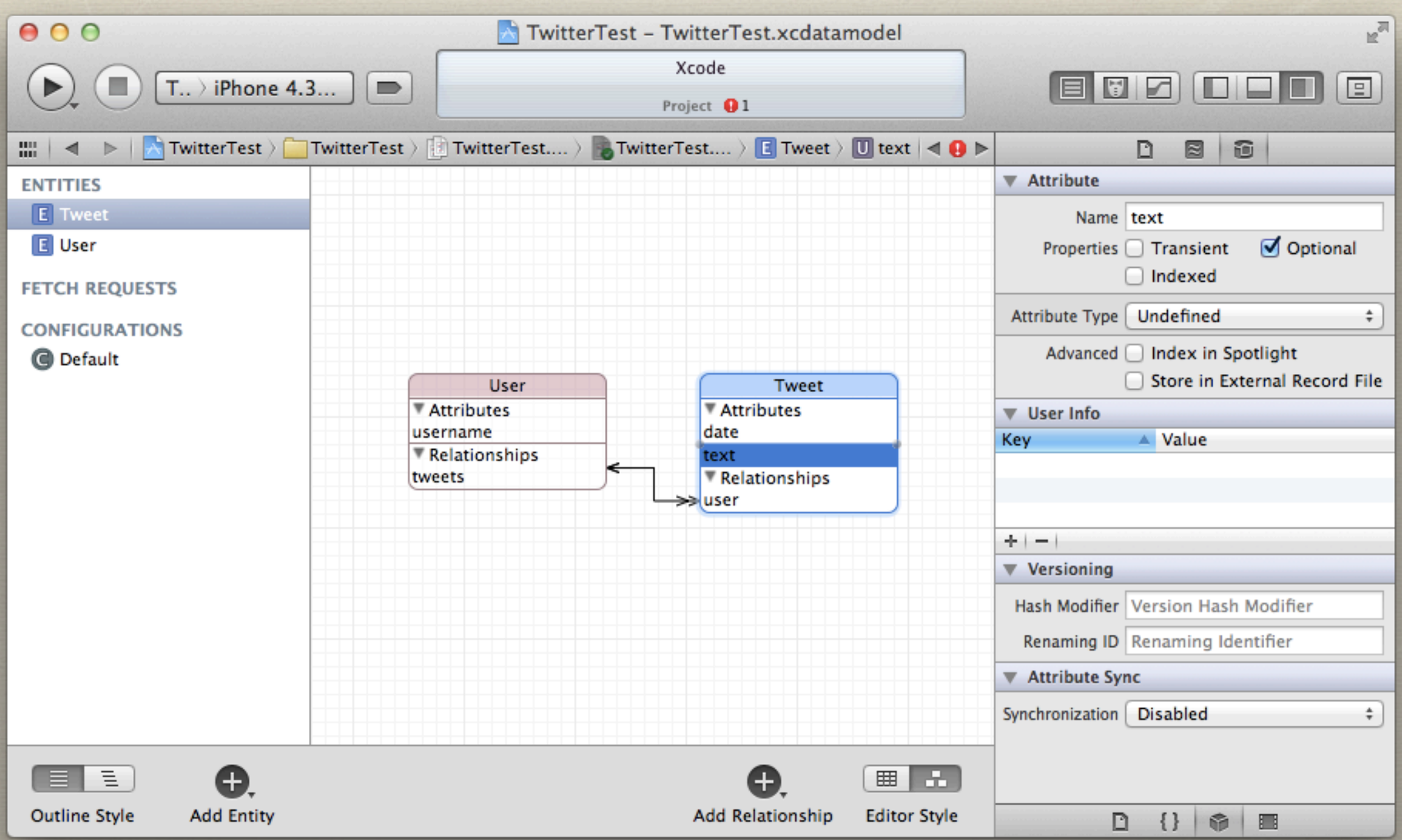
Versioning

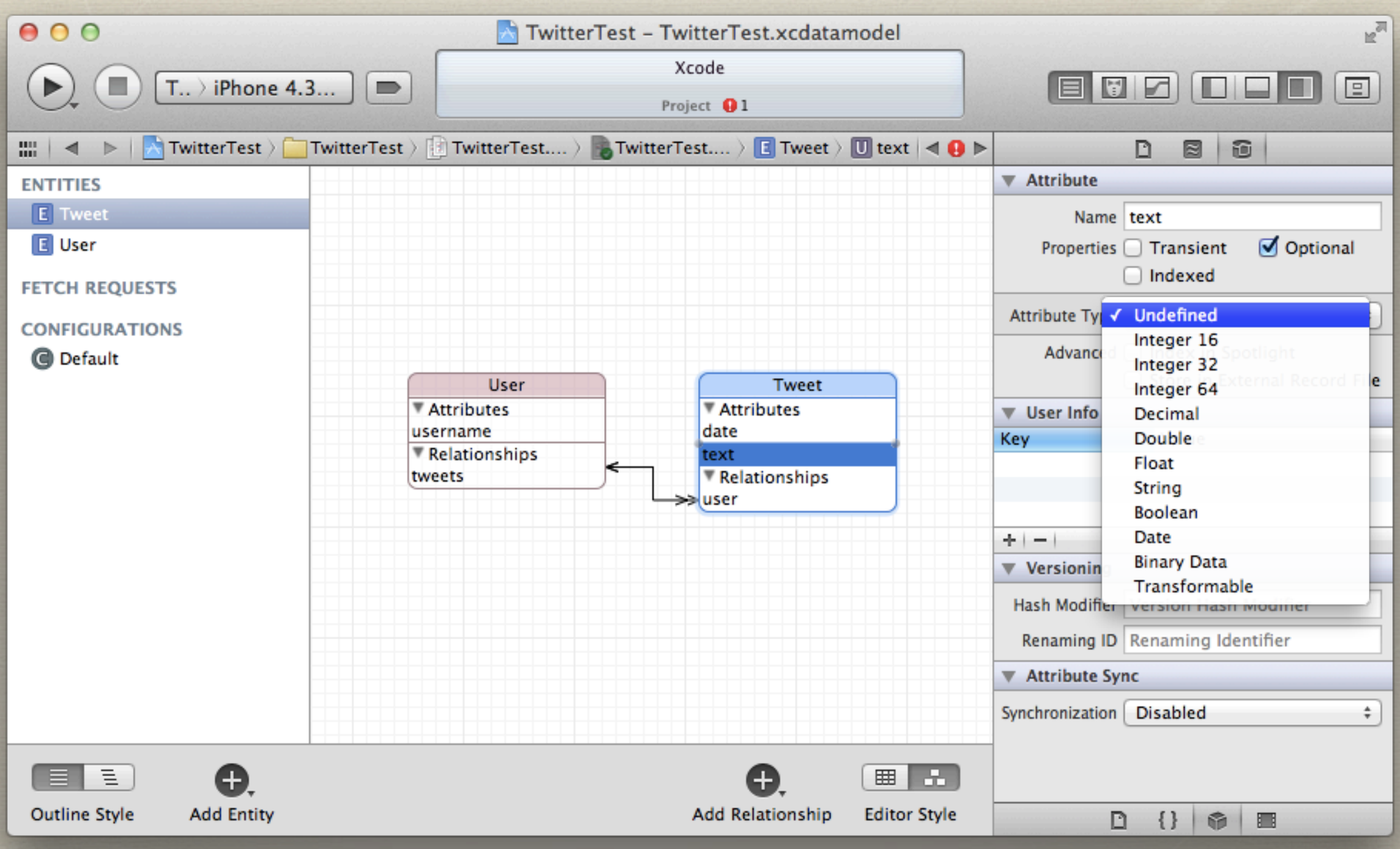
Hash Modifier Version Hash Modifier

Renaming ID Renaming Identifier

Attribute Sync

Outline Style Add Entity Add Relationship Editor Style


TwitterTest – TwitterTest.xcdatamodel
Xcode
Project 1
T.. iPhone 4.3...
TwitterTest
TwitterTest
TwitterTest....
TwitterTest....
Tweet
text
ENTITIES
Tweet
User
FETCH REQUESTS
CONFIGURATIONS
Default
User
Attributes
username
Relationships
tweets
Tweet
Attributes
date
text
Relationships
user
Attribute
Name text
Properties Transient Optional
Indexed
Attribute Type Undefined
Advanced Index in Spotlight
Store in External Record File
User Info
Key
Value
Versioning
Hash Modifier Version Hash Modifier
Renaming ID Renaming Identifier
Attribute Sync
Synchronization Disabled
Outline Style
Add Entity
Add Relationship
Editor Style


TwitterTest – TwitterTest.xcdatamodel
Xcode
Project 1
T.. iPhone 4.3...
TwitterTest
TwitterTest
TwitterTest....
TwitterTest....
Tweet
text
ENTITIES
Tweet
User
FETCH REQUESTS
CONFIGURATIONS
Default
User
Attributes
username
Relationships
tweets
Tweet
Attributes
date
text
Relationships
user
Attribute
Name text
Properties Transient Optional
Indexed
Attribute Ty
Undefined
Integer 16
Integer 32
Integer 64
Decimal
Double
Float
String
Boolean
Date
Binary Data
Transformable
User Info
Key
Versionin
Hash Modifier
Renaming ID Renaming Identifier
Attribute Sync
Synchronization Disabled
Outline Style
Add Entity
Add Relationship
Editor Style

TwitterTest – TwitterTest.xcdatamodel

Xcode

No Issues

T.. › iPhone 4.3...

TwitterTest › TwitterTest › TwitterTest.xcd... › TwitterTest.xcd... › Tweet › text

ENTITIES

Tweet

User

FETCH REQUESTS

CONFIGURATIONS

Default

User
Attributes
username
Relationships
tweets

Tweet
Attributes
date
text
Relationships
user

Attribute

Name: text

Properties: Transient, Optional, Indexed

Attribute Type: String

Validation: No Value, Min Length; No Value, Max Length

Default Value: Default Value

Reg. Ex.: Regular Expression

Advanced: Index in Spotlight, Store in External Record File

User Info

Key | Value

Versioning

Hash Modifier: Version Hash Modifier

Renaming ID: Renaming Identifier

Outline Style | Add Entity | Add Relationship | Editor Style

TwitterTest – TwitterTest.xcdatamodel

Xcode

No Issues

T.. 〉 iPhone 4.3...

TwitterTest 〉 TwitterTest 〉 TwitterTest.xcdata... 〉 TwitterTest.xcdatamodel 〉 Tweet

ENTITIES

- Tweet
- User

FETCH REQUESTS

CONFIGURATIONS

- Default

Attributes

Attribute	Type
date	Date
text	String

Relationships

Relationship	Destination	Inverse
user	User	tweets

Fetched Properties

Fetched Property	Predicate

Entity

Name: Tweet

Class: NSManagedObject

Abstract Entity

Parent Entity: No Parent Entity

Indexes

User Info

Key	Value

Versioning

Hash Modifier: Version Hash Modifier

Renaming ID: Renaming Identifier

Entity Sync

Synchronization: Disabled

Outline Style | Add Entity | Add Relationship | Editor Style

TwitterTest – TwitterTest.xcdatamodel

Xcode

No Issues

T.. 〉 iPhone 4.3...

TwitterTest 〉 TwitterTest 〉 TwitterTest.xcdatam... 〉 TwitterTest.xcdatamodel 〉 User

ENTITIES

- Tweet
- User

FETCH REQUESTS

CONFIGURATIONS

- Default

Attributes

Attribute	Type
username	String

Relationships

Relationship	Destination	Inverse
tweets	Tweet	user

Fetched Properties

Fetched Property	Predicate

Entity

Name: User

Class: NSManagedObject

Abstract Entity

Parent Entity: No Parent Entity

Indexes

User Info

Key	Value

Versioning

Hash Modifier: Version Hash Modifier

Renaming ID: Renaming Identifier

Entity Sync

Synchronization: Disabled

Outline Style | Add Entity | Add Relationship | Editor Style

Core Data Stack

NSPersistentStoreCoordinator

NSManagedObjectModel

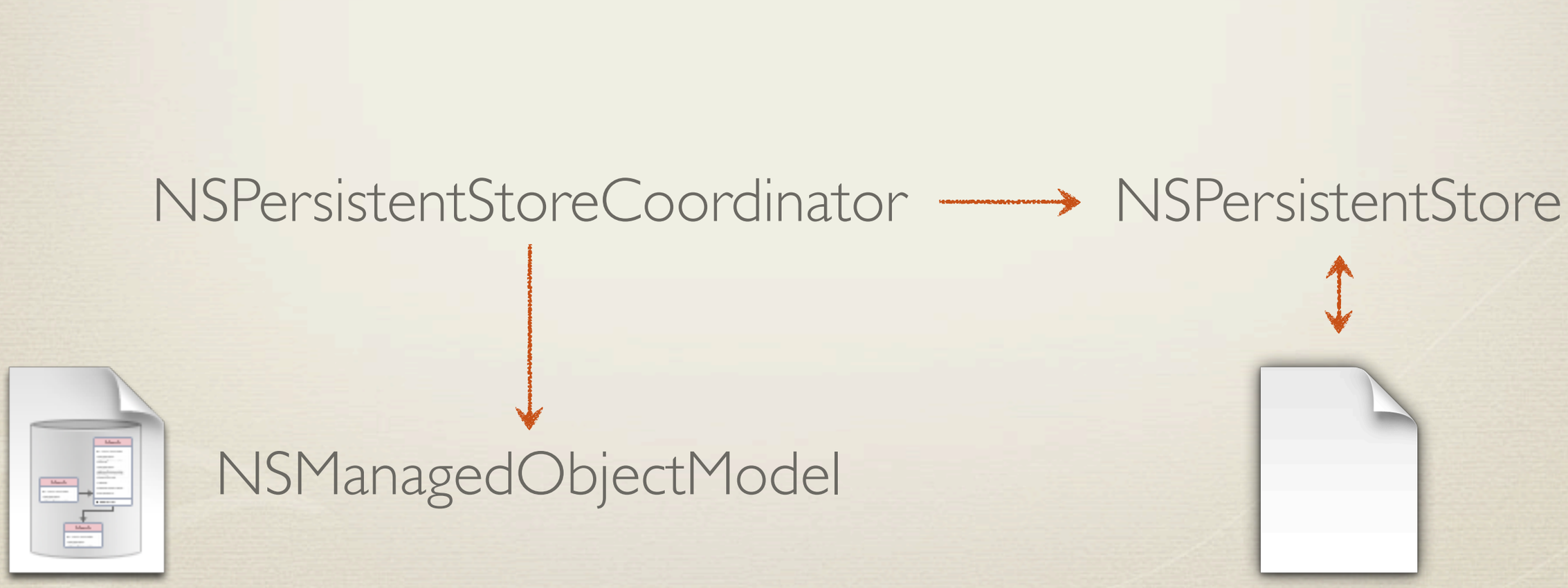
Core Data Stack

NSPersistentStoreCoordinator → NSPersistentStore

↓

NSManagedObjectModel

Core Data Stack



Core Data Stack

NSManagedObjectContext

NSPersistentStoreCoordinator → NSPersistentStore

NSManagedObjectModel

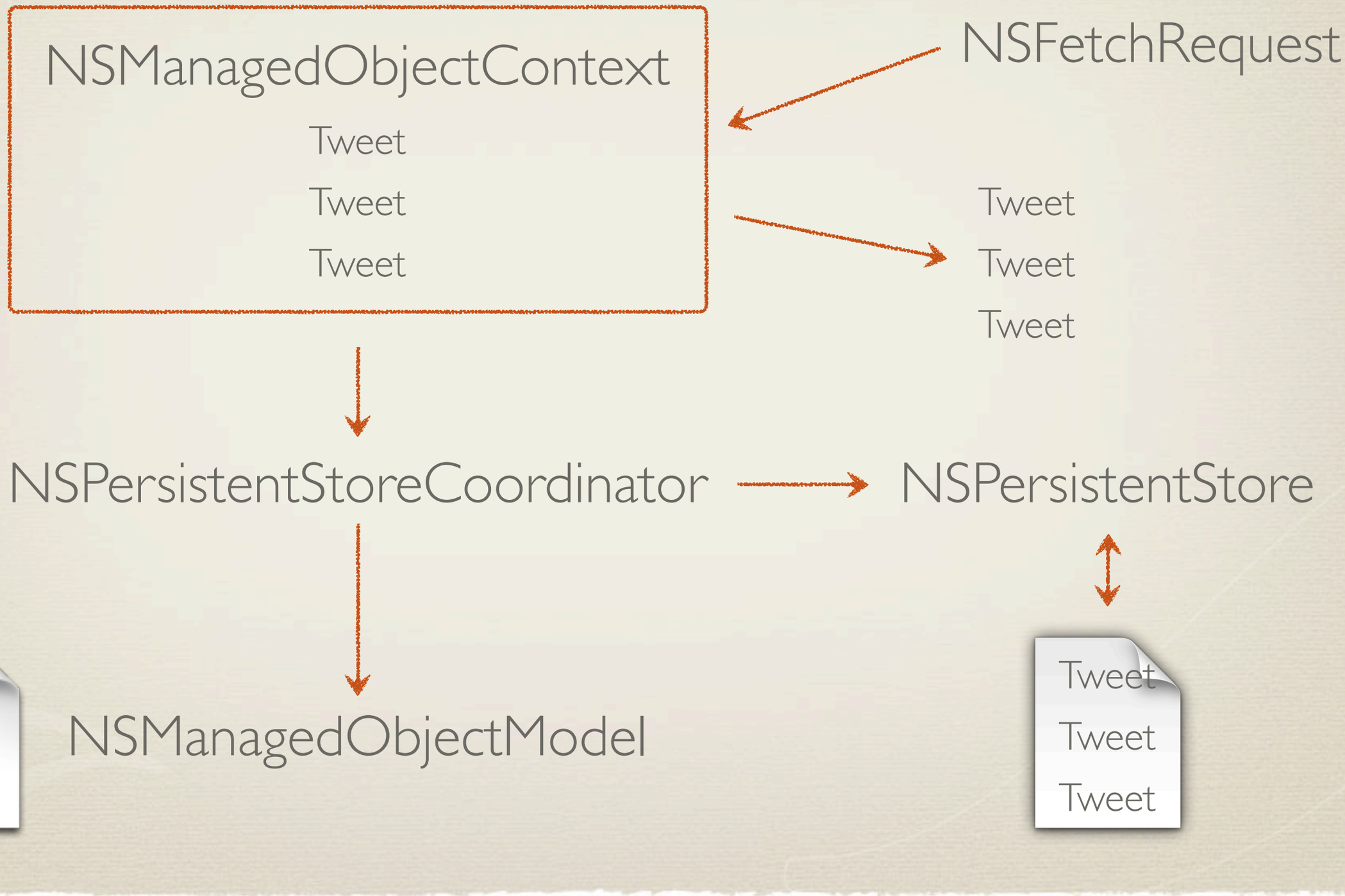
Fetching Objects

Create a fetch request

- Set entity
- Set a predicate (optional)
- Set sort descriptors (optional)

Ask the context to execute the request

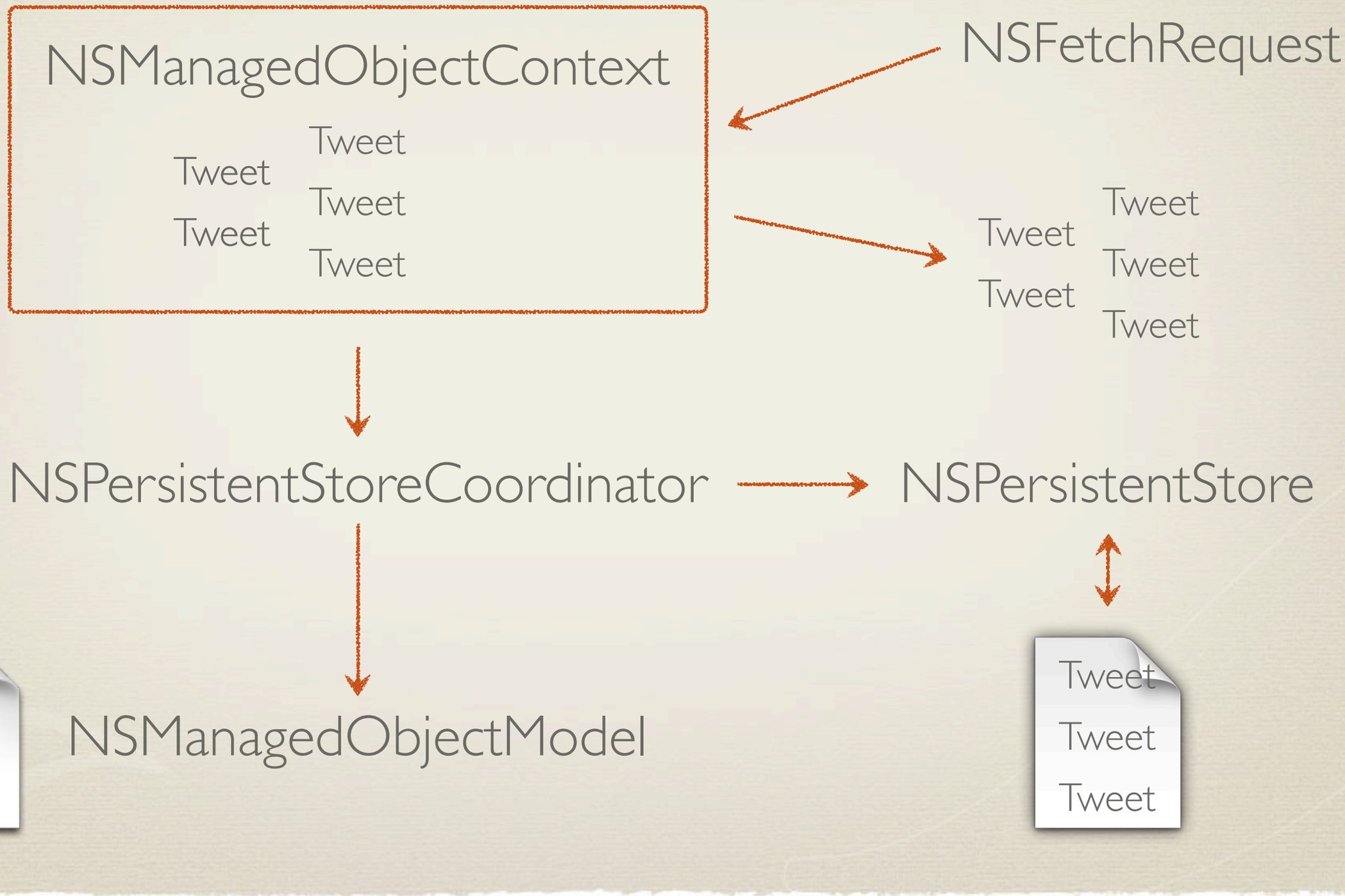
Fetching Objects



Fetching Objects

```
NSEntityDescription *entity = [NSEntityDescription
                                        entityForName:@“Tweet”
                                    inManagedObjectContext:context];

NSPredicate *predicate = [NSPredicate
                          predicateWithFormat:@“text CONTAINS %@”, string];

NSSortDescriptor *sortDescriptor = [[NSSortDescriptor alloc]
                                          initWithKey:@“date” ascending:NO];

NSArray *descriptors = [NSArray arrayWithObject:sortDescriptor];

NSFetchRequest *request = [[NSFetchRequest alloc] init];
[request setEntity:entity];
[request setPredicate:predicate];
[request setSortDescriptors:sortDescriptors];

NSArray *fetchResult = [context executeFetchRequest:request
error:&error];
```

Fetching Objects



Create and Save Objects

Call NSEntityDescription class method to insert a new managed object

Set the object's properties

Save the managed object context

Create and Save Objects

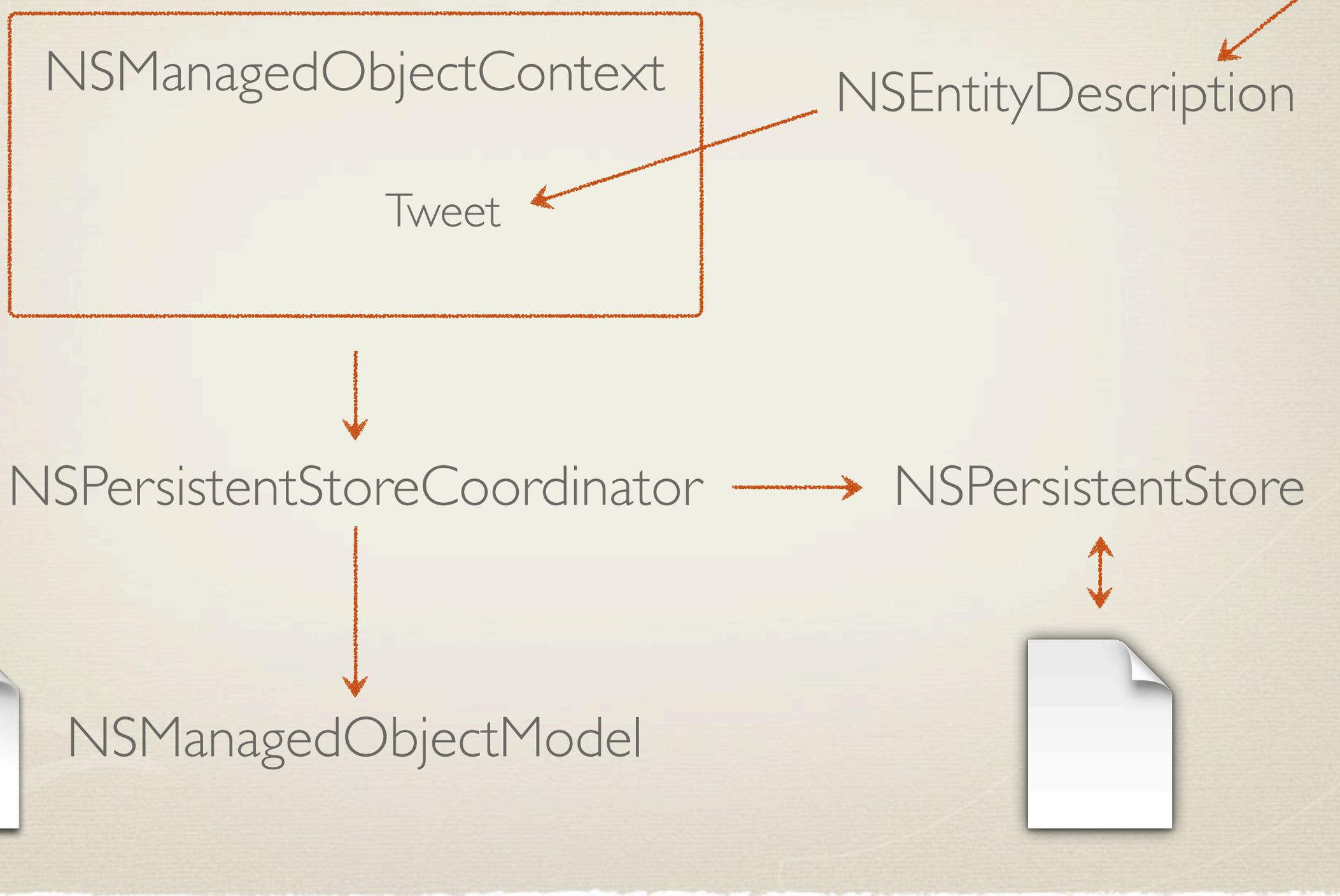
NSManagedObjectContext

NSEntityDescription

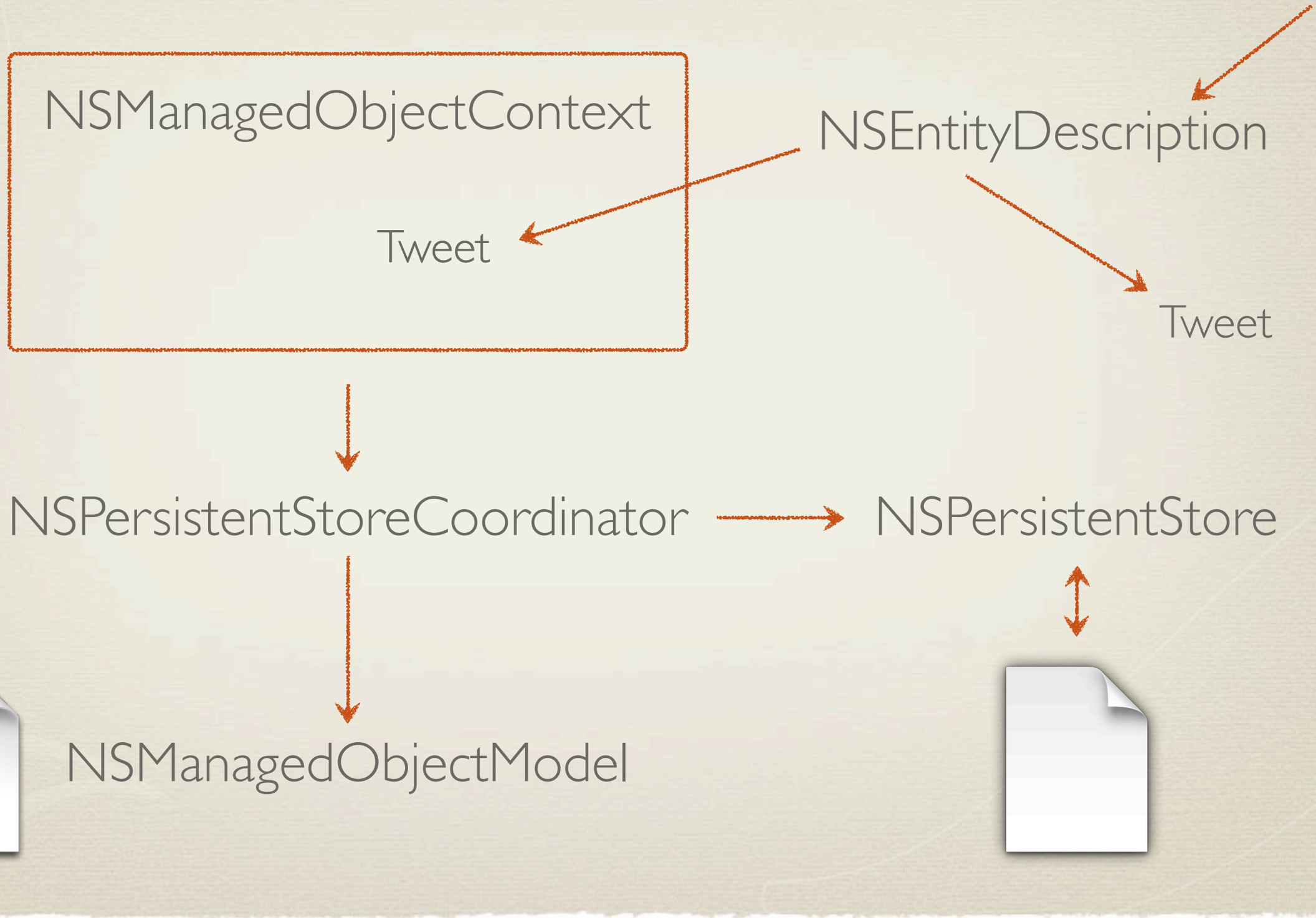
NSPersistentStoreCoordinator → NSPersistentStore

NSManagedObjectModel

Create and Save Objects



Create and Save Objects



Create and Save Objects

```
Tweet *tweet = [NSEntityDescription
insertNewObjectForEntityForName:@“Tweet”
inManagedObjectContext:context];

tweet.text = @“This is a new tweet”;

tweet.date = [NSDate date];

User *someUser = // fetch a user from the context

tweet.user = someUser;

NSError *error = nil;

if (![managedObjectContext save:&error])
  NSLog(@”%@”, error); // handle error
```

NSFetchedResultsController

NSFetchedResultsController

A method of keeping a table view showing the current data in a managed object context

Give it a context and fetch request

It will notify of changes to the data to allow you to update the table view

The APIs gel nicely with one another

NSFetchedResultsController

Fetched Results Controller

```
controllerWillChangeContent:

controller:
didChangeSection:
atIndex:
forChangeType:

controller:
didChangeObject:
atIndexPath:
forChangeType:
newIndexPath:

controllerDidChangeContent:
```

UITableView

```
beginUpdates:

insertSections:    deleteSections:
withRowAnimation:  withRowAnimation:

insertRowsAtIndexPaths:
      withRowAnimation:

deleteRowsAtIndexPaths:
      withRowAnimation:

reloadRowsAtIndexPaths:
      withRowAnimation:

endUpdates:
```

TableView

```
withRowAnimation:UITableViewRowAnimationFade];

break;

case NSFetchedResultsChangeUpdate:

[self configureCell:[tableView cellForRowAtIndexPath:indexPath]
atIndexPath:indexPath];

break;

case NSFetchedResultsChangeMove:

[tableView deleteRowsAtIndexPaths:[NSArray
arrayWithObject:indexPath]
withRowAnimation:UITableViewRowAnimationFade];

[tableView insertRowsAtIndexPaths:[NSArray
arrayWithObject:newIndexPath]
withRowAnimation:UITableViewRowAnimationFade];

break;

}

}


- (void)controllerDidChangeContent:(NSFetchedResultsController
*)controller {

    [self.tableView endUpdates];

}
```

NSFetchedResultsController Issues

- Somewhat broken on iPhone OS 3.0
  - Apple suggest not using the change methods and just hard reload the table view in did change
- Works well on iOS 4
  - Very rarely, it can still get the table and data out of sync
  - Do a check when getting the cell and hard reload the table view if it's out of sync

NSFetchedResultsController Issues

```
- (UITableViewCell *)tableView:(UITableView *)tableView
         cellForRowAtIndexPath:(NSIndexPath *)indexPath {

    NSInteger numberOfRows = [self tableView:tableView
numberOfRowsInSection:indexPath.section];

    if (indexPath.row >= numberOfRows) {
        NSLog(@"FORCE RELOADING TABLE VIEW");

        [tableView reloadData];

        return [[[UITableViewCell alloc]
initWithStyle:UITableViewCellStyleDefault reuseIdentifier:@"blah"]
autorelease];

    }

    return someTableViewCell;

}
```

NSFetchedResultsController Issues

```
- (UITableViewCell *)tableView:(UITableView *)tableView
         cellForRowAtIndexPath:(NSIndexPath *)indexPath {

    NSInteger numberOfRows = [self tableView:tableView
numberOfRowsInSection:indexPath.section];

    if (indexPath.row >= numberOfRows) {

        NSLog(@"FORCE RELOADING TABLE VIEW");

        [tableView reloadData];

        return [[[UITableViewCell alloc]
initWithStyle:UITableViewCellStyleDefault reuseIdentifier:@"blah"]
autorelease];

    }

    return someTableViewCell;

}
```

NSFetchedResultsController Issues

```
- (UITableViewCell *)tableView:(UITableView *)tableView
         cellForRowAtIndexPath:(NSIndexPath *)indexPath {

    NSInteger numberOfRows = [self tableView:tableView
numberOfRowsInSection:indexPath.section];

    if (indexPath.row >= numberOfRows) {

        NSLog(@"FORCE RELOADING TABLE VIEW");

        [tableView reloadData];

        return [[[UITableViewCell alloc]
initWithStyle:UITableViewCellStyleDefault reuseIdentifier:@"blah"]
autorelease];

    }

    return someTableViewCell;

}
```

NSFetchedResultsController Issues

```
- (UITableViewCell *)tableView:(UITableView *)tableView
         cellForRowAtIndexPath:(NSIndexPath *)indexPath {

    NSInteger numberOfRows = [self tableView:tableView
numberOfRowsInSection:indexPath.section];

    if (indexPath.row >= numberOfRows) {

        NSLog(@"FORCE RELOADING TABLE VIEW");

        [tableView reloadData];

        return [[[UITableViewCell alloc]
initWithStyle:UITableViewCellStyleDefault reuseIdentifier:@"blah"]
autorelease];

    }

    return someTableViewCell;

}
```

NSFetchedResultsController Issues

```
- (UITableViewCell *)tableView:(UITableView *)tableView
         cellForRowAtIndexPath:(NSIndexPath *)indexPath {

    NSInteger numberOfRows = [self tableView:tableView
numberOfRowsInSection:indexPath.section];

    if (indexPath.row >= numberOfRows) {

        NSLog(@"FORCE RELOADING TABLE VIEW");

        [tableView reloadData];

        return [[[UITableViewCell alloc]
initWithStyle:UITableViewCellStyleDefault reuseIdentifier:@"blah"]
autorelease];

    }

    return someTableViewCell;

}
```

Concurrency

Concurrency

Use a different Managed Object Context per thread

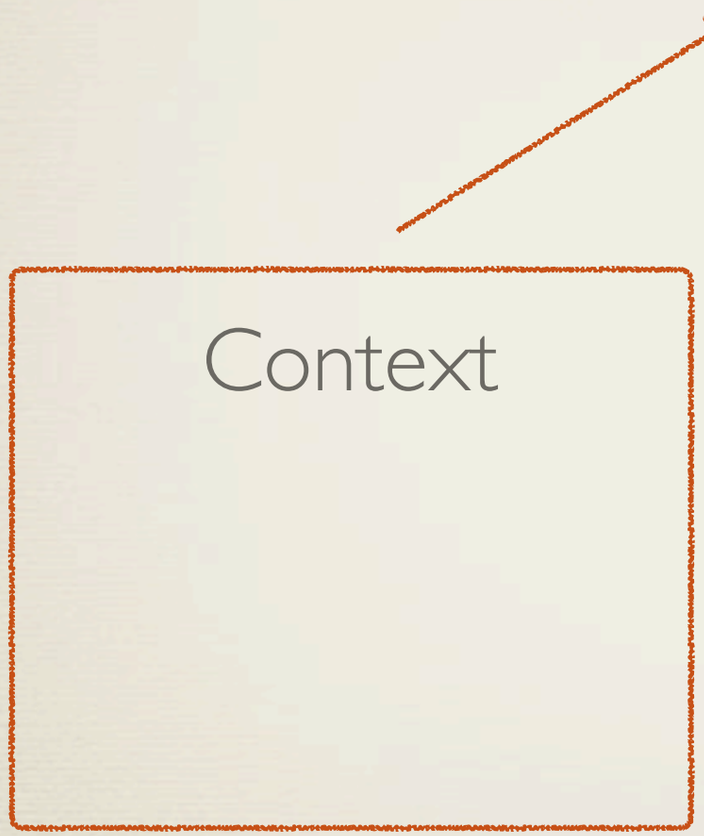
Cannot use a managed object from one context in another context

NSManagedObjectID is thread safe and used to reference objects across threads

Use a notification to merge changes from one context to another

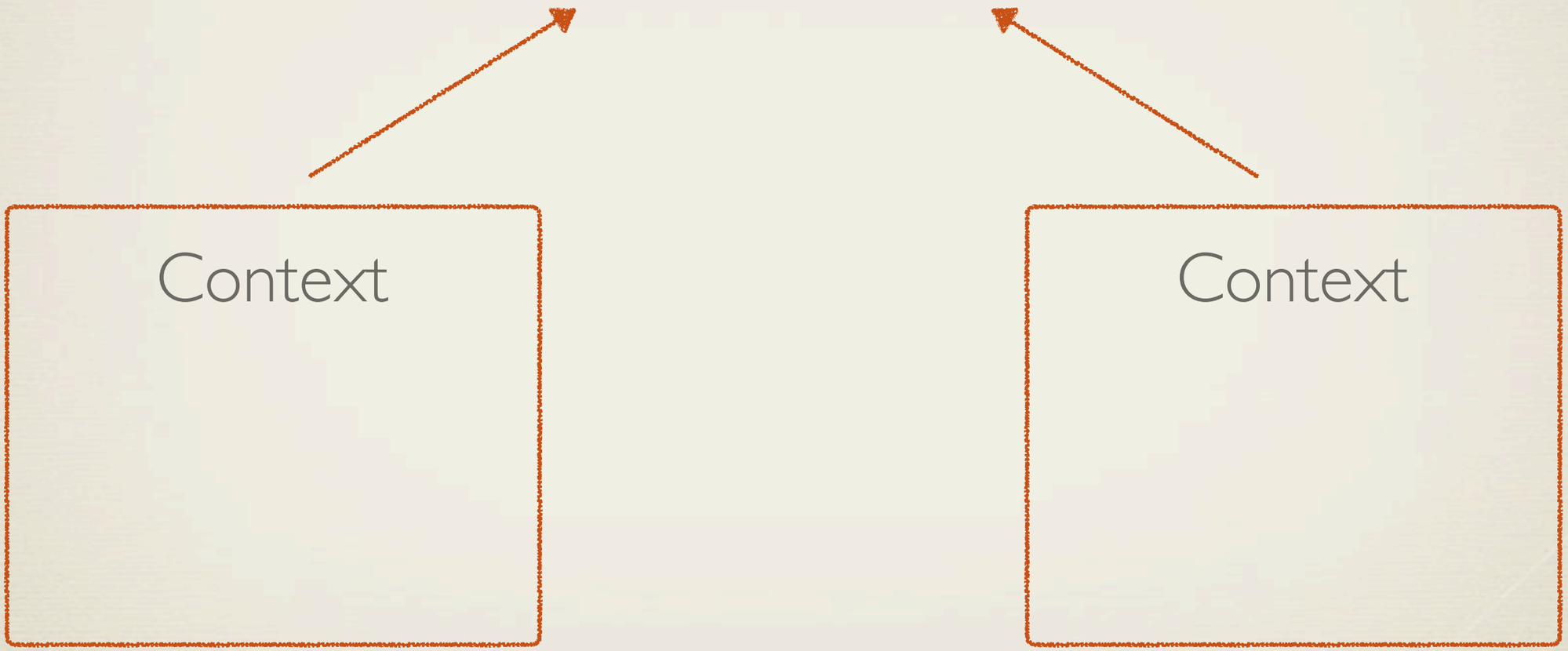
Cross-referencing Managed Objects

Persistent Store Coordinator



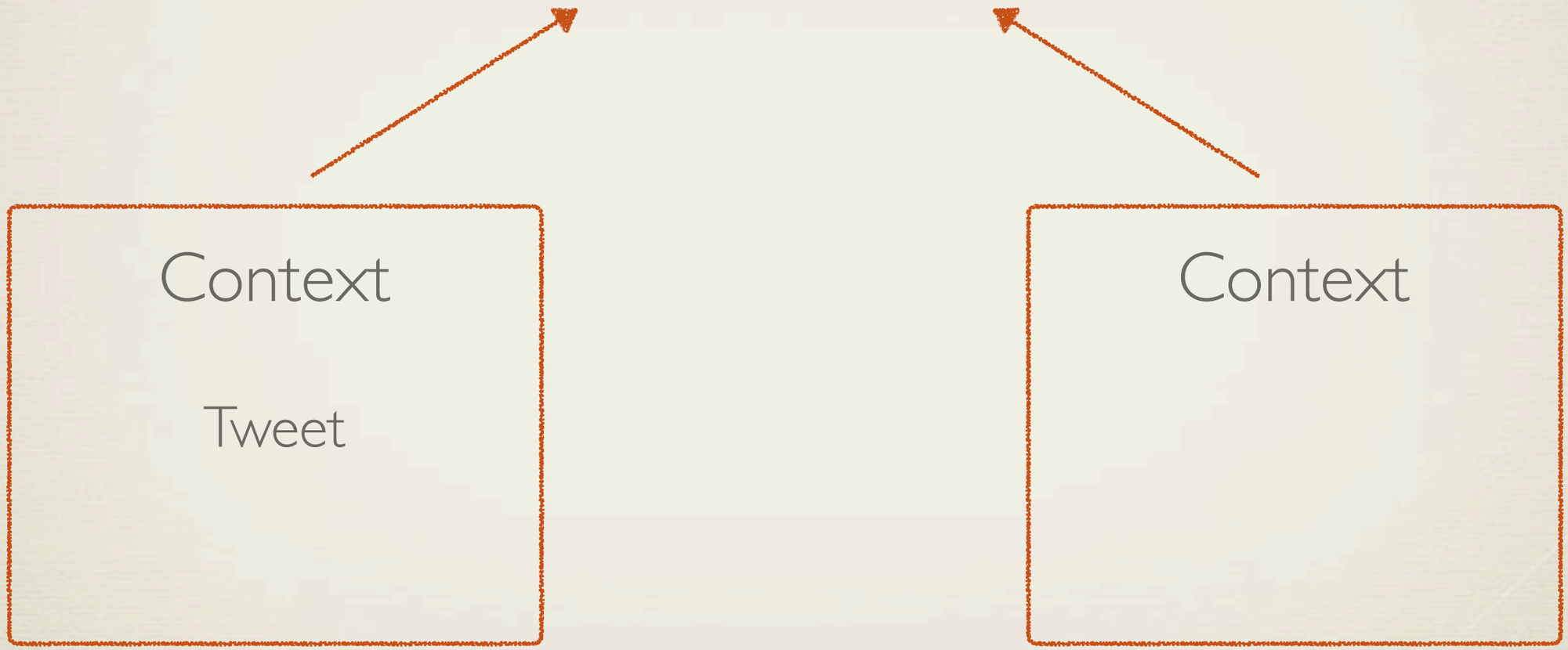
Cross-referencing Managed Objects

Persistent Store Coordinator



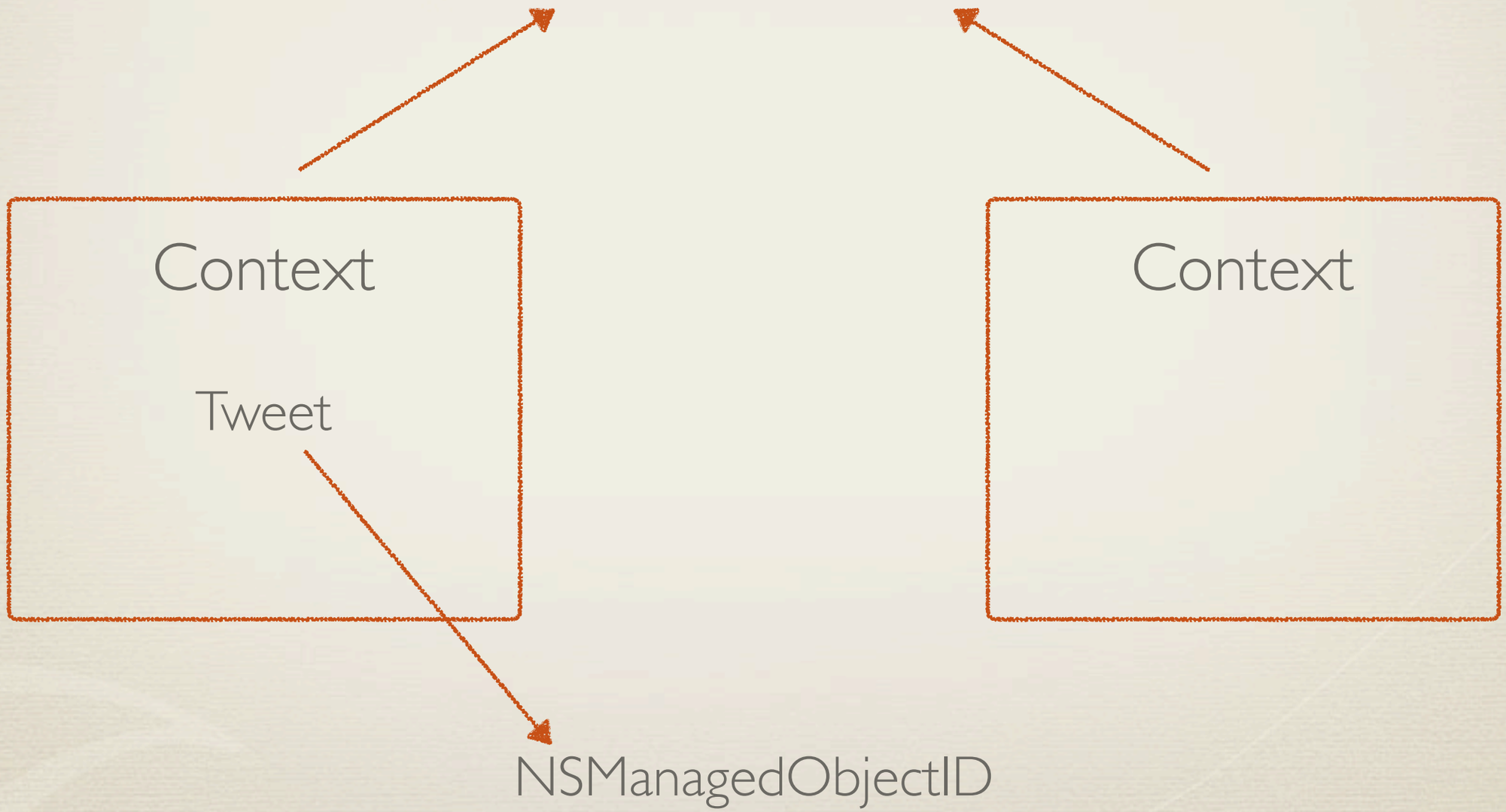
Cross-referencing Managed Objects

Persistent Store Coordinator

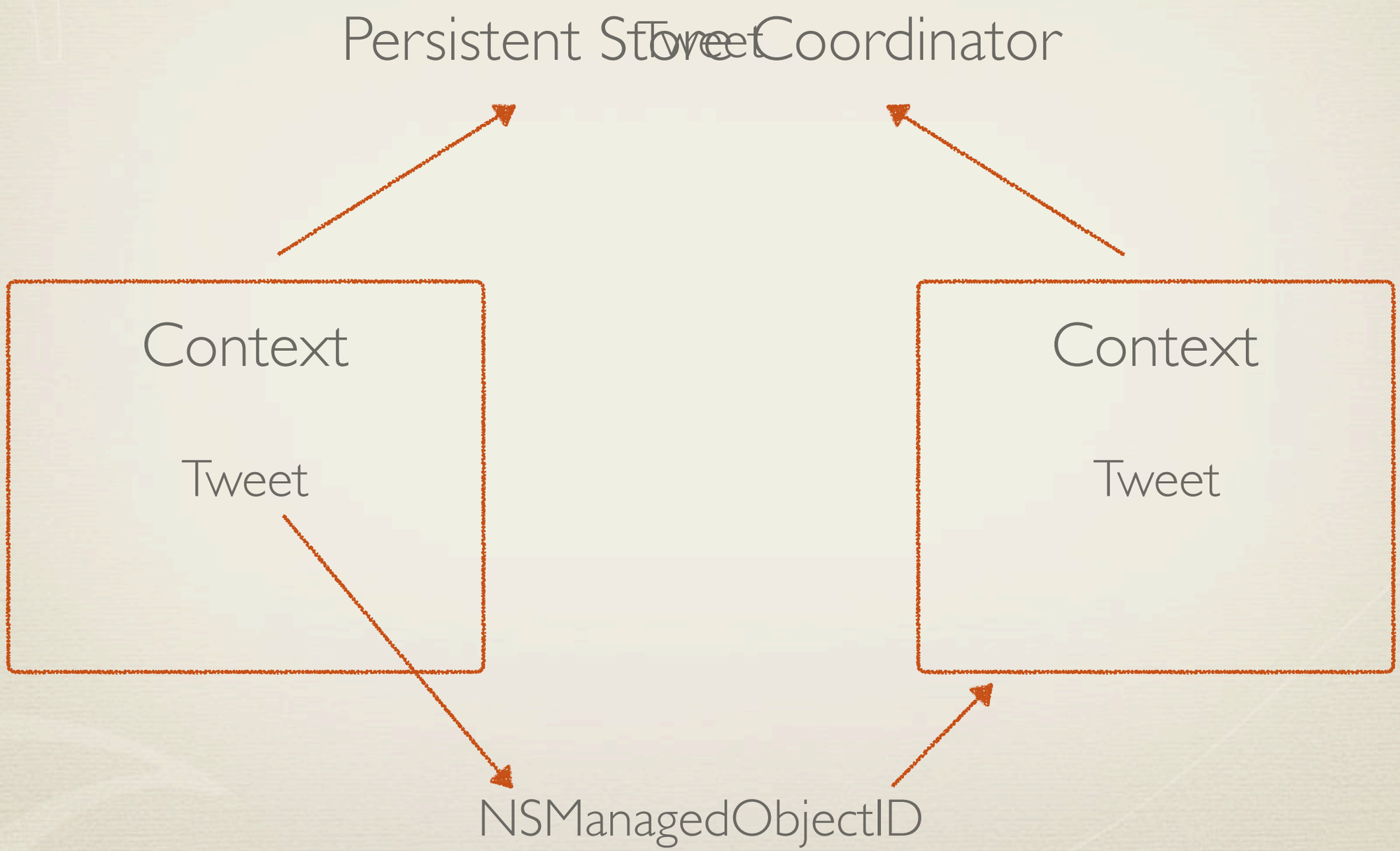


Cross-referencing Managed Objects

Persistent Store Coordinator



Cross-referencing Managed Objects



Merging Contexts

Persistent Store Coordinator

Context

User

Context

User

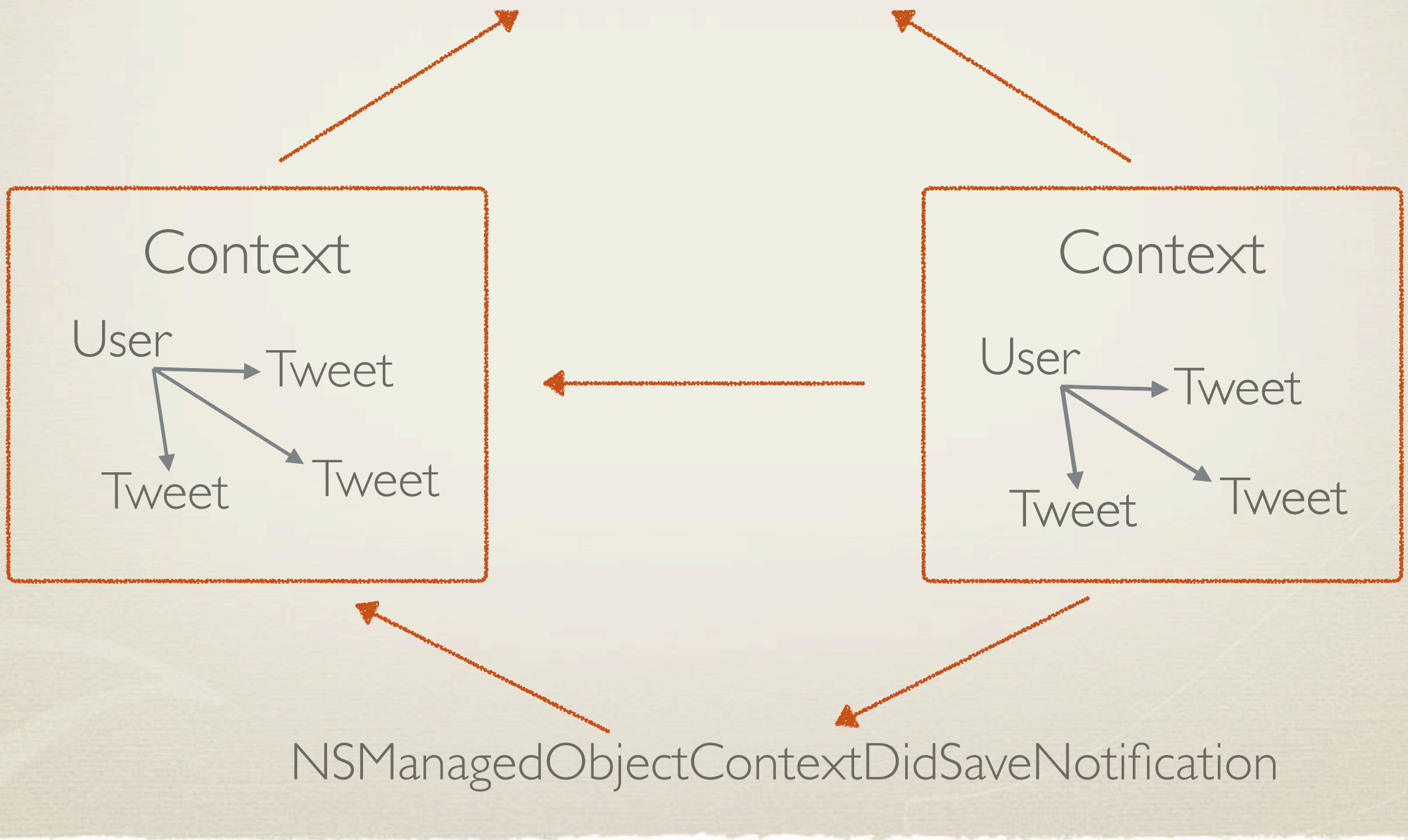
Tweet

Tweet

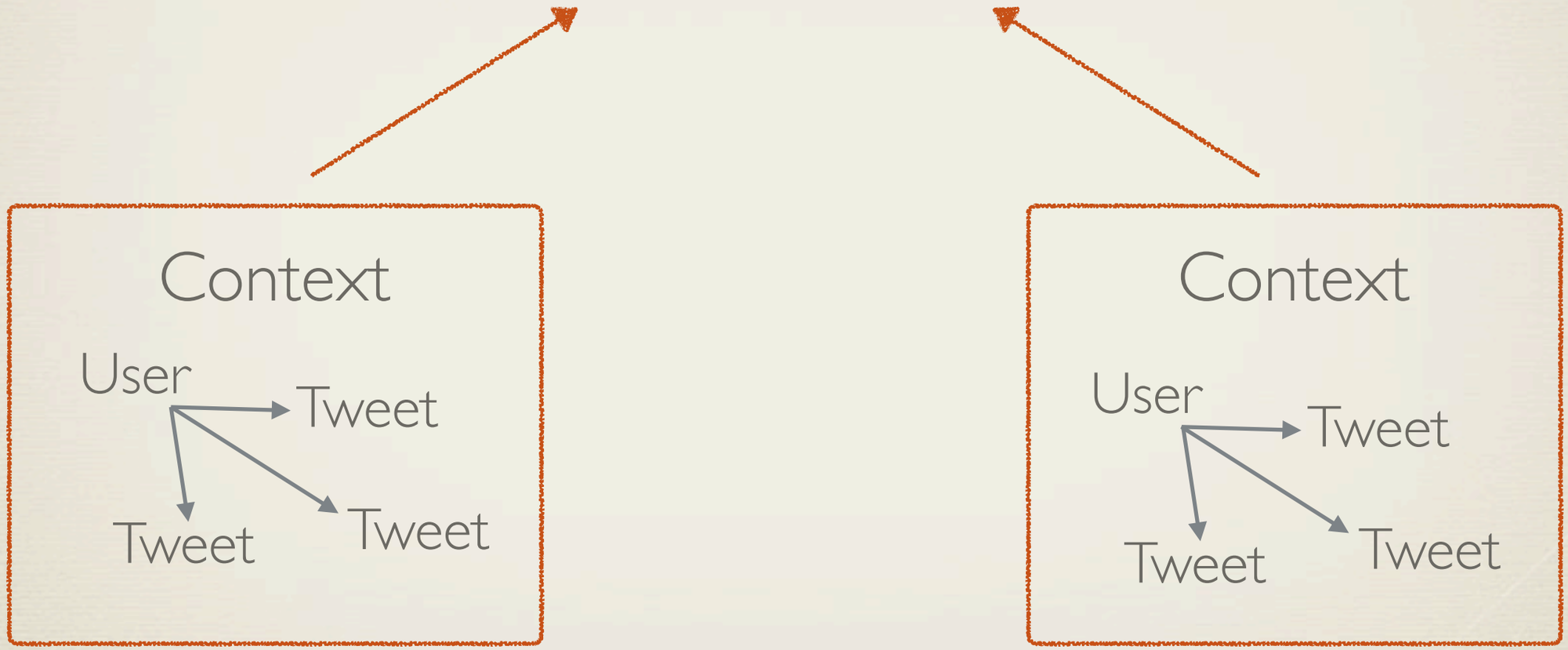
Tweet

NSManagedObjectContextDidSaveNotification

Merging Contexts



Merging Contexts



Merging Contexts

```
- (void)threadedContextDidSave:(NSNotification *)notification {
    [self setMergePolicy:NSMergeByPropertyStoreTrumpMergePolicy];
    [mainContext mergeChangesFromContextDidSaveNotification:notification];
}
```

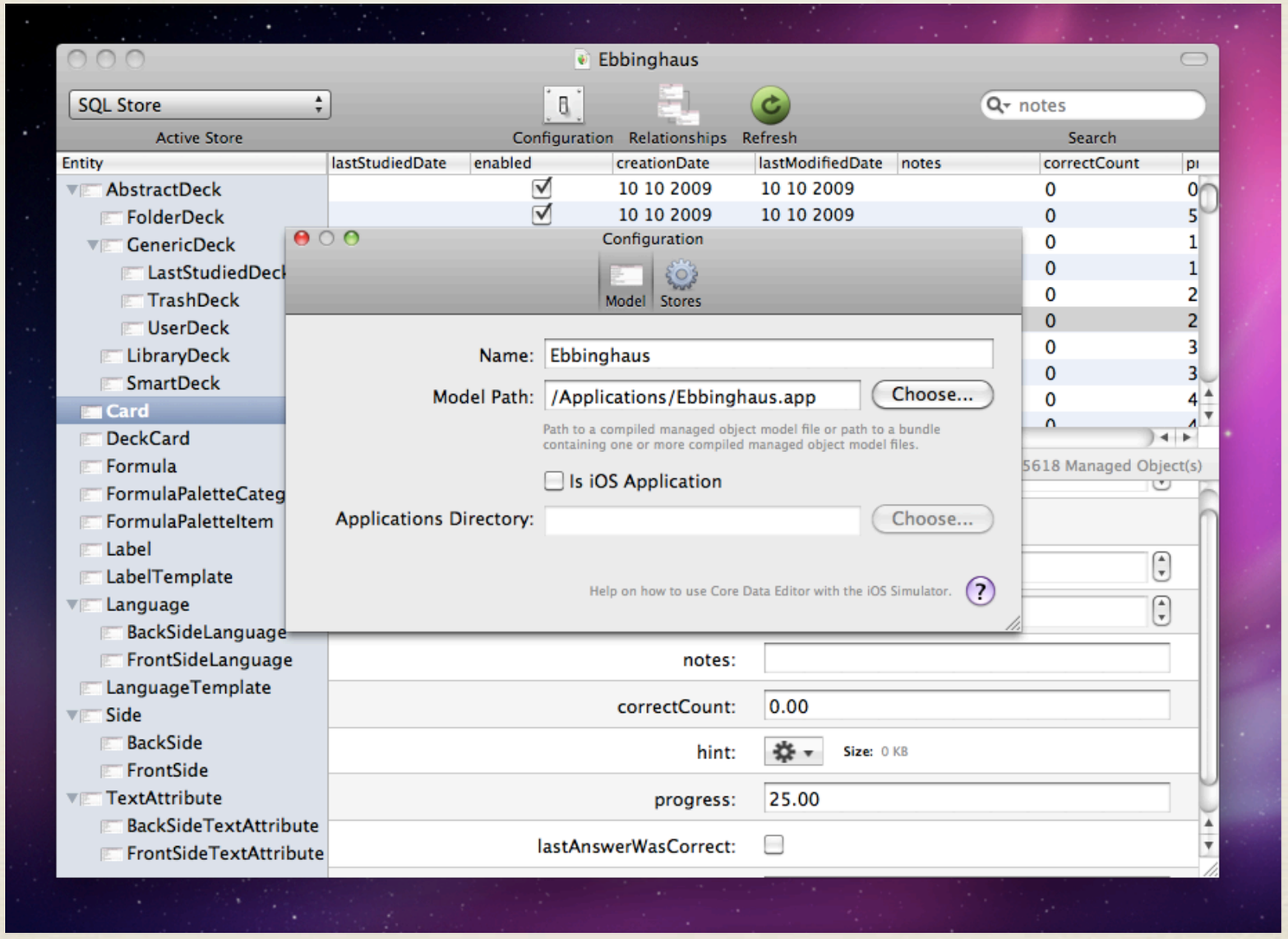
3rd Party Tools

mogenerator

* Manages two classes per entity
  * Machine (_EntityClass)
  * Human (EntityClass)
* This allows you to modify the human class file without fear of being overwritten

github.com/rentzsch/mogenerator

Core Data Editor



christian-kienle.de/CoreDataEditor

DCTCoreData

* Convenient fetching from the managed object context
* A category to handle the ordering of related objects
* Asynchronous tasks and fetching with blocks
* Automated creation of managed objects from an NSDictionary representation

github.com/danielctull/DCTCoreData

DCTCoreData

> “I owe you a drink; You saved me from a task that I thought would take 5 or 6 hours”
>
> – Chris Ross, this morning

github.com/danielctull/DCTCoreData

Daniel Tull

@danielctull

danieltull.co.uk