

iPhone Specialist Lab

L09: Data Persistency

*Prof. Dr. Jan Borchers, Florian Heller, Jonathan Diehl
Media Computing Group, RWTH Aachen University*

2011

<http://hci.rwth-aachen.de/iphone>



Archive and Serialization



Property Lists

- Property lists are serialized collections
 - NSArray or NSDictionary
- Can be read/written directly
- Many APIs for plist handling available
 - PHP, Ruby, Java
- Plists can be included as resources in a project
- Does not work with custom models!
 - Only standard data types



content.plist

Key	Type	Value
▼ Root	Array	(5 items)
▼ Item 0	Dictionary	(3 items)
name	String	John von Neumann
shortVita	String	John von Neumann\n (December 28, 1903 - February 8,
photo	String	vonNeumann.jpg
▼ Item 1	Dictionary	(3 items)
name	String	Alan Turing
shortVita	String	Alan Mathison Turing\n (23 June 1912 - 7 June 1954), v
photo	String	turing.jpg
▼ Item 2	Dictionary	(3 items)
name	String	Douglas Carl Engelbart
shortVita	String	Douglas Carl Engelbart\n (born January 30, 1925) is an .
photo	String	engelbart.jpg
▼ Item 3	Dictionary	(3 items)
name	String	J. C. R. Licklider
shortVita	String	Joseph Carl Robnett Licklider (March 11, 1915 - June 26
photo	String	lick.jpg
▼ Item 4	Dictionary	(3 items)
name	String	Ivan Sutherland
shortVita	String	Ivan Edward Sutherland (born 1938 in Hastings, Nebrask
photo	String	sutherland.jpg



Managing Property Lists

- File Paths
 - `[[NSBundle mainBundle] pathForResource:@"content" ofType:@"plist"]`
 - `NSSearchPathForDirectoriesInDomains(NSDocumentDirectory, NSUserDomainMask, YES)`
- Reading
 - `[[NSArray alloc] initWithContentsOfFile:path]`
- Writing
 - `[array writeToFile:path atomically:NO]`



Core Data

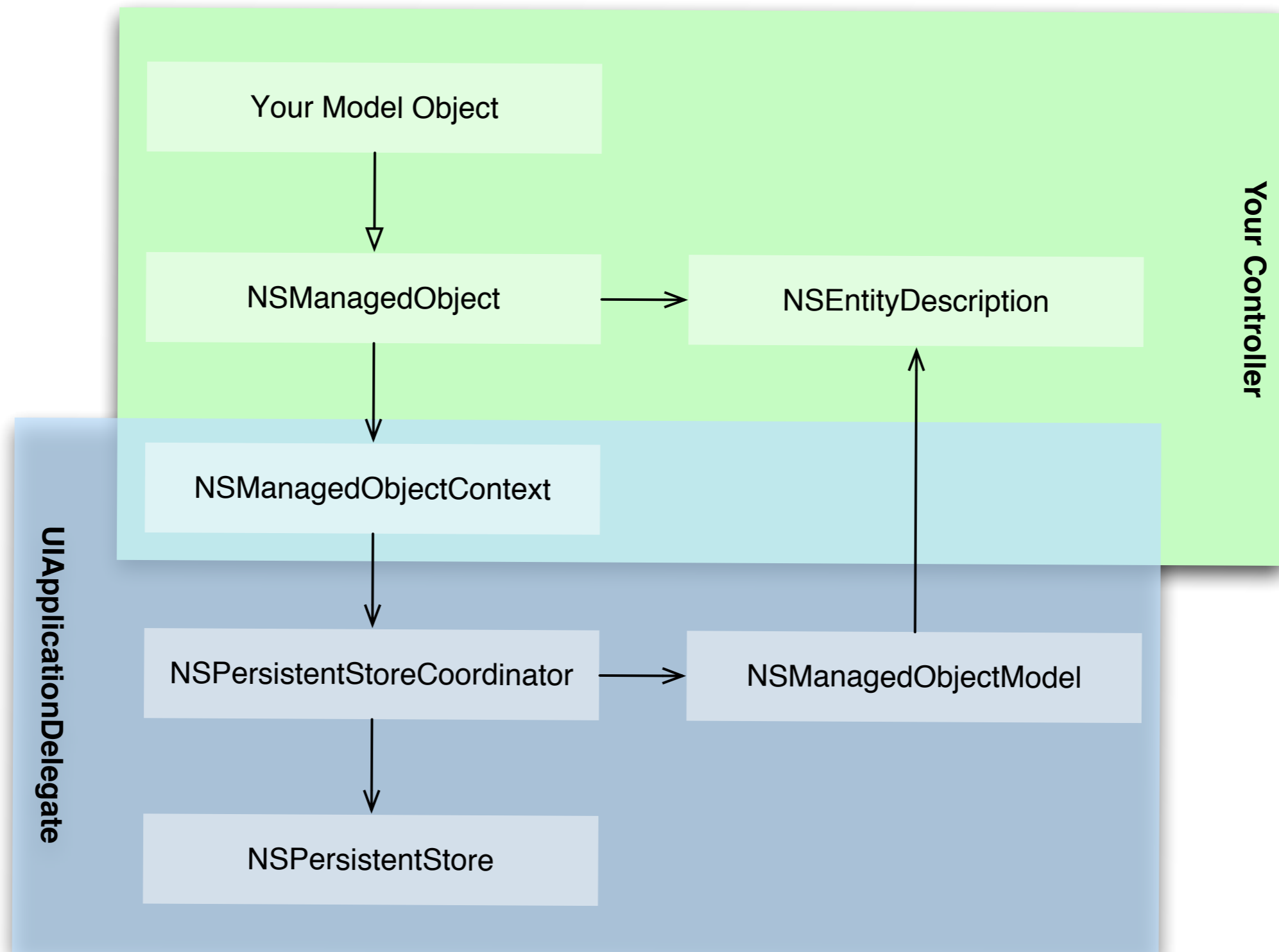


Core Data

- High level relational data storage
- Automatic persistence
- High performance access
- Get undo/redo for free
- Available data stores
 - memory
 - binary file
 - SQLite



Core Data Stack



Recipes.xcdatamodel

Recipes.xcdatamodel Recipe

Entity	Abs	Class	Property	Kind	Type or
Image	<input type="checkbox"/>	NSManage	image	Relationship	Image
Ingredient	<input type="checkbox"/>	Ingredient	ingredients	Relationship	Ingredient
Recipe	<input type="checkbox"/>	Recipe	instructions	Attribute	String
RecipeType	<input type="checkbox"/>	NSManage	name	Attribute	String
			overview	Attribute	String
			prepTime	Attribute	String
			thumbnailImage	Attribute	Transformable
			type	Relationship	RecipeType

Relationship

Name: image

Optional Transient

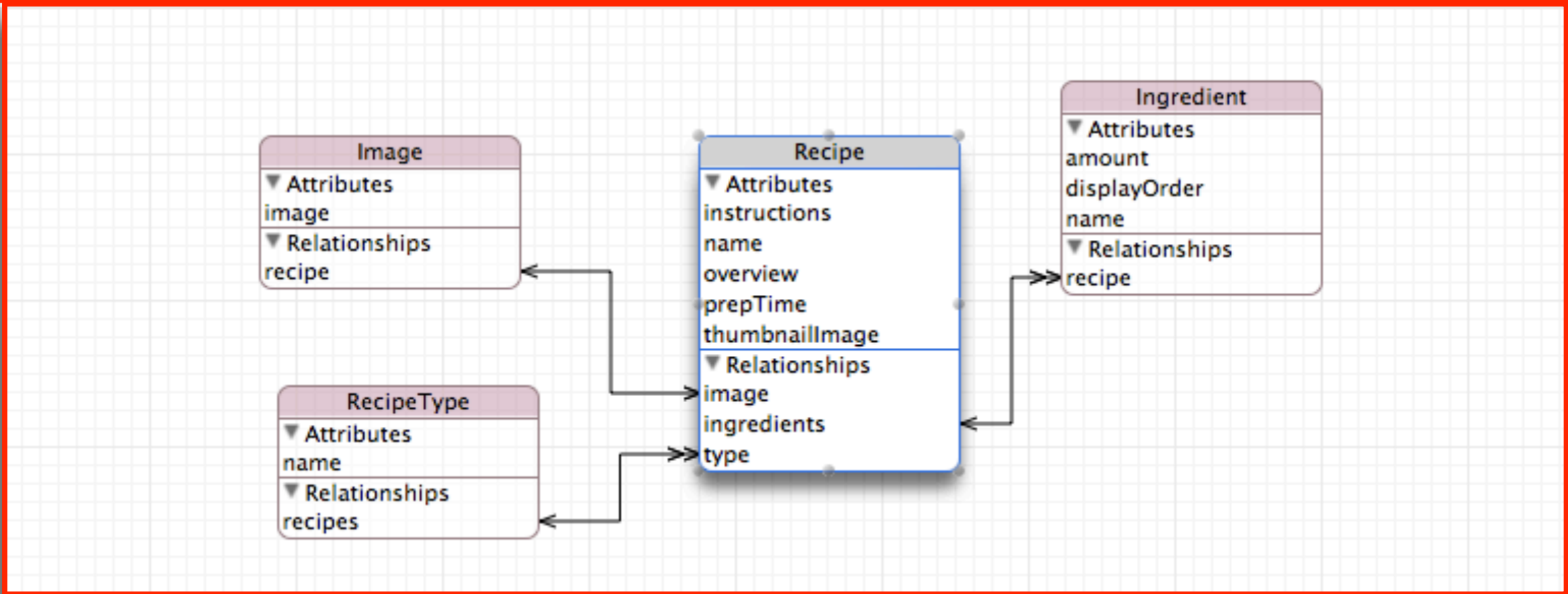
Destination: Image

Inverse: recipe

To-Many Relationship

Min Count: 1 Max Count: 1

Delete Rule: Cascade



Custom Model Class

```
// Recipe.h  
@interface Recipe : NSObject {}  
@property(retain) NSString *title;  
@end
```

```
// Recipe.m  
#import "Recipe.h"  
  
@implementation Recipe  
@dynamic title;  
@end
```



Creating / Deleting Managed Objects

- Create (Insert)

- NSManagedObjectContext
- entity name

```
[NSEntityDescription insertNewObjectForEntityForName:  
[entity name] inManagedObjectContext:context]
```

- Delete

- NSManagedObjectContext
- NSManagedObject

```
[context deleteObject:managedObject];
```

- Save context to make change persistent



Retrieving Managed Objects

- **NSFetchRequest**
 - entity: `NSEntityDescription`
 - sorting criteria: `NSSortDescriptors`
 - search criteria: `NSPredicate`
- **NSFetchedResultsController**
 - tailored to provide data for `UITableViews`
 - requires: `NSFetchRequest`, `NSManagedObjectContext`
 - change tracking via delegate



Fetch Data

```
//Set up the fetch request
NSFetchRequest *request = [[NSFetchRequest alloc] init];
NSEntityDescription *entity = [NSEntityDescription entityForName:@"Recipe"
    inManagedObjectContext:managedObjectContext];
[request setEntity:entity];

//Set up the sort descriptor
NSSortDescriptor *sortDescriptor = [[NSSortDescriptor alloc] initWithKey:@"creationDate"
    ascending:NO];
[request setSortDescriptors: [NSArray arrayWithObject:sortDescriptor]];
[sortDescriptor release];

// Execute the request
NSError *error = nil;
NSArray *fetchResults;
fetchResults = [managedObjectContext executeFetchRequest:request error:&error];
if (fetchResults == nil) {
    // Handle the error.
}
```



Demo



Debugging and Performance Tweaking



Debug Logging

- Log important / problematic operations
- Implement description method:

```
- (NSString *)description {  
    return [NSString stringWithFormat:  
        @"<Person name:%@>", self.name];  
}
```



#	Thread-1-<com.apple.main-thread> ⇅
0	-[AppViewController viewDidLoad]
1	-[UIViewController view]
2	-[AppAppDelegate application:didFinishLaunchingWithOptions:]
3	-[UIApplication _callInitializationDelegatesForURL:payload:suspended:]
4	-[UIApplication _runWithURL:payload:launchOrientation:statusBar...
5	-[UIApplication handleEvent:withNewEvent:]
6	-[UIApplication sendEvent:]

Variable	Value	Summary
▼ Arguments		
▶ self	0x6a26ae0	
_cmd	0x2dd5	
▼ Locals		
▶ objc_super	{...}	
▶ string	0x3030	Hello World!
count	0	

```

AppViewController.m:22 ⇅ -viewDidLoad ⇅
{
    [super viewDidLoad];

    NSString *string = @"Hello World!";
    Unused variable 'string'

    int count = 0;
    int sum = 10;

    float average = sum / count;
    Unused variable 'average'
}
    
```

```

[Session started at 2010-10-25 18:01:31 +0200.]

[Session started at 2010-10-25 18:01:52 +0200.]
GNU gdb 6.3.50-20050815 (Apple version gdb-1472) (Wed Jul 21 10:53:12 UTC 2010)
Copyright 2004 Free Software Foundation, Inc.
GDB is free software, covered by the GNU General Public License, and you are
welcome to change it and/or distribute copies of it under certain conditions.
Type "show copying" to see the conditions.
There is absolutely no warranty for GDB. Type "show warranty" for details.
This GDB was configured as "x86_64-apple-darwin".sharedlibrary apply-load-rules all
Attaching to process 6129.
Program received signal: "EXC_ARITHMETIC".
(gdb)
    
```

Console

- Dump variables
 - po for objects, p for C-types
 - perform method calls
- Examine the call stack and source files
- Much more...



Zombies

