

# Разработка приложений для OS X

## Лекция 7

Undo для текста. Архивация. Сохранение/загрузка  
файлов.

Hexlet University, 2012

# Новый курс

## Финансовая инженерия



Дмитрий Нестерук  
<http://nesteruk.wordpress.com/>

- Принципы функционирования финансовых рынков – его участники, механизмы взаимодействия, инструменты и технологии работы.
- Финансовая математика – методы применения математики в финансовых расчетах.
- Программирование – как писать программное обеспечение, которое умеет работать на финансовых рынках.
- [MATLAB](#) или бесплатный аналог [GNU Octave](#)
- Microsoft Excel полезен из общих соображений, т.к. это самый простой и понятный инструмент для работы с данными.
- Visual Studio любой версии с поддержкой F#, или же F# компилятор отдельно.

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS \frac{\partial V}{\partial S} - rV = 0$$

# Новости 2

Hexlet Hub

# Новость 2

Hexlet Hub

**u.hexlet.org**

Лекции

Тесты

Домашние работы

Экзамены

+

Простые обсуждения  
учебного процесса

**hub.hexlet.org**

Сообщество

Блоги, статьи

Большие обсуждения

Обмен опытом

+

Знакомства, новые  
команды, проекты :)

конкурсы с инвайтами на хабр и дёрти

# Новости 3

## Недельные проекты

- Небольшие проекты для самостоятельной работы
- Не оцениваются
- Совместная работа, взаимопомощь — в блоге курса на **хабе**
- Можете предлагать свои



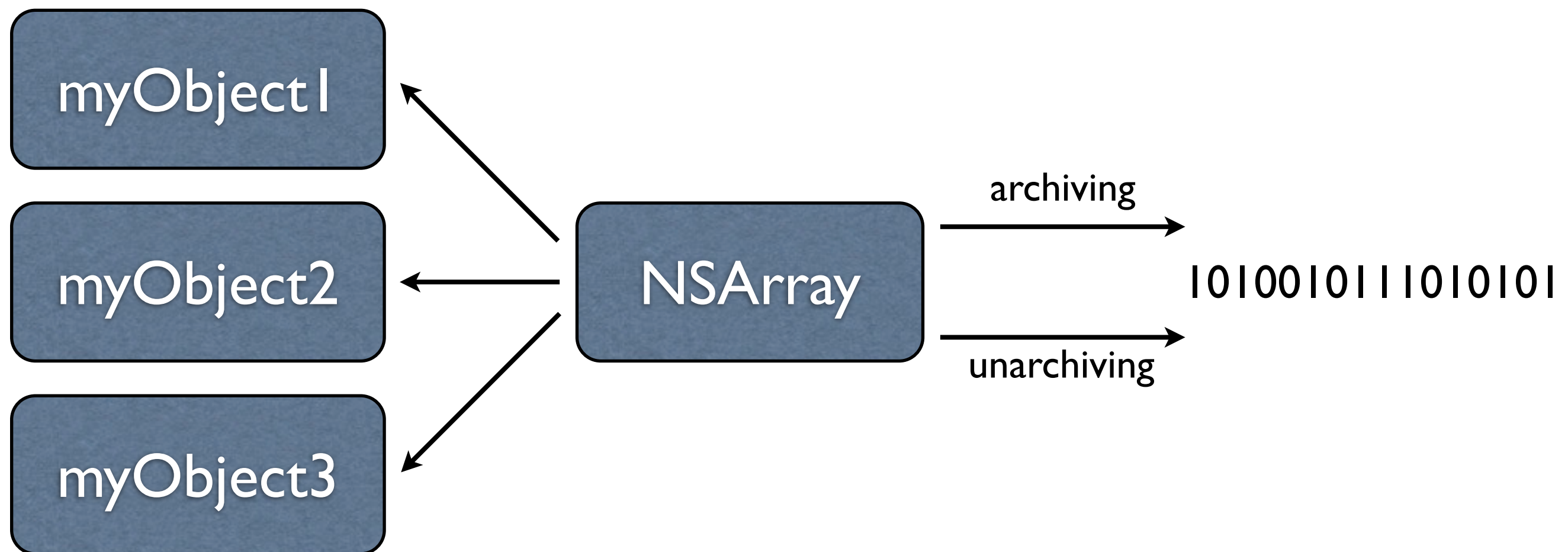
# Undo для текста

- KVO!
  - Нужно сообщать документу о каждом изменении текста
- `(void)addObserver:(NSObject *)observer  
forKeyPath:(NSString *)keyPath  
options:(NSKeyValueObservingOptions)options  
context:(void *)context;`
  - `(void)observeValueForKeyPath:(NSString *)keyPath  
ofObject:(id)object  
change:(NSDictionary *)change  
context:(void *)context;`



# Архивация

- Представление объектов в виде потока байтов
- В других языках – “сериализация”



# Протокол NSCoder

- `(id)initWithCoder:(NSCoder *)coder;`
  - `(void)encodeWithCoder:(NSCoder *)coder;`
- 
- `init` – прочитать данные из кодера, записать в переменные
  - `encode` – прочитать из переменных, отправить кодеру



# Класс NSCoder

- (`void`)encodeObject:(`id`)anObject forKey:(`NSString *`)aKey
- Абстрактный класс
- Отправляет anObject кодеру и связывает его с aKey
- Произойдет вызов encodeWithCoder объекта anObject

# Encoding primitives

- `(void)encodeBool:(BOOL)boolv forKey:(NSString *)key`
- `(void)encodeDouble:(double)realv forKey:(NSString *)key`
- `(void)encodeFloat:(float)realv forKey:(NSString *)key`
- `(void)encodeInt:(int)intv forKey:(NSString *)key`

# Encoding Person

- Давайте закодируем нашего Person



# Decoding

- (`id`) `decodeObjectForKey:(NSString *)aKey`
- (`BOOL`) `decodeBoolForKey:(NSString *)key`
- (`double`) `decodeDoubleForKey:(NSString *)key`
- (`float`) `decodeFloatForKey:(NSString *)key`
- (`int`) `decodeIntForKey:(NSString *)key`

# Decoding a Person

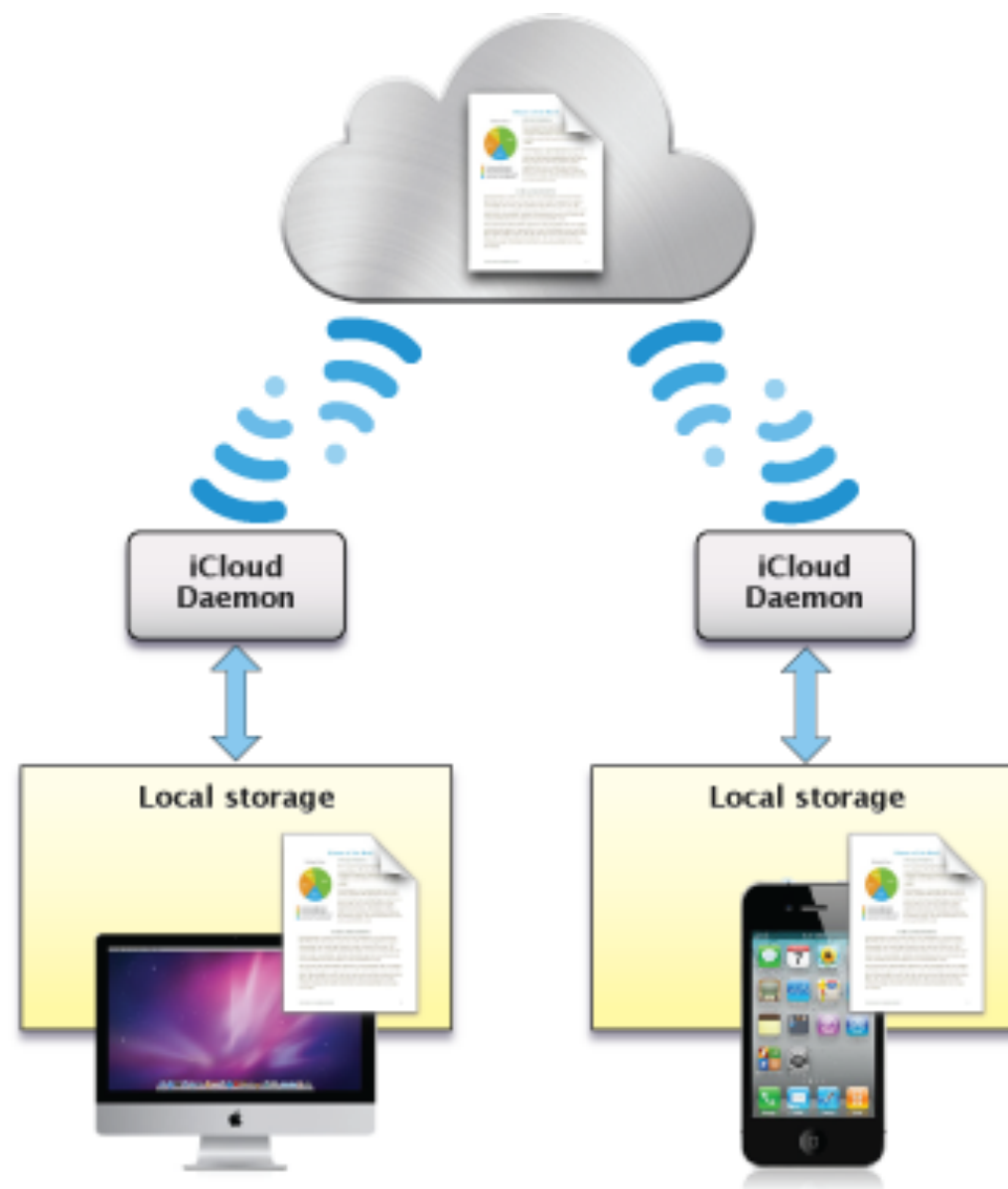
- А теперь раскодируем Person



# Autosave

- Начиная с OS X 10.7 программы могут сохранять файлы автоматически
- Интервал между сохранениями можно изменять  
`setAutosavingDelay:`
- Сохранение может быть асинхронным  
`canAsynchronouslyWriteToURL:ofType:forSaveOperation:`

# Поддержка iCloud



# Хранение в iCloud

- Хранение документов (на компьютере синхронизируется сразу, на iOS – по запросу)
- Хранение данных (key – value) (везде синхронизируется сразу)
- Передаются только изменения



# Сохранение файла

- (NSData \*)dataOfType:(NSString \*)aType error:(NSError \*\*)e
- (NSFileWrapper \*)fileWrapperOfType:(NSString \*)aType  
error:(NSError \*\*)e
- (BOOL)writeToURL:(NSURL \*)absoluteURL ofType:(NSString \*)typeName  
error:(NSError \*\*)outError;

# Загрузка файла

- (BOOL)readFromData:(NSData \*)data ofType:(NSString \*)typeName error:(NSError \*\*)outError
- (BOOL)readFromFileWrapper:(NSFileWrapper \*)fileWrapper ofType:(NSString \*)typeName error:(NSError \*\*)outError;
  - (BOOL)readFromURL:(NSURL \*)absoluteURL ofType:(NSString \*)typeName error:(NSError \*\*)outError;

# Сохранение

- Создадим объект класса NSData
- Используем NSKeyedArchiver, его метод:

+ (NSData \*)archivedDataWithRootObject:(id)rootObject

Он архивирует объект байт-буфер объекта NSData

# Загрузка

- Используем метод `NSKeyedUnarchiver`'а

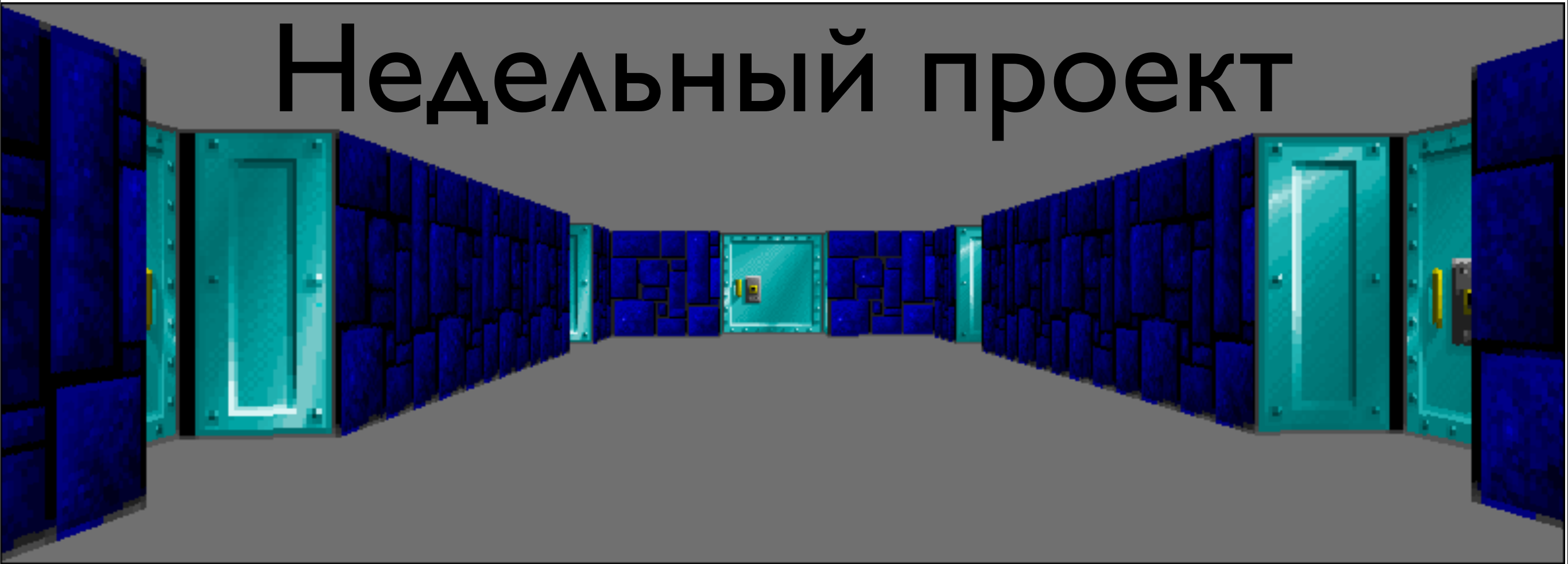
+ `(id)unarchiveObjectWithData:(NSData *)data`



# Сохранение и загрузка людей



# Недельный проект



# Недельный проект



Добавить текстовый undo посимвольно

# Недельный проект



Добавить текстовый undo посимвольно



Сохранение выделенных людей  
Если ничего не выделено — сохранять все



# Недельный проект



Добавить текстовый undo посимвольно



Сохранение выделенных людей  
Если ничего не выделено — сохранять все



Загрузка с дополнением в текущий документ  
При дублировании имен запрашивать пользователя о замене или отмене загрузки дублирующей записи

# Недельный проект



Добавить текстовый undo посимвольно



Сохранение выделенных людей  
Если ничего не выделено — сохранять все



Загрузка с дополнением в текущий документ  
При дублировании имен запрашивать пользователя о замене или отмене загрузки дублирующей записи



iOS-клиент с синхронизацией через iCloud