



**Ecole d'ingénieurs et d'architectes de Fribourg**  
**Hochschule für Technik und Architektur Freiburg**

# Mobile development 3

## Report TD 3 :

Jonathan Stoppani et Elias Medawar

Version: March 15, 2011

## 1 KVC

### 1.1 Source code for the address

```
// Implement viewDidLoad to do additional setup after loading the view, typically from a nib.
- (void)viewDidLoad {
    company = [[Company alloc] init];
    Employee *employee = [company employee];
    Address *address = [employee address];

    // Using getter
    NSLog(@"Using getter %@", [address street]);

    // Using the "." method
    company.employee.address.street = @"Av. de la gare 12";
    NSLog(@"Using the . method %@", company.employee.address.street);

    // Using KVC
    [ address setValue:@"Rue du midi 12" forKey:@"street"];
    NSLog(@"Using KVC %@", [ address valueForKey:@"street" ]);

    // From employee using KVC path
    [ employee setValue:@"Rue de la paix 11" forKeyPath:@"address.street"];
    NSLog(@"From employee using KVC path %@",
        [employee valueForKeyPath:@"address.street" ]);

    // From company using KVC path
    [ company setValue:@"Rue du midi 11" forKeyPath:@"employee.address.street"];
    NSLog(@"From employee using KVC path %@",
        [ company valueForKeyPath:@"employee.address.street" ]);

    NSString* nubmber = [NSString stringWithFormat:@"%d", [address number]];
    [NSString stringWithFormat:@"%d", address.number];

    [super viewDidLoad];
}
```

### 1.2 Output

```
-----
2011-03-14 22:28:51.195 TD3[14720:207] Using getter Bd de Pérolles
2011-03-14 22:28:51.197 TD3[14720:207] Using the . method Av. de la gare 12
2011-03-14 22:28:51.197 TD3[14720:207] Using KVC Rue du midi 12
2011-03-14 22:28:51.198 TD3[14720:207] From employee using KVC path Rue de la paix 11
2011-03-14 22:28:51.198 TD3[14720:207] From employee using KVC path Rue du midi 11
-----
```

## 2 KVO

### 2.1 M file of the address with the int and numberDouble

```
@implementation Address

@synthesize street;
@synthesize number;

- (id) init
{
    self = [super init];
    self.street = @"Bd de Pérolles";
    self.number = 80;
    return self;
}

-(void) setNumber:(int) n
{
    // Will raise the notification of the modification of numberDouble
    [self willChangeValueForKey:@"numeroDouble"];
    number = n;
    [self didChangeValueForKey:@"numeroDouble"];
}

-(int) numberDouble
{
    return number*2;
}

-(void) dealloc
{
    [street release];
    // We dont need to relase number it's not an object!!/
    [super dealloc];
}

@end
```

The numberDouble is a calculated value so we have to raise “manually “ the notifications.

### 2.2 H file of the address with the int and numberDouble

```
#import <Foundation/Foundation.h>

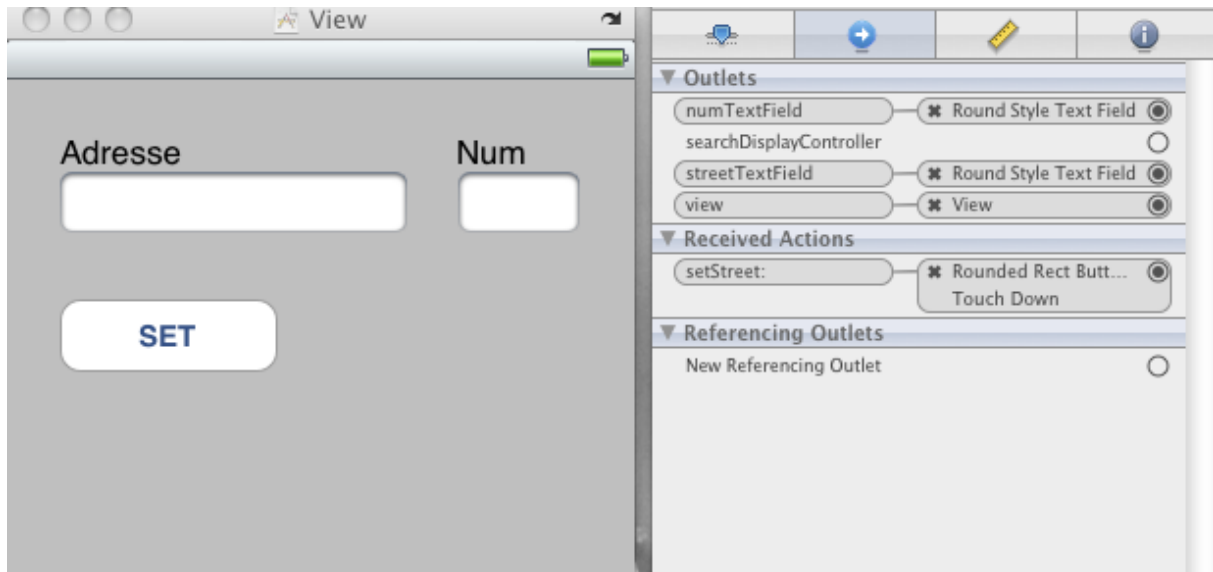
@interface Address : NSObject
{
    NSString *street;
    int number;
}

@property (nonatomic, retain) NSString *street;
@property (nonatomic) int number;

// numberDouble is not a propertey so we need to implement getter and setter
-(int) numberDouble;

@end
```

## 2.3 UI with 2 text-field and a button



Here we see the creation and links in **Interface builder**

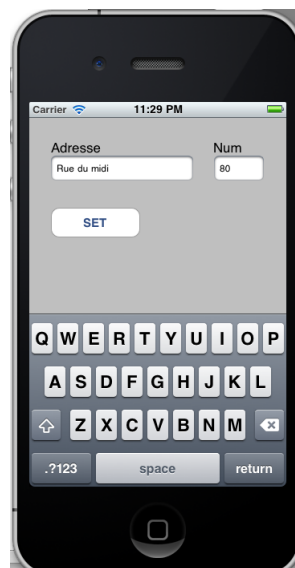
## 2.4 Setting value of text fields

```
- (void)viewDidLoad {  
    ...  
  
    //Transform int to string  
    NSString* nubmber = [NSString stringWithFormat:@"%d", [address number]];  
    streetTextField.text = address.street; // Display values  
    numTextField.text = nubmber;  
  
    [super viewDidLoad];  
}
```

At the load of the application we set the value that will be displayed

The conversion from int to string was found on this website: <http://stackoverflow.com/questions/169925/how-to-do-string-conversions-in-objective-c>

## 2.5 Result after loading



## 2.6 Registering and receiving the KVO notification in the employee class

```
#import "Employee.h"
#import "Address.h"

@implementation Employee

@synthesize address;

-(id) init
{
    self = [super init];
    address = [[Address alloc] init];

    // Register the KVO notification
    [address addObserver:self forKeyPath:@"street" options:0 context:NULL];
    [address addObserver:self forKeyPath:@"number" options:0 context:NULL];
    [address addObserver:self forKeyPath:@"numberDouble" options:0 context:NULL];

    return self;
}

// This method will be called on changes
- (void)observeValueForKeyPath:(NSString *)keyPath ofObject:(id)object
    change:(NSDictionary *)change context:(void *)context
{
    if([keyPath isEqualToString:@"number"])
        NSLog(@"The variable number changed :%@", [object valueForKeyPath:keyPath]);
    else if([keyPath isEqualToString:@"street"])
        NSLog(@"The street value changed :%@", [object valueForKeyPath:keyPath]);
    else if([keyPath isEqualToString:@"numberDouble"])
        NSLog(@"The numberDouble value changed :%@", [object valueForKeyPath:keyPath]);
}

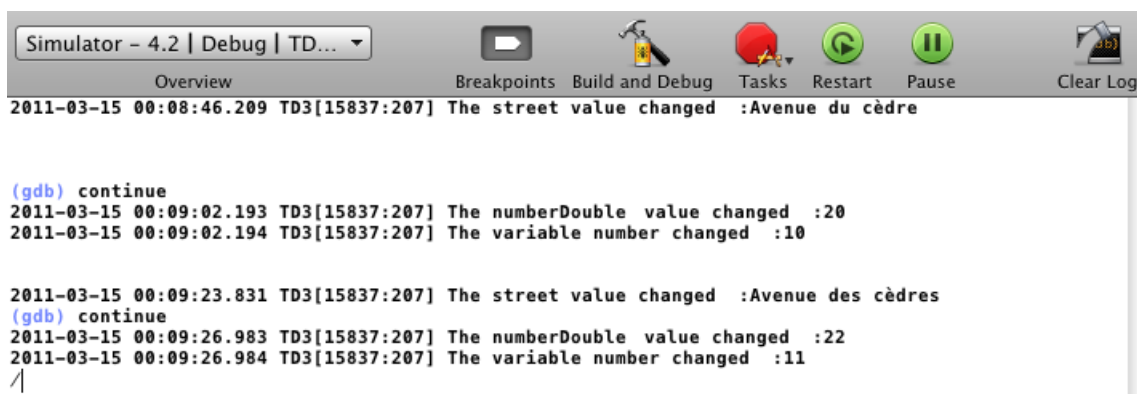
-(void) dealloc
{
    [address removeObserver:self forKeyPath:@"street"];
    [address removeObserver:self forKeyPath:@"numero"];
    [address release];

    [super dealloc];
}

@end
```

We are responsible to unsubscribe the notification in the dealloc method.

## 2.7 Result of the execution



The screenshot shows the Xcode simulator interface. At the top, there's a toolbar with buttons for Overview, Breakpoints, Build and Debug, Tasks, Restart, Pause, and Clear Log. Below the toolbar, the console displays the following log messages:

```
2011-03-15 00:08:46.209 TD3[15837:207] The street value changed :Avenue du cèdre

(gdb) continue
2011-03-15 00:09:02.193 TD3[15837:207] The numberDouble value changed :20
2011-03-15 00:09:02.194 TD3[15837:207] The variable number changed :10

2011-03-15 00:09:23.831 TD3[15837:207] The street value changed :Avenue des cèdres
(gdb) continue
2011-03-15 00:09:26.983 TD3[15837:207] The numberDouble value changed :22
2011-03-15 00:09:26.984 TD3[15837:207] The variable number changed :11
^
```

1. For this tests , we change at first the street value then press set
2. The we change the number value and press set
3. Finally we change both value and press set

We tried to call just the `didChangeValueForKey` in the `setNumber` method of the address class. But the notification doesn't work without both call `will` and `didChangeValueForKey`.

### 3 Notifications

#### 3.1 M file of the company class

```
#import "Company.h"
#import "Employee.h"

@implementation Company

@synthesize employee;

- (id) init
{
    self = [super init];
    employee = [[Employee alloc] init];
    // ADD the observer to the NotificationCenter
    [[NSNotificationCenter defaultCenter] addObserver:self selector: @selector(streetChanged:)
                                                name: @"streetChanged" object: nil];

    return self;
}

// The callback method of the notification
-(void) streetChanged:(NSNotification *)notification
{
    NSLog(@"The streetChanged notification was raised.");
}

-(void) dealloc
{
    [employee release];
    [super dealloc];
}

@end
```

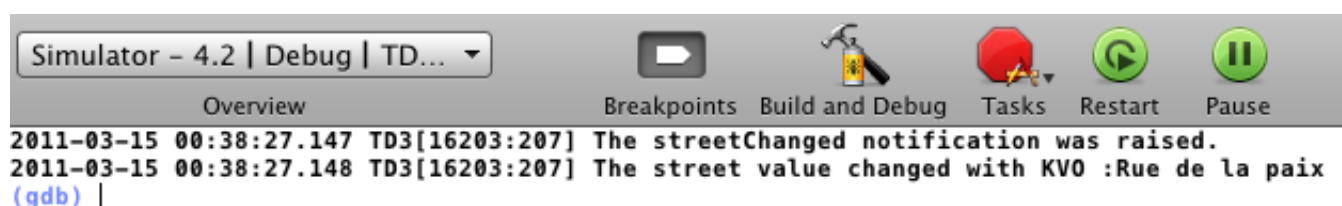
Here we have added an observer and the callback function to receive the specific information.

#### 3.2 Raising notification

```
-(void) setStreet:(NSString*) s
{
    if (![s isEqualToString: street]){
        // Will raise the notification of the modification of numberDouble
        street = s;
        [[NSNotificationCenter defaultCenter] postNotificationName: @"streetChanged" object:nil];
    }
}
```

In the address class when we change the value of the street we have to post a notification

#### 3.3 Result



The result in the console.



## 4 Conclusion

KVC is very powerful but we have had some problem with the spelling of variable name. And this errors was not detected at the compilation time but at the run time.

All the sources and documentation is available on gitHub at this address:

<https://github.com/eia-fr/mobileDev/tree/master/TD3>