

# Cocoa Design Patterns

5 November 2005, Utrecht  
© Patrick Machiels  
[patrick@hieper.nl](mailto:patrick@hieper.nl)

## Definitie en oorsprong

### // omschrijving

- Design Patterns zijn standaardoplossingen voor vaak voorkomende problemen in softwareontwikkeling
- Hergebruik van een manier van aanpak. Een aantal Design Patterns is ingebakken in het ontwerp van Cocoa.

### // 'het' boek

- 'Design Patterns' boek van Gamma / Helm / Johnson / Vlissides

# Cocoa Design Patterns

// programma structuur

- Model - View - Controller

// view - controller communicatie

- Target - Action

// '1 to 1' communicatie

- Delegation

// 'many to many' communicatie

- Notification

# Model – View – Controller

// beschrijving

In Cocoa programma's worden de Model, View en Controller onderdelen zoveel mogelijk gescheiden

- 'Model' is de data, of de kernfunctie.
- 'View' is de user interface.
- 'Controller' is de logica die 'View' en 'Model' verbindt.

# Target – Action (1)

// beschrijving

Alle Cocoa bedieningselementen (NSControl sub-class), zoals knoppen en menu's, kunnen als ze geactiveerd worden een willekeurige methode in een willekeurig object aanroepen.

- 'Target' is een willekeurig Cocoa object (van het type 'id').
- 'Action' is een willekeurige methode signatuur van het type 'SEL'.

## Target – Action (2)

Een Action methode heeft de vorm:

```
IBAction actionNaam:(id)sender
```

IBAction is gelijk aan void, en is een hulpmiddel voor Interface Builder

Target en Action van een NSControl tijdens het uitvoeren van het programma eenvoudig worden veranderd. Als de Target `nil` is, wordt de 'first responder' het doelobject. ('nil targeted action')

# Delegation

// beschrijving

Cocoa objecten kunnen 'delegate methoden' definiëren die bij bepaalde gebeurtenissen worden aangeroepen. Een 'delegate object' kan die methoden implementeren om dan code uit te voeren, maar hoeft dat niet.

Een Cocoa object kan slechts 1 delegate object hebben.

# Notification

// beschrijving

Bij specifieke gebeurtenissen kan een Cocoa object een 'Notificatie' sturen aan alle daarin geïnteresseerde objecten.

- Zender verstuurt de notificatie aan NSNotificationCenter ('postkantoor')
- Objecten die een bepaalde notificatie willen ontvangen melden zich eerst aan bij NSNotificationCenter
- Meerdere zenders en ontvangers mogelijk