

The Observer Pattern

Schéma de conception pour la Communication événementielle

Noury Bouraqadi
<http://car.mines-douai.fr/noury>

Option ISIC
 Ecole des Mines de Douai

2

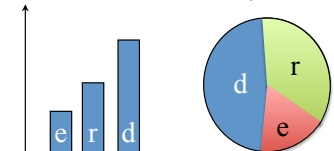
Fondement

- **Objectif :**
 - interaction entre objets qui ne se connaissent pas a priori
 - communication "multi-cast" : 1 émetteur – n destinataires
 - Connexion-Déconnexion en cours d'exécution

- **Exemple :**
 - Répartition du temps d'un enseignant-chercheur
 - enseignement, recherche, divers (contrats, encadrement, ...)
 - Différents affichages (éventuellement simultanés)

- Texte
- Barres
- Camembert

e = 20 %
 r = 35 %
 d = 45 %



Patron de conception "Observateur"

- **cf. Livre "Design Patterns"**
- **Egalement connu sous le nom : "publish-subscribe"**
 - Inventé dans Smalltalk : Model-View-Controller (MVC)
- **dépendance "un vers plusieurs"**
 - un objet change \Rightarrow plusieurs objets se mettent à jour
 - Objet qui change : **sujet**
 - Objets qui se mettent à jour : **observateurs**
- **Le sujet ne connais pas a priori ses observateurs**
 - Les observateurs s'enregistrent en cours d'exécution
 - Il peuvent être "déconnectés" en cours d'exécution

3

Généralisation : utilisation des événements

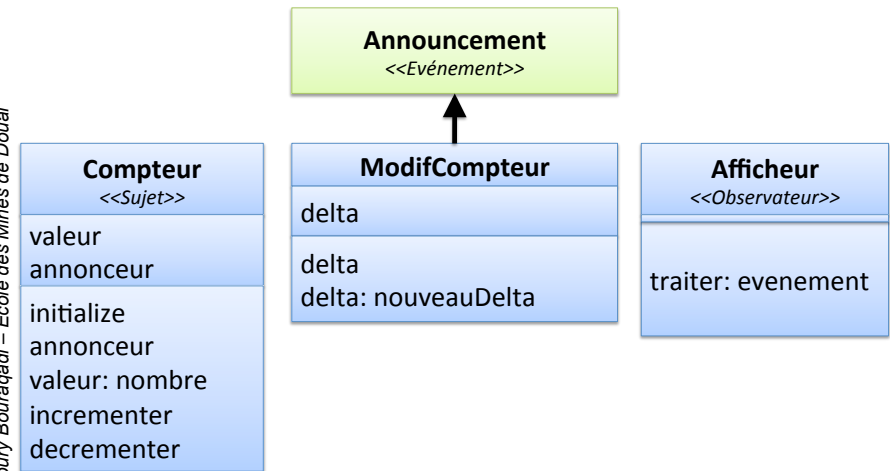
- **Evénement =**
 - Objet créé à chaque "changement" dans un sujet
 - Contient les infos sur le contexte du changement
 - sujet, données changées, ...
- **Différents classes d'événements**
 - 1 classe par "famille" d'événements
 - 1 famille = 1 cause de changement
- **Sujet = source d'événements**
- **Observateur = puits d'événements**

4

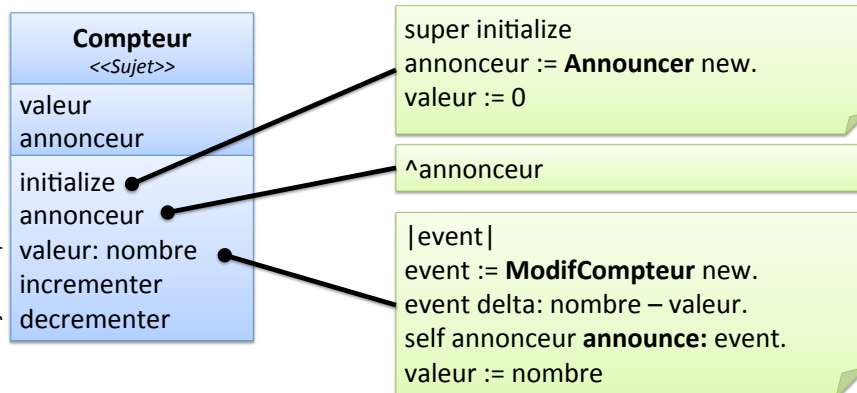
La bibliothèque *Announcement*

- **Classes d'événements** : sous-classes de **Announcement**
- **La gestion est faite par une instance de *Announcer***
- **Le modèle** :
 1. crée des événements ;
 2. les envoie via *Announcer*
- **Les observateurs** :
 - s'inscrivent via les messages
 - **Announcer>>on:do:**
 - **Announcer>>on:send:to:**

Exemple avec Announcement



Exemple avec Announcement



Exemple avec Announcement

"Lien sujet-observateur"
 compteur := Compteur new.
 afficheur := Afficheur new.
 compteur annonceur
on: ModifCompteur send: #traiter: to: afficheur