

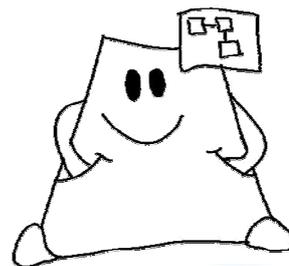
GARANTIE DES CONFORMITÉS PAR REJEU DE PATTERNS

PRÉSENTATION DES PERSONNAGES :



LE DÉVELOPPEUR

Définit un modèle devant vérifier plusieurs contraintes.



LES PATTERNS CONFORMANTS

dont l'application garantit la mise en place des conformités associées

Expression structurelle des requis et contraintes liés à une problématique à travers des modèles et des transformations de modèles.

SCÉNARIO DE REJEU :



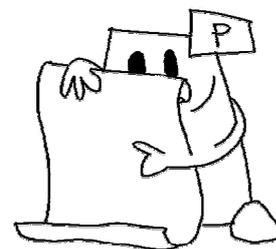
L'évolution de ce modèle va prendre en compte :

- L'authentification des connexions au serveur par pattern
- L'ajout d'une relation d'un client au serveur par action utilisateur

Le développeur applique le pattern P pour l'authentification.

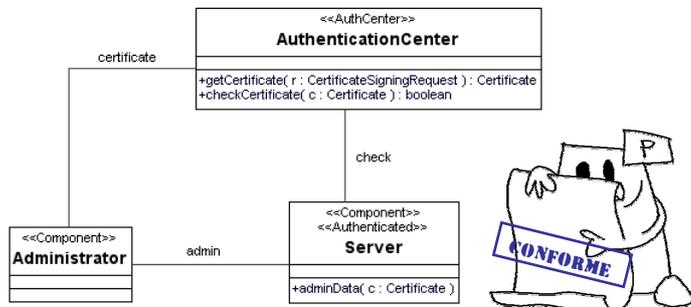
Conformité de P :

- Toutes les méthodes de Server prennent en paramètre d'entrée un Certificate.
- Toutes les classes reliées au Server doivent aussi être reliées au Centre d'Authentification.

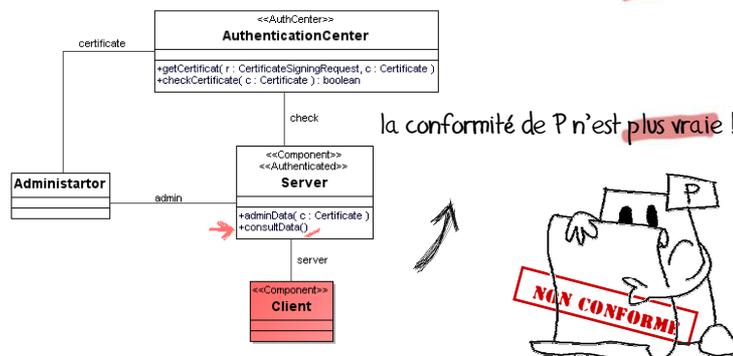


P est défini par des transformations de modèles (Prolog).

Un nouveau modèle vérifiant la conformité de P est généré par application des transformations.

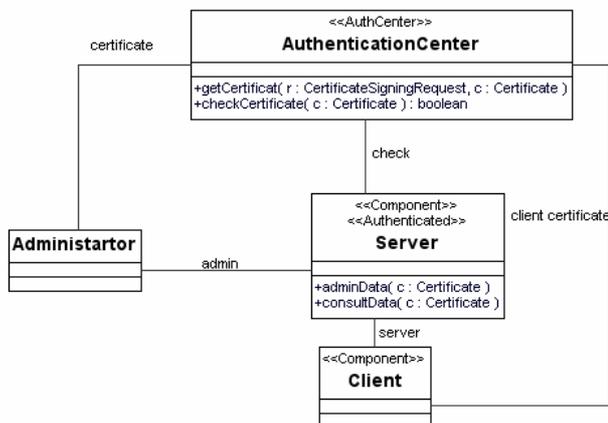


Le développeur met en place la relation Client/Server, mais ...



la conformité de P n'est plus vraie !

Le pattern P est alors rejoué : les transformations sont ré-appliquées et leurs actions réparent la conformité.



CONCLUSION :

- Les patterns conformants sont définis par des métamodèles et des transformations de modèles.
- L'application des transformations établit ou répare les conformités.
- L'application d'un pattern ne fait rien si ses conformités sont vraies (Idempotence).

