

Représentation des vues d'architecture avec UML

Pierre-Alain Muller

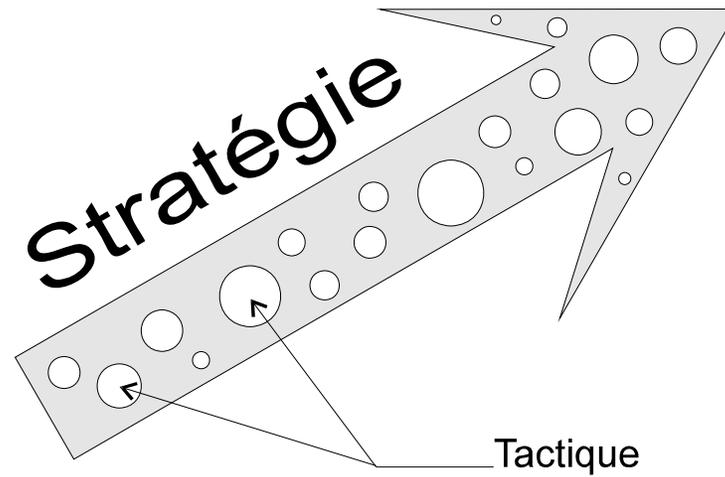
ENSISA

pa.muller@uha.fr

03.89.33.69.65

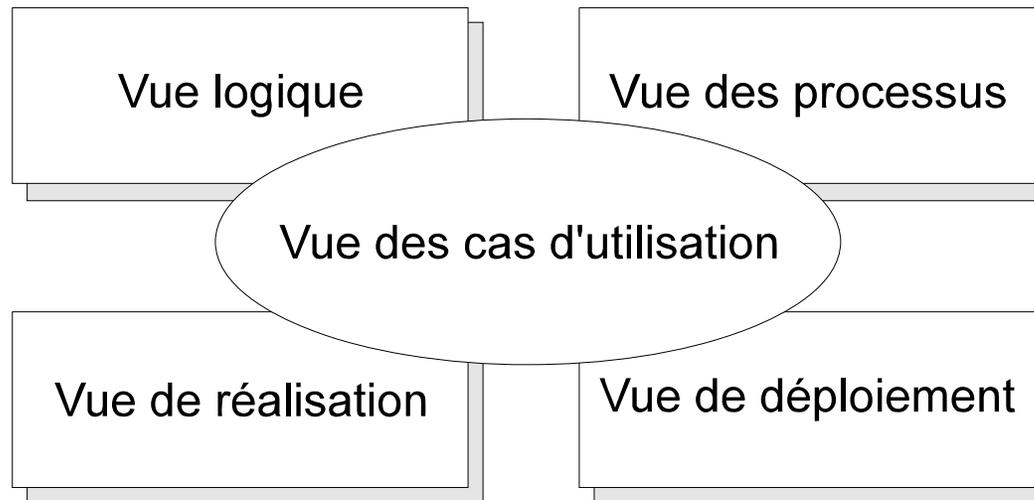
Architecture logicielle

- Architecture =
Eléments + Formes + Motivations
- Architecture = Stratégie + Tactique



La vision de l'architecte

- Il n'existe pas une seule manière de regarder un système
 - Philippe Kruchten, le modèle 4 + 1 vues, IEEE Software, Nov. 95



Le modèle 4 + 1 vues

- La vue logique
- La vue de réalisation
- La vue des processus
- La vue de déploiement
- La vue des cas d'utilisation



La vue logique

- Aspects statiques et dynamiques
- Les éléments
 - Les objets
 - Les classes
 - Les collaborations
 - Les interactions
 - Les paquetages <<Catégorie>>



La vue de réalisation

- Organisation des modules dans l'environnement de développement
- Les éléments
 - Les modules
 - Les sous-programmes
 - Les tâches (en tant qu'unités de programme, comme en Ada)
 - Les paquetages << sous-système >>



La vue des processus

- Décomposition en flots d'exécution et synchronisation entre ces flots
- Les éléments
 - Les tâches
 - Les threads
 - Les processus
 - Les interactions



La vue de déploiement

- Les ressources matérielles et l'implantation du logiciel dans ces ressources
- Les éléments
 - Les noeuds
 - Les modules
 - Les programmes principaux



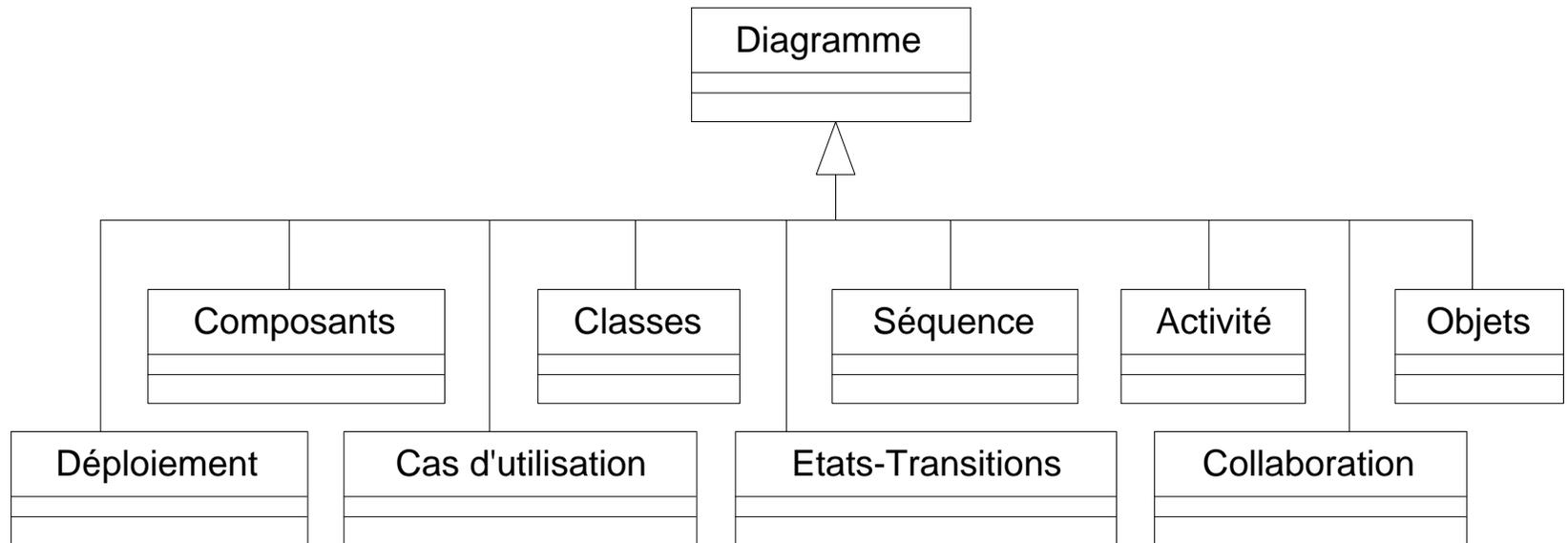
La vue des cas d'utilisation

- La colle entre les autres vues
- Les éléments
 - Les acteurs
 - Les cas d'utilisation
 - Les classes
 - Les collaborations

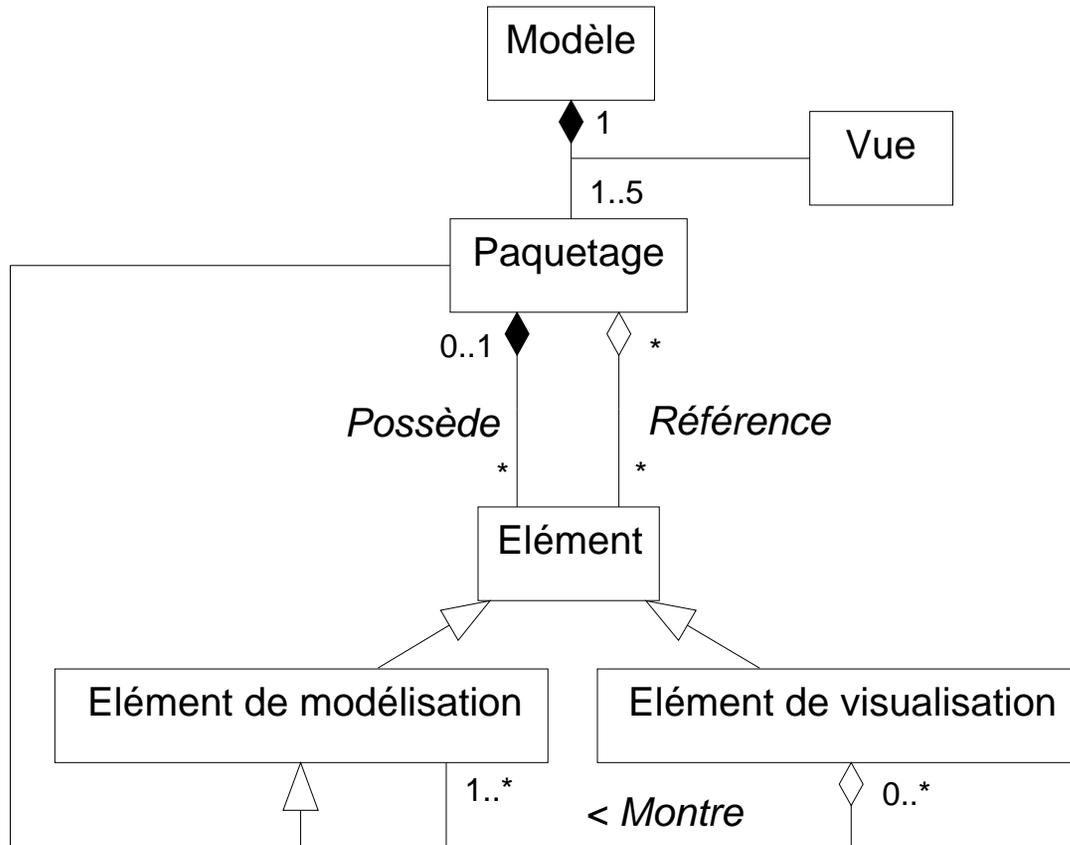


Articulation des diagrammes

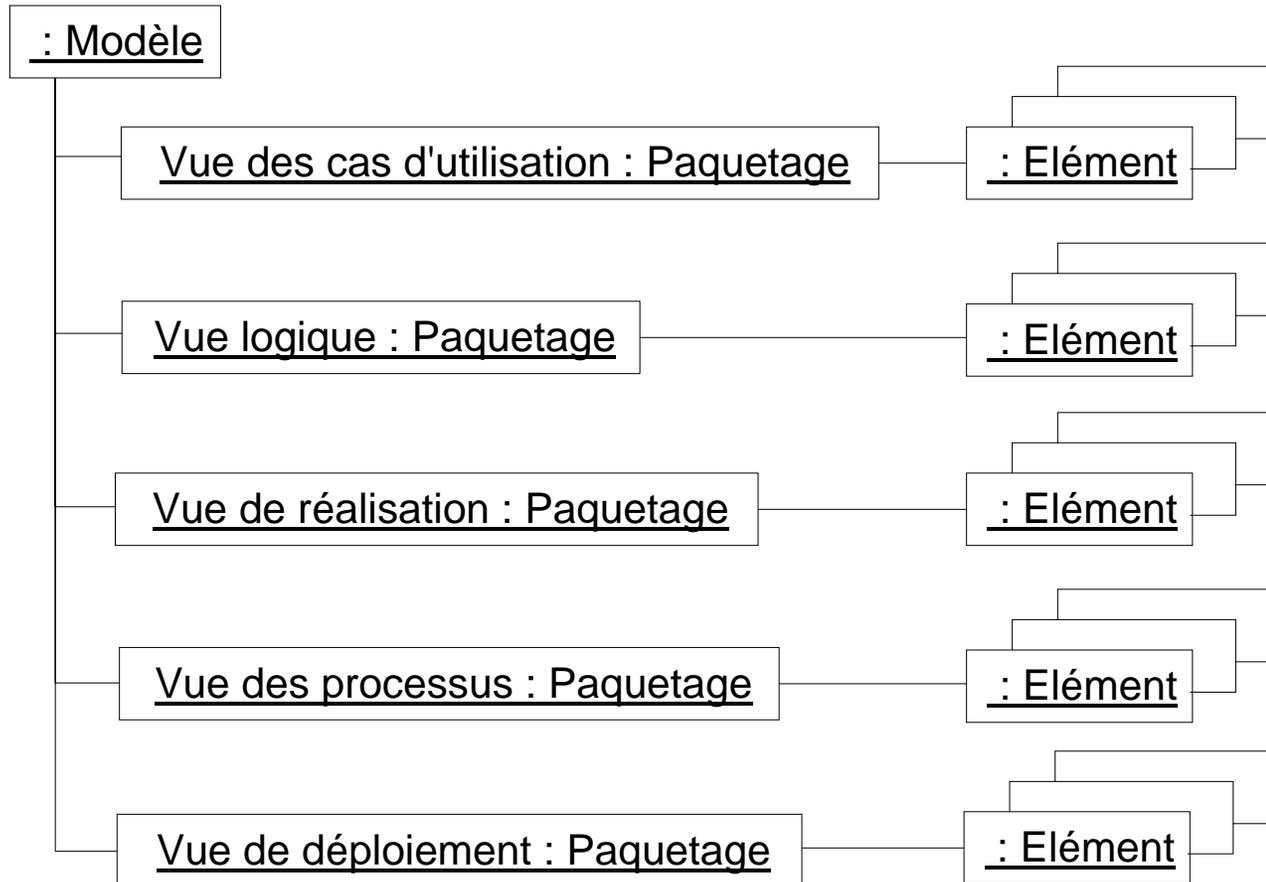
- UML propose 9 types de diagrammes



Organisation des modèles

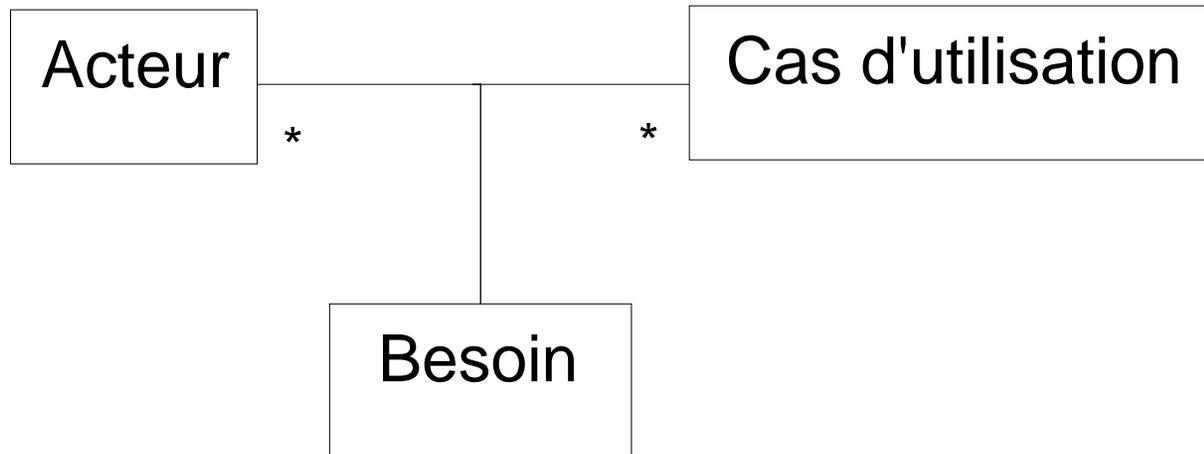


Structuration des modèles



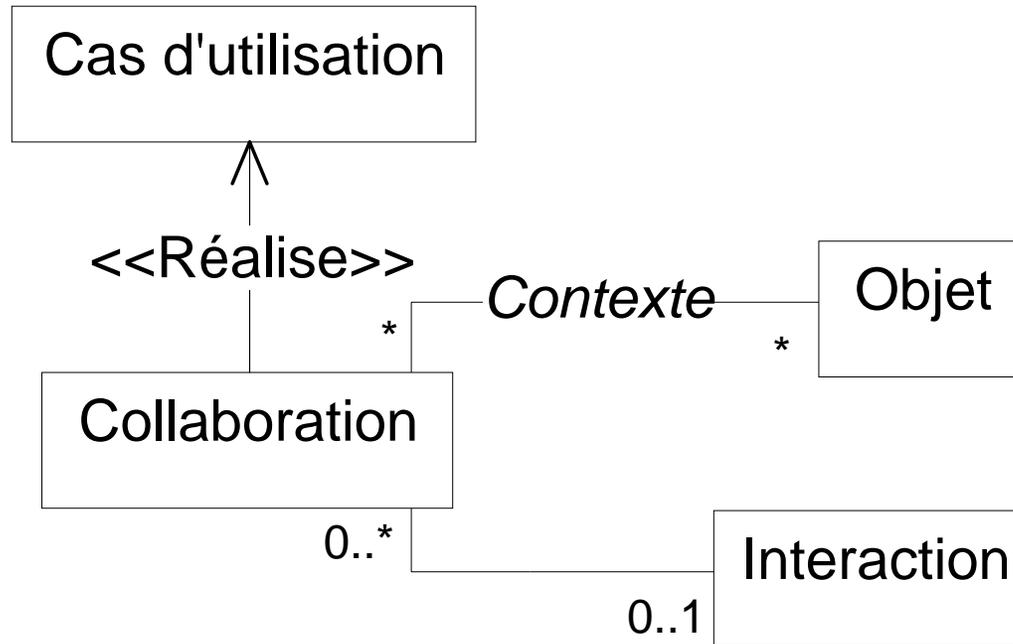
Expression des besoins

- Les cas d'utilisation

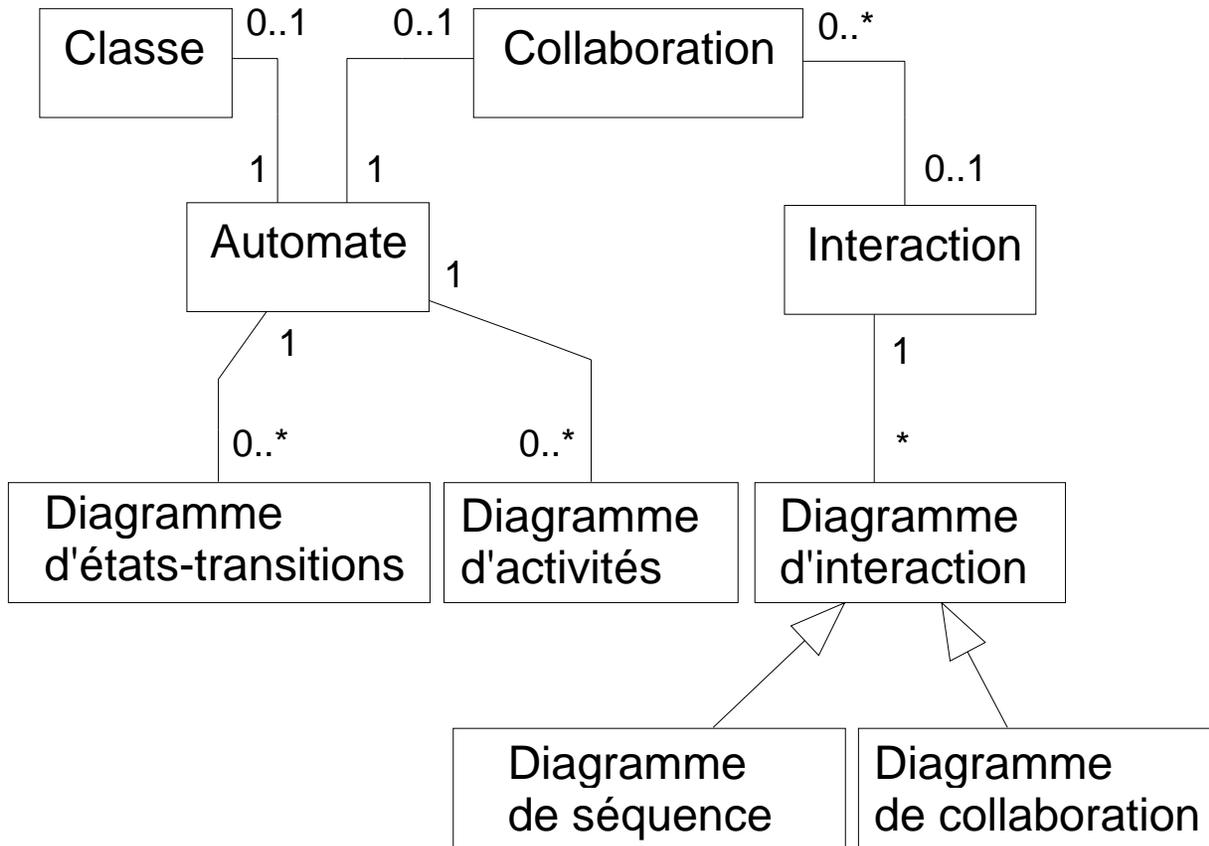


Transition vers l'objet

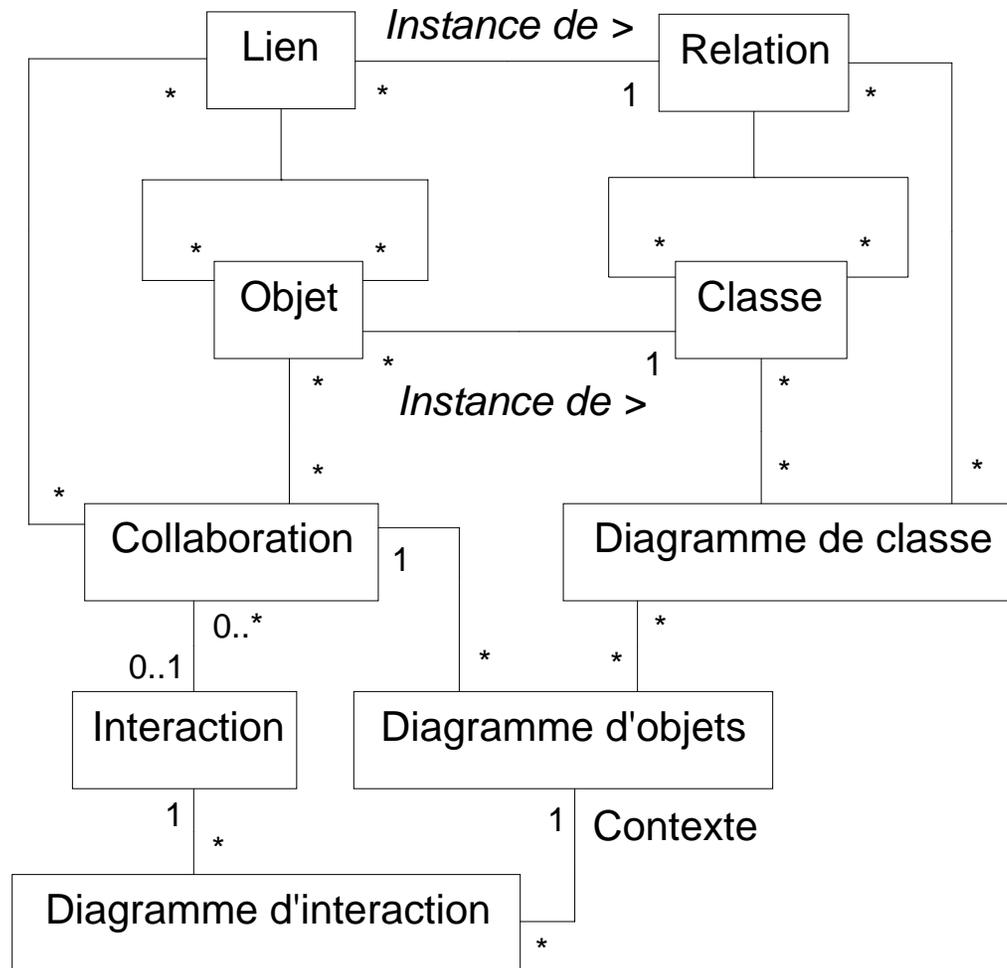
- Les collaborations



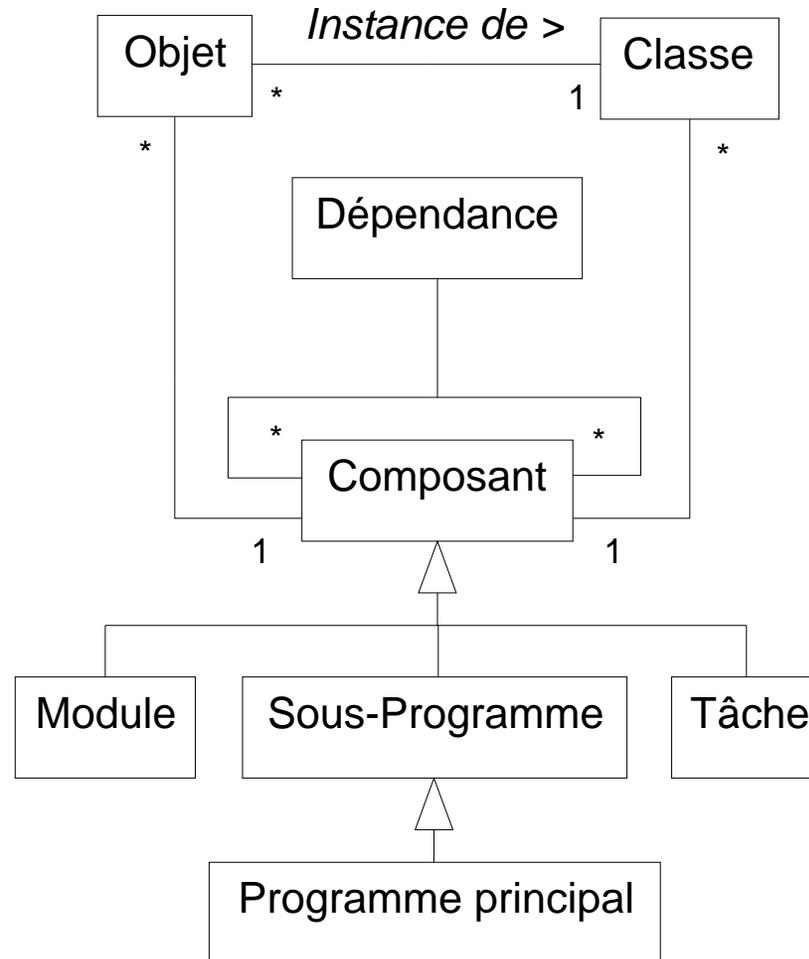
Expression du comportement



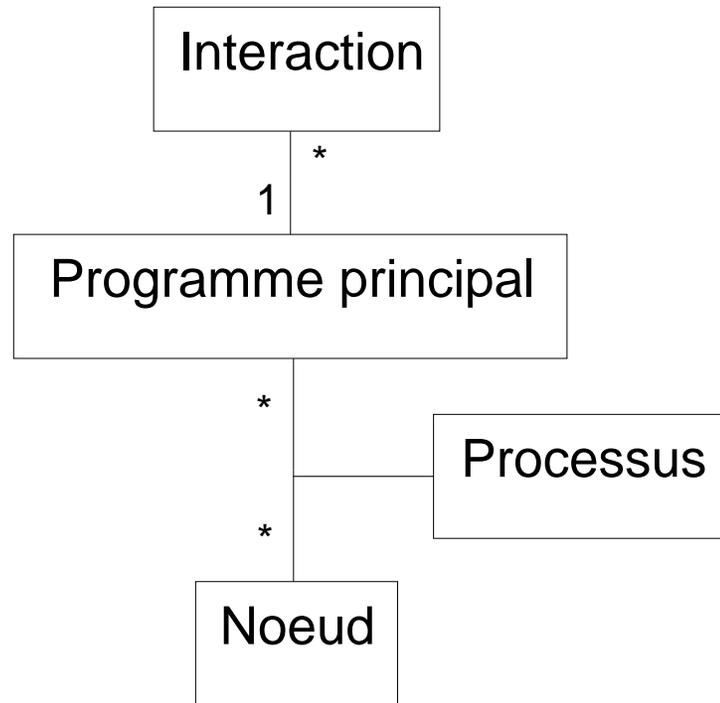
Représentation de la structure



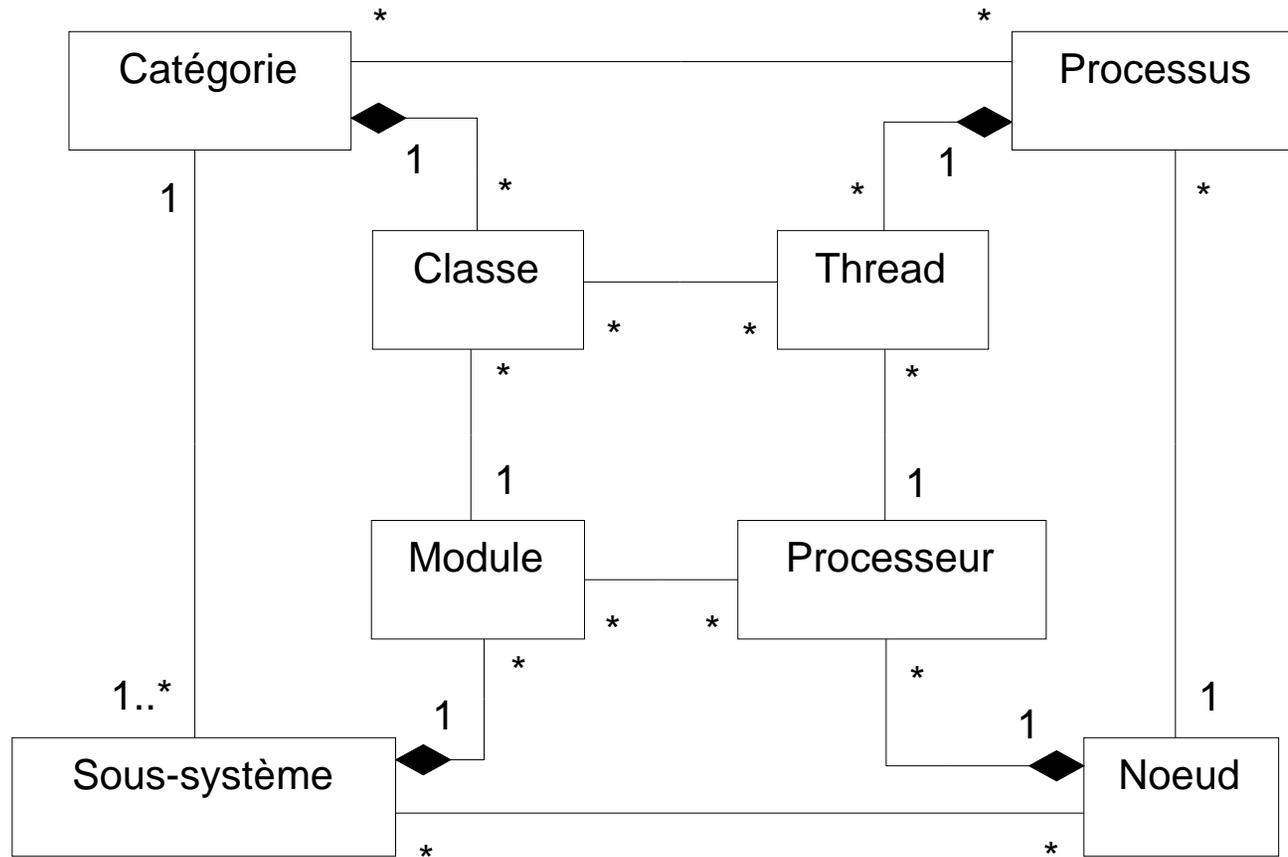
Réalisation des objets et des classes



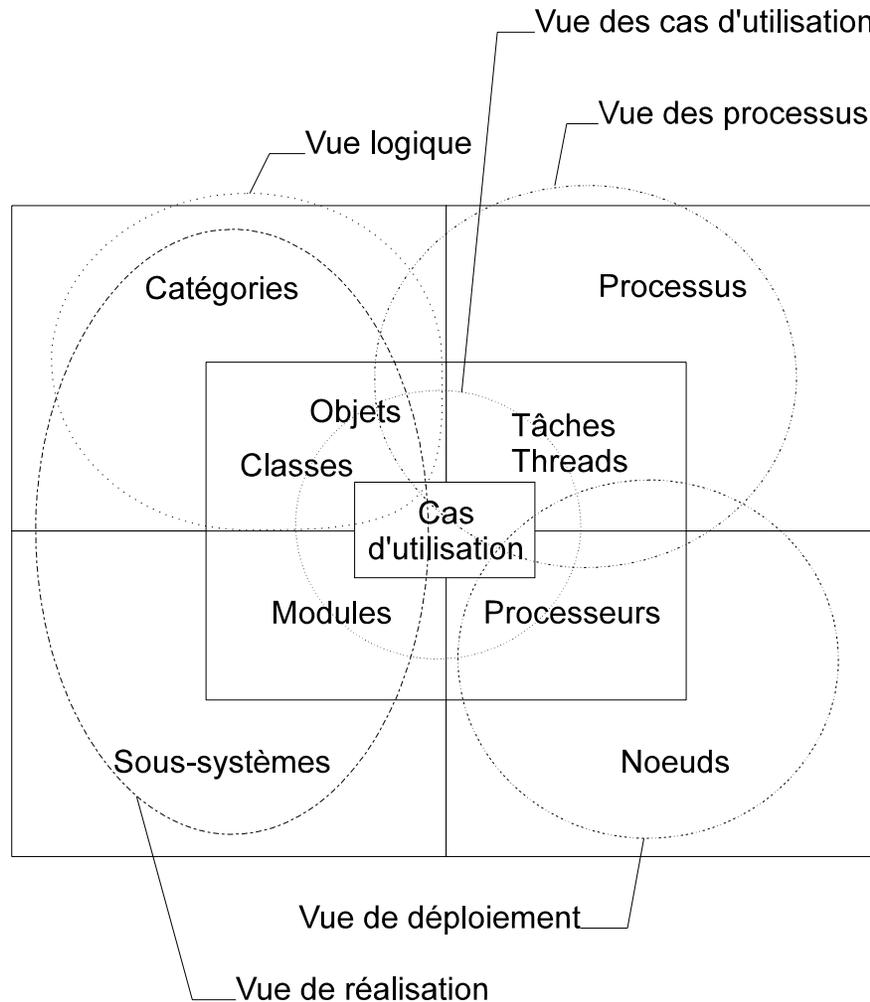
Déploiement du code exécutable



Granularité des éléments



Granularité (suite)



Récapitulatif

| | Vue des cas d'utilisation | Vue logique | Vue de réalisation | Vue des processus | Vue de déploiement |
|--------------------------------|---------------------------------------|--|--------------------|-----------------------------|--------------------|
| Diagramme de cas d'utilisation | Acteurs Cas d'utilisation | | | | |
| Diagramme de classes | | Classes Relations | | | |
| Diagramme d'objets | Objets Liens | Classes Objets Liens | | | |
| Diagramme de séquence | Acteurs Objets Messages | Acteurs Objets Messages | | Objets Messages | |
| Diagramme de collaboration | Acteurs Objets Liens Message | Acteurs Objets Liens Messages | | Objets Liens Messages | |



Récapitulatif (suite)

| | Vue des cas d'utilisation | Vue logique | Vue de réalisation | Vue des processus | Vue de déploiement |
|-------------------------------|---------------------------|--------------------------|--------------------|--------------------------|--------------------|
| Diagramme d'états-transitions | Etats Transitions | Etats Transitions | | Etats Transitions | |
| Diagramme d'activité | Activités Transitions | Activités Transitions | | Activités Transitions | |
| Diagramme de composants | | | Composants | Composants | Composants |
| Diagramme de déploiement | | | | | Noeuds Liens |



Conclusion

- Il y a beaucoup de diagrammes
- Il est important de bien saisir leur articulation
- UML se prête bien à la représentation de l'architecture

