

Aperçu de la méthodologie ACCORD/UML

Aperçu de la méthode avec l'exemple du Régulateur de vitesse

I. Analyse préliminaire

1. Constituer le dictionnaire
2. Décrire les cas d'utilisation
3. Décrire les scénarios de haut niveau
4. Classifier les acteurs

II. Analyse détaillée

1. Construire l'architecture de la vue structurelle
2. Décrire la vue structurelle
3. Décrire la vue comportementale
4. Décrire la vue des interactions

I. Analyse préliminaire

Constituer le dictionnaire = analyser le cahier des charges

Name (class or actor)	Qualifier (attribute or relationship)	Verb (operation)
Speed regulator		place (regulation) in service
		shut down (regulation)
	(relationship with speed display)	refresh (display)
		interrupt regulation
		reinstate regulation
Speed	Setpoint	demand (setpoint)
	Normal	achieve (normal speed)
Brake pedal		depress
		release
Accelerator		depress
		release

Objectifs :

Noms



Acteurs
ou classes

Qualificatifs



Attributs
ou relations

Verbes



Opérations

Décrire les cas d'utilisation

- Identifier l'environnement

... to generate a signal to the **engine**...

... The resulting signal is then routed to the vehicle **drive train**...

... the **engine control** and **drive systems** have been grouped together as an **engine unit**...

...The regulation system is started when the driver actuates the **regulator on/off button**...

...by depressing the **brake pedal**...

... by stopping the **engine** (via the **starter** ...

... Moreover, if the driver depresses the **accelerator** ...

... The speed regulator is equipped with a **screen** for **display** of speed setting...

- Identifier les services

... designed to **maintain** vehicle speed on a setpoint value...

... The regulation system is **started**...

... System **stop** ...

... regulation is **interrupted** until...

... The regulator then **reinstates** control...

- Identifier les relations

... Speed is maintained with respect to changes in torque, which are **signalled to the engine control system** ...

... Vehicle speed is measured by a speedometer ...

... The regulation system is started when the driver actuates the regulator on/off button , thus generating a start signal ...

... System stop... by depressing the brake pedal ...

... System stop...by actuating the on/off button ...

... System stop... by stopping the engine...

...System stop... when vehicle speed falls below 50 km/h ...

...if the driver **depresses the accelerator**... regulation is interrupted ...

... is interrupted until the accelerator is released ...

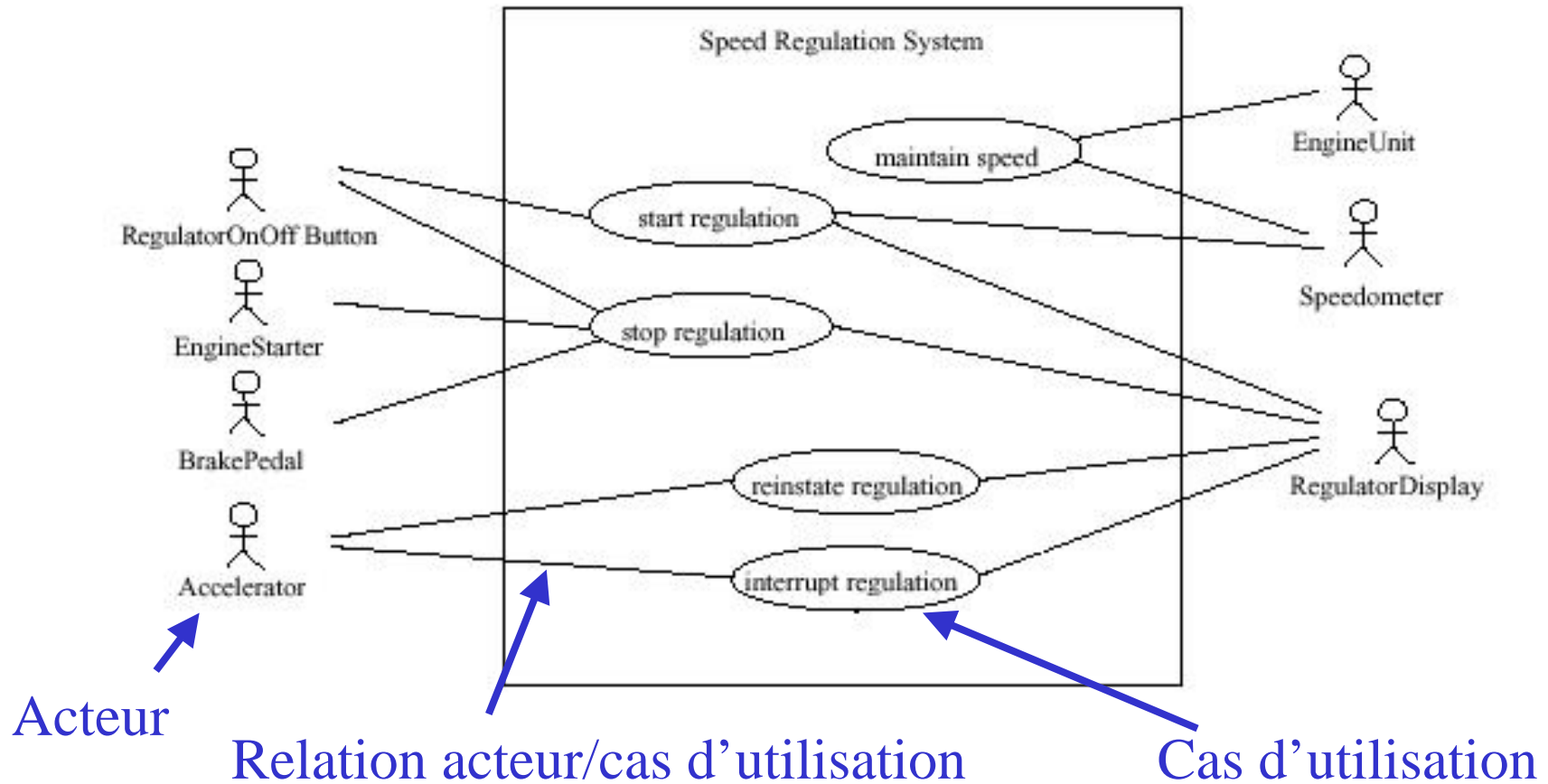
... with a screen for display of speed setting and regulation system status ...

Key :

Expression associated with an actor

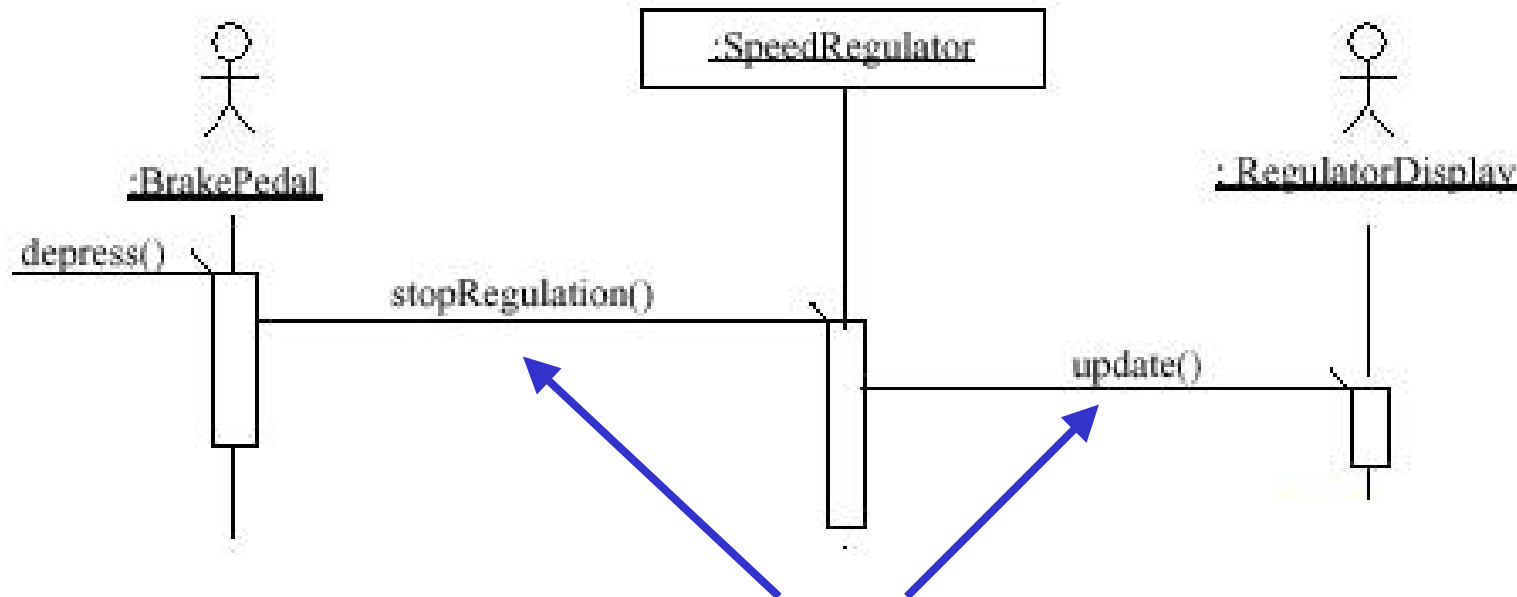
Expression associated with a use case

Diagramme de cas d'utilisation



Décrire les scénarios de haut niveau

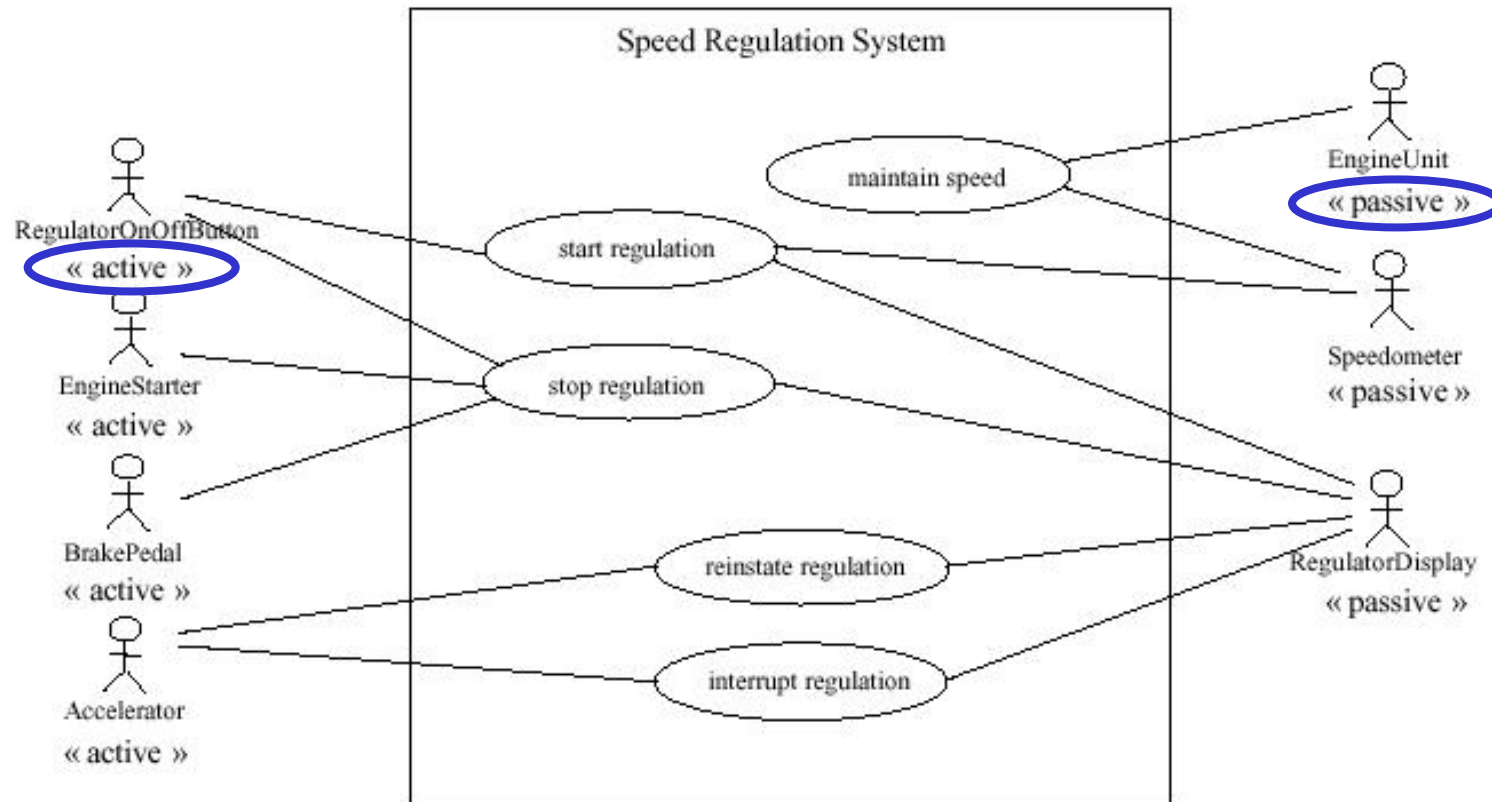
*Un diagramme de séquence
du cas d'utilisation interrupt regulation*



Interactions entre le système et les acteurs

Classifier les acteurs

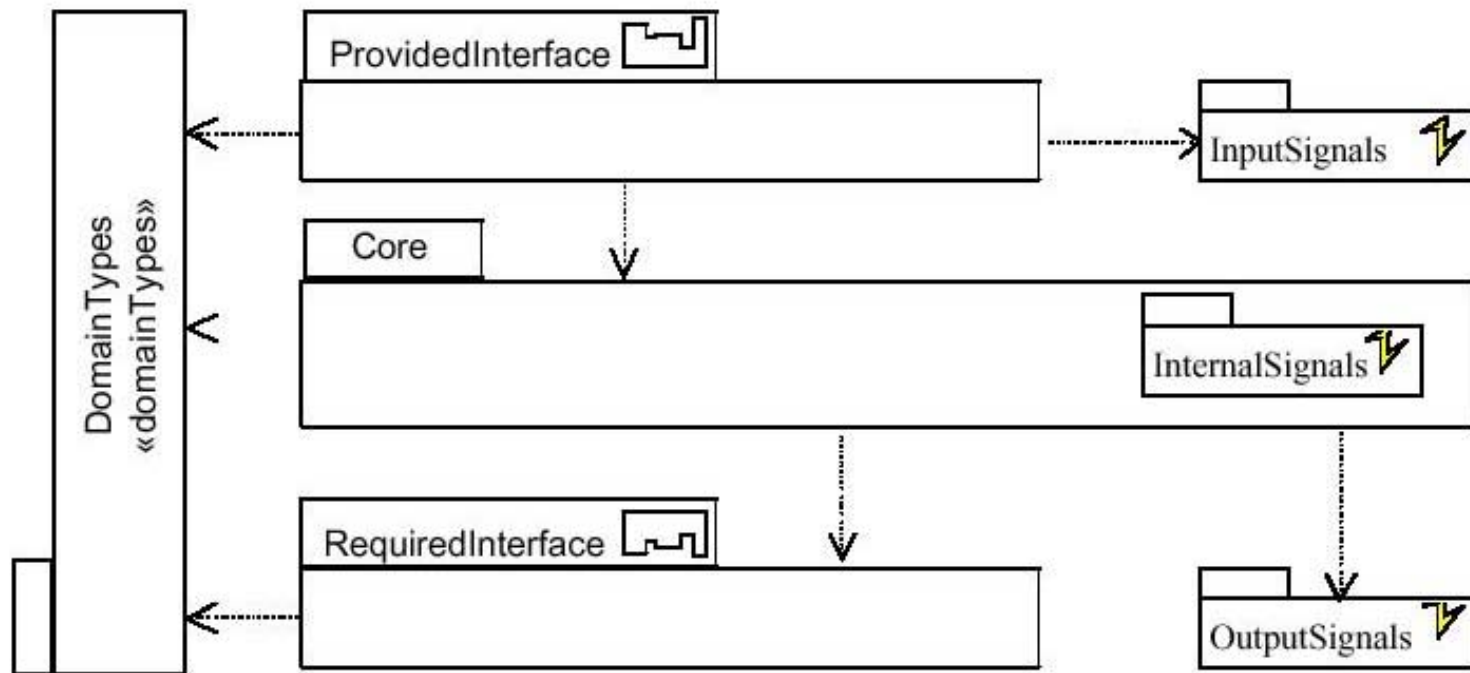
- Ajouter aux acteurs le stéréotype « active » ou « passive »



II. Analyse détaillée

Construire l'architecture de la vue structurelle

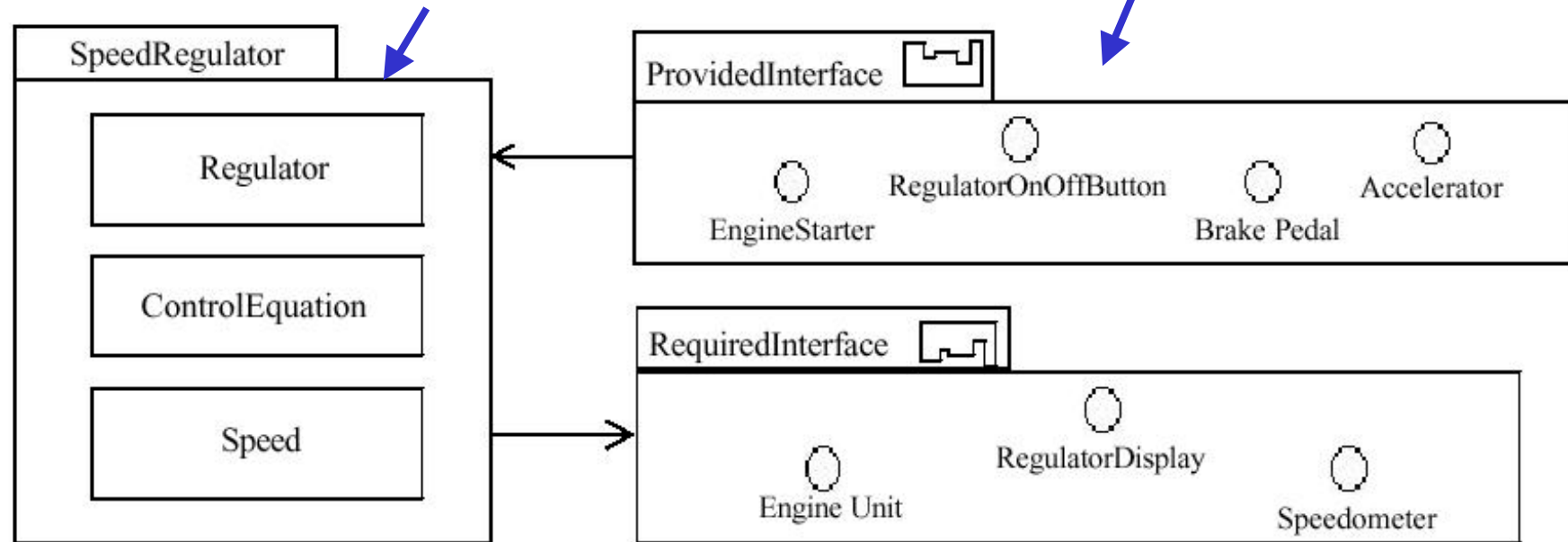
Diagramme de classe global



Définir les classes

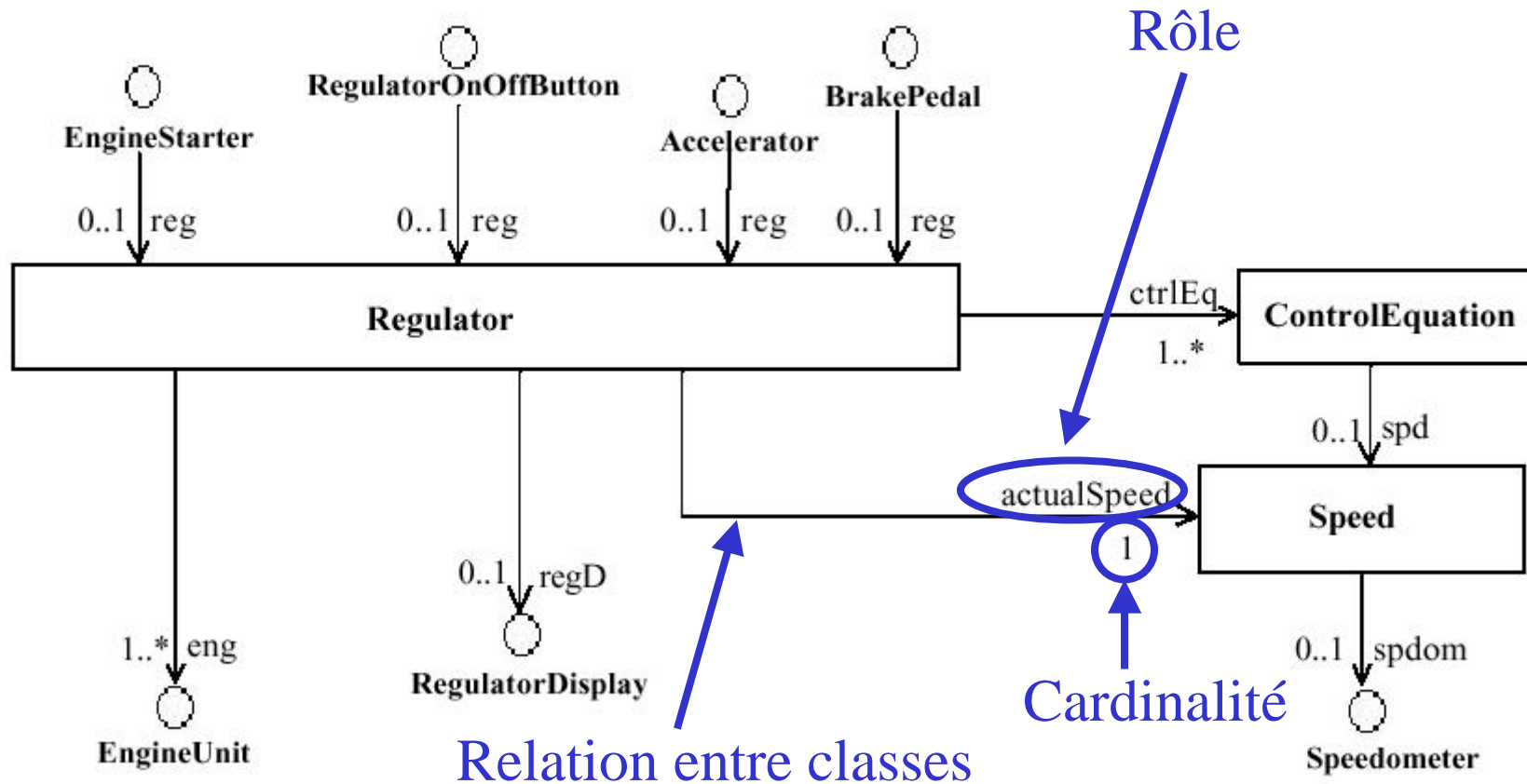
Décomposition du système en plusieurs classes

Transformation des acteurs actifs en classes interfaces



Transformation des acteurs passifs en classes interfaces

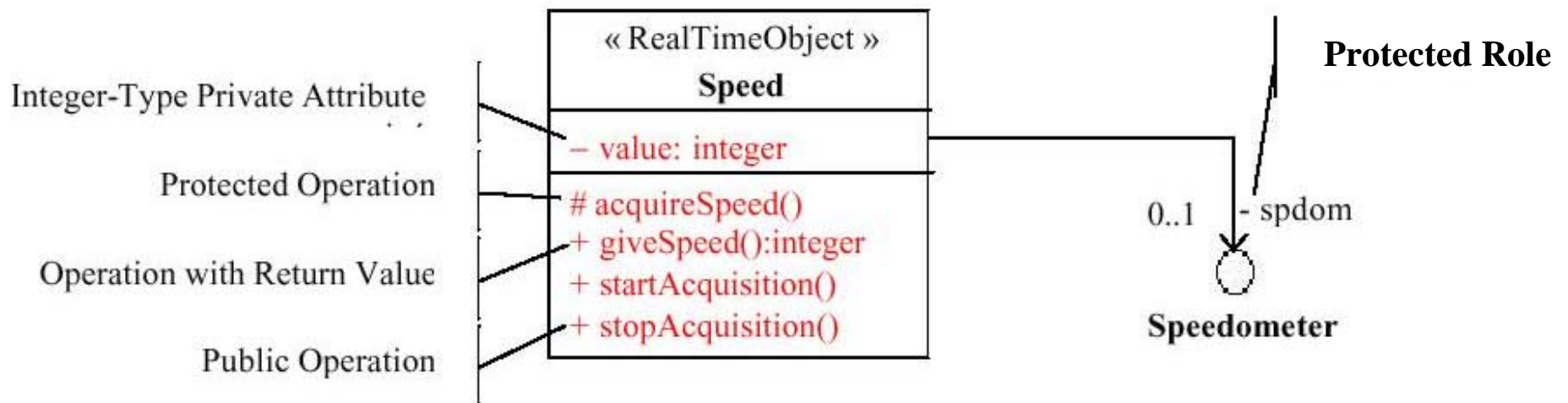
Définir les relations entre classes



Décrire les services des classes

- Déclarer les attributs
- Déclarer les méthodes
- Préciser les visibilités des attributs et des méthodes

La classe Speed



Spécifier les communications entre classes

- Déclarer les signaux
- Déclarer les classes émettrices des signaux
- Déclarer les classes sensibles aux signaux

*Deux signaux d'entrée
pour le régulateur de vitesse*

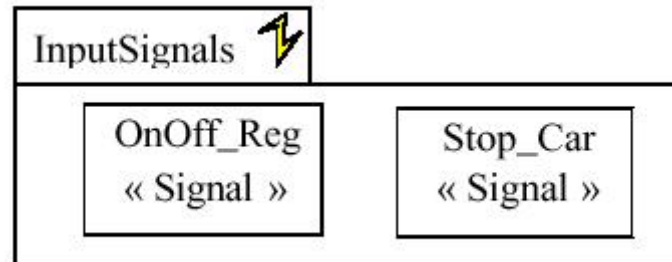
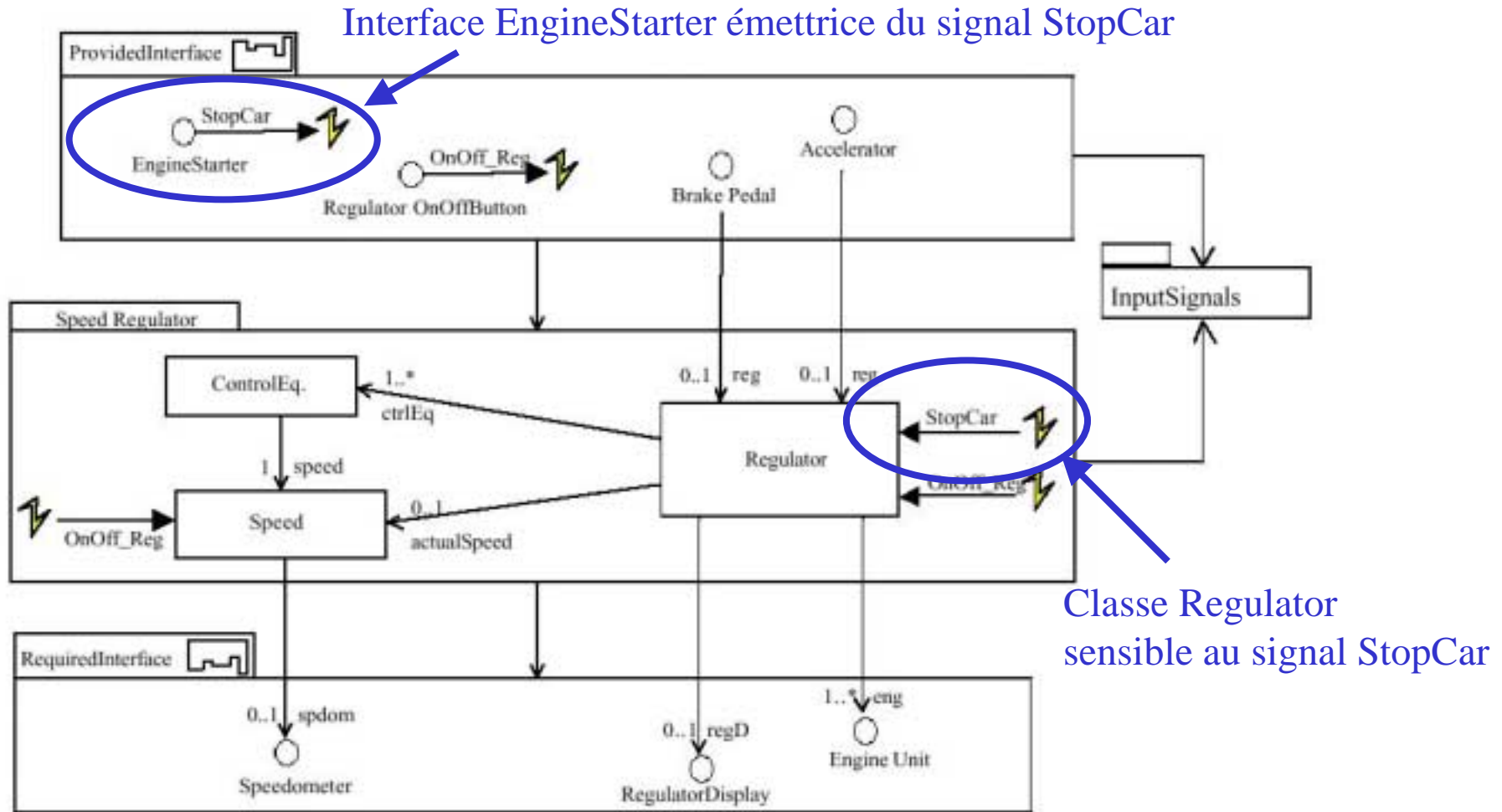


Diagramme de classes global



Déclarer les ressources actives

Classe active = Classe capable de gérer la réception des signaux

- Ajouter du stéréotype « RealTimeObject »
- Préciser la tagged value concurrencyMode pour les méthodes

La classe Speed

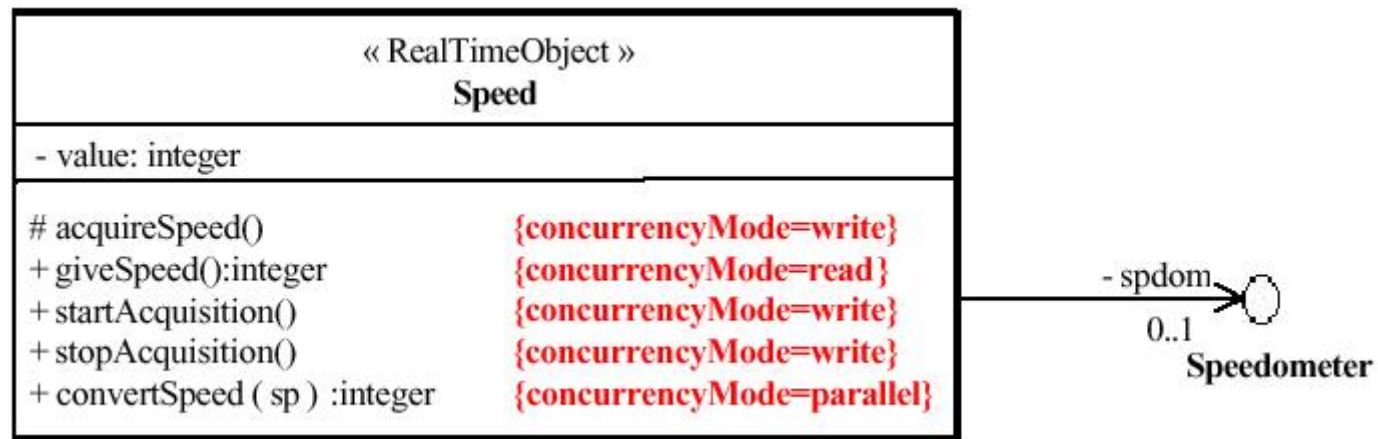
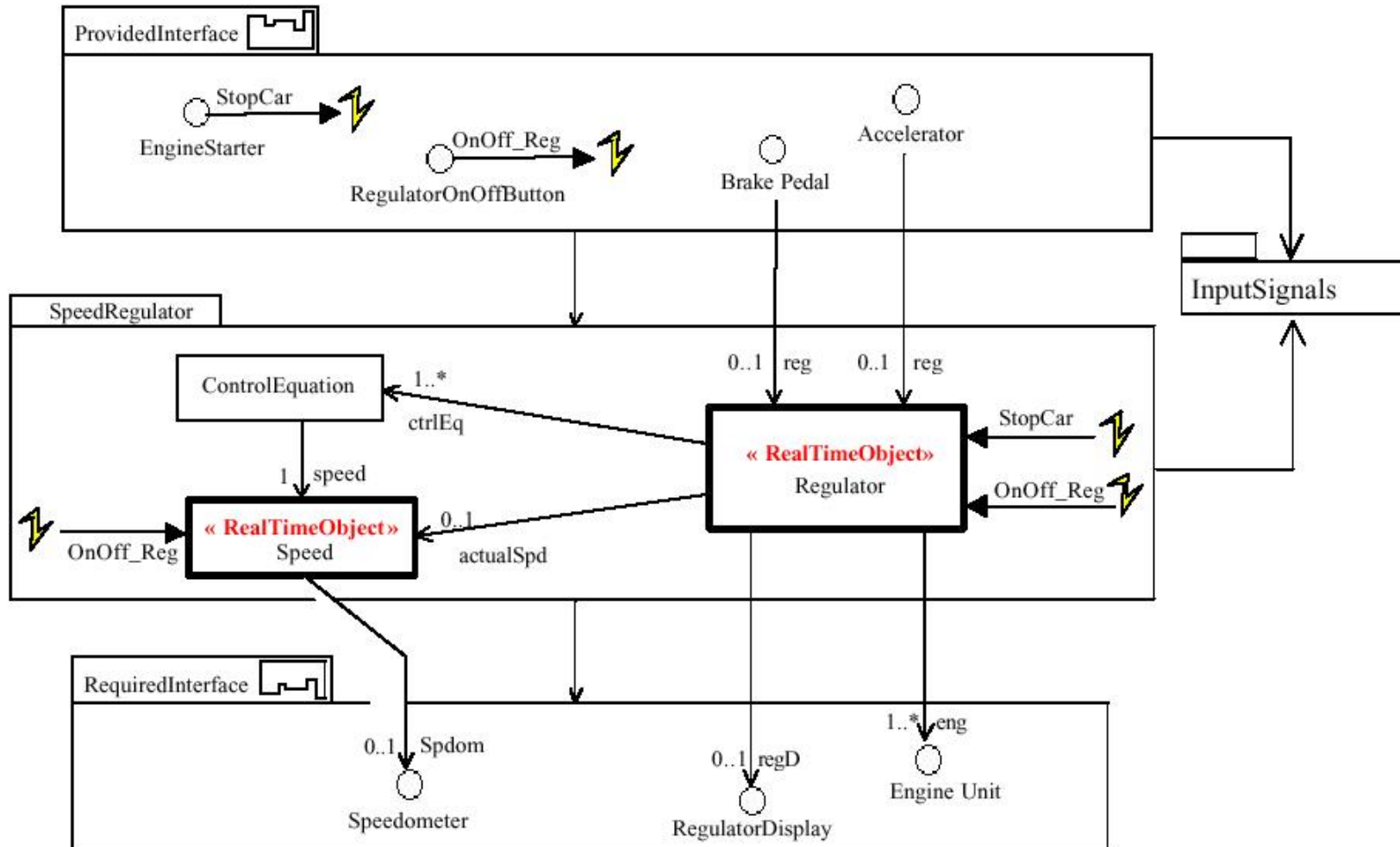


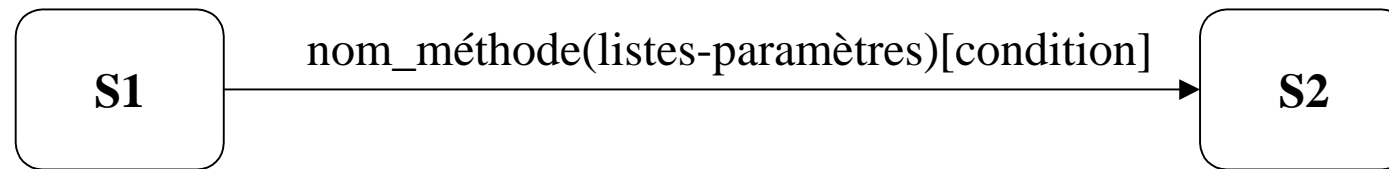
Diagramme de classes global



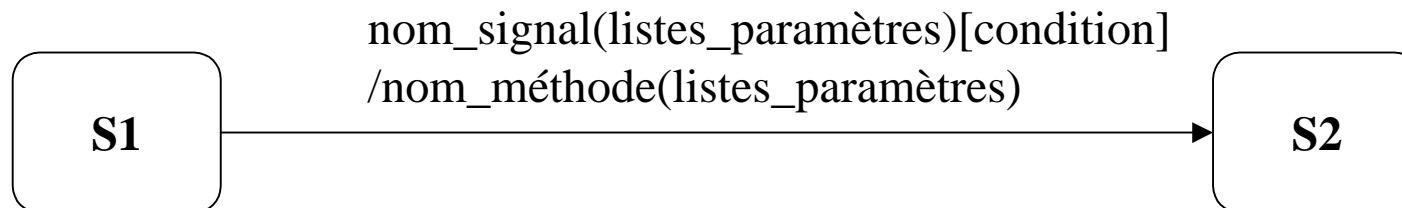
Décrire la vue comportementale pour les classes actives

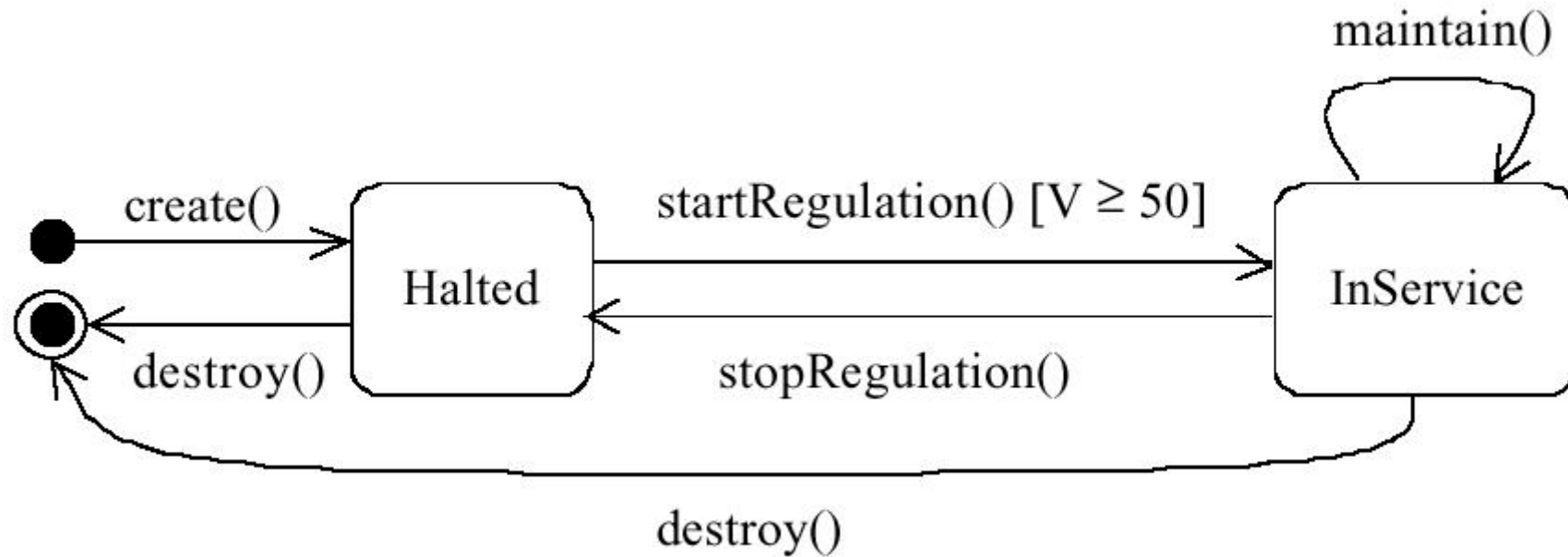
→ 2 diagrammes d'états-transitions

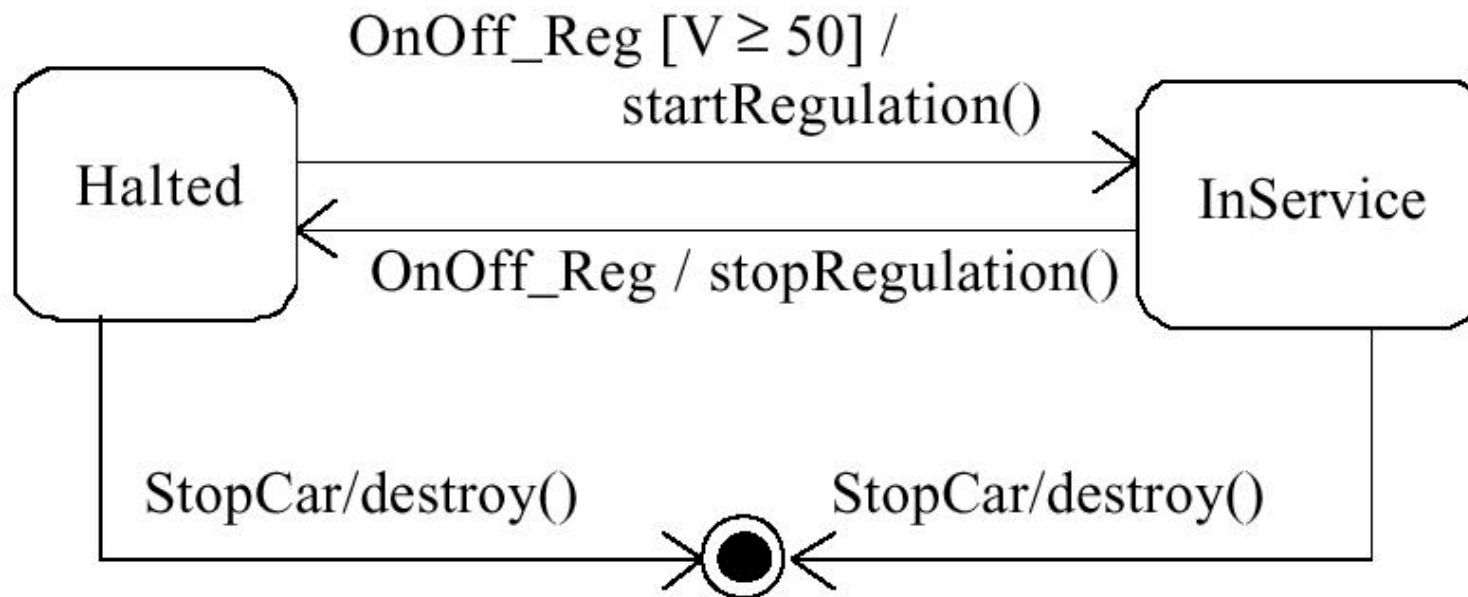
- Décrire la vue protocole (comportement/appel d'opérations)



- Décrire la vue déclenchement (comportement/réception de signaux)



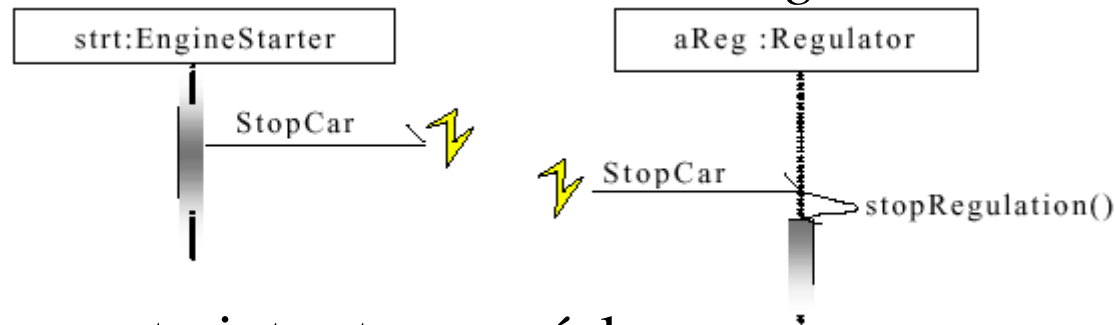
Vue protocole du régulateur de vitesse

Vue réactive du régulateur de vitesse

Décrire la vue des interactions

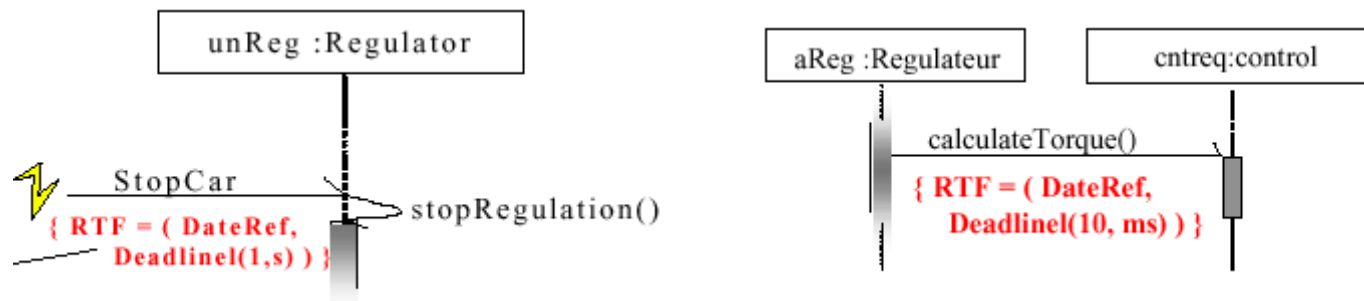
- Raffiner les diagrammes de séquences
(envois de messages et signaux signaux entre classes)

Interactions avec envois de signaux



- Préciser les contraintes temps réel

Utilisation de la tagged value RTF



*Remarques sur l'utilisation de la Méthode ACCORD-UML
pour modéliser l'application marine*

- Une méthodologie *orientée* informatique !
- Modèle du Régulateur construit pour illustrer la méthode !
- Nécessité de travailler avec des signaux discrets ?
- Objectif de la modélisation de l'application marine ?
- Préciser les sous-systèmes à modéliser dans l'application marine.