

OCM 2000 – 18 Mai 2000

Éric CARIOU

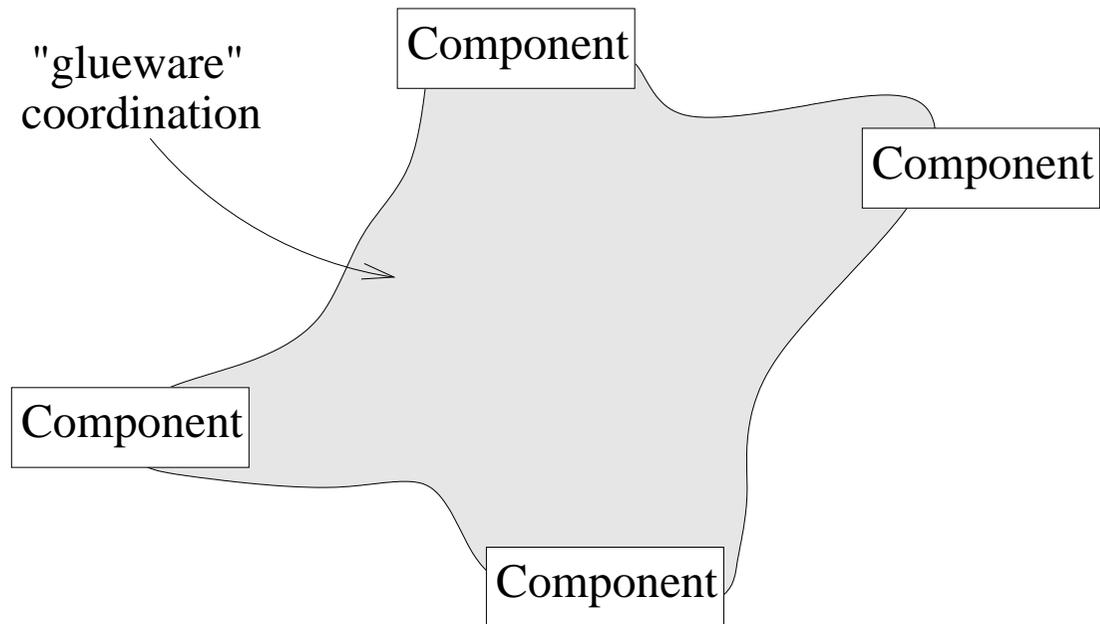
Spécification de Composants de
Communication en UML



PLAN

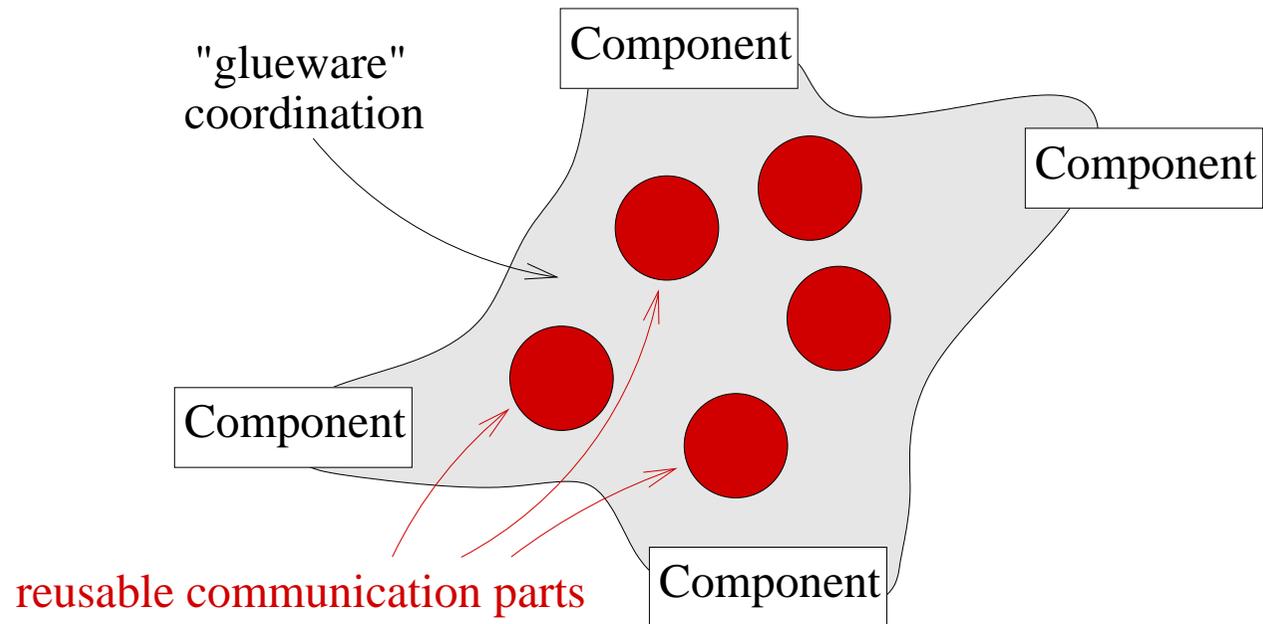
1. La notion de composant de communication
2. Objectif de la spécification
3. Deux exemples d'application
4. Spécification d'un composant de vote
5. Conclusion et perspectives

PROBLÉMATIQUE



- « glueware coordination » = partie spécifique sur mesure d'une application liant des composants réutilisables

PROBLÉMATIQUE



- Des parties de cette « glueware coordination » sont réutilisables
⇒ des **composants de communication** (ou médiums)

LE MÉDIUM DE COMMUNICATION

- Composant logiciel réutilisable
- Implémente un service ou protocole de communication : consensus, diffusion, vote...
- Composants distribués communiquent via des médiums
- ⇒ *réifie la communication inter-composants*

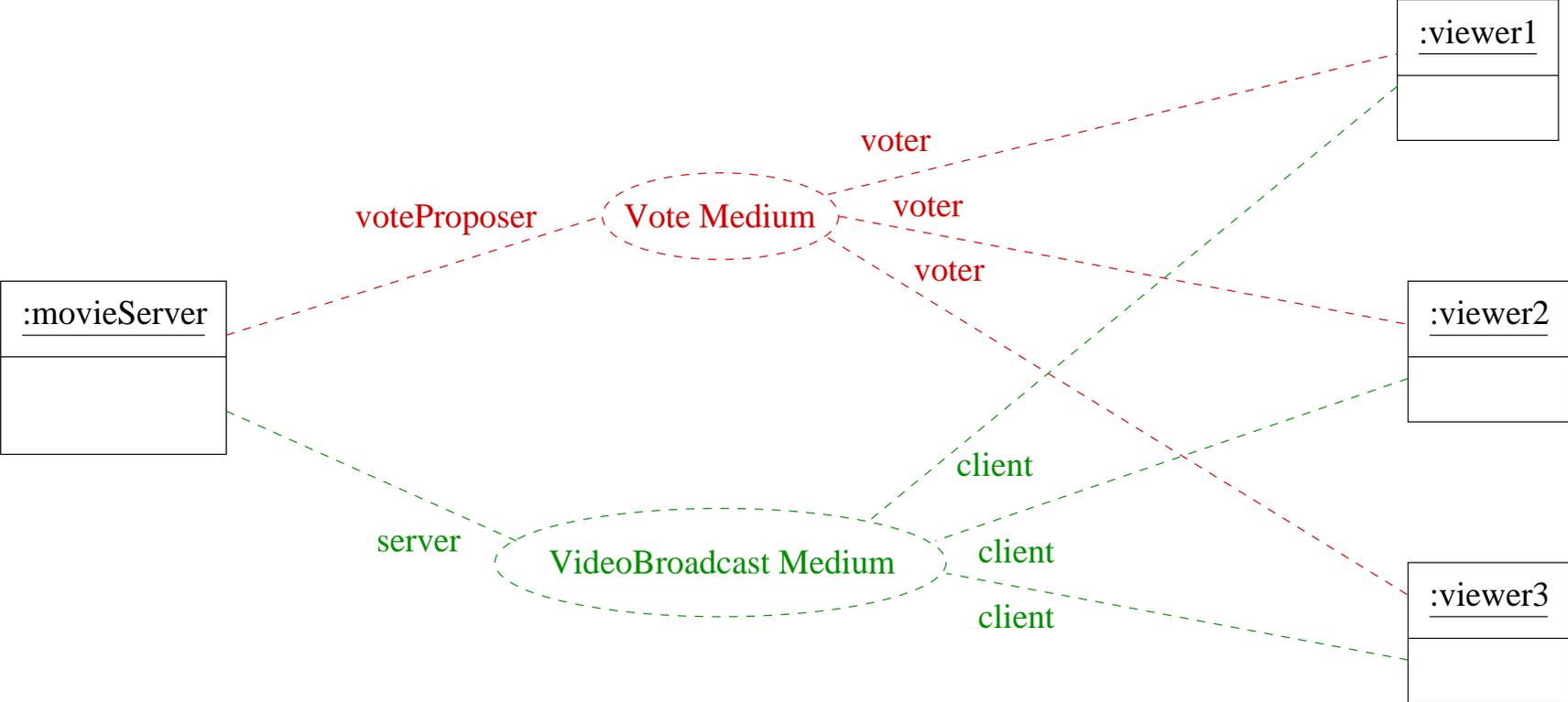
SPÉCIFICATION DE MÉDIUM

- Spécification de médium \Rightarrow spécification précise de son **contrat de fonctionnement** :
 - ↳ Liste des services qu'il offre
 - ↳ Liste des services nécessaires chez les composants
 - ↳ Spécification comportementale du et de ses services
 - ↳ Toute information décrivant son fonctionnement

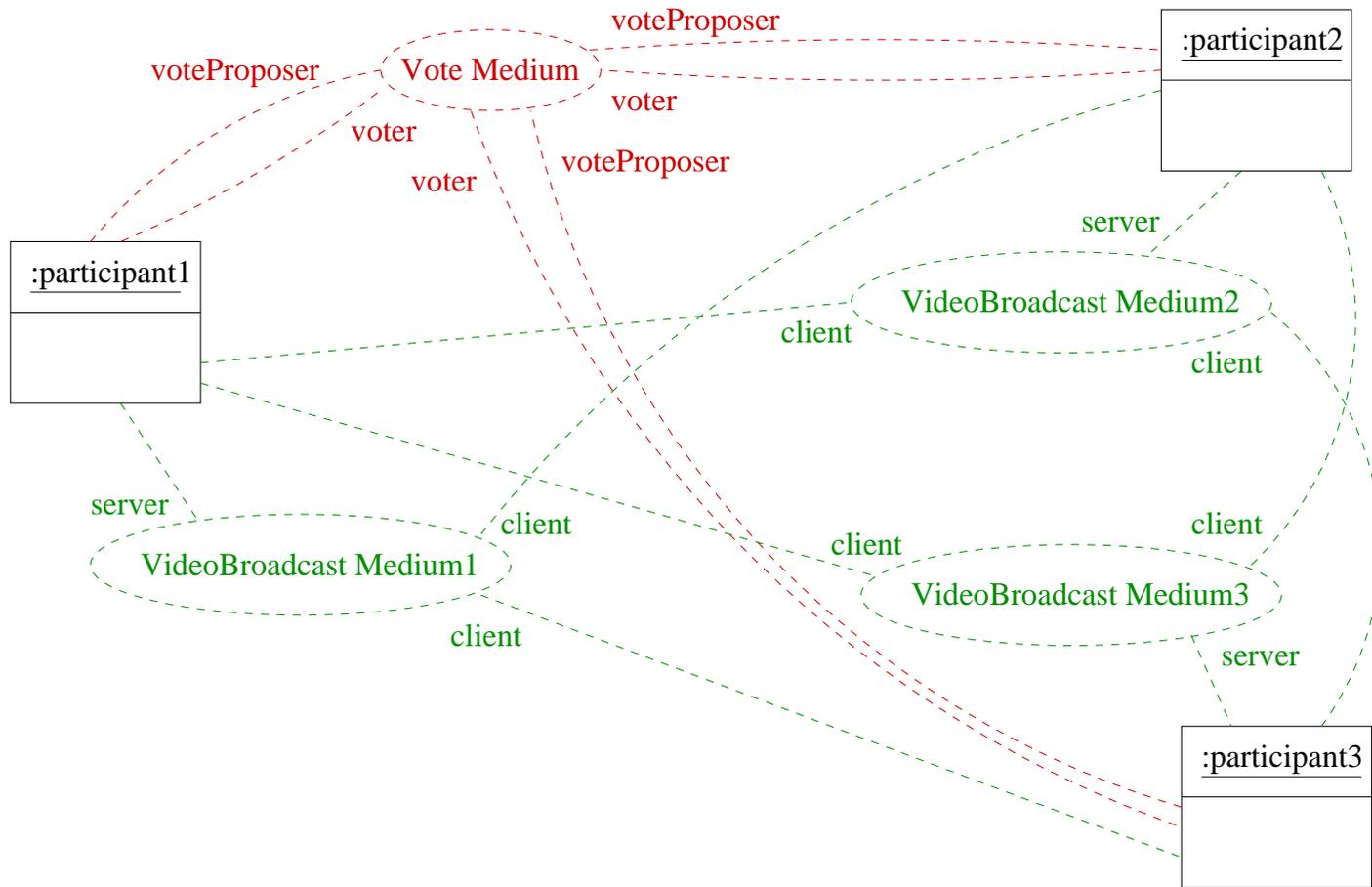
SPÉCIFICATION DE MÉDIUM

- Spécification de médium \Rightarrow spécification précise de son **contrat de fonctionnement** :
 - ↳ Liste des services qu'il offre
 - ↳ Liste des services nécessaires chez les composants
 - ↳ Spécification comportementale du et de ses services
 - ↳ Toute information décrivant son fonctionnement
 - ⇒ **En UML** : standard dans le monde du génie logiciel

EXEMPLE : APPLICATION DE VIDÉO INTERACTIVE



EXEMPLE : APPLICATION DE VISIOCONFÉRENCE



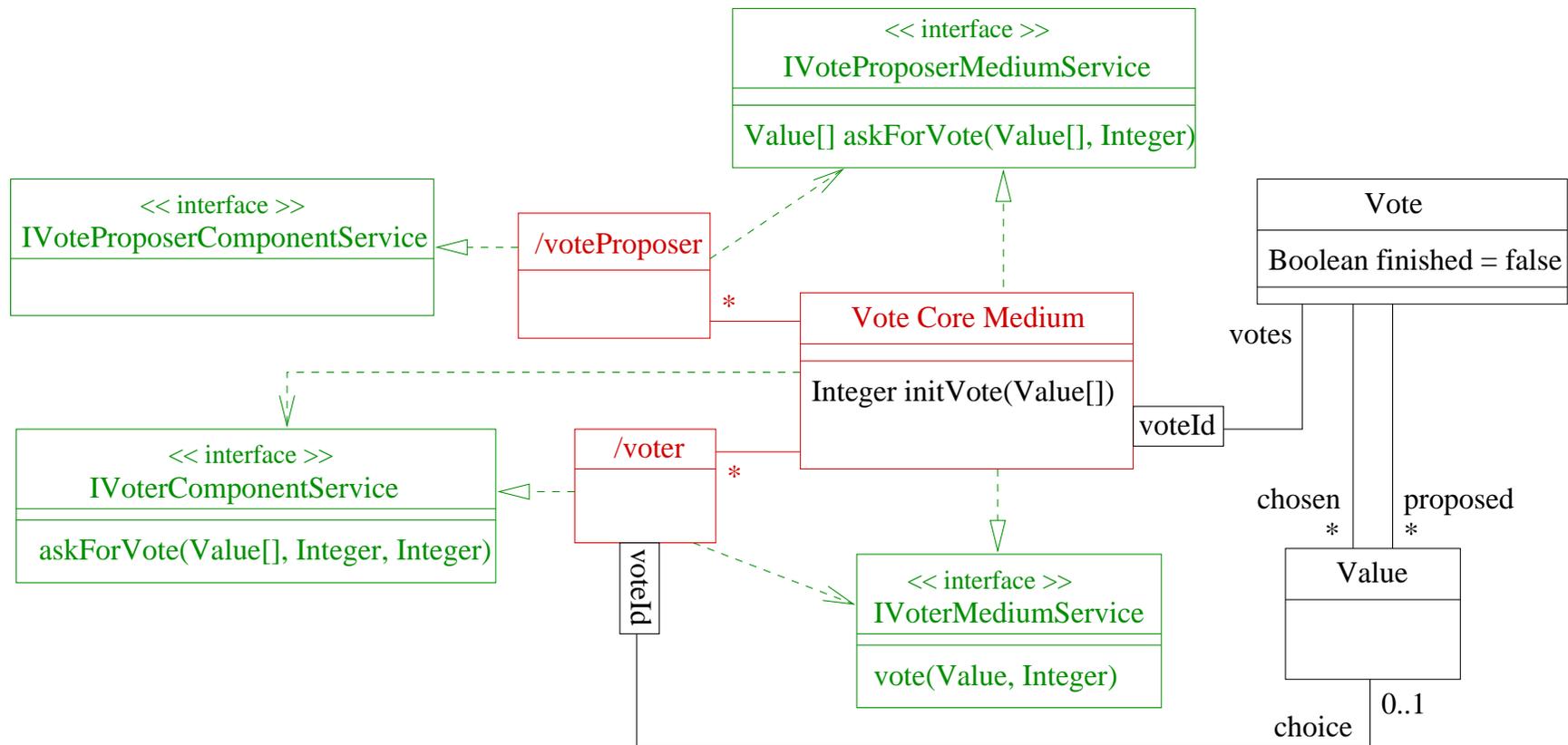
SPÉCIFICATION DE CONTRAT DE MÉDIUM EN UML

- **Vue structurelle** : collaboration (description d'interaction)
- Liste des **services** offerts et requis (signatures) :
 - ⇒ **Interfaces** dans *diagramme de collaboration* et **rôles**
- **Comportement** du médium et des services offerts :
 - ⇒ statique : *invariants et pré/postconditions en OCL*
 - ⇒ dynamique (contraintes temporelles, synchronisation) : **diagrammes d'états** et **interactions** (*diagramme de collaboration*)

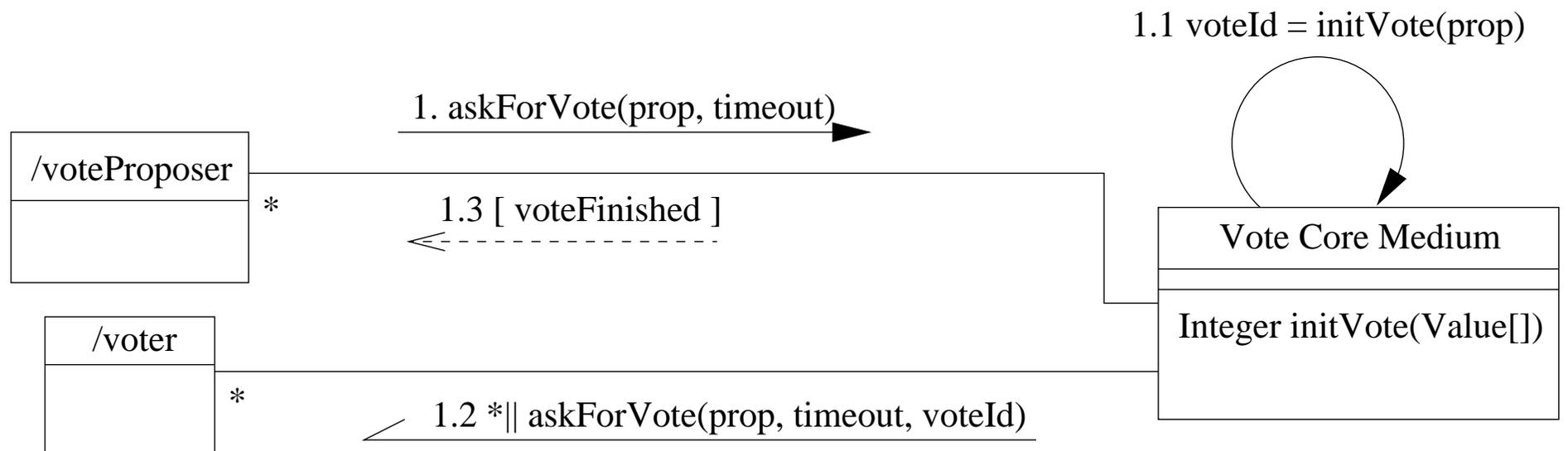
SPÉCIFICATION DE CONTRAT DE MÉDIUM EN UML

- **Vue structurelle** : collaboration (description d'interaction)
- Liste des **services** offerts et requis (signatures) :
 - ⇒ *Interfaces dans diagramme de collaboration et rôles*
- **Comportement** du médium et des services offerts :
 - ⇒ statique : *invariants et pré/postconditions en OCL*
 - ⇒ dynamique (contraintes temporelles, synchronisation) : *diagrammes d'états et interactions (diagramme de collaboration)*
- ⇒ **en généralisant l'utilisation d'OCL** : *lien formel et cohérence entre les différentes vues*

MÉDIUM DE VOTE : VUE STRUCTURELLE



MÉDIUM DE VOTE : VUE INTERACTION



MÉDIUM DE VOTE : VUE OCL

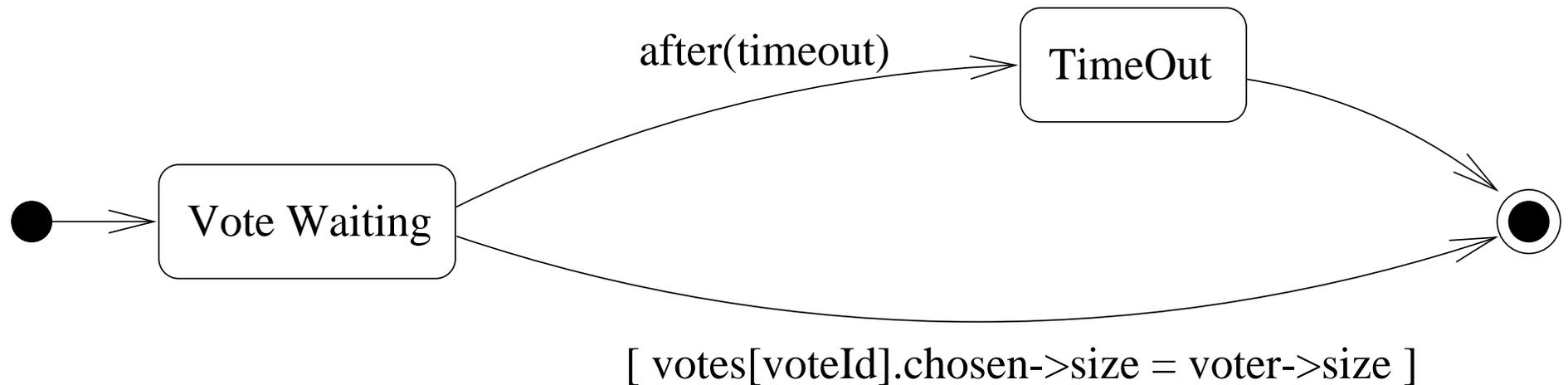
context Vote Core Medium::

```
askForVote(Value prop[], Integer timeout) : Value[]
```

post:

```
let voteId = self.oclCallOperation(initVote, prop) in  
voter.olcCallOperation(askForVote, prop, timeout, voteId) and  
votes[voteId].finished = true and  
voteFinished = true and  
result = votes[voteId].chosen
```

MÉDIUM DE VOTE : VUE DIAGRAMME D'ÉTATS



```
voteFinised = (voter->size = votes[voteId].chosen->size)
               or oclInState(TimeOut)
```

CONCLUSION

- **Apport des médiums** : modification de l'analyse, la conception et l'implémentation des systèmes par :
 - ↳ capitalisation des moyens de communication
 - ↳ une nouvelle façon de penser les applications distribuées

CONCLUSION

- **Apport des médiums** : modification de l'analyse, la conception et l'implémentation des systèmes par :
 - ↳ capitalisation des moyens de communication
 - ↳ une nouvelle façon de penser les applications distribuées
- **Limites d'UML:**
 - ↳ **Comportement dynamique** assez difficile à spécifier
 - ↳ **Spécification éclatée** dans plusieurs vues : recoupement pas toujours évident
 - ↳ **OCL formel mais limité**
 - ⇒ *mais arrivée d'ASL (Action Semantics Language)...*

PERSPECTIVES

- Développement d'un **catalogue** de médium (spécification + implémentation)
- Développement d'un **framework** pour l'implémentation de médiums
- Intégration de cette méthodologie de spécification dans un **AGL UML** (Umlaut) pour :
 - ↳ Génération automatique de code
 - ↳ Validation de médiums, test

POUR EN SAVOIR PLUS

- Umlaut : librement téléchargeable sur
<http://www.irisa.fr/pampa/UMLAUT>
- Informations sur les composants de communication :
<http://www-info.enst-bretagne.fr/medium>