

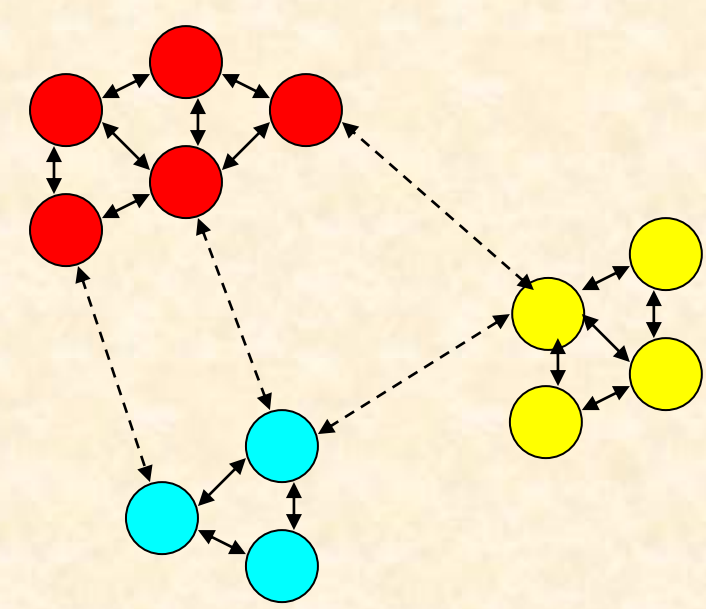
Un protocole de communication auto-adaptatif pour le calcul intensif pair à pair

The Tung Nguyen, Didier El Baz

CNRS ; LAAS ; 7 avenue du colonel Roche, F-31077 Toulouse, France

ANR-07-CIS7-011-01

Motivations



Calcul Intensif Pair à pair

- Parallélisme massif
- Disponibilité
- Solution économique
- Hétérogénéité
- Asynchronisme
- Volatilité
- Scalabilité

Protocoles de communication existants (TCP, UDP, etc.) : statiques
=> Non-adaptés à des environnements pair à pair



Protocole de communication auto-adaptatif : se-reconfiguration en adoptant le mode de communication le plus approprié entre les pairs en fonction de choix faits au niveau de l'application et de la topologie

1 – Micro-protocoles

❖ Approche micro-protocole

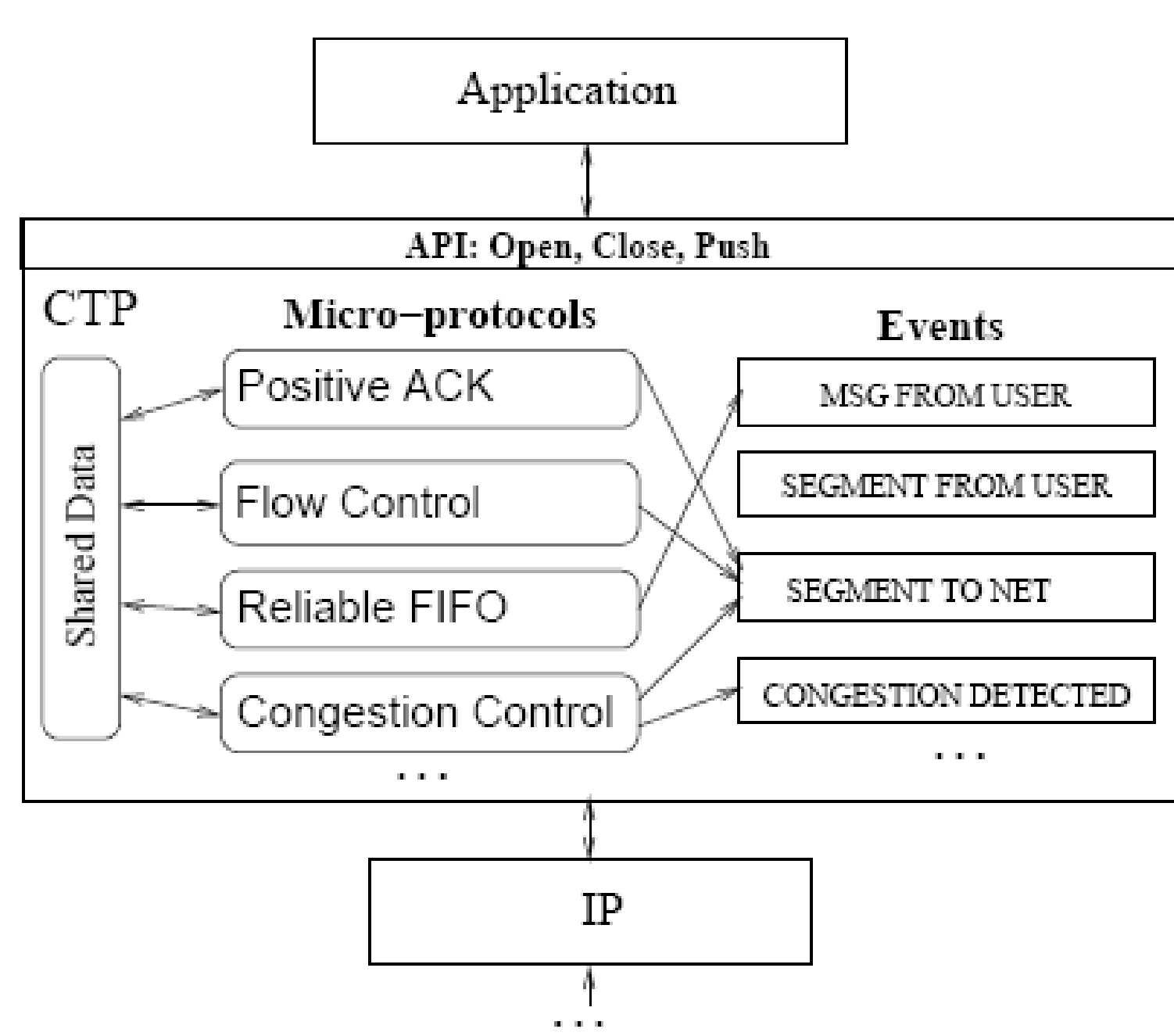
- *Un micro-protocole* ⇔ *une fonction du protocole.*
- *Protocole : composition d'un ensemble de micro-protocoles.*
- ⇒ *réutilisation de code, facilitation de conception, reconfiguration dynamique, etc.*

❖ Frameworks de composition

- *Hierarchique : x-kernel, Appia.*
- *Non-hierarchique : Coyote, ADAPTIVE.*
- *Hybride : XQoS, Cactus.*

❖ Framework Cactus

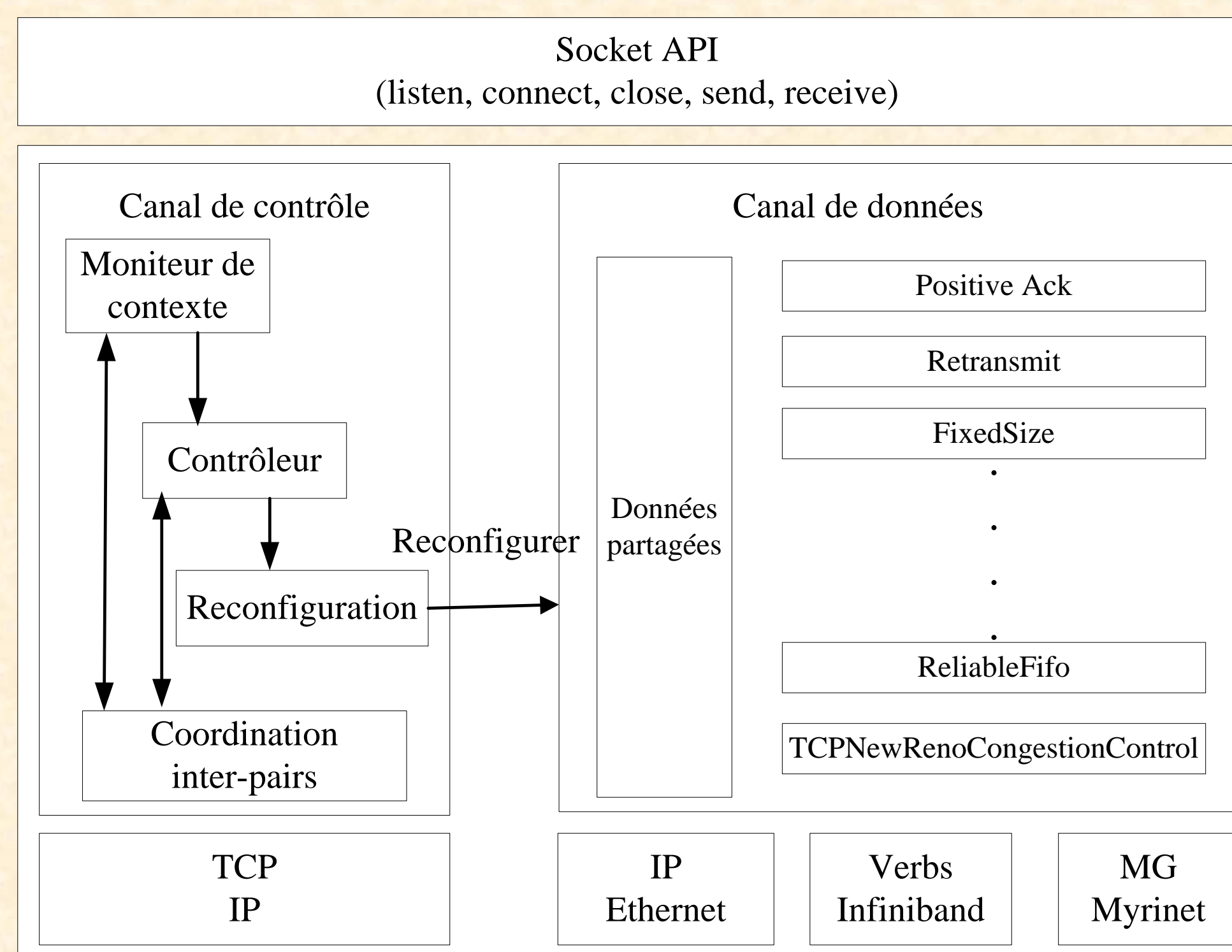
- *2 niveaux de granularités :*
 - ✓ *Protocole composite : composition des micro-protocoles, non-hierarchique.*
 - ✓ *Pile de protocoles : hierarchique.*
- *Basé sur les événements.*
- *API riche*



CTP – Exemple d'un protocole composite

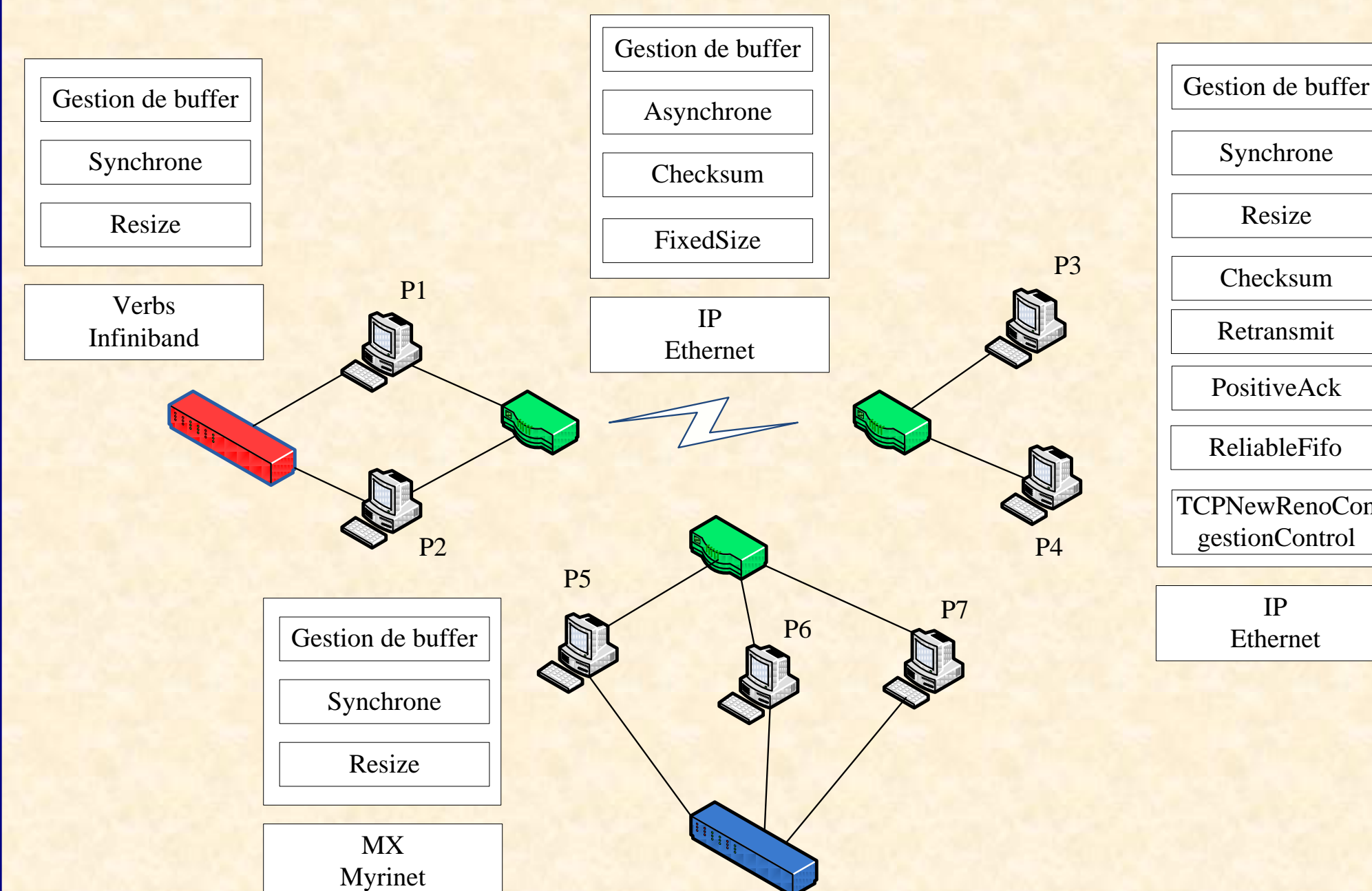
- **Nos modifications pour Cactus**
- . *Permission d'exécution concurrente.*
- . *Elimination des copies de message non-nécessaires.*
- . *Extension de API Cactus pour la suppression d'un micro-protocole.*

2 – Protocole auto-adaptatif

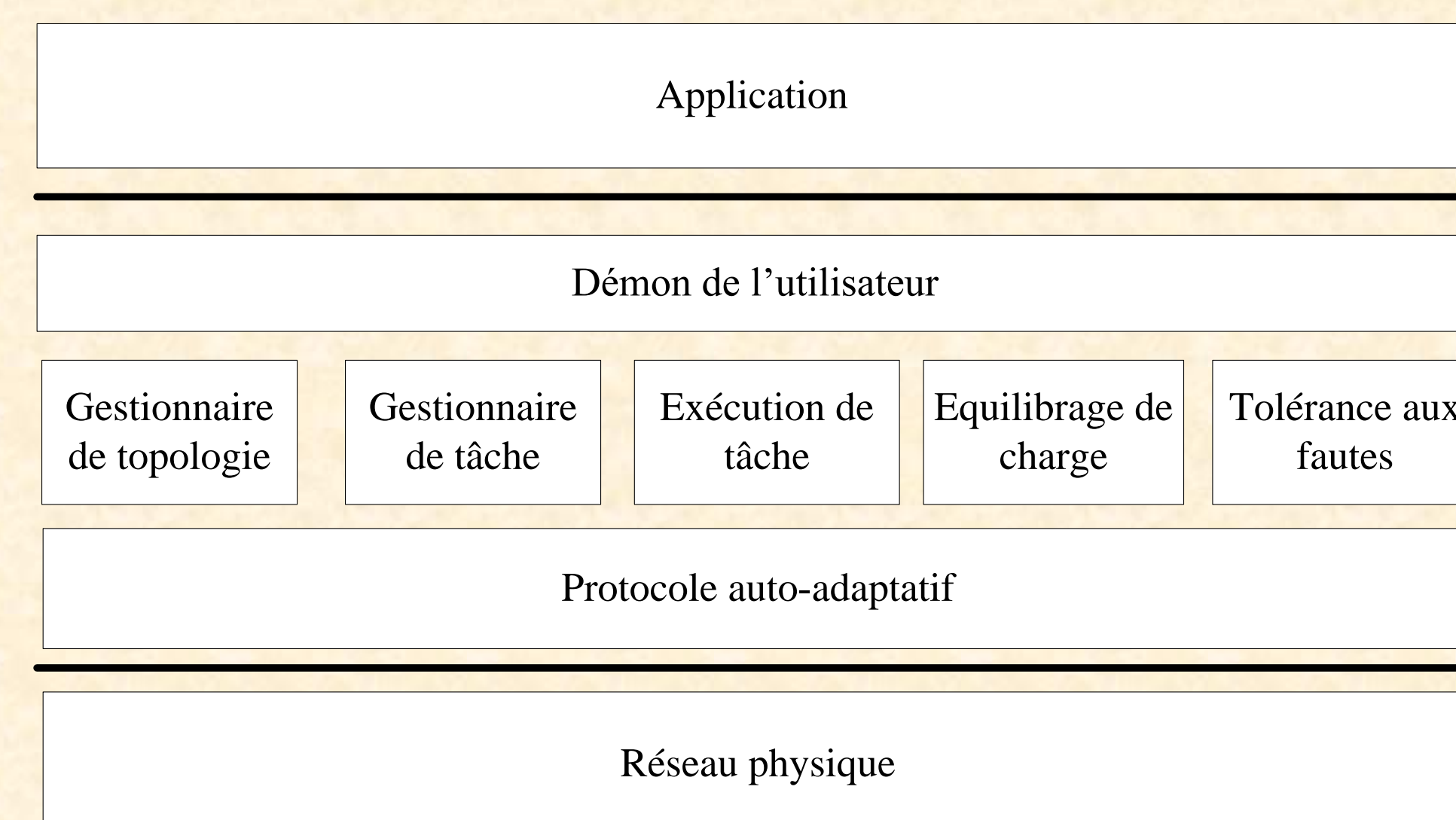


Architecture du protocole auto-adaptatif

- **Nouveaux micro-protocoles :**
- . *Synchrone, Asynchrone.*
- . *Gestion de buffer.*
- . *Congestion de contrôle : TCP NewReno, H-TCP*



Exemple de fonctionnement



Architecture de l'environnement

3 – Expérimentation

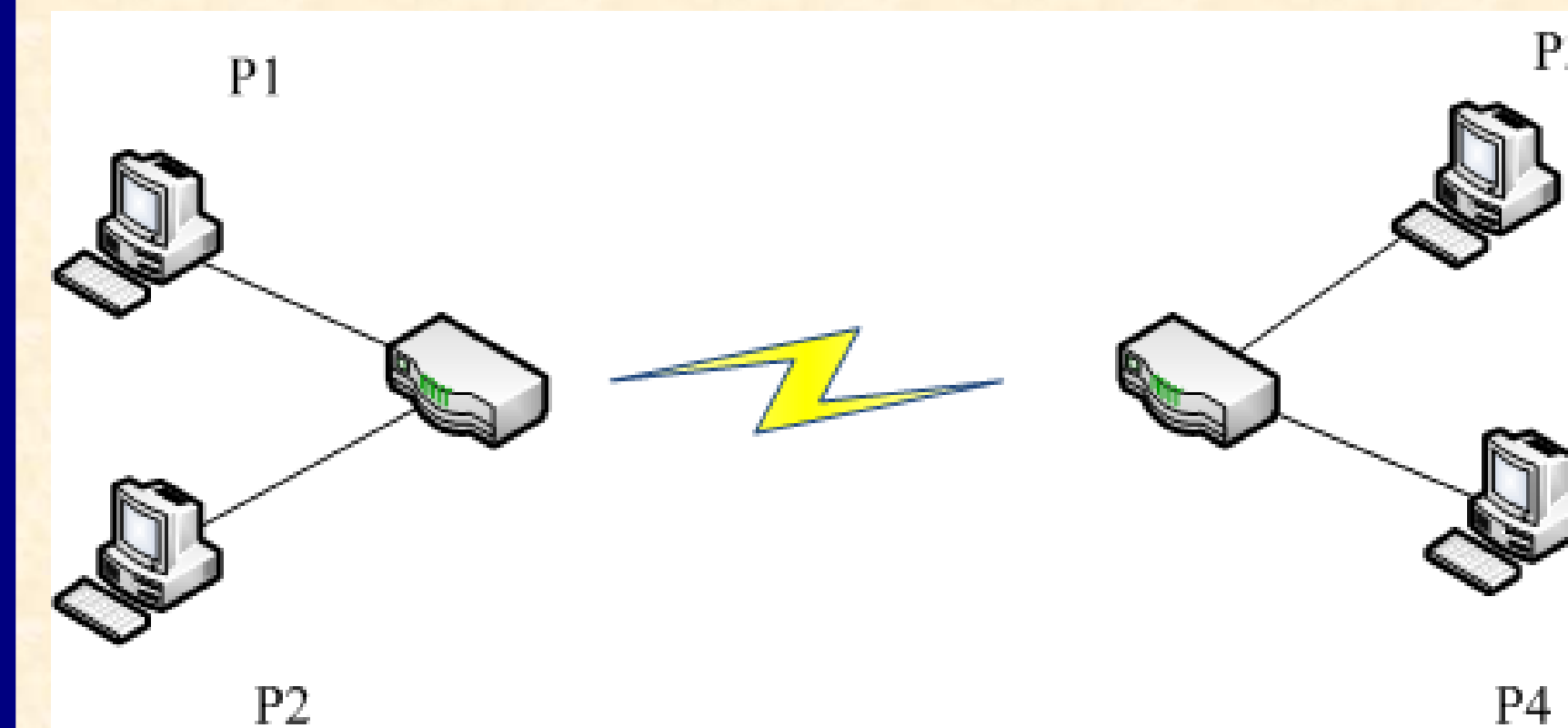
- **Problème d'optimisation des flots dans les réseaux**
 . $G=(V,A)$: *Grphe directive connectée*

- N : *ensemble de nœuds*
- $A \subseteq N \times N$: *ensemble d'arcs*
- c_{ij} : *fonction de coût*
- f_{ij} : *flux de l'arc (i,j)*

$$\min \sum_{(i,j) \in A} c_{ij}(f_{ij})$$

$$\text{sous } \sum_{(i,j) \in A} f_{ij} - \sum_{(m,j) \in A} f_{mj} = b_i, \forall i \in N.$$

. *Méthode itérative de résolution : Méthode de type gradient*



Topologie de réseau d'expérimentation

	Synchrone	Asynchrone	Hybride
Intra-cluster	Synchrone Fiable Com.	Asynchrone Fiable Com.	Synchrone Fiable Com.
Inter-cluster	Synchrone Fiable Com	Asynchrone Non fiable Com.	Asynchrone Non fiable Com.

Règles d'adaptation

Nombre de PCs	Méthode	Nombre d'itérations				Temps (s)	Accélération	Efficacité
		P1	P2	P3	P4			
1	-	399813	-	-	-	2135	-	-
2	Syn	400694	400694	-	-	1481	1,44	0,72
2	Asyn	385780	583735	-	-	1209	1,76	0,88
4	Syn	402056	402056	402056	402056	1241	1,72	0,43
4	Hybrid	449372	449372	398421	398421	935	2,28	0,57
4	Asyn	419175	389144	464128	743636	656	3,25	0,81

Résultats d'expérimentation

CONCLUSIONS

Travaux effectués:

- *Conception du protocole de communication auto-adaptatif.*
- *Implémentation du protocole en utilisant le framework Cactus.*
- *Expérimentation du problème d'optimisation des flots dans les réseaux.*

Travaux futurs:

- *Langage de spécification pour les règles d'adaptation.*
- *Conception de l'environnement décentralisé.*
- *Expérimentation d'autres problèmes en mathématique financière.*