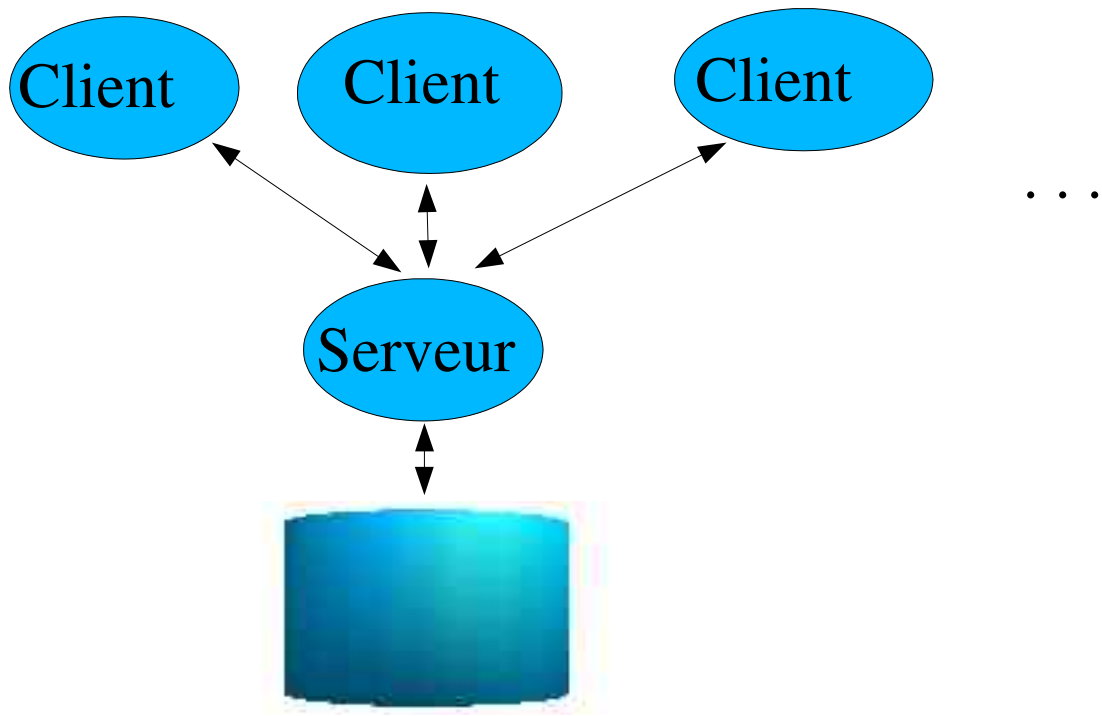
Contexte



- Sur mêmes données
- Pas de parallélisme (temps partagé)

Le problème : exemple

- Réservations : nom indéfini si place libre

Nom	N°place
Toto	1
Titi	2
Tutu	3
null	4
null	5

- Jules et Jim veulent réserver chacun une place

Prochaine place libre

next_libre() =

select n°place

from resa

where nom is null

= -1 si complet

n (libre) sinon

Programme de réservation pour le client c

Si libre = (next_libre() <> -1)

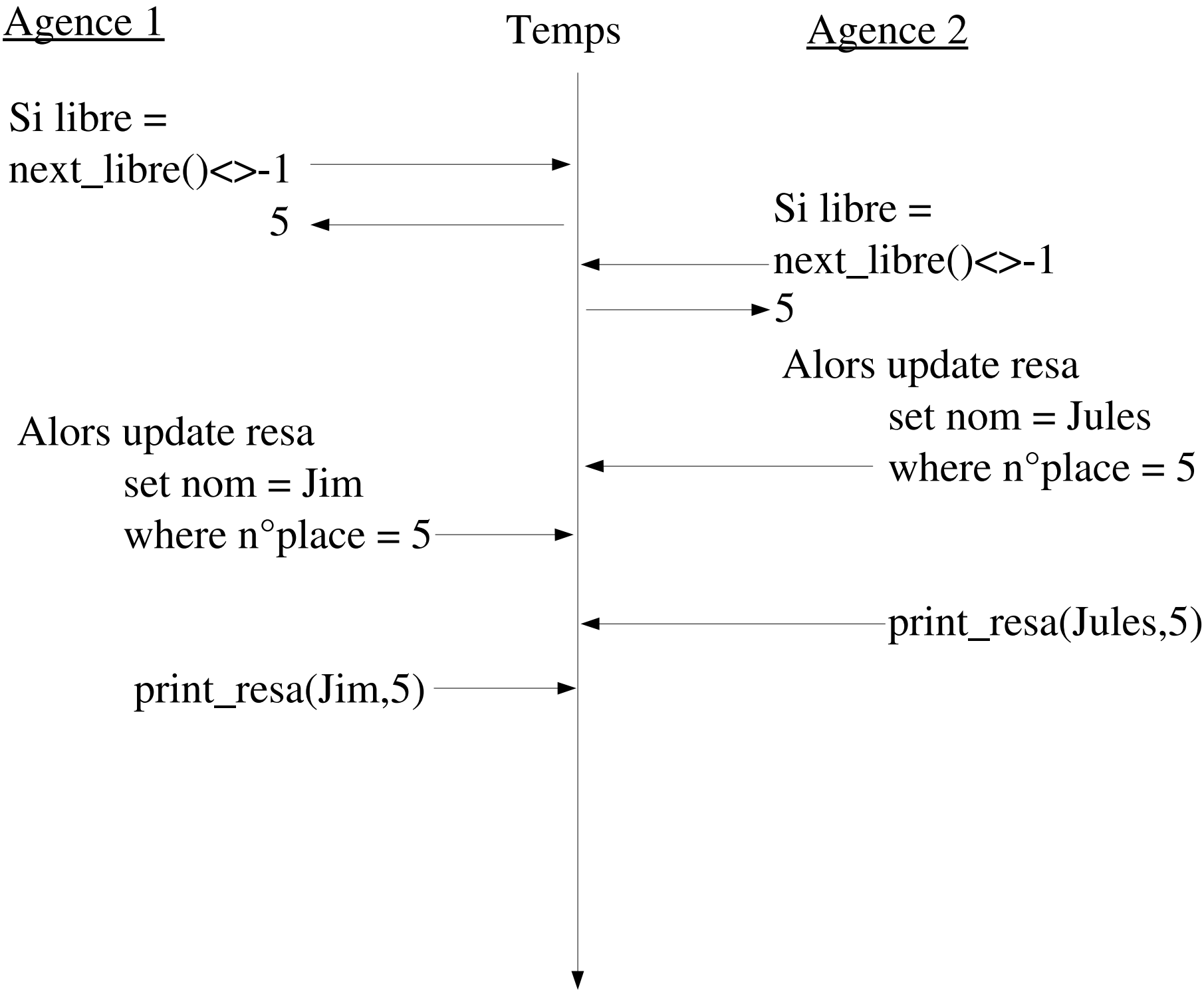
Alors

    update resa

    set nom = c

    where n°place = libre;

    print_resa(c, libre);


Agence 1
Temps
Agence 2
Si libre =
next_libre()<>-1
5
Si libre =
next_libre()<>-1
5
Alors update resa
set nom = Jules
where n°place = 5
Alors update resa
set nom = Jim
where n°place = 5
print_resa(Jules,5)
print_resa(Jim,5)

Problème

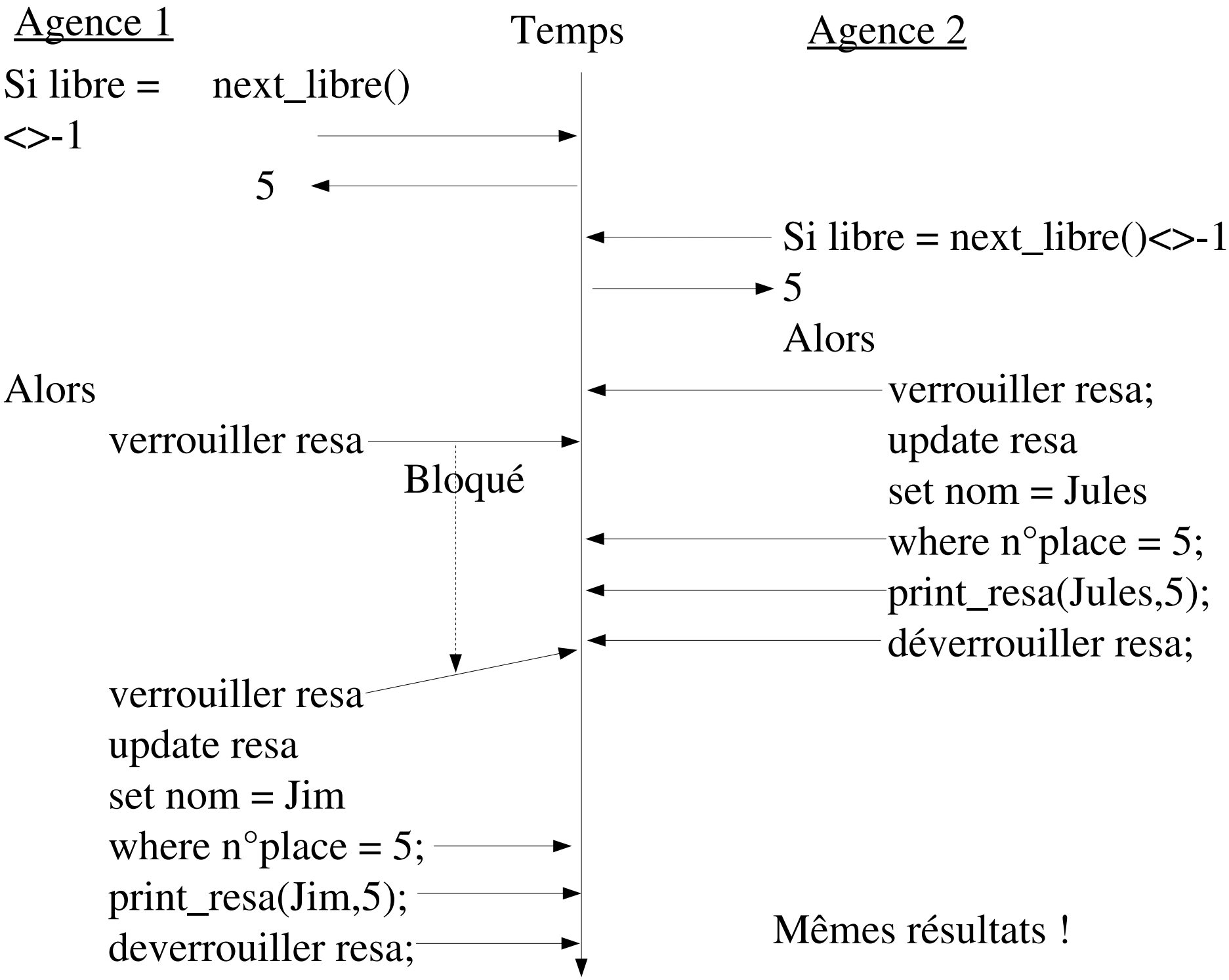
- Conséquence
  - 2 réservations imprimées pour la même place
  - 1 seule dans la table
  - Clients pas au courant
  - Incohérence print / table
- Vocabulaire
  - Situation de concurrence
  - Contrôle de concurrence

Que s'est-il passé ?

- Chaque client comporté comme si seul : données accédées inchangées
  - Lues pour décider
  - Mises à jour
- Solution : forcer «seul» et «inchangées»
- Outil : verrous

Solution « naturelle »

```
Si libre = (next_libre() <> -1)
Alors
    verrouiller resa;
    update resa
    set nom = c
    where n°place = libre;
    print_resa(c, libre);
    déverrouiller resa;
```


Agence 1
Temps
Agence 2
Si libre = <>-1
next_libre()
5
Si libre = next_libre()<>-1
5
Alors
Alors
verrouiller resa;
update resa
set nom = Jules
where n°place = 5;
print_resa(Jules,5);
déverrouiller resa;
verrouiller resa
Bloqué
verrouiller resa
update resa
set nom = Jim
where n°place = 5;
print_resa(Jim,5);
deverrouiller resa;
Mêmes résultats !

La bonne : coopérative

verrouiller resa;

Si libre = (next_libre() <> -1)

Alors

update resa

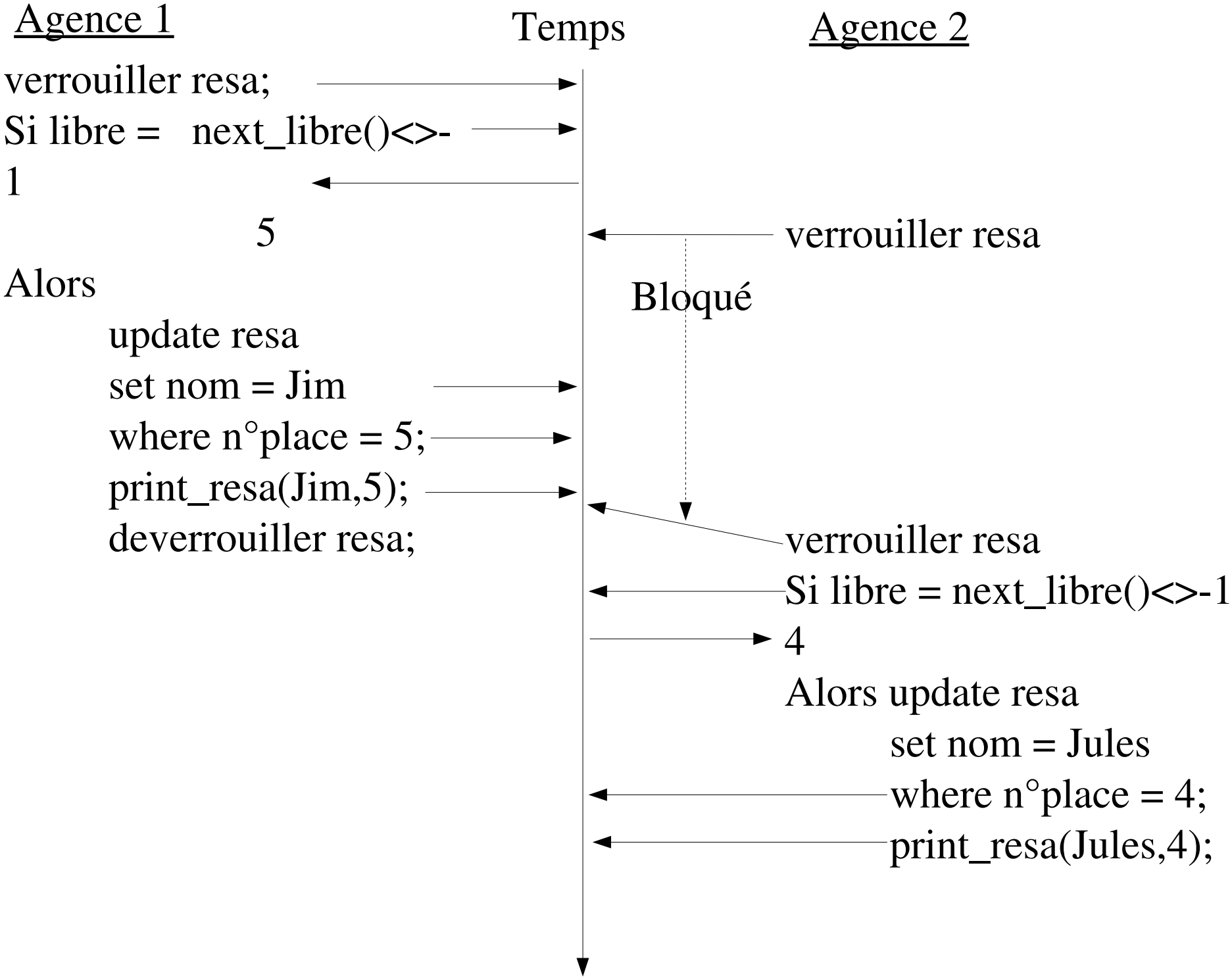
set nom = c

where n°place = libre;

print_resa(c, libre);

déverrouiller resa;

- Pour toutes les agences


Agence 1
Temps
Agence 2
verrouiller resa;
Si libre = next_libre()<>-1
5
verrouiller resa
Bloqué
Alors
update resa
set nom = Jim
where n°place = 5;
print_resa(Jim,5);
deverrouiller resa;
verrouiller resa
Si libre = next_libre()<>-1
4
Alors update resa
set nom = Jules
where n°place = 4;
print_resa(Jules,4);