



UNIVERSITÉ de Picardie

Jules Verne

Algorithmique distribuée stabilisante

Présentation du 22/4/2010

Par Alain Cournier

Plan de l'exposé

1. Plantons le décor
 - a. Qu'est-ce qu'un réseau ?
 - b. Qu'est-ce qu'un algorithme stabilisant ?
 - c. Quel modèle allons-nous utiliser ?
 - d. Comment mesurer le temps ?
2. Notion de service : Comment évaluer la qualité du service rendu par un algorithme?
3. Perspectives.

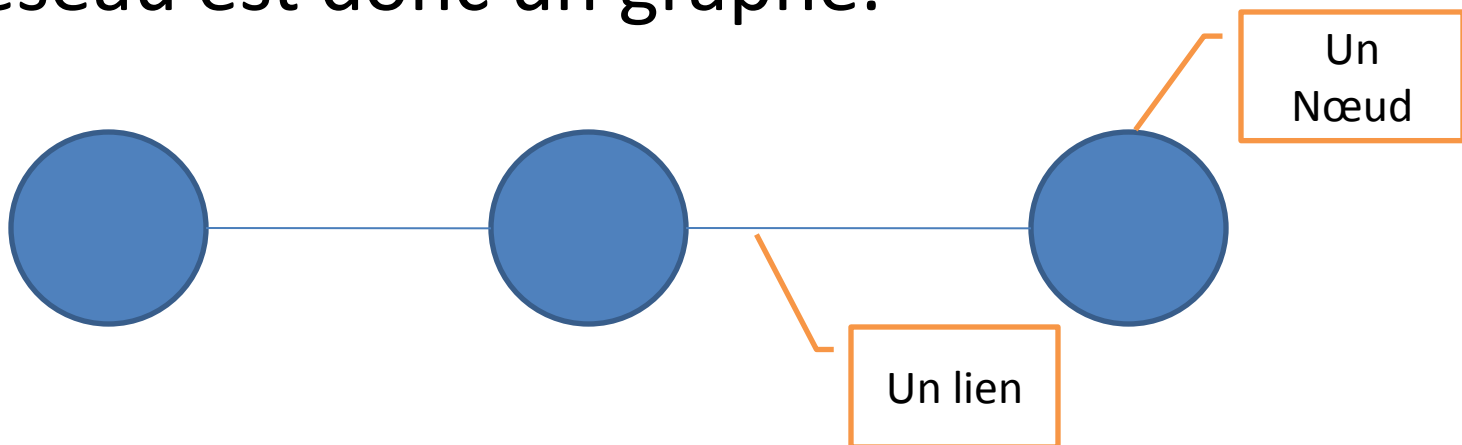
Plantons le décor :

Qu'est-ce qu'un réseau ?

Un réseau est :

- Un ensemble de nœuds où sont stockés des valeurs et où les calculs sont effectués;
- Un ensemble de liens par lesquels deux nœuds voisins peuvent échanger des informations.

Un réseau est donc un graphe.



Plantons le décor :

Qu'est-ce qu'un algorithme distribué ?

- Sur les nœuds, des programmes s'exécutent.
- Ces programmes peuvent réaliser :
 - des opérations locales;
 - faire des calculs en collaboration avec d'autres machines, on parle alors d'algorithme distribué.

Plantons le décor : Qu'est-ce qu'un algorithme stabilisant ?

- Un algorithme auto-stabilisant converge en un temps fini vers le comportement attendu, indépendamment de son état initial (Dij 74).
- Partant d'une configuration quelconque, un algorithme instantanément stabilisant répond toujours aux spécifications du problème à résoudre (BDPV 99).

Plantons le décor :

Quel modèle allons-nous utiliser ?

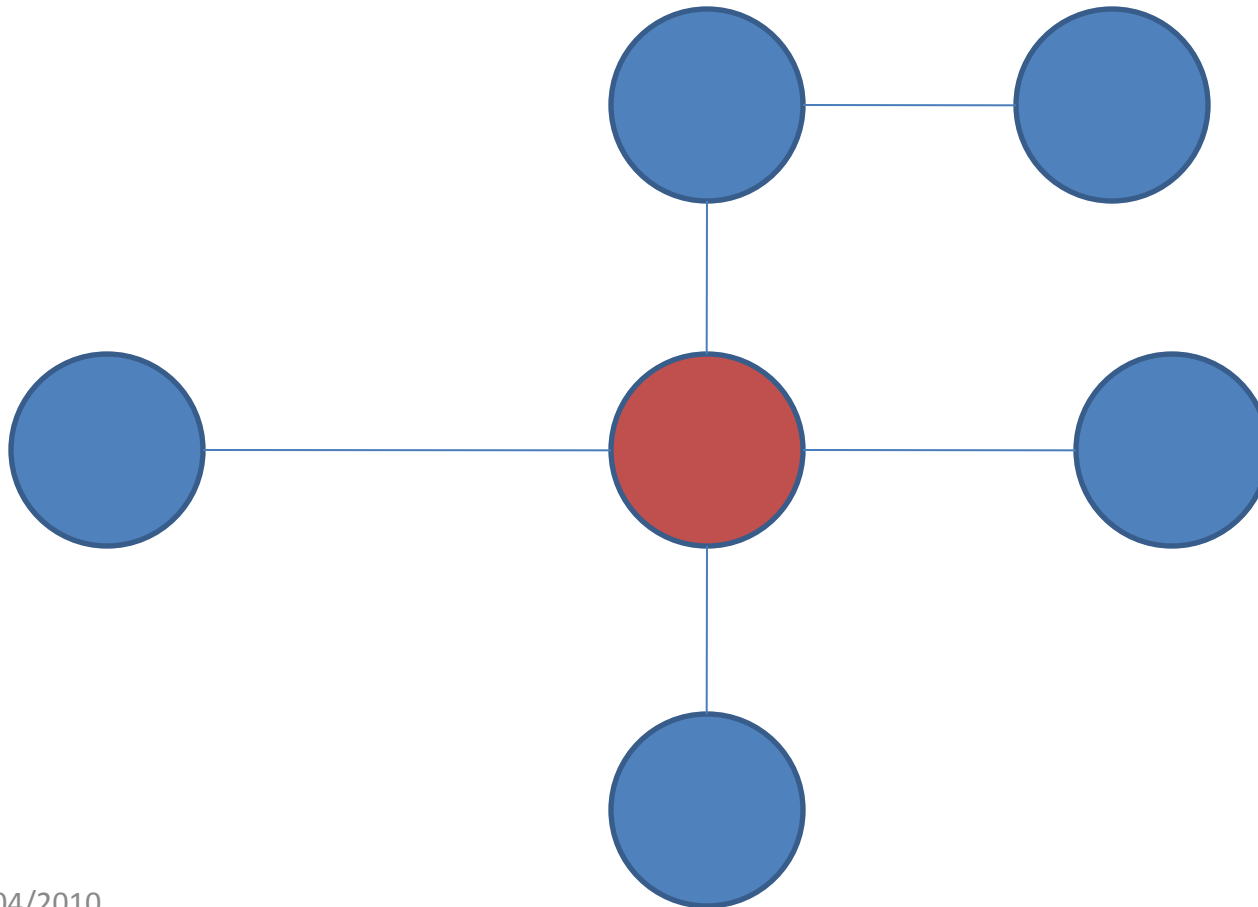
Plusieurs modèles existent :

- Le modèle à mémoire partagée ou modèle à états.
- Le modèle à registres.
- Le modèle à passage de messages.

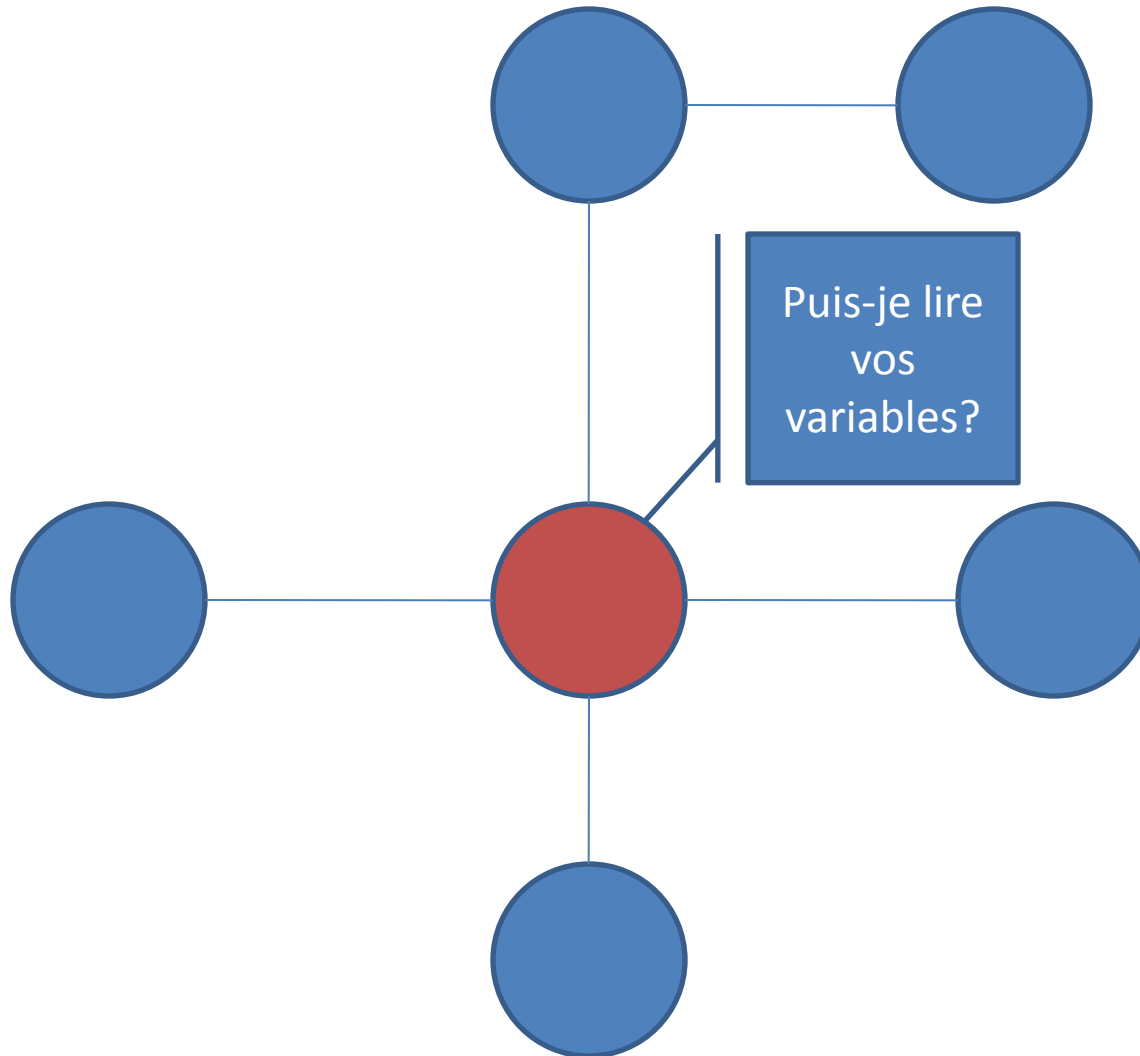
Plantons le décor :

Quel modèle allons-nous utiliser ?

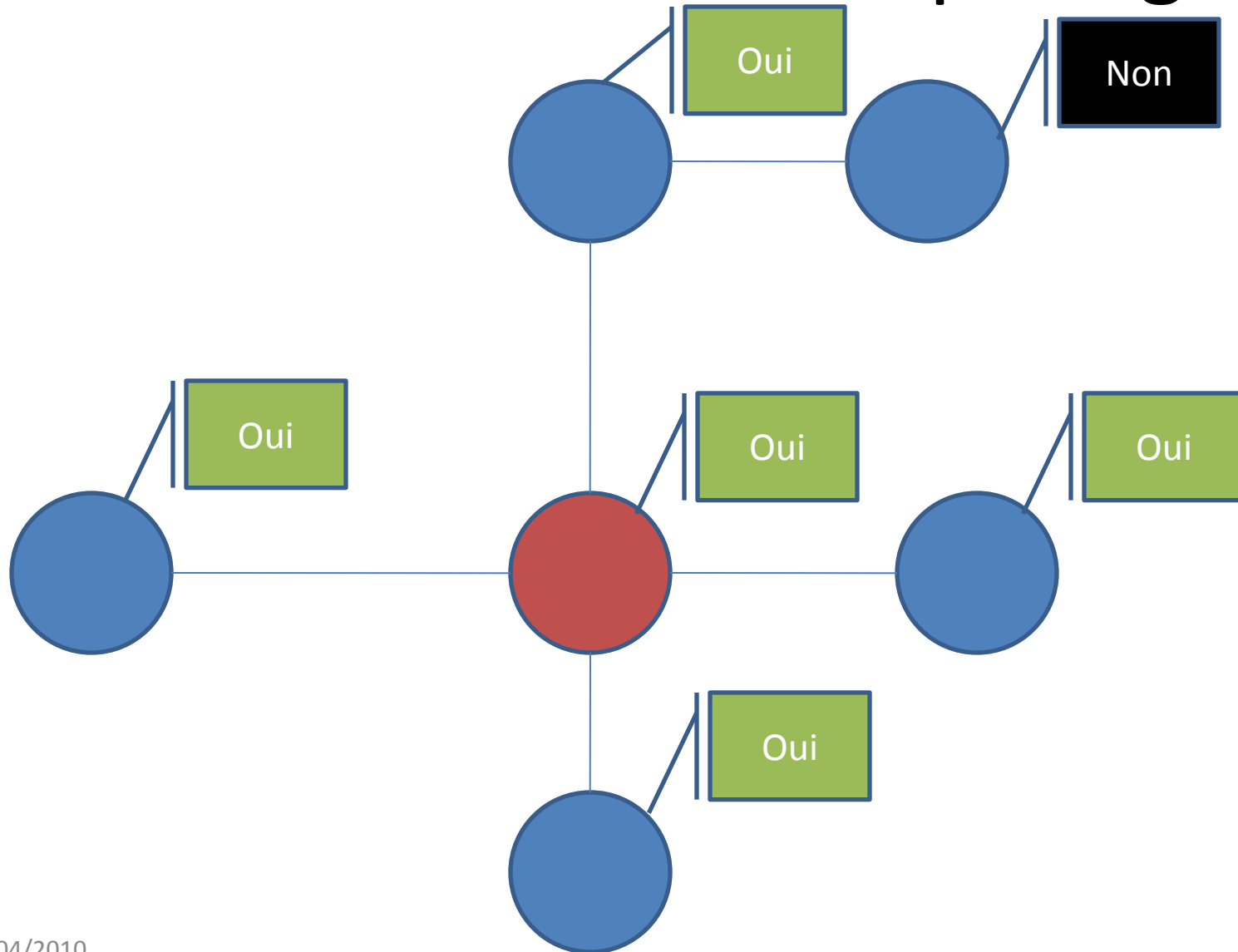
- Le Modèle à mémoire partagée



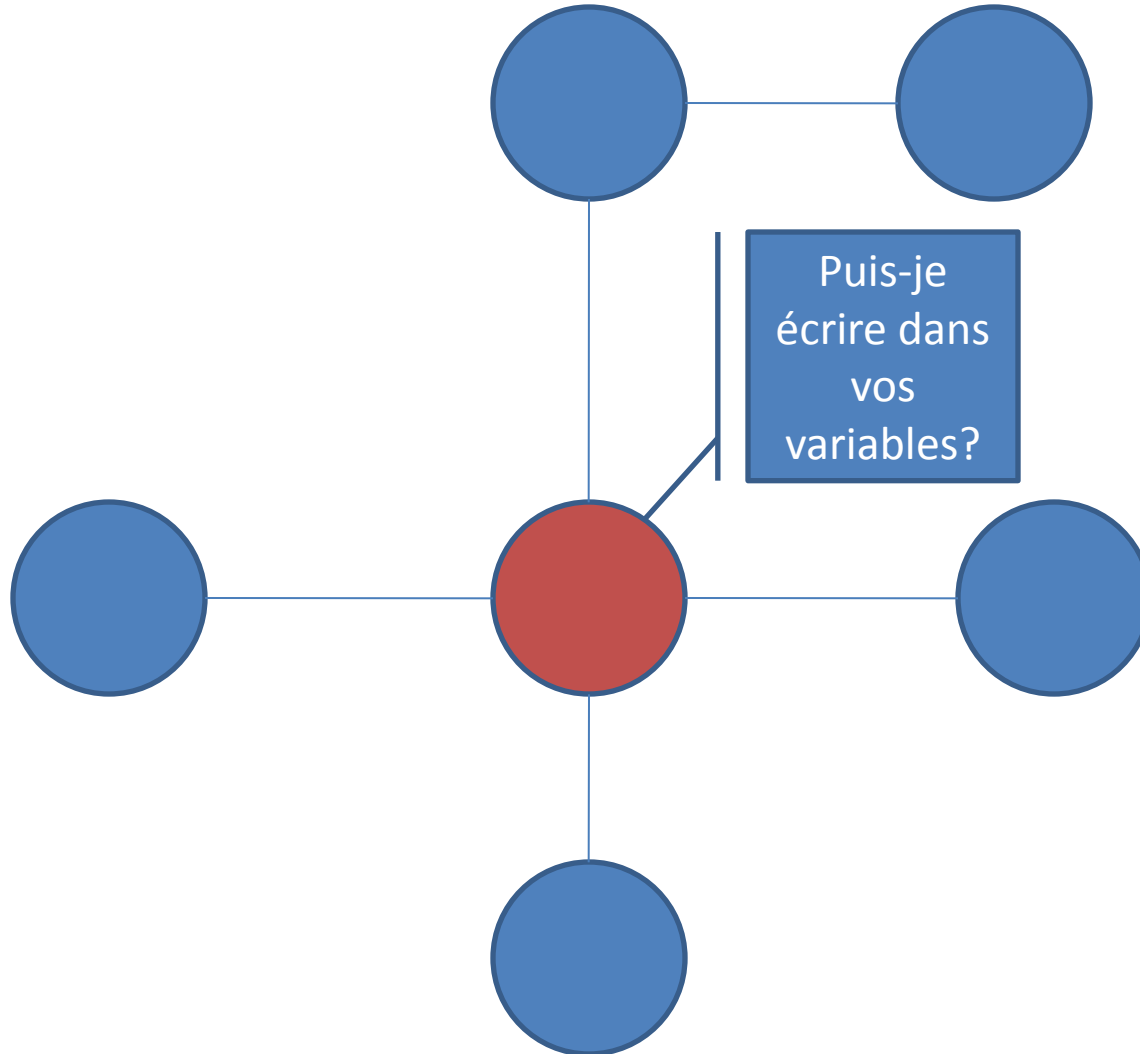
Le Modèle à mémoire partagée



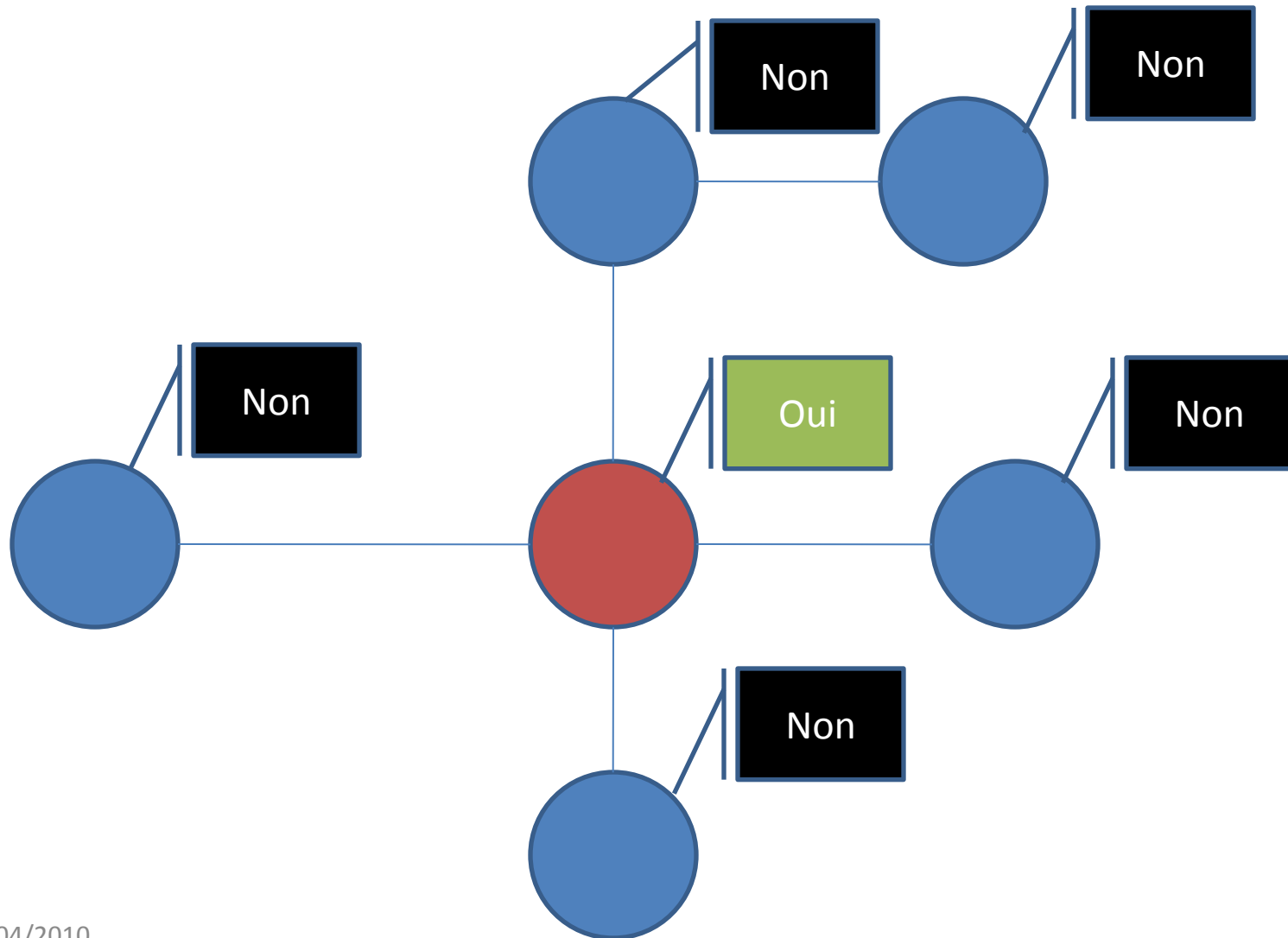
Le Modèle à mémoire partagée



Le Modèle à mémoire partagée



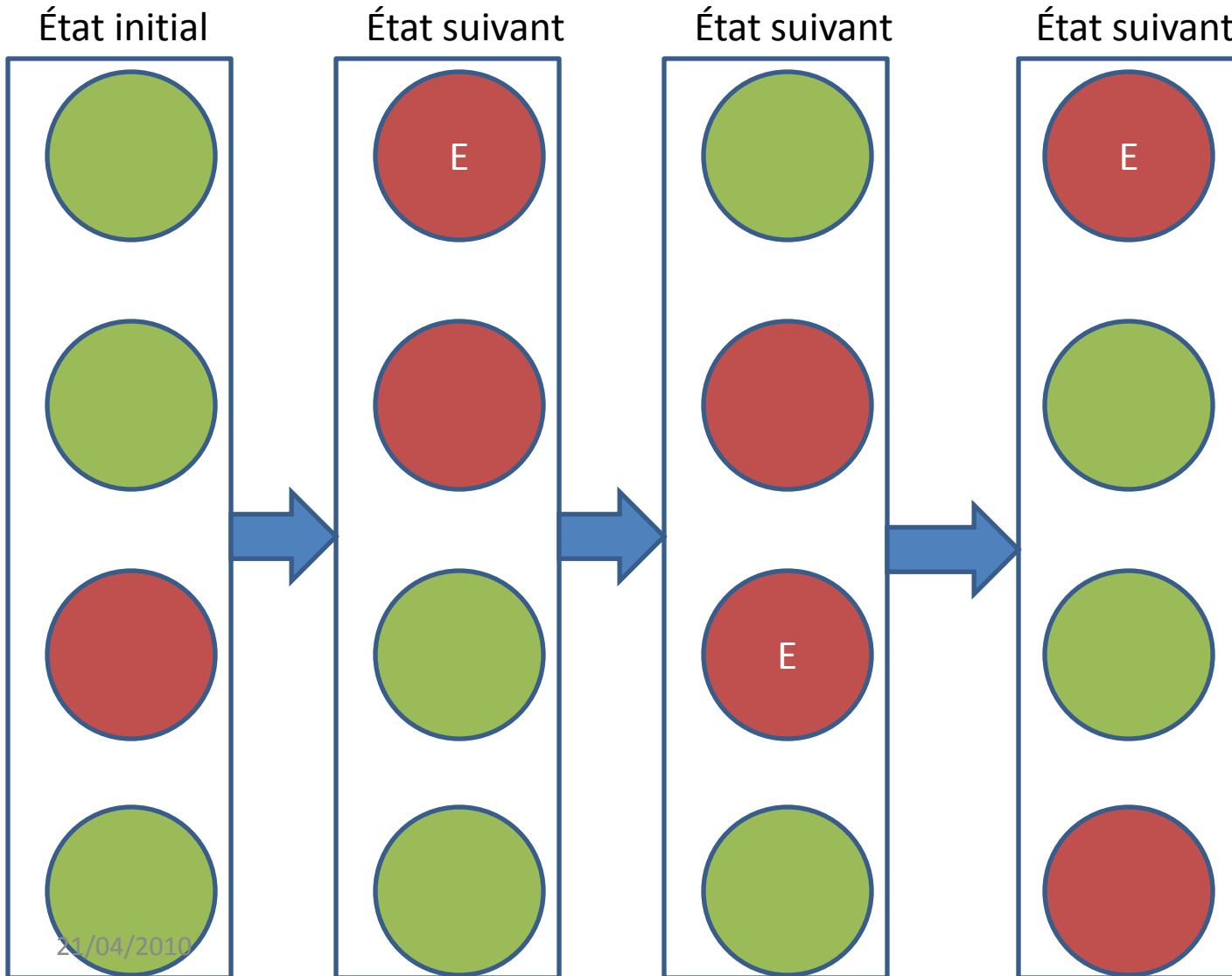
Le Modèle à mémoire partagée



Plantons le décor : Comment mesurer la complexité en temps ?

- En termes de nombre de pas de calcul : C'est le nombre de pas de calcul durant une exécution, dans le pire des cas.
- En termes de nombre de rondes: Cette mesure prend en compte la vitesse du plus lent des processeurs durant une exécution, dans le pire des cas (DIM 93).

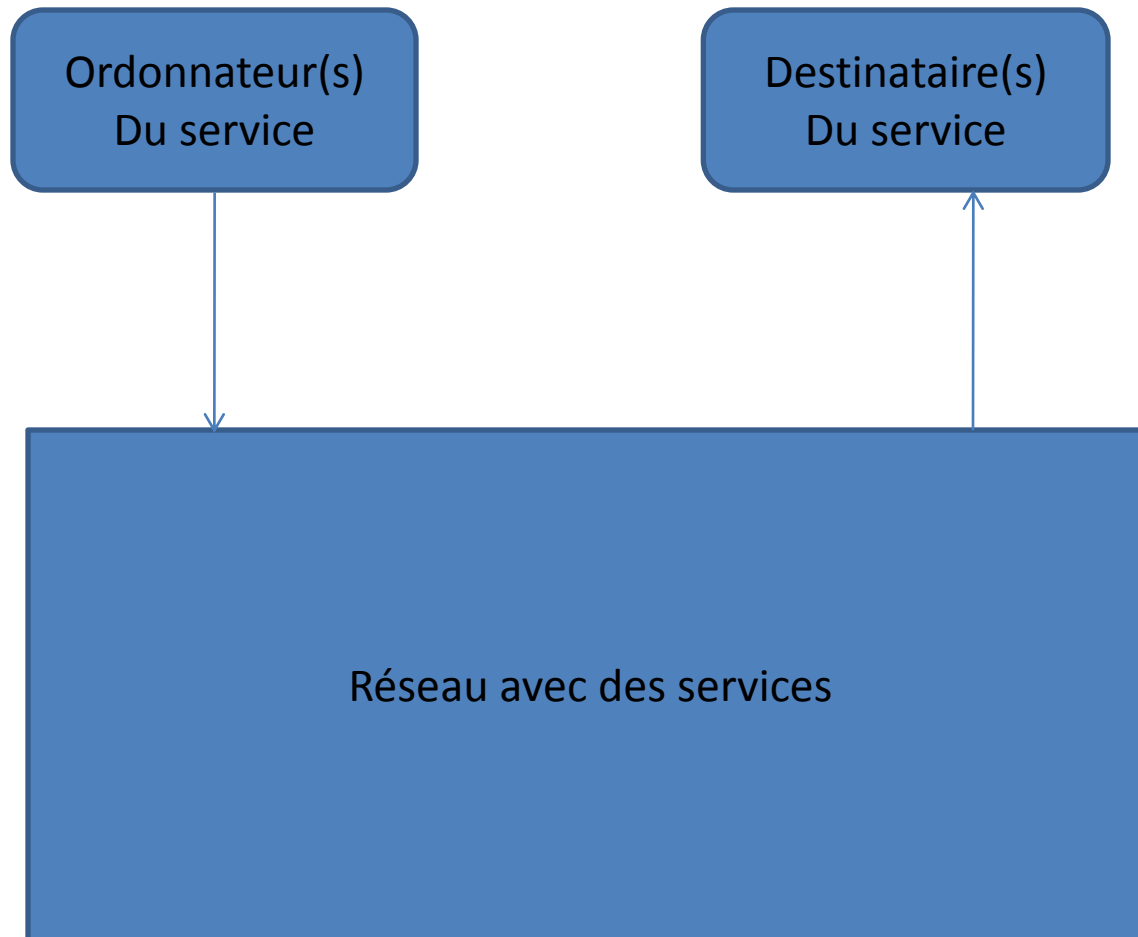
La ronde



Notion de service rendu

- Si un algorithme est exécuté, c'est qu'il doit rendre un service.
 - Ce service est défini par des spécifications;
- Pour nos algorithmes on peut distinguer :
 - L'ordonnateur du service;
 - Le(s) client(s) du service;
 - Les relais du service (réseau).

Notion de service



Mesures de qualité du service

- Coté Réseau : Limiter la charge (Le temps, le nombre de messages, l'espace mémoire...)
- Coté ordonnateur(s) : Limiter le nombre de fois où le service doit être « actionné » pour être sûr de sa réalisation.
- Coté destinataire(s) : Limiter le nombre de fois où le service peut être rendu de façon inappropriée.

Un exemple : la Propagation d'Information avec Accusé-Réception

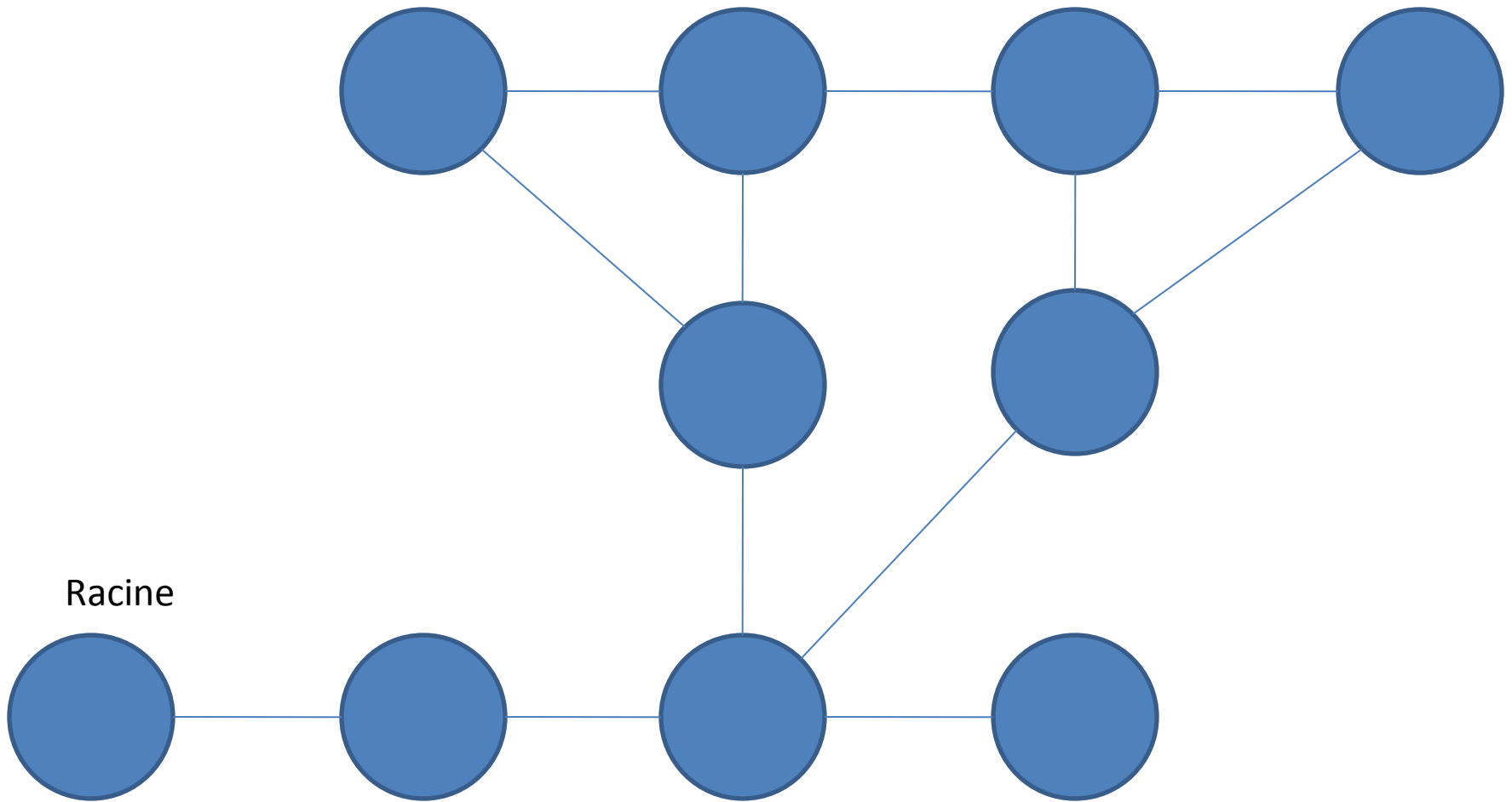
Spécifications du problème

- L'initiateur du protocole initie la vague en diffusant un message m ;
- La décision (dernière action de la vague) est prise au nœud racine;
- La racine prend la décision associée à la vague si et seulement si tous les nœuds du réseau ont accusé réception du message m .

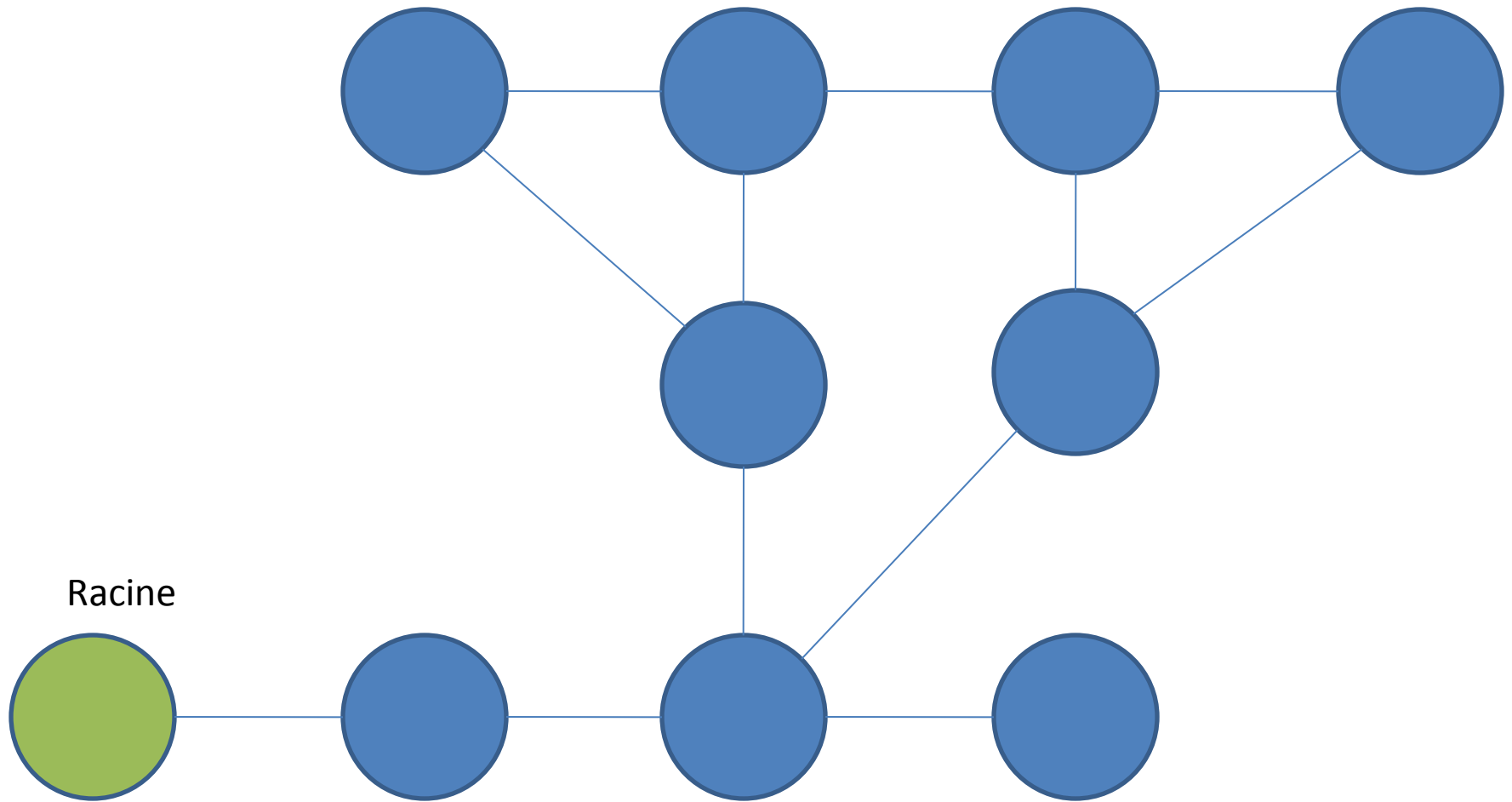
Un exemple : la Propagation d'Information avec Accusé-Réception

- Principes des algorithmes :
 1. Construire un arbre couvrant du réseau et diffuser le message sur cet arbre.
 2. Lors de la détection de la fin de la construction de l'arbre, envoyer l'accusé réception.
 3. Faire un nettoyage de la structure pour la réalisation de la phase suivante.
 4. Neutraliser les erreurs éventuelles.

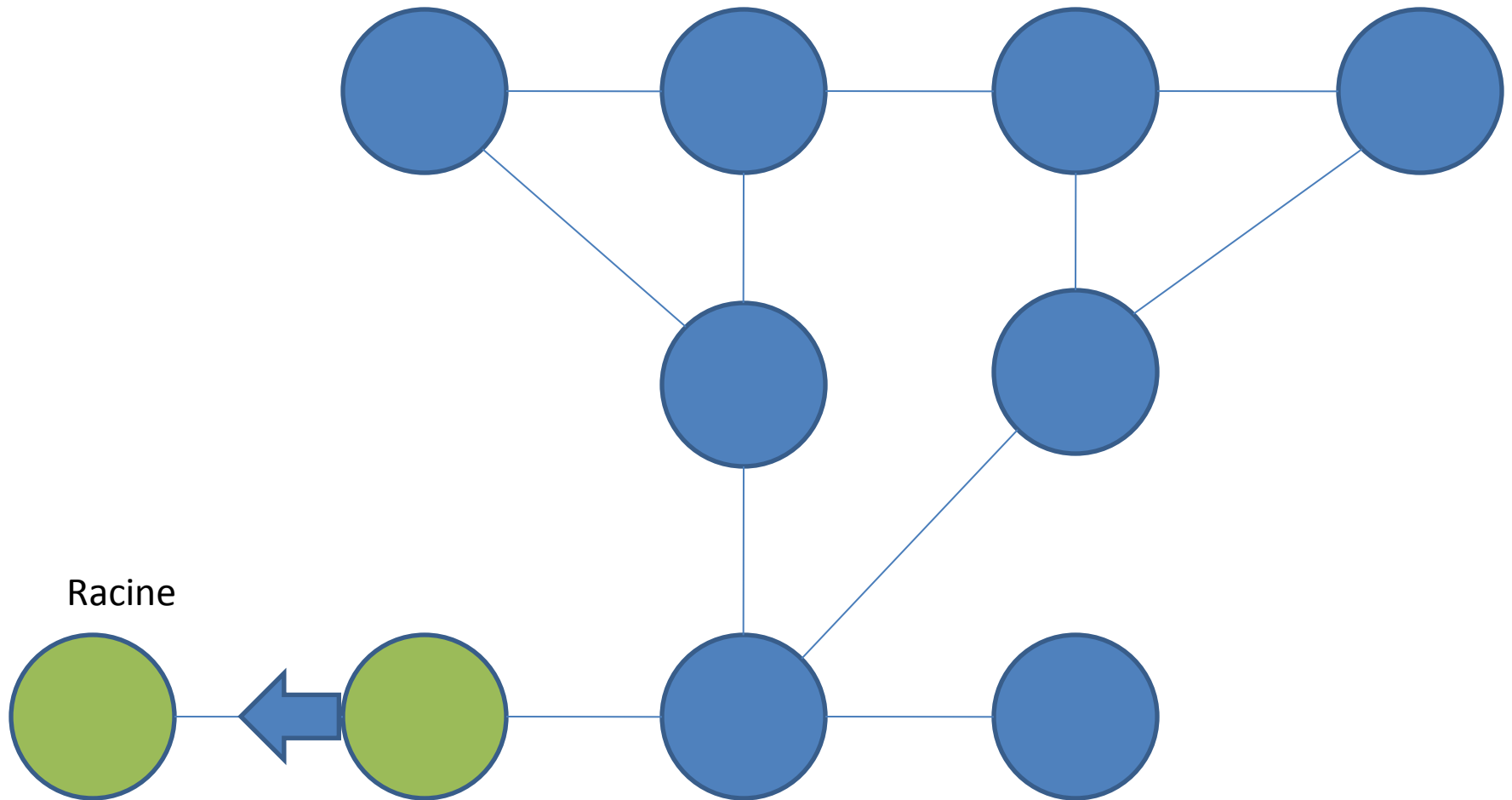
Un exemple : la Propagation d'Information avec Accusé-Réception



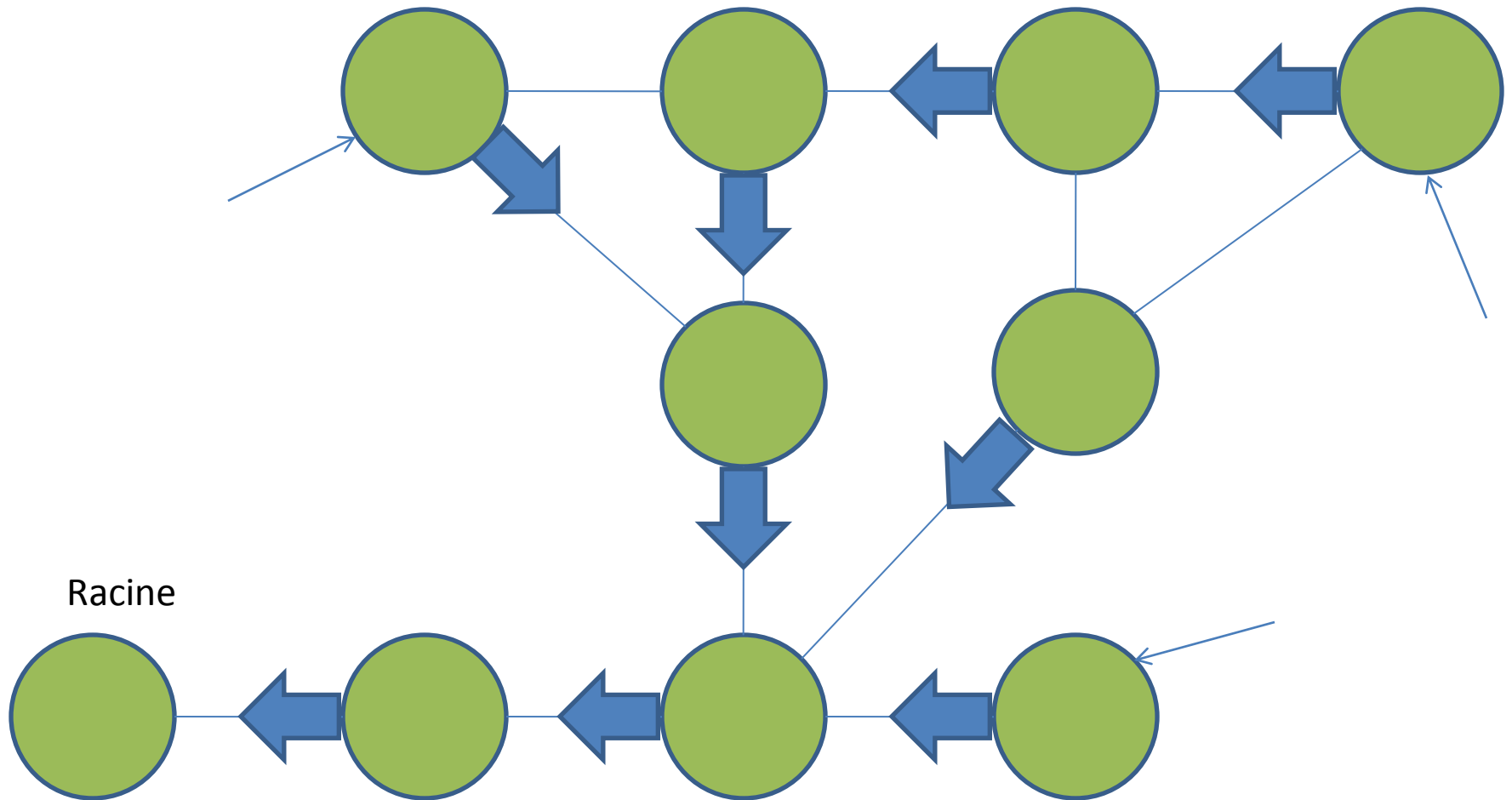
Un exemple : la Propagation d'Information avec Accusé-Réception



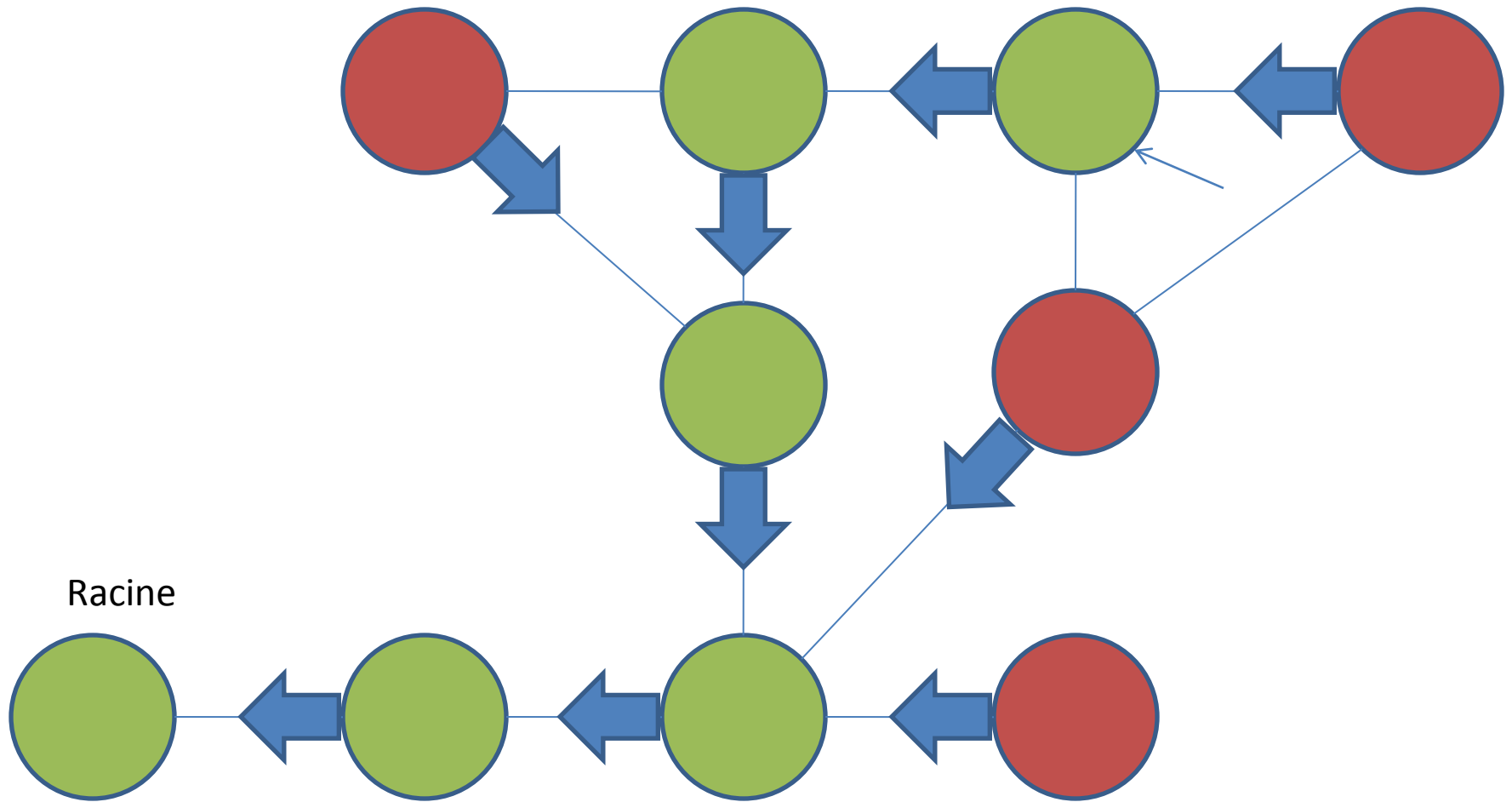
Un exemple : la Propagation d'Information avec Accusé-Réception



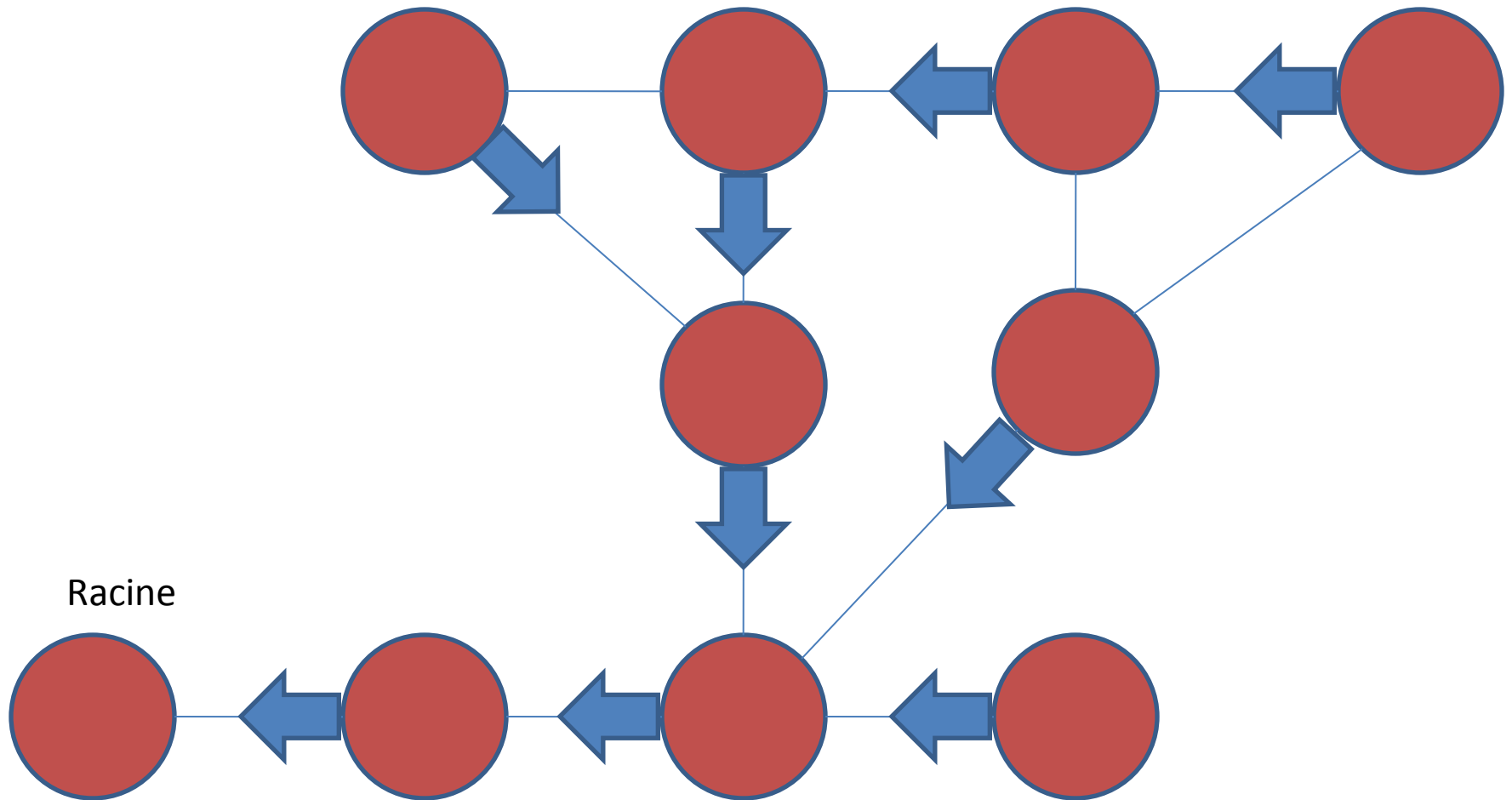
Un exemple : la Propagation d'Information avec Accusé-Réception



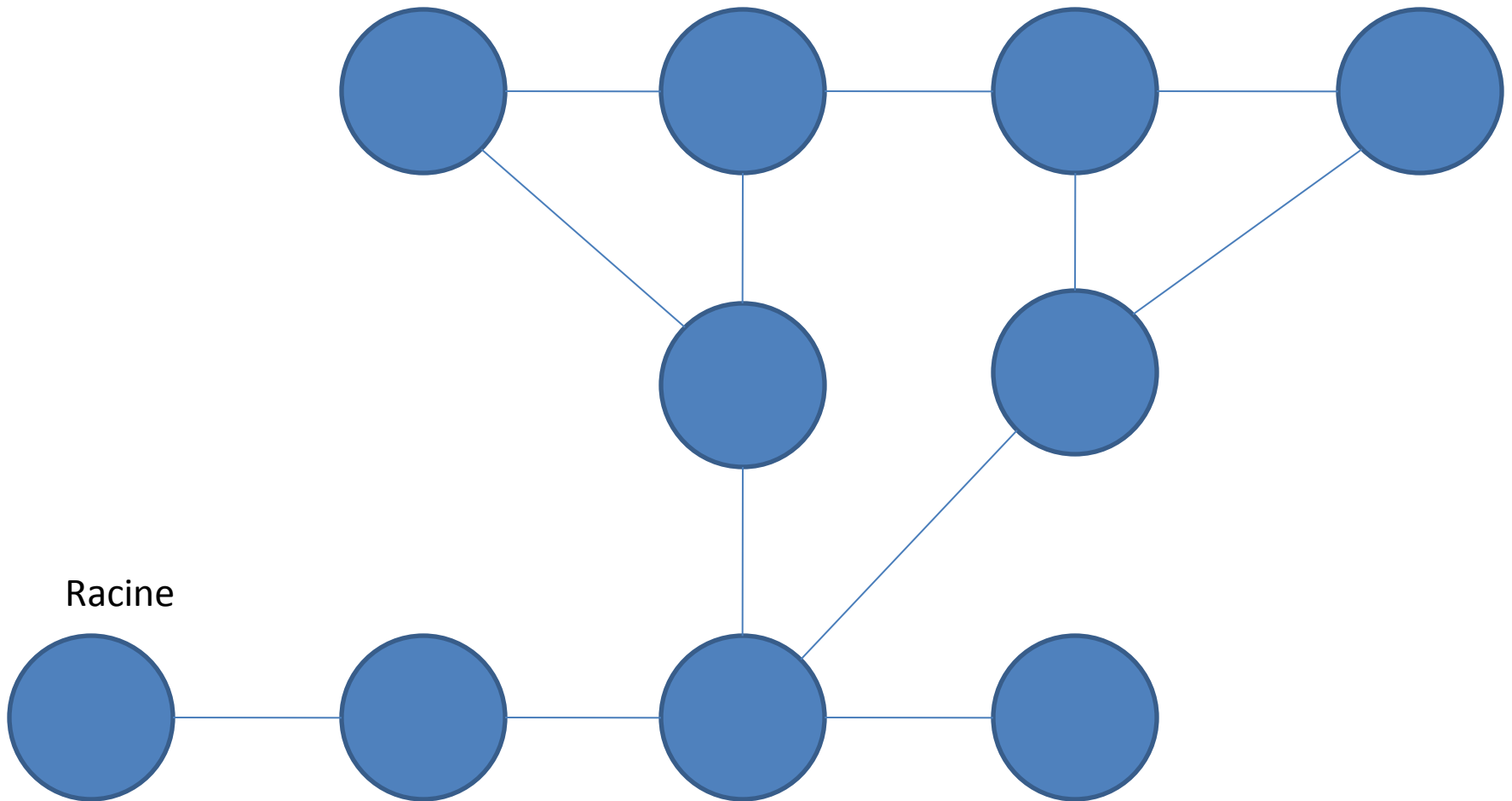
Un exemple : la Propagation d'Information avec Accusé-Réception



Un exemple : la Propagation d'Information avec Accusé-Réception

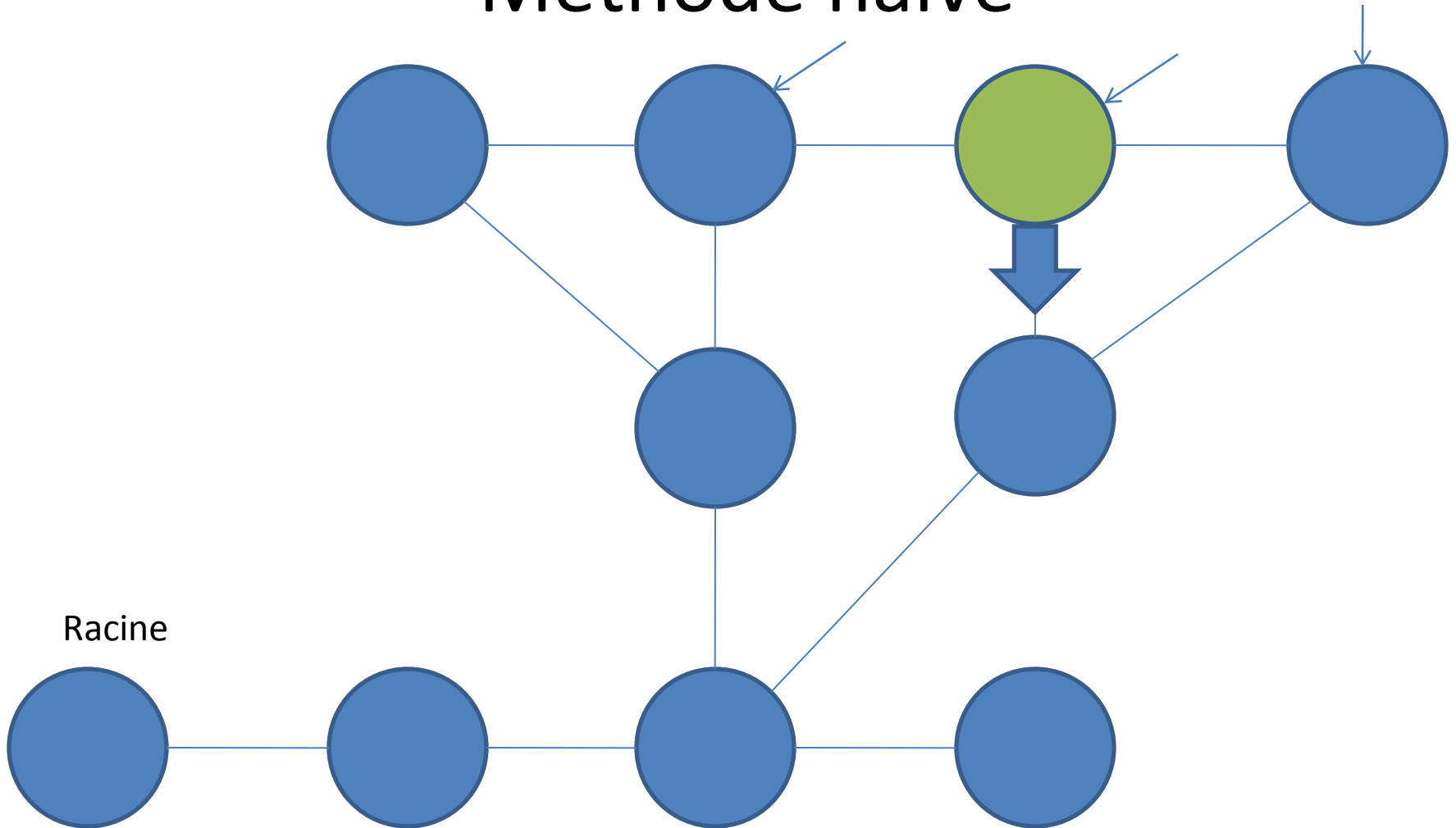


Un exemple : la Propagation d'Information avec Accusé-Réception

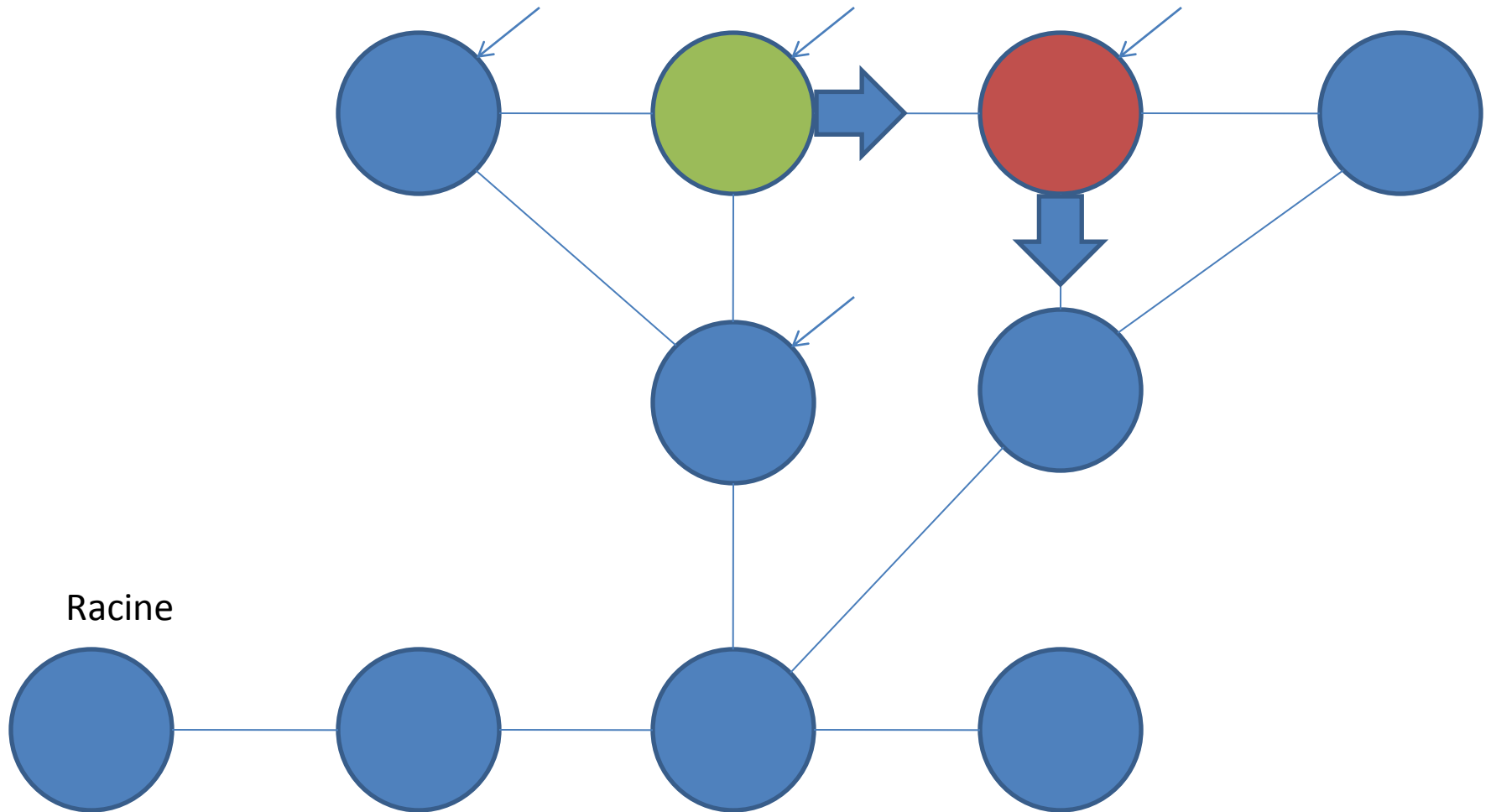


Racine

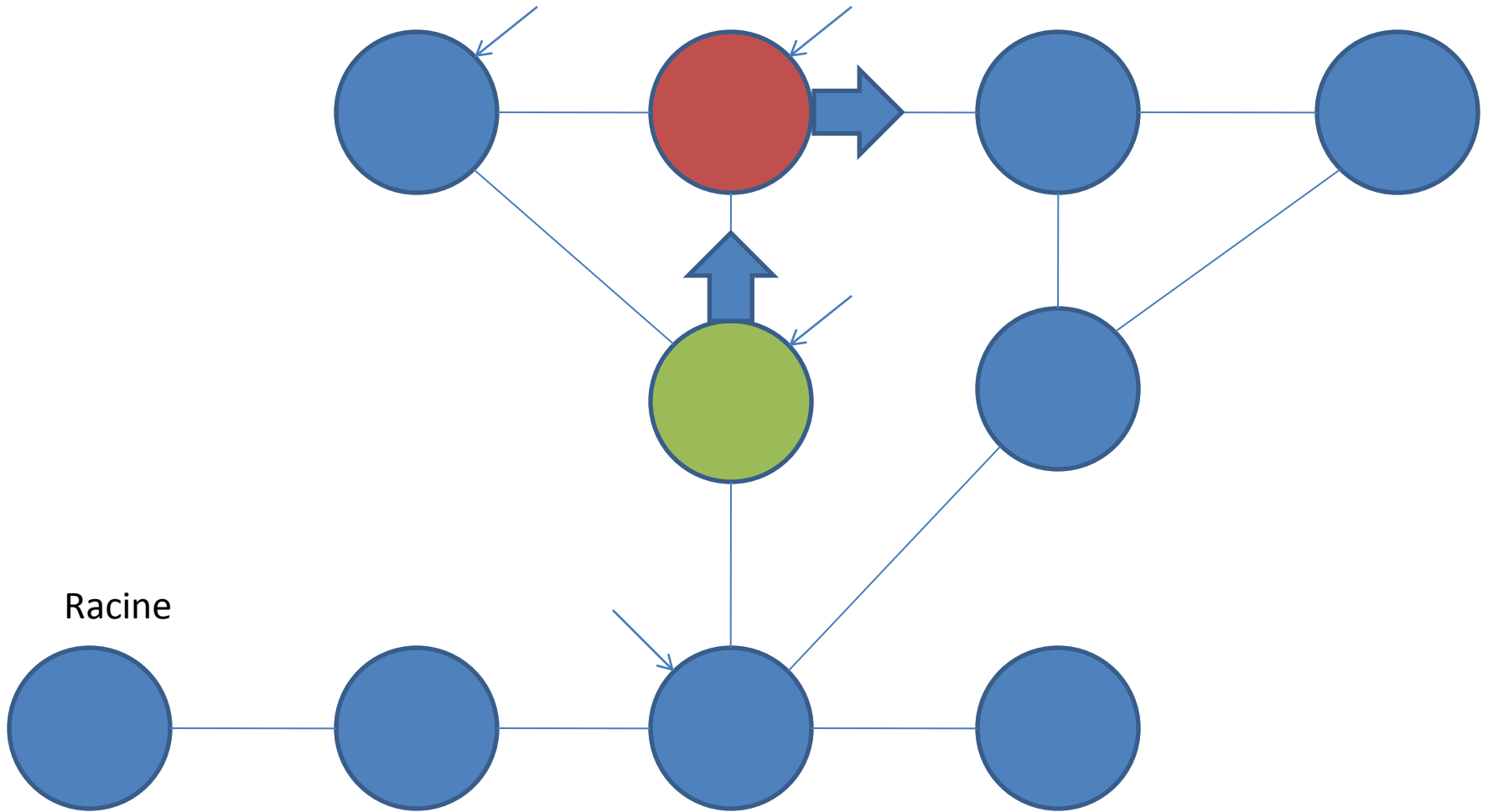
La gestion d'erreurs : Méthode naïve



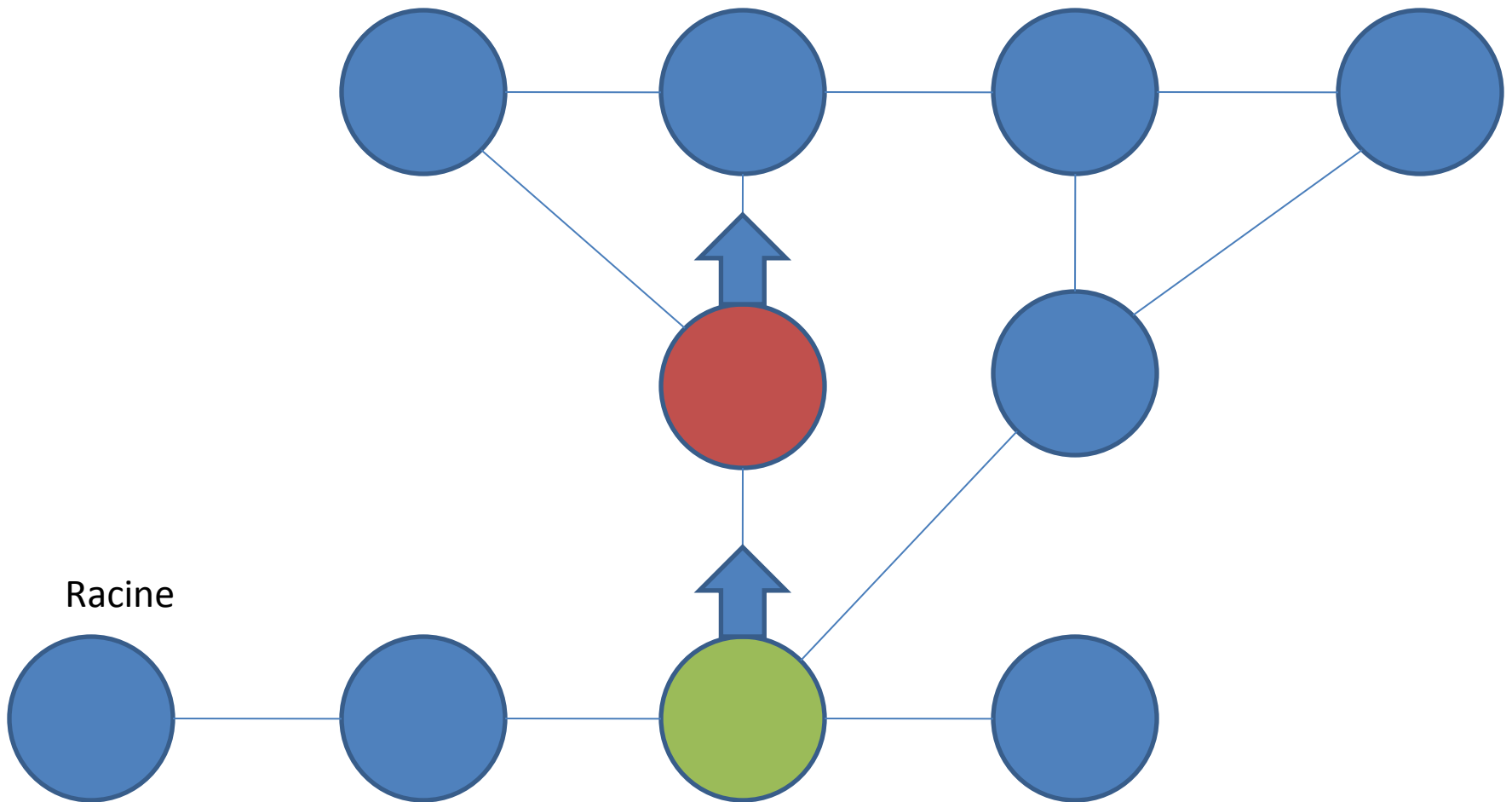
La gestion d'erreurs : Méthode naïve



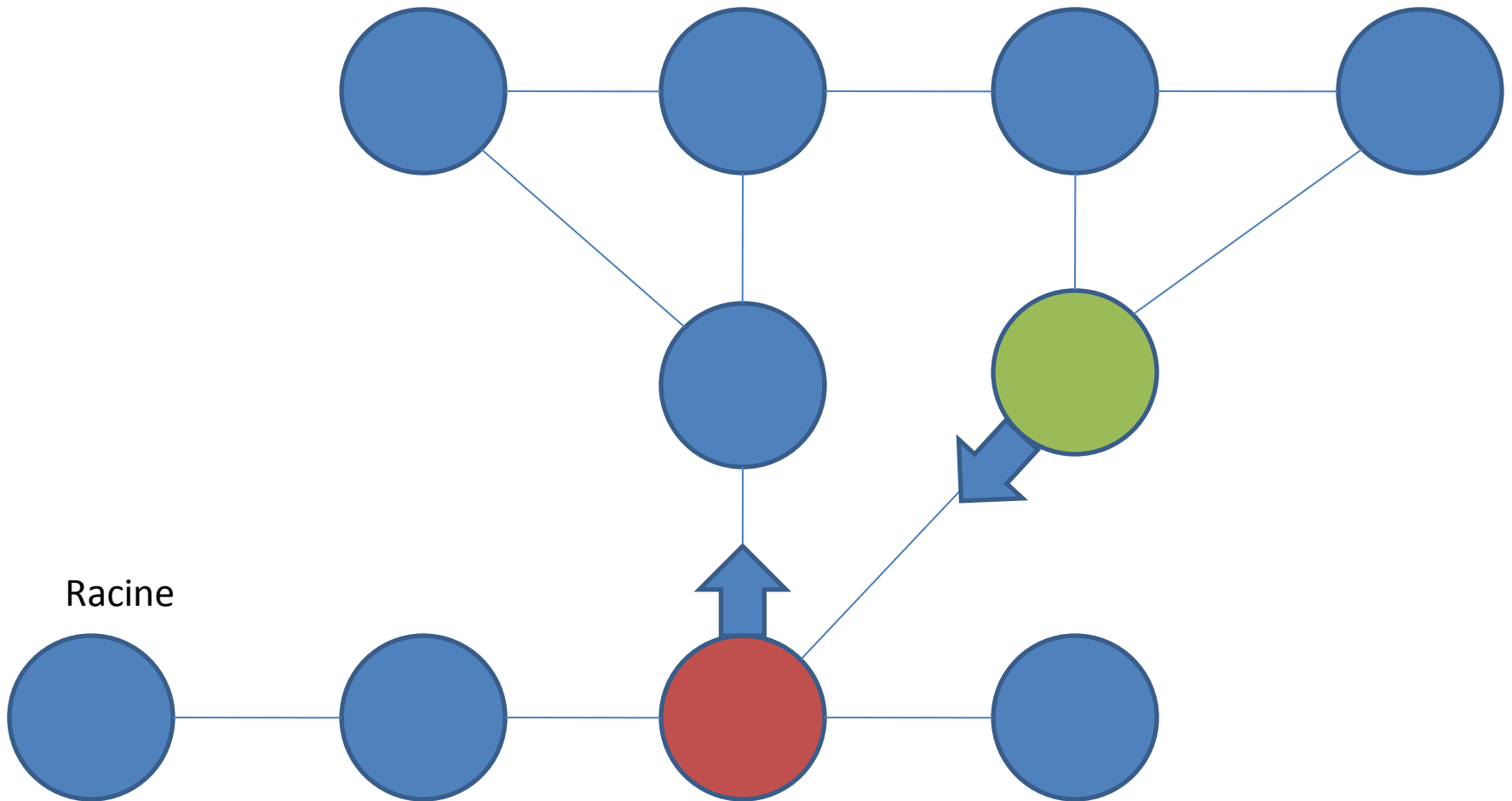
La gestion d'erreurs : Méthode naïve



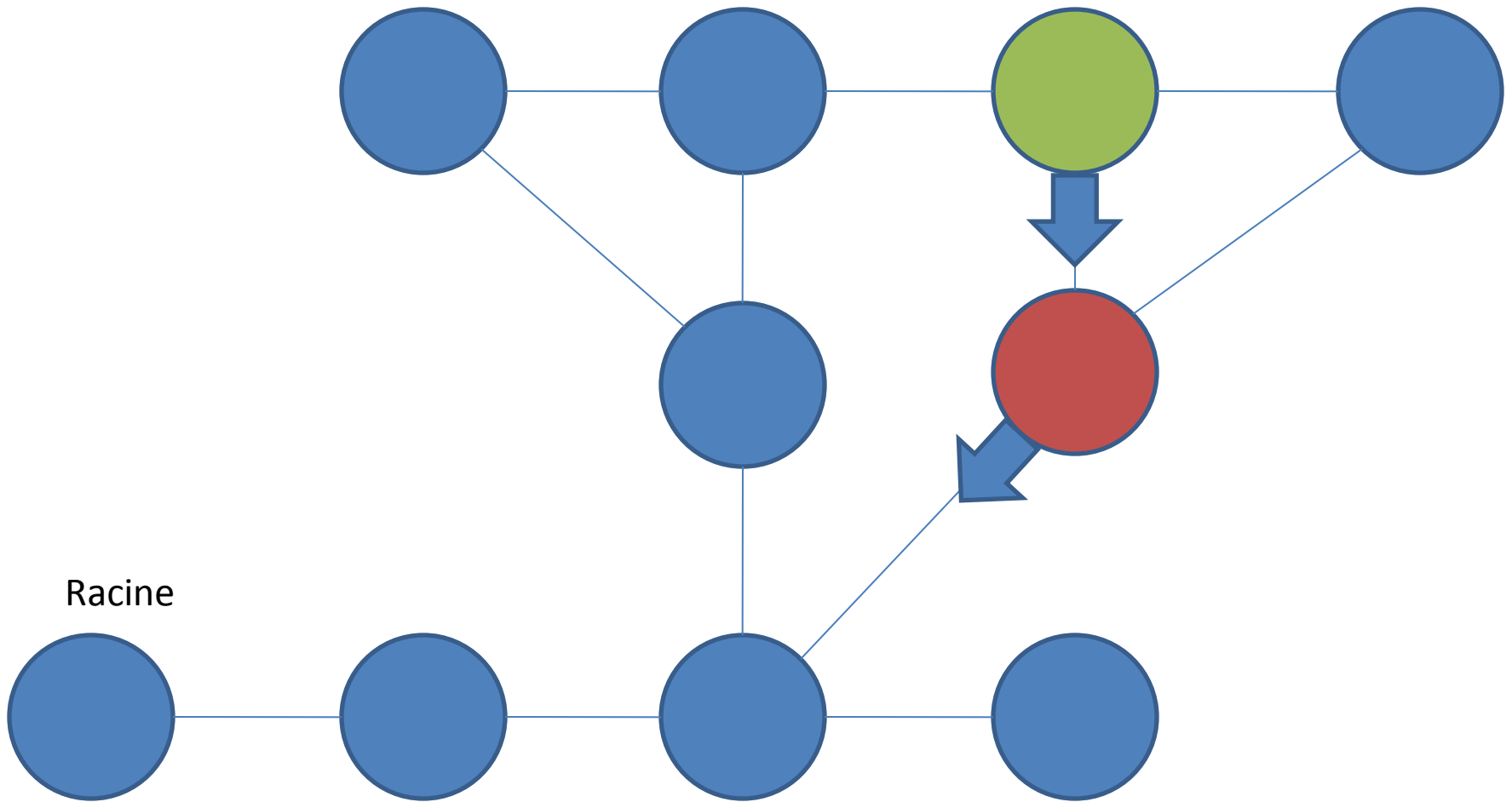
La gestion d'erreurs : Méthode naïve



La gestion d'erreurs : Méthode naïve



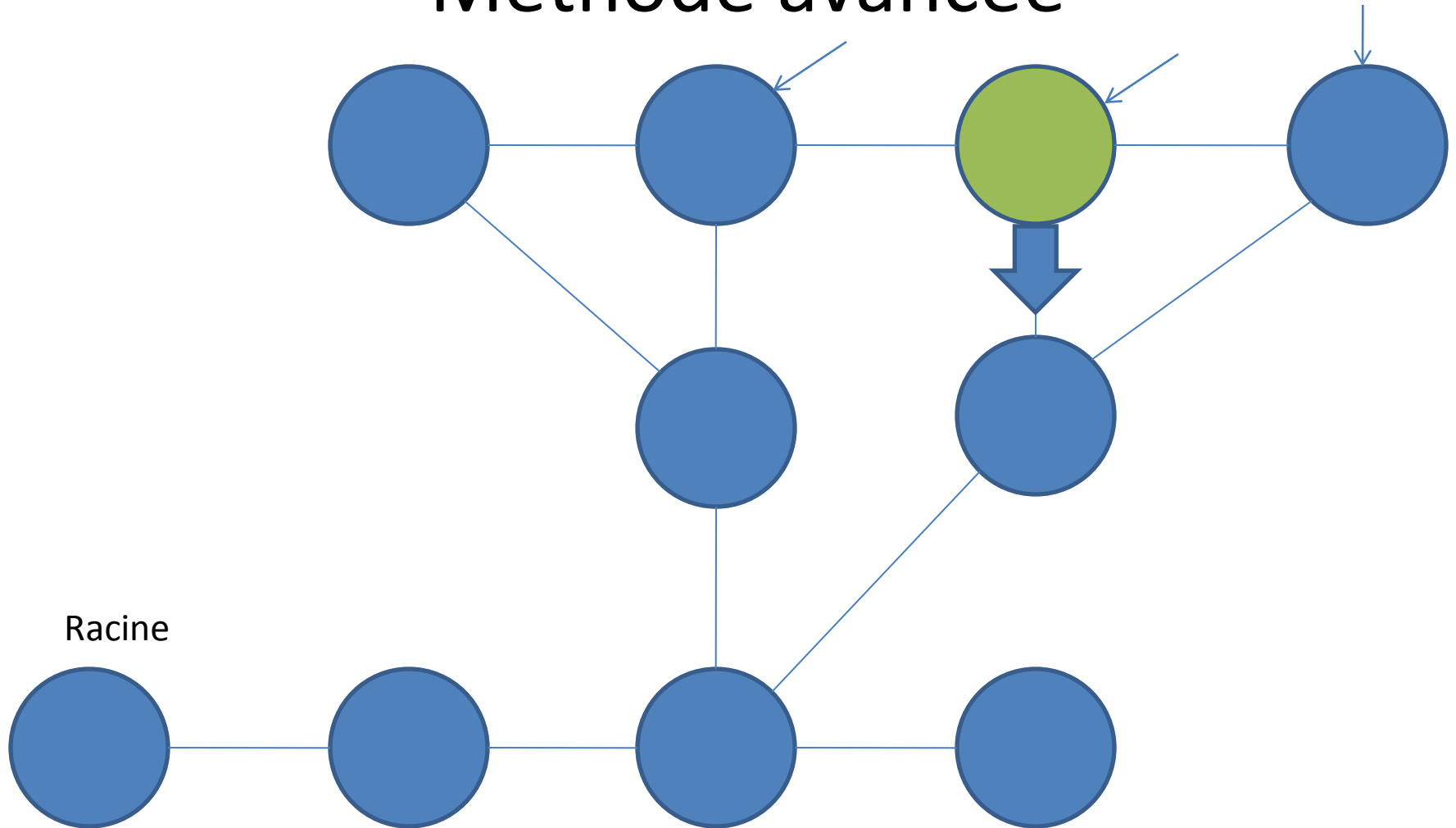
La gestion d'erreurs : Méthode naïve



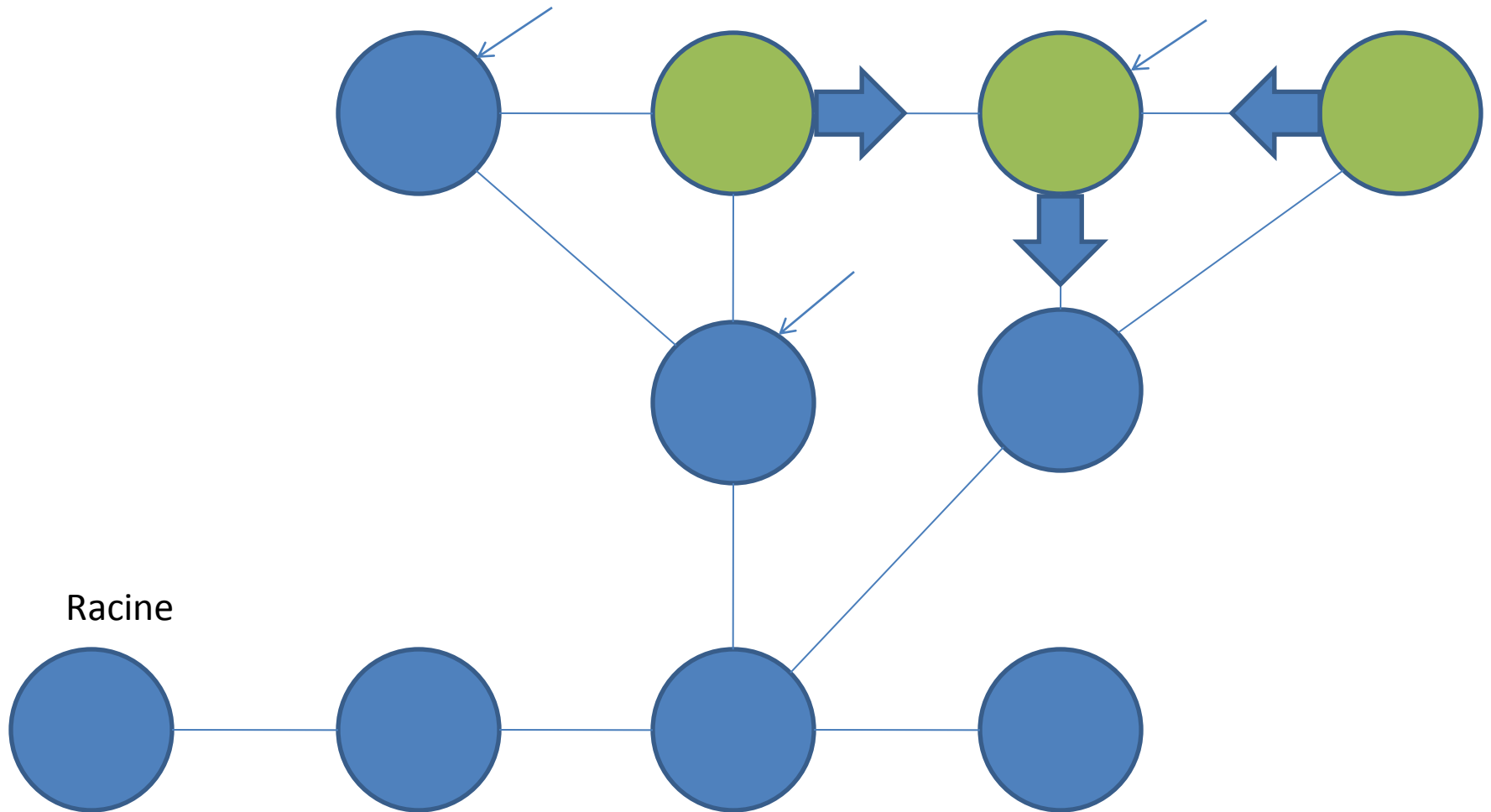
La gestion d'erreurs : Méthode Avancée

- But : Neutraliser les conséquences de l'erreur avant toute correction.

La gestion d'erreurs : Méthode avancée

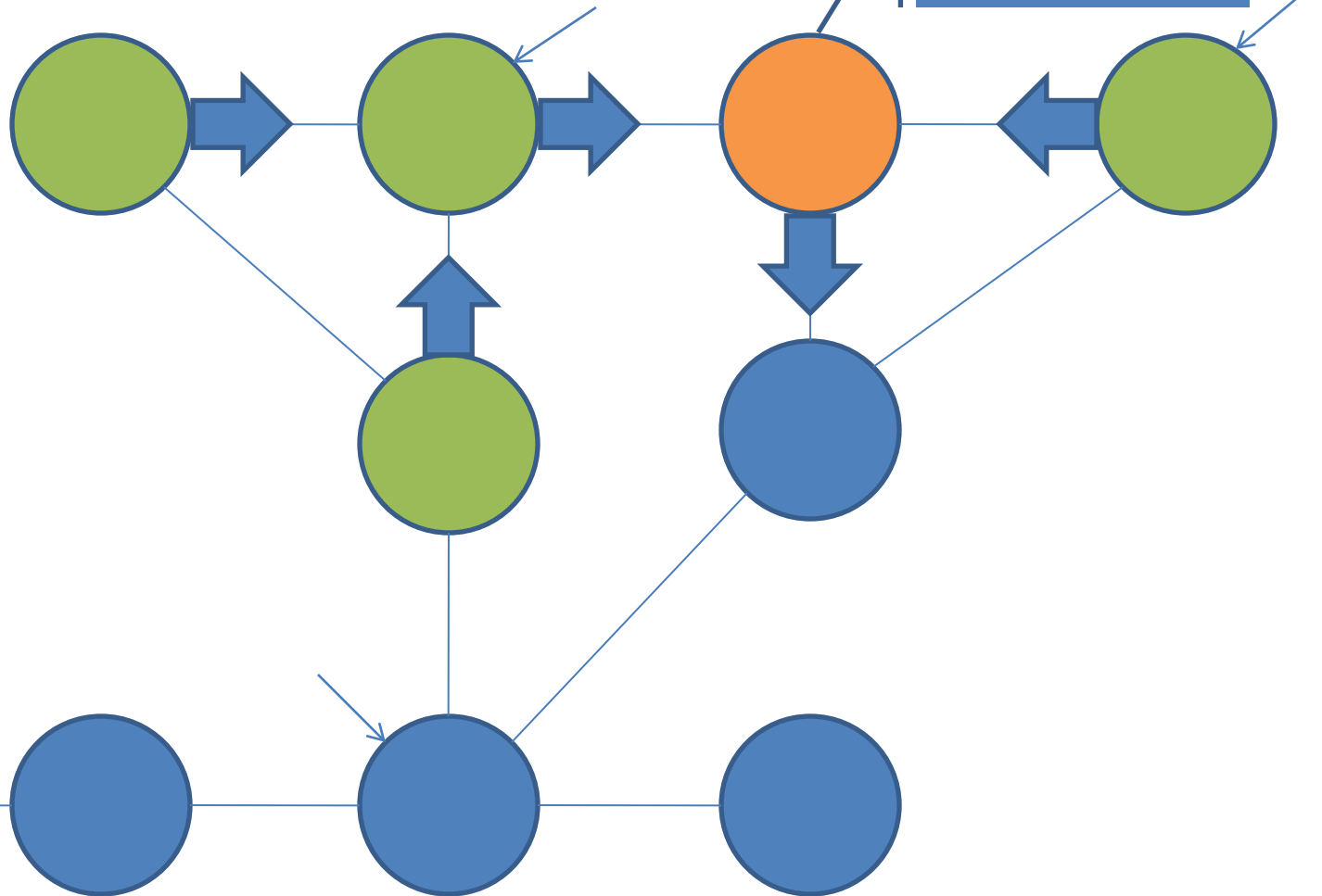


La gestion d'erreurs : Méthode avancée

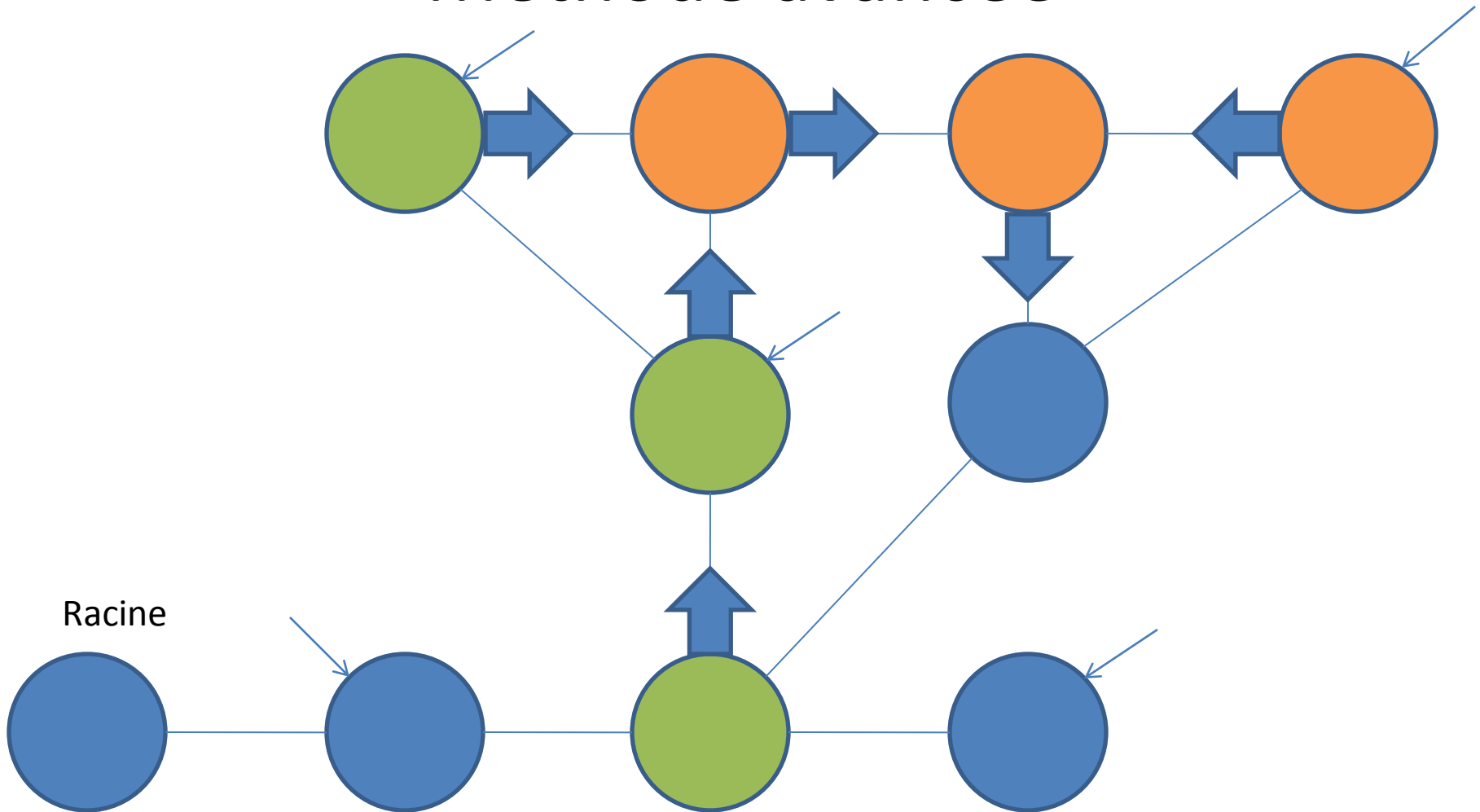


La gestion d'erreurs : Méthode avancée

Attention Il y a
une erreur

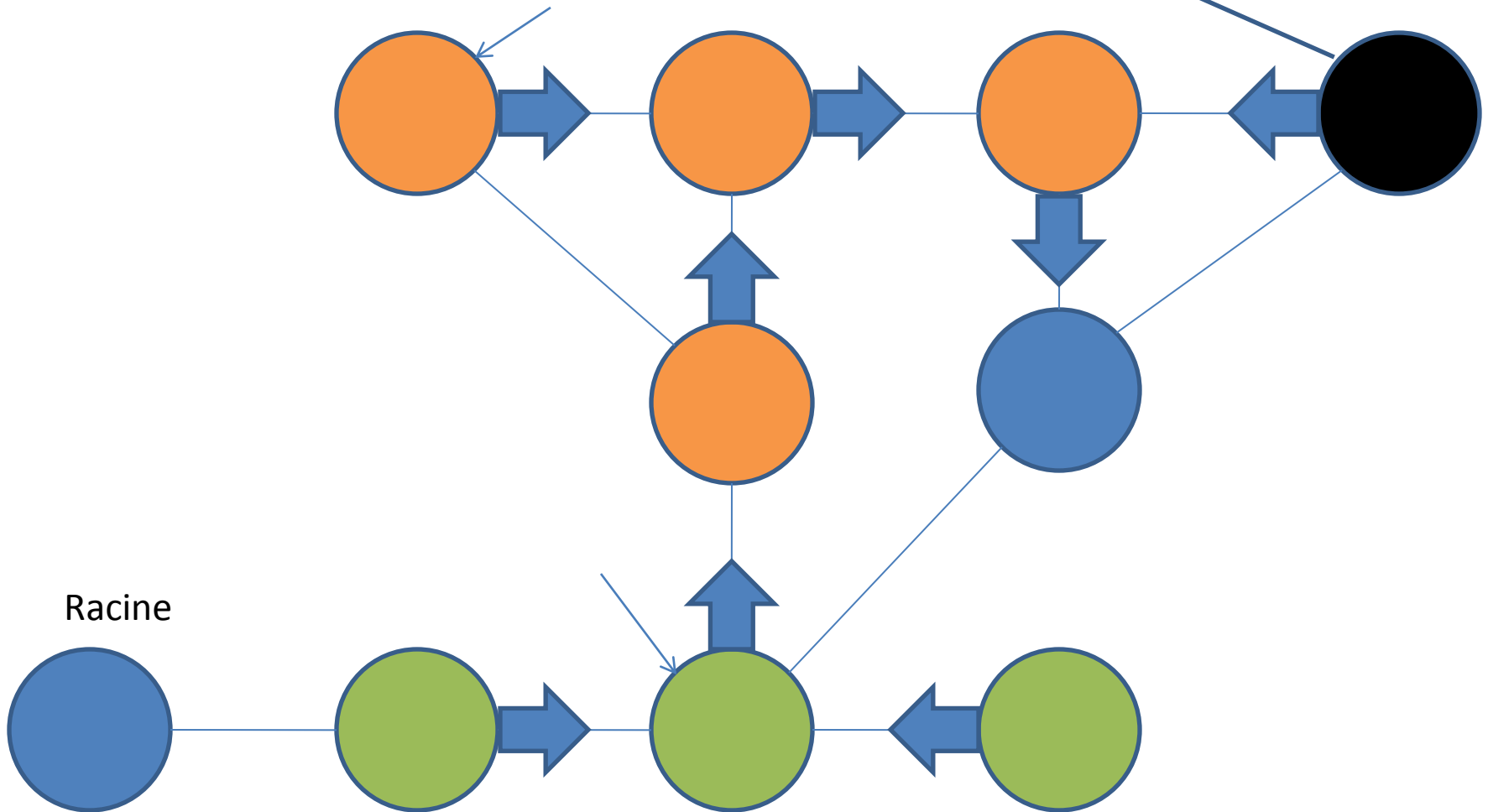


La gestion d'erreurs : Méthode avancée

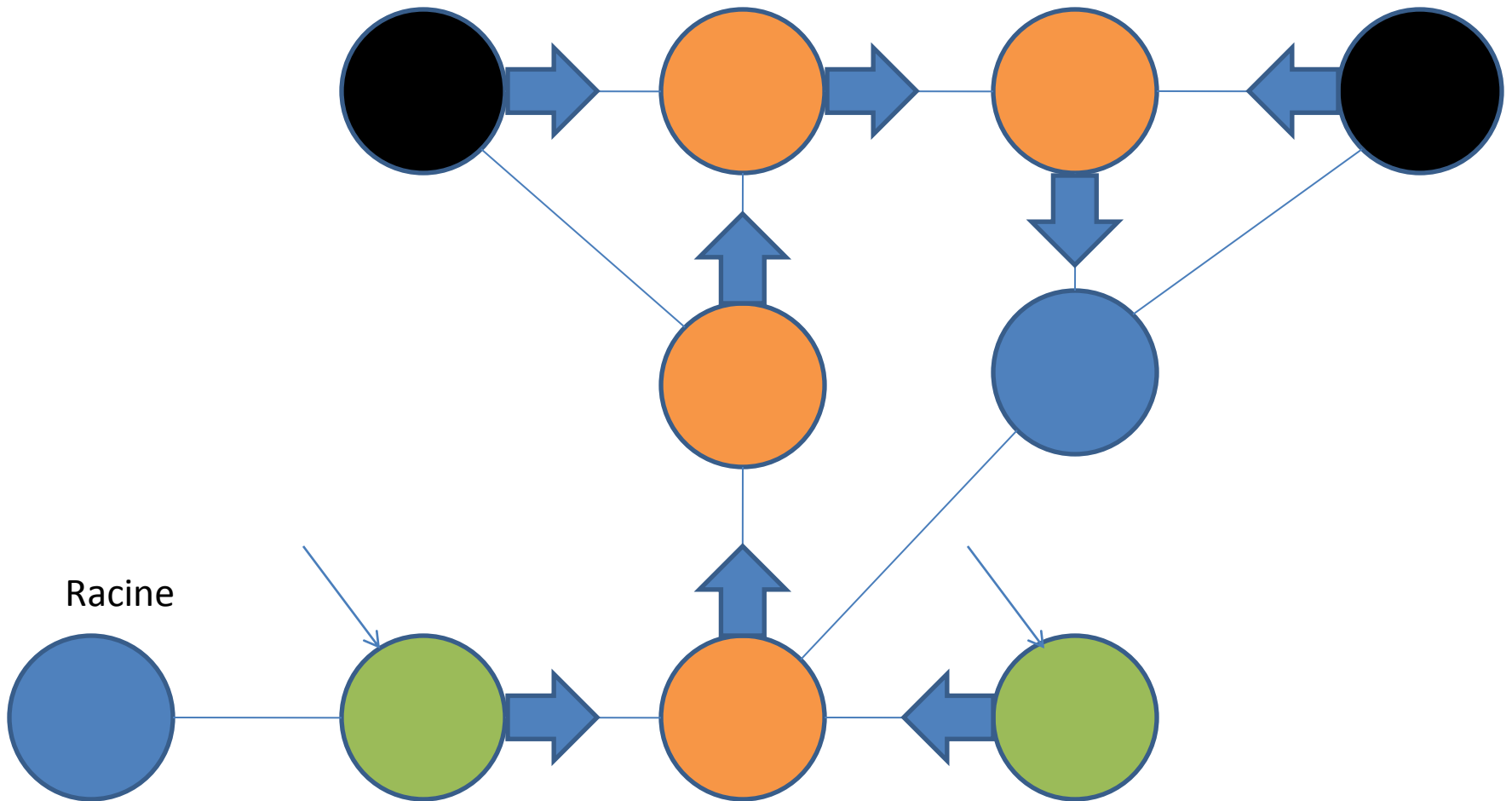


La gestion d'erreurs : Méthode avancée

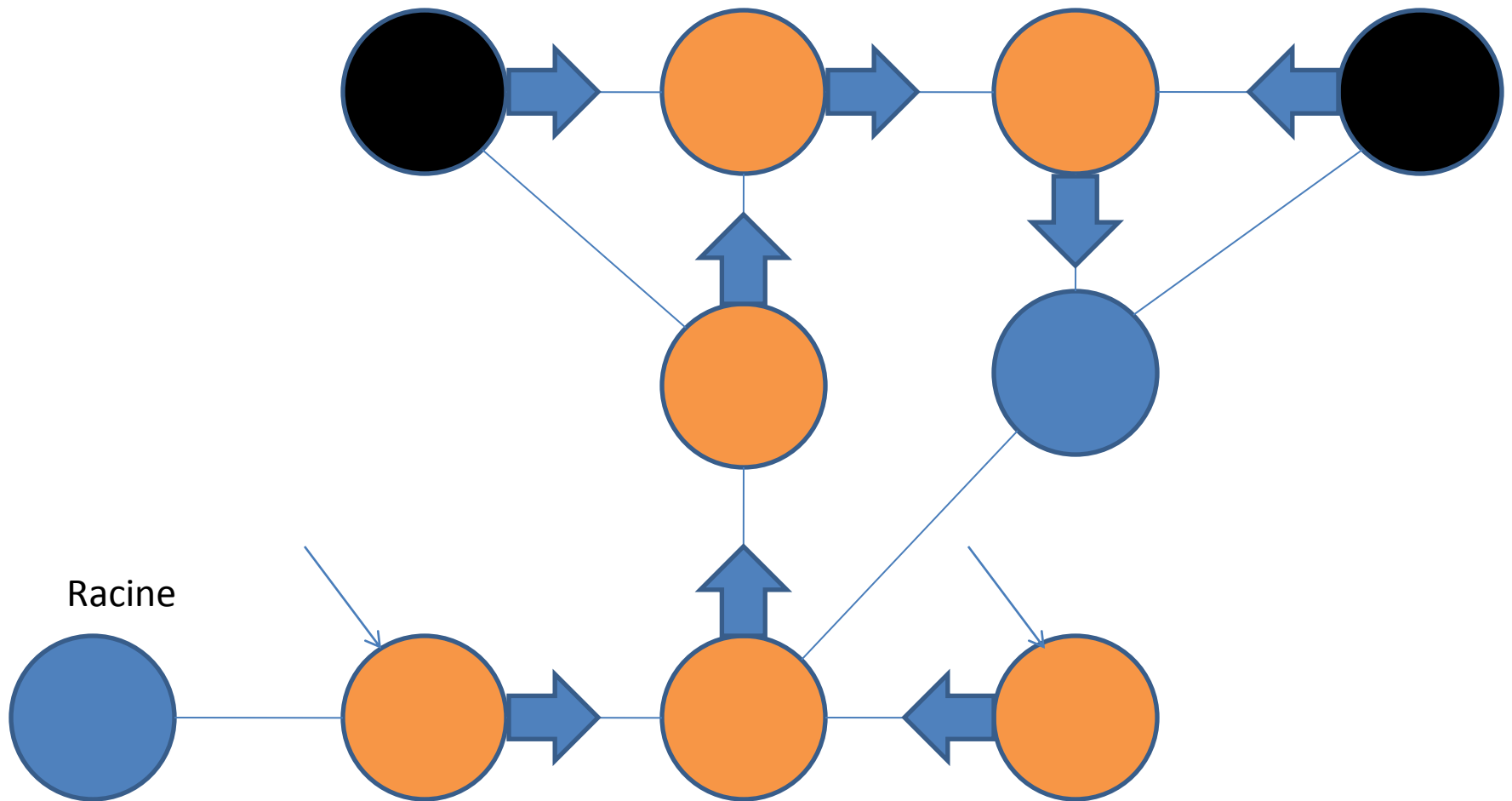
J'ai neutralisé les
conséquences



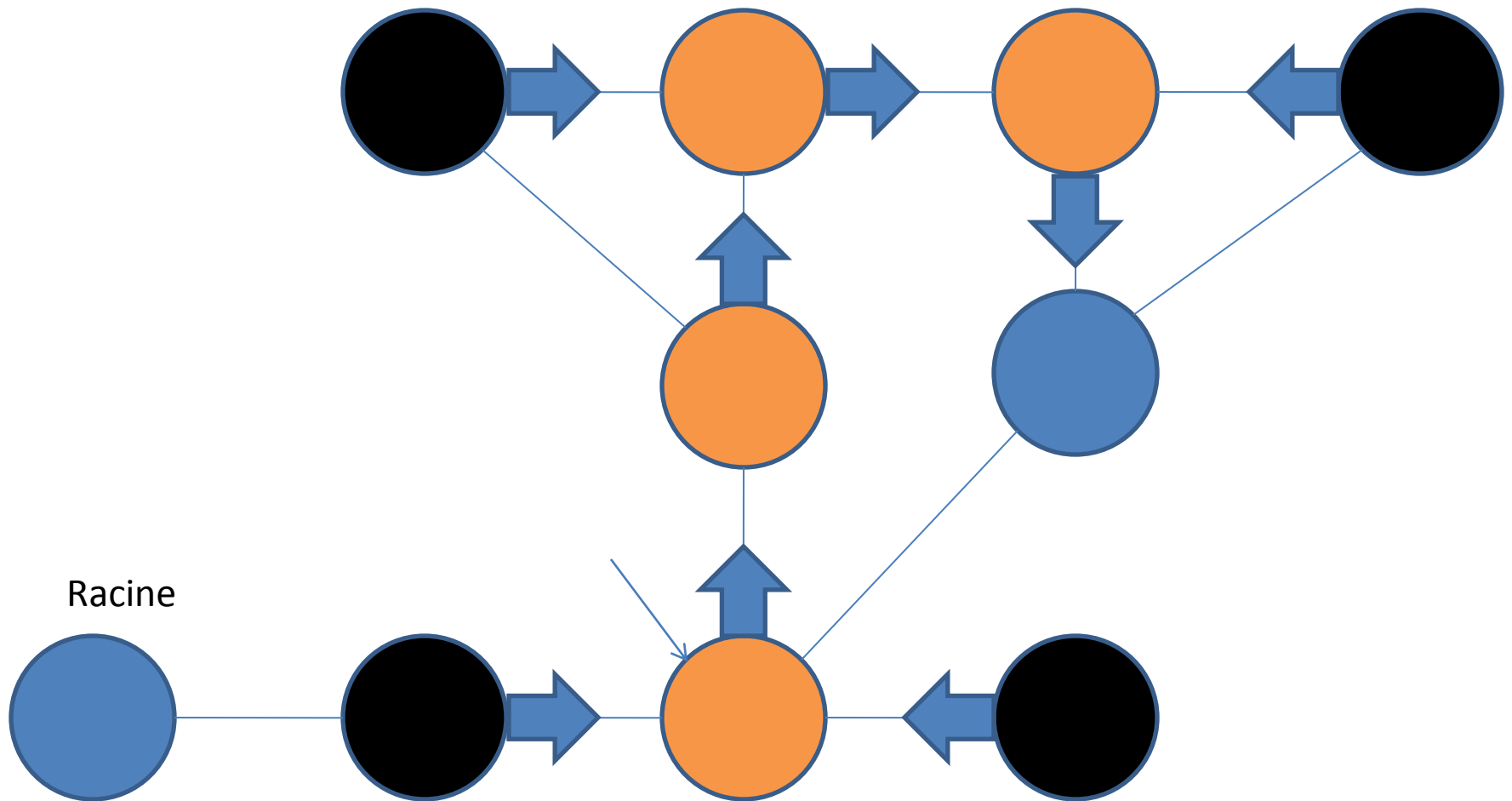
La gestion d'erreurs : Méthode avancée



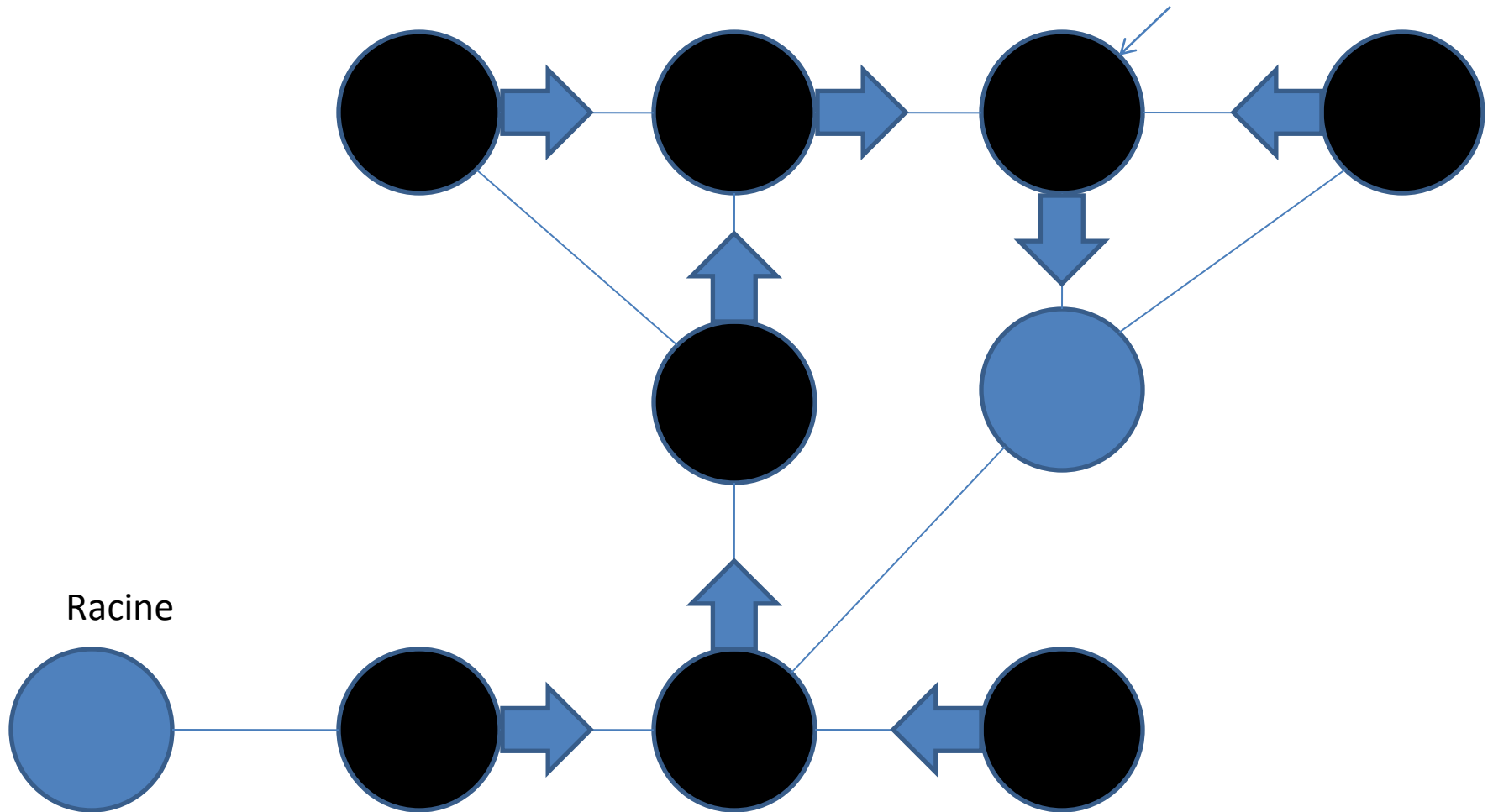
La gestion d'erreurs : Méthode avancée



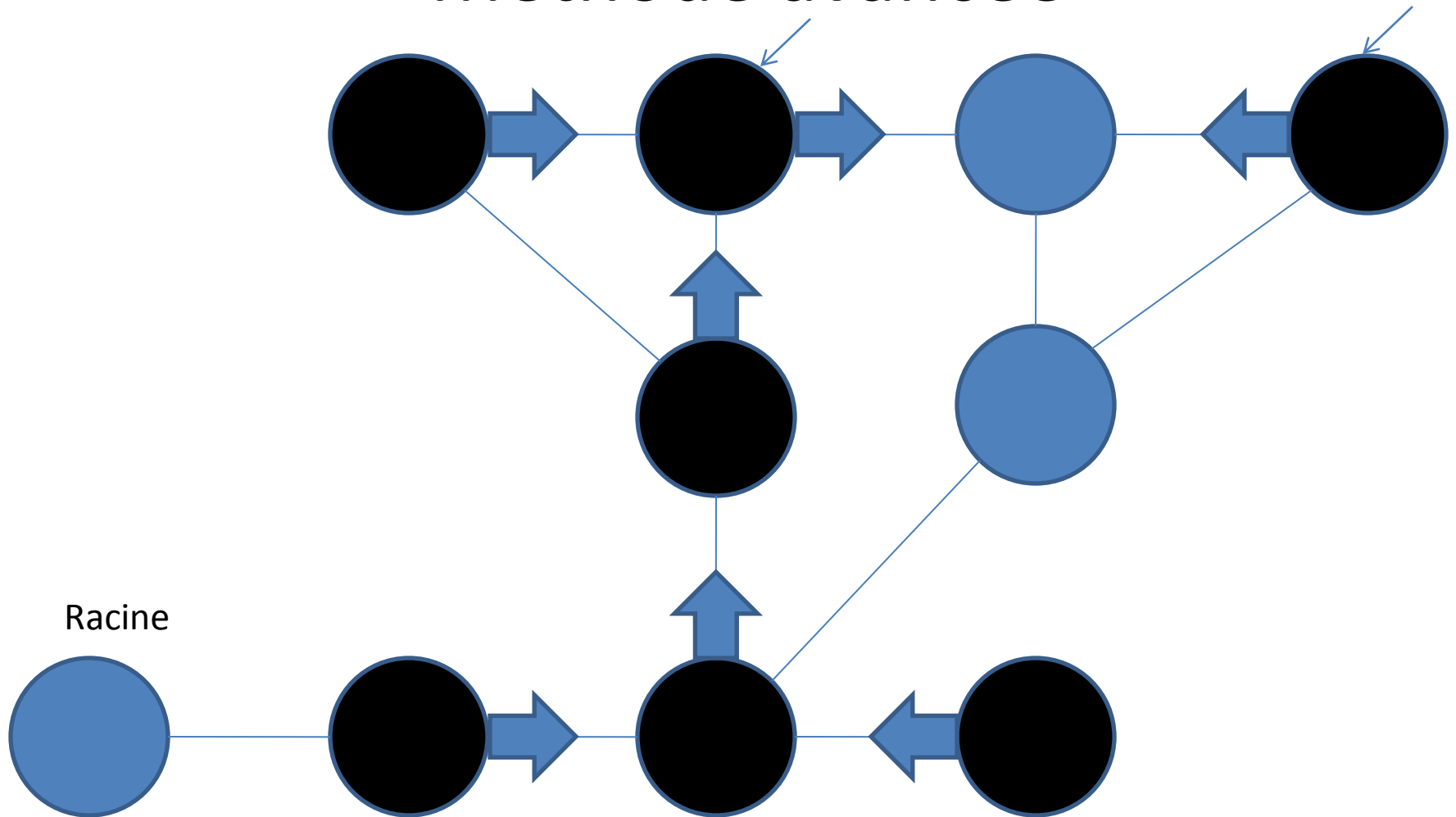
La gestion d'erreurs : Méthode avancée



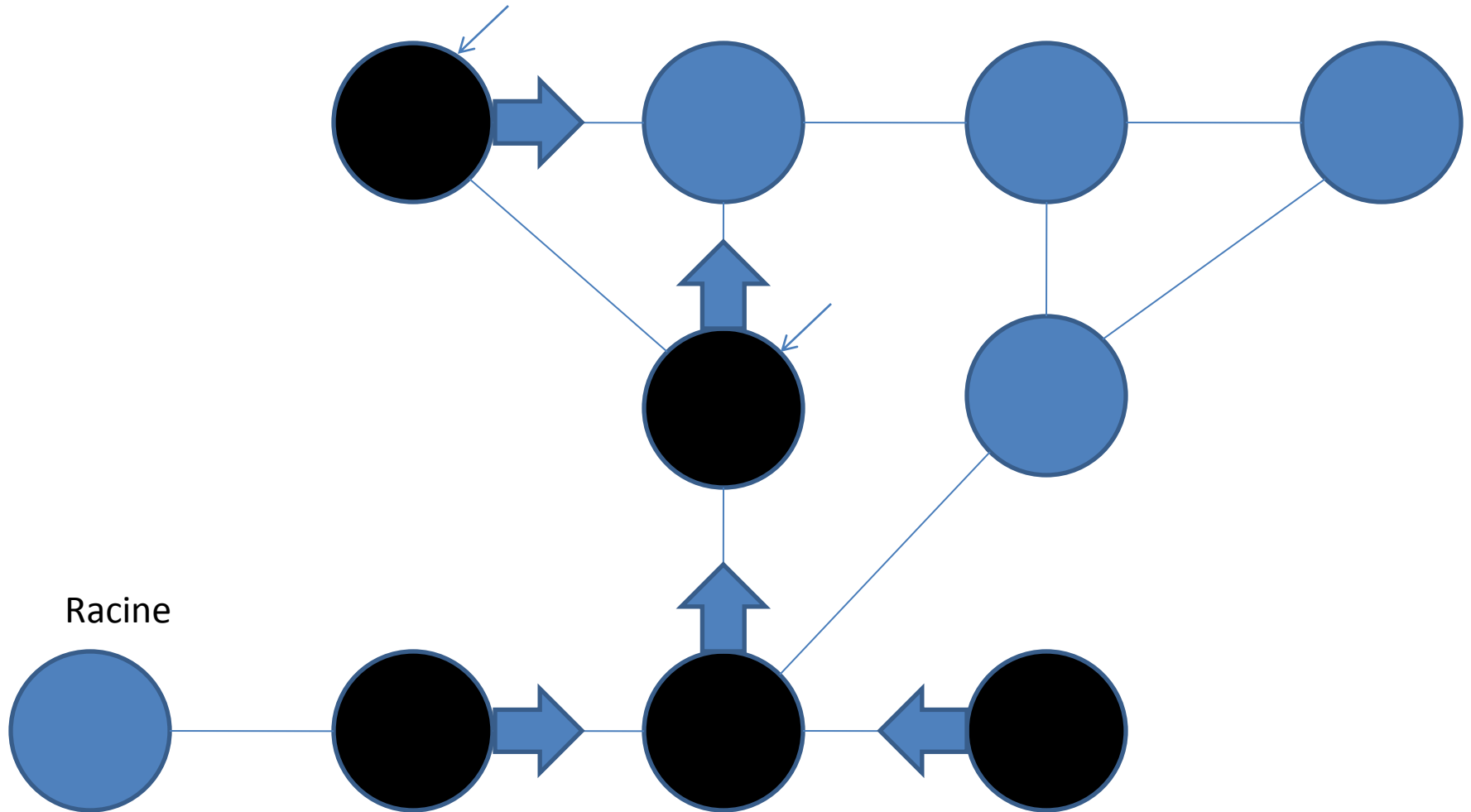
La gestion d'erreurs : Méthode avancée



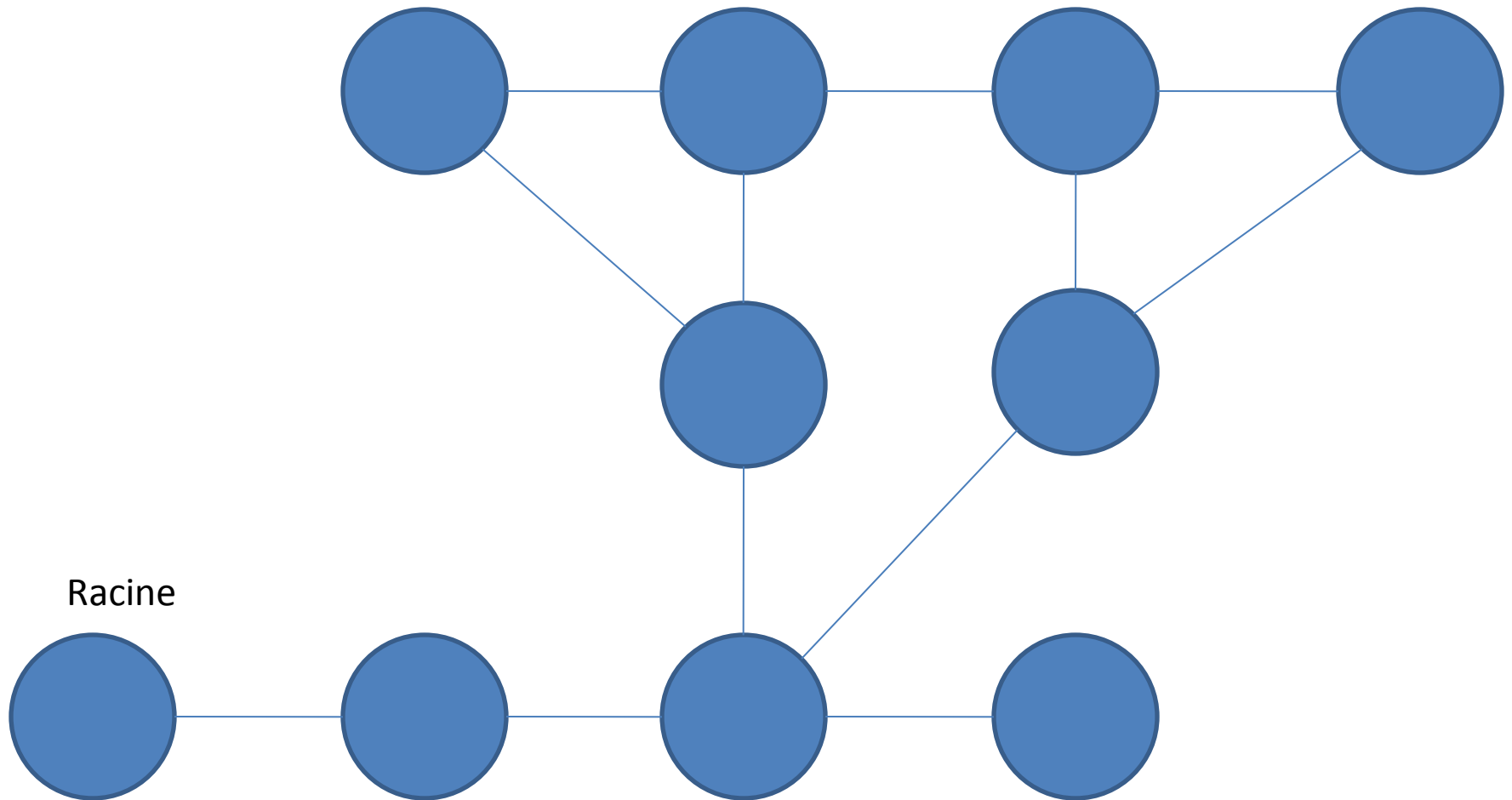
La gestion d'erreurs : Méthode avancée



La gestion d'erreurs : Méthode avancée



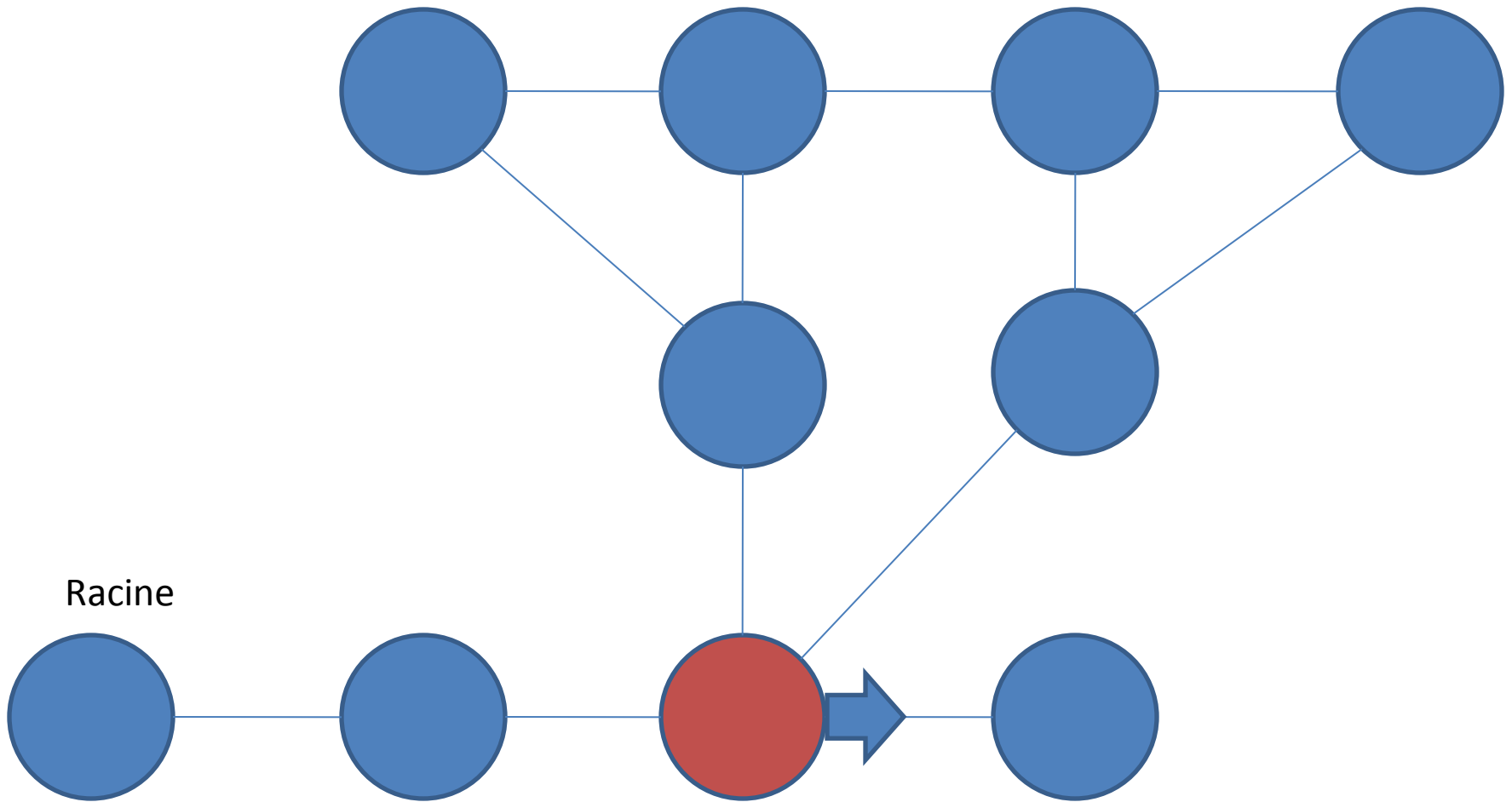
La gestion d'erreurs : Méthode avancée



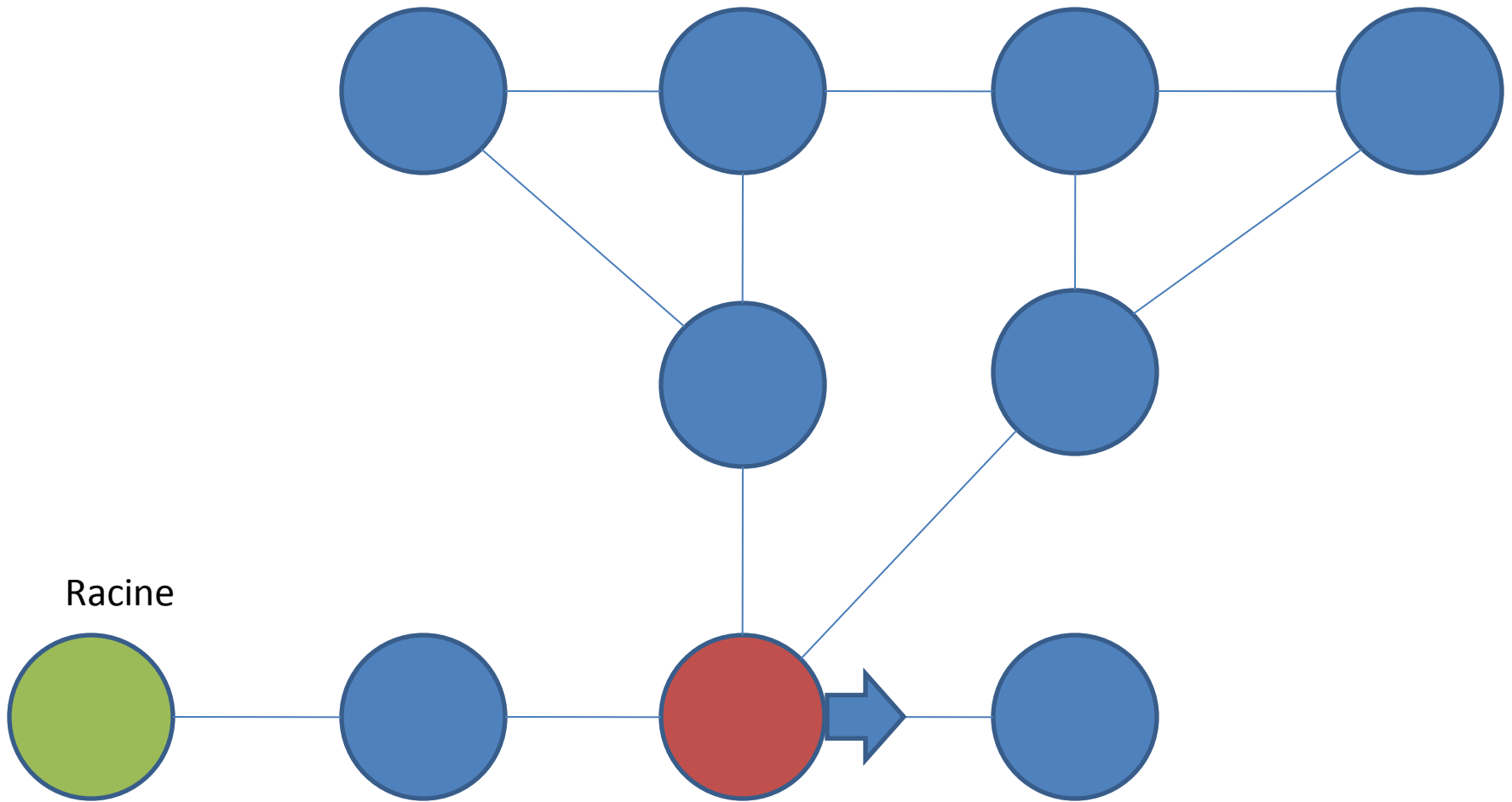
Comparaison de ces deux méthodes

- La méthode naïve entraîne :
 1. Une variable de niveau bornée par une constante Max avec $Max > n-1$.
 2. Un nombre de rondes proportionnel à cette constante Max.
- La méthode avancée entraîne :
 1. Une variable de niveau non obligatoirement bornée.
 2. Un nombre de rondes proportionnel à n .
 3. Limitation du nombre de faux services sur un nœud.

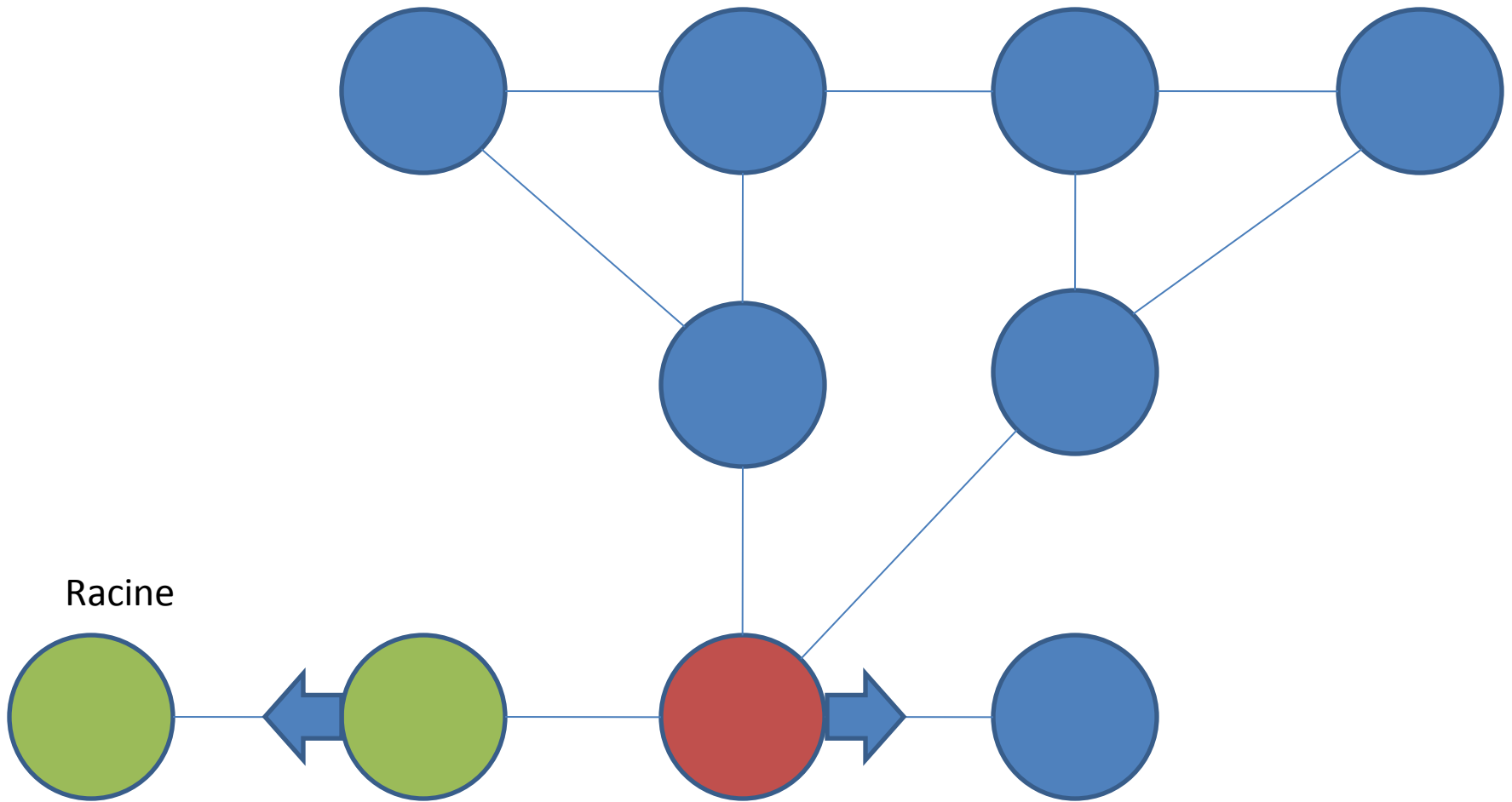
Détecter la fin de la construction de l'arbre couvrant : Méthode naïve



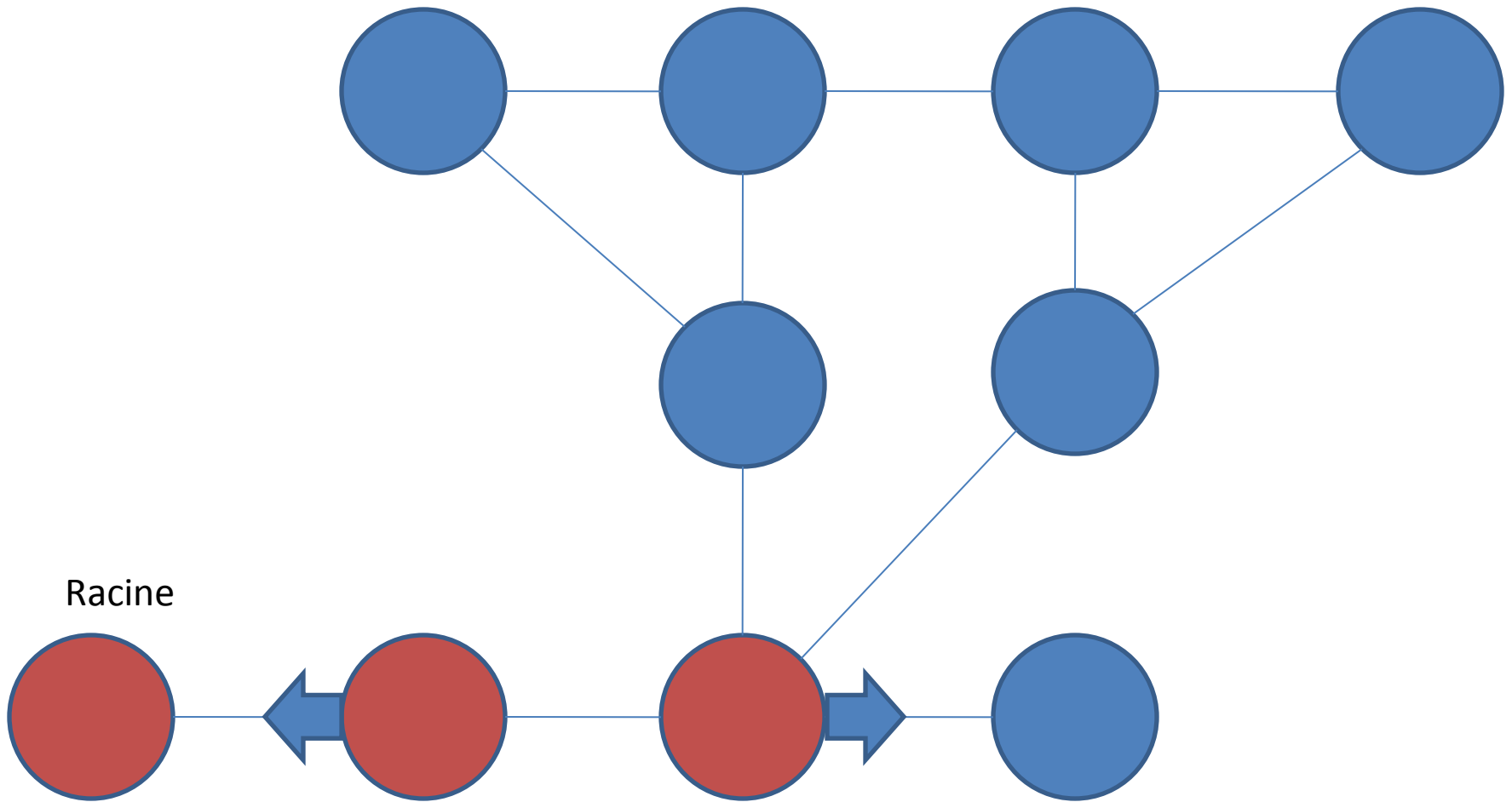
Détecter la fin de la construction de l'arbre couvrant : Méthode naïve



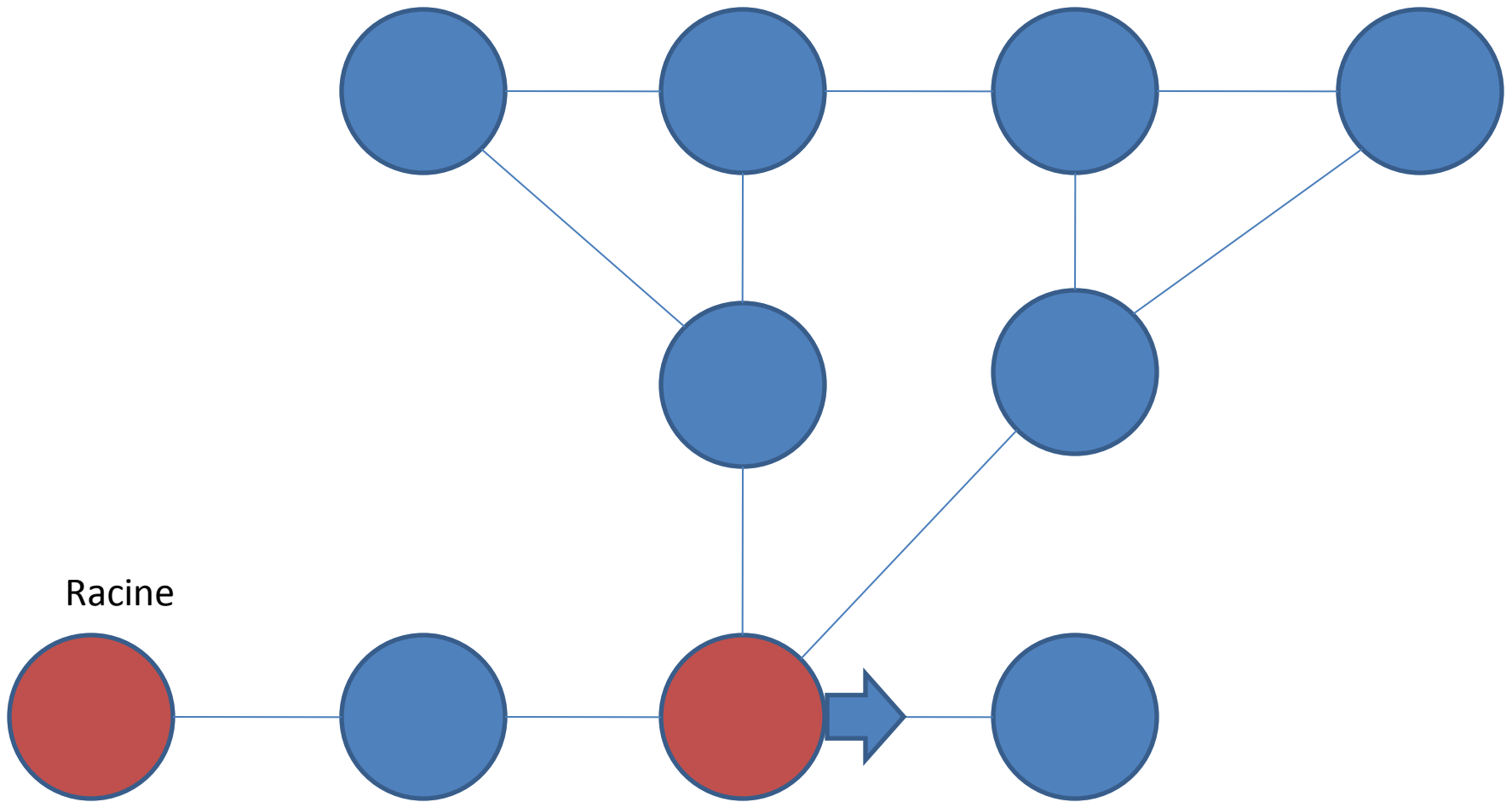
Détecter la fin de la construction de l'arbre couvrant : Méthode naïve



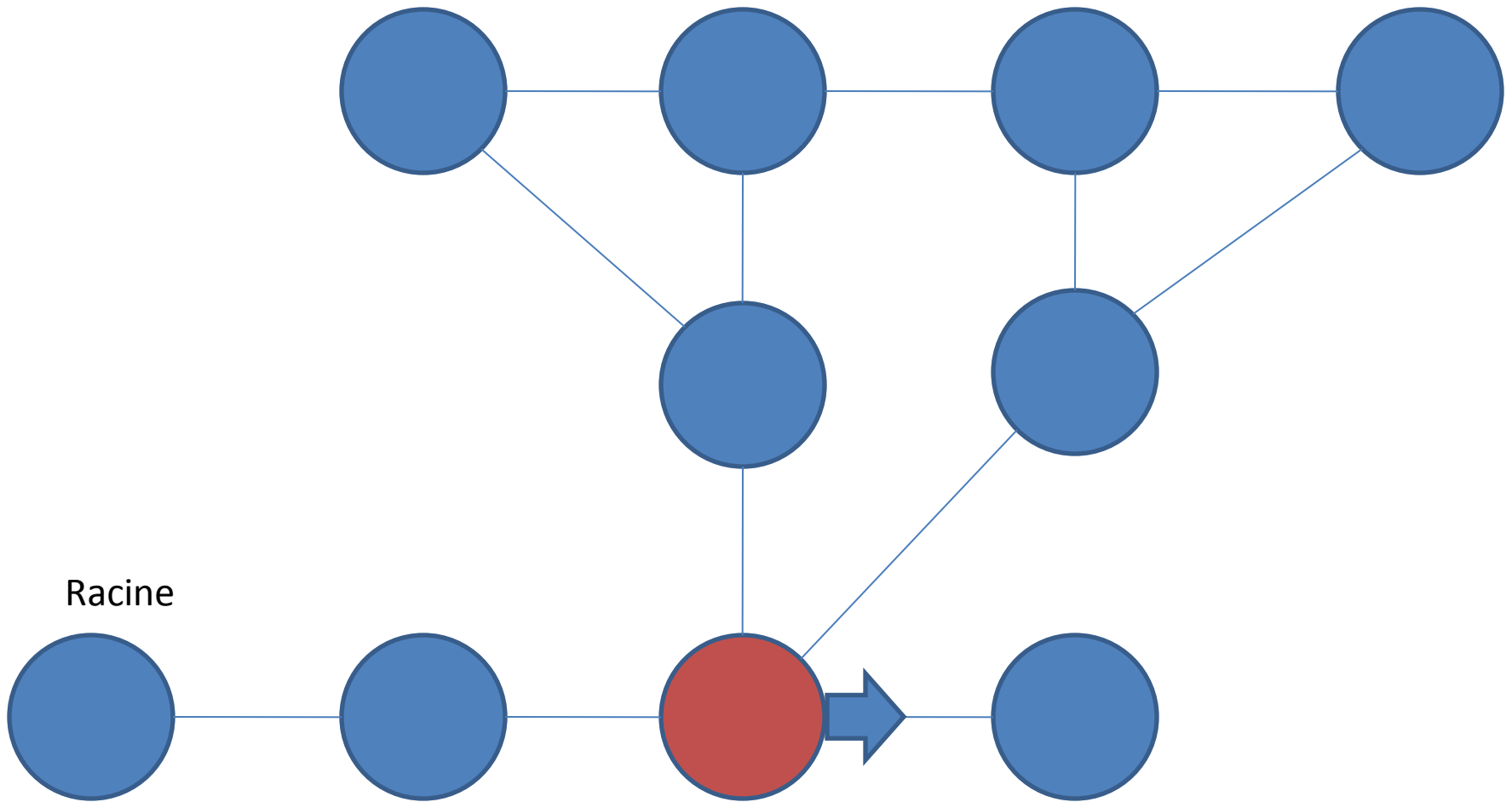
Détecter la fin de la construction de l'arbre couvrant : Méthode naïve



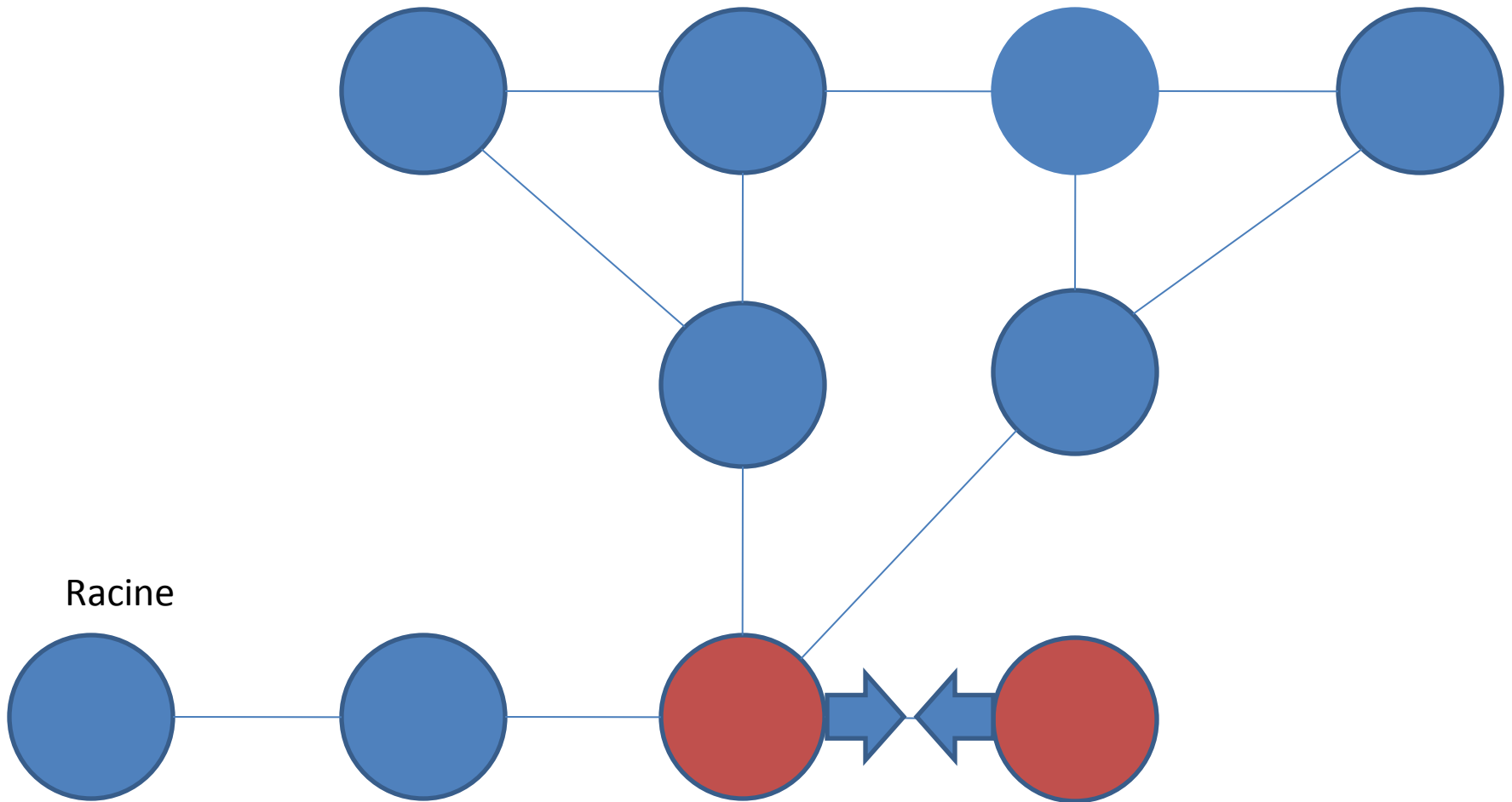
Détecter la fin de la construction de l'arbre couvrant : Méthode naïve



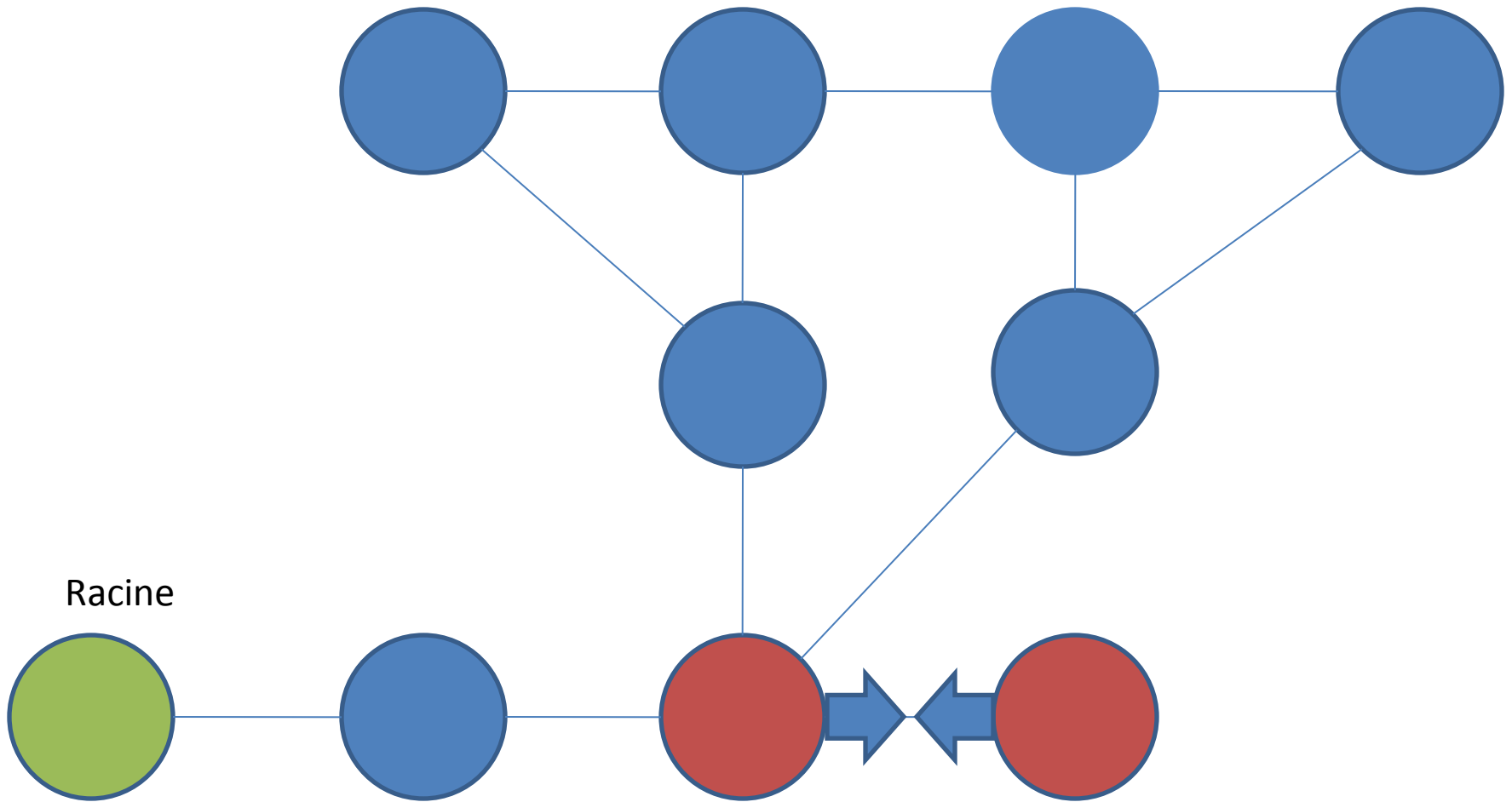
Détecter la fin de la construction de l'arbre couvrant : Méthode naïve



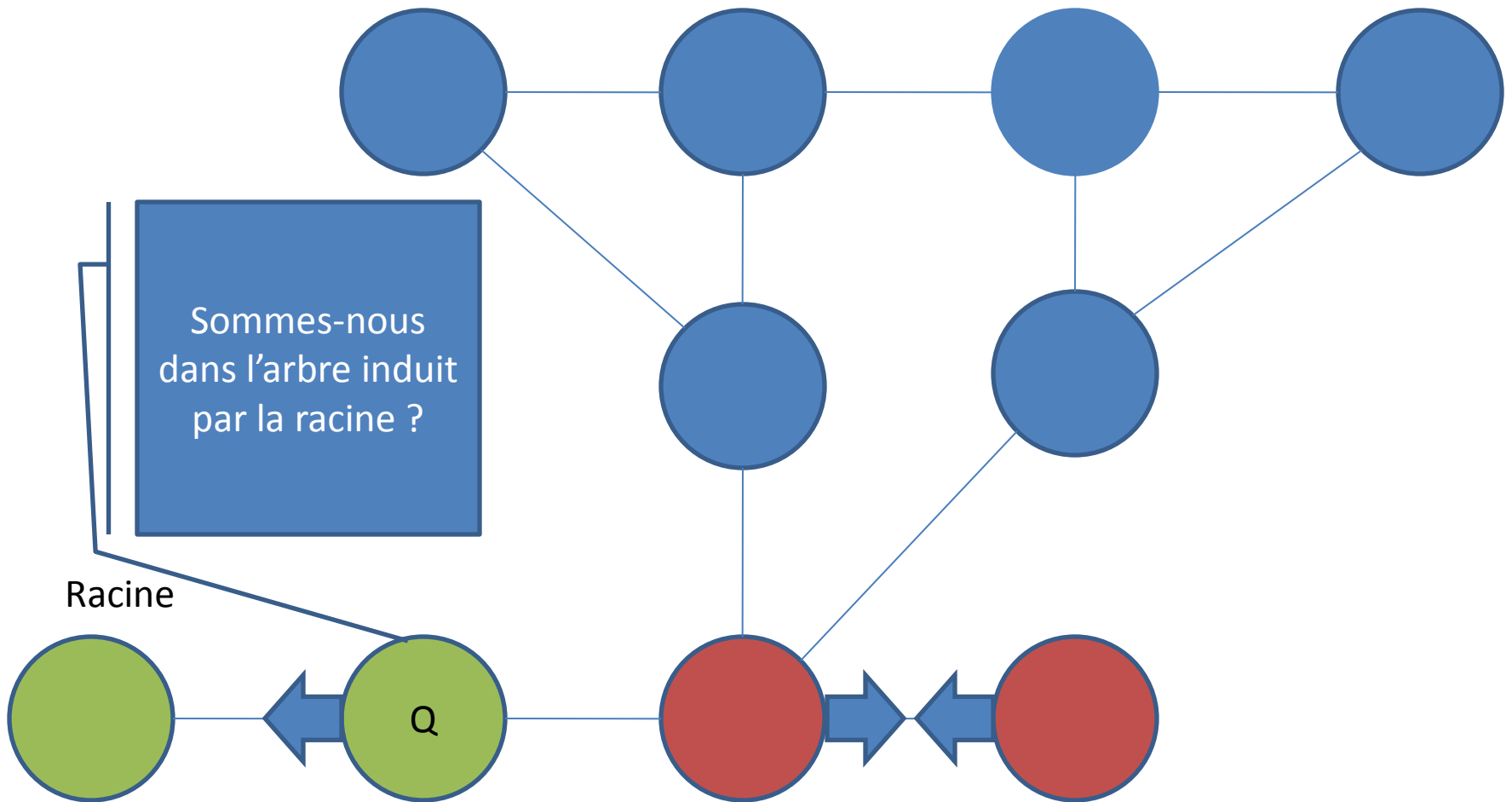
Détecter la fin de la construction de l'arbre couvrant : Méthode avancée



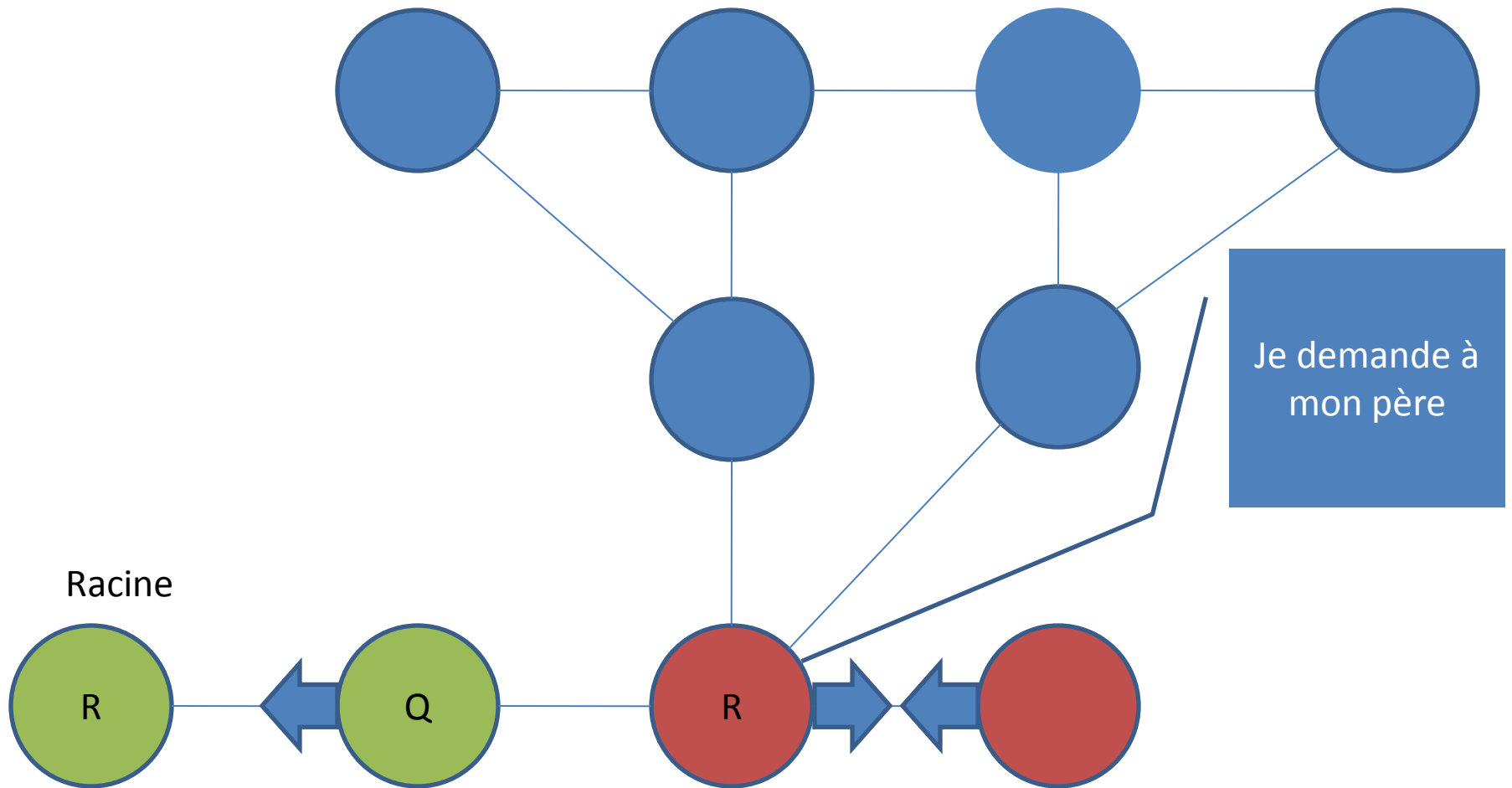
Détecter la fin de la construction de l'arbre couvrant : Méthode avancée



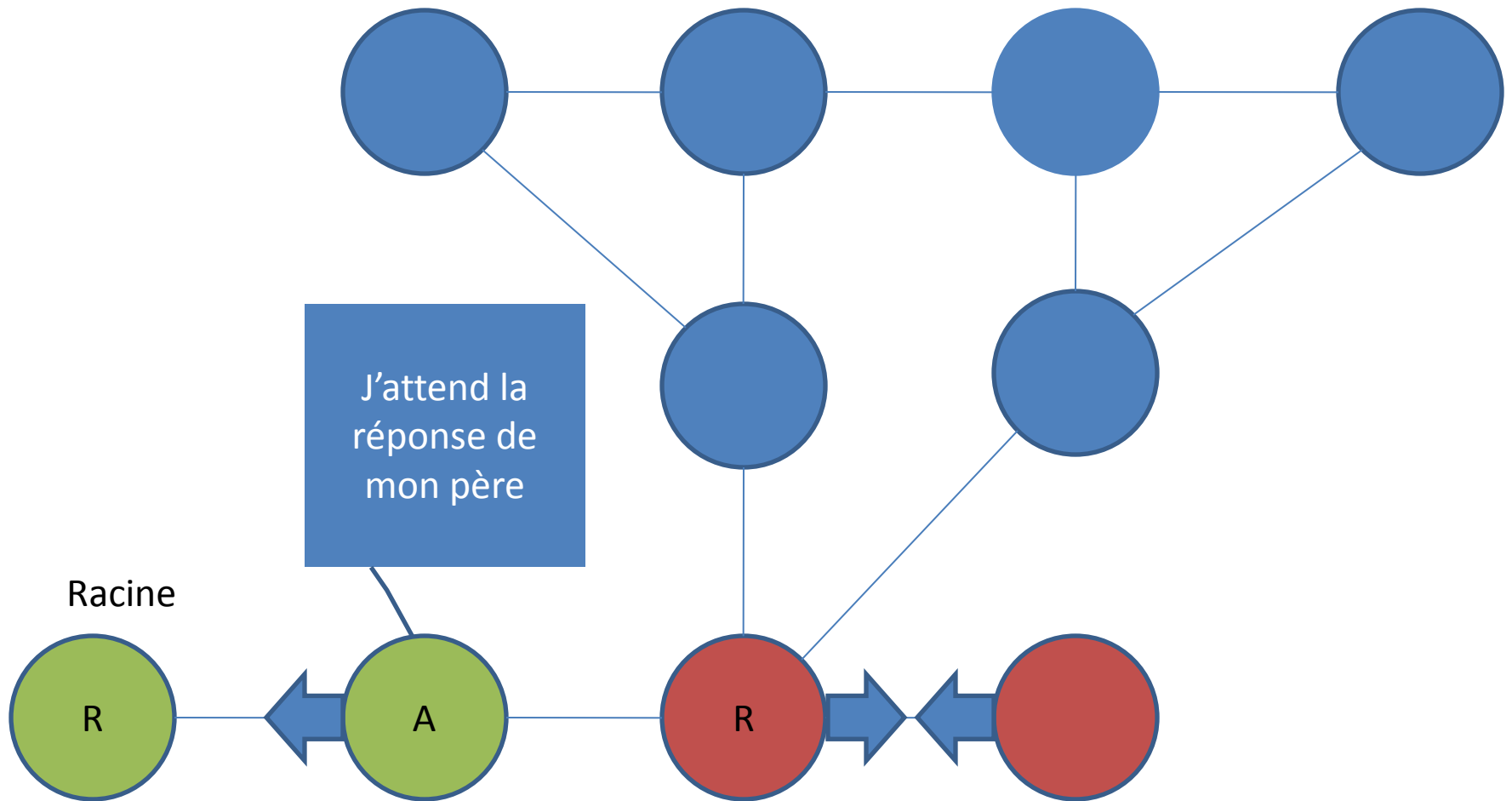
Détecter la fin de la construction de l'arbre couvrant : Méthode avancée



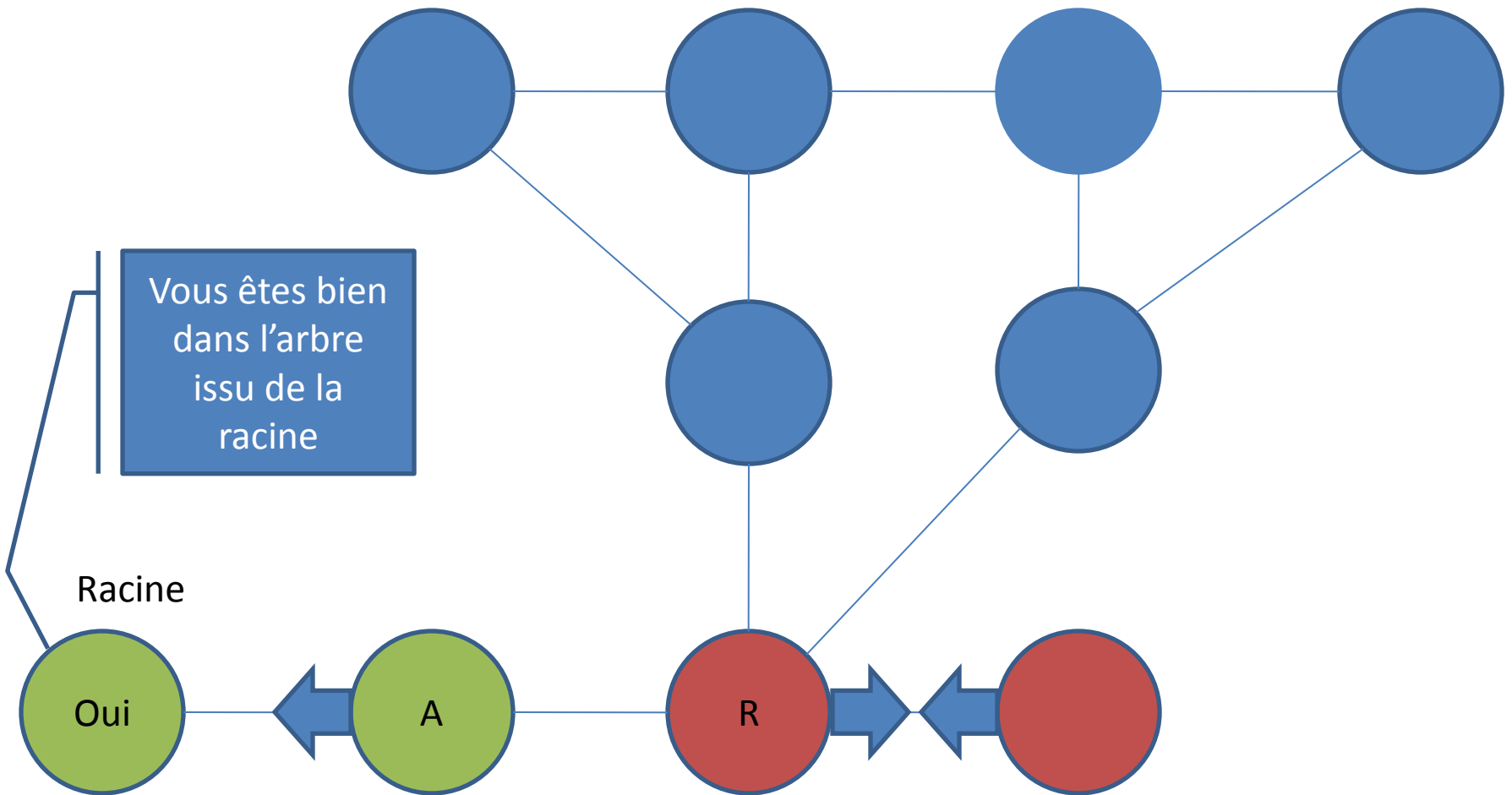
Détecter la fin de la construction de l'arbre couvrant : Méthode avancée



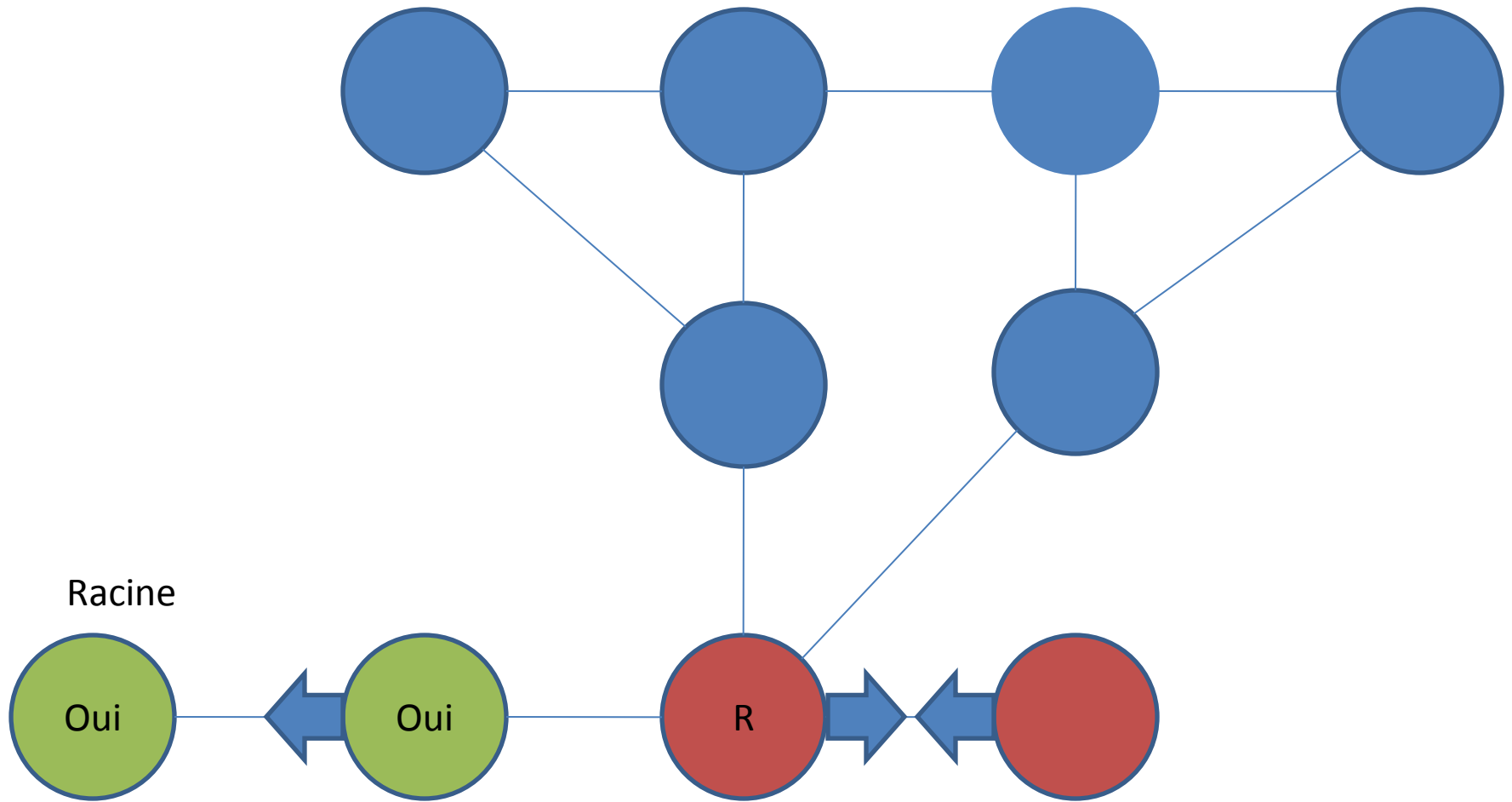
Détecter la fin de la construction de l'arbre couvrant : Méthode avancée



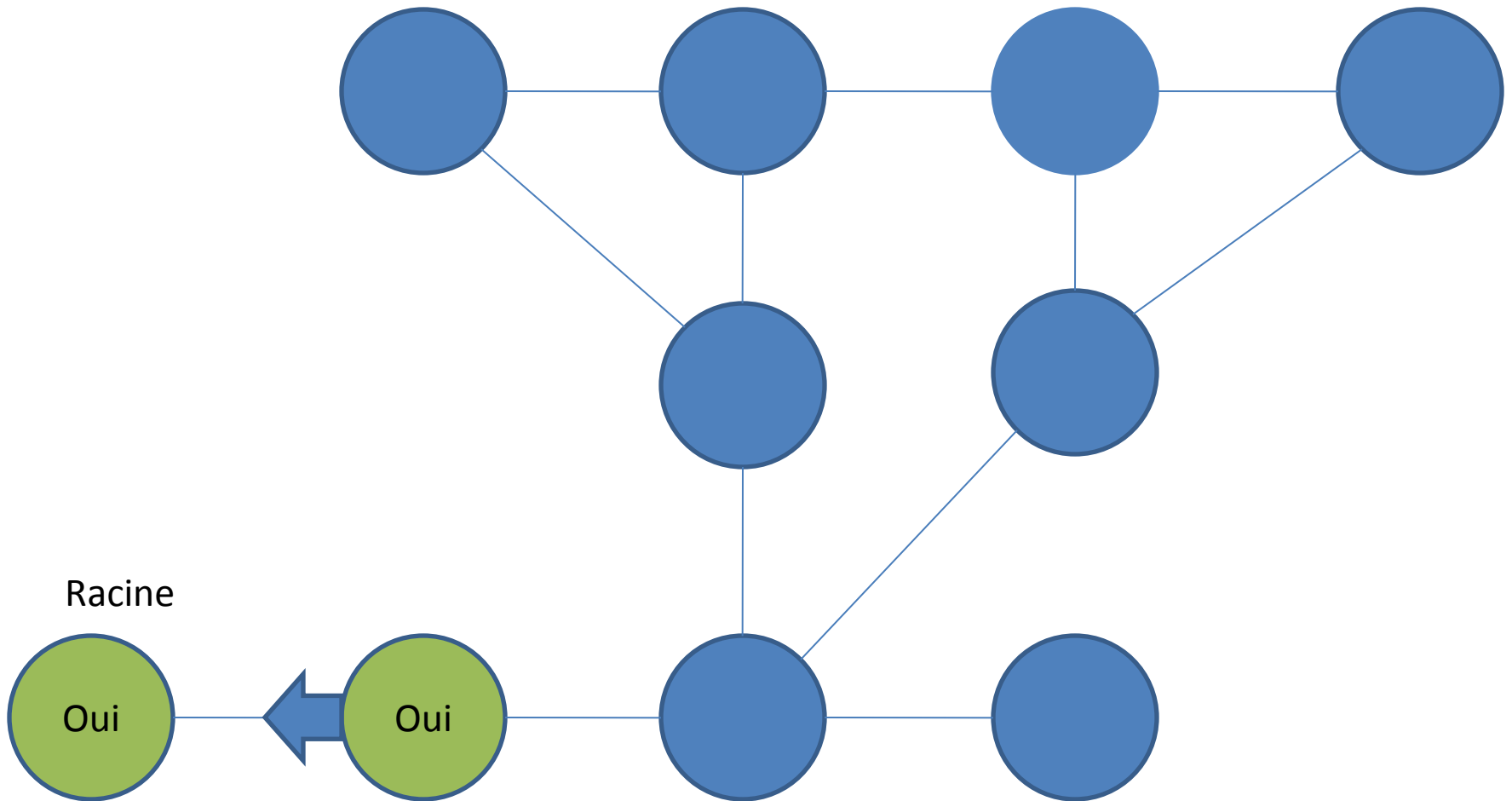
Détecter la fin de la construction de l'arbre couvrant : Méthode avancée



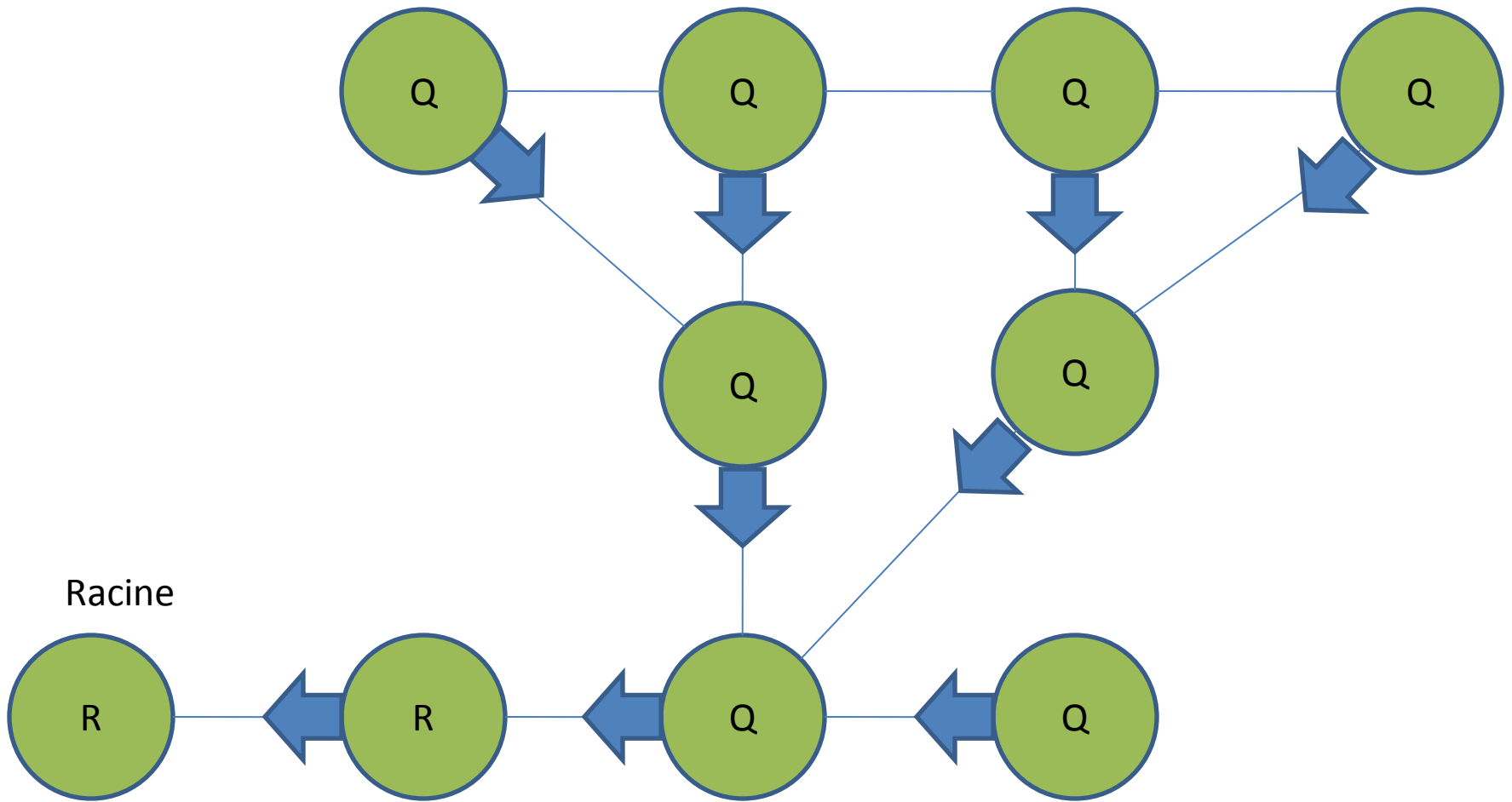
Détecter la fin de la construction de l'arbre couvrant : Méthode avancée



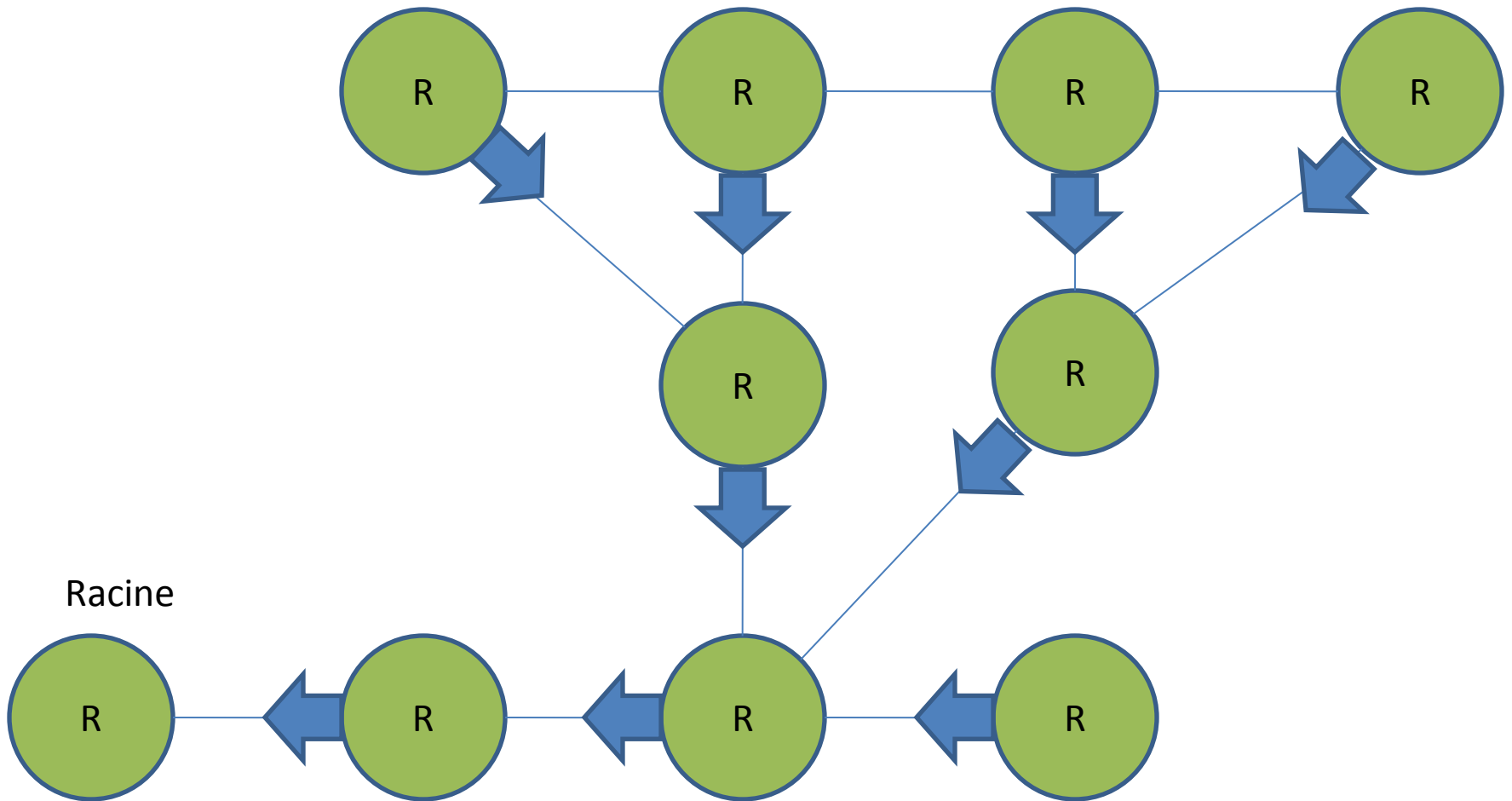
Détecter la fin de la construction de l'arbre couvrant : Méthode avancée



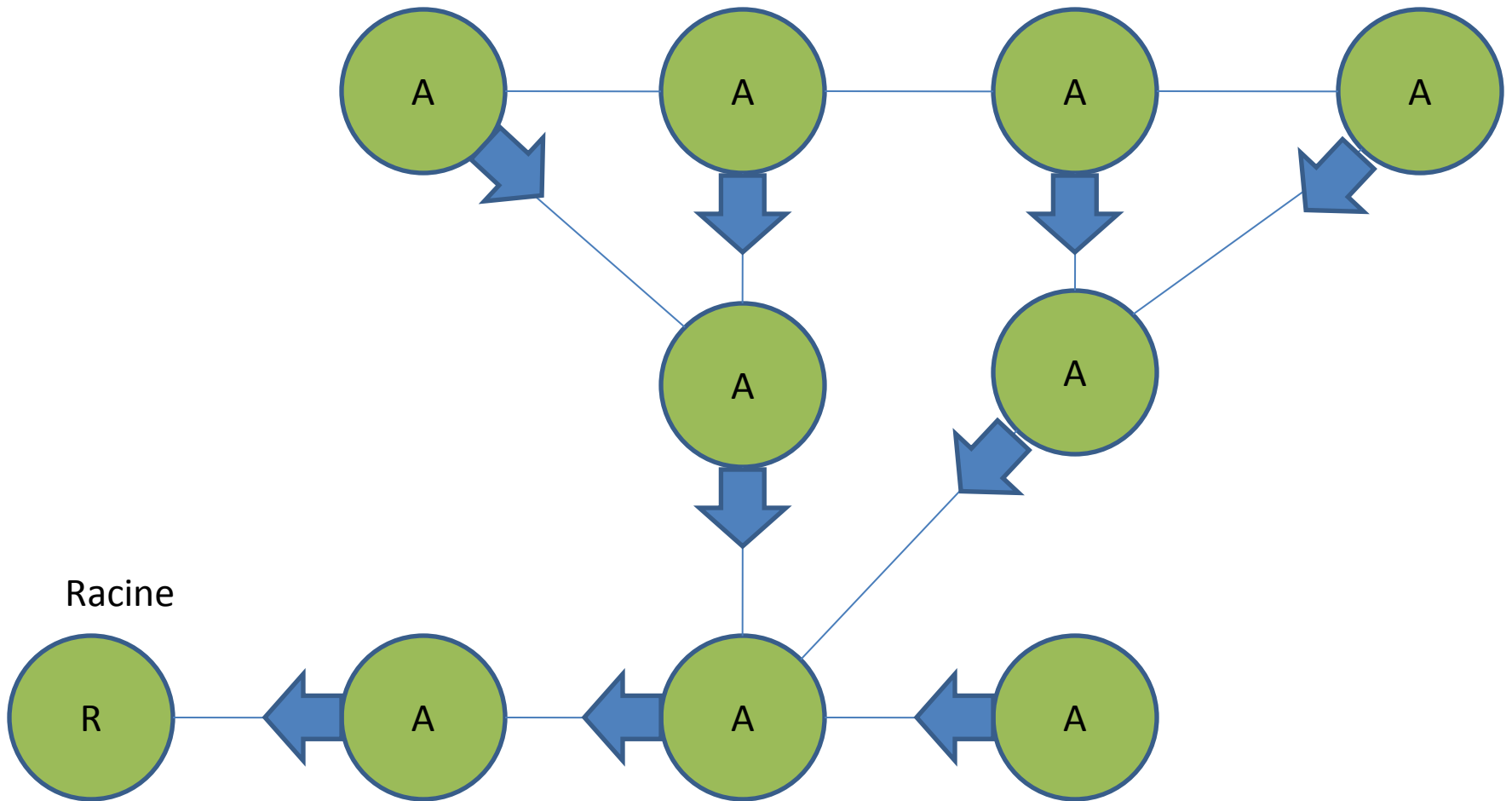
Détecter la fin de la construction de l'arbre couvrant : Méthode avancée



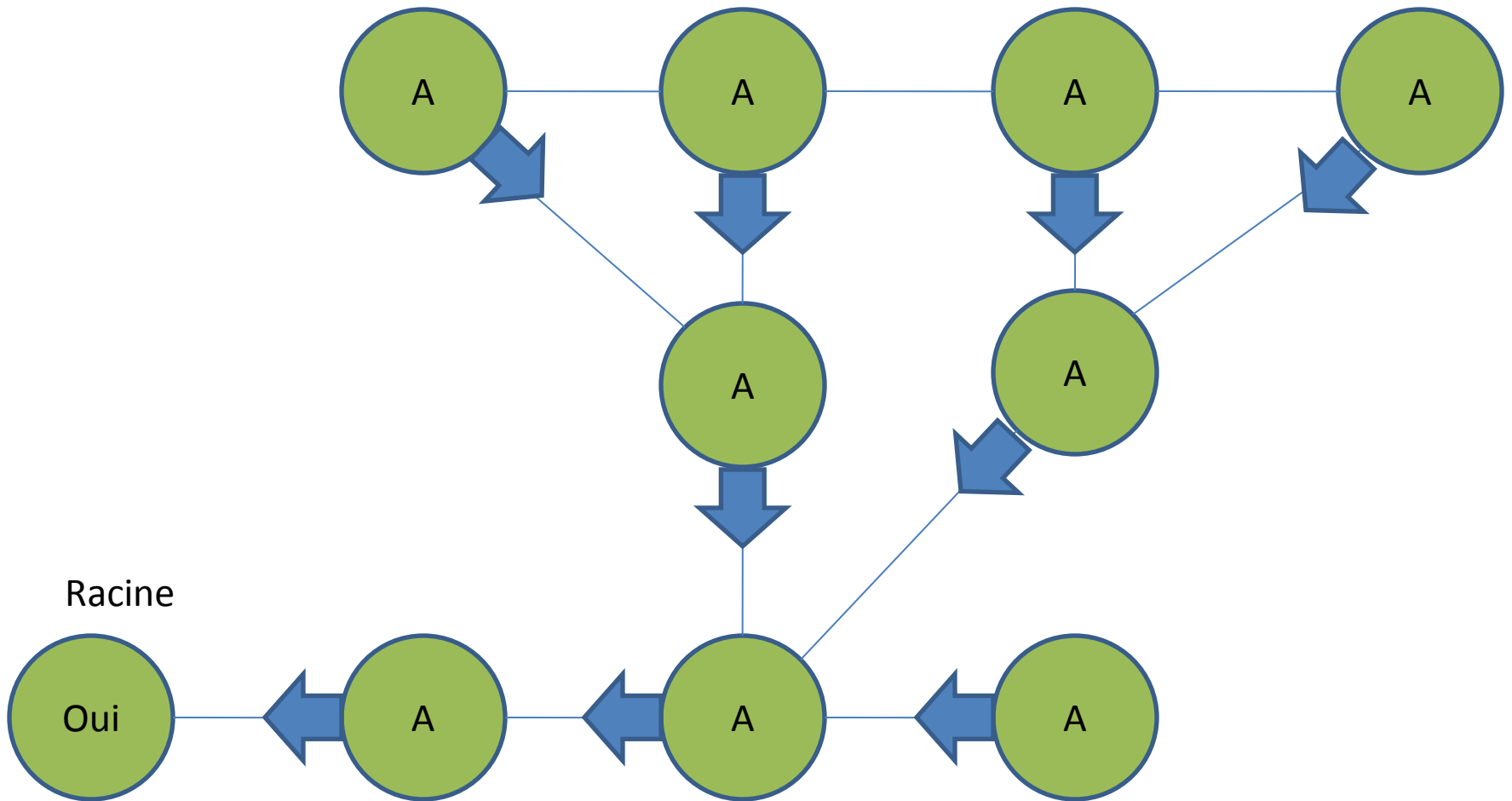
Détecter la fin de la construction de l'arbre couvrant : Méthode avancée



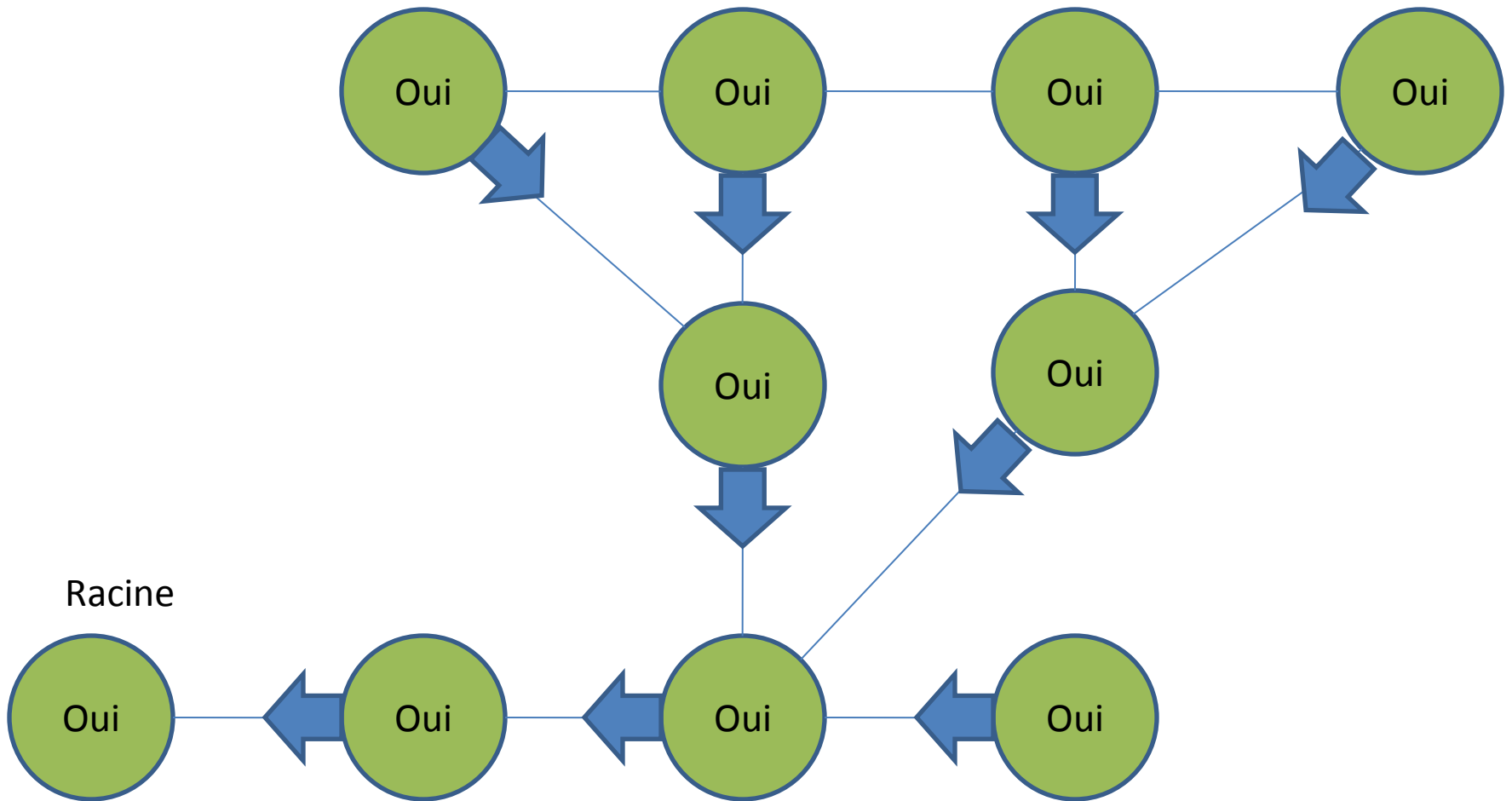
Détecter la fin de la construction de l'arbre couvrant : Méthode avancée



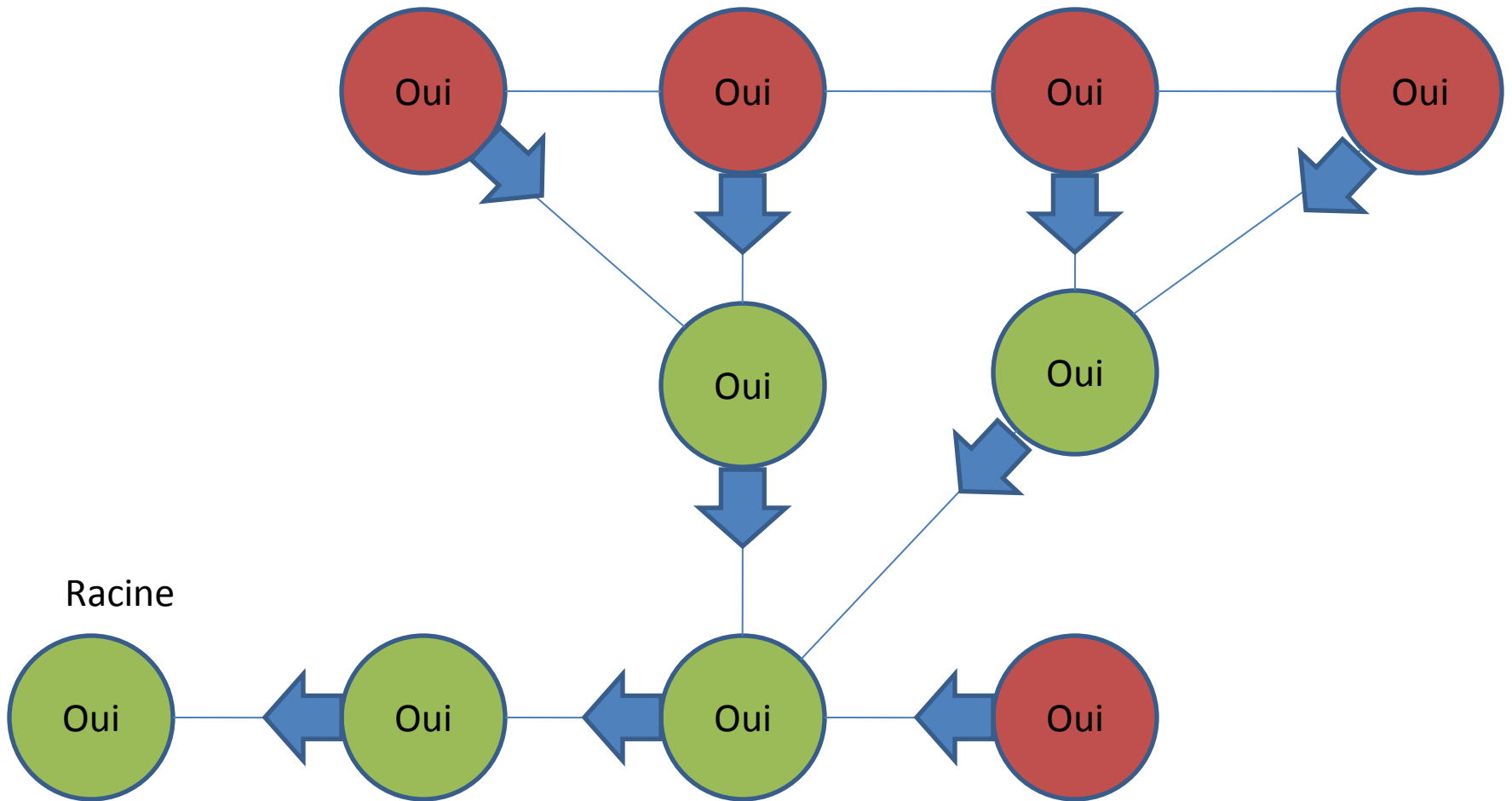
Détecter la fin de la construction de l'arbre couvrant : Méthode avancée



Détecter la fin de la construction de l'arbre couvrant : Méthode avancée



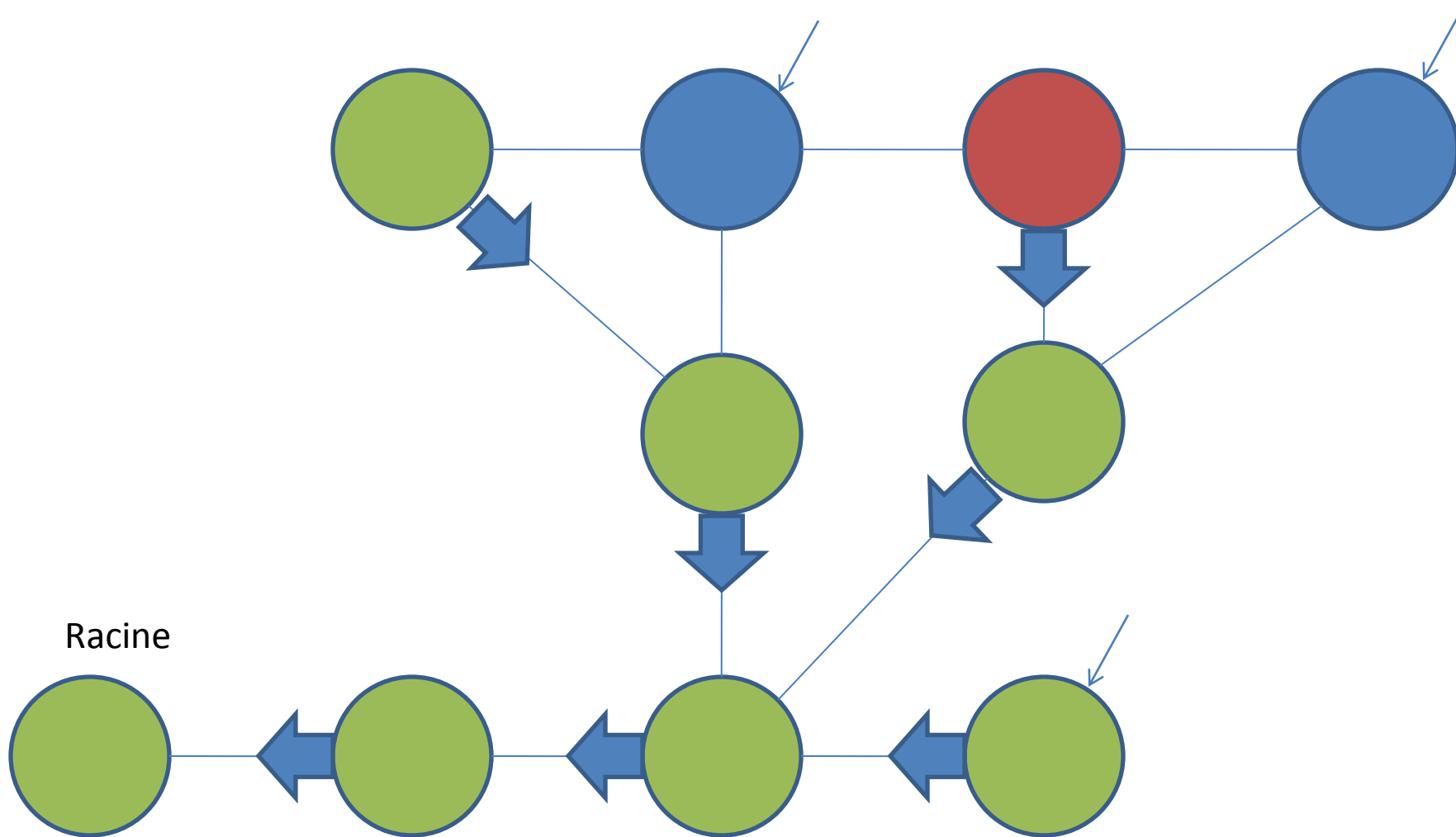
Détecter la fin de la construction de l'arbre couvrant : Méthode avancée



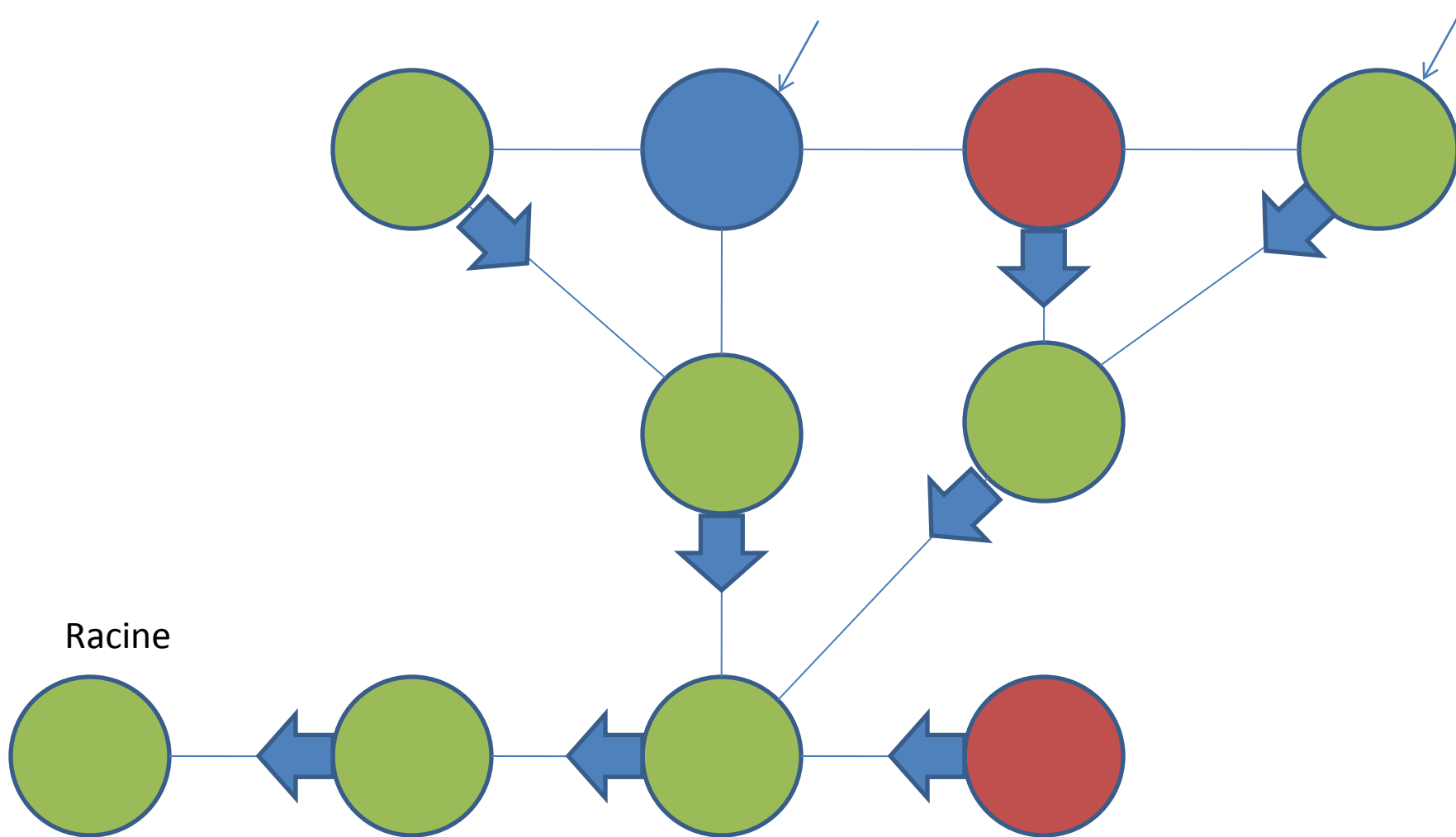
Comparaison des deux méthodes

- Méthode naïve :
 - Algorithme strictement auto-stabilisant
- Méthode avancée :
 - Algorithme instantanément stabilisant
 - Tout message émis par la racine sera reçu au moins une fois par chaque nœud
 - Tout message réellement émis par la racine sera reçu exactement une fois par chaque nœud.

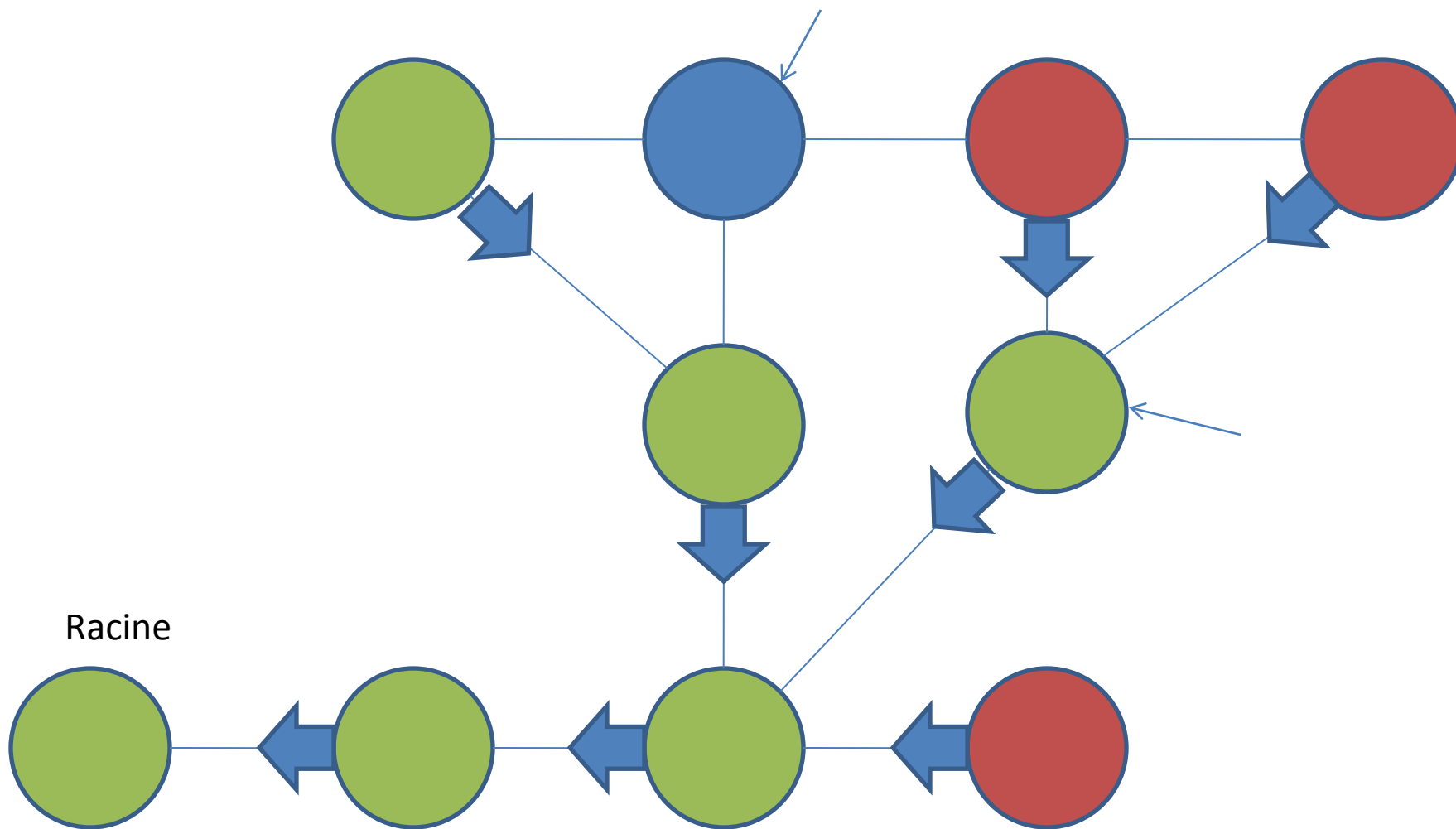
Le nettoyage : Méthode naïve



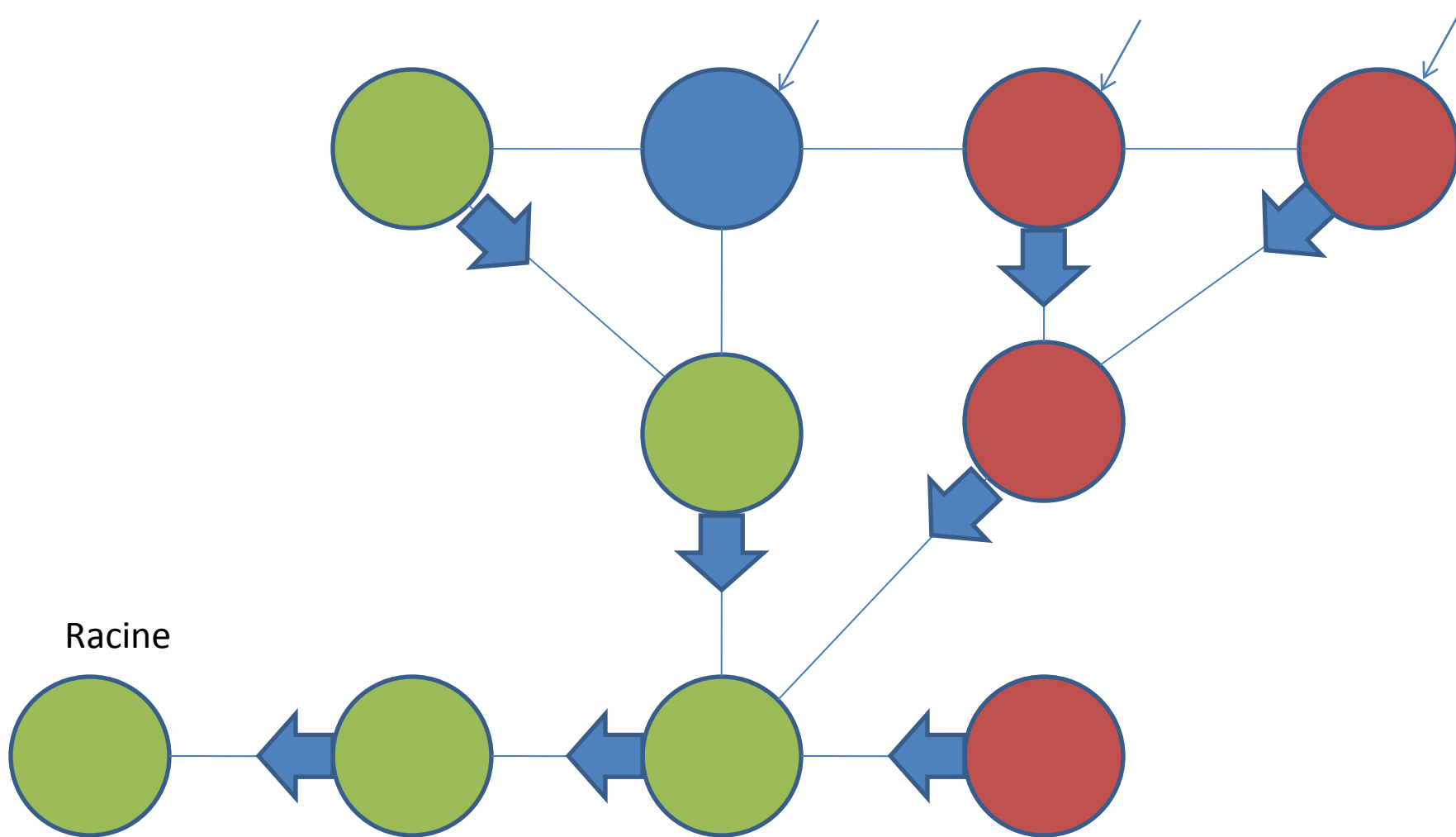
Le nettoyage : Méthode naïve



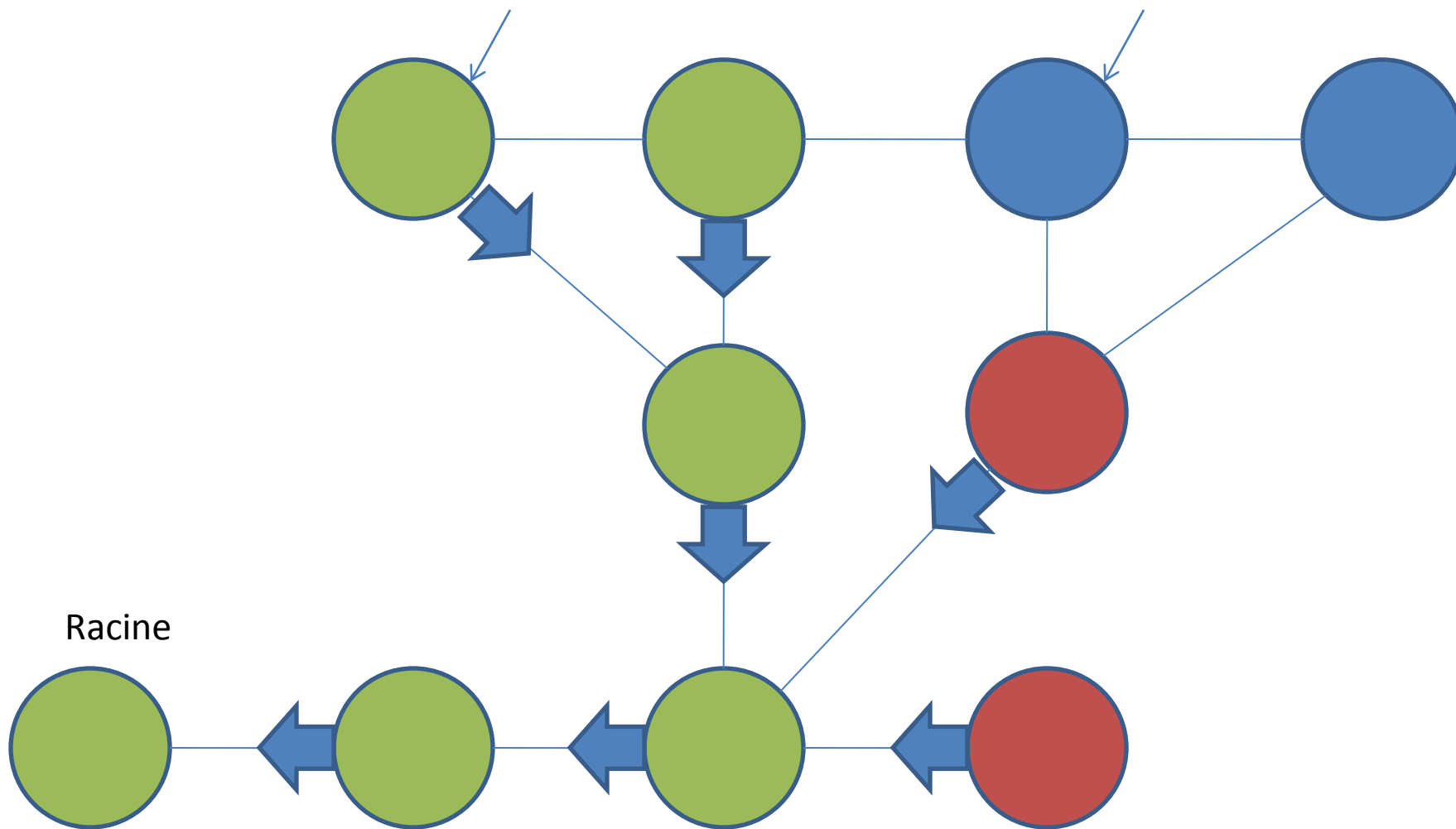
Le nettoyage : Méthode naïve



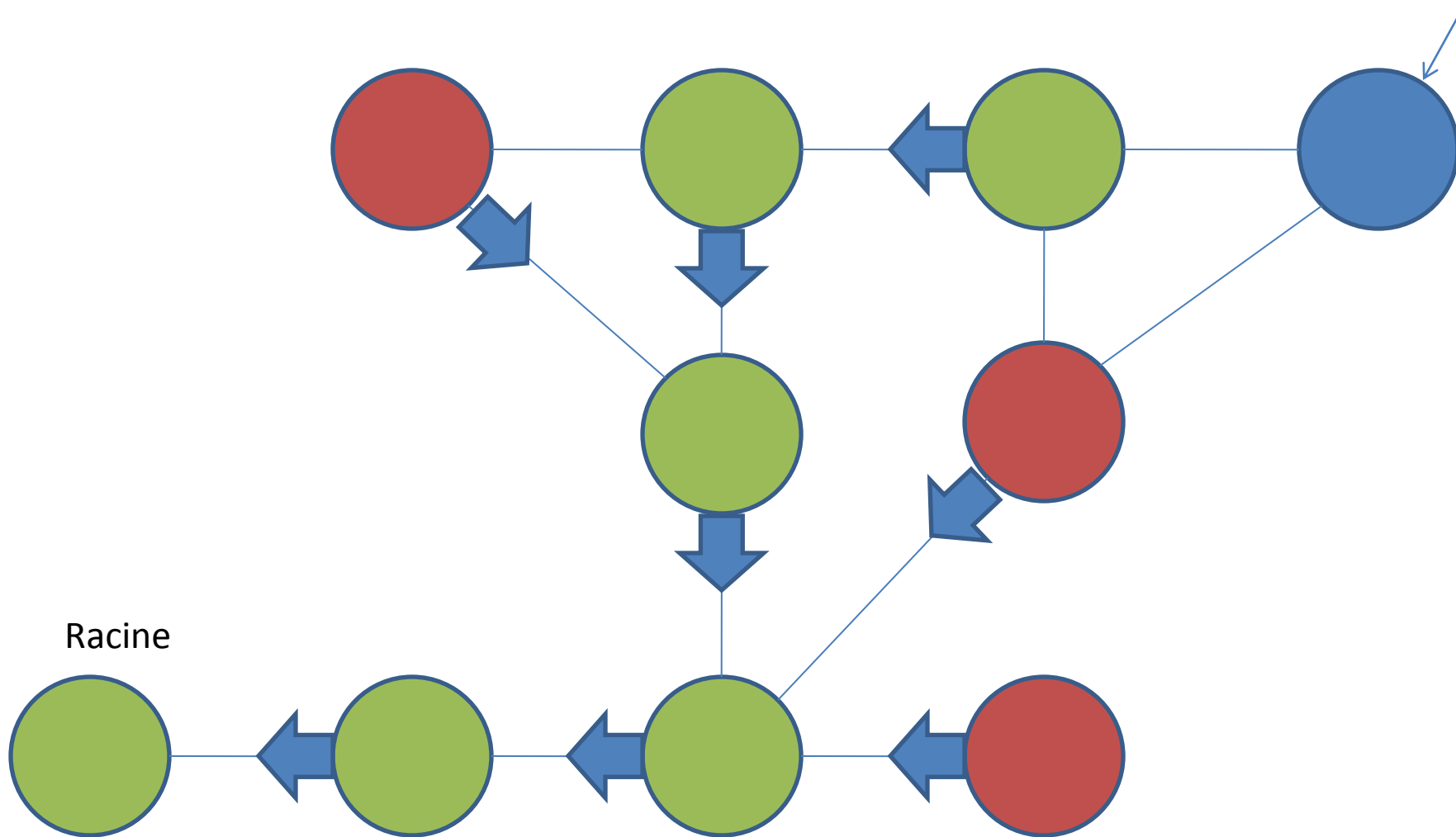
Le nettoyage : Méthode naïve



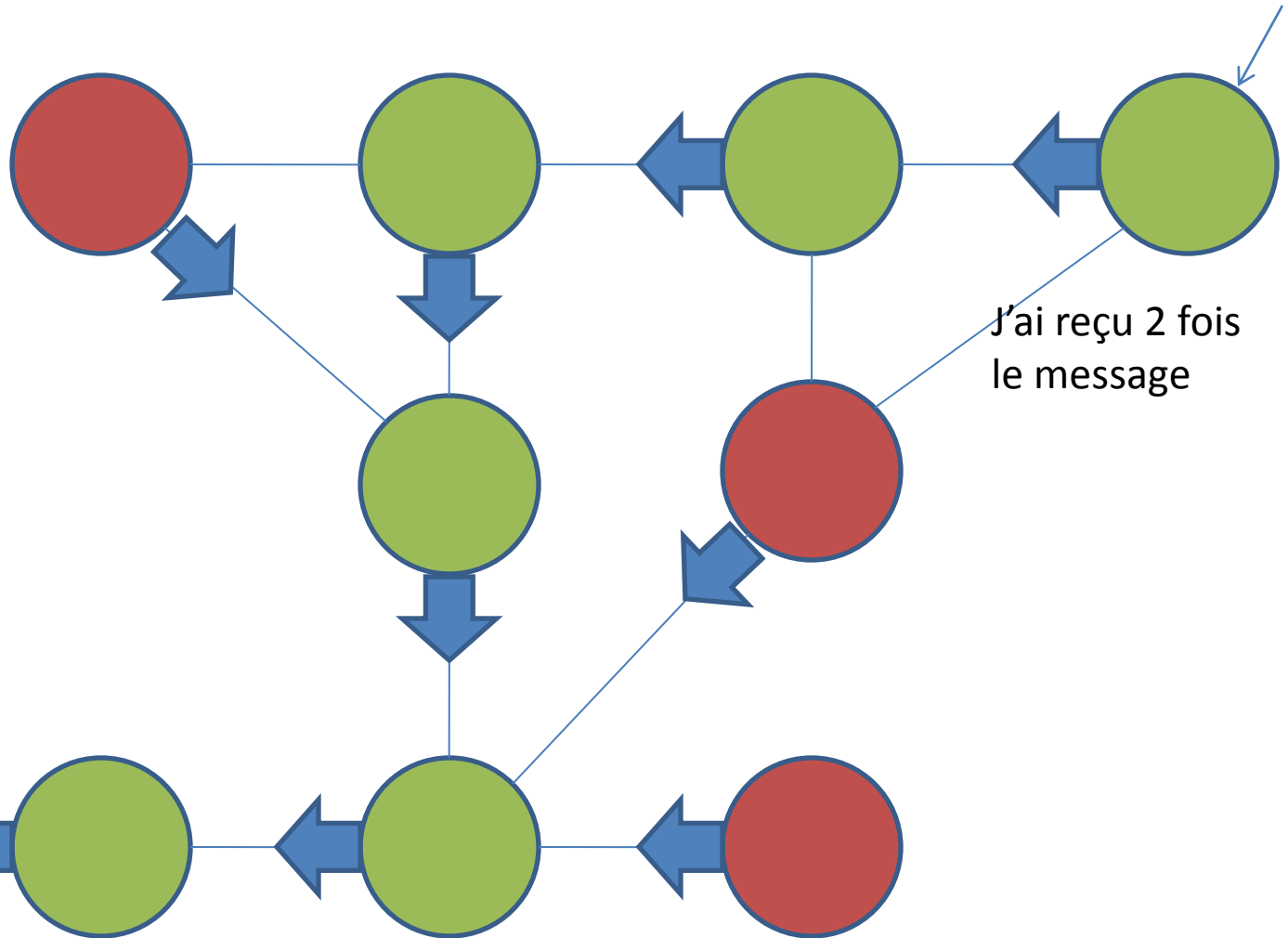
Le nettoyage : Méthode naïve



Le nettoyage : Méthode naïve



Le nettoyage : Méthode naïve

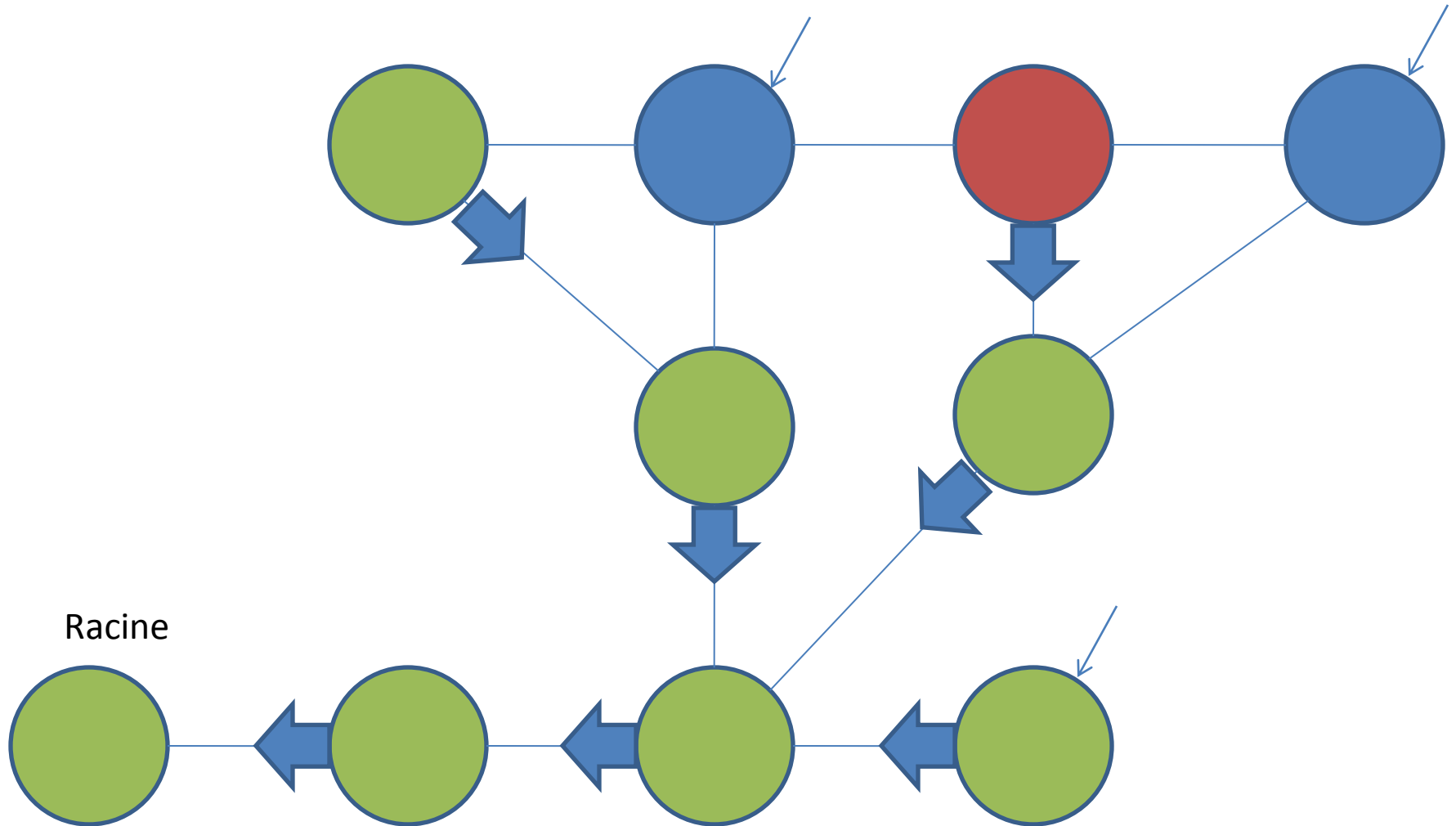


J'ai reçu 2 fois le message

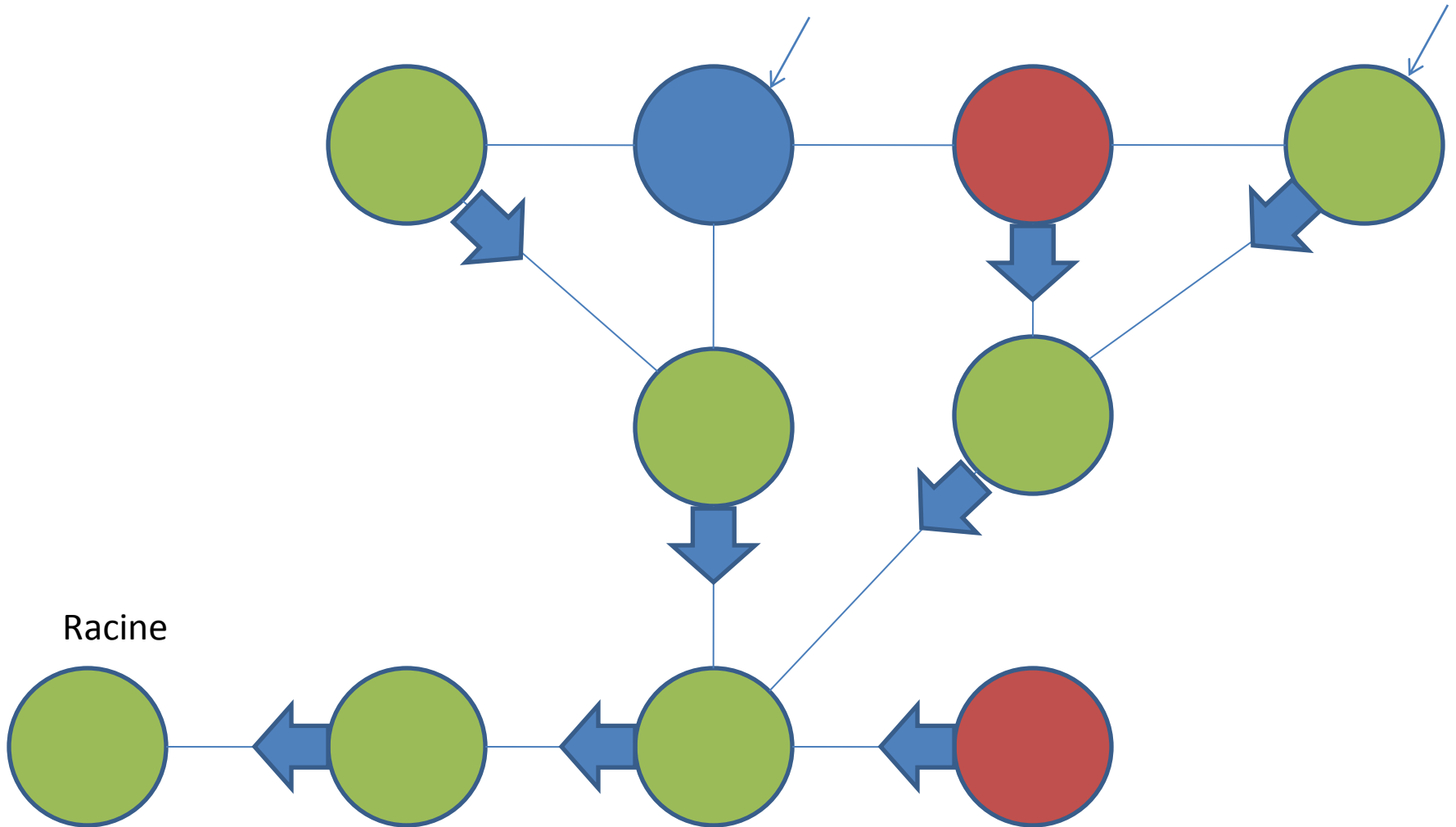
Racine

Je ne l'ai même pas envoyé

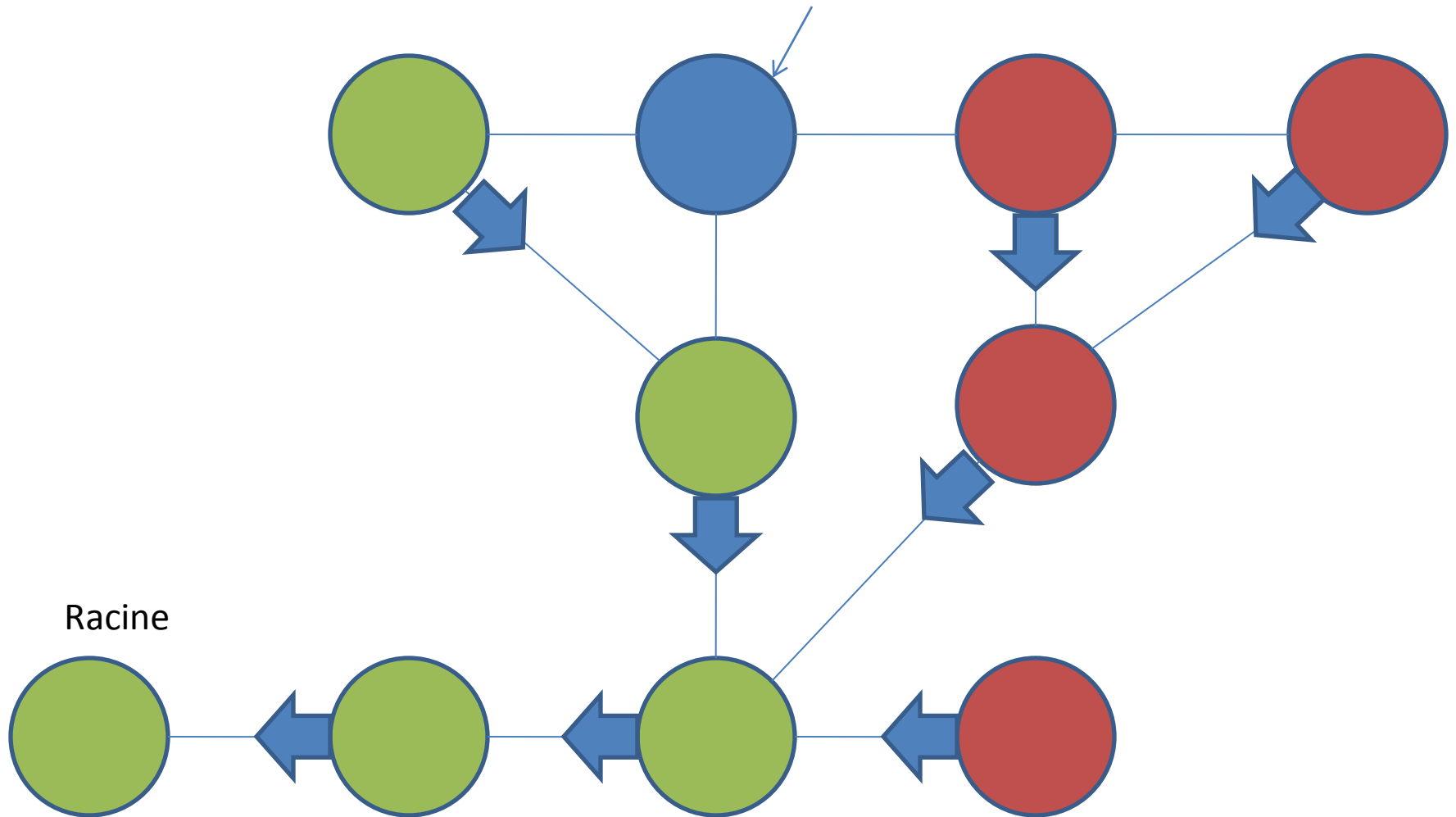
Le nettoyage : Méthode avancée



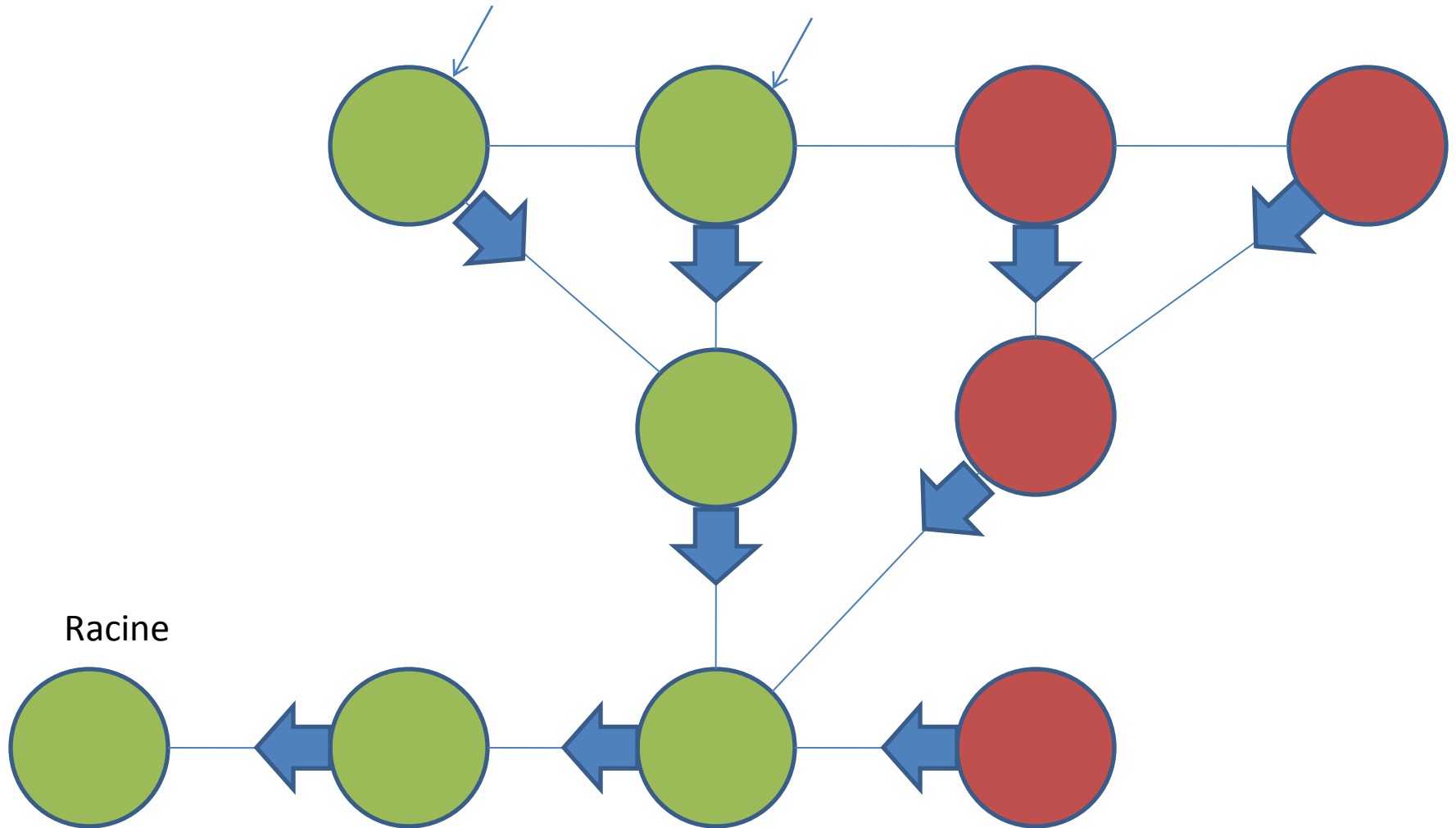
Le nettoyage : Méthode avancée



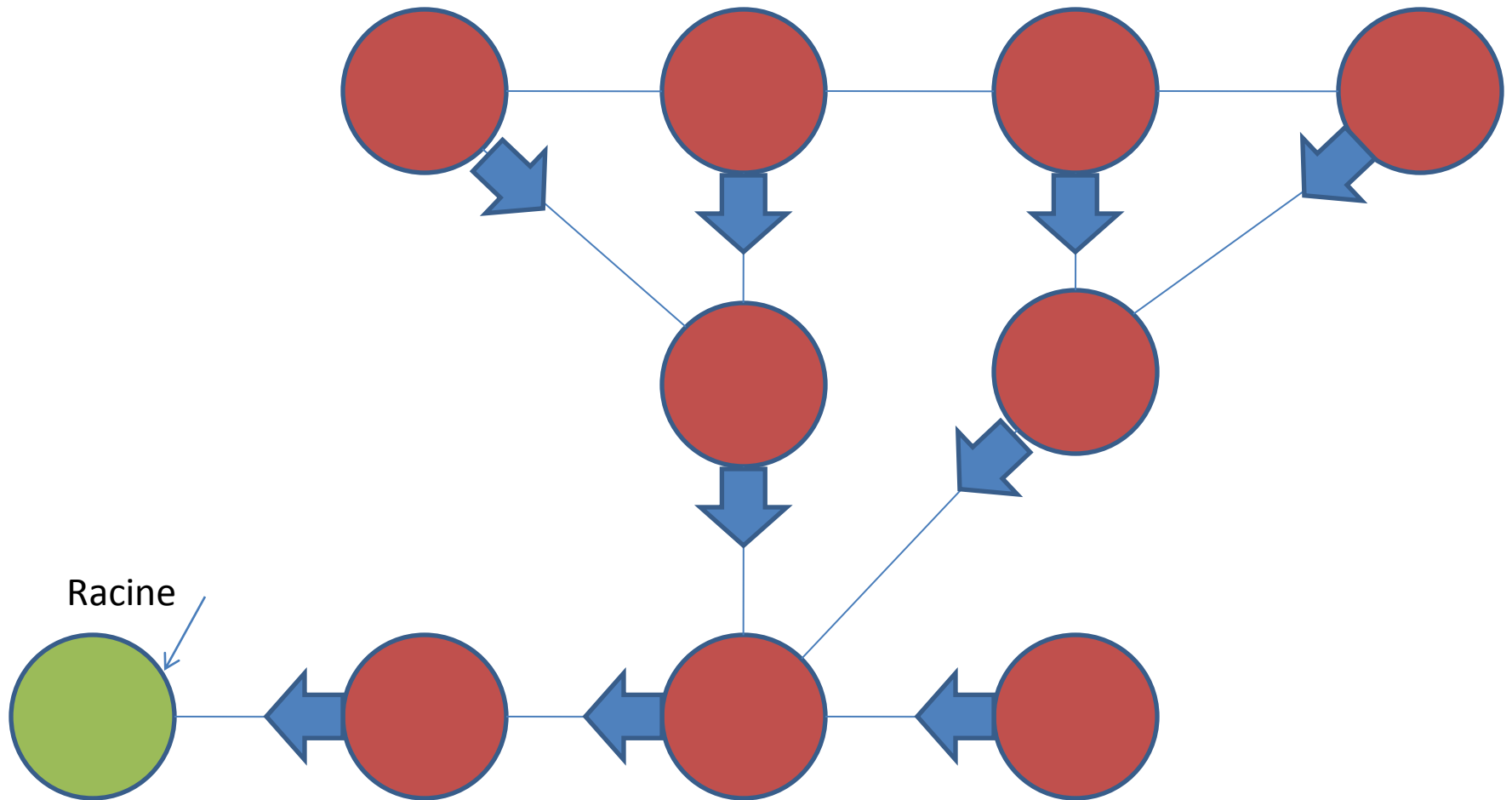
Le nettoyage : Méthode avancée



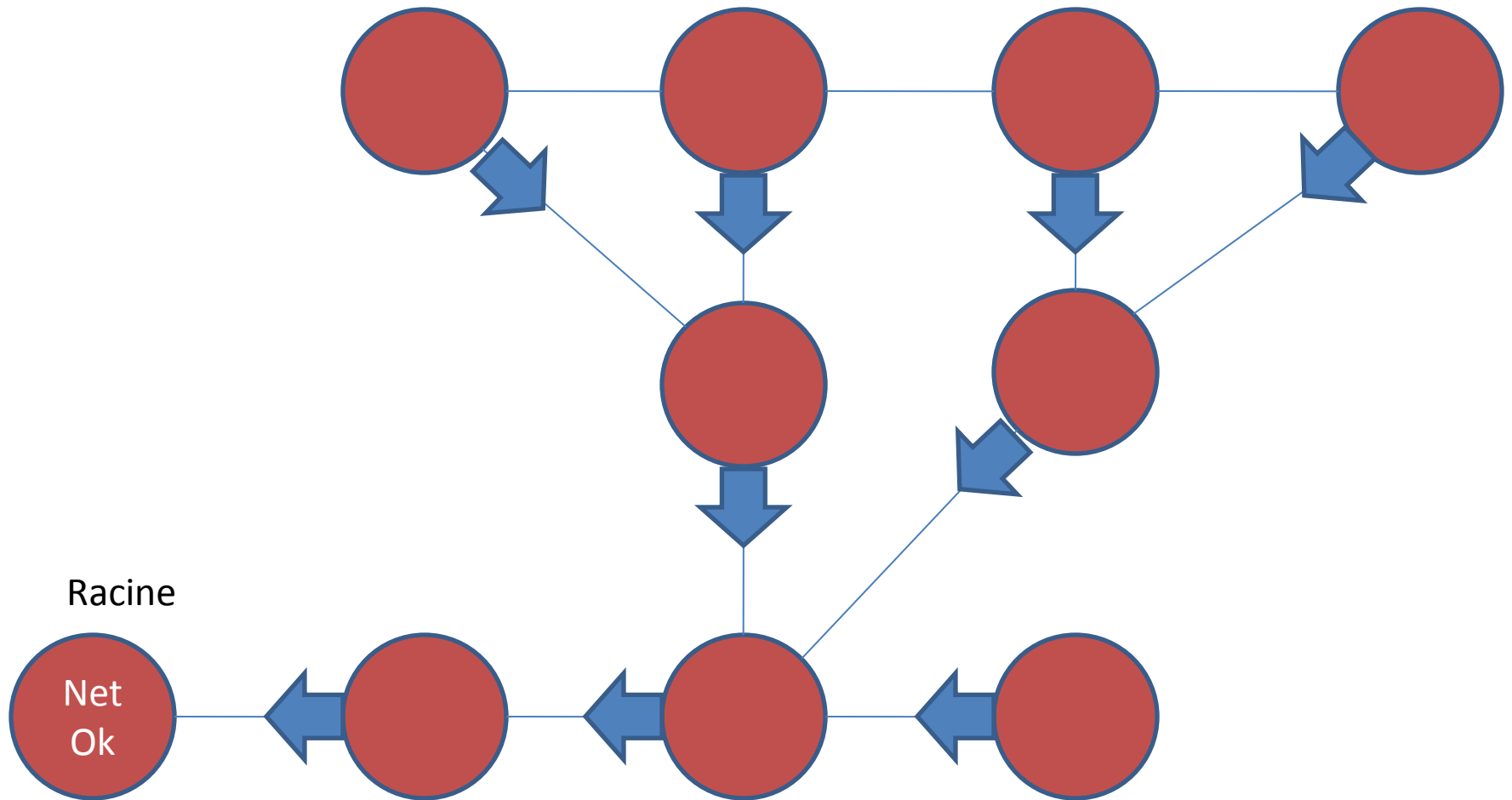
Le nettoyage : Méthode avancée



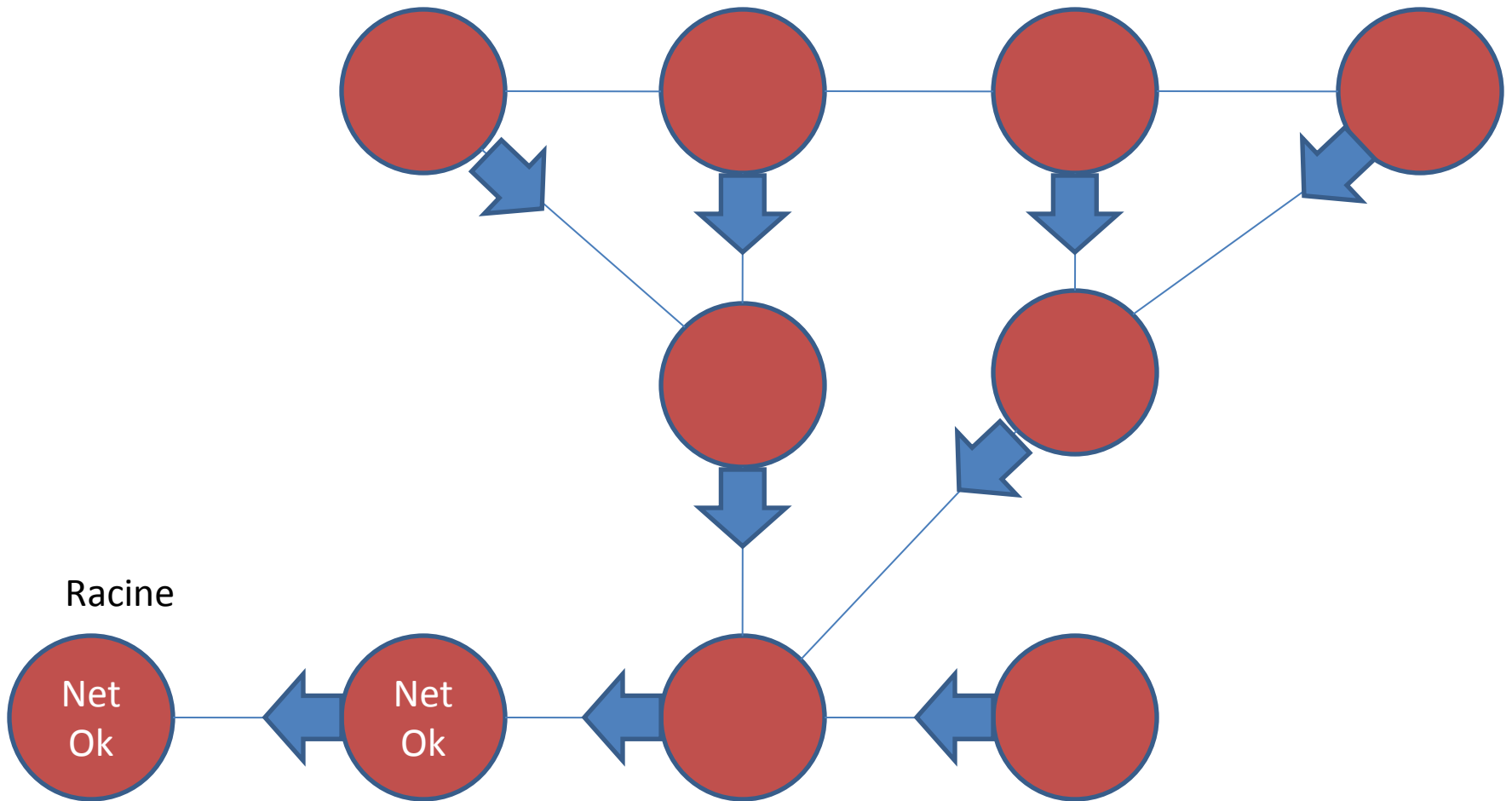
Le nettoyage : Méthode avancée



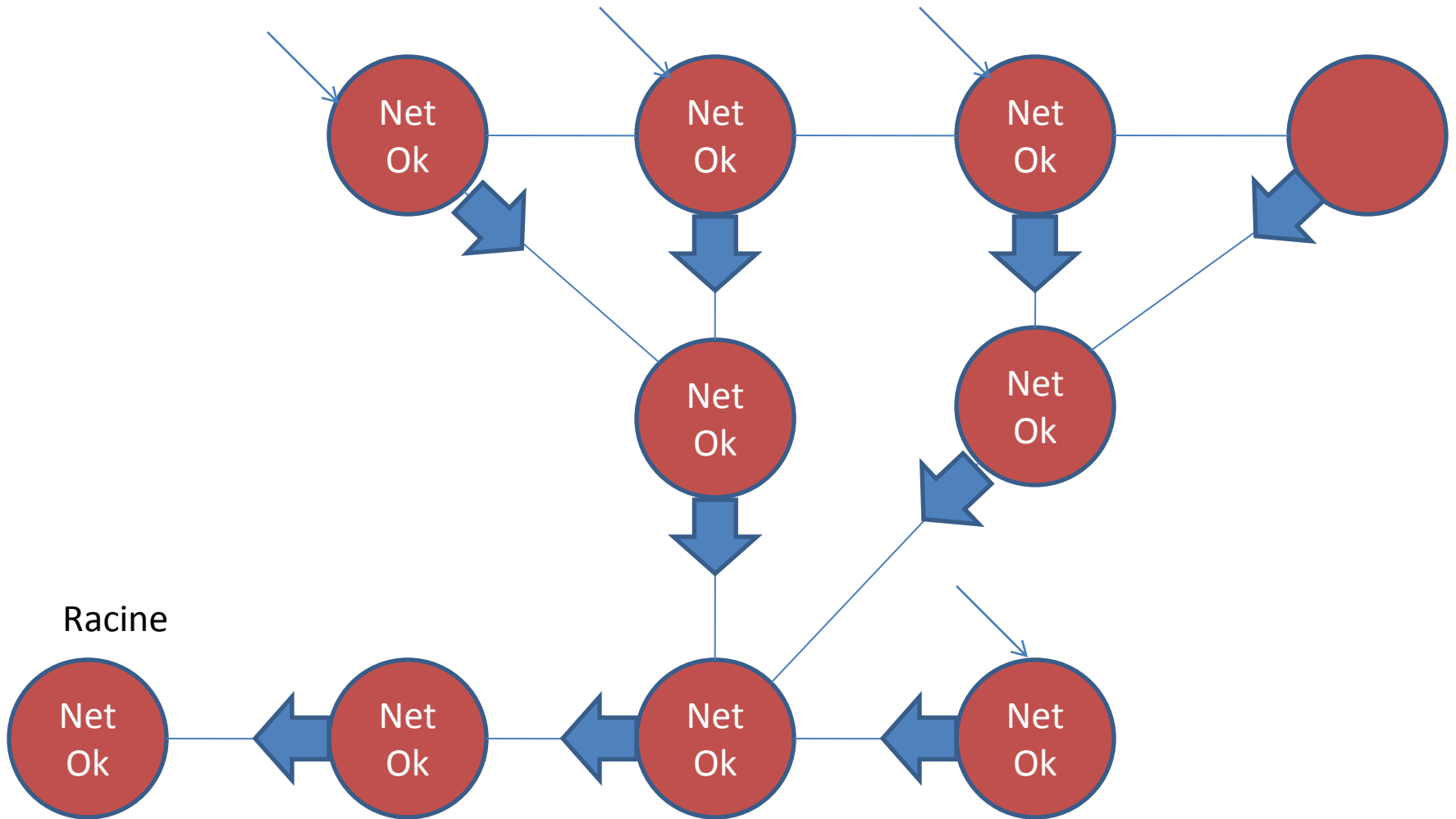
Le nettoyage : Méthode avancée



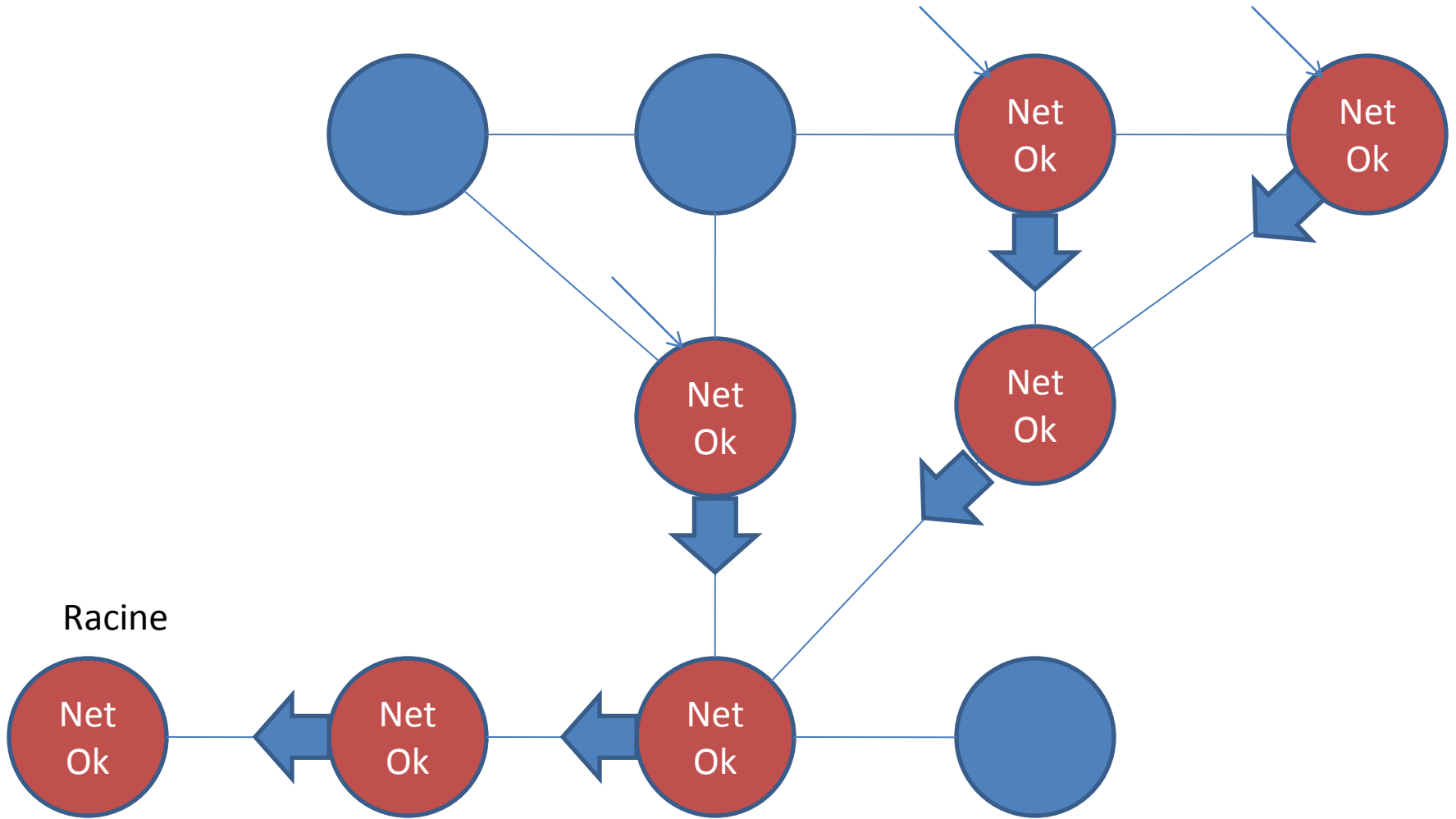
Le nettoyage : Méthode avancée



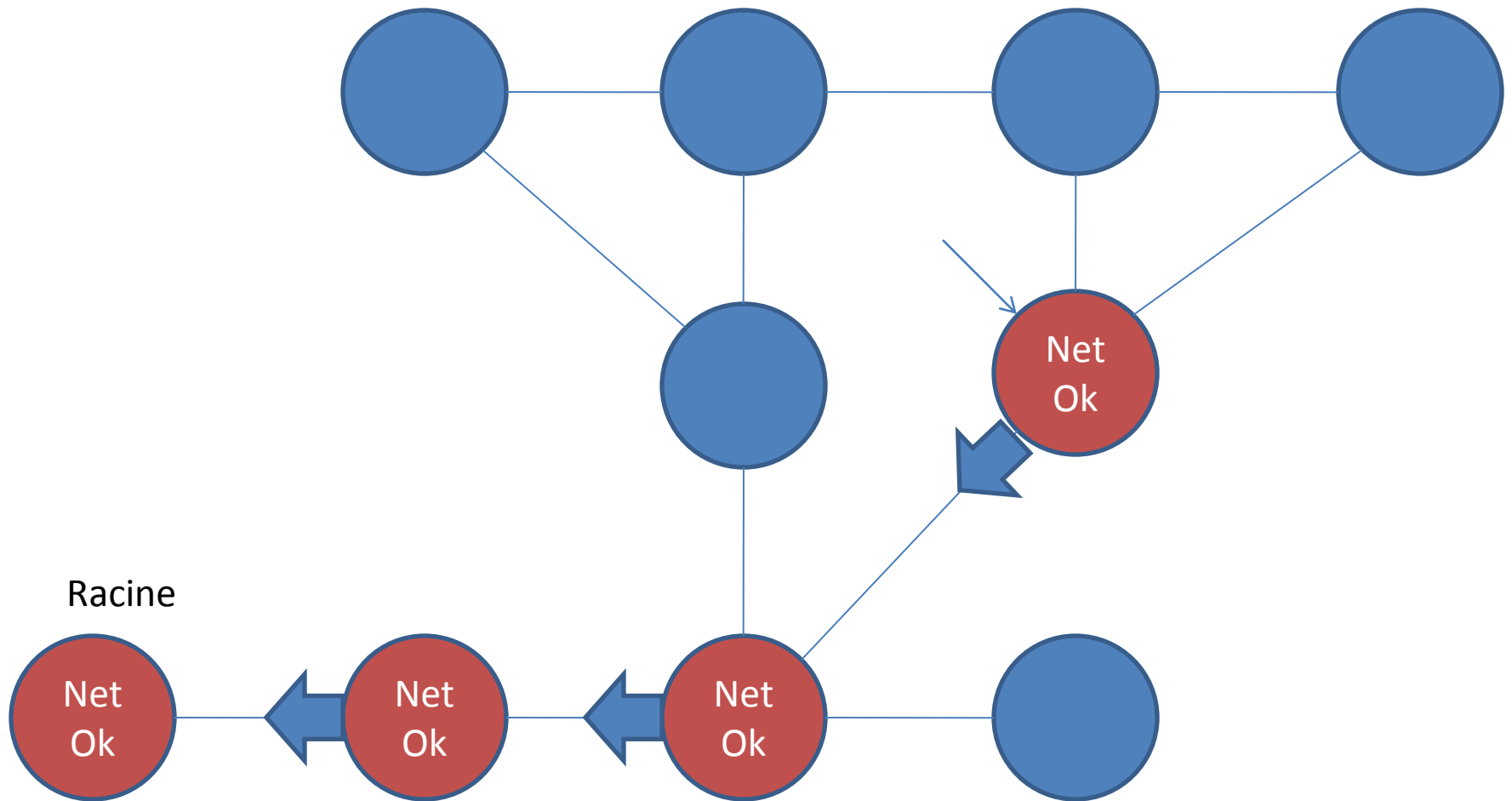
Le nettoyage : Méthode avancée



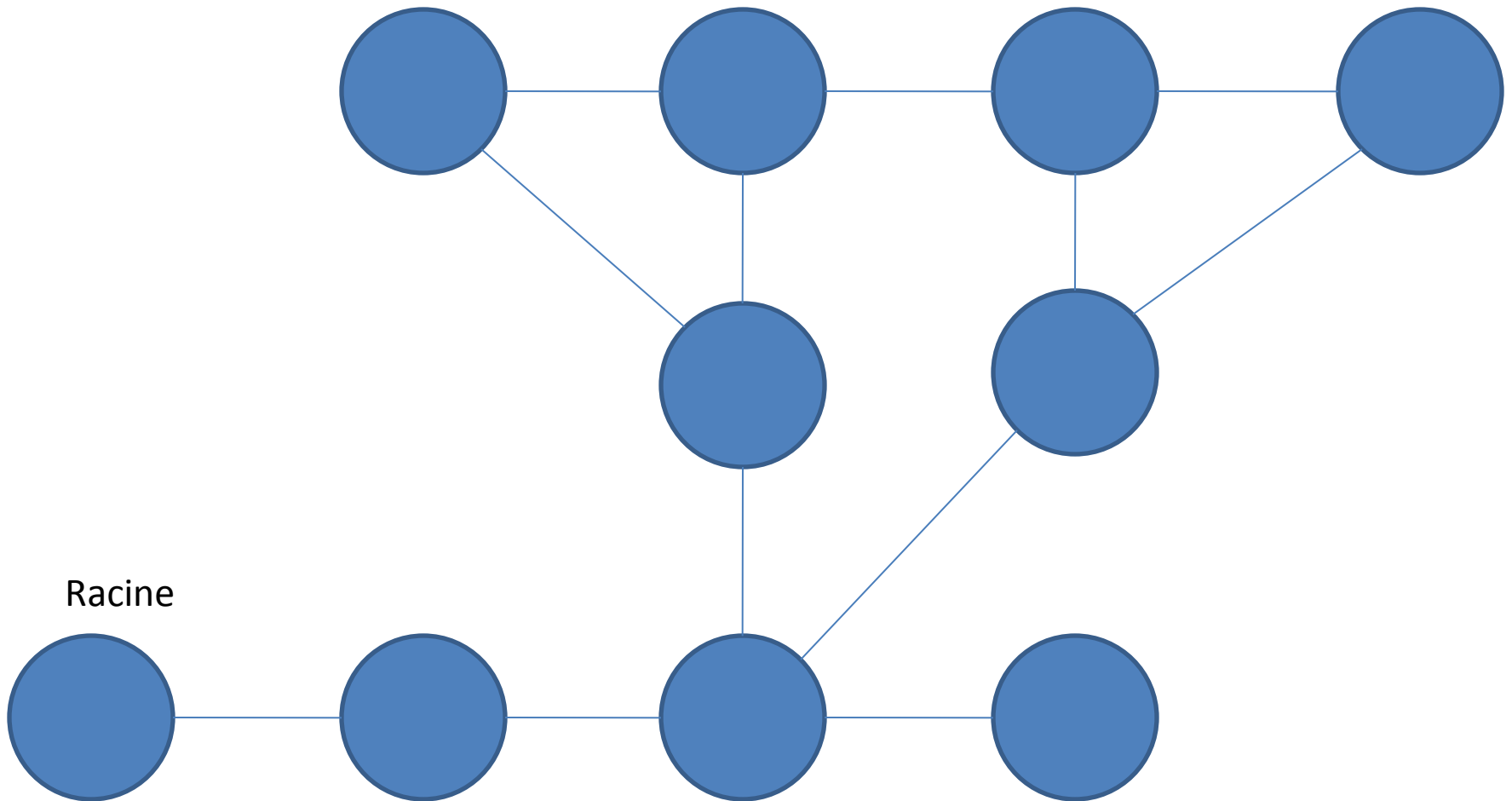
Le nettoyage : Méthode avancée



Le nettoyage : Méthode avancée



Le nettoyage : Méthode avancée



Comparaison des deux méthodes

- Méthode naïve :
 - Possibilité de recevoir plusieurs fois un même message issu de la racine.
- Méthode avancée :
 - Tout message issu de la racine sera reçu au plus une fois.

Bilan des quatre articles

	Gestion Erreurs	Fin Construction	Nettoyage
CDPV 2001	Naïve	Naïve	Naïve
CDPV 2002	Naïve	Autre méthode	Naïve
BCV 2002	Avancée	Avancée	Naïve
CDV 2006 et 2009	Avancée	Avancée	Avancée

Bilan des quatre articles :

Côté ordonnateur

	Nombre de fois où le service doit être actionné pour être rendu une fois	Algorithme
CDPV 2001	Non borné	Auto-stabilisant
CDPV 2002	1	Instantanément stabilisant
BCV 2002	1	Instantanément stabilisant
CDV 2006 et 2009	1	Instantanément stabilisant

Bilan des quatre articles : Côté destinataires

	Nombre de fois où le service est faux	Non détecté comme faux
CDPV 2001	Non borné	Non borné
CDPV 2002	Beaucoup	Beaucoup
BCV 2002	Beaucoup	Beaucoup
CDV 2006 et 2009	n	2

Beaucoup veut dire sans doute exponentiel mais non prouvé

Bilan des quatre articles :

Côté réseau

	Nombre de rondes	Nombre de pas de calcul
CDPV 2001	Proportionnel à la borne	Non borné
CDPV 2002	Proportionnel à la borne	Beaucoup
BCV 2002	n	Beaucoup
CDV 2006 et 2009	n	n^3

Beaucoup veut dire sans doute exponentiel mais non prouvé

Bilan des quatre articles

- Le quatrième article autorise un protocole pour authentifier les messages.
 - L'ordonnateur envoie 3 fois successivement le même message m .
 - À la troisième réception successive d'un même message m le destinataire est sûr de sa validité.

Questions

- Est-on obligé de coupler la construction de l'arbre et la diffusion du message ?
- Le nettoyage doit-il obligatoirement détruire la structure d'arbre couvrant ?
- Le réseau doit-il donner immédiatement le message au destinataire ?

Conséquences

- Tout problème admettant une solution auto-stabilisante admet une solution instantanément stabilisante (Transformateur « théorique »).
- Cette transformation peut être efficace.
- Exclusion mutuelle instantanément stabilisante triviale.

Conséquences

- Les outils développés pour ce problème ont été transposés dans les problèmes suivant :
 1. Construction d'arbre en profondeur
 2. Construction d'arbre couvrant

Dans ces deux cas ces outils ont été appropriés pour répondre aux questions de qualité de service.

Autre application de la méthode

- L'acheminement de messages.

Perspectives

- Quels sont les problèmes pour lesquels il existe un algorithme stabilisant dont l'exécution respecte les deux propriétés suivantes :
 - Le nombre de rondes est borné par un polynôme en le Diamètre du réseau;
 - Le nombre d'étapes est borné par un polynôme en le nombre de nœuds du réseau.

Perspectives

- Que dire lorsque l'on change de modèle :
 - Le modèle à registres;
 - Le modèle à passage de messages.
- La série de questions posées pour le problème de la diffusion est-elle transposable vers d'autres problèmes ?

Publications

- 4 articles dans des journaux internationaux ;
- 17 articles dans des conférences internationales avec comité de lectures et actes ;
- 2 conférences sur invitation.

Questions ?

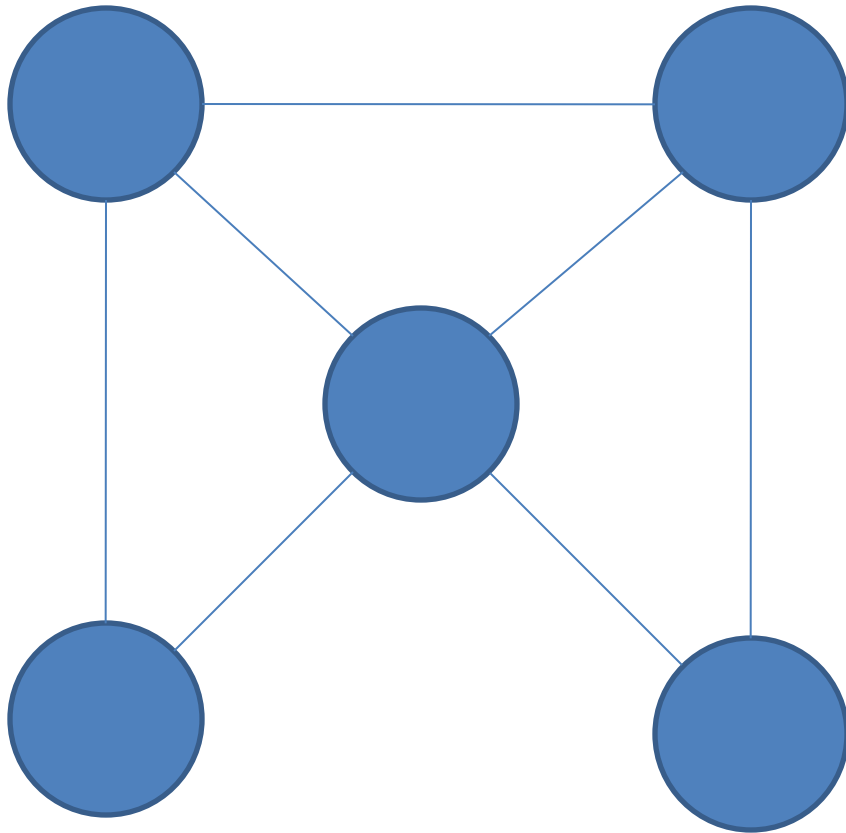
Faut-il seulement compter les rondes ?

Le cas de l'arbre couvrant

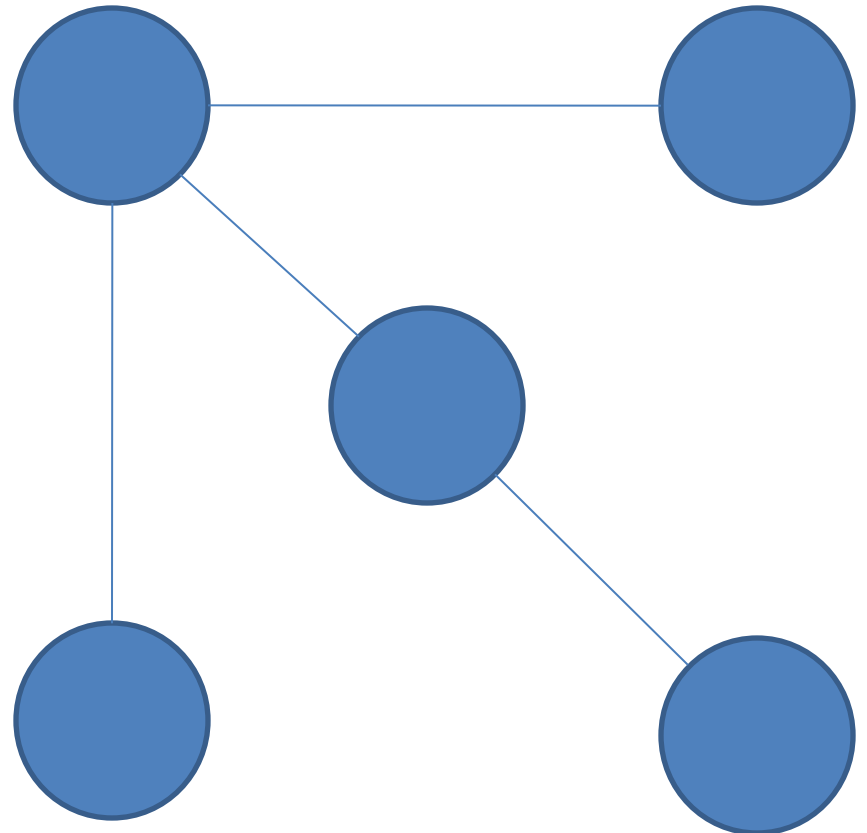
- $G'=(V',E')$ est un arbre couvrant du graphe connexe $G=(V,E)$ si et seulement si :
 - $V' = V$
 - E' est un sous-ensemble de E
 - $|E'| = |V| - 1$
 - G' est connexe

Un exemple

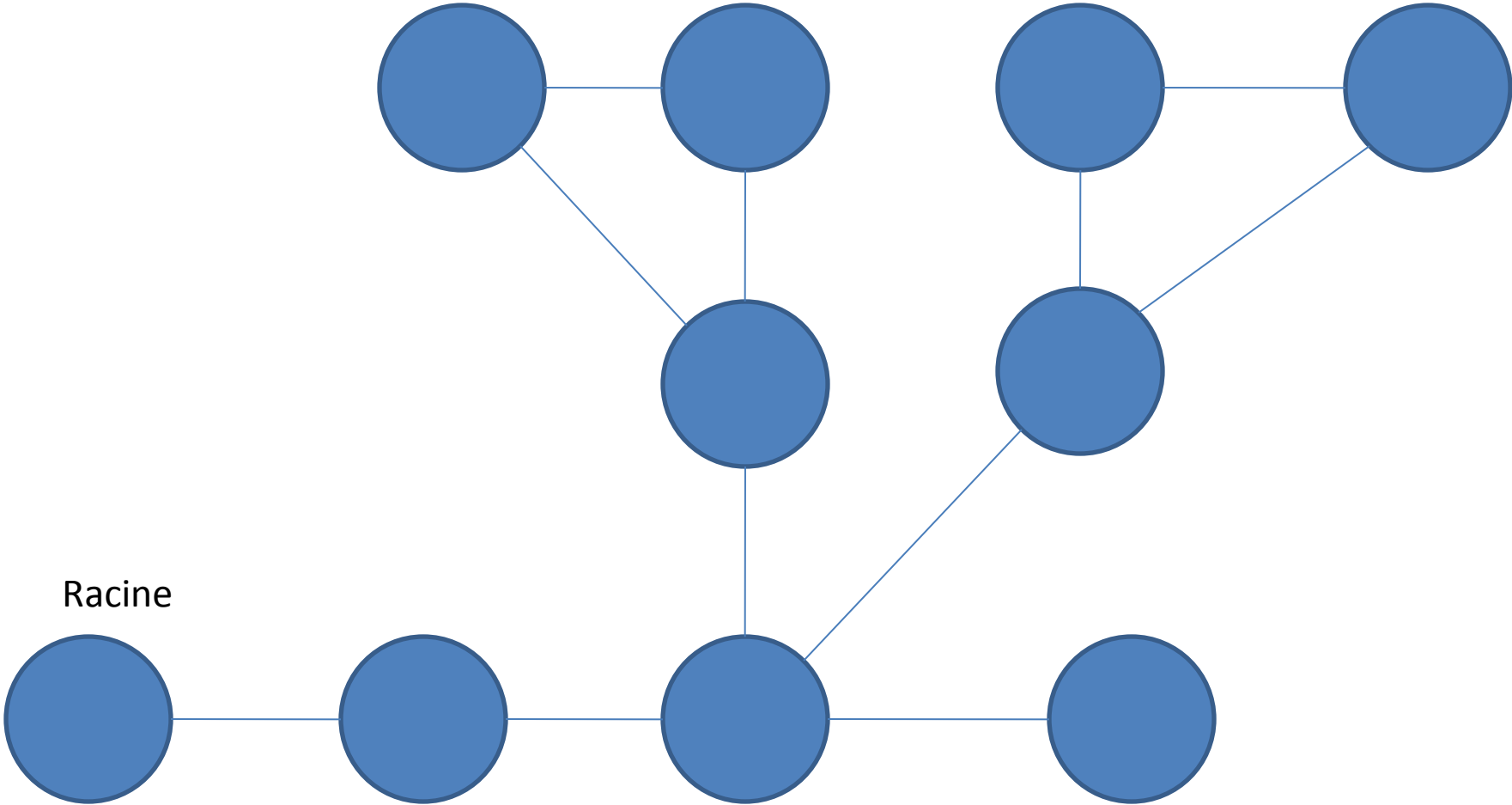
- Un graphe connexe



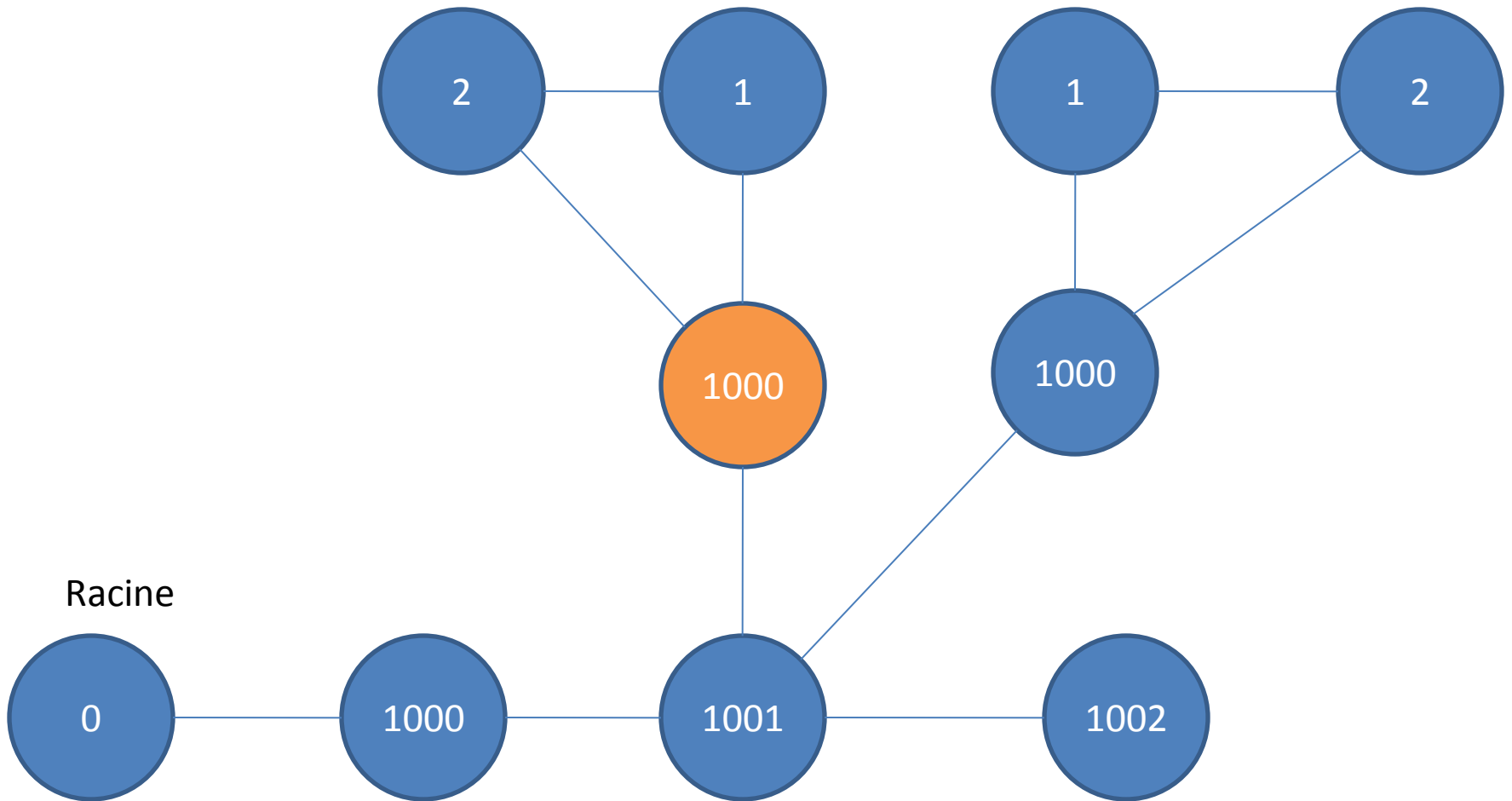
- Un arbre couvrant



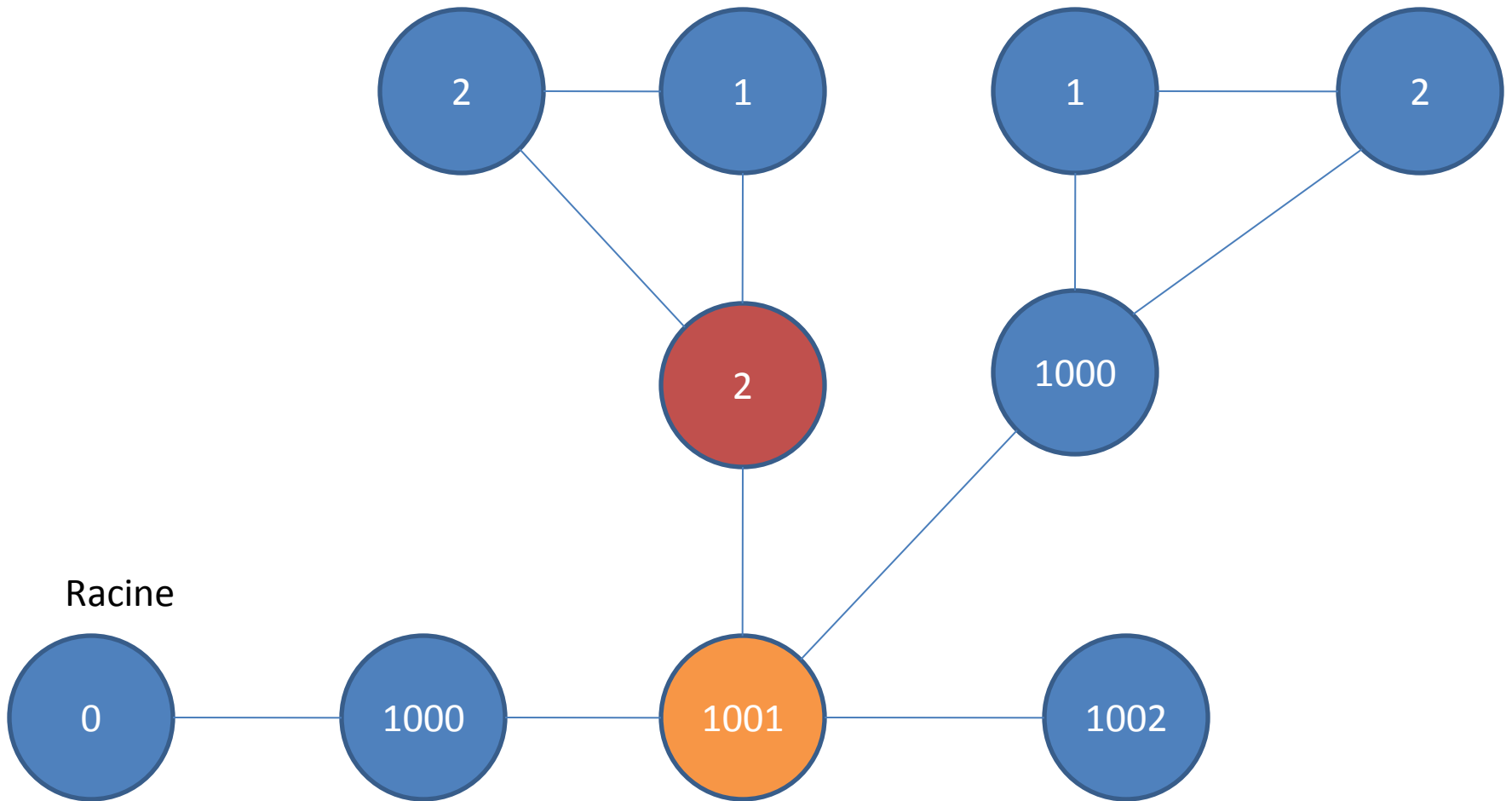
Un exemple : L'algorithme du Min+1



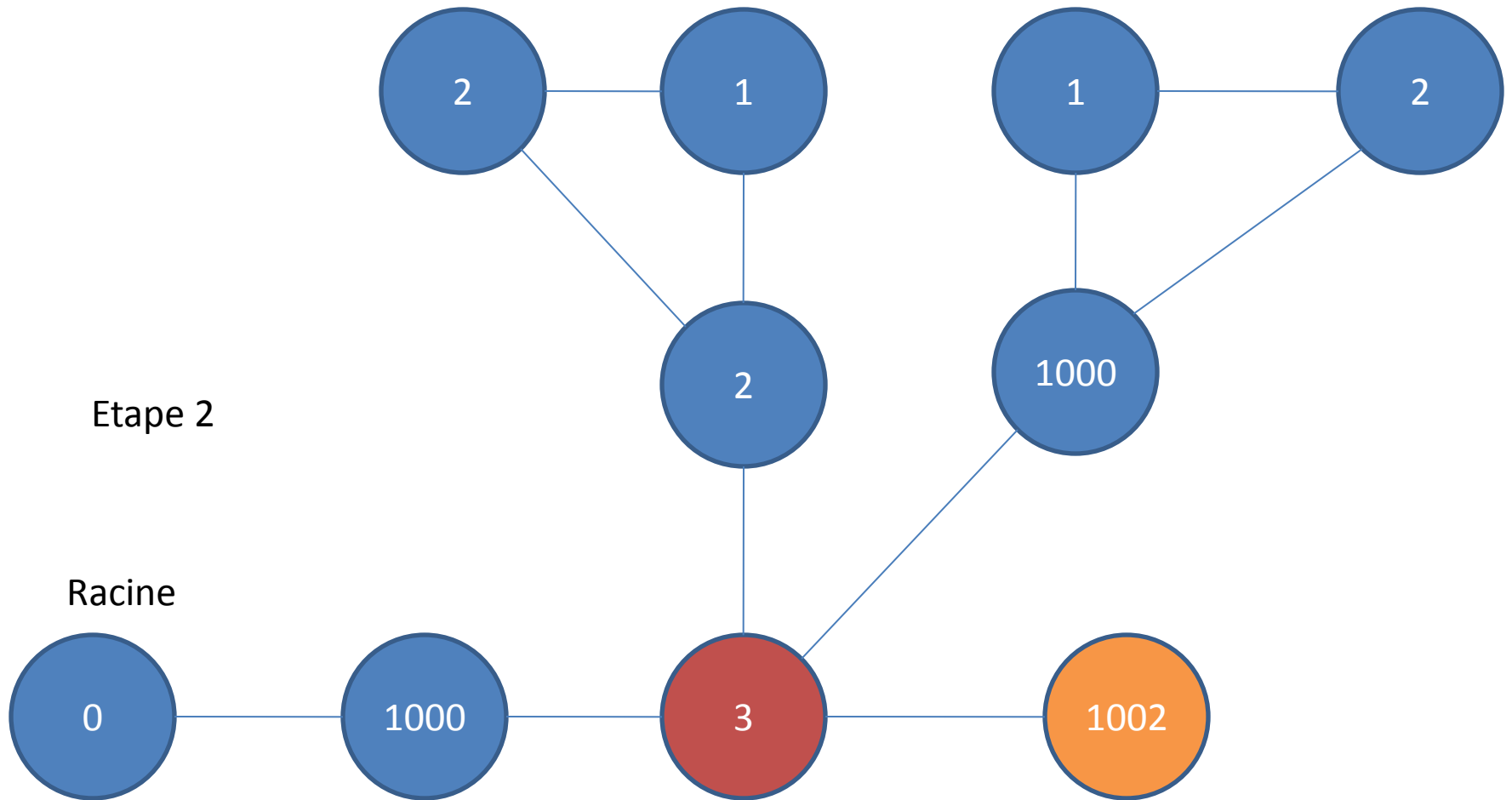
Un exemple : L'algorithme du Min+1



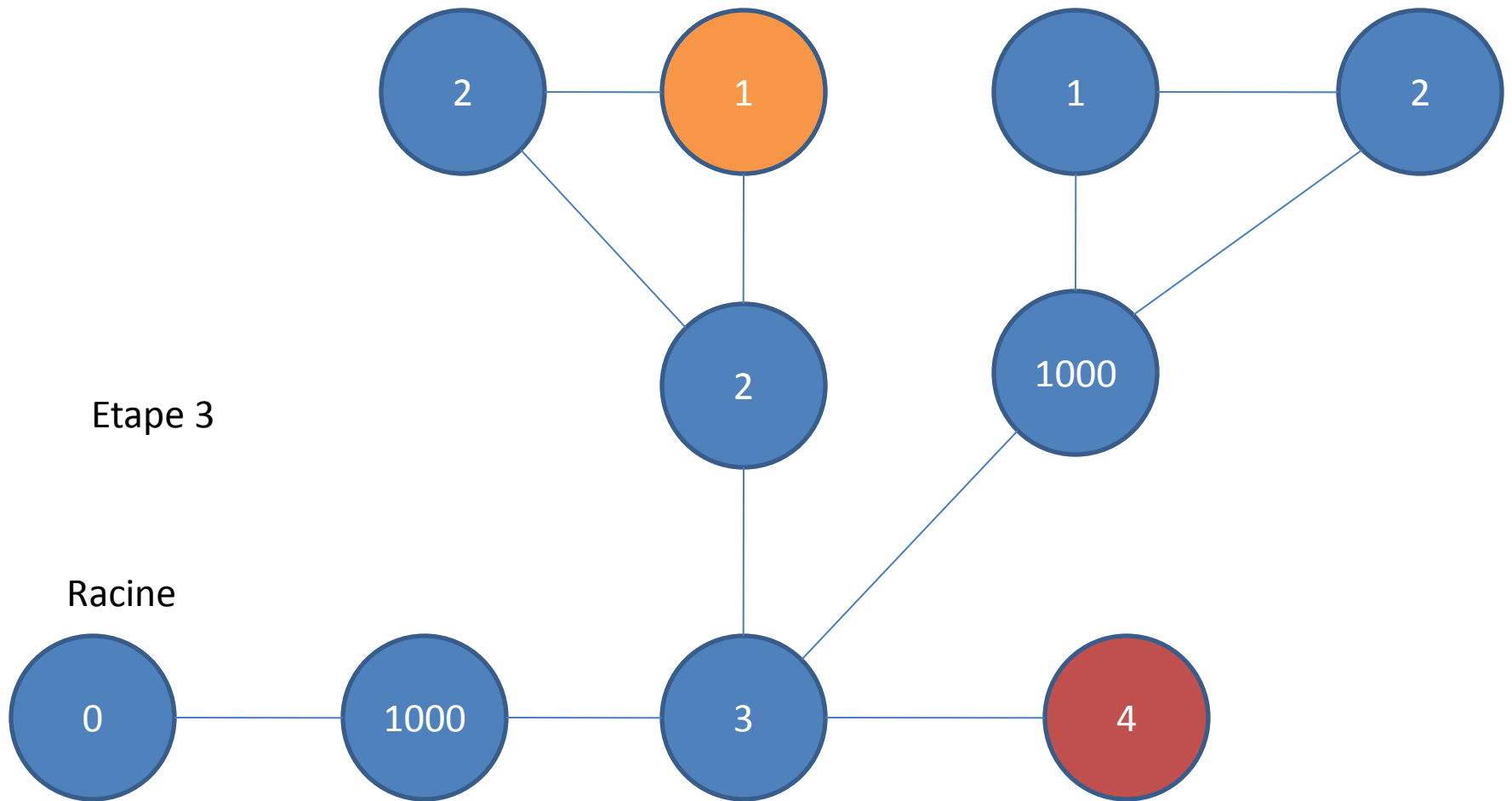
Un exemple : L'algorithme du Min+1



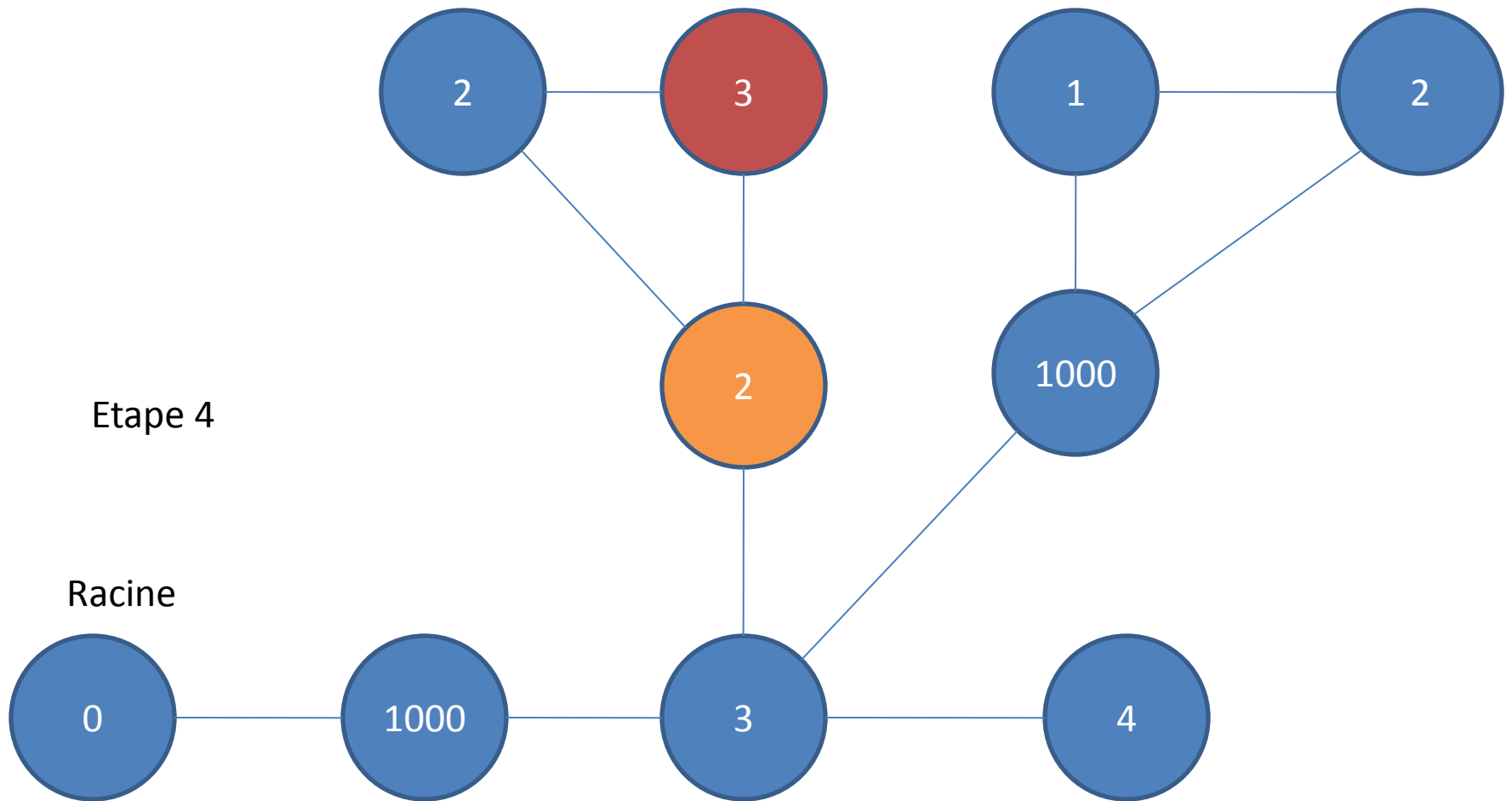
Un exemple : L'algorithme du Min+1



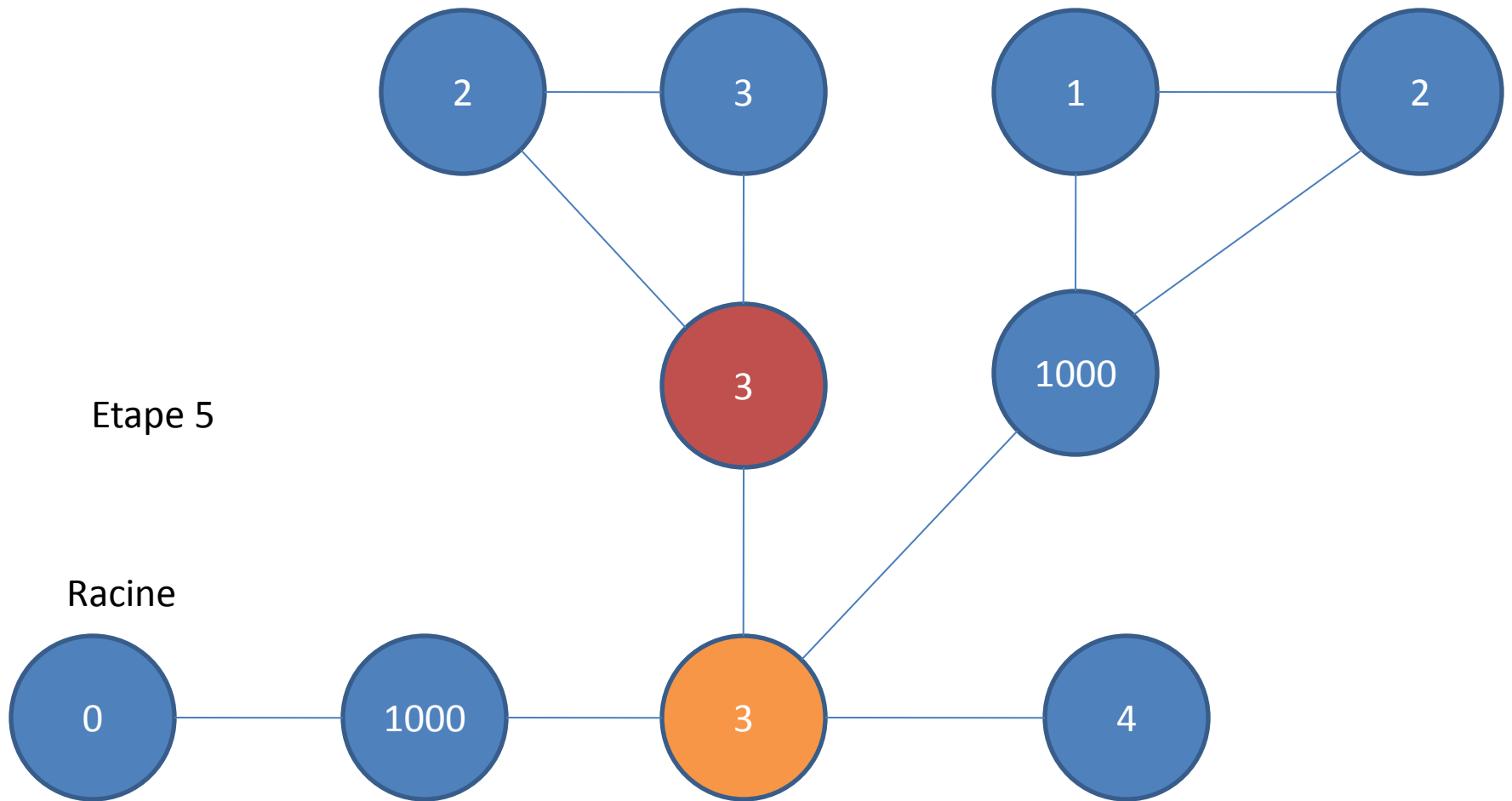
Un exemple : L'algorithme du Min+1



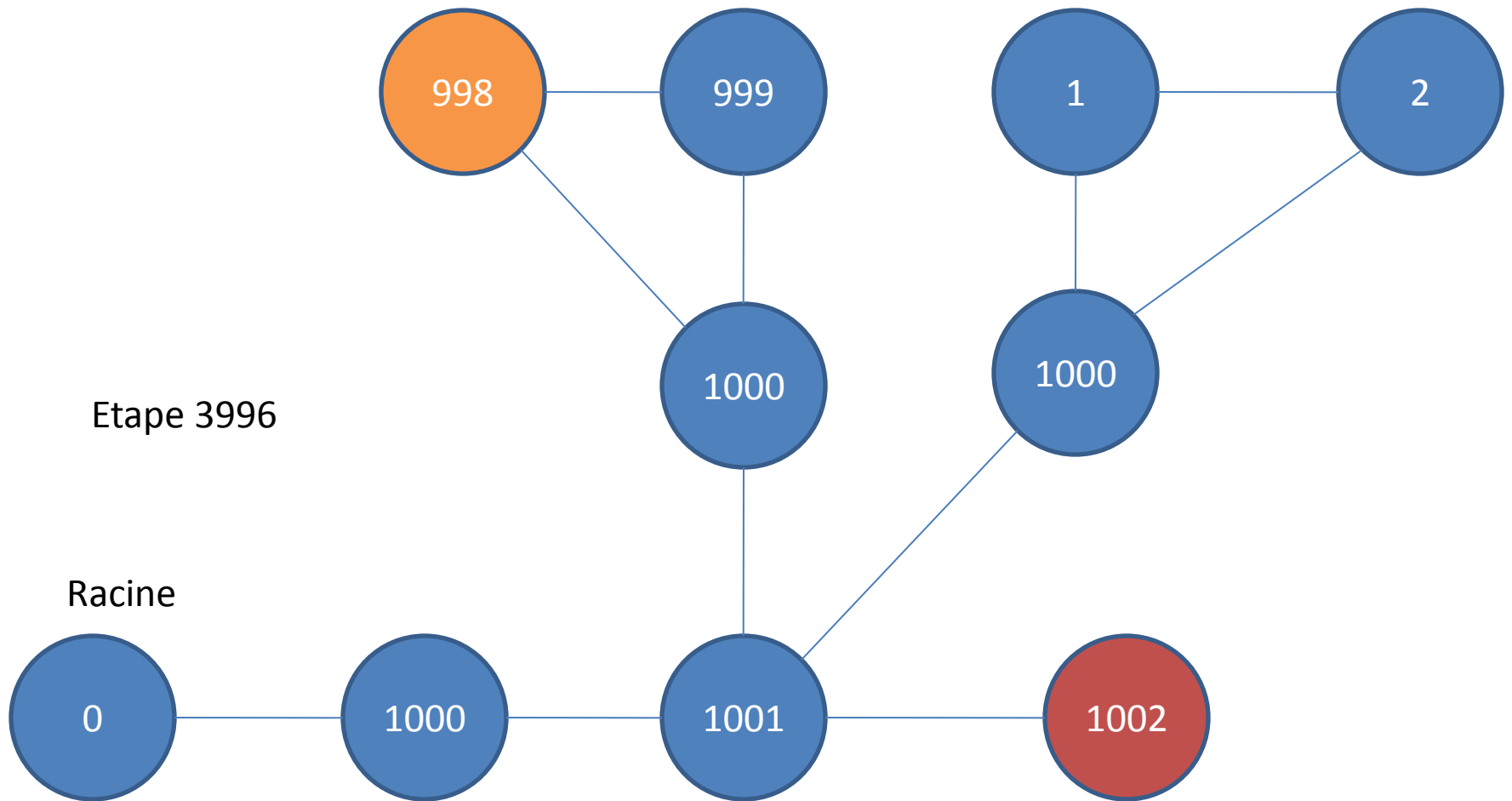
Un exemple : L'algorithme du Min+1



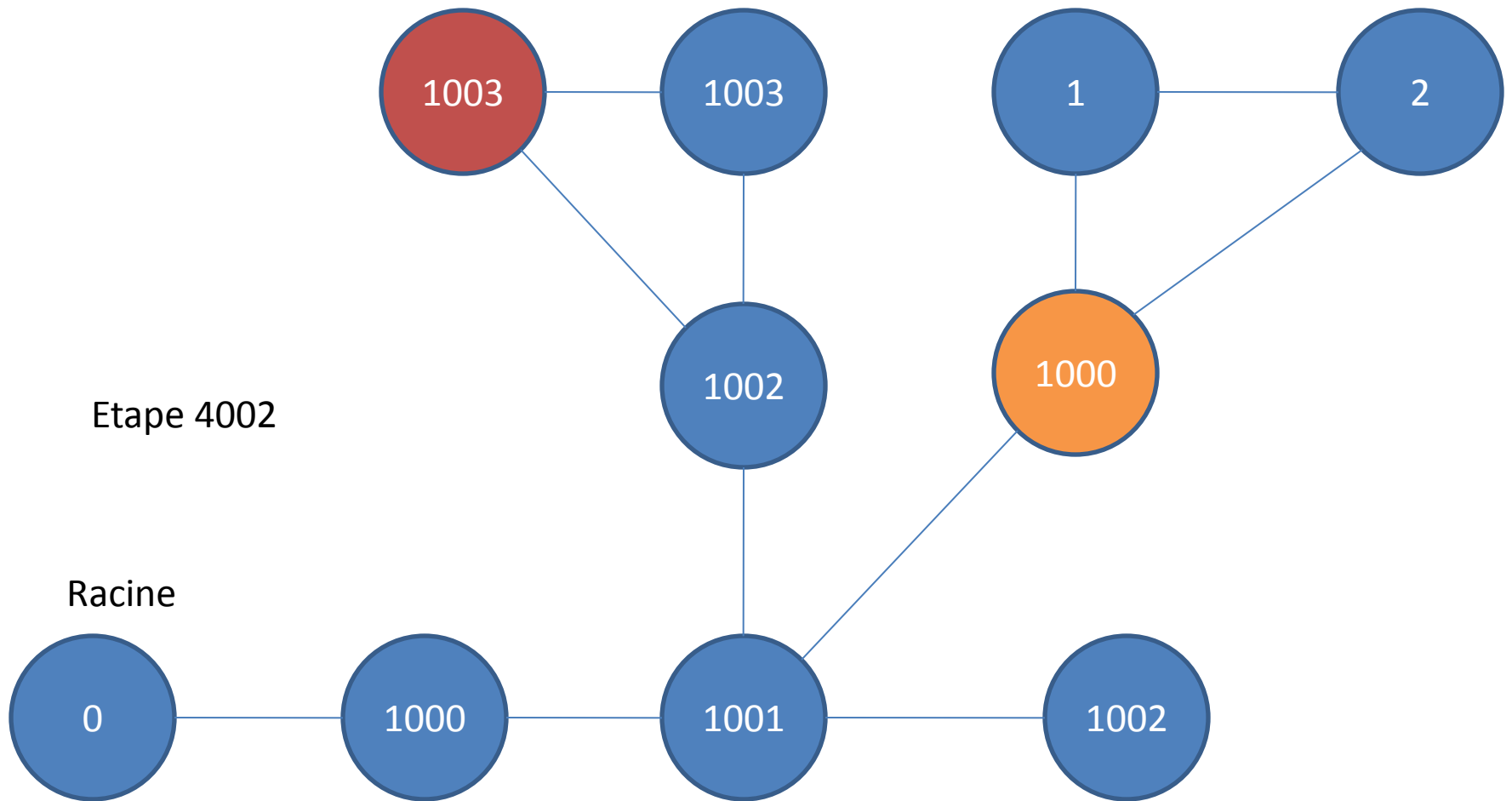
Un exemple : L'algorithme du Min+1



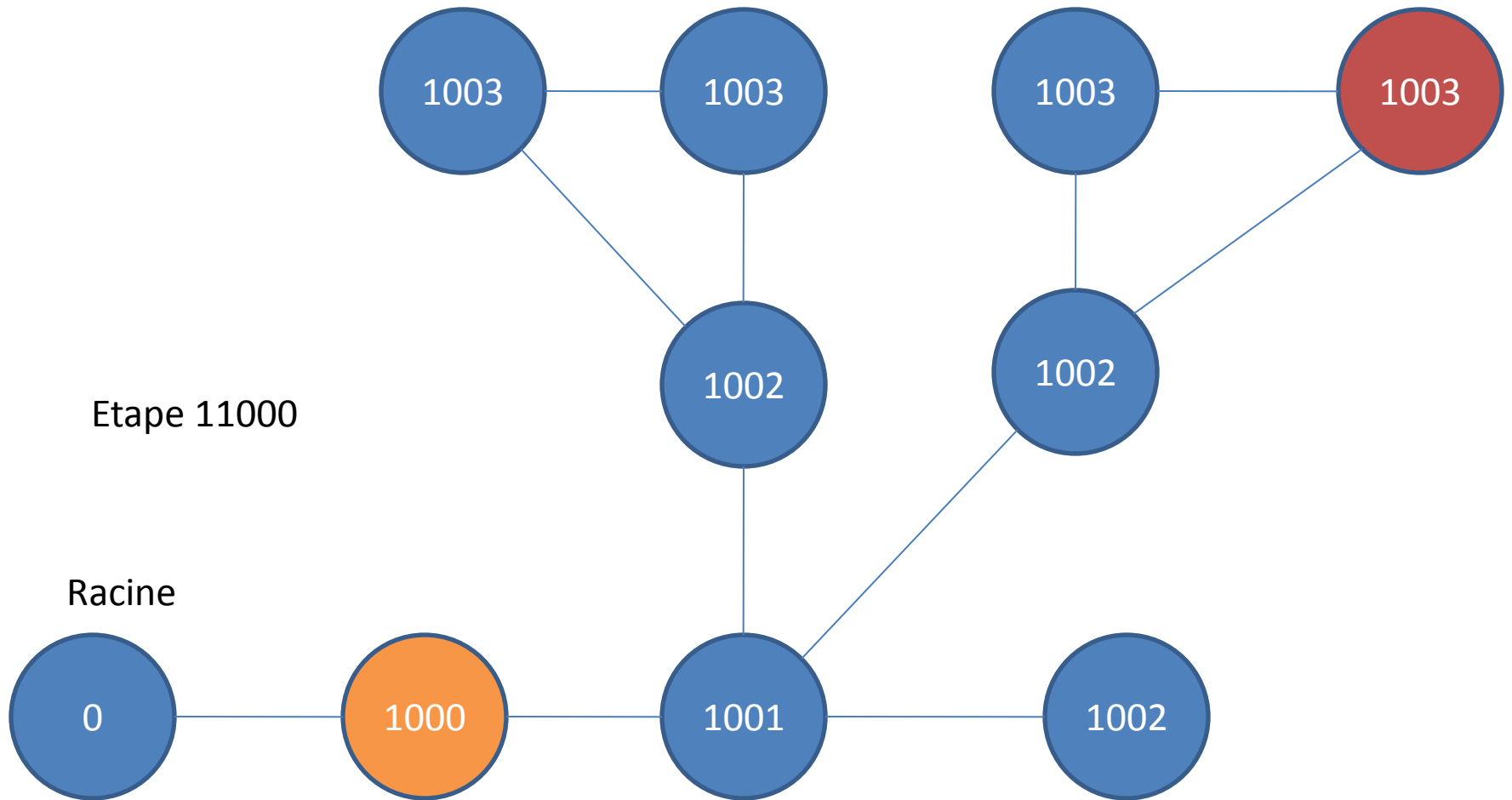
Un exemple : L'algorithme du Min+1



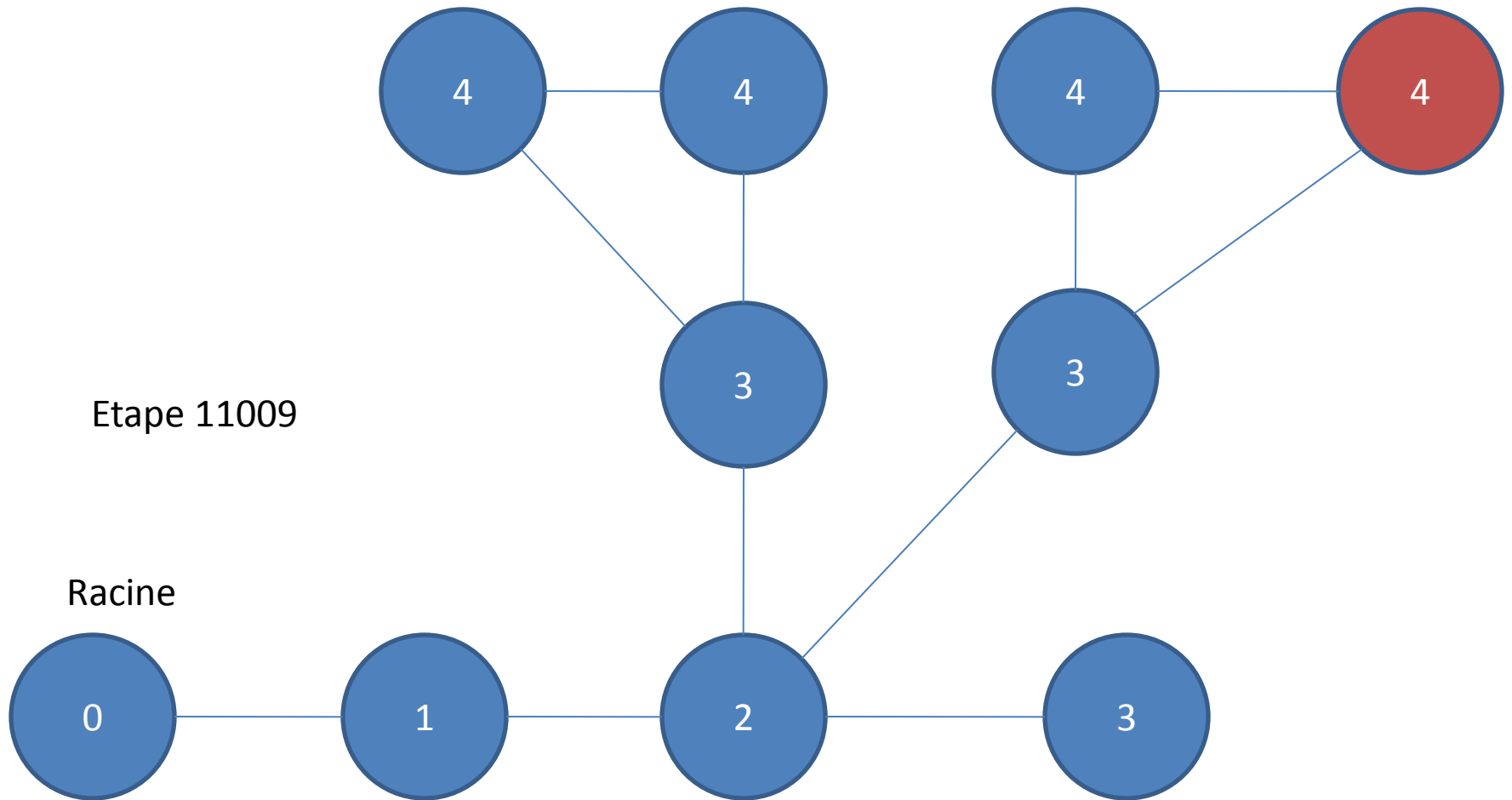
Un exemple : L'algorithme du Min+1



Un exemple : L'algorithme du Min+1



Un exemple : L'algorithme du Min+1



Un exemple : L'algorithme du Min+1

- Complexité en temps de cet algorithme :
 - $\theta(d)$ rondes;
 - $\theta(\max n^2)$ pas de calcul.

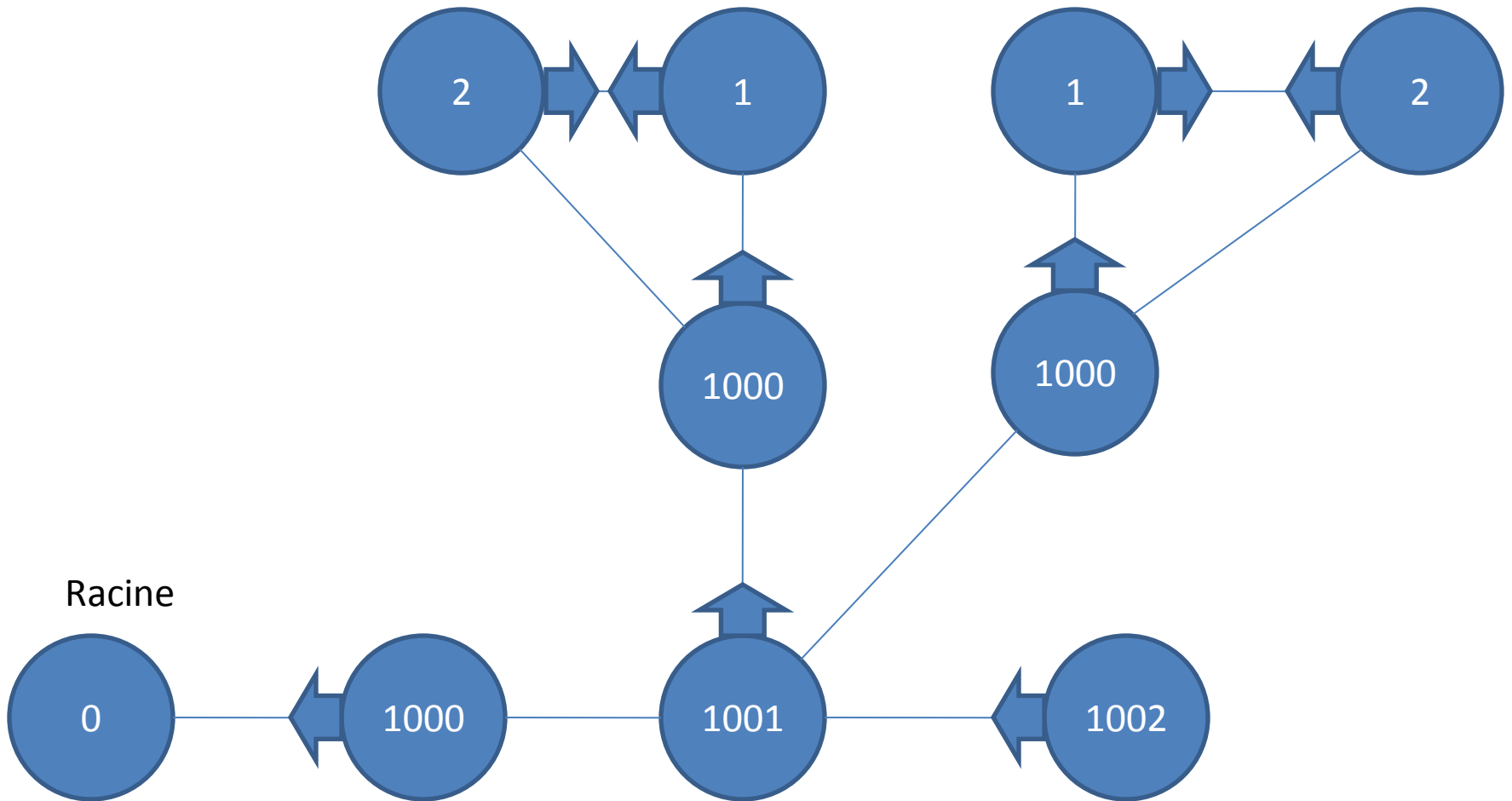
L'algorithme de Kosowski kuszner

- Cet algorithme utilise l'opérateur Max + 1.
 - $\theta(n)$ rondes;
 - $\theta(n^2 \text{ Diam})$ pas de calcul.

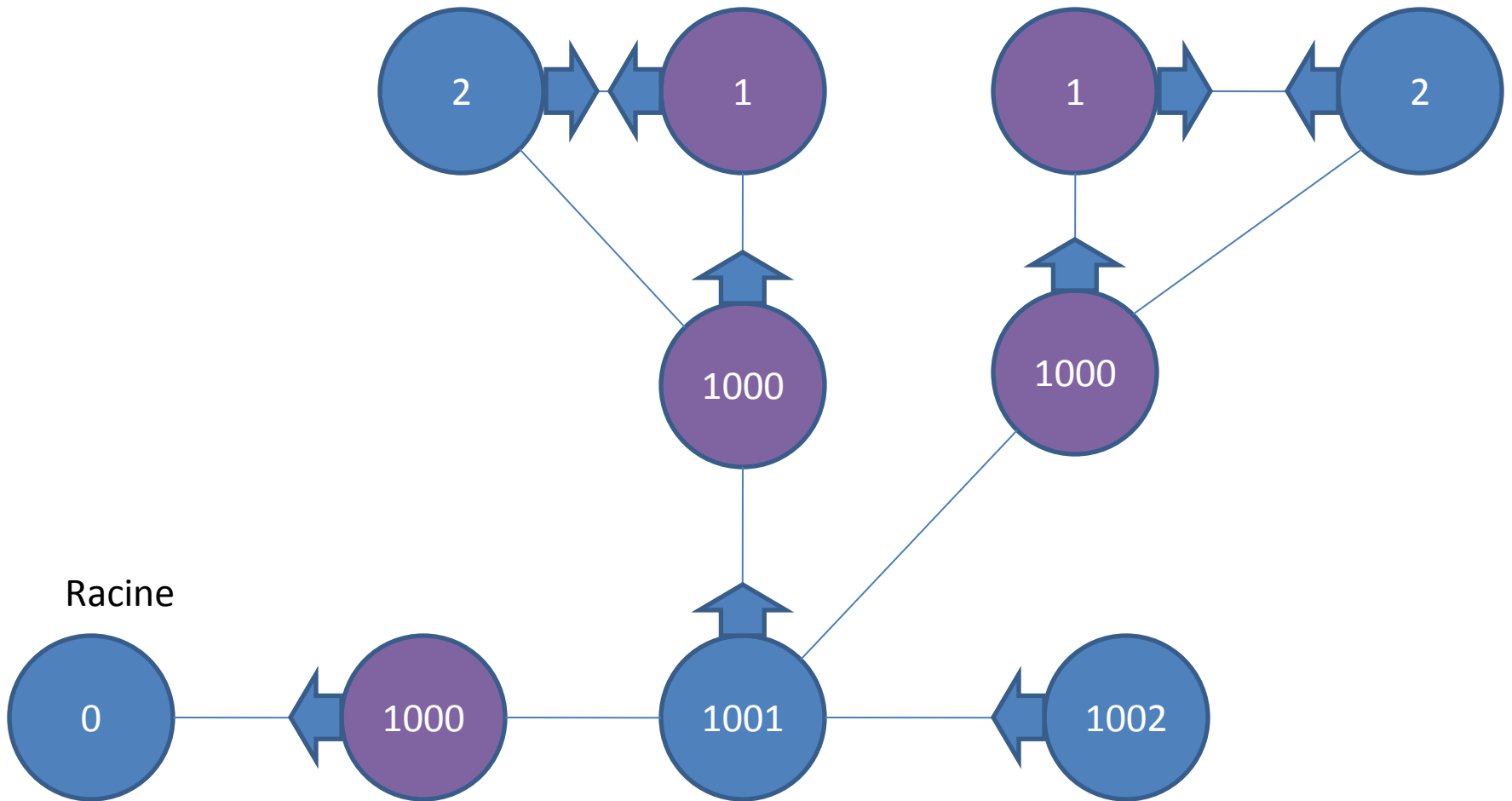
Un autre algorithme

- Le but de cet algorithme est la construction d'un arbre couvrant. Il utilise 3 variables
 - L : un entier naturel donne la longueur du chemin du nœud à la racine de l'arbre.
 - Par : Un pointeur qui détermine le père du nœud dans l'arbre.
 - EA : une variable pouvant prendre 4 valeurs C, EB, EF, WT. Elles sont utiles pour l'administration des erreurs.

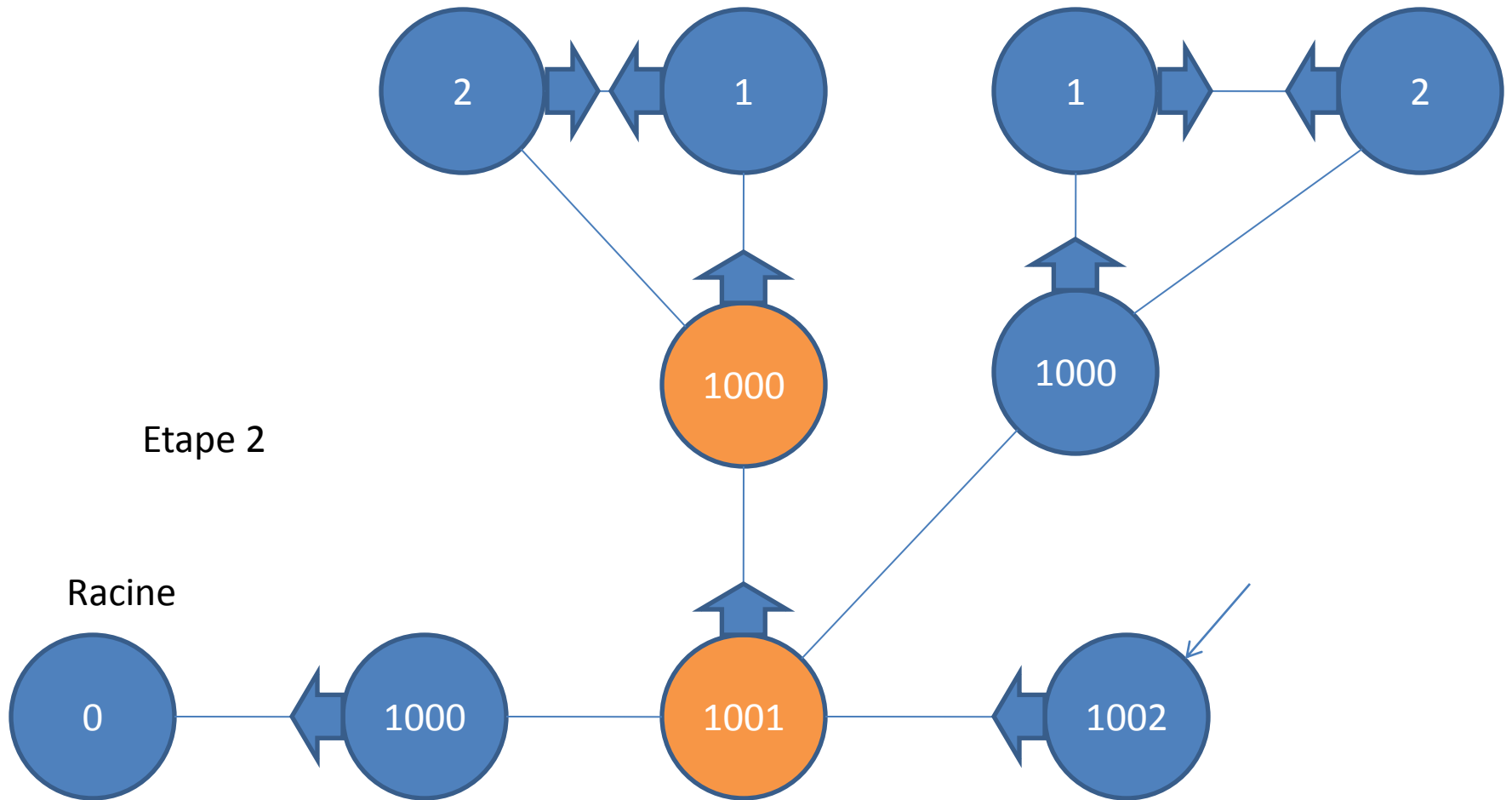
Un Exemple :



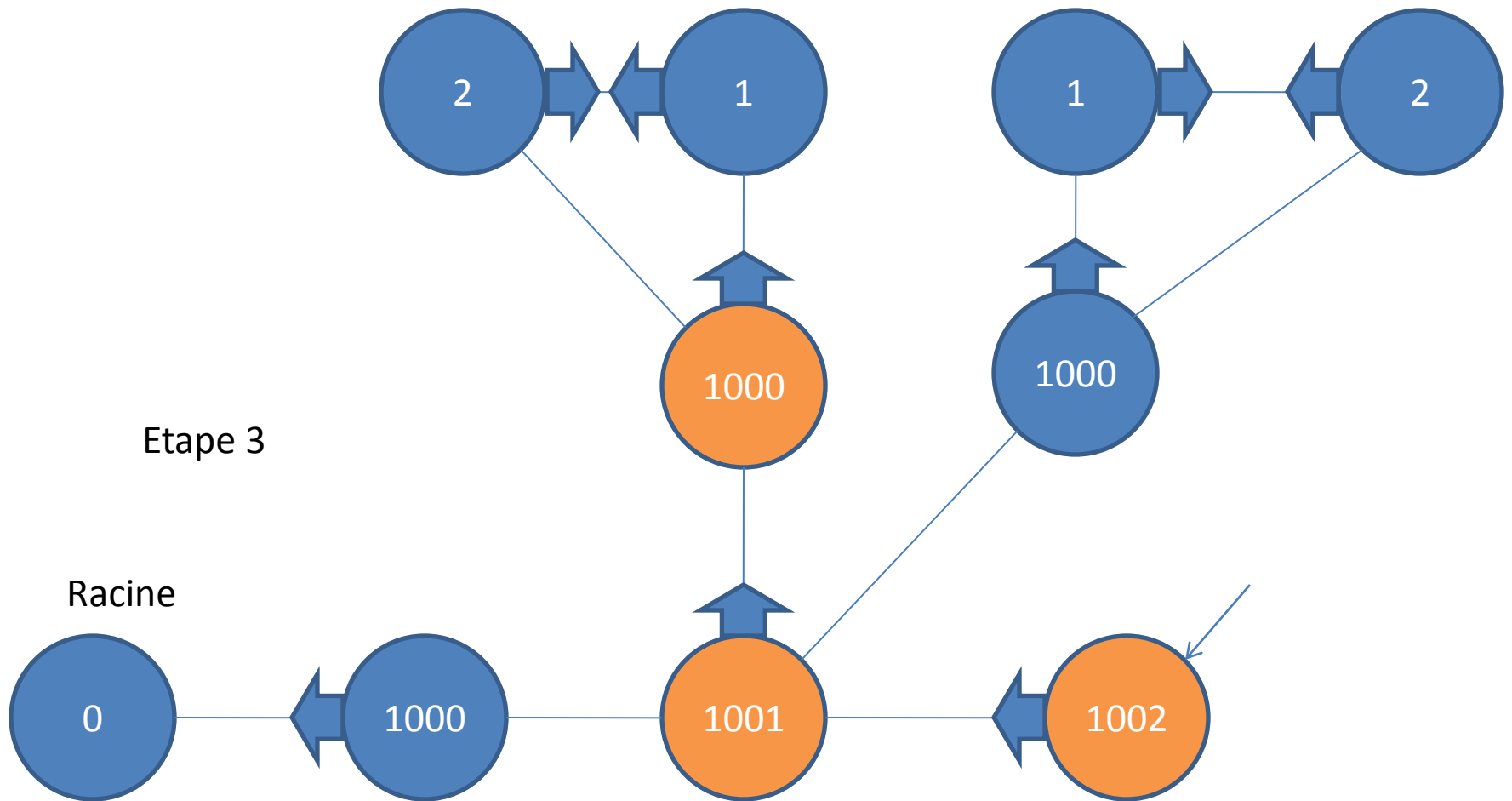
Un Exemple :



Un Exemple :



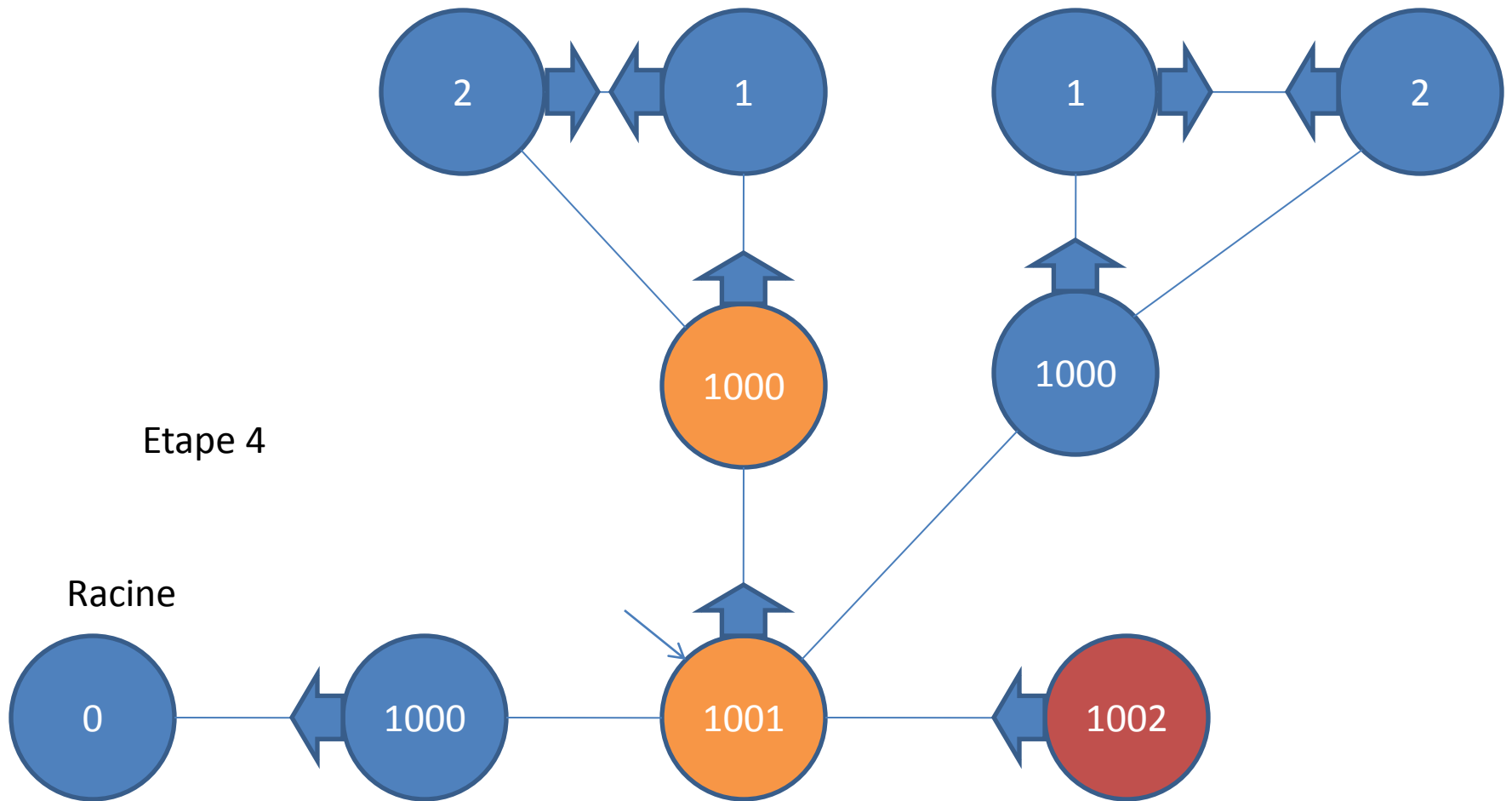
Un Exemple :



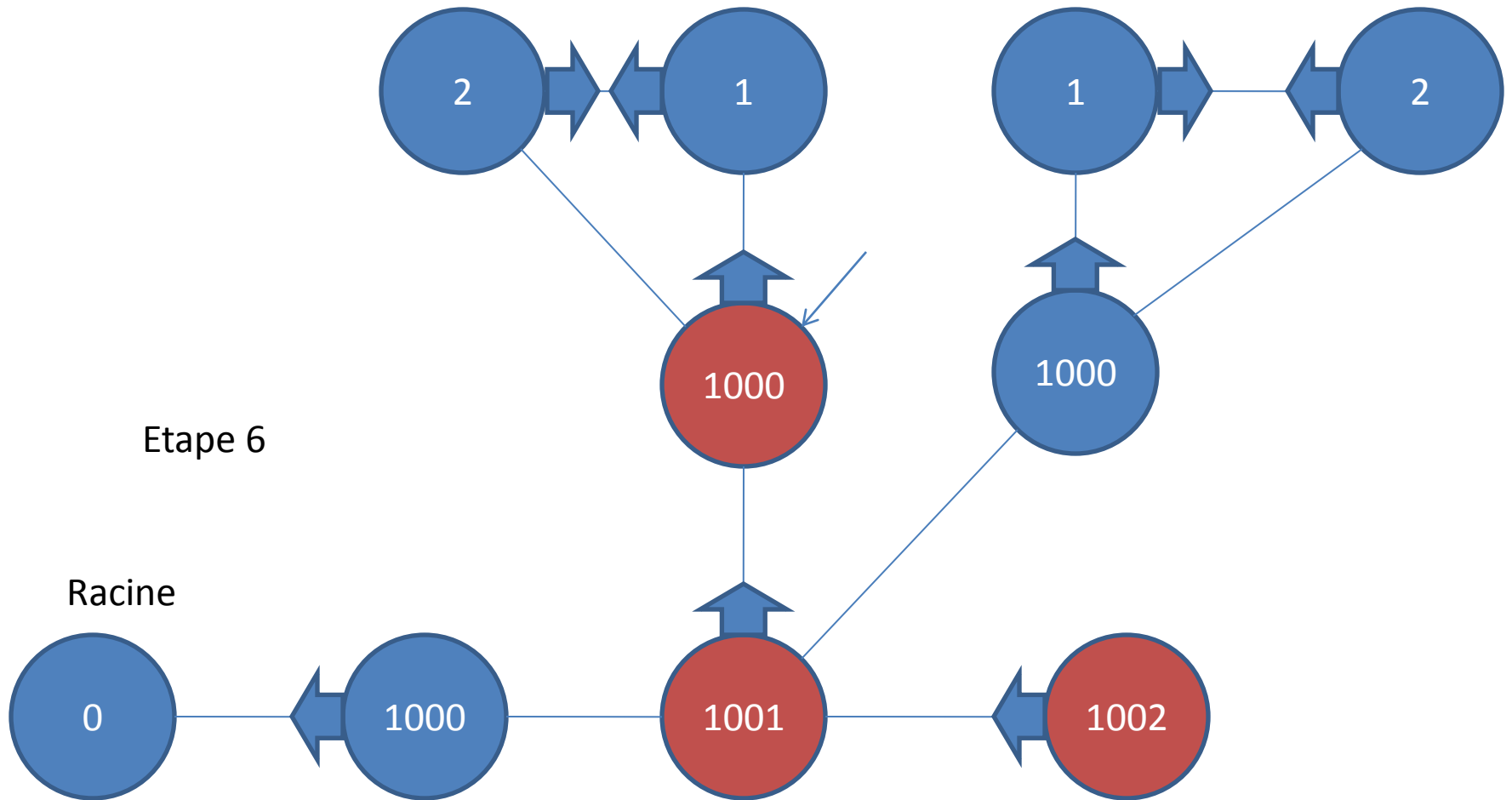
Etape 3

Racine

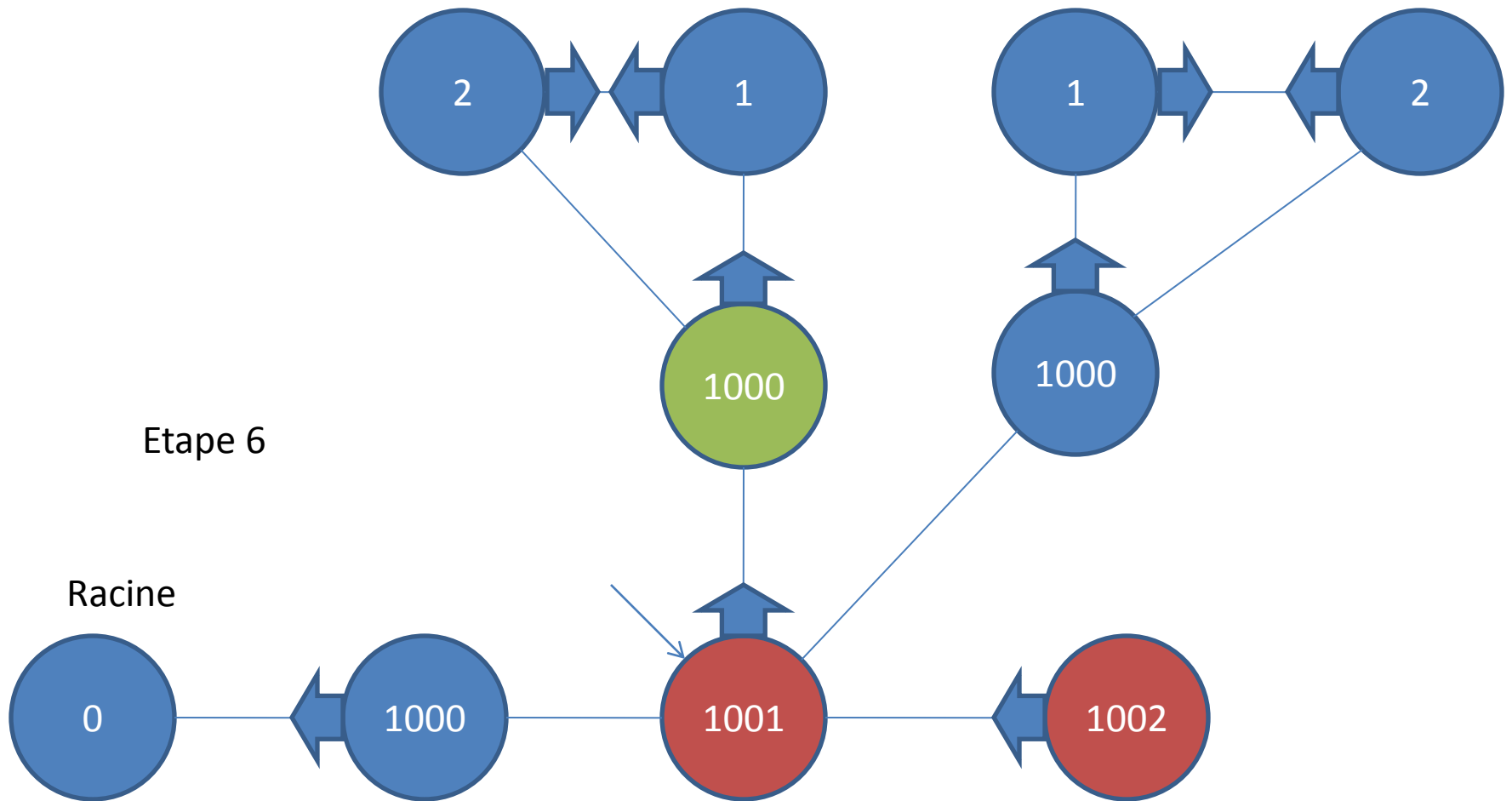
Un Exemple :



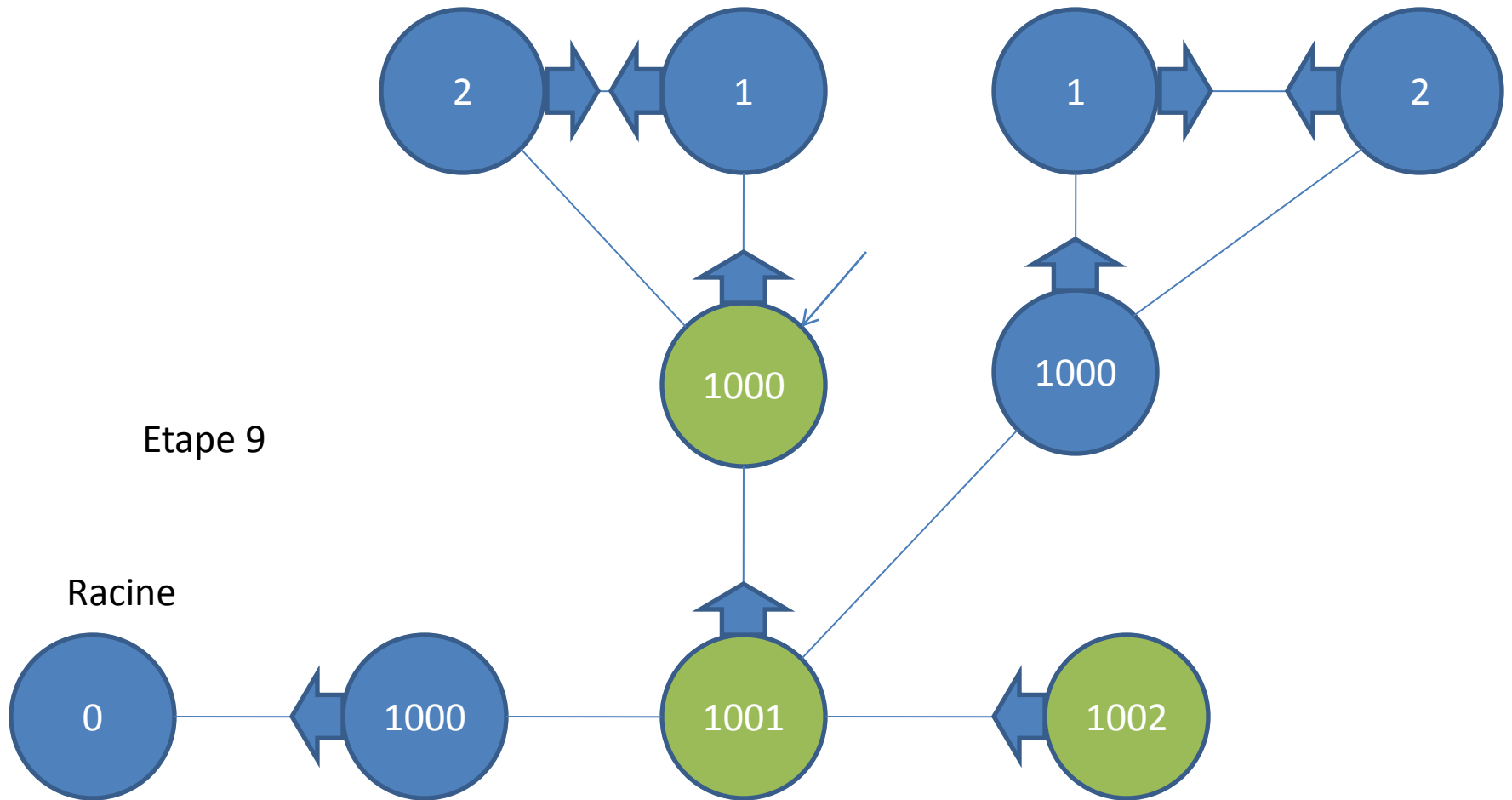
Un Exemple :



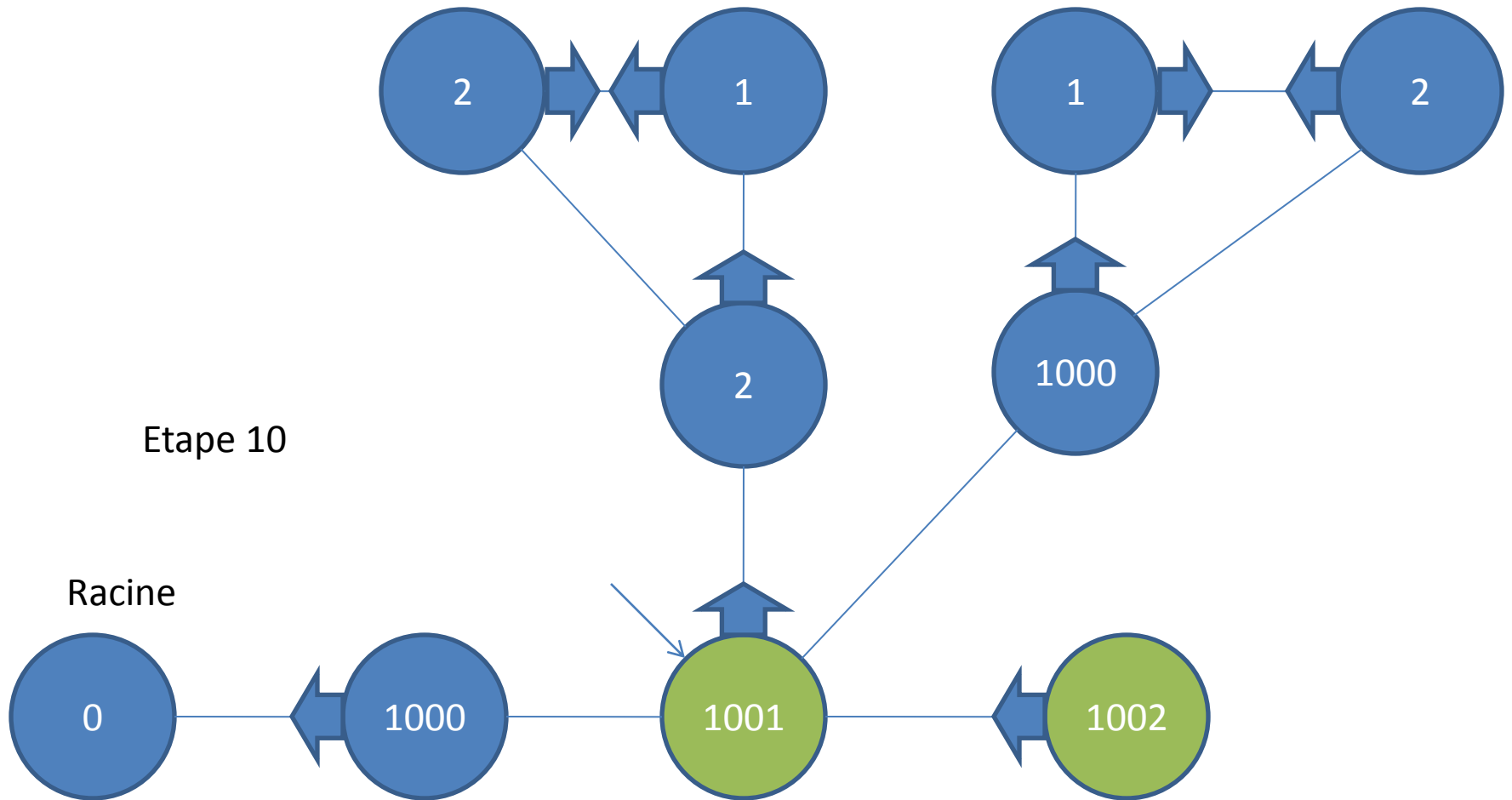
Un Exemple :



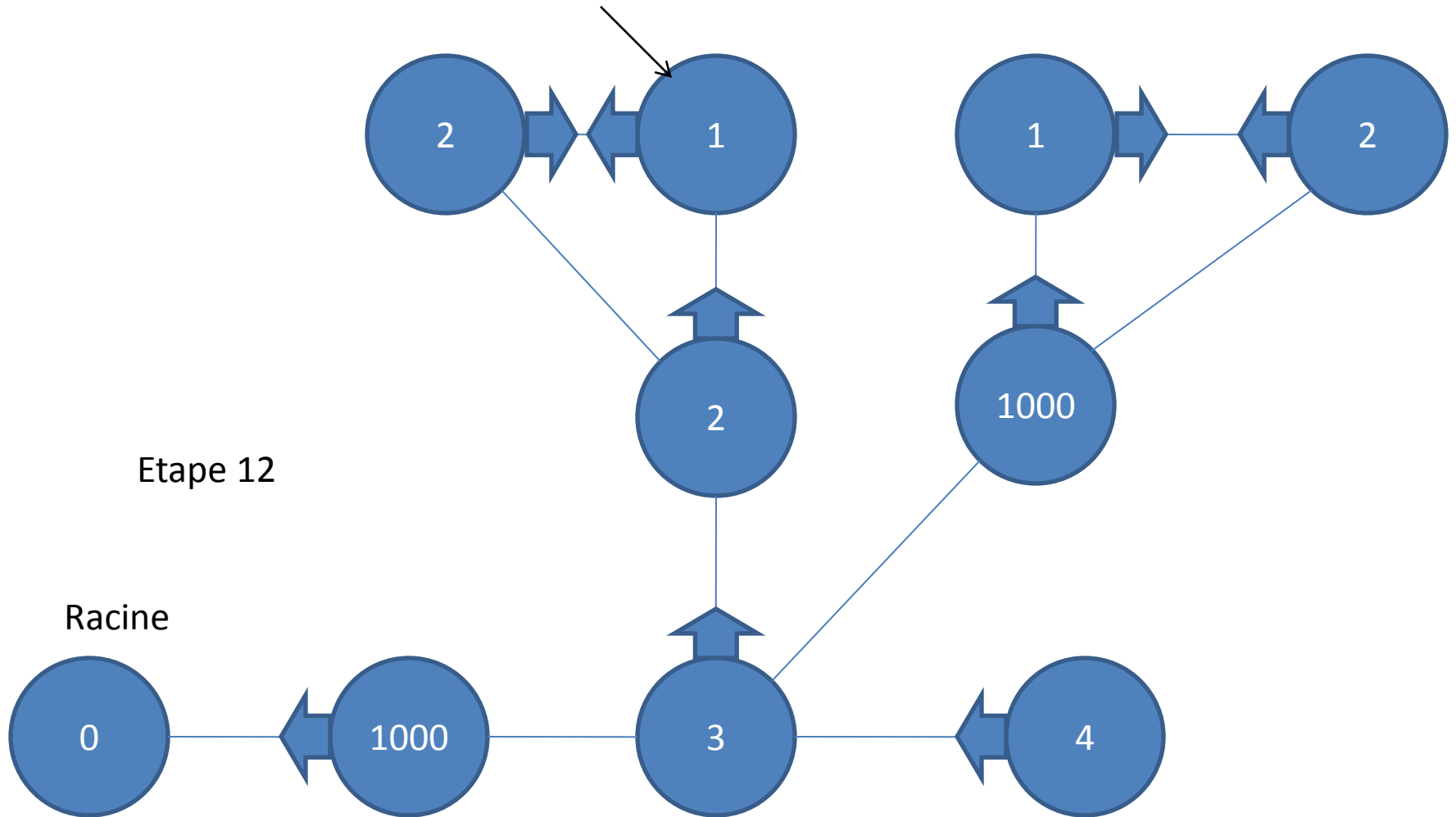
Un Exemple :



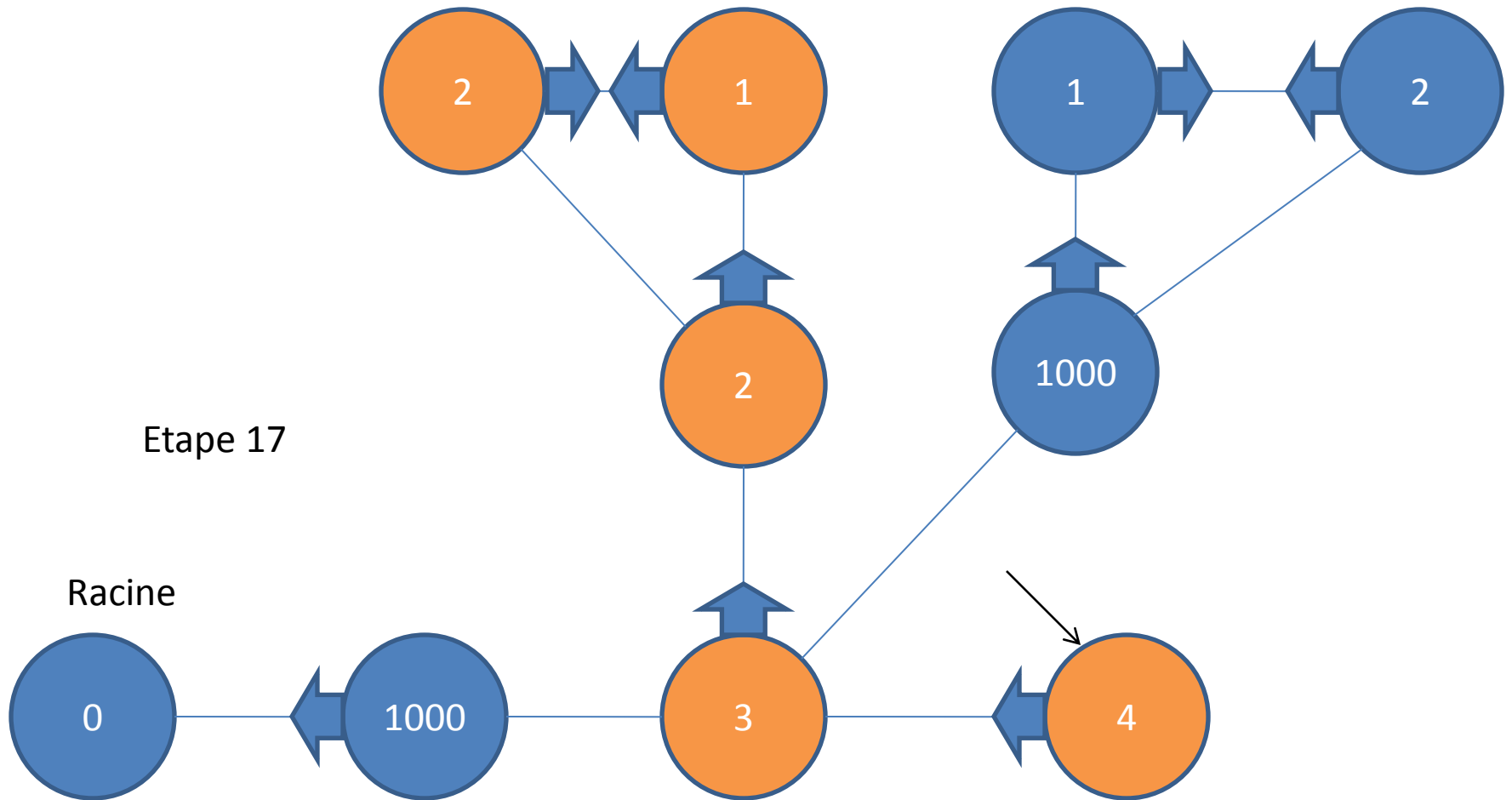
Un Exemple :



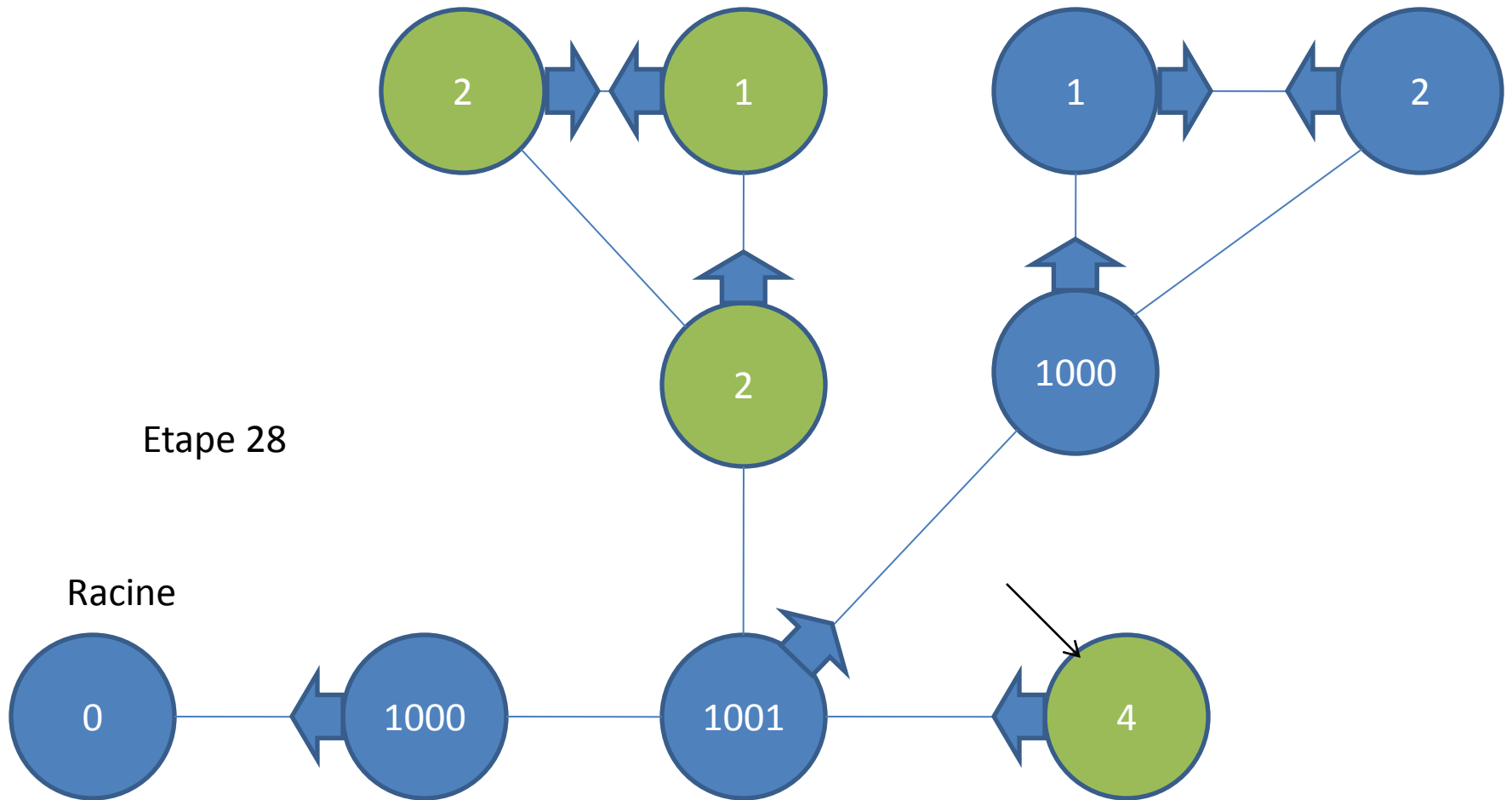
Un Exemple :



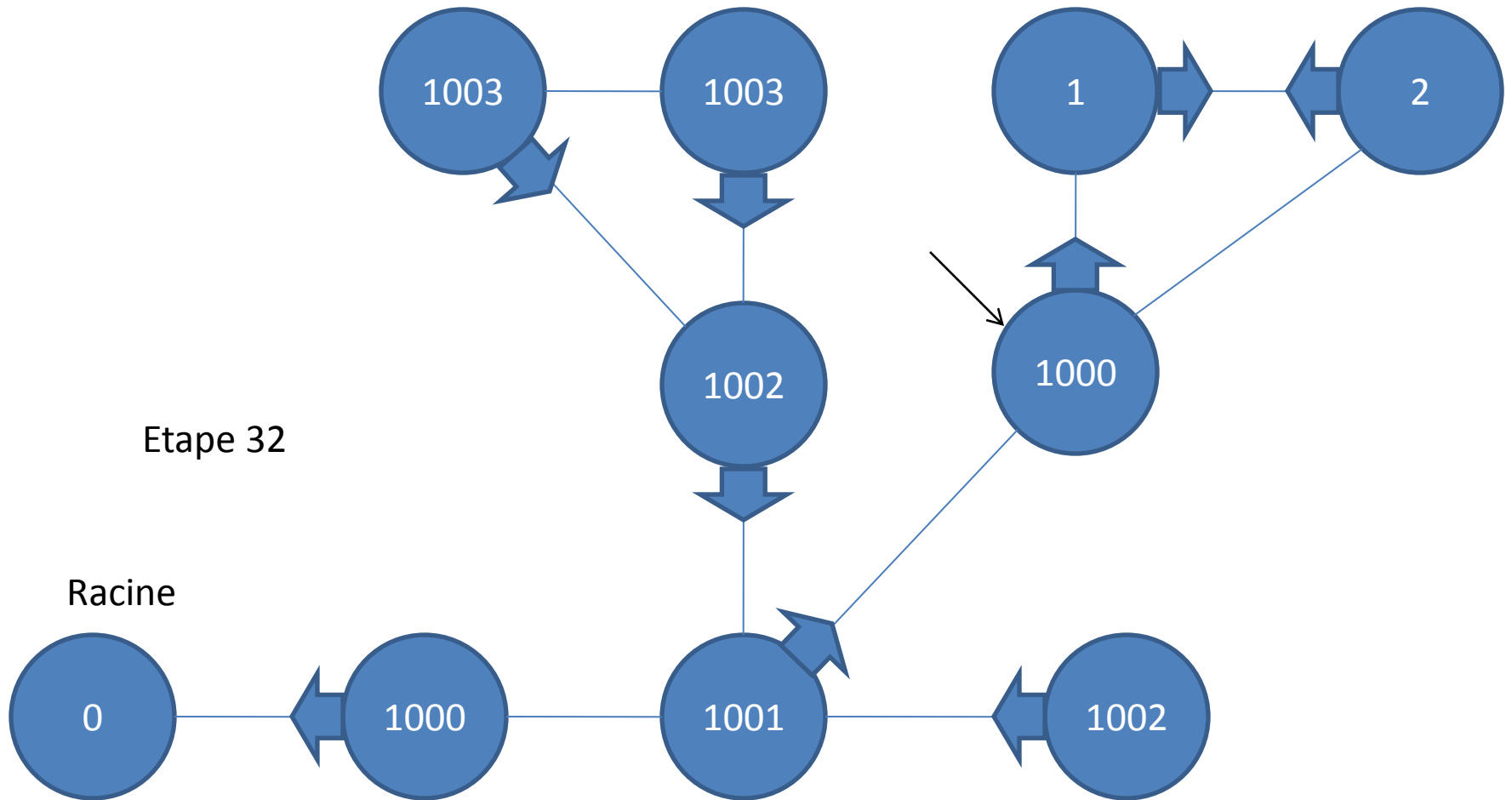
Un Exemple :



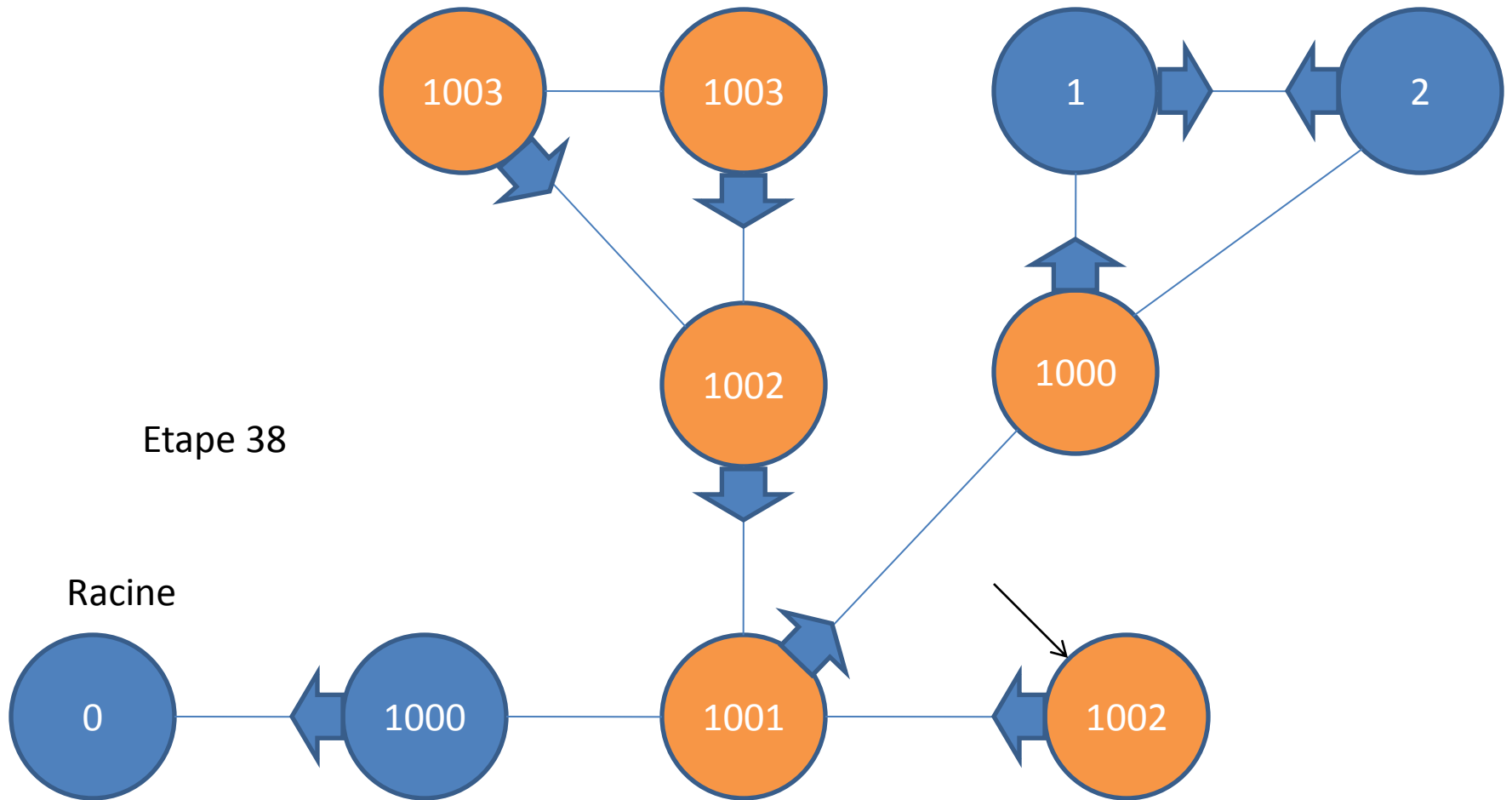
Un Exemple :



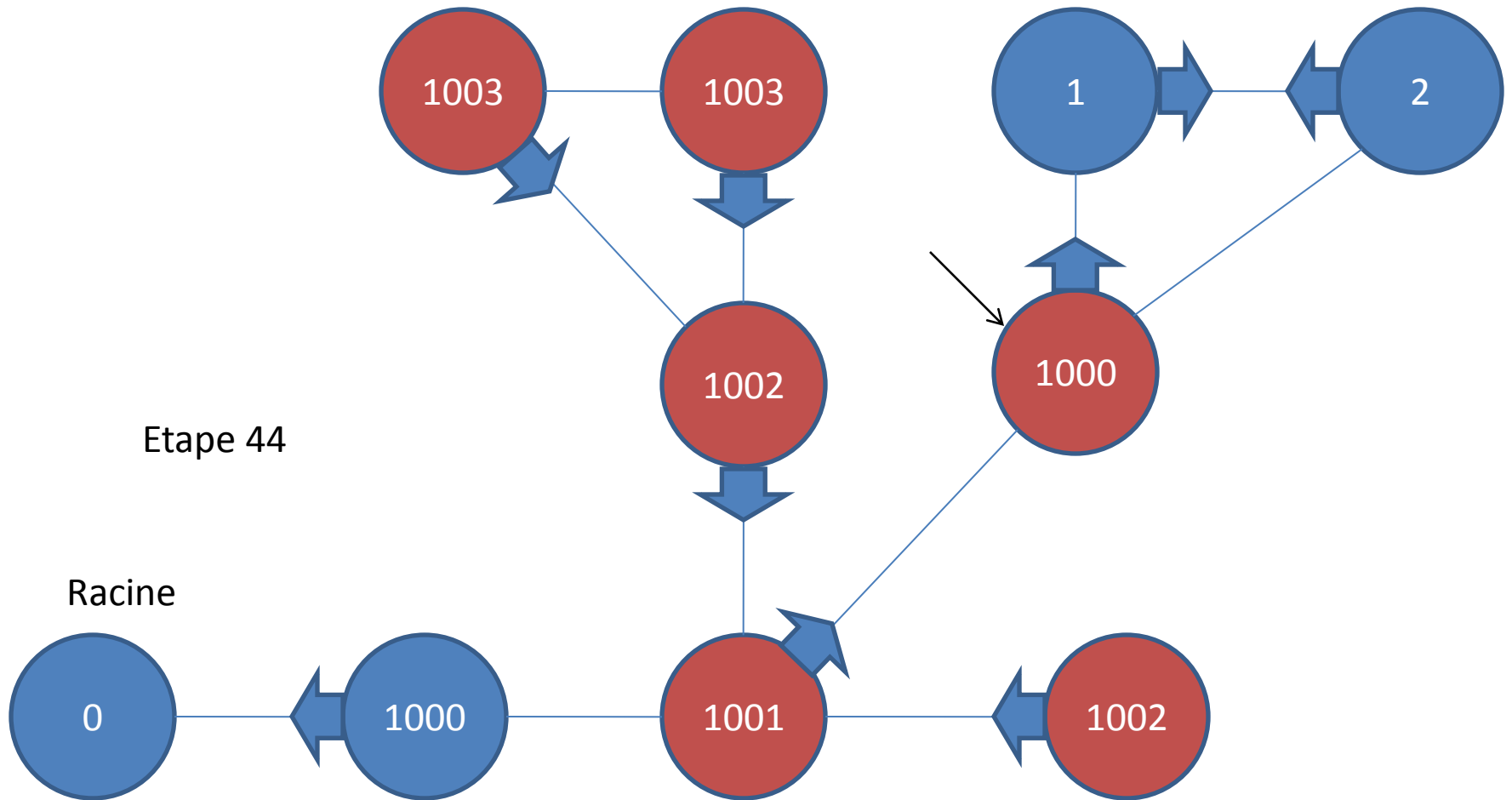
Un Exemple :



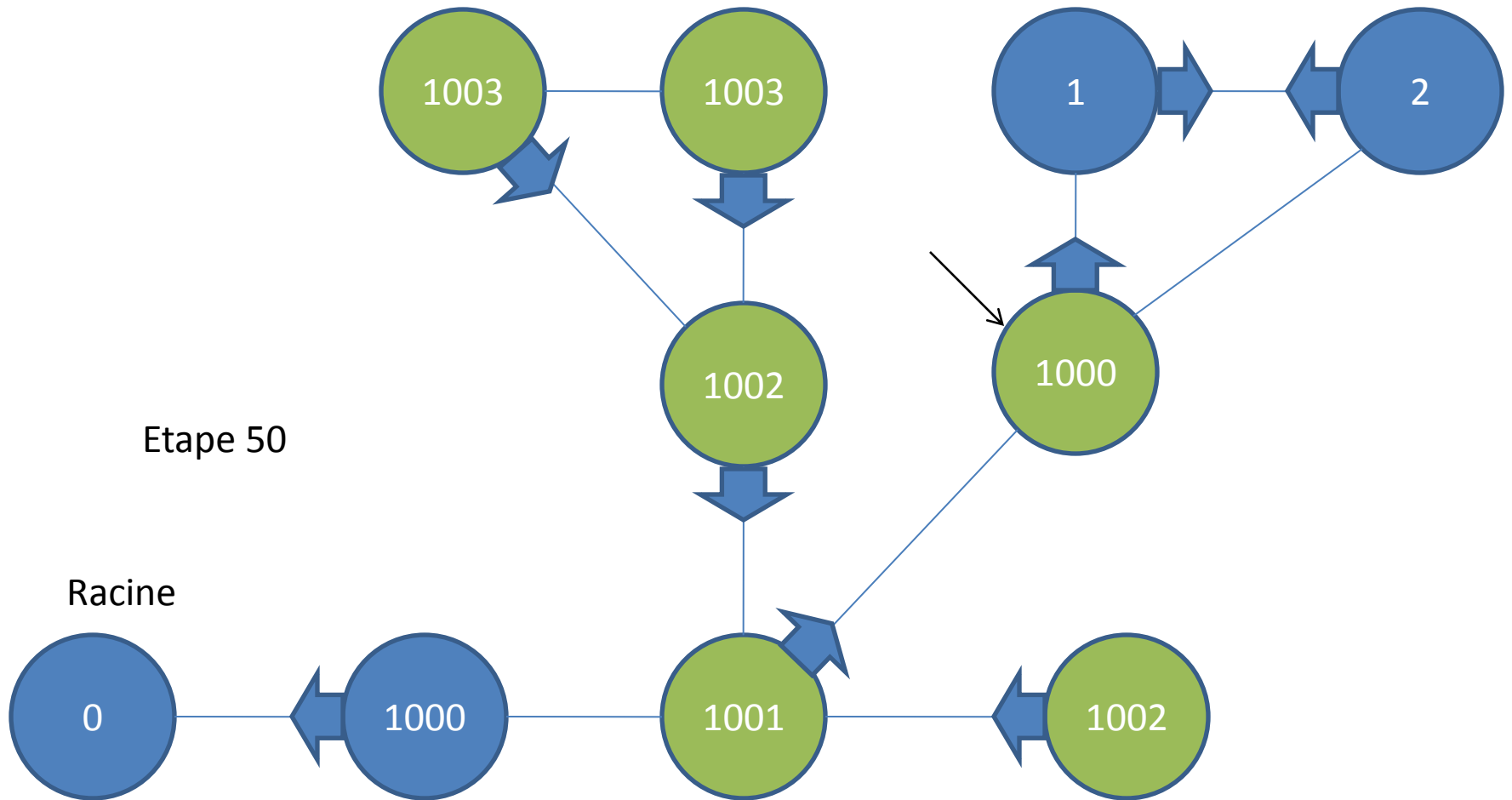
Un Exemple :



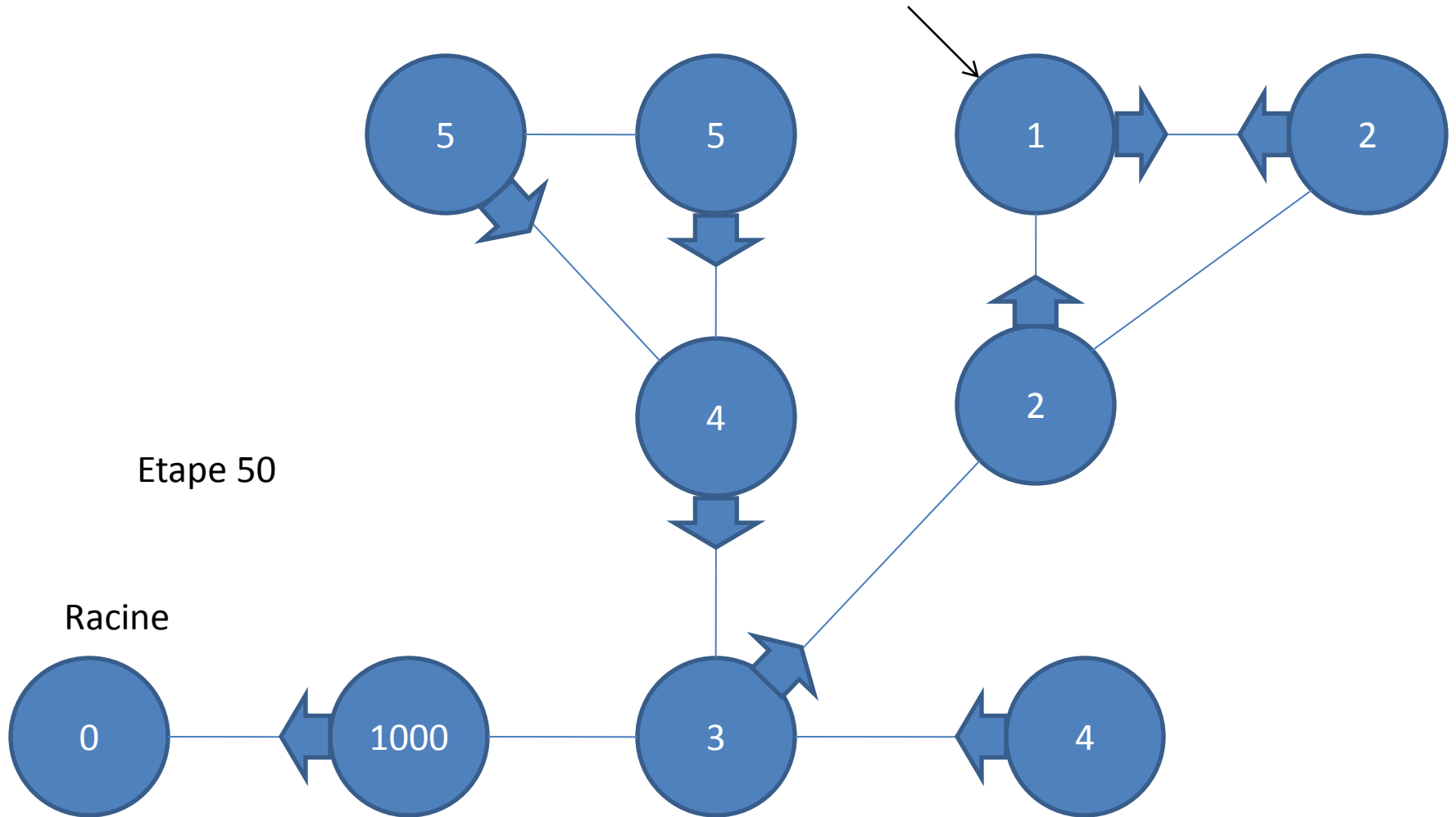
Un Exemple :



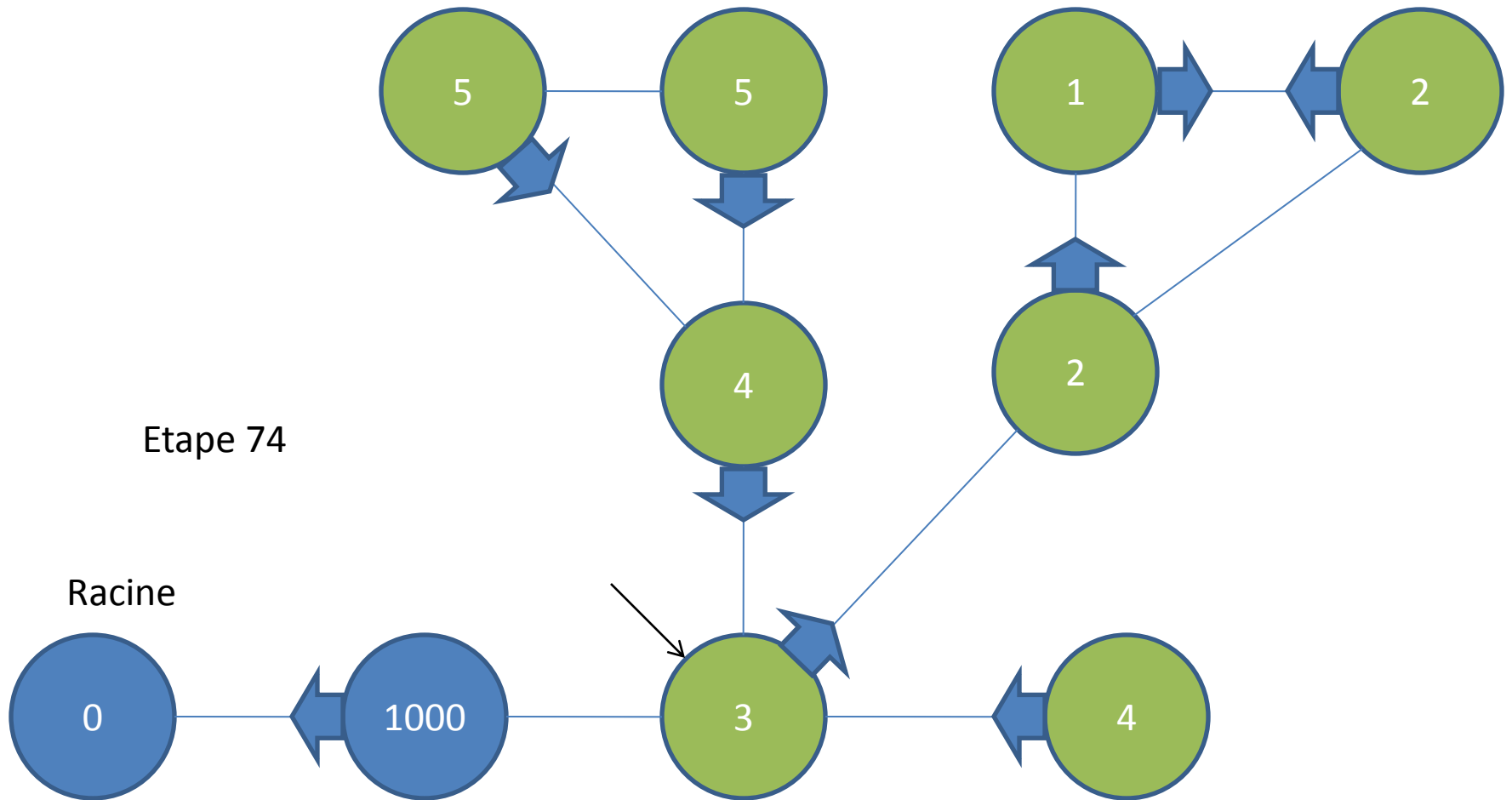
Un Exemple :



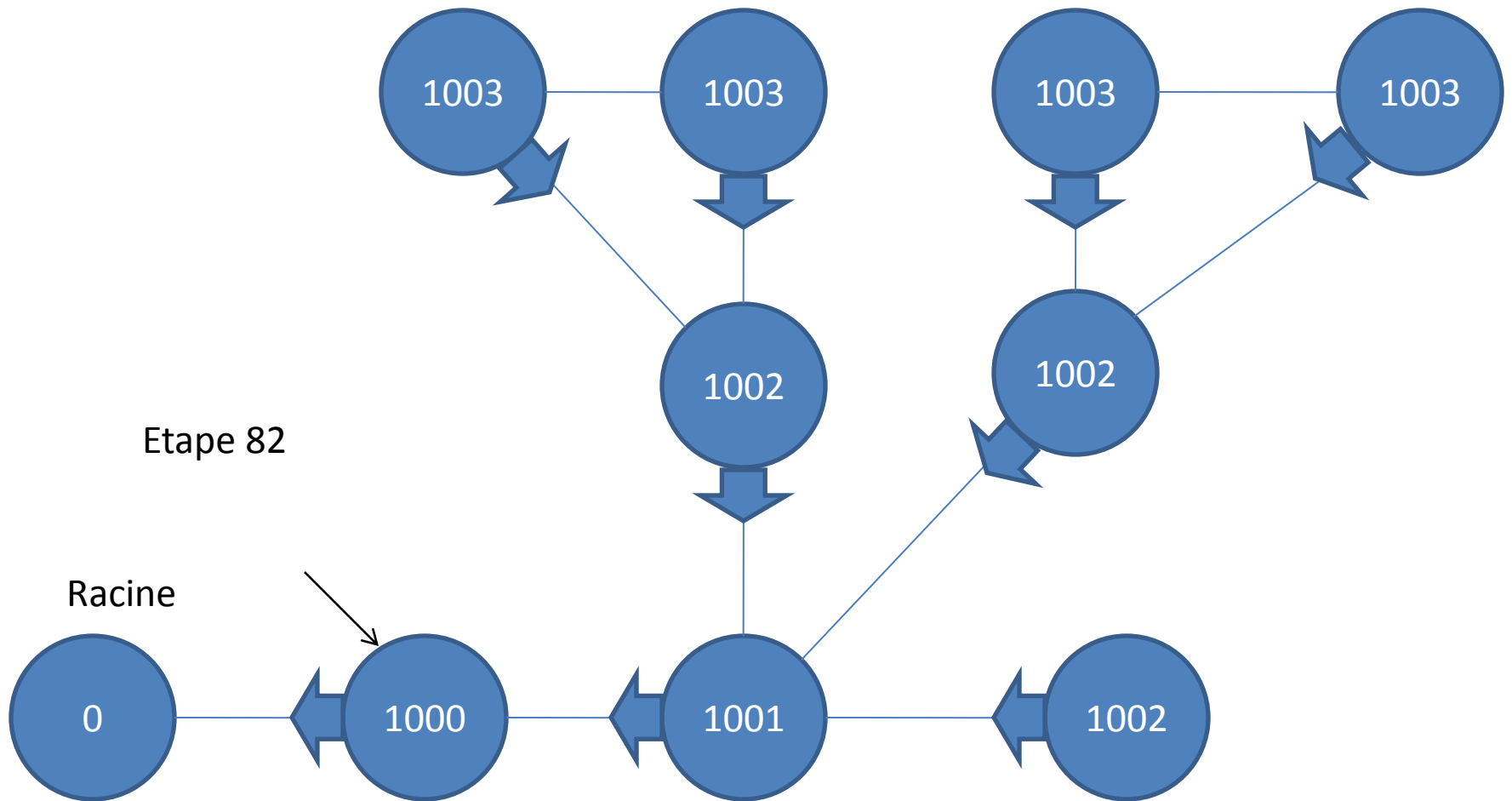
Un Exemple :



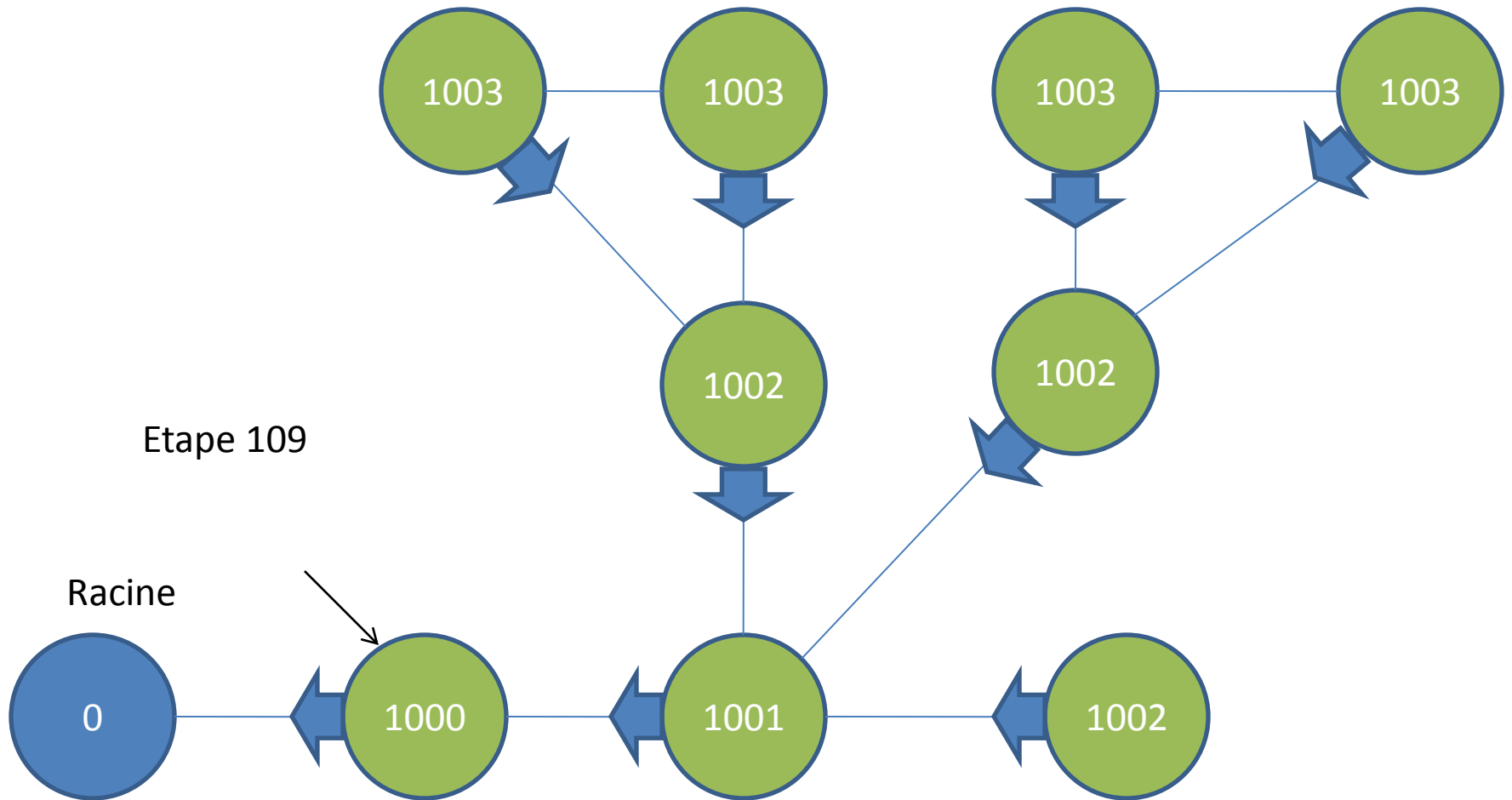
Un Exemple :



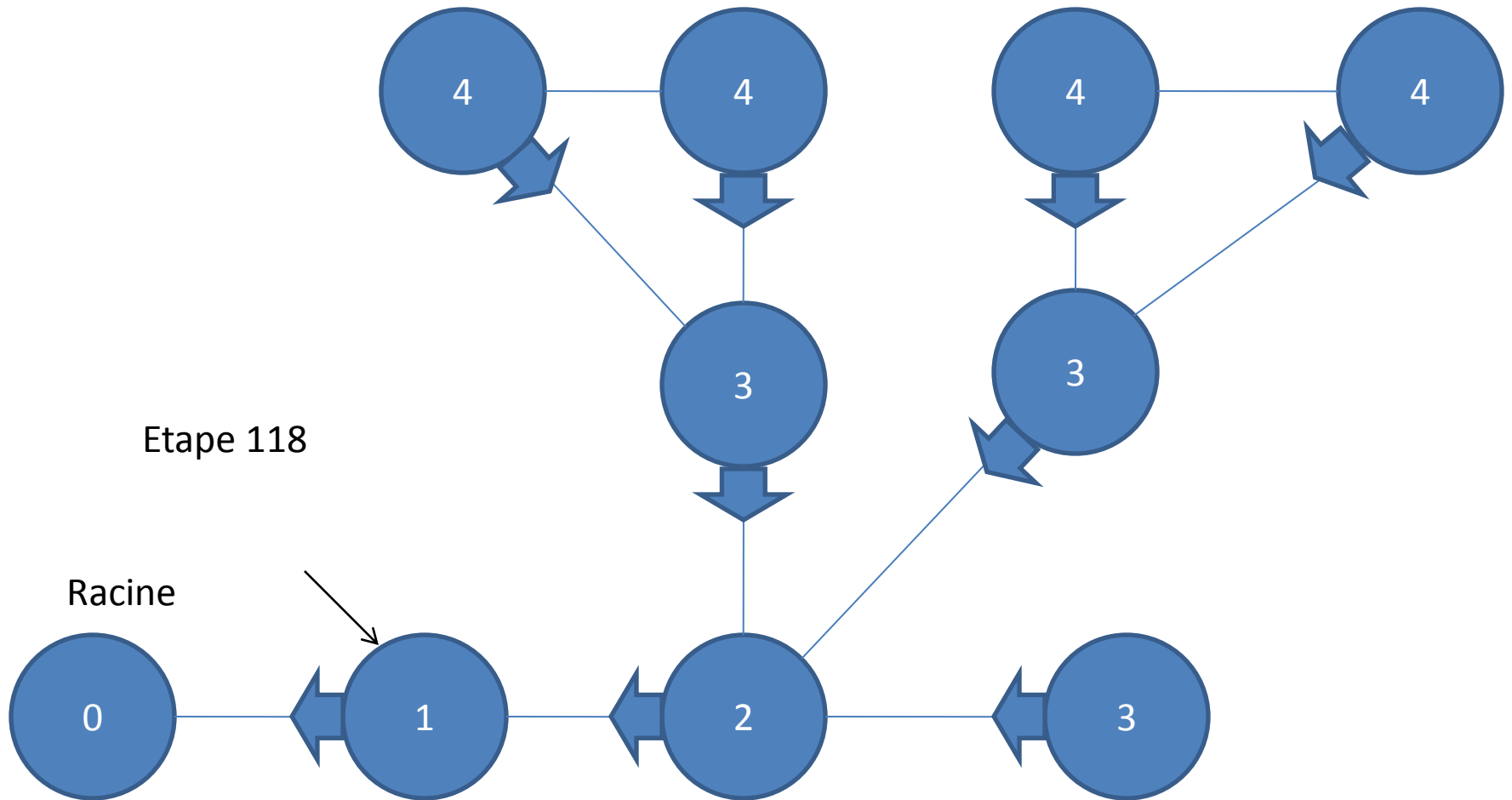
Un Exemple :



Un Exemple :



Un Exemple :



Faut-il seulement compter les rondes ?

Le cas de l'arbre couvrant

- Cet algorithme a pour complexité :
 - $\theta(n)$ rondes;
 - $\theta(n^2)$ étapes, dans le pire des cas.
- Il stabilise même lorsque le réseau n'est pas connexe.