Un rappel, un défi

on a vu que

G : $\{ S \rightarrow a\,S\,b + \varepsilon$

engendre le langage $L = \{a^n b^n, n \geq 0\}$

Je vous affirme que $L' = \{a^n b^p, n \neq p\}$ est également algébrique

- **Quiz** - trouver une grammaire qui engendre L' est ~~super facile~~ assez difficile

Comme toujours en informatique, lorsqu'on ne sait pas résoudre un problème, on essaie de le ramener à des problèmes plus simples.

$n \neq p$ c'est $n > p$ ou $n < p$

$$G \left\{ \begin{array}{l} S \rightarrow a \;\; S \;\; b \;\; + T \\ T \rightarrow T \;\; b + b \end{array} \right.$$

- **Quiz** - cette grammaire traite le cas $\{a^n\, b^p,\ n>p\}$ $\{a^n\, b^p,\ n<p\}$

Clôture des algébriques par union

- Si L et M sont algébriques, alors $L \cup M$ est algébrique.
- Exemple : soit deux grammaires engendrant respect.^{t} L_1 et L_2

$$G \begin{cases} S \to a \;\; S \;\; b \;\; + T \\ T \to T \;\; b + b \end{cases}$$

$$G' \begin{cases} S \to a \;\; S \;\; b \;\; + T \\ T \to a\,T + a \end{cases}$$

- on construit une grammaire engendrant $L_1 \cup L_2$, ce qui prouve que le langage $L_1 \cup L_2 = \{a^n\, b^p,\ n \neq p\}$ est également algébrique.

Clôture des algébriques par union (cas général)

- Soient L et M deux langages algébriques, la seule façon *pour l'instant* de montrer que $L \cup M$ est algébrique est de trouver une grammaire qui l'engendre.

  - on construit une grammaire engendrant $L \cup M$ à partir de deux grammaires engendrant respectivement L et M,

  - il s'agit bien entendu de donner un modèle général qui convienne quels que soient les langages algébriques L et M considérés.

- Soient L et M deux langages engendrés respectivement par $<X,V, S, P>$ et $<X,V', S', P'>$, où V et V' sont disjoints, soit S'' un nouveau non-terminal, on peut montrer que la grammaire $<X,V\cup V'\cup\{S''\}, S'', P \cup P'\cup\{S''\rightarrow S , S''\rightarrow S'\}>$ engendre $L \cup M$.

Clôtures des algébriques

- Alg(X*) et Alg sont closes par

  - union

  - produit (cf TD)

  - étoile (cf TD)

- Alg n'est pas close par

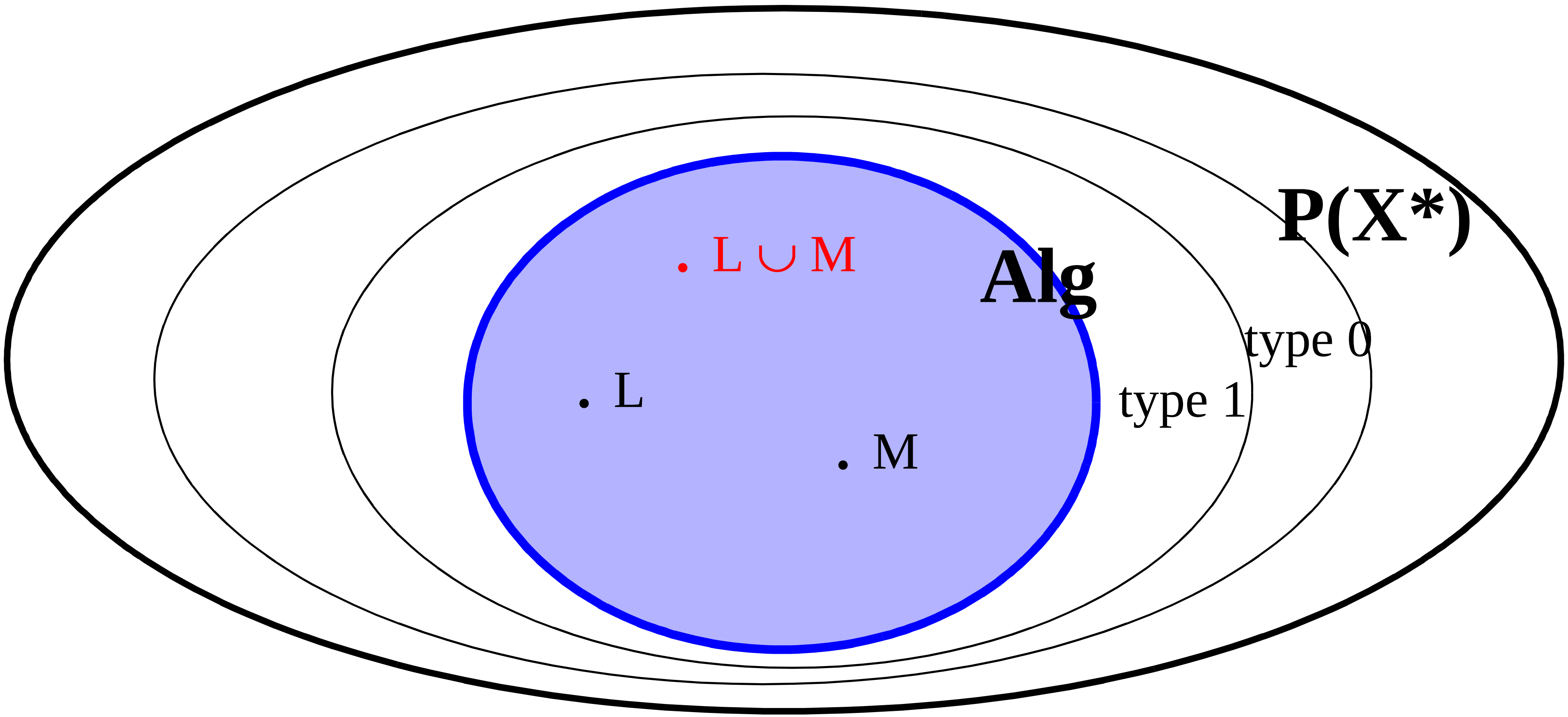
  - intersection (cf cours)

  - complémentation

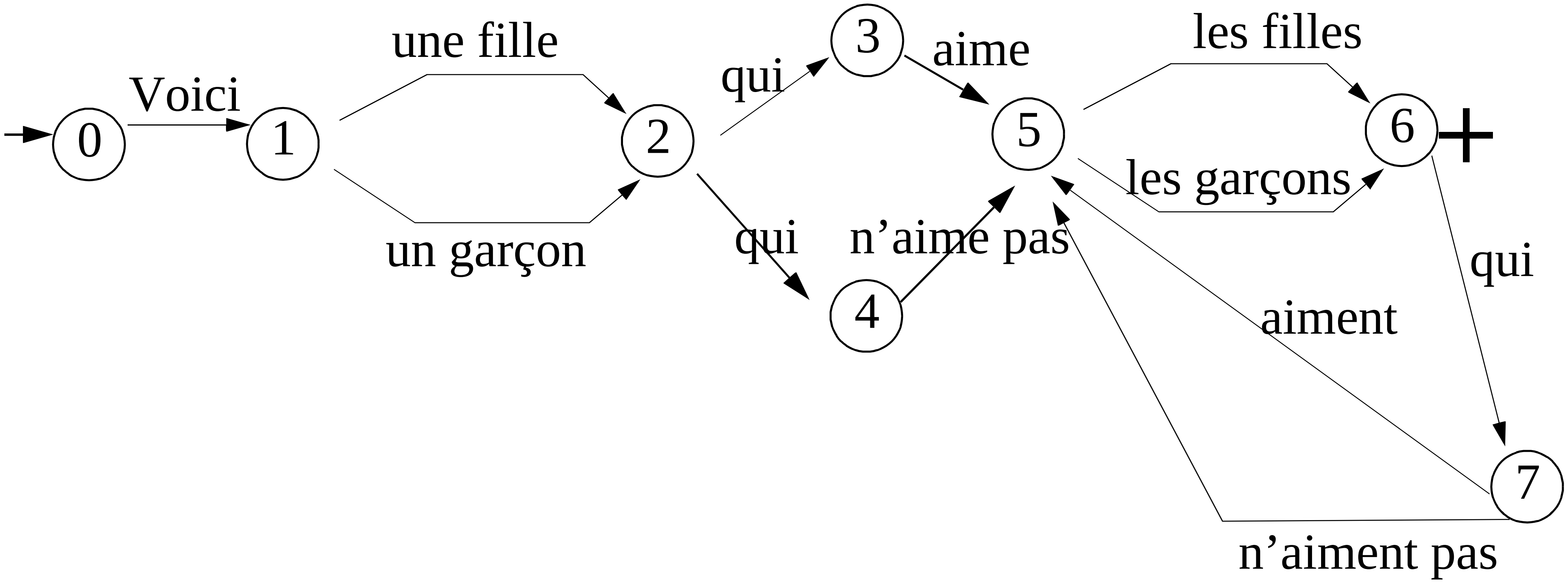
Ce qui signifie ...

Quiz - sur ce schéma, $L \cap M$ se trouve

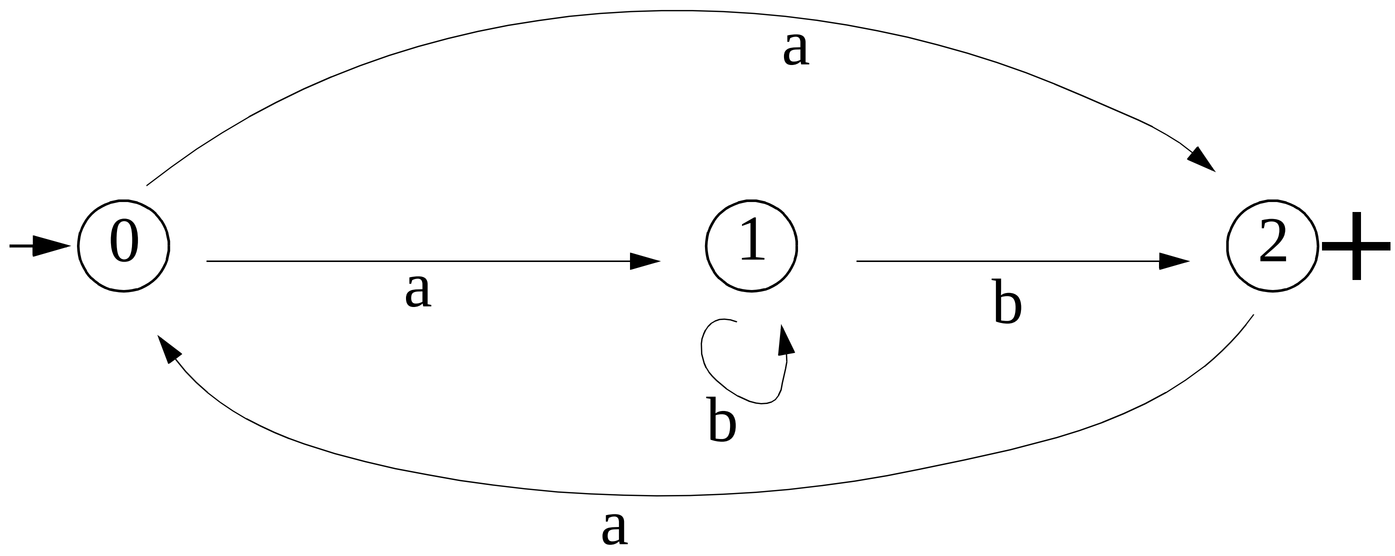
toujours dans le bleu
ça dépend des cas
toujours dans le blanc



Exemple d'automate fini



Exemple d'automate fini



Automate fini : définitions

- Un automate fini est un quintuplet $\mathcal{A}$=< X , Q , q_o , F, δ > où :

  - X est un alphabet, dit alphabet d'entrée

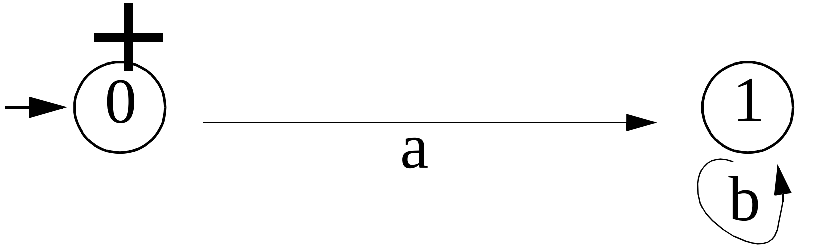
  - Q est un ensemble fini, appelé ensemble des états de l'automate

  - $q_0 \in Q$ est l'état initial

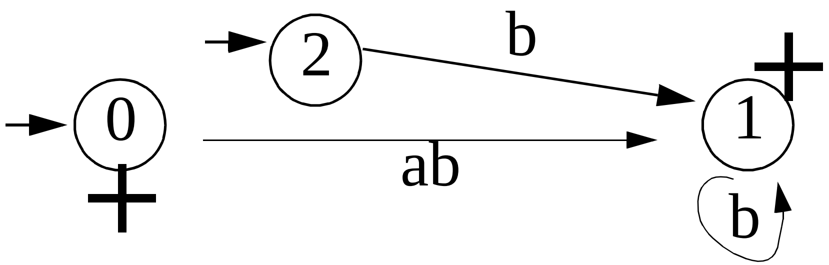
  - $F \subseteq Q$ est l'ensemble des états d'acceptation (états **finals** cf Larousse)

  - $\delta \subseteq Q \times X \times Q$ est l'ensemble des transitions de l'automate

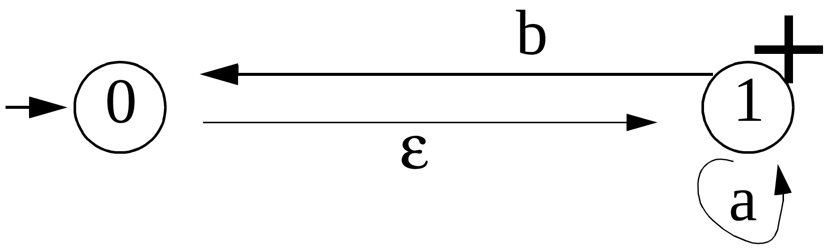
Pour fixer les choses



Quiz 1 - est un automate fini vrai faux



Quiz 2 - est un automate fini vrai faux



Quiz 3 - est un automate fini vrai faux

Automate fini : définitions (suite)

- chemin
- chemin vide
- trace, longueur d'un chemin
- mot **reconnu** par un automate
- **langage** reconnu par un automate L($\mathcal{A}$)
- famille des langages reconnaissables : **Rec**, **Rec(X*)**

Propriétés de la famille Rec(X*)

- **Quiz** - le langage vide est reconnaissable vrai faux
- X* est reconnaissable
- **Quiz** - tout langage fini est reconnaissable vrai faux

- Rec est close par union
- Rec est également close par

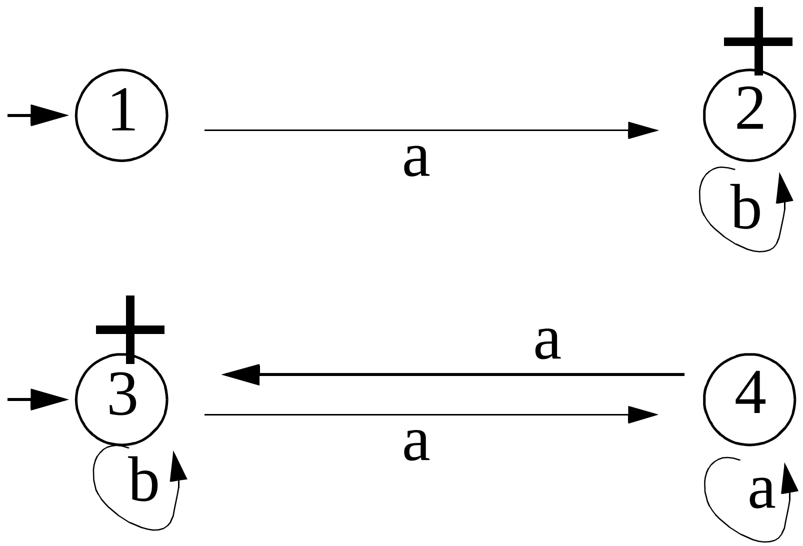
  - produit (cf TD3)

  - étoile (cf cours 5)

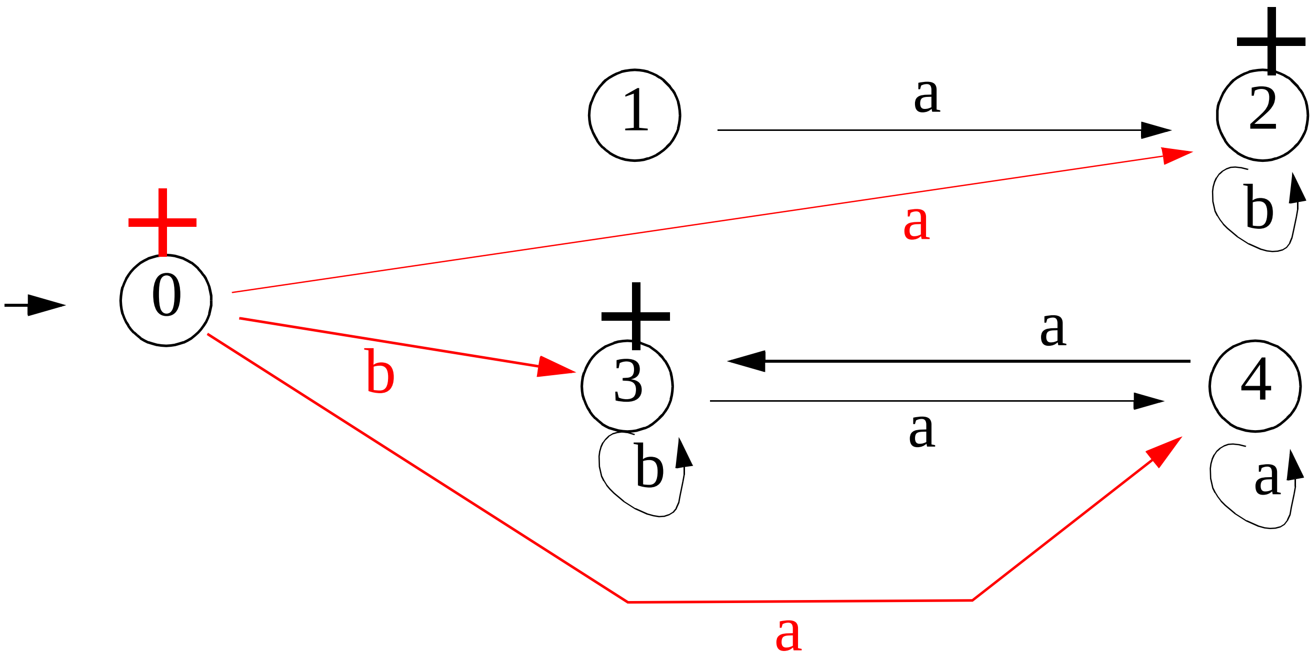
  - intersection (cf cours 5 et TD8)

  - complémentation (cf TD5)

Clôture des reconnaissables par union (exemple)



(suite exemple)



Clôture des reconnaissables par union (cas général)

- Soient L et M deux langages reconnaissables, la seule façon *pour l'instant* de montrer que $L \cup M$ est reconnaissable est de trouver un automate fini qui le reconnaît.

  - on construit un automate reconnaissant $L \cup M$ à partir de deux automates reconnaissant respectivement L et M,

  - il s'agit bien entendu de donner une construction générale qui convienne quels que soient les langages L et M considérés.

On a ainsi la propriété

- la famille de langages Rec(X^*) contient les parties finies (de X^*) et est close par union, produit et étoile.

(et beaucoup plus)