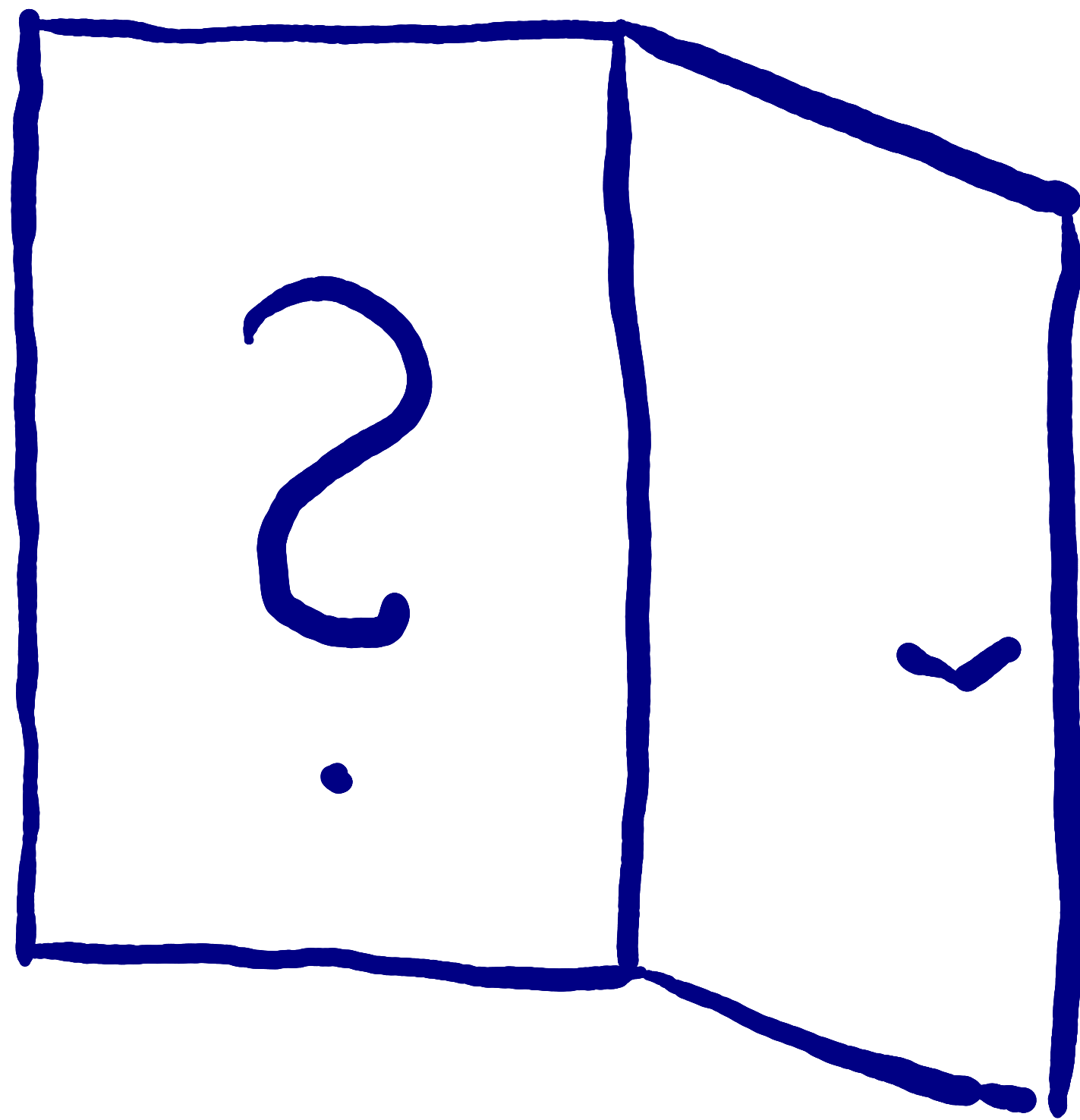


# Deallocation Sequences 2.0

- Into the Unknown -



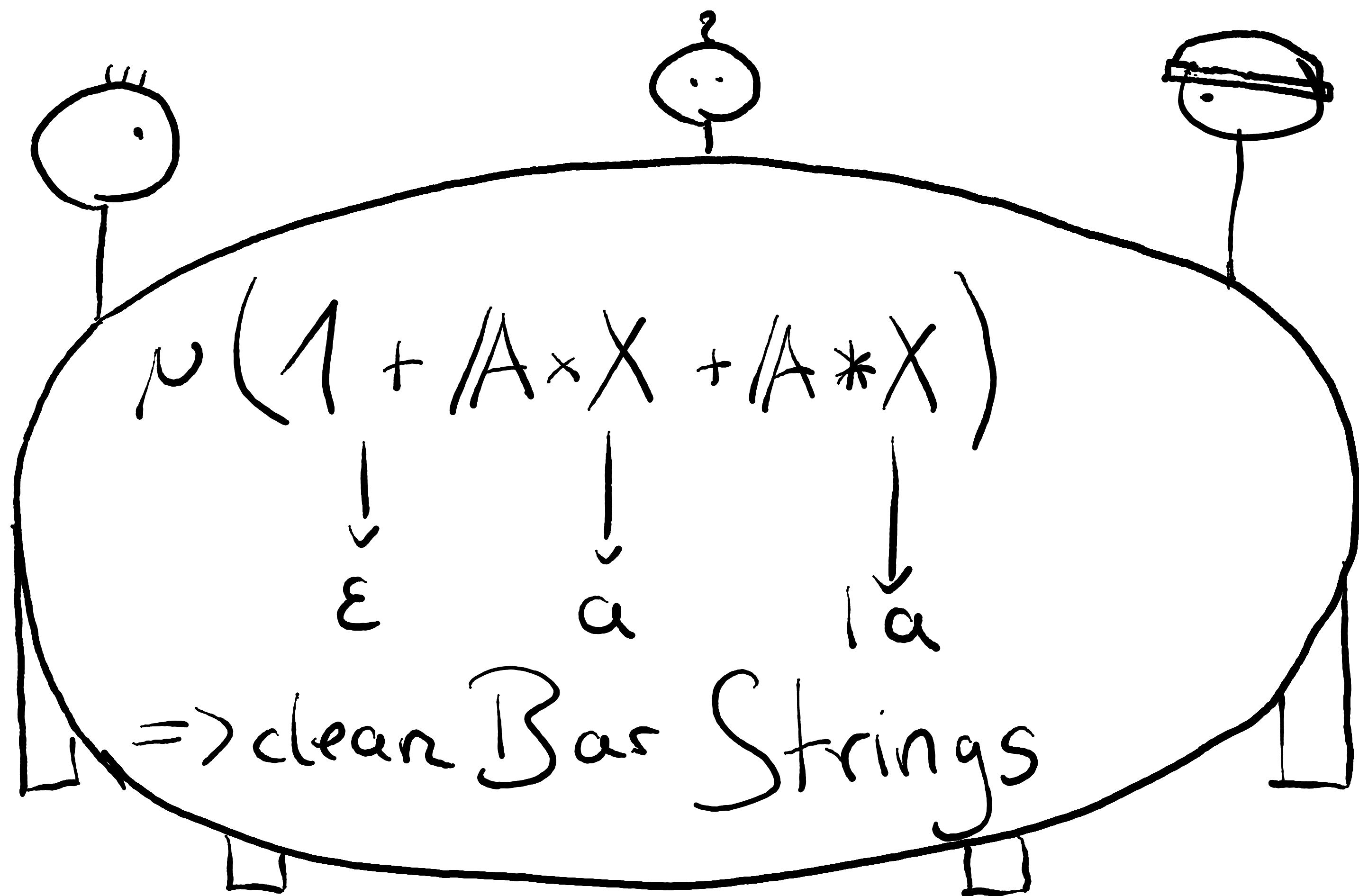
Work together with Stefan and Lutz

# Deallocation Sequences

or at least what is left from them, because we changed everything several times, and still do not really know if this works out (we are kind of confident it will, though).

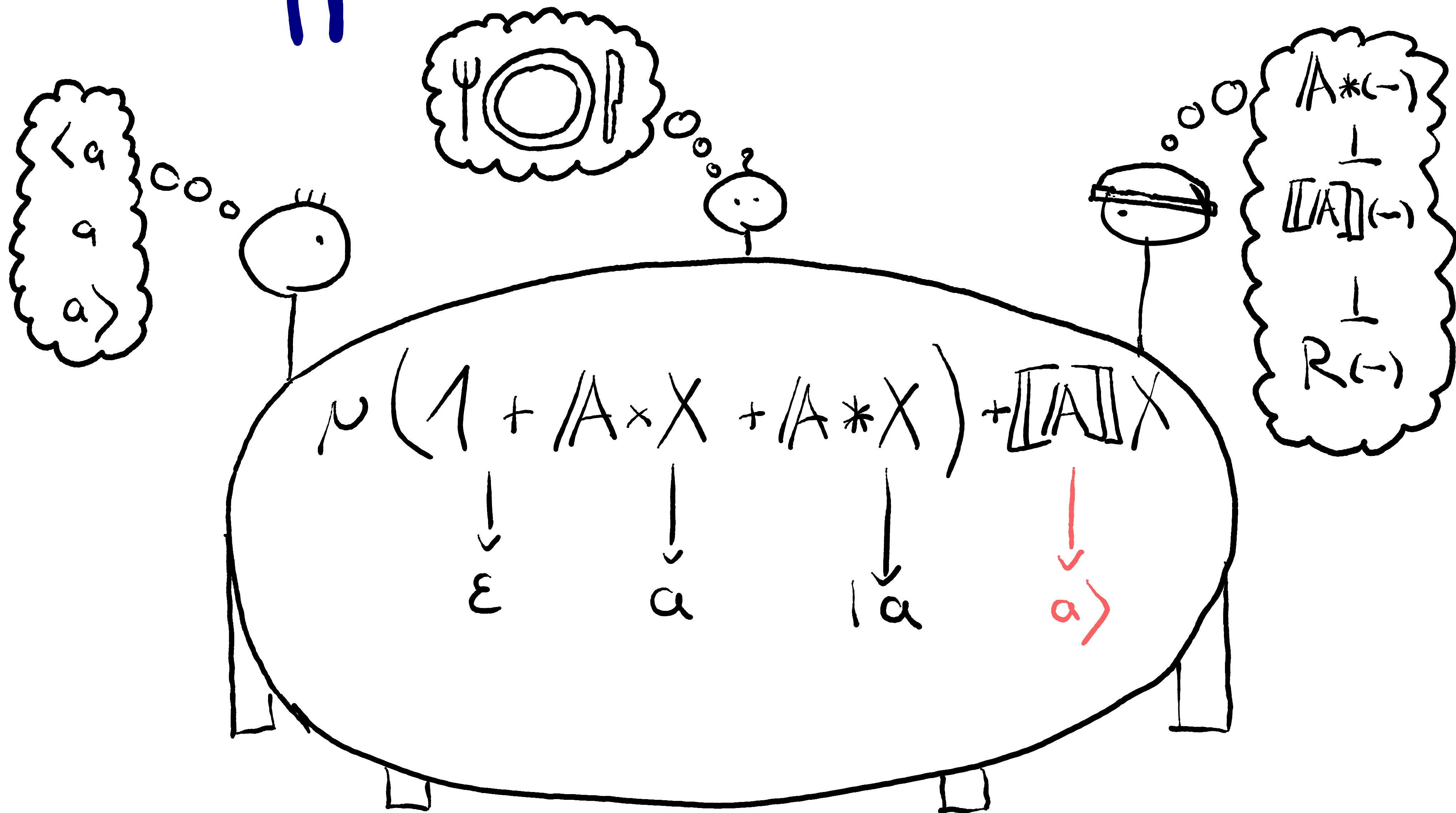
Work together with Stefan and Lutz

# What happened in the last season:



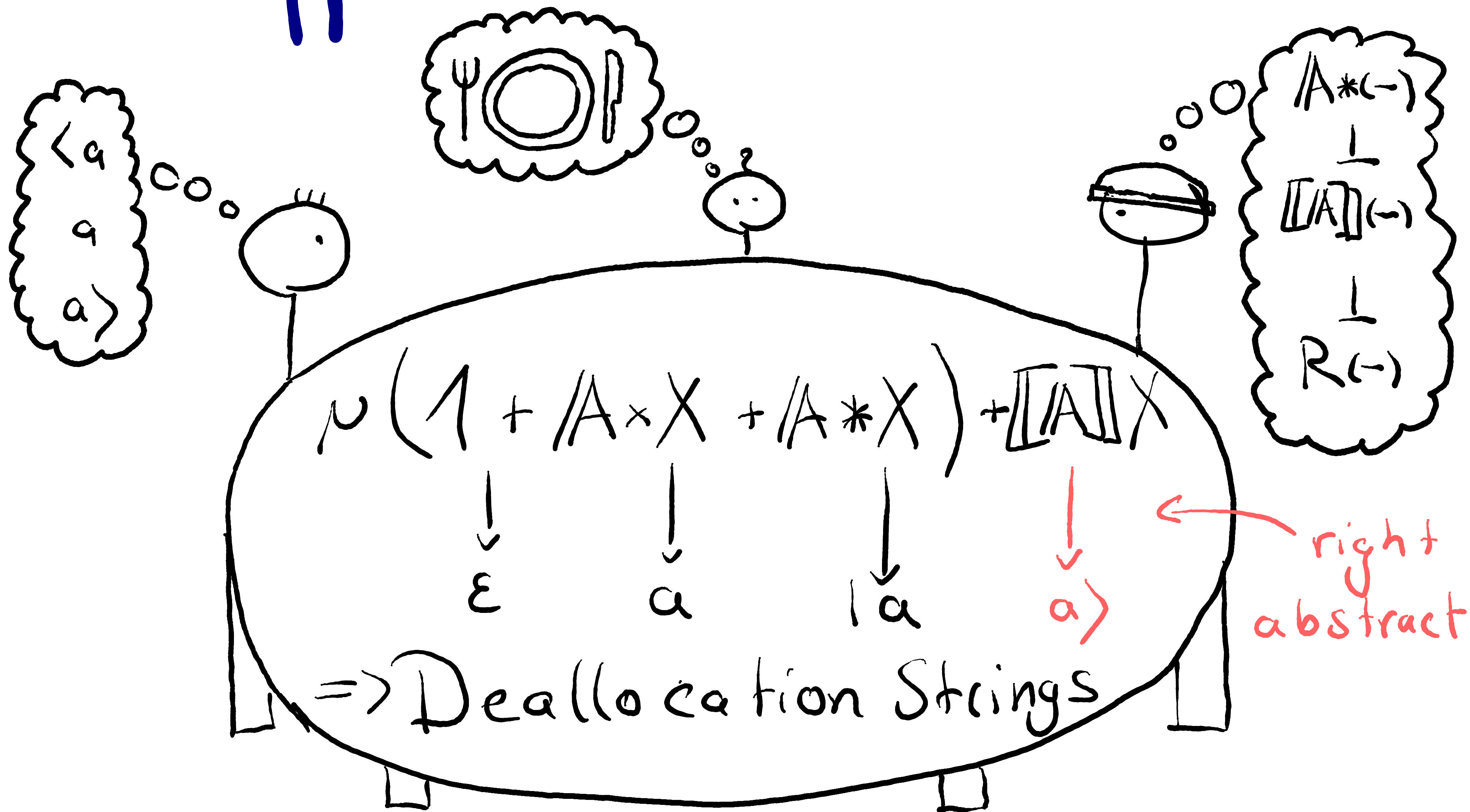


# What happened in the last season:

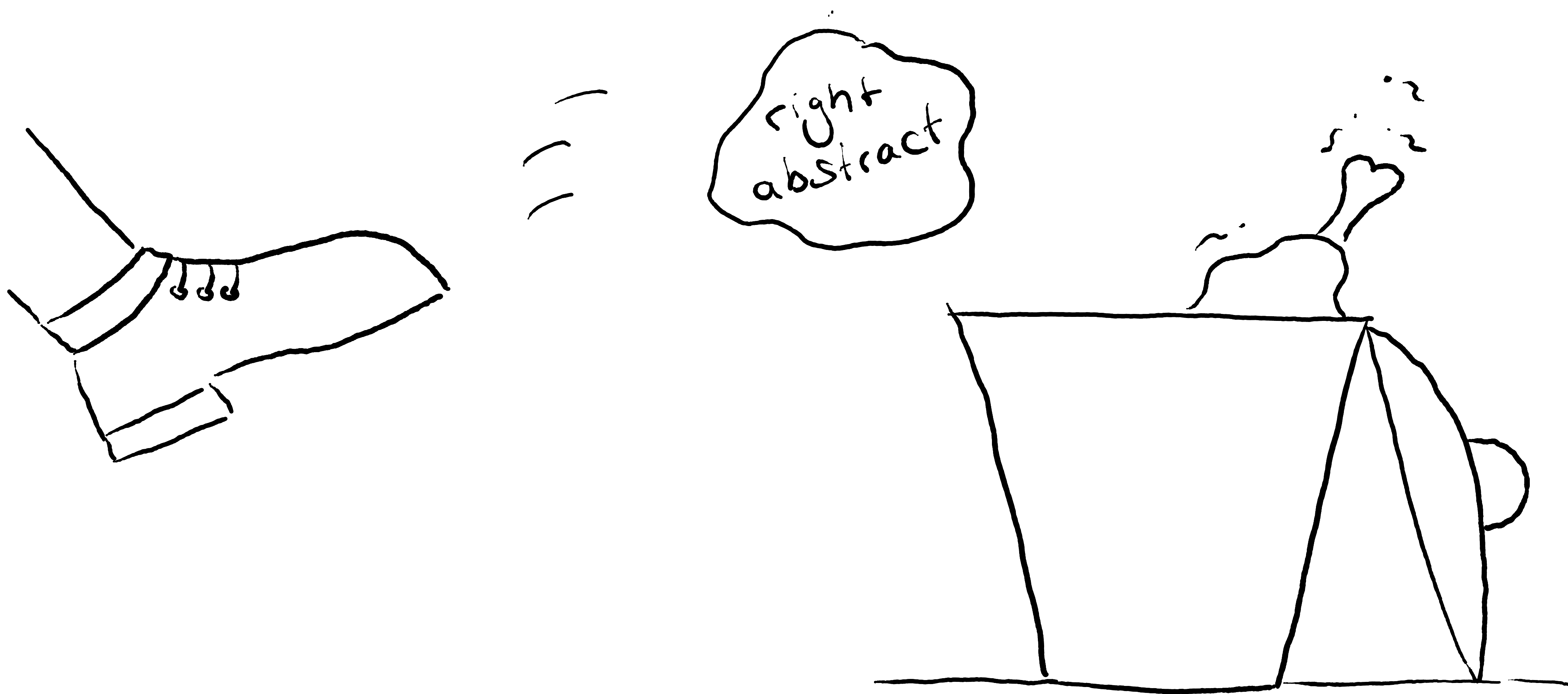




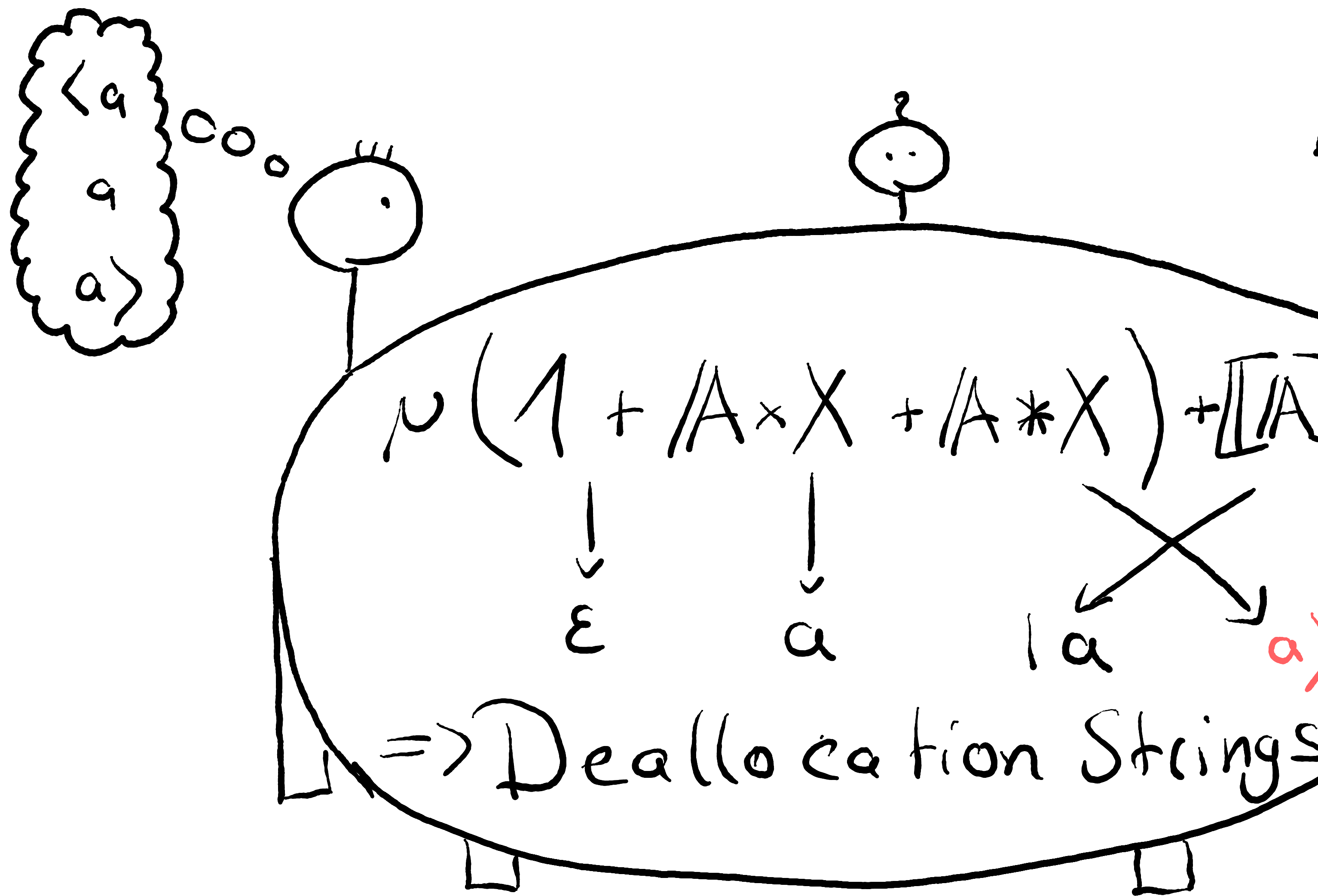
# What happened in the last season:



# What happened in the last season:

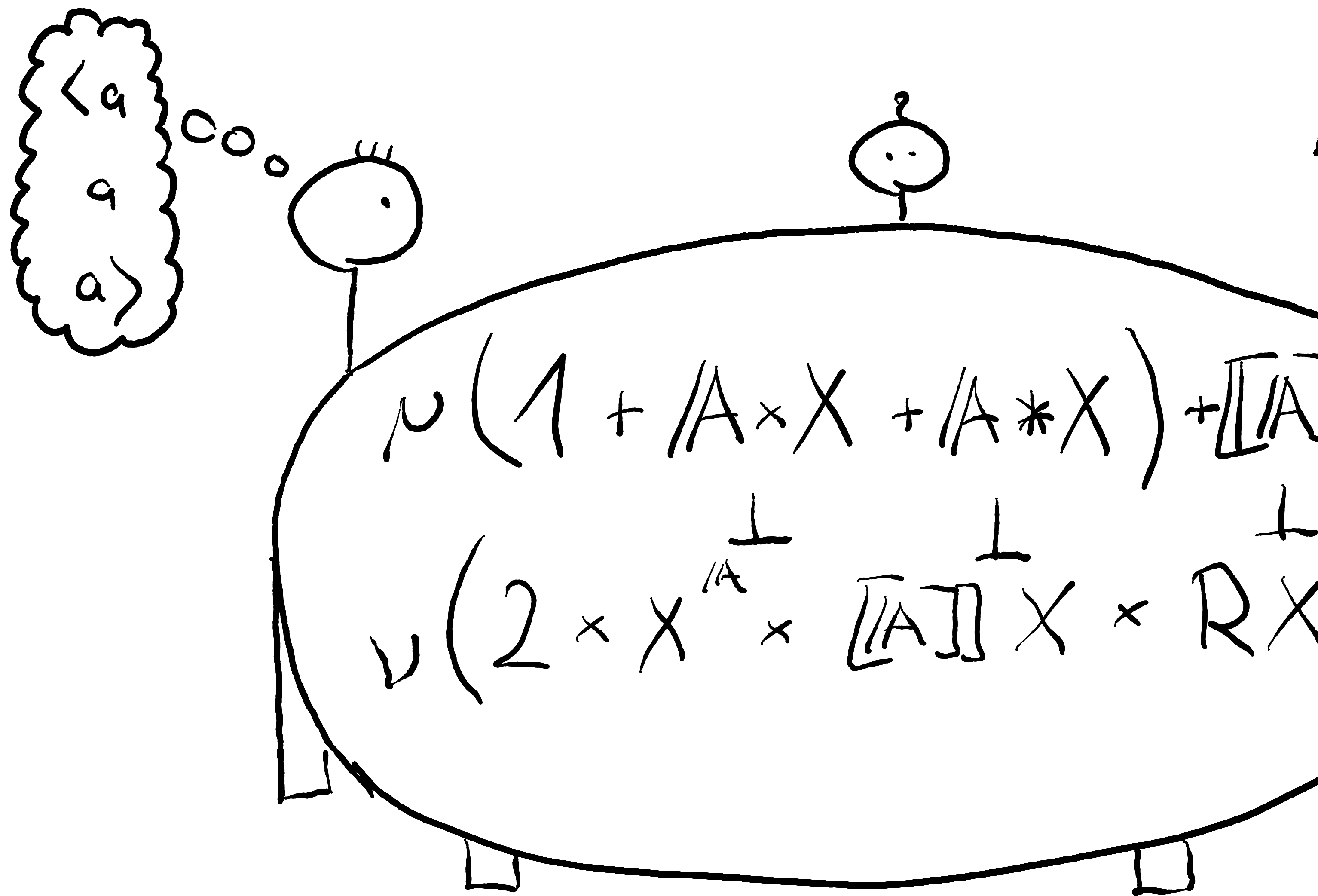


# What happened in the last session

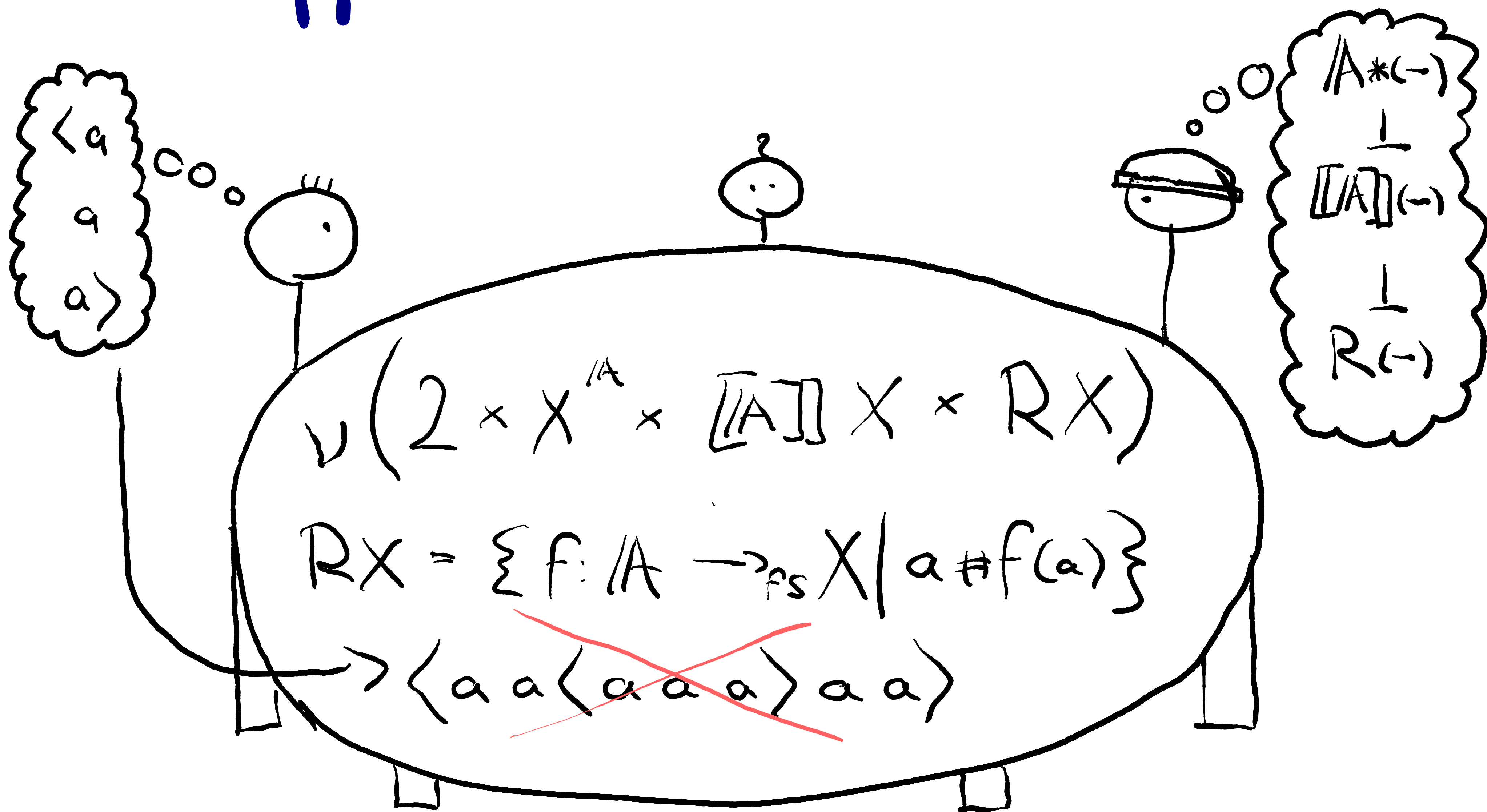




# What happened in the last session



# What happened in the last season:

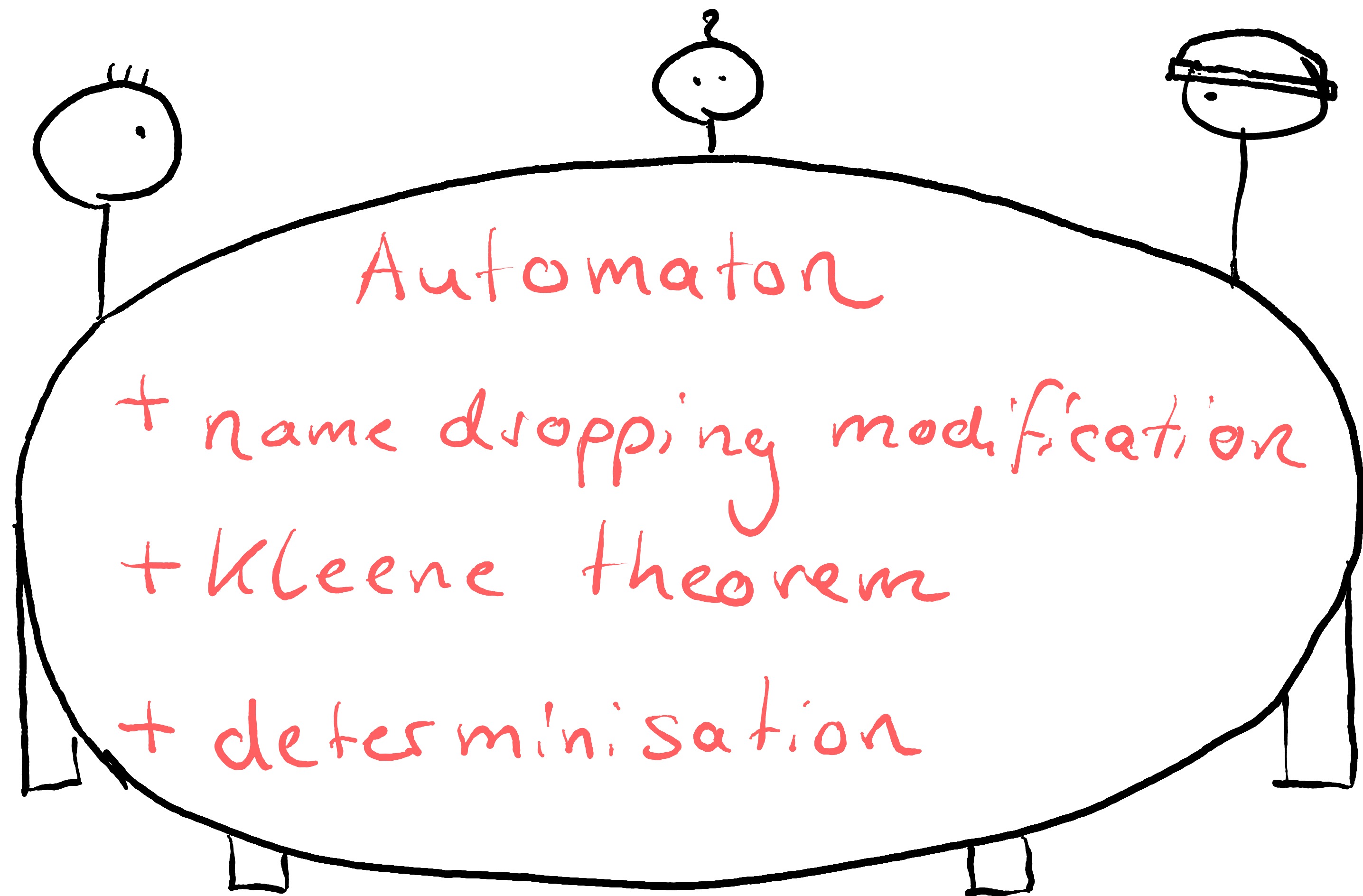


# What happened in the last season:





# What happened in the last season:



Spoiler: It did not work out

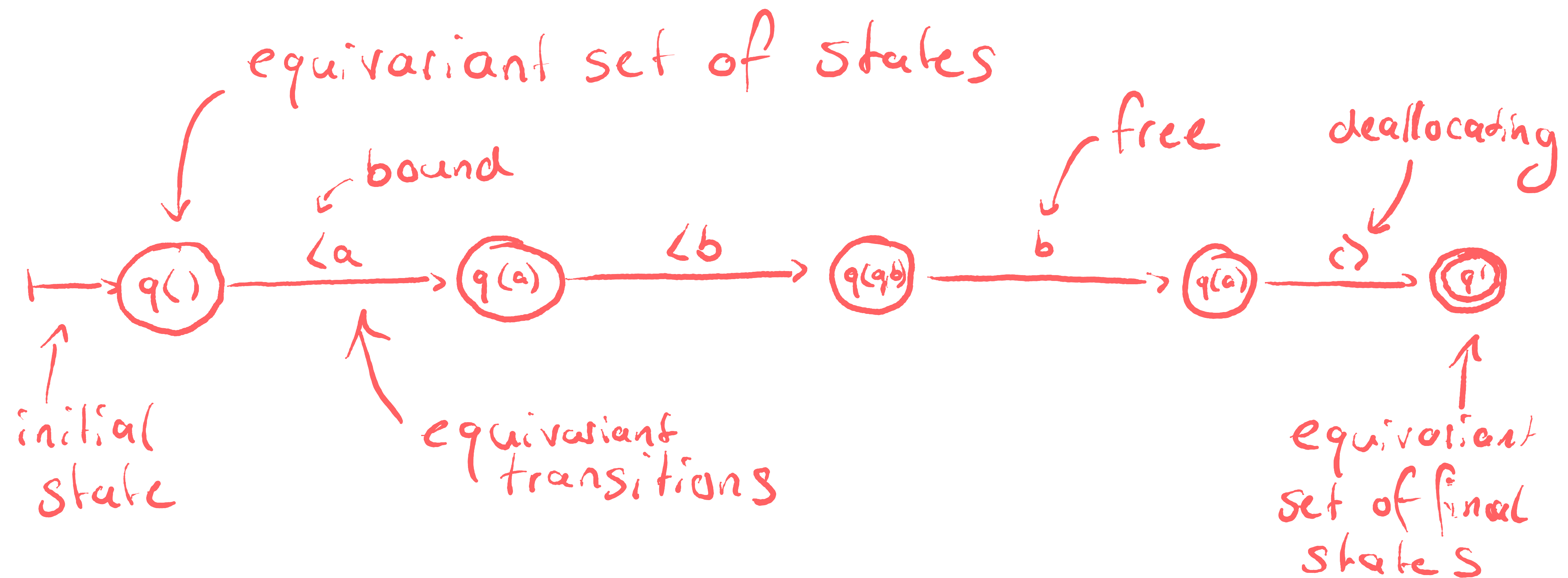


# Agenda:

- What we currently work with
- Our current challenges
- A novel interpretation

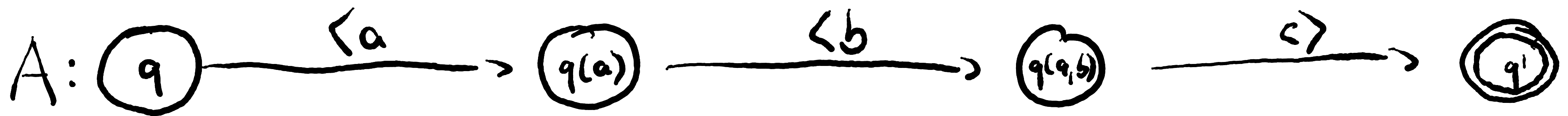
# Deallocation Strings and NOA

— Strings made from  $A + \text{Binders} \langle + \rangle$





# How name-dropping fails so far



$\hookrightarrow$  accepts  $\{\langle a \langle b c \rangle \mid a \neq b \neq c, a \neq c\}$  What about  $\langle a \langle ab \rangle \rangle$ ?

# Why our old semantics seem wrong

- Abstraction from right does not fit automaton
- Separated product  $A * X$  forbids multiple deallocs, still the automaton would accept some of these words
- Name capture artifact when name-dropping
- Sometimes abstract  $a \rangle$  and not  $aa \rangle$ ?

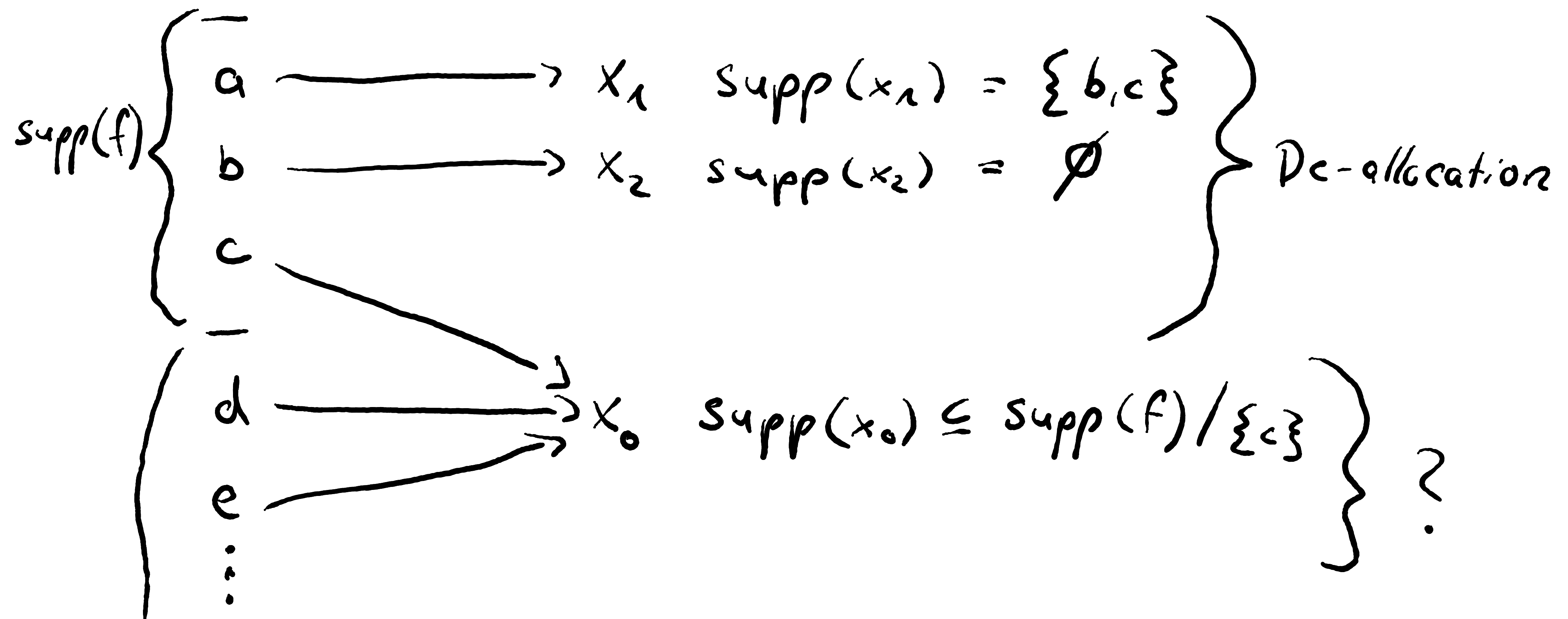
=> Apparently we have to understand the R for fresh names differently

# Recall: R Functor

$$[[A]](-) \dashv R(-)$$

$$RX = \{ f: A \rightarrow_{fs} X \mid a \# f(a) \}$$

In other terms:





# New (meta) symbol

allocation  $\{ a \quad a \quad \}$  deallocation  $\{ a \}$  + ?  
free  $\}$  unknown

- ? is intuitively like freeing an unallocated address in C or binding an anonymous parameter in functional programming languages

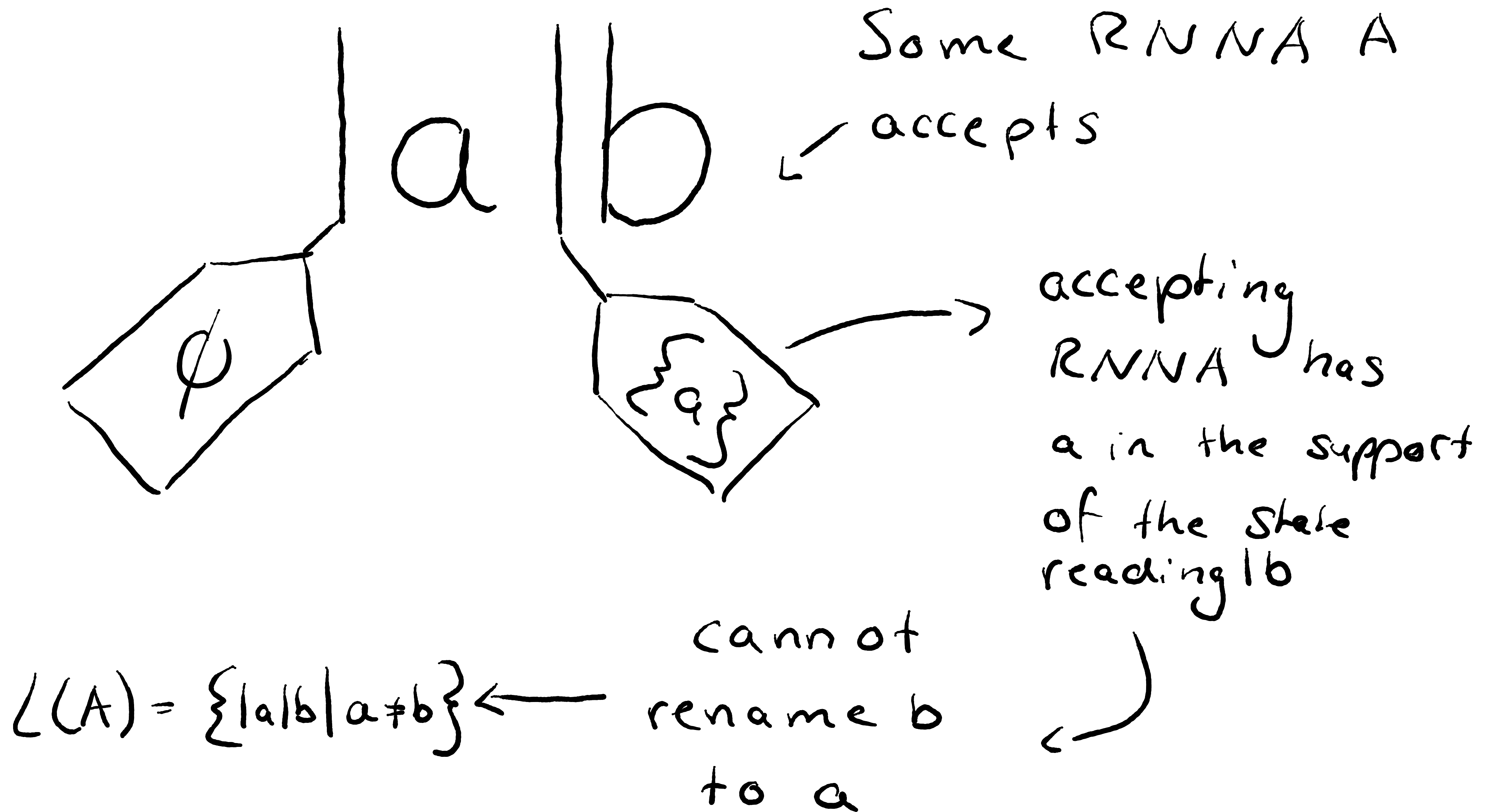


# Flashback: Bar Languages

RNNA    A    accepts     $1a1b$

Does    A    accept     $1a1a$  ?

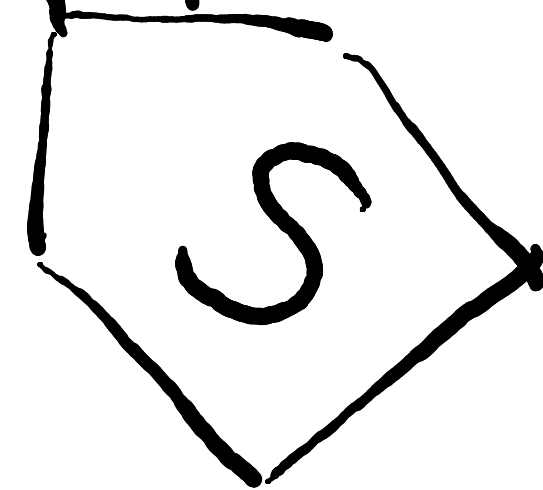
# How to imagine renaming:



Which 'Label' to attach to '?' ?

$\langle a \mid b \rangle_t$

-  $S \subseteq FN(t) \cup DN(t)$

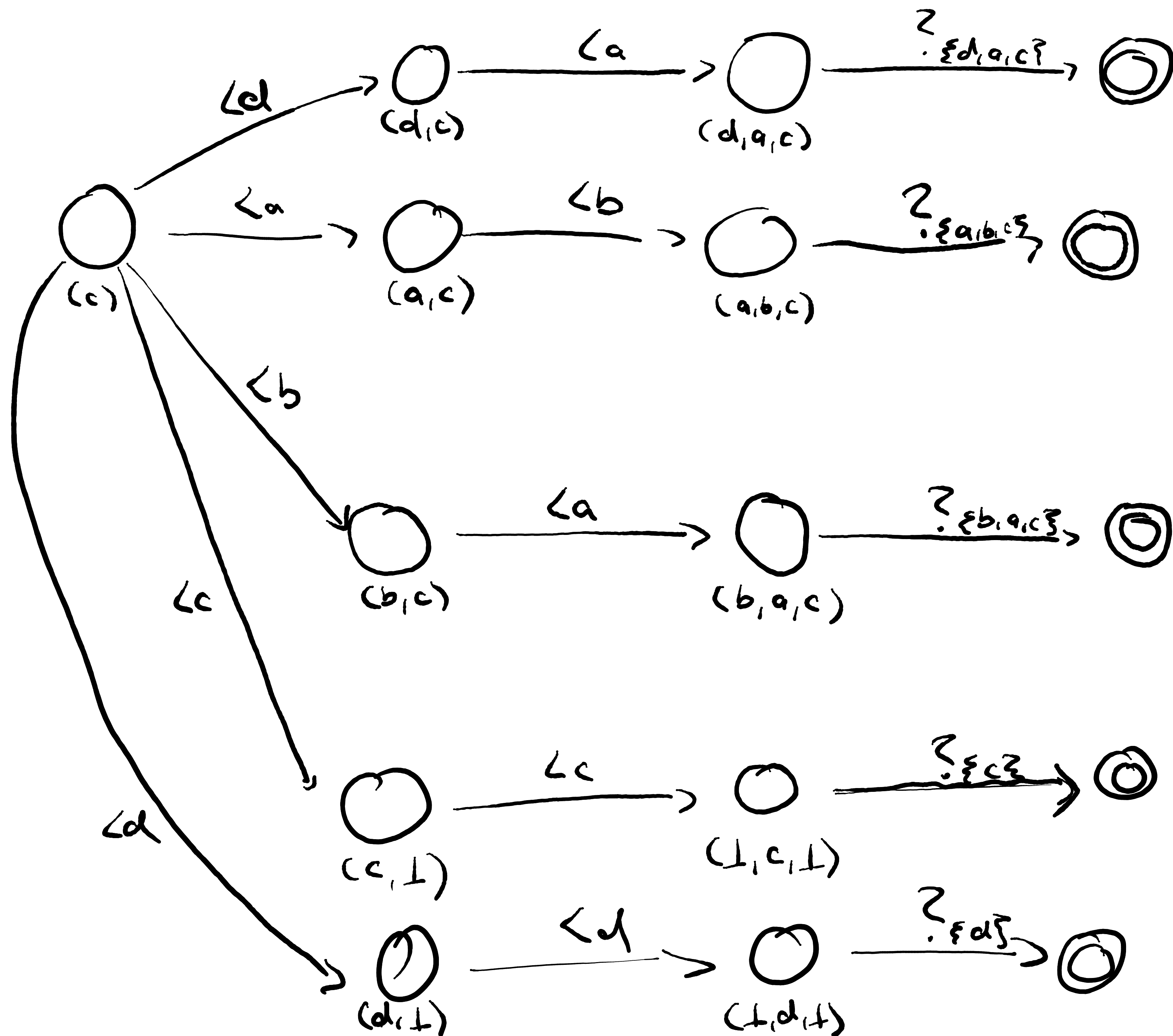


- Everything bound to this point?

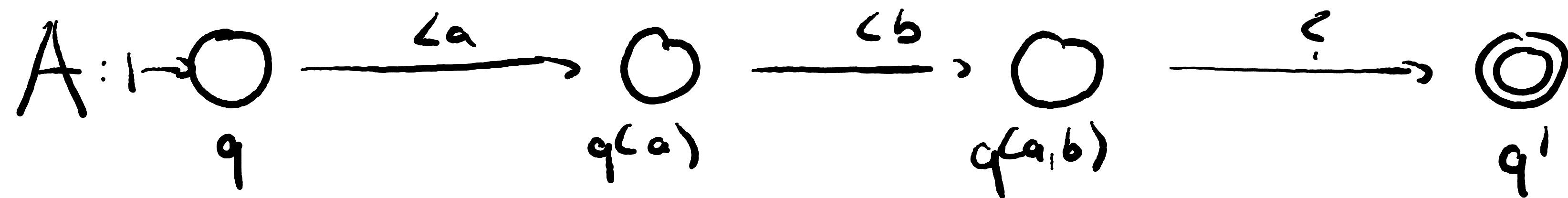
- Everything currently not overwritten



Example:



$$\langle b \rangle \langle b \rangle a \equiv_2 \langle b \rangle \langle b \rangle ? \equiv_q \langle a \rangle \langle b \rangle ? \equiv_2 \langle a \rangle \langle b \rangle a$$



$$L(A) = \{ \langle a \rangle \langle b \rangle ? \mid a \neq b \}$$



something but a and b



<a <b ?

a ≠ ? ≠ b

<a <b a

<a ? a



something but a

<a ? ?



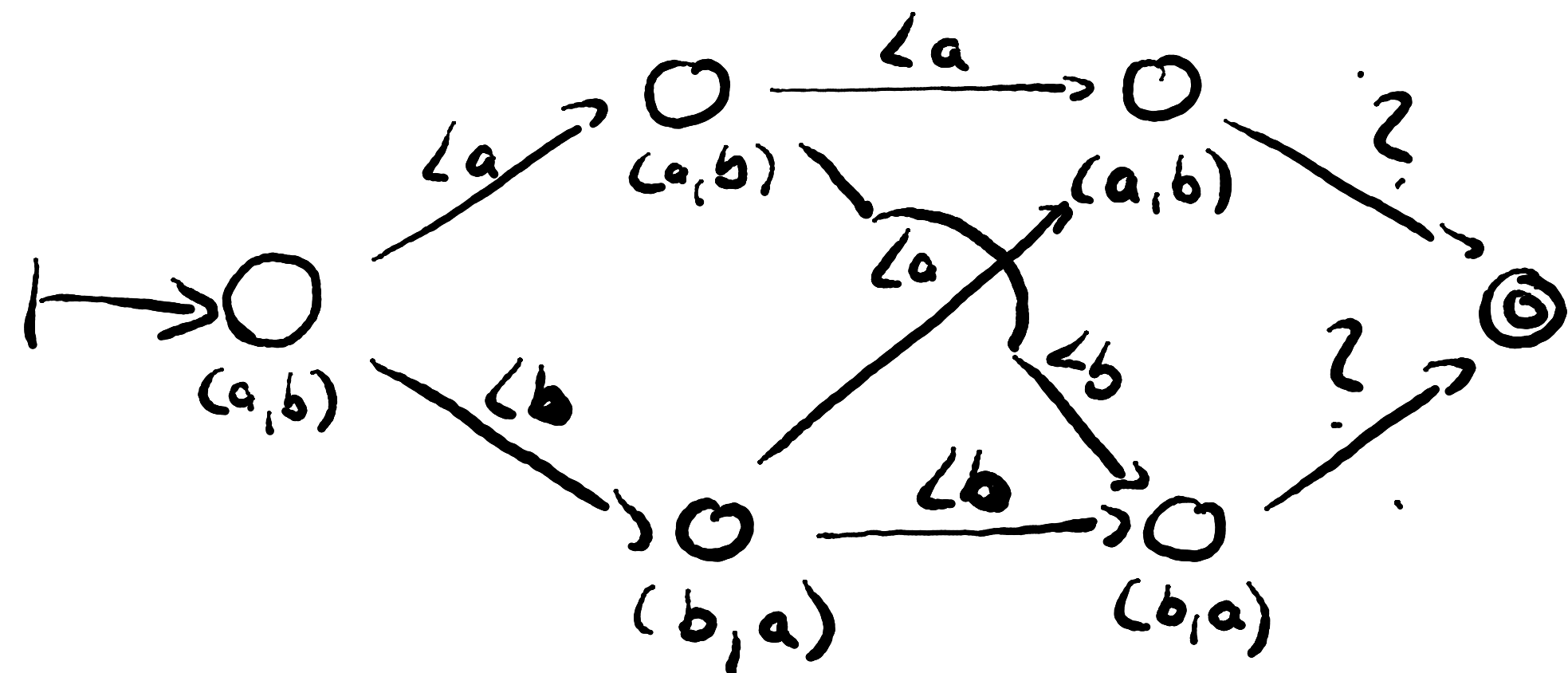
something but a

not a and b

$\neq a$   $\langle a \langle b ?$

$\langle a \langle a ?$

not this a



Distinct from the currently bound and not  
overwritten set of names



