

# Occurs Check



## Übersicht

- ◆ Wiederholung: Wann sind zwei Terme unifizierbar?
- ◆ Beispiel für das Problem
- ◆ Lösung des Problems: Occurs Check
- ◆ Verhalten von Prolog

## Ziel

- ◆ Verstehen, worin das Problem liegt

# Wann sind zwei Terme unifizierbar?

**Zwei Terme  $S$  und  $T$  sind unifizierbar, genau dann wenn:**

- ◆  $S$  und  $T$  sind dasselbe Atom,
- ◆ oder einer von ihnen ist eine freie Variable (in diesem Fall wird die Variable gebunden),
- ◆ oder  $S$  und  $T$  sind komplexe Terme, wobei
  - ◆  $S$  und  $T$  haben denselben Funktor,
  - ◆ und  $S$  und  $T$  haben dieselbe Stelligkeit,
  - ◆ und die einzelnen Argumente sind jeweils paarweise unifizierbar.

# A Problem Occurs ...

---

**Das Problem zeigt sich bei folgender Frage:**

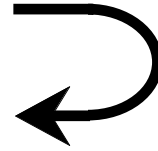
- ◆ Sind die Terme  $X$  und  $f(X)$  miteinander unifizierbar?
- ◆ Wenn ja, was ist das Ergebnis?

$X$

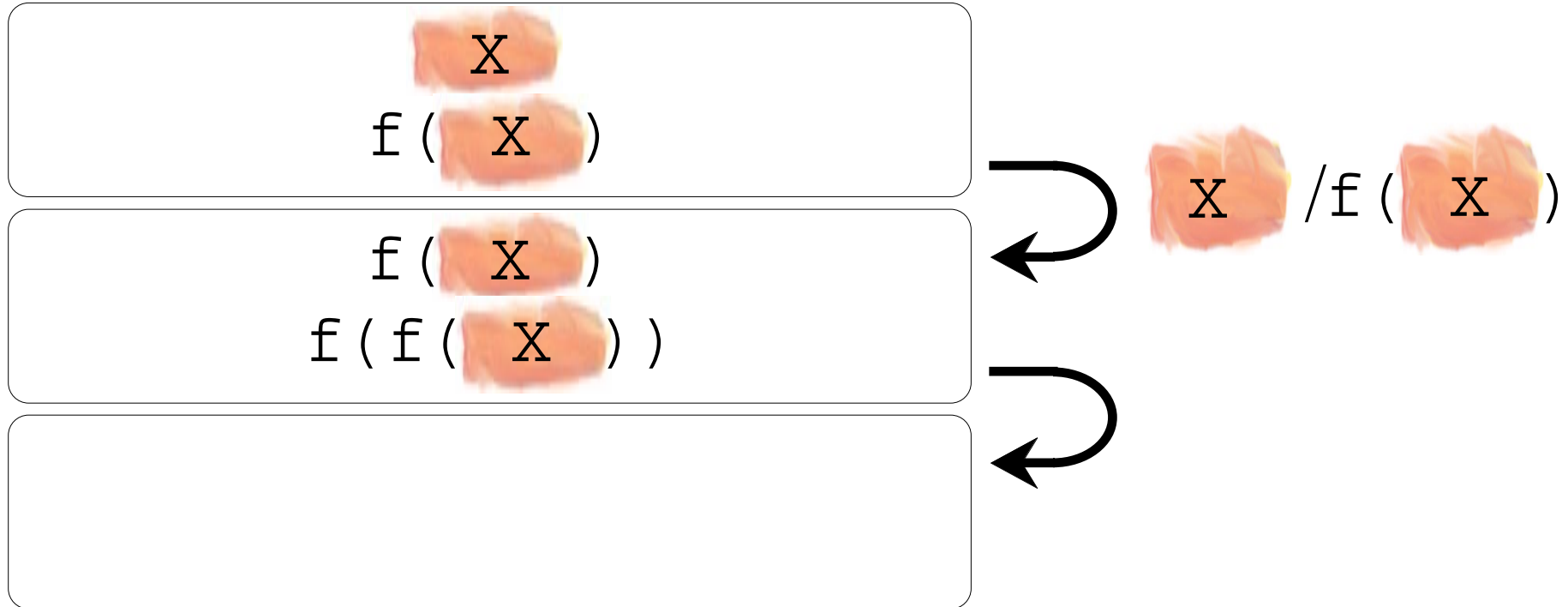
$f(X)$

# A Problem Occurs ...

X  
f ( X )



# A Problem Occurs ...

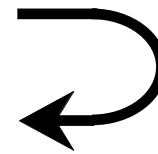
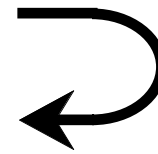
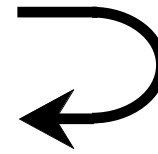


# A Problem Occurs ...

X  
f ( X )

f ( X )  
f ( f ( X ) )

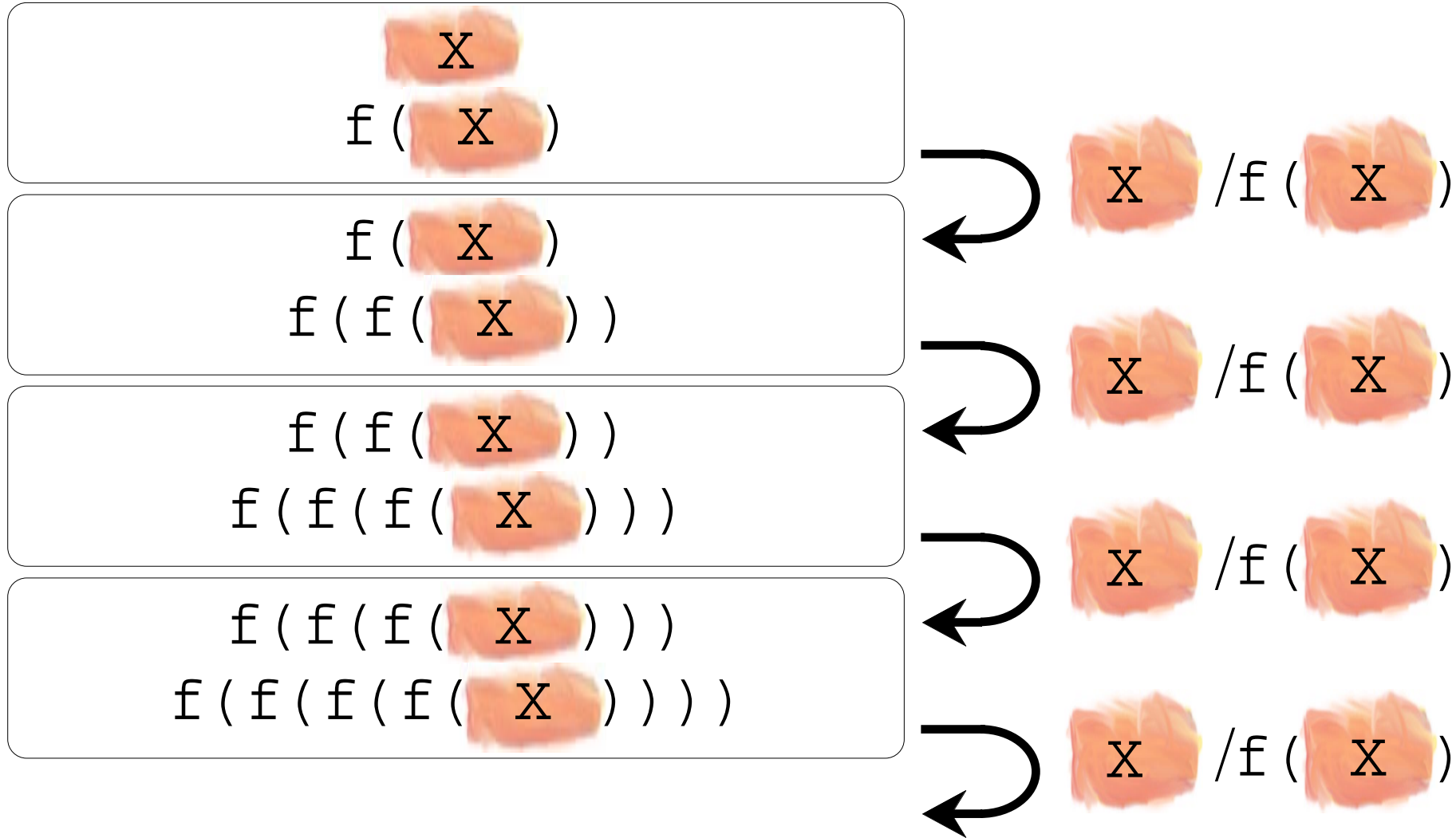
f ( f ( X ) )  
f ( f ( f ( X ) ) )



X / f ( X )

X / f ( X )

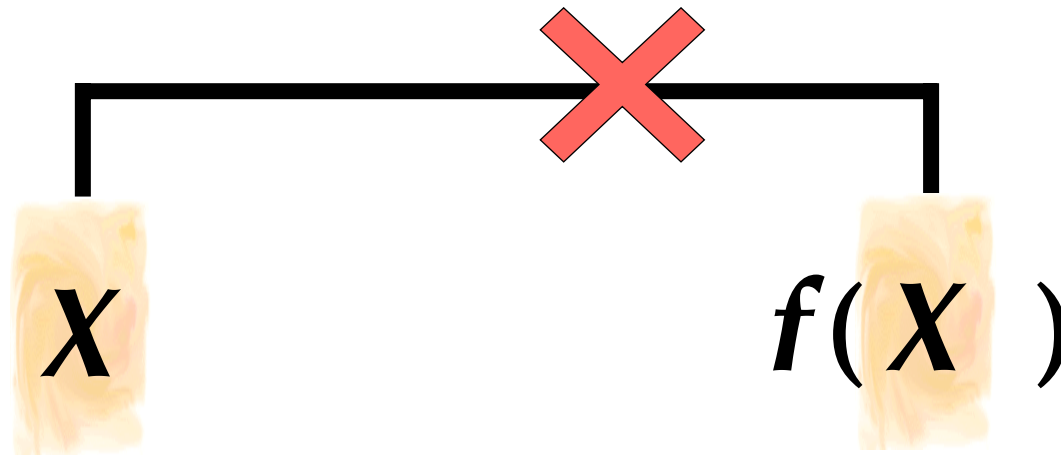
# A Problem Occurs ...



# Vermeiden des Problems

Eine Variable darf nicht an einen Term gebunden werden, der dieselbe Variable enthält.

- ◆ Dieser Sicherheitstest heisst **Occurs Check** (»Auftauchttest«).
- ◆ Gemäss mathematisch korrekter Definition der Unifikation wären  $X$  und  $f(X)$  nicht miteinander unifizierbar.







# Was macht Prolog?



## Prolog führt den *Occurs Check* offensichtlich nicht durch.

- ◆ Der Test ist relativ aufwendig durchzuführen
- ◆ Die meisten Prolog-Interpreter lassen ihn daher einfach weg
  - ... denn in der Praxis kommt das Problem eher selten vor
- ◆ Allerdings bieten viele Prolog-Implementationen ein besonderes Prädikat für die Unifikation *mit Occurs Check* an

## Es wäre auch möglich, zyklische Strukturen zuzulassen.

- ◆ andere Definition der Unifikation
- ◆ von einigen Prolog-Versionen unterstützt