An Introduction to Minimalist Grammars:

Locality - Late Adjunction and Extraposition

(July 23, 2009)

Gregory Kobele
Humboldt Universität zu Berlin
University of Chicago
kobele@rz.hu-berlin.de

Jens Michaelis
Universität Bielefeld
jens.michaelis@uni-bielefeld.de

Further outlook (taken from slides July 22)

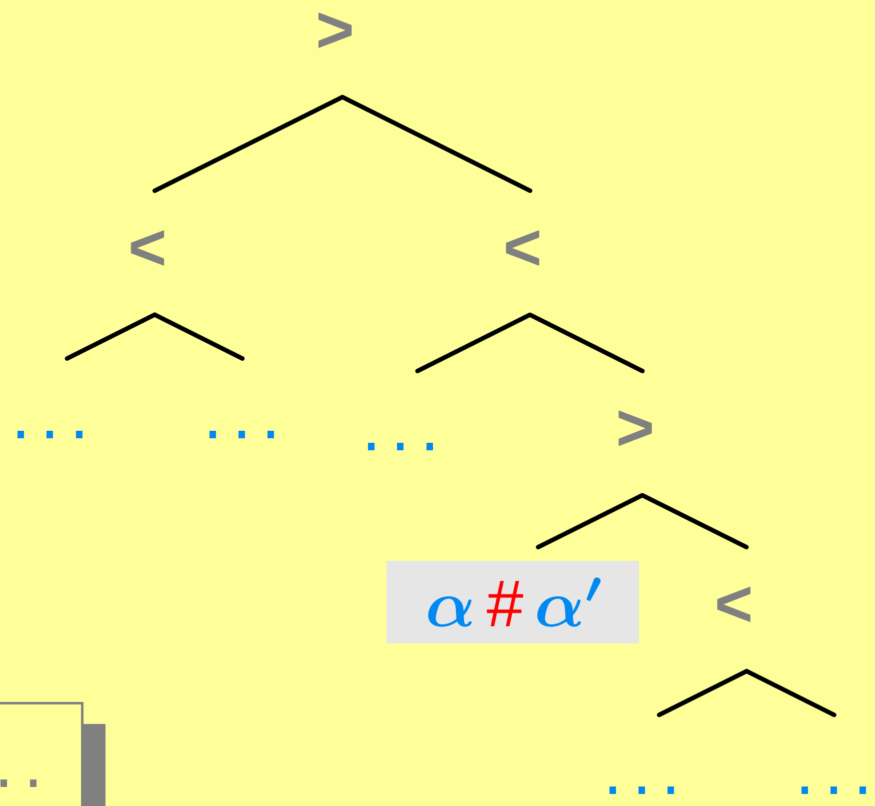
- MGs can be extended with the operations adjoin and scramble involving two new types of syntactic features and a unilateral checking of their instantiations (Frey & Gärtner 2002, Gärtner & Michaelis 2003).

- If, in particular, categorial features are not deleted after checking, but marked as checked — and thus are still accessible — acyclic (“late”) adjunction can be defined as a subtype of adjoin.

- As to the interaction of the SMC and a corresponding adjunct island constraint (AIC), the addition of the AIC has no effect, independently of the presence of the SMC.

Minimalist expressions

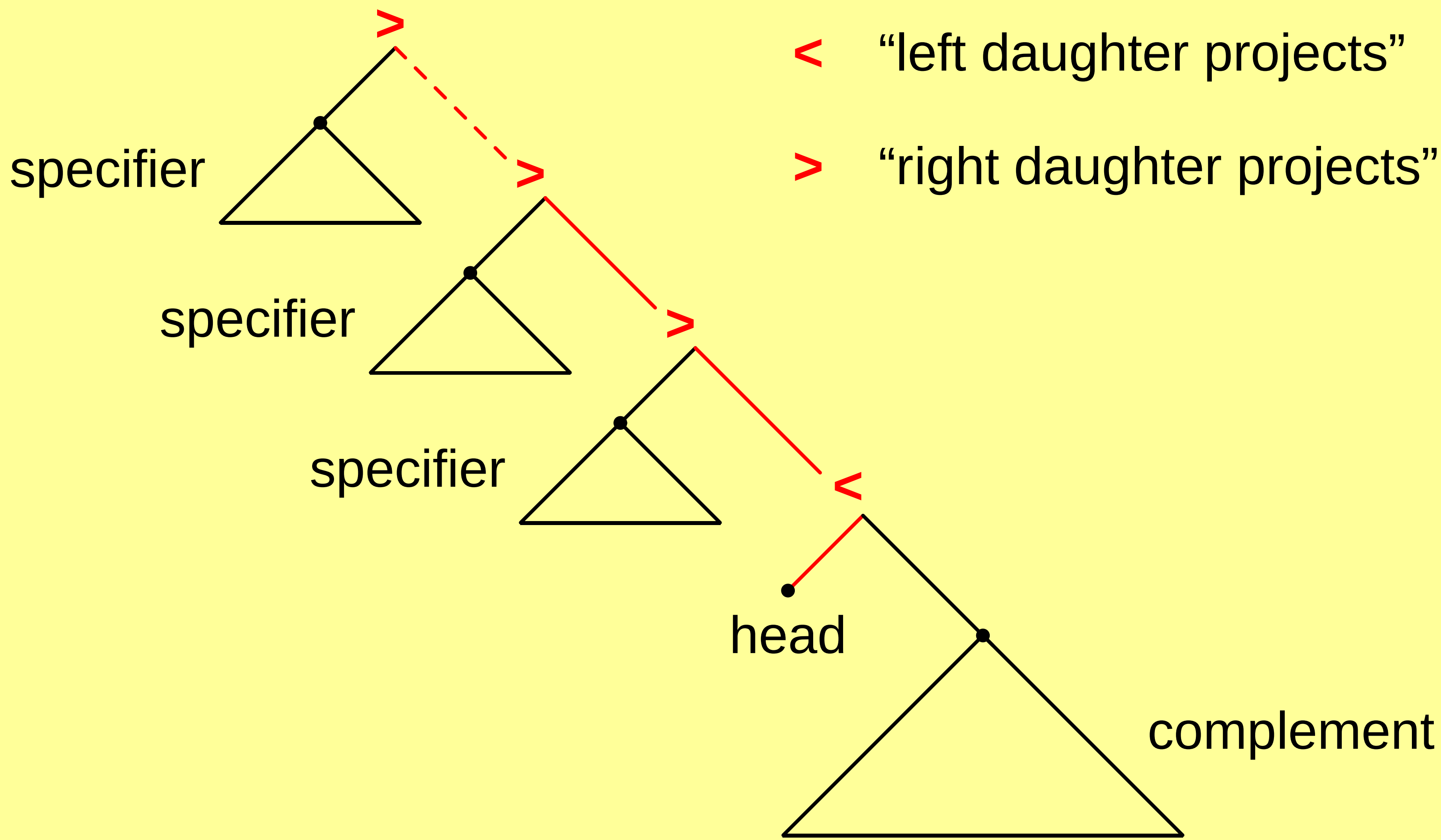
Vocabulary	non-syntactic features / terminals
SynFeatures	syntactic features



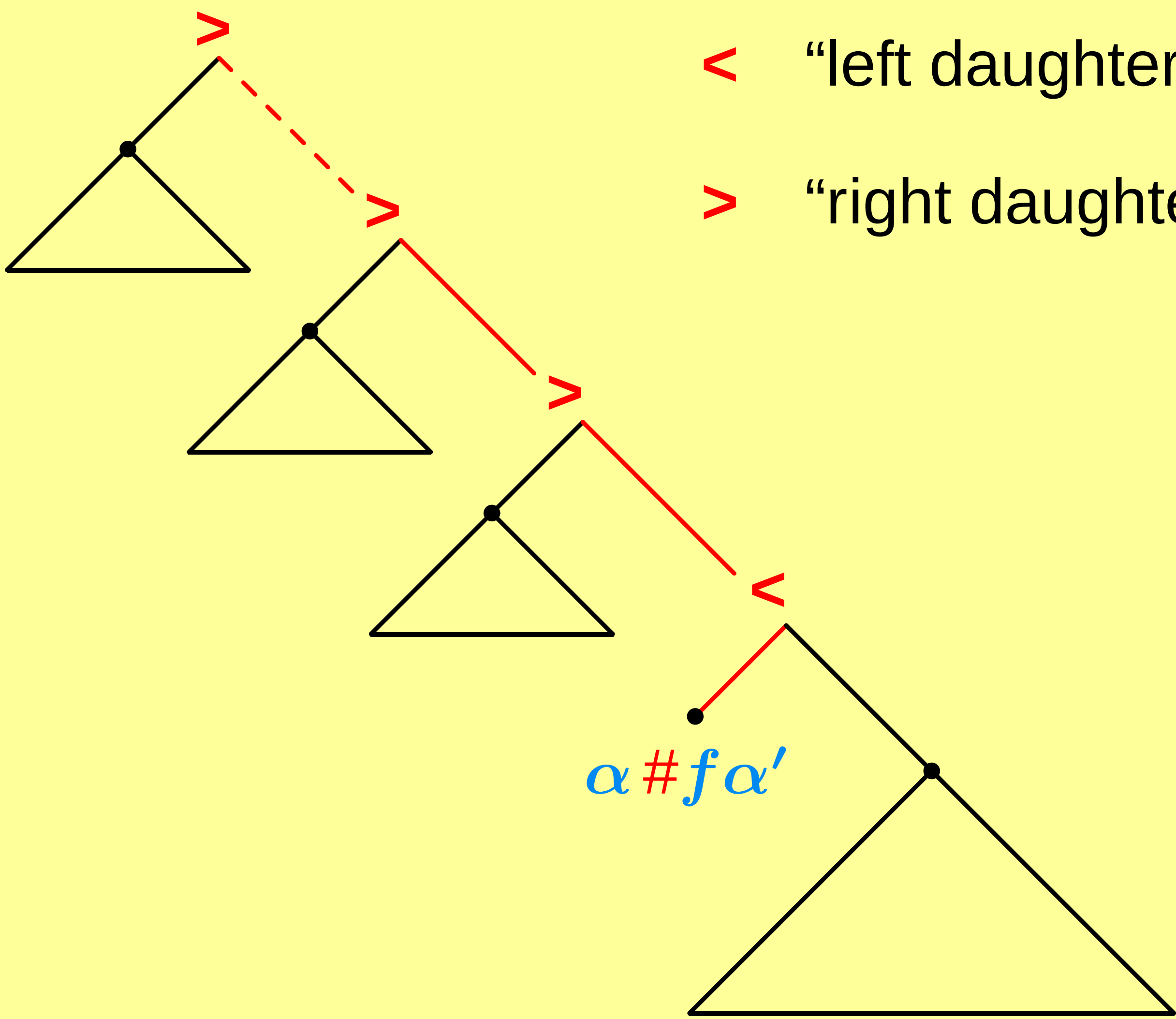
finite, binary labeled trees such that . . .

- non-leaf-labels are from $\{ < , > \}$ [“projection”]
- leaf-labels are from SynFeatures* {#} SynFeatures* . Vocabulary*

Minimalist expressions



Minimalist expressions



< “left daughter projects”

> “right daughter projects”

tree displays feature f $:\Longleftrightarrow$ head-label is of the form $\ldots \# f \ldots$

Building minimalist expressions

- Starting from a finite set of simple expressions (a lexicon),

  minimalist expressions can be built up recursively

  - by applying structure building functions

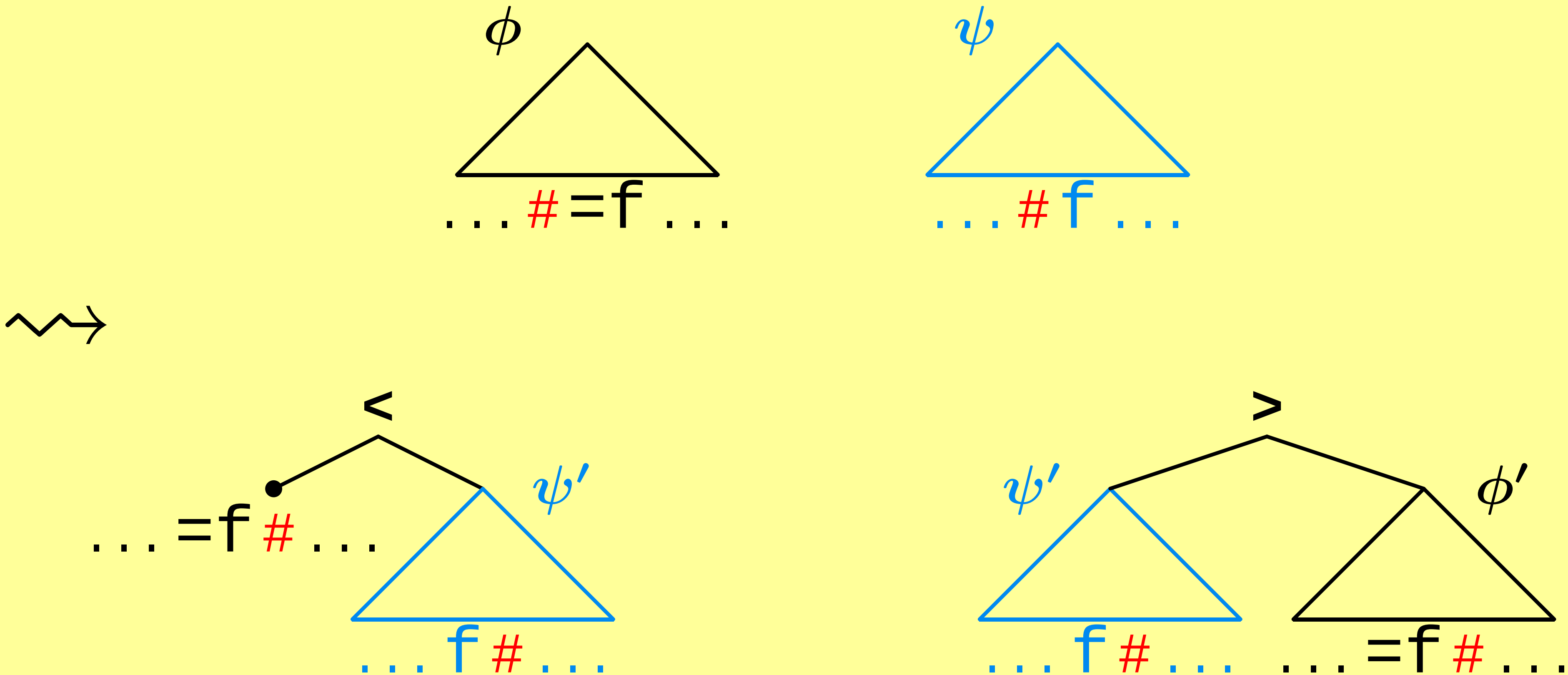
    checking off instances of syntactic features “from left to right,”

  - and after an application the

    triggering feature instances are marked as checked.

Structure building functions

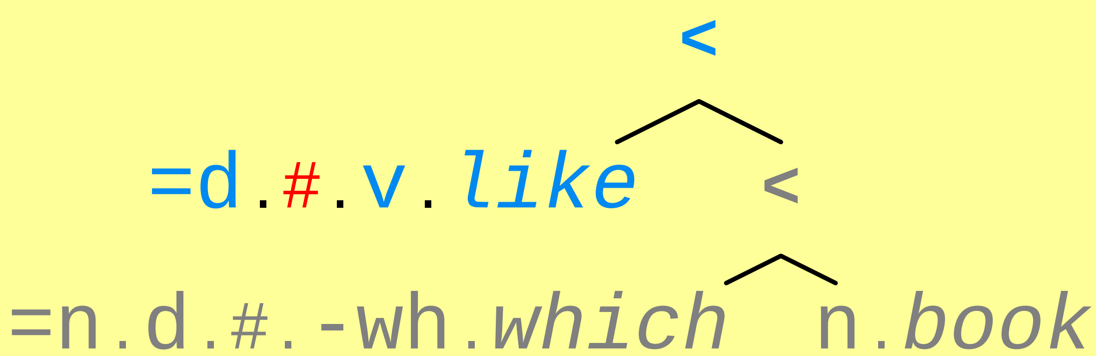
$$\mathsf{merge} : \mathsf{Trees} \times \mathsf{Trees} \xrightarrow{\text{part}} \mathsf{Trees}$$



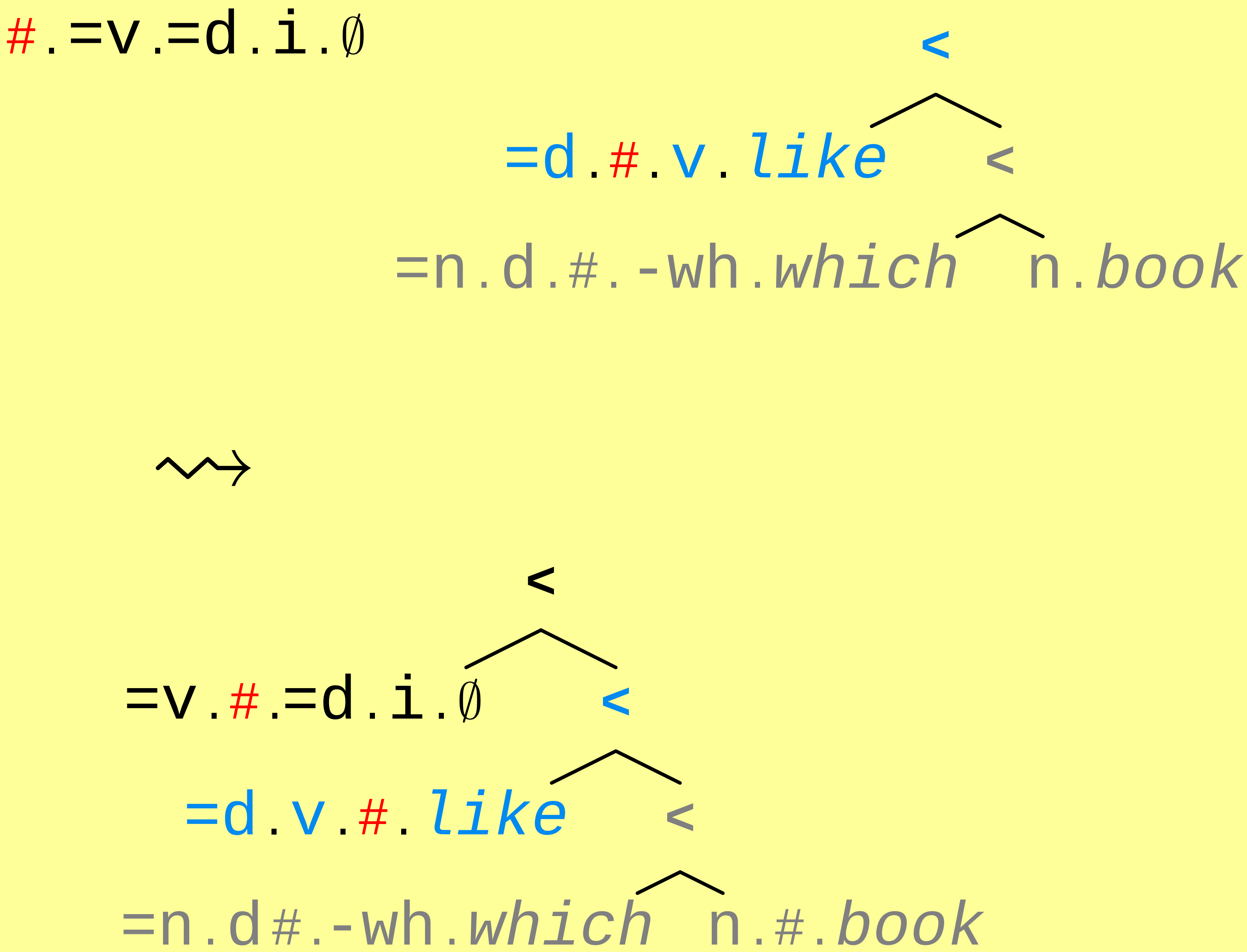
selecting ϕ simple

selecting ϕ complex

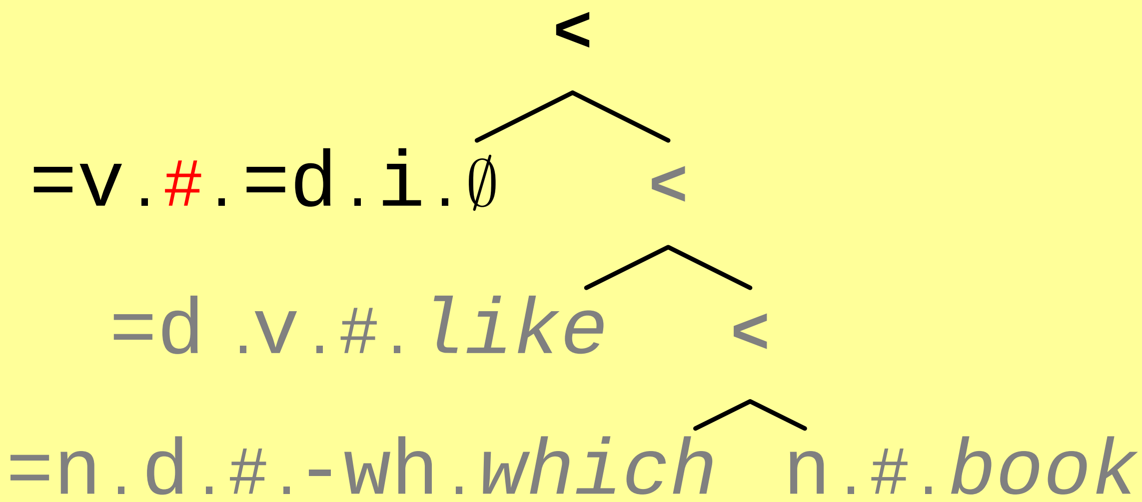
#.=v.=d.i.∅



merge (selecting tree is simple)

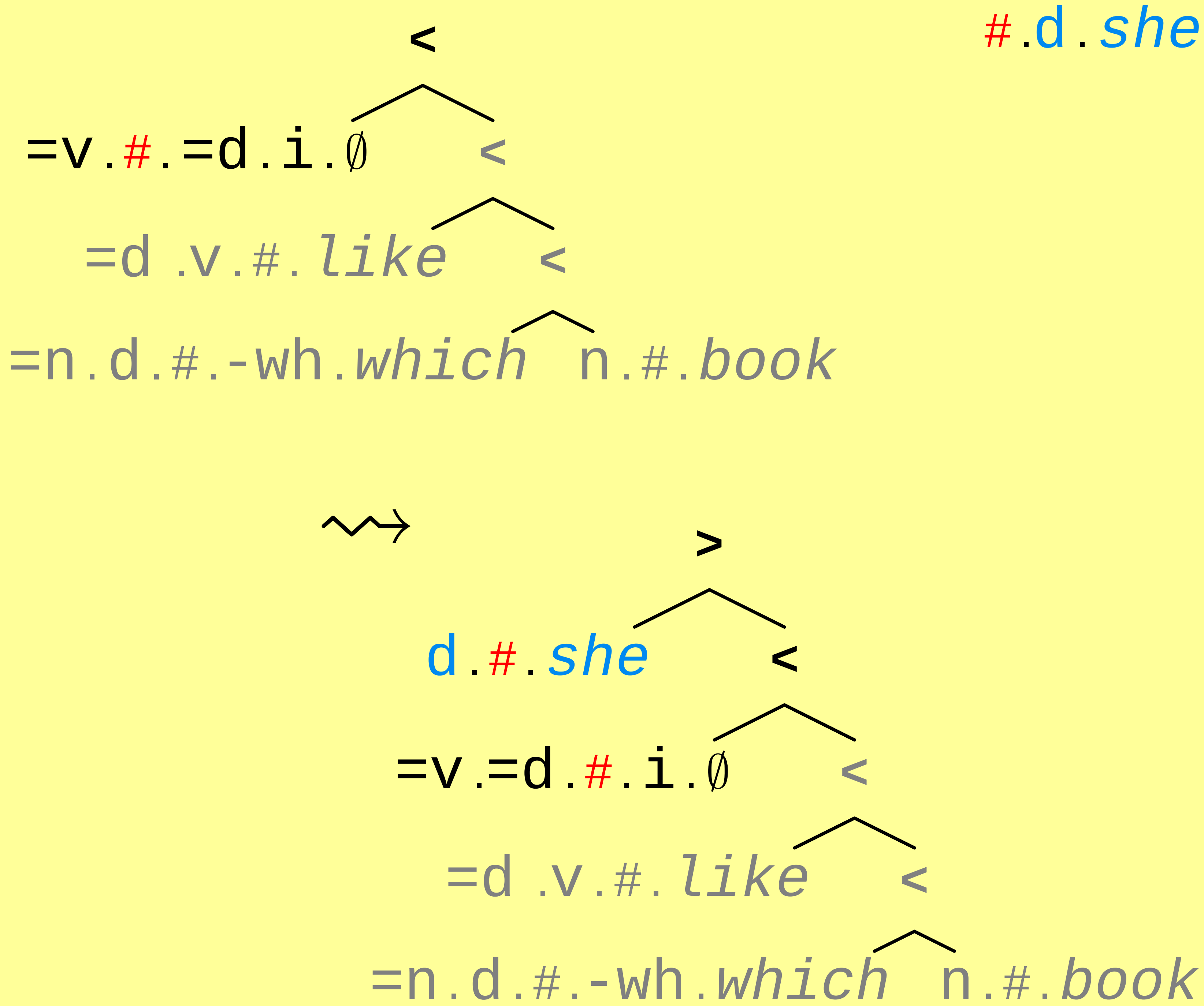


merge (selecting tree is complex)



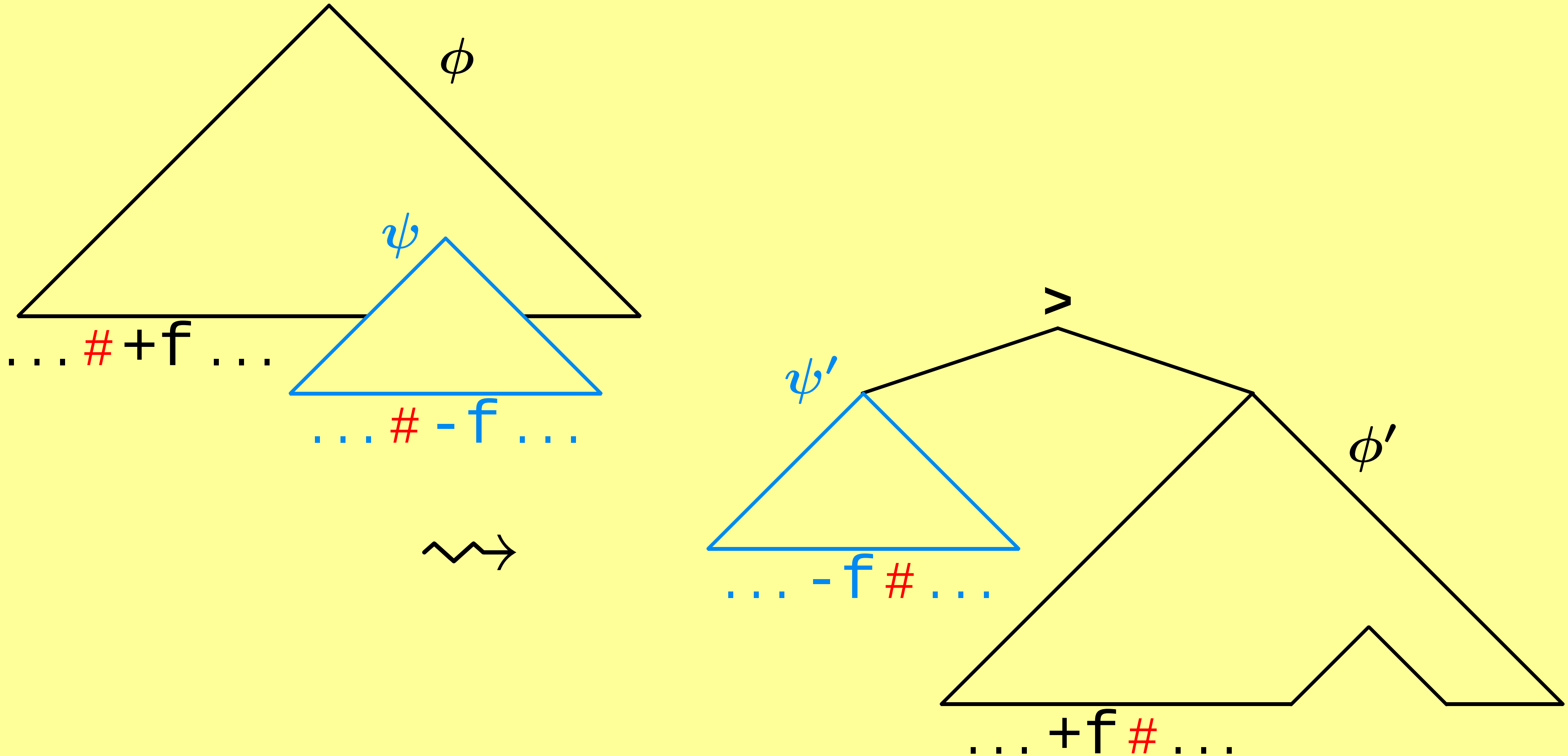
#.d.*she*

merge (selecting tree is complex)

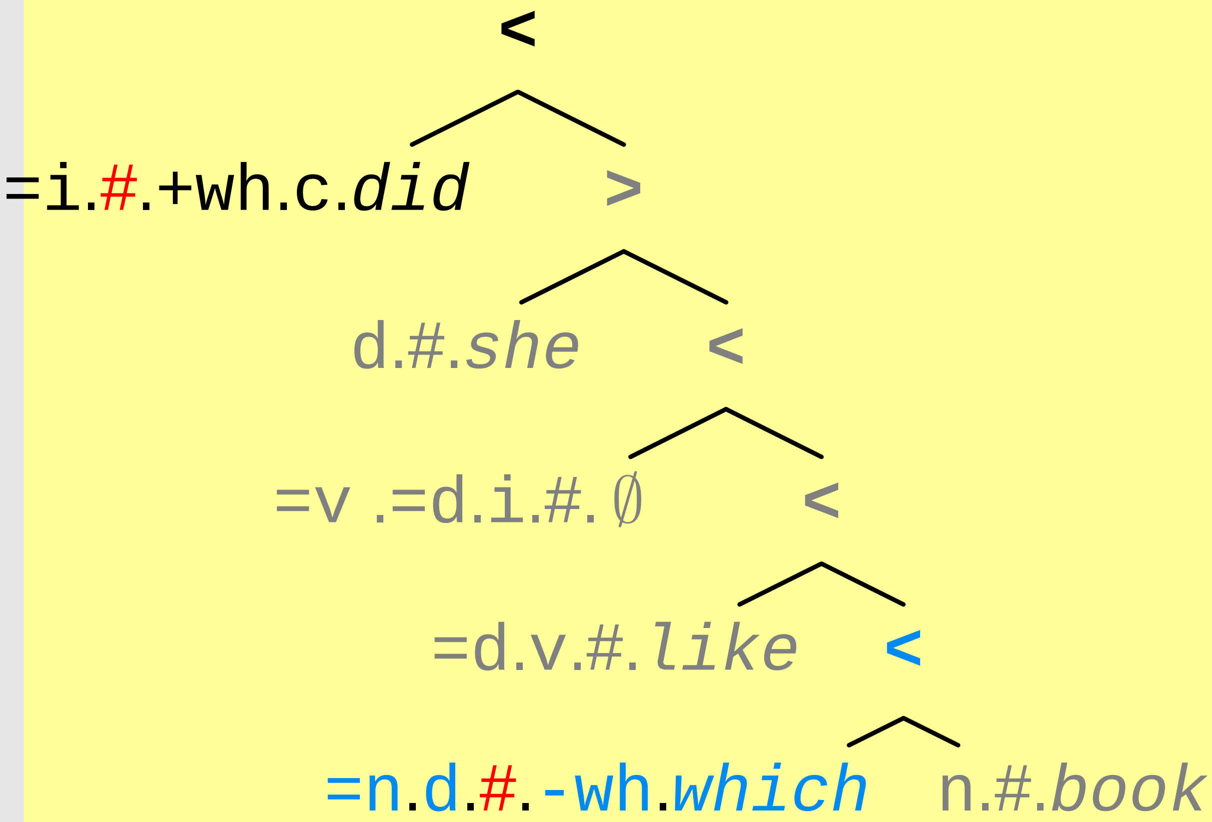


Structure building functions

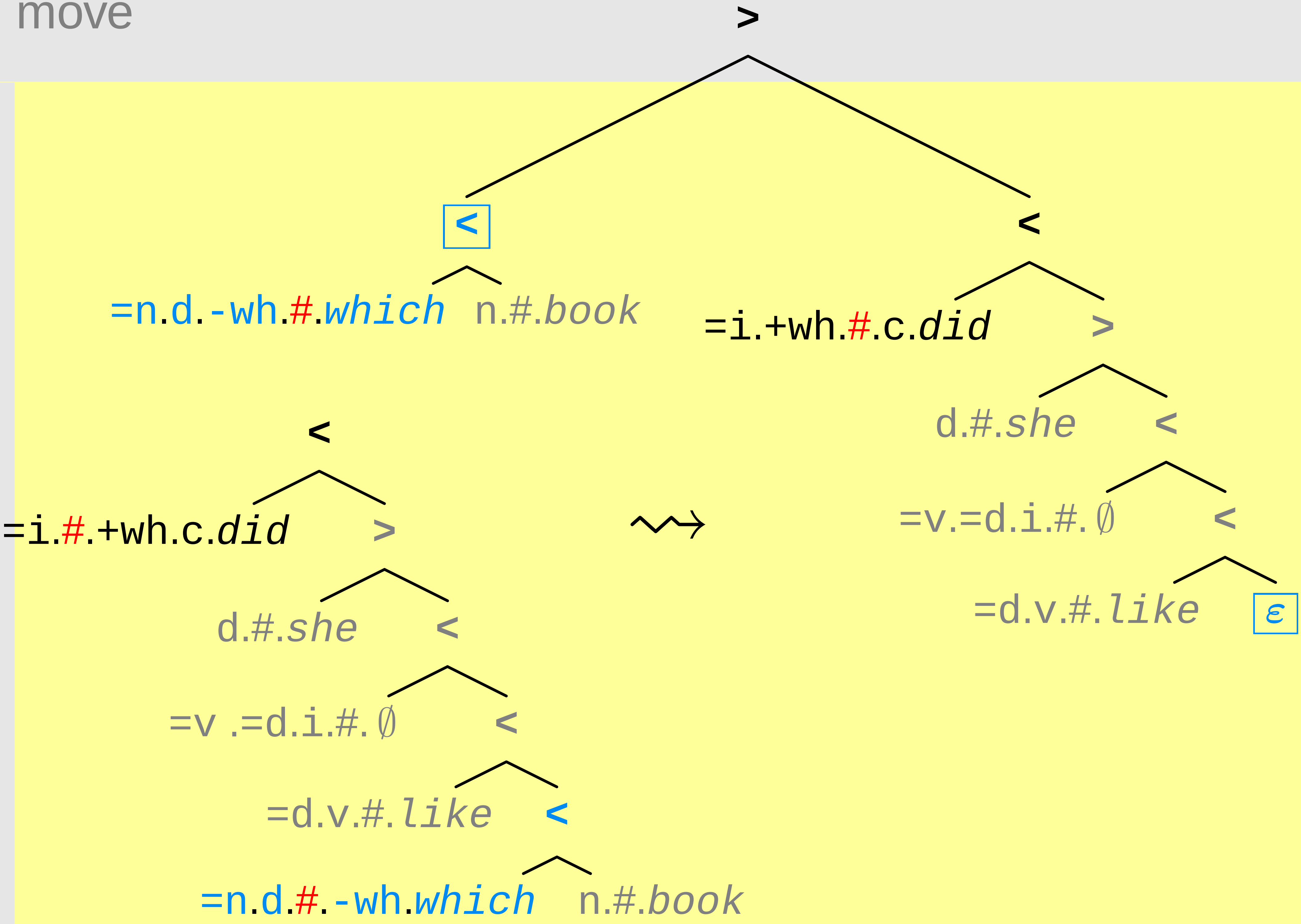
$$\text{move} : \text{Trees} \xrightarrow{\text{part}} 2^{\text{Trees}}$$



move

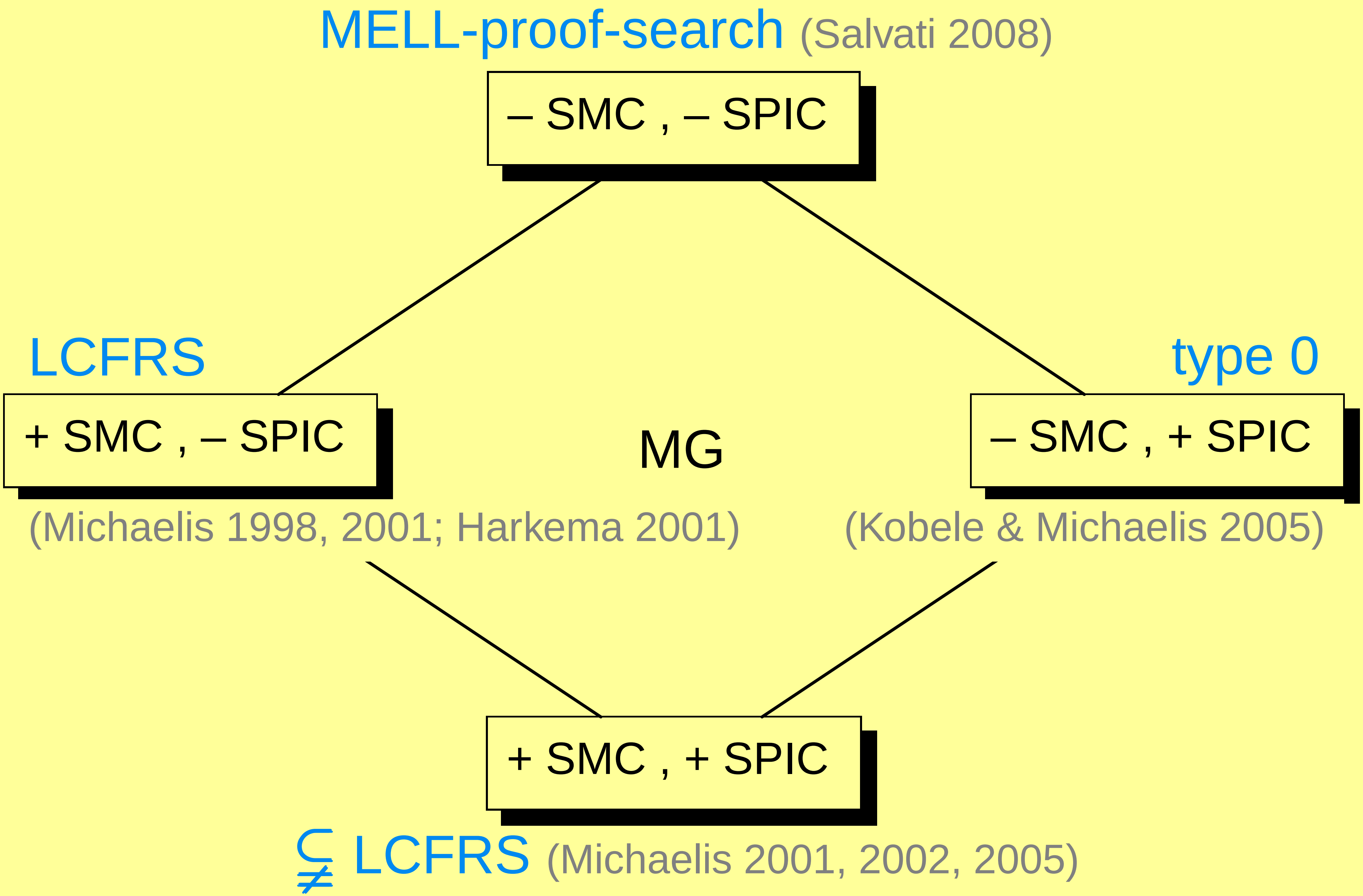


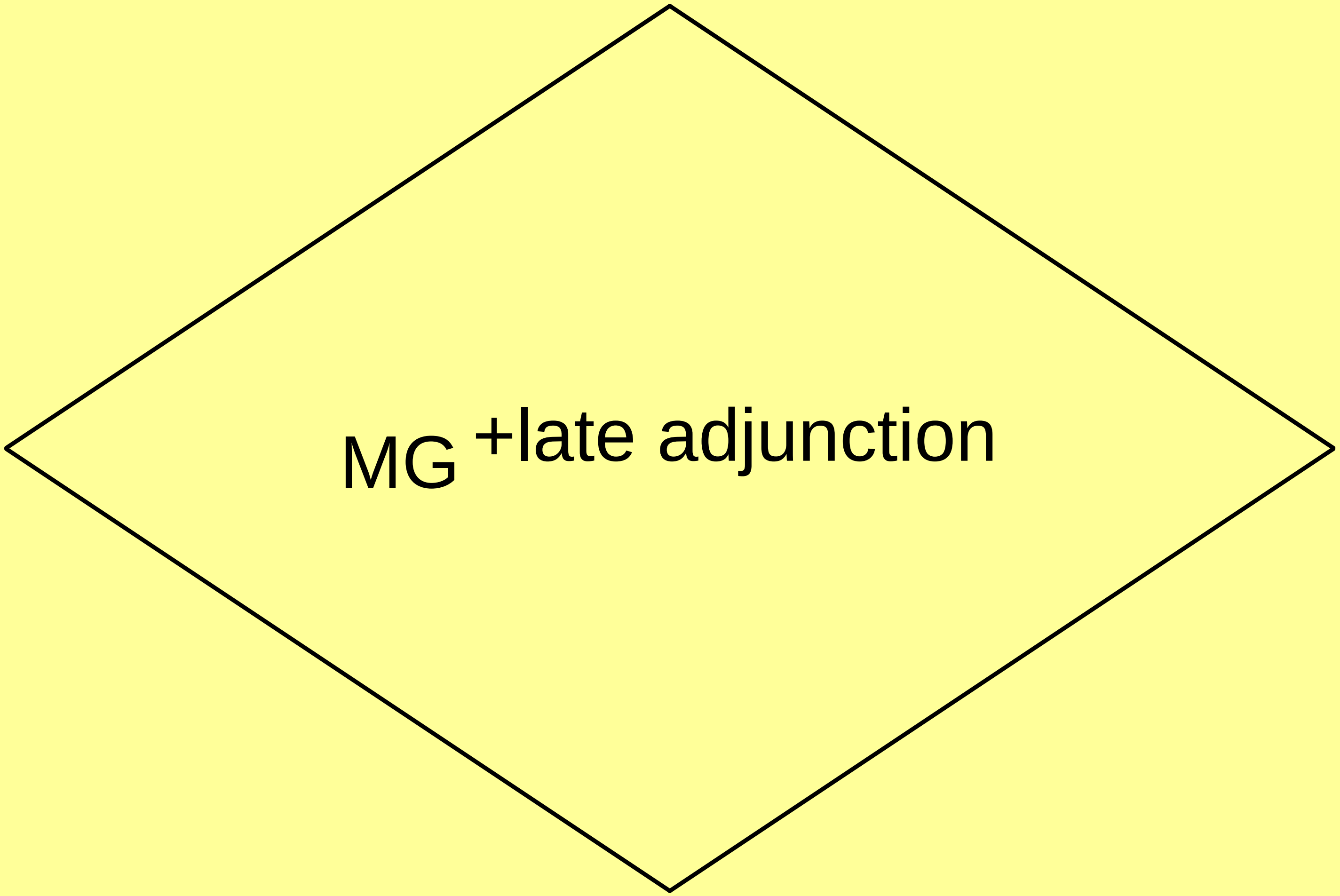
move



>
<
=n.d.-wh.#.which
n.#.book
<
=i.+wh.#.c.did
>
<
d.#.she
<
=i.#.+wh.c.did
>
=v.=d.i.#.∅
<
d.#.she
<
=d.v.#.like
ε
=v .=d.i.#.∅
<
=d.v.#.like
<
=n.d.#.-wh.which
n.#.book

SMC and SPIC — restricting the move-operator domain




MG +late adjunction

Countercyclic adjunction — a “classical” motivation

a) *She_i denied the claim [that $Mary_i$ fell asleep]

b) *She_i liked the book [that $Mary_i$ read]

c) *Which claim [that $Mary_i$ fell asleep] did she_i deny

d) Which book [that $Mary_i$ read] did she_i like

Principle C:

R-expressions like Mary must not be c-commanded by any coindexed constituent which book [that Mary_i read] did she_i like *(derived acyclically)*

$[_{DP}$ which book]

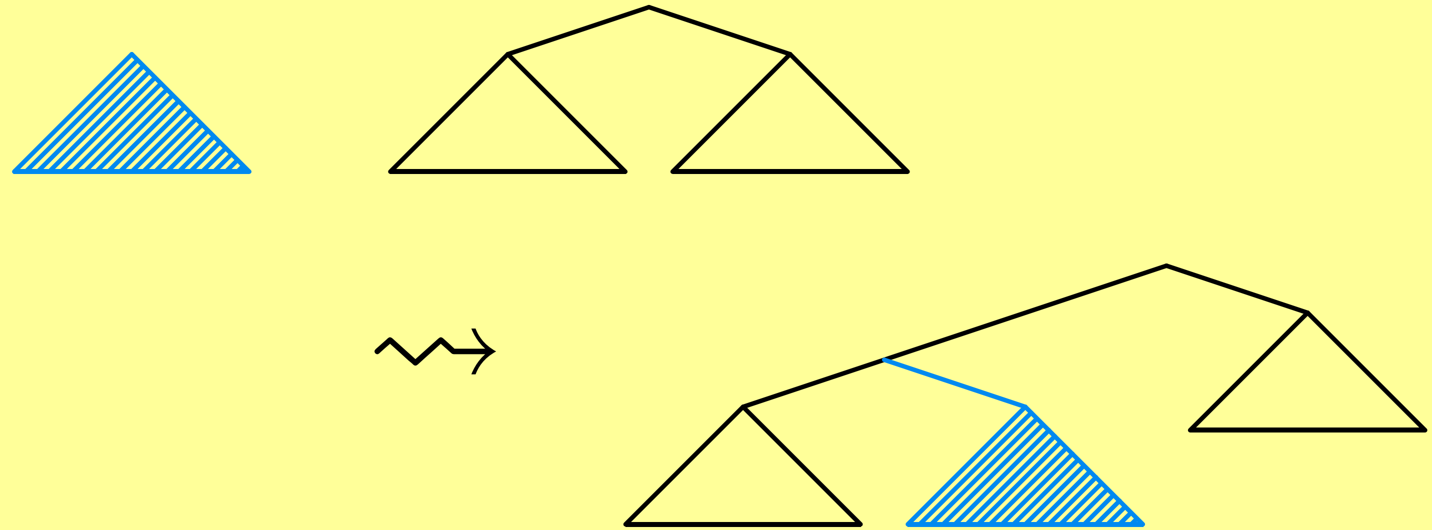
⋮

$[_{C'}$ did $[_{IP}$ she_i $[_{VP}$ like $[_{DP}$ which book]]]]

$[_{CP}$ $[_{DP}$ which book] $[_{C'}$ did $[_{IP}$ she_i $[_{VP}$ like *t*]]]]

$[_{CP}$ $[_{DP}$ [which book] [that Mary_i read]] $[_{C'}$ did $[_{IP}$ she_i $[_{VP}$ like *t*]]]]

Countercyclic Adjunction



- Adjunction is a variant of merge.
- Late adjunction allows this kind of merge countercyclically inside a tree, wherever there is an “adjunction site” of the right category.
- Incorporating late adjunction into the MG(+SMC)-formalism has a very desirable effect: multiple extraposition can be captured.

Extraposition: example

[nur diejenigen Aufsätze t_k] hat [jeder t_j] gelesen

only those papers has everyone read

[der den Kurs besuchte]$_j$ [die sich mit Adjunktion beschäftigen]$_k$

who the class visited which REFL with adjunction deal

Only those papers which deal with adjunction did everyone who visited the class read.

Extraposition: example

[nur diejenigen Aufsätze t_k] hat [jeder t_j] gelesen

only those papers has everyone read

[der den Kurs besuchte $]_j$ [die sich mit Adjunktion beschäftigen $]_k$

who the class visited which REFL with adjunction deal

Only those papers which deal with adjunction did everyone who visited the class read.

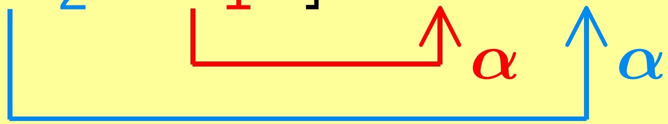
- Roughly comparable to:

[$_{CP}$ [[[a man t_i t_j] came in] [with blond hair $]_i$] [who was laughing $]_j$]

Extraposition and MGs

- Problem: SMC-violation

  $* \;[_{CP} \quad CP_2 \;\; CP_1 \;]$



  Two features of same kind displayed silmutaneously block movement

- Derivational way out:

$[_{CP} \quad CP_1^{\alpha} \;]$		start here
$[_{CP} \quad \_\_ \;]$	$CP_1^{\not\alpha}$	move CP_1, check α
$[_{CP} \quad CP_2^{\alpha} \;\; \_\_ \;]$	$CP_1^{\not\alpha}$	late adjoin CP_2
$[_{CP} \quad \_\_ \;\; \_\_ \;]$	$CP_1^{\not\alpha} \;\; CP_2^{\not\alpha}$	move CP_2, check α

Extraposition and MGs

- Adjoining adjuncts lately which allow subsequent extraction opens up the possibility of "bypassing" the SMC.

- But, we will not treat extraposition by means of the move-operator introduced earlier.

  Instead, we formally employ the scramble-operator introduced in Frey & Gärtner 2002.

Syntactic features

- Different structure building operations are triggered by different types of syntactic features.

(basic) categories:	x , y , z , . . .
(merge-) selectors:	=x , =y , =z , . . .
(move-) licensees:	-x , -y , -z , . . .
(move-) licensors:	+x , +y , +z , . . .
a(djoin)-selectors:	≈x , ≈y , ≈z , . . .
s(cramble)-licensees:	∼x , ∼y , ∼z , . . .

Structure building functions

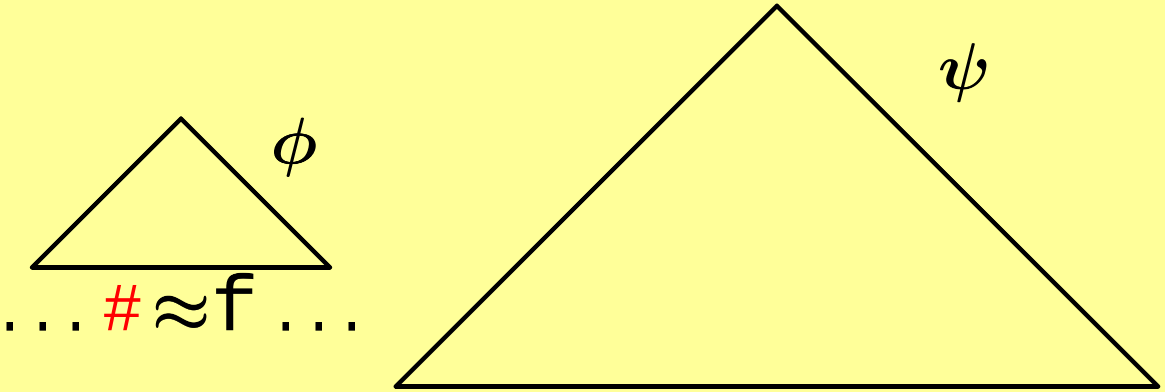
adjoin : Trees $\times$ Trees $\xrightarrow{\text{part}}$ 2^{Trees}

- $\langle \phi, \psi \rangle \in$ Domain(adjoin) $:\Longleftrightarrow$

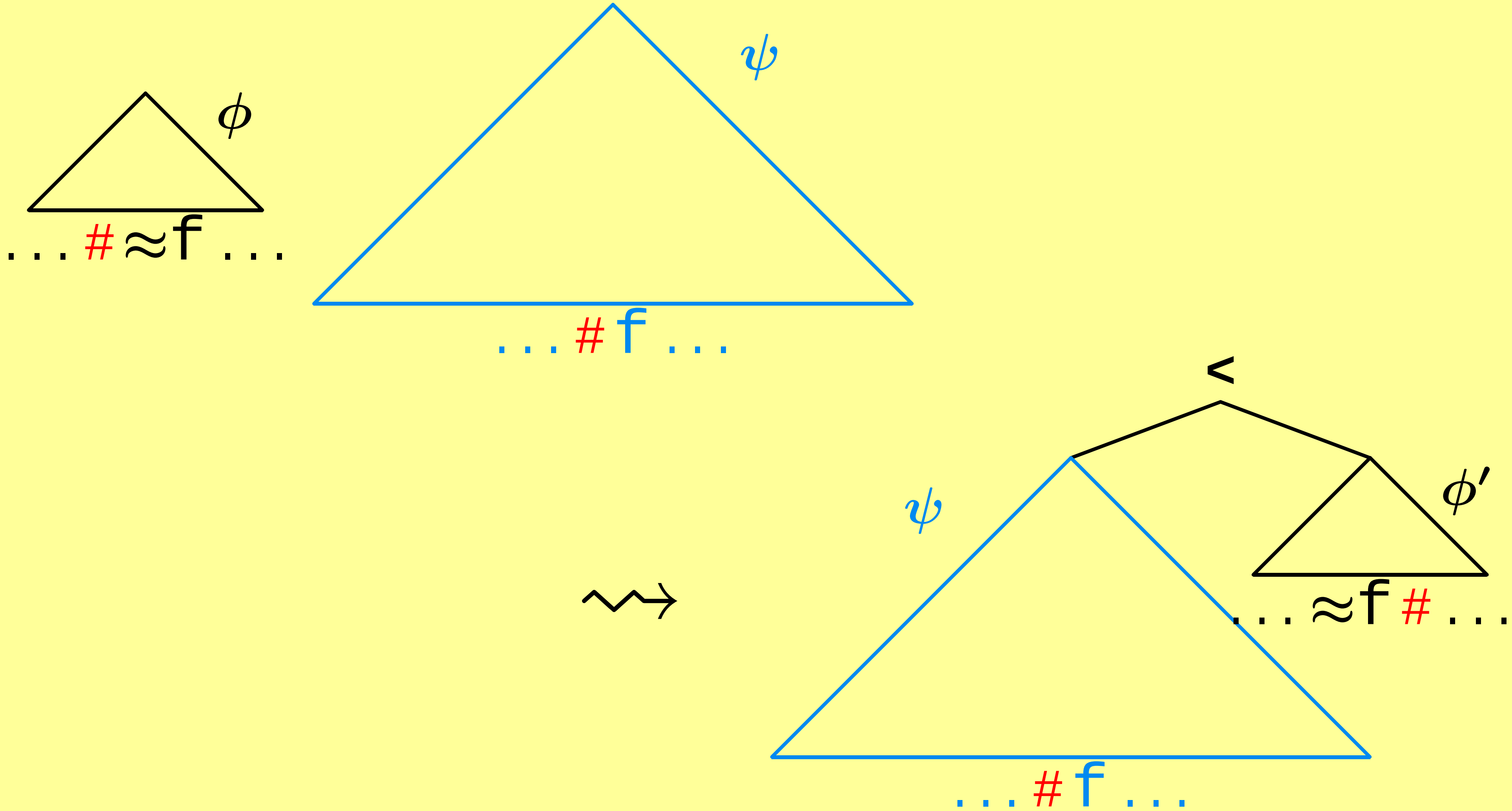
  - the head-label of ψ is of the form `...#≈f...`
  - the head-label of ϕ is of the form `...#f...` or (— *not exclusively* —) there is a maximal projection χ within ϕ whose head-label is of the form `...f...#...`

Distinguish two cases of adjunction

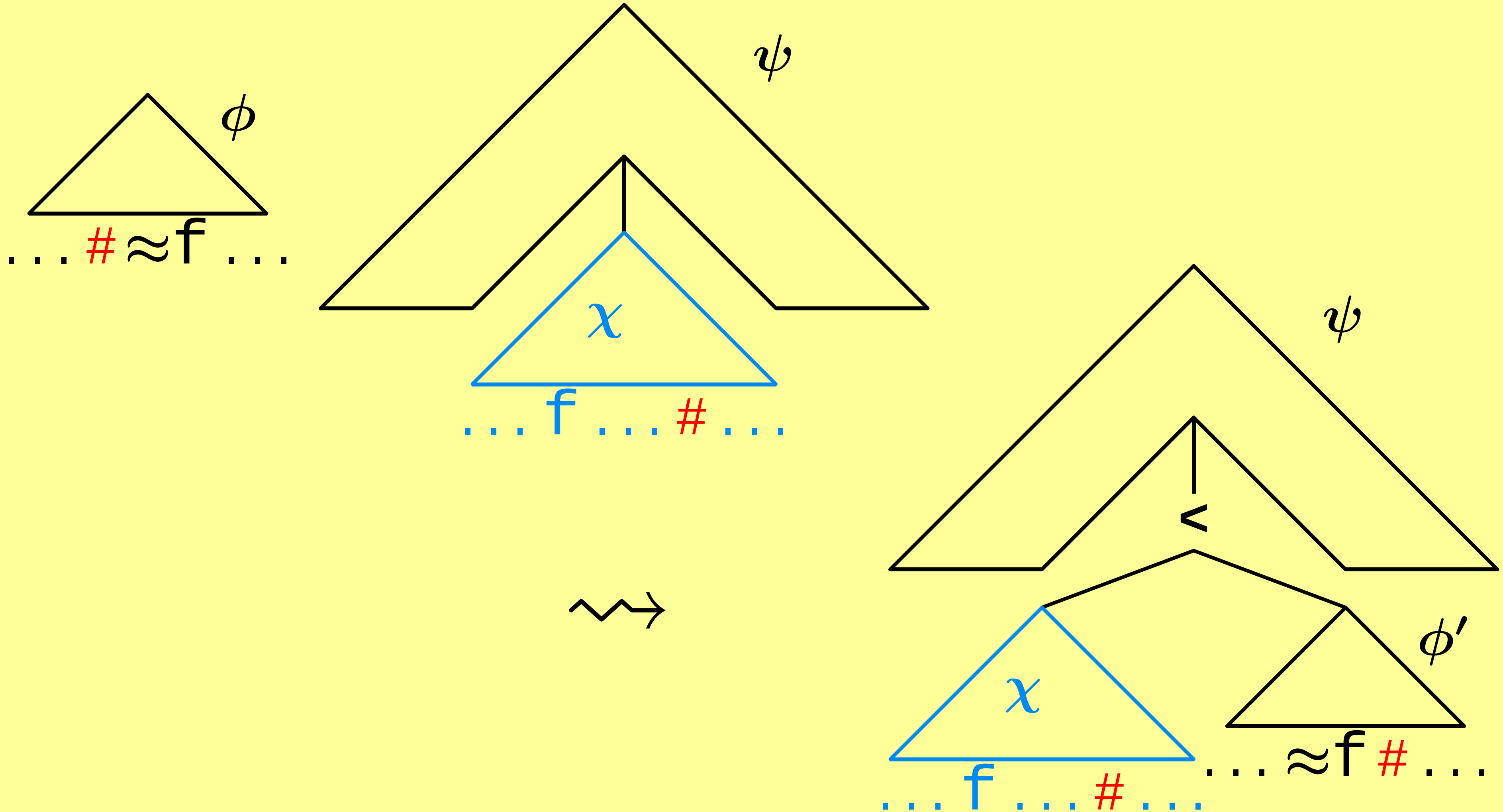
$$\text{adjoin} : \text{Trees} \times \text{Trees} \xrightarrow{\text{part}} 2^{\text{Trees}}$$



$$\text{adjoin} : \text{Trees} \times \text{Trees} \xrightarrow{\text{part}} 2^{\text{Trees}}$$



$$\text{adjoin} : \text{Trees} \times \text{Trees} \xrightarrow{\text{part}} 2^{\text{Trees}}$$

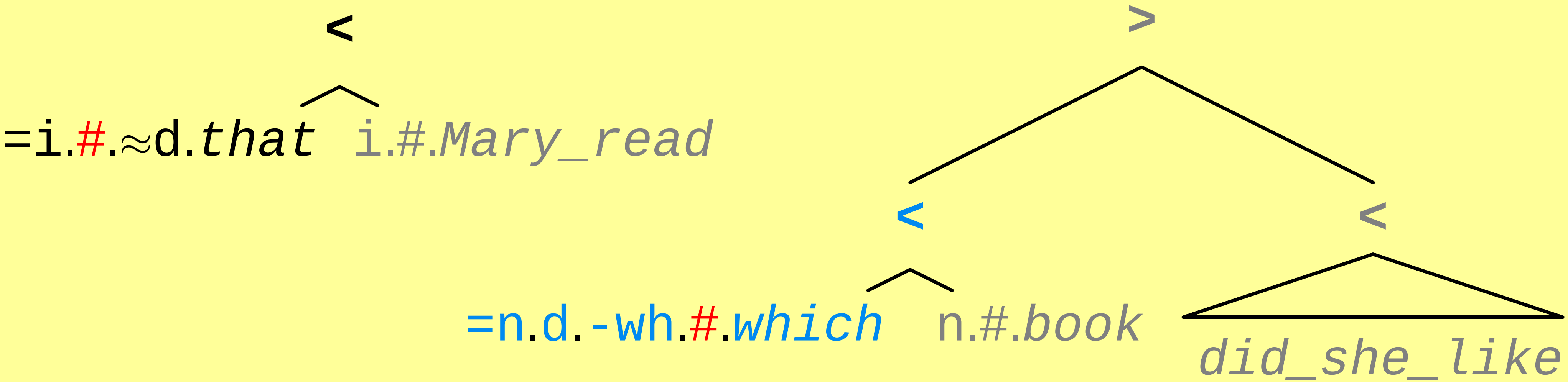


(late) adjunction

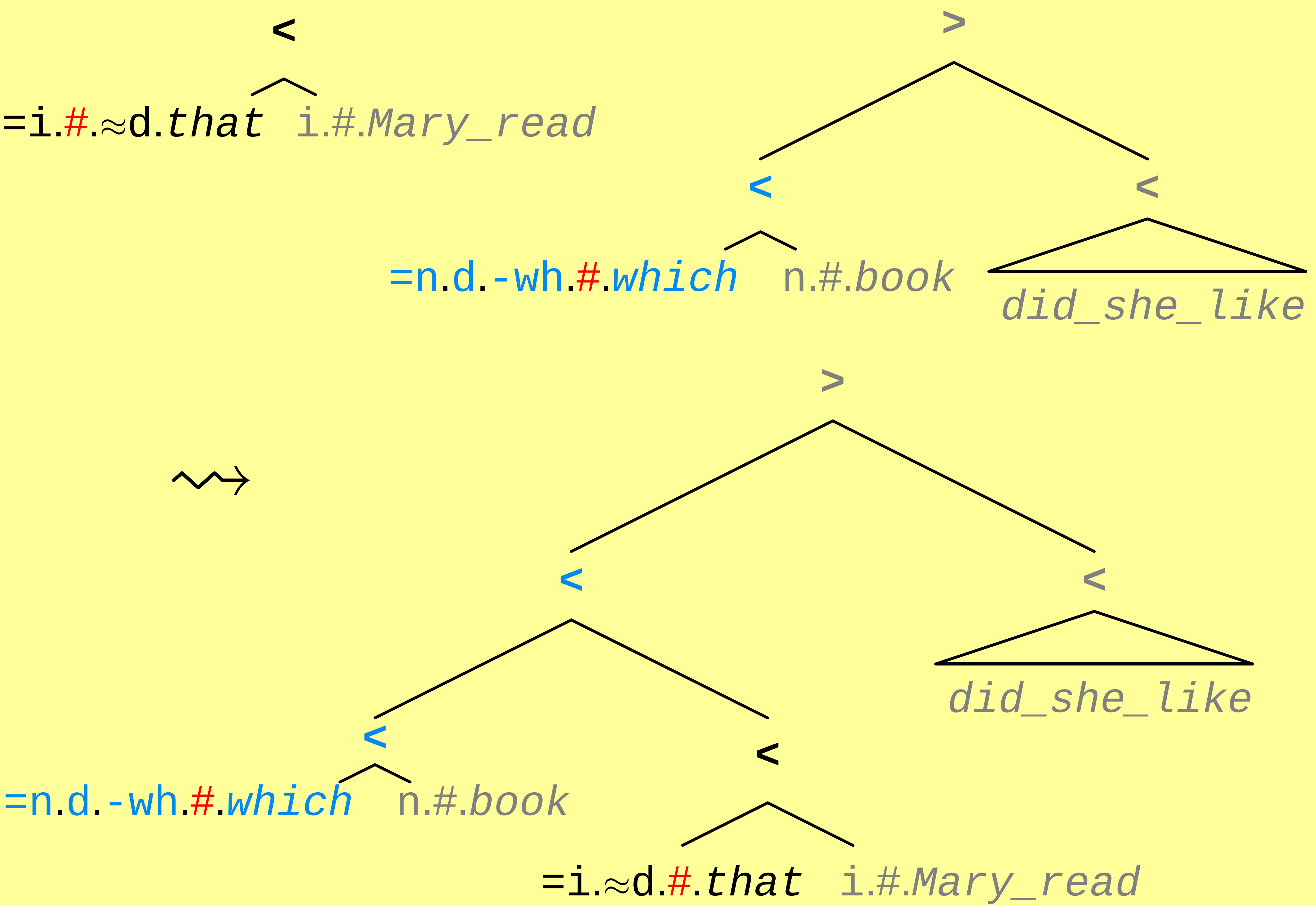
<

=i.#.≈d.*that*   i.#.*Mary_read*

(late) adjunction



(late) adjunction



Structure building functions (Frey & Gärtner 2002)

scramble : Trees $\xrightarrow{\text{part}}$ 2^{Trees}

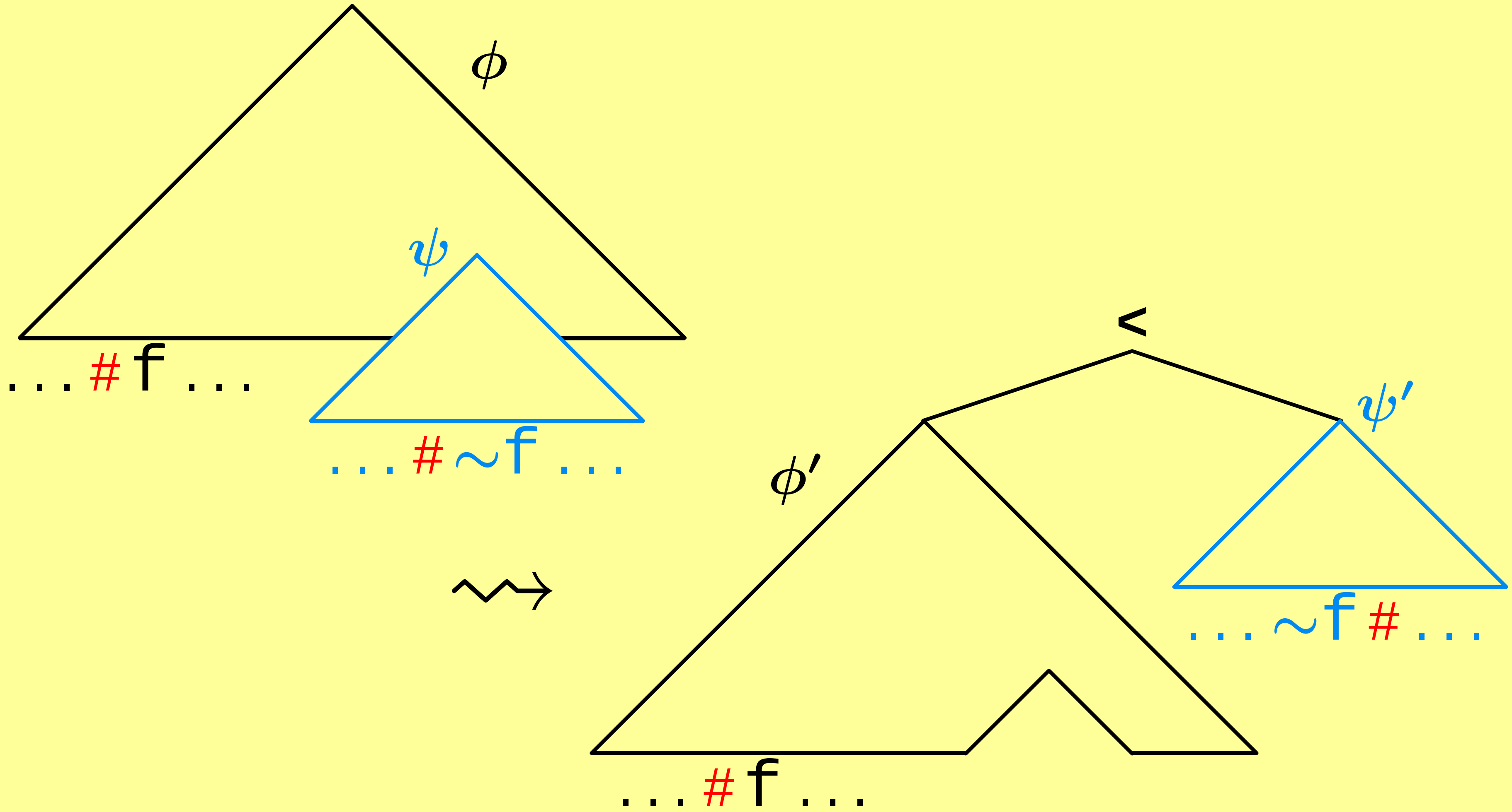
- $\phi \in$ Domain(scramble) $:\Longleftrightarrow$
  - ϕ displays feature $\mathtt{f} \in$ Base
  - there is a (unique [SMC]) maximal projection ψ within ϕ that displays feature $\sim\mathtt{f} \in$ S-Licensees

- scramble($\phi[\ldots]$) $=$

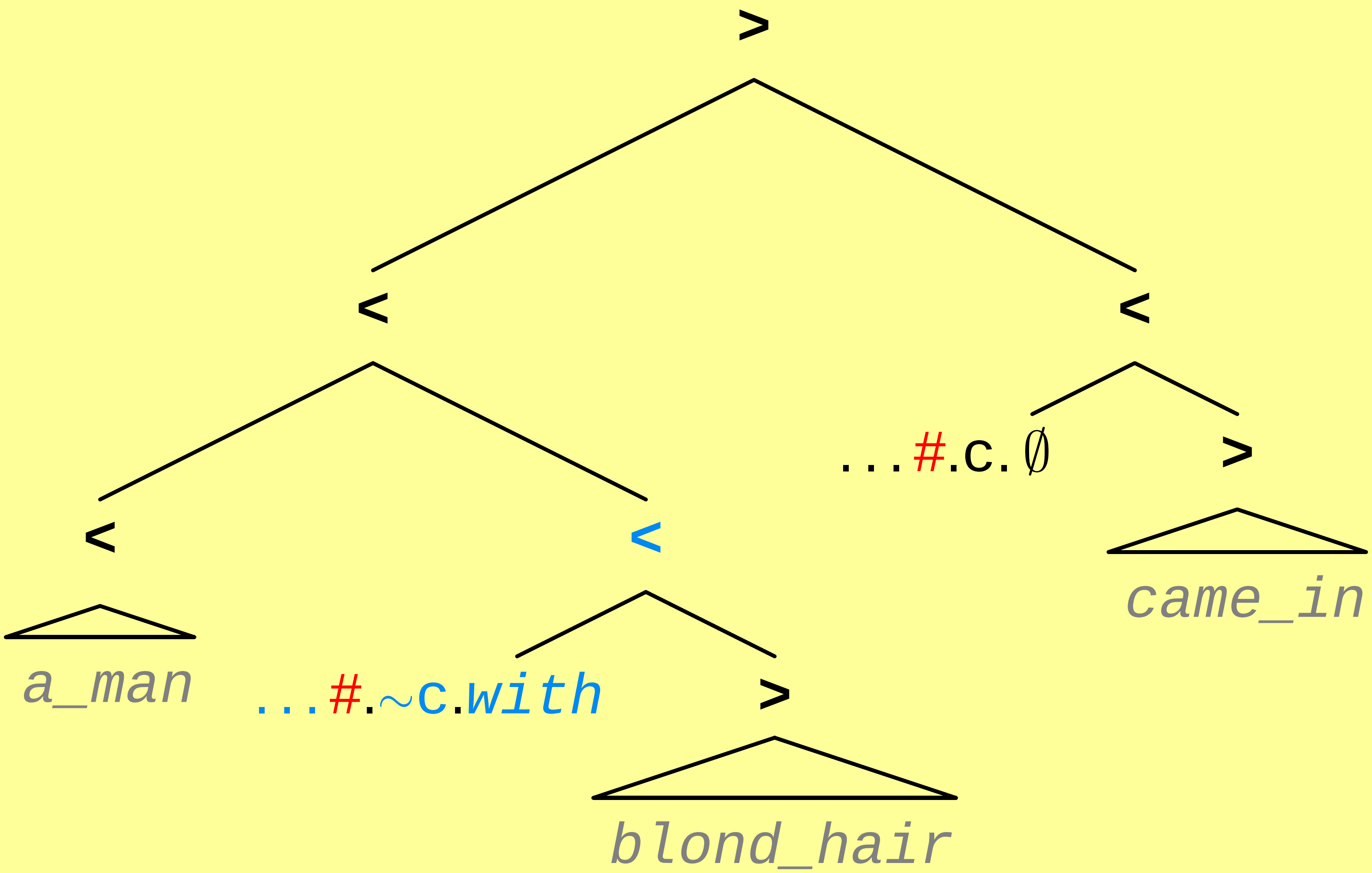
<

$\phi[\ldots]\{\psi[\ldots\#\sim\mathtt{f}\ldots] \longmapsto \varepsilon\}$ $\quad$ $\psi[\ldots\sim\mathtt{f}\#\ldots]$

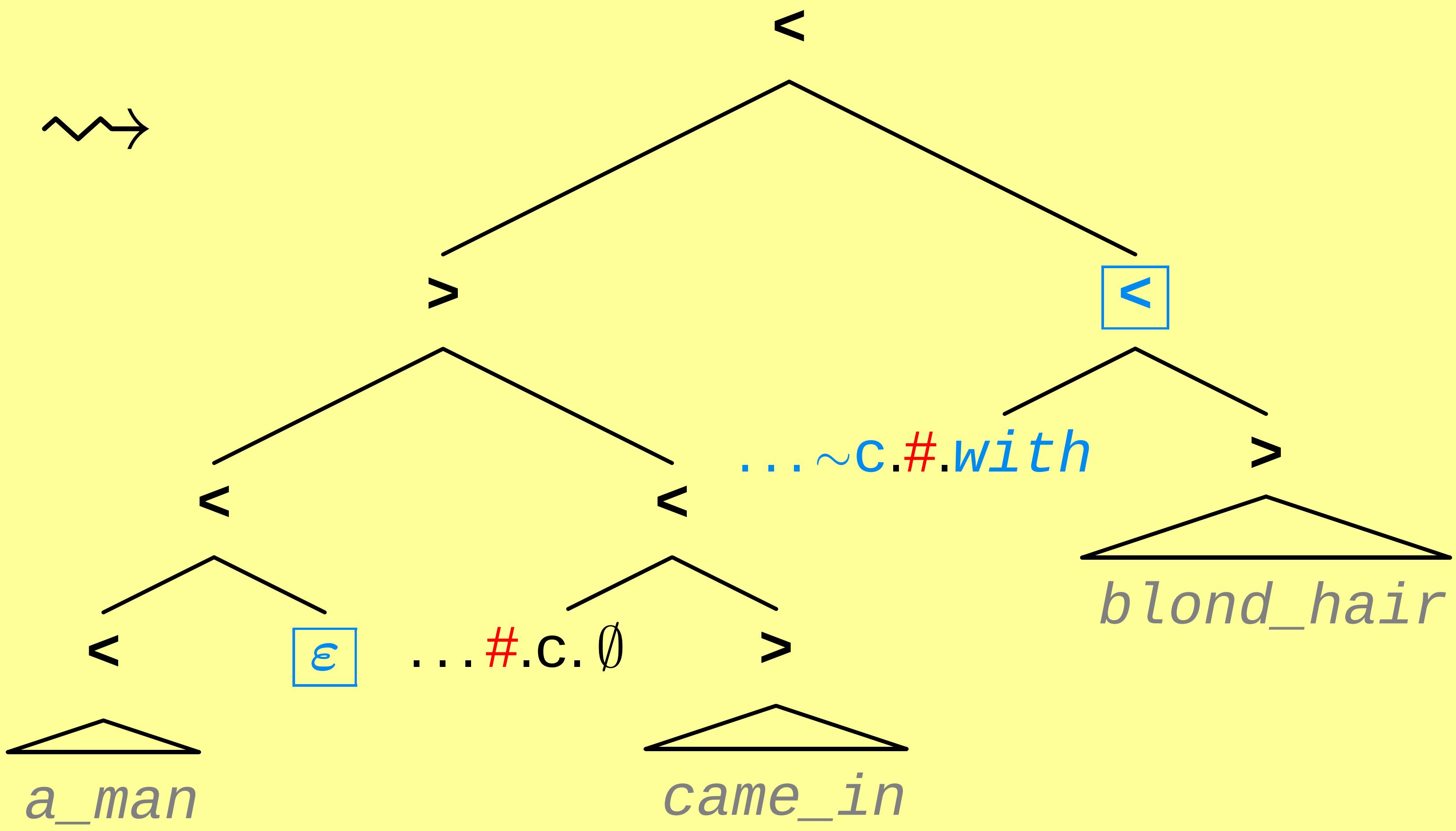
$$\text{scramble} : \text{Trees} \xrightarrow{\text{part}} 2^{\text{Trees}}$$



extraposition



extraposition



Generative capacity: MGs vs. MGs$^{\text{+late adjunction}}$

- $\mathfrak{ML}$(-adjunction) $=$ $\mathfrak{ML}$(+cyclic adjunction)
- $\mathfrak{ML}$(-adjunction) $\stackrel{?}{=}$ $\mathfrak{ML}$(+generalized adjunction)

Generative capacity: MGs vs. MGs$^{\text{+late adjunction}}$

- $\mathcal{ML}(\text{- adjunction}) = \mathcal{ML}(\text{+ cyclic adjunction})$
- $\mathcal{ML}(\text{- adjunction}) \stackrel{?}{=} \mathcal{ML}(\text{+ generalized adjunction})$
- $\mathcal{MG}(\text{+ cyclic adjunction}) \stackrel{?}{=} \mathcal{MG}(\text{+ generalized adjunction})$

Generative capacity: MGs vs. MGs$^{\text{+late adjunction}}$

- $\mathcal{ML}$(-adjunction) $=$ $\mathcal{ML}$(+cyclic adjunction)

- $\mathcal{ML}$(-adjunction) $\stackrel{?}{=}$ $\mathcal{ML}$(+generalized adjunction)

- $\mathcal{MG}$(+cyclic adjunction) $\stackrel{?}{=}$ $\mathcal{MG}$(+generalized adjunction)

  No difference between cyclic (“earliest”) adjunction and late adjunction as long as the adjuncts do not introduce unchecked instances of licensees that allow subsequent extraction.

Generative capacity: MGs vs. MGs$^{+\text{late adjunction}}$

- $\mathcal{ML}$(- adjunction) $=$ $\mathcal{ML}$(+ cyclic adjunction)

- $\mathcal{ML}$(- adjunction) $\stackrel{?}{=}$ $\mathcal{ML}$(+ generalized adjunction)

- $\mathcal{MG}$(+ cyclic adjunction) $\stackrel{?}{=}$ $\mathcal{MG}$(+ generalized adjunction)

  No difference between cyclic (“earliest”) adjunction and late adjunction as long as the adjuncts do not introduce unchecked instances of licensees that allow subsequent extraction.

♦ There is an effect on what can be called the derivational generative capacity in the sense of Becker et al. 1992.

Generative capacity: MGs vs. MGs$^{\text{+late adjunction}}$

- $\mathcal{ML}$(- adjunction) $=$ $\mathcal{ML}$(+ cyclic adjunction)

- $\mathcal{ML}$(- adjunction) $\stackrel{?}{=}$ $\mathcal{ML}$(+ generalized adjunction)

- $\mathcal{MG}$(+ cyclic adjunction) $\stackrel{?}{=}$ $\mathcal{MG}$(+ generalized adjunction)

  No difference between cyclic (“earliest”) adjunction and late adjunction as long as the adjuncts do not introduce unchecked instances of licensees that allow subsequent extraction.

- There is an effect on what can be called the derivational generative capacity in the sense of Becker et al. 1992.

a man came in with blond hair who was laughing

[a man t_j t_k] came in [with blond hair $]_j$ [who was laughing $]_k$

a man came in with blond hair who was laughing *(derived acyclically)*

⋮

$[_{CP}$ [a man] came in]

$[_{CP}$ [[a man] [with blond hair]] came in]

$[_{CP}$ [[a man t_i] came in] [with blond hair $]_i$]

$[_{CP}$ [[[a man t_i] [who was laughing]] came in] [with blond hair $]_i$]

$[_{CP}$ [[[a man t_i t_j] came in] [with blond hair $]_i$] [who was laughing $]_j$]

a man came in with blond hair who was laughing *(derived cyclically)*

⋮

α.#.~**C** β.#.~**C**

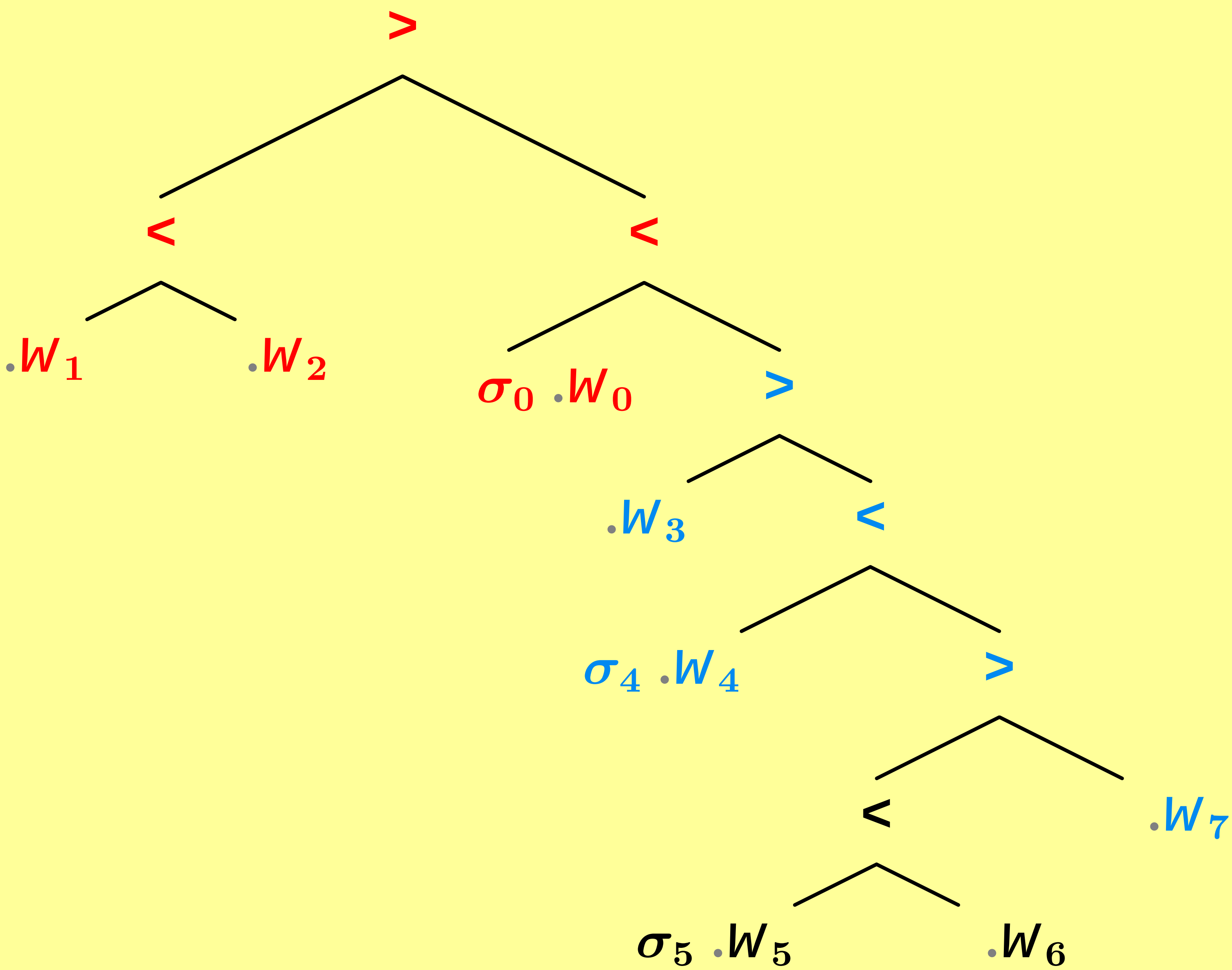
[$_{CP}$ [a man [with blond hair] [who was laughing]] came in]

Complexity of late adjunction

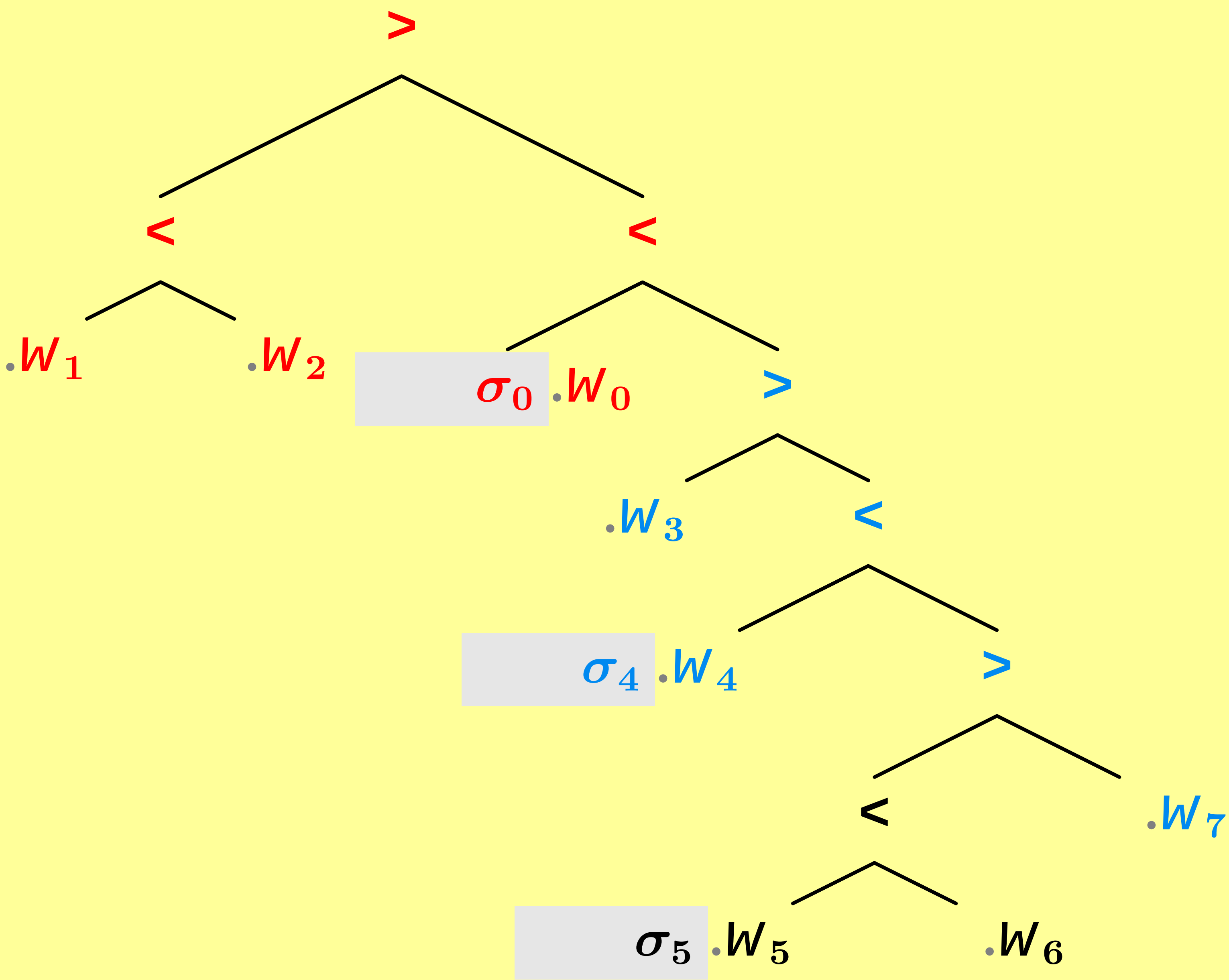
- Abstractly, the problem with late adjunction is that in order to locate the adjunction sites, an a priori not bounded amount of (categorial) information has to be stored during a derivation.

- In fact, this prevents us from directly adopting the methods, in particular,
  - developed to prove that MGs provide a weakly equivalent subclass of LCFRSs (cf. Michaelis 1998), and
  - leading to the succinct, chain-based MG-reformulation presented in Stabler & Keenan 2000 [2003] — reducing “classical” MGs to their “bare essentials.”

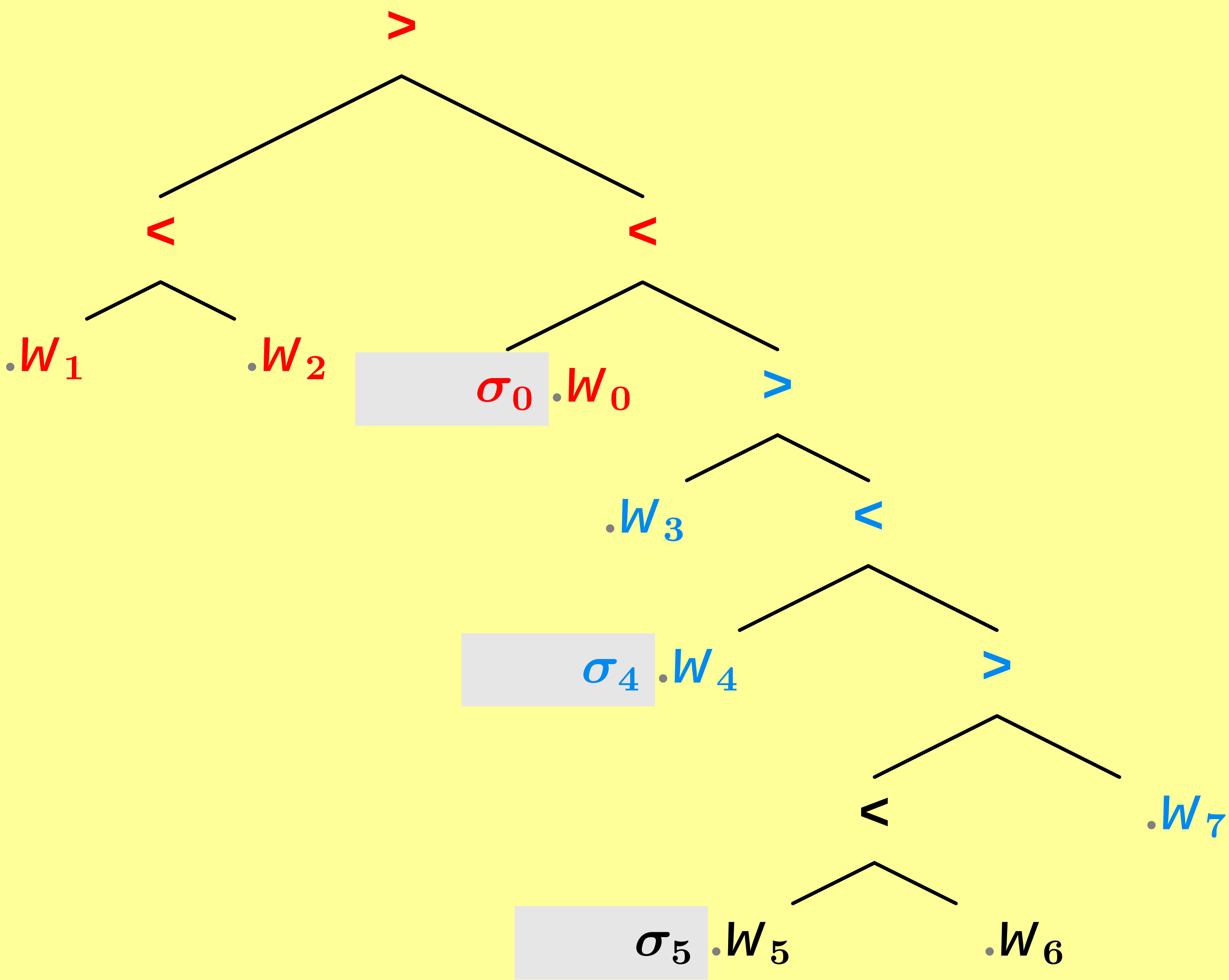
Reducing an MG(+SMC,-late adjunction)



Reducing an MG(+SMC,-late adjunction)

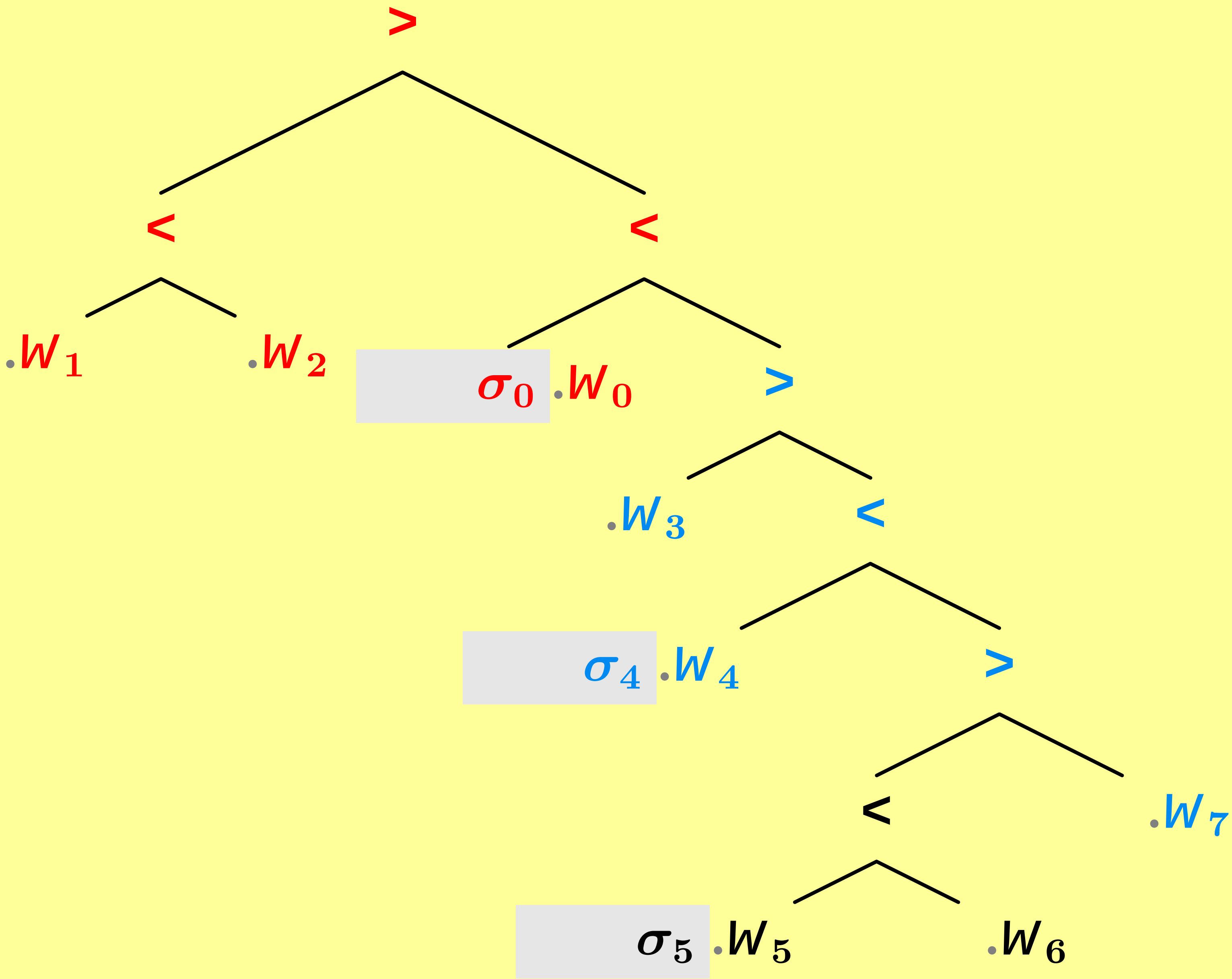


Reducing an MG(+SMC,- late adjunction)



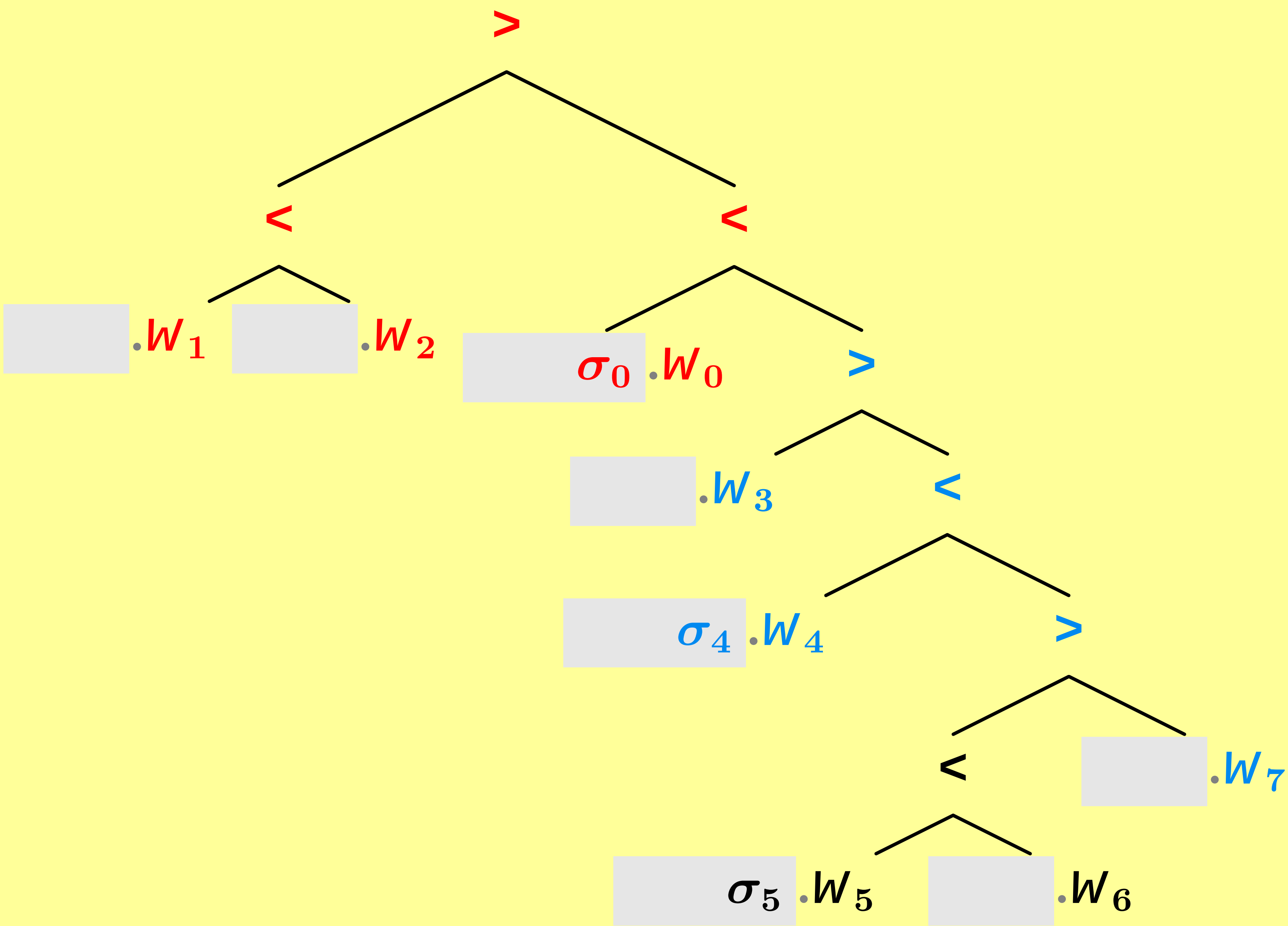
$$\left\langle \sigma_0 \, . \, w_1 w_2 w_0 \; , \; \sigma_4 \, . \, w_3 w_4 w_7 \; , \; \sigma_5 \, . \, w_5 w_6 \right\rangle$$

Reducing an MG(+SMC,-late adjunction)



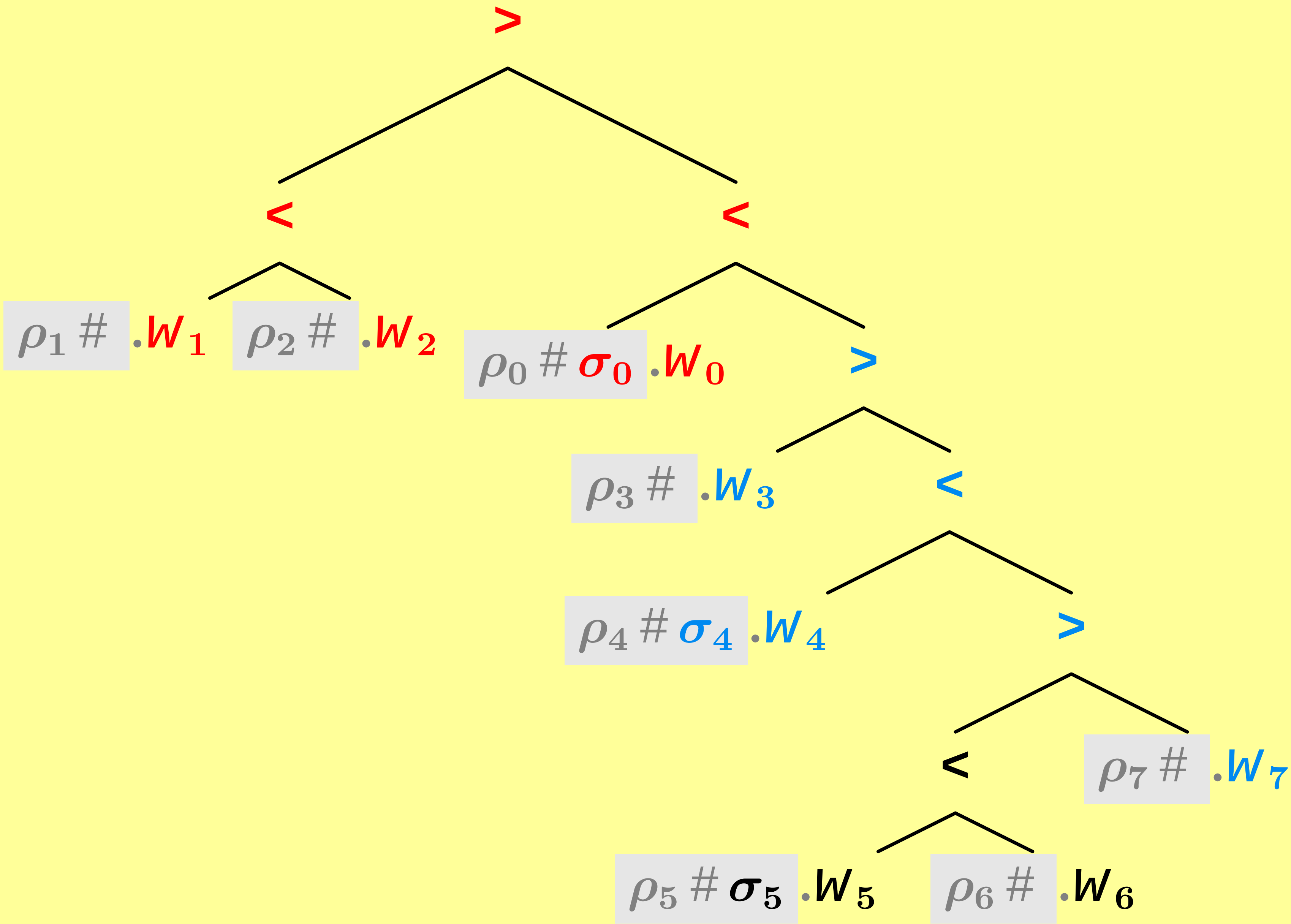
$$\langle \sigma_0 \quad , \quad \sigma_4 \quad , \quad \sigma_5 \rangle$$

Reducing an MG(+SMC,-late adjunction)



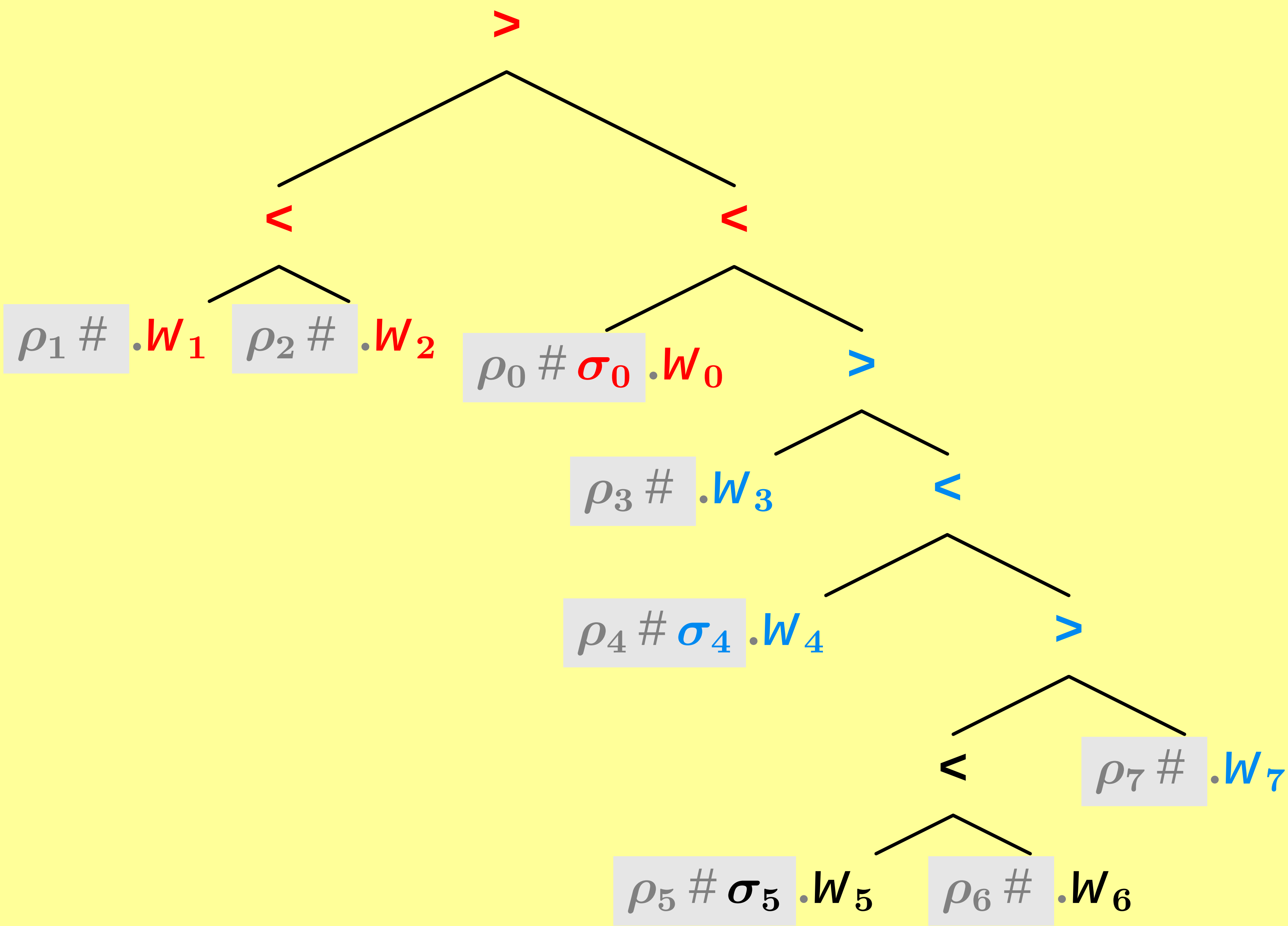
$$\langle \sigma_0 \quad , \quad \sigma_4 \quad , \quad \sigma_5 \rangle$$

Reducing an MG(+ SMC,- late adjunction)



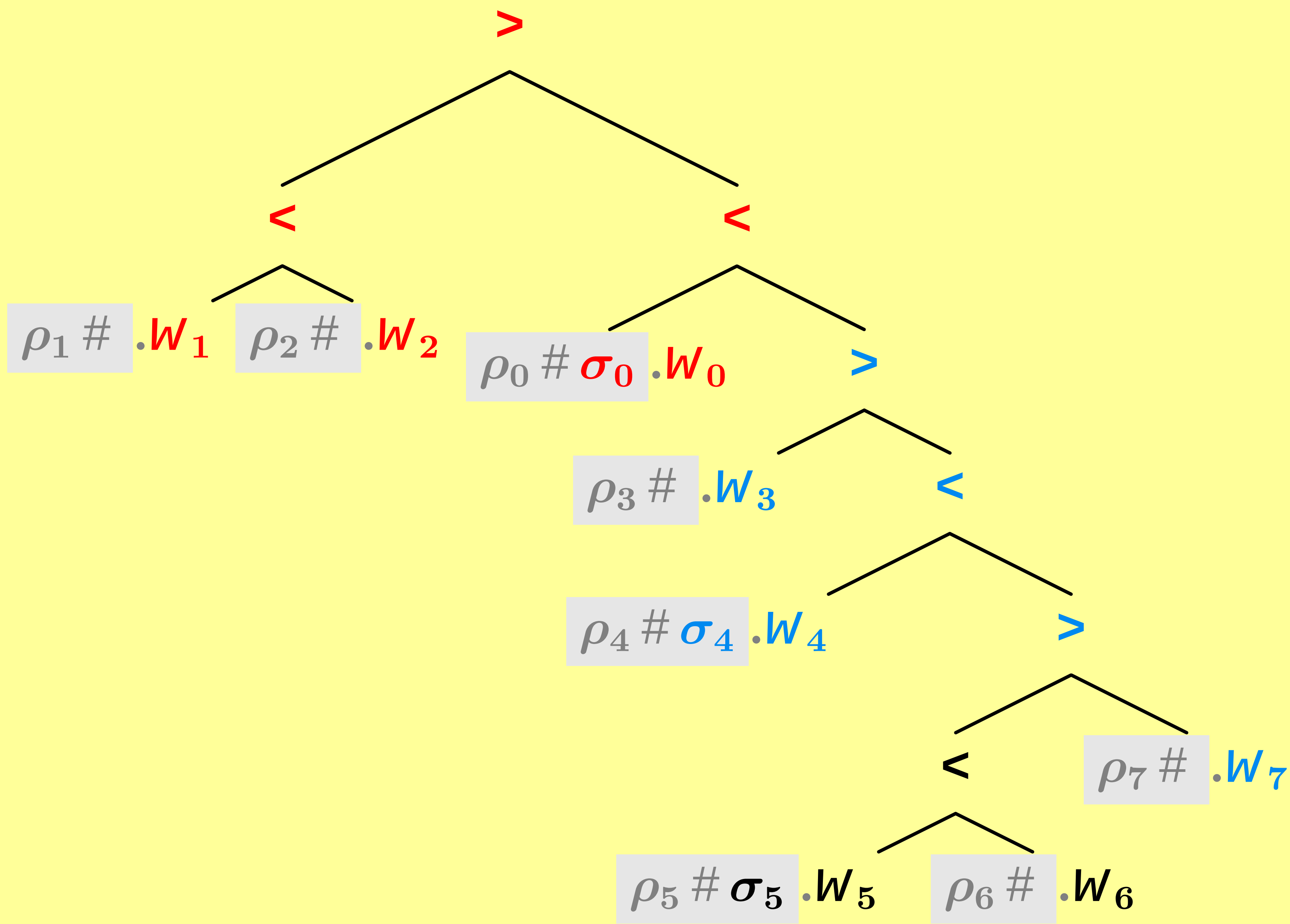
$$\langle \sigma_0 \quad , \quad \sigma_4 \quad , \quad \sigma_5 \rangle$$

Reducing an MG(+SMC,-late adjunction)



$$\langle\ \sigma_0 . w_1 w_2 w_0\ ,\ \sigma_4 . w_3 w_4 w_7\ ,\ \sigma_5 . w_5 w_6\ \rangle$$

Reducing an MG(+SMC,+late adjunction)



$$\left\langle \rho_0\,\#\,\boldsymbol{\sigma}_0\,.w_0,\ \rho_1\,\#\,.w_1,\ \rho_2\,\#\,.w_2,\ \rho_3\,\#\,.w_3,\ \rho_4\,\#\,\boldsymbol{\sigma}_4\,.w_4,\ \rho_5\,\#\,\boldsymbol{\sigma}_5\,.w_5,\ \rho_6\,\#\,.w_6,\ \rho_7\,\#\,.w_7 \right\rangle$$

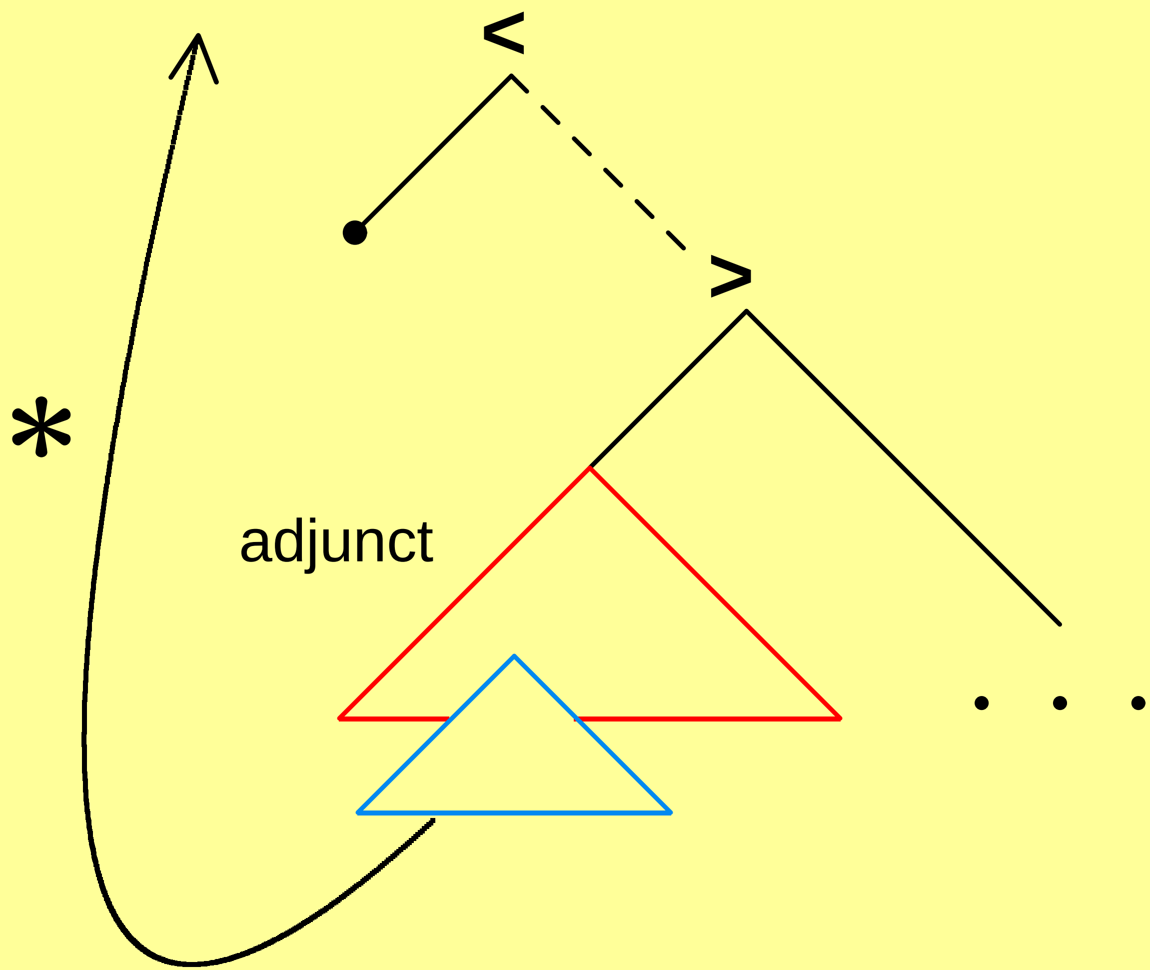
Complexity of late adjunction

- The proof that MGs without late adjunction are mildly context-sensitve rests on the technical possibility of removing checked features from the structures.

  Therefore it is unclear, whether, in general, MGs allowing late adjunction still belong to the same complexity class.

- If, however, the AIC (adjunct island condition) is imposed, we can apply a specific reduction method in proving that for the resulting MGs the old complexity result holds.

- If at all, only full adjuncts but no proper subpart of them can extract.



Complexity of late adjunction

Distinguish three cases

(i) late-adjoined adjuncts and their subtrees cannot extract further,

(ii) late-adjoined adjuncts, but not proper subtrees can extract further,

(iii) late-adjoined adjuncts and their subtrees can extract further.

Complexity of late adjunction

Distinguish three cases

(i) late-adjoined adjuncts and their subtrees cannot extract further,

Solutions allowing a modified method of MG-reducing in order to define a weakly equivalent LCFRS:

(i) (strong equivalent MG-) treatment in terms of cyclic adjunction,

Complexity of late adjunction

Distinguish three cases

(ii) late-adjoined adjuncts, but not proper subtrees can extract further,

Solutions allowing a modified method of MG-reducing in order to define a weakly equivalent LCFRS:

(ii) an additional “0|1-register” for each basic category recording the absence|presence of at least one instance of that category,

Complexity of late adjunction

Distinguish three cases

(iii) late-adjoined adjuncts and their subtrees can extract further.

Solutions allowing a modified method of MG-reducing in order to define a weakly equivalent LCFRS:

(iii) none (?) /* cyclic adjunction-treatment causes SMC-conflict */

MG-diamond — shortest move (SMC) and adjunct islands (AIC)