Incremental Processing

Incremental Semantic Parsing

Referent
Nikolina Koleva

16 June 2011

Motivation

Why incremental **semantic** parsing?

- more suitable for slot filling
- enables modeling dialogue phenomena
  - *shared utterances, ellipsis, alignment*

☛ for dialogue systems

Overview

1. RUBISC's Incremental Chunkig

    1.1 Regular Grammar for Semantics

    1.2 Unification

    1.3 Robustness

    1.4 Incrementality

    1.5 End-of-Sentence Detection

    1.6 Evaluation

2. Incremental Parsing, or Incremental Grammar?

    2.1 Parsing vs. Generation

    2.2 Contextual Model

    2.3 Shared Utterances

    2.4 Cross-Speaker Ellipsis

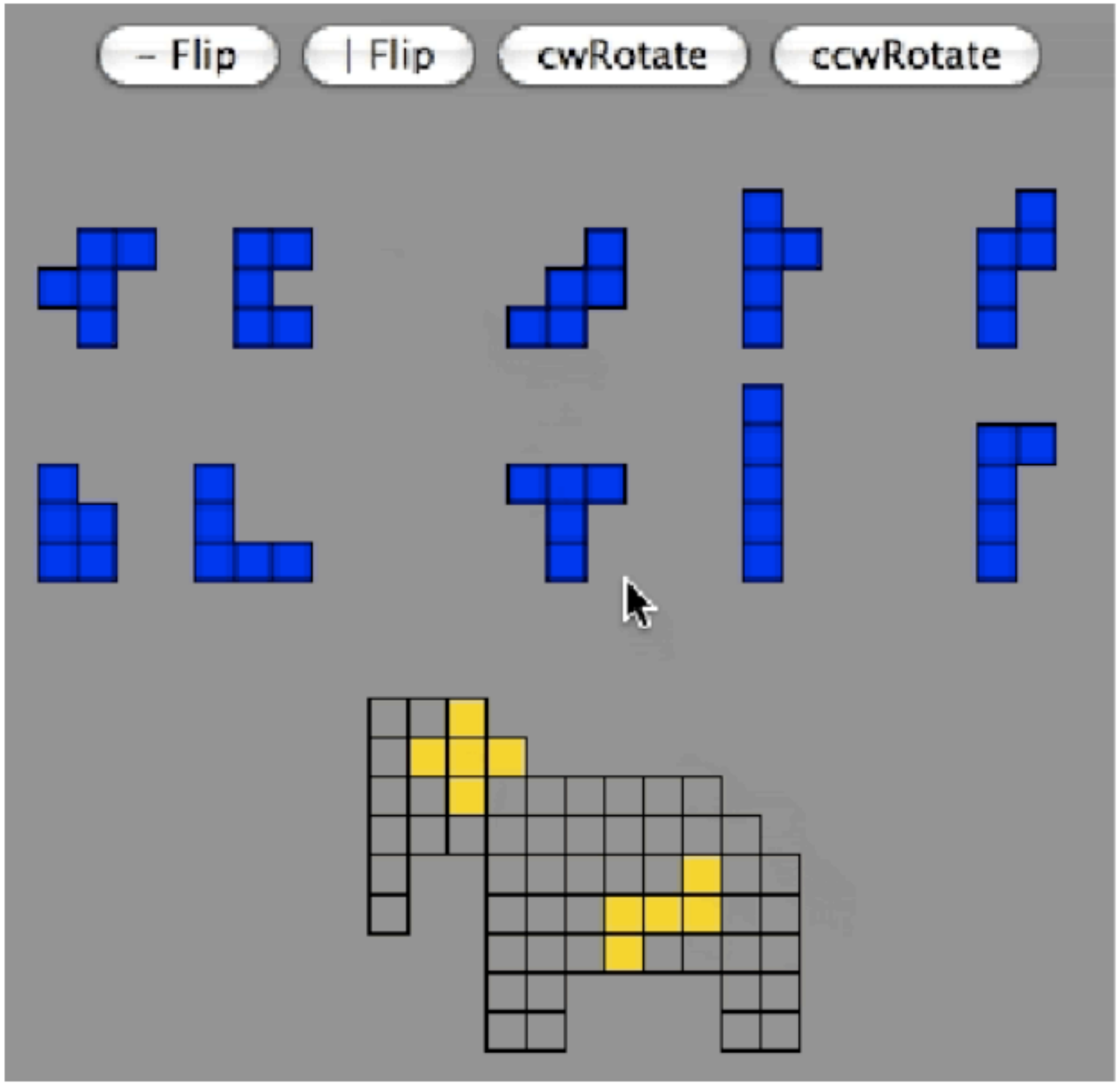
    2.5 Alignment

RUBISC

a **R**obust **U**nification-**B**ased **I**ncremental **S**emantic **C**hunker

- incremental slot filling
- chunks based on semantic content rather than syntax
- partial or complete semantics during the incoming speech
- evaluated on German speech corpus

RUBISC's Domain



- Actions

  turn, *flip*, *move*, *grasp*

- Objects

  cross, *l* etc.

- End positions

  head, *back*, *leg* etc.

Semantic Slots & Grammar Rules

@: action — @: entity: xpos

@: entity: name — @: entity: ypos — @: end

action: grasping, end: empty → nimm | nehme *{take}*

action: turning → drehe? *{turn}*

action: flipping → spig(le | el) *{flip}*

action: movement → bewegt *{moved}*

entity: name: x → kreuz | plus | ((das | ein) x$) *{cross | plus (the | a) x}*

entity: name: w → treppe | ((das | ein) w$) *{staircase | (the | a) w}*

end: head → (in | an) den kopf *{(on | in) the head}*

end: leg2 → ins? das (hinterbein | hintere bein | rechte bein | zweites bein) *{in the hind leg | back leg | right leg | second leg}*

entity: ypos: 1 → der (unteren | zweiten) reihe *{(lower | second) row}*

entity: xpos: 1 → das erste *{the first}*

entity: xpos: -1 → das letzte *{the last}*

end: horizontal, action: flipping → horizontal *{horizontally}*

Semantic Slots & Grammar Rules

@: action | @: entity: xpos
@: entity: name | @: entity: ypos | @: end

action: grasping, end: empty → nimm | nehme *{take}*

action: flipping → spieg(le | el) *{flip}*

action: turning → drehe? *{turn}*

action: movement → bewegt *{moved}*

entity: name: x → kreuz | plus | ((das | ein) x$) *{cross | plus (the | a) x}*

entity: name: w → treppe | ((das | ein) w$) *{staircase | (the | a) w}*

end: head → (in | an) den kopf *{(on | in) the head}*

end: leg2 → ins? das (hinterbein | hintere bein | rechte bein | zweites bein)
{in the hind leg | back leg | right leg | second leg}

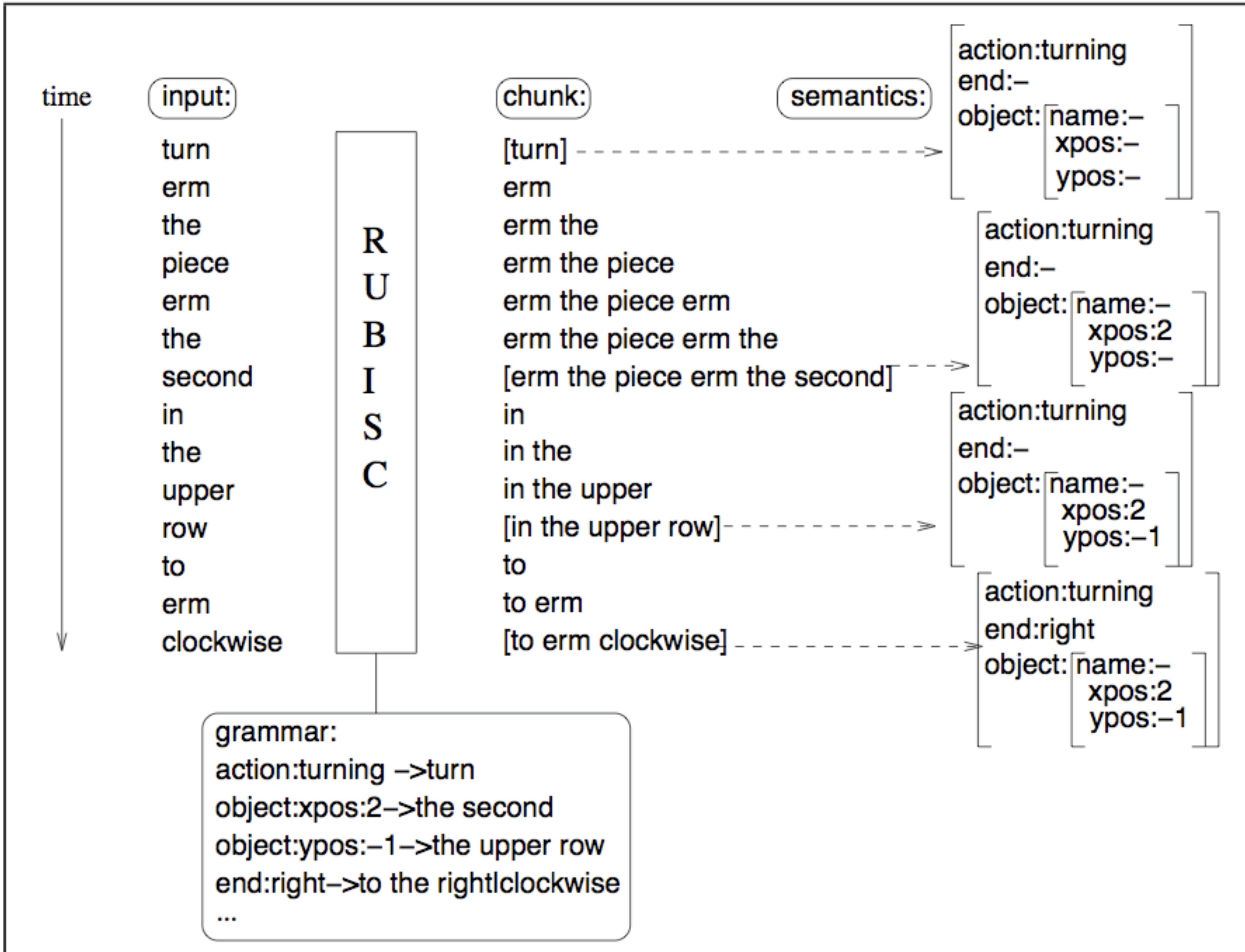
entity: ypos: 1 → der (unteren | zweiten) reihe *{(lower | second) row}*

entity: xpos: 1 → das erste *{the first}*

entity: xpos: -1 → das letzte *{the last}*

end: horizontal, action: flipping → horizontal *{horizontally}*

Incremental Chunking



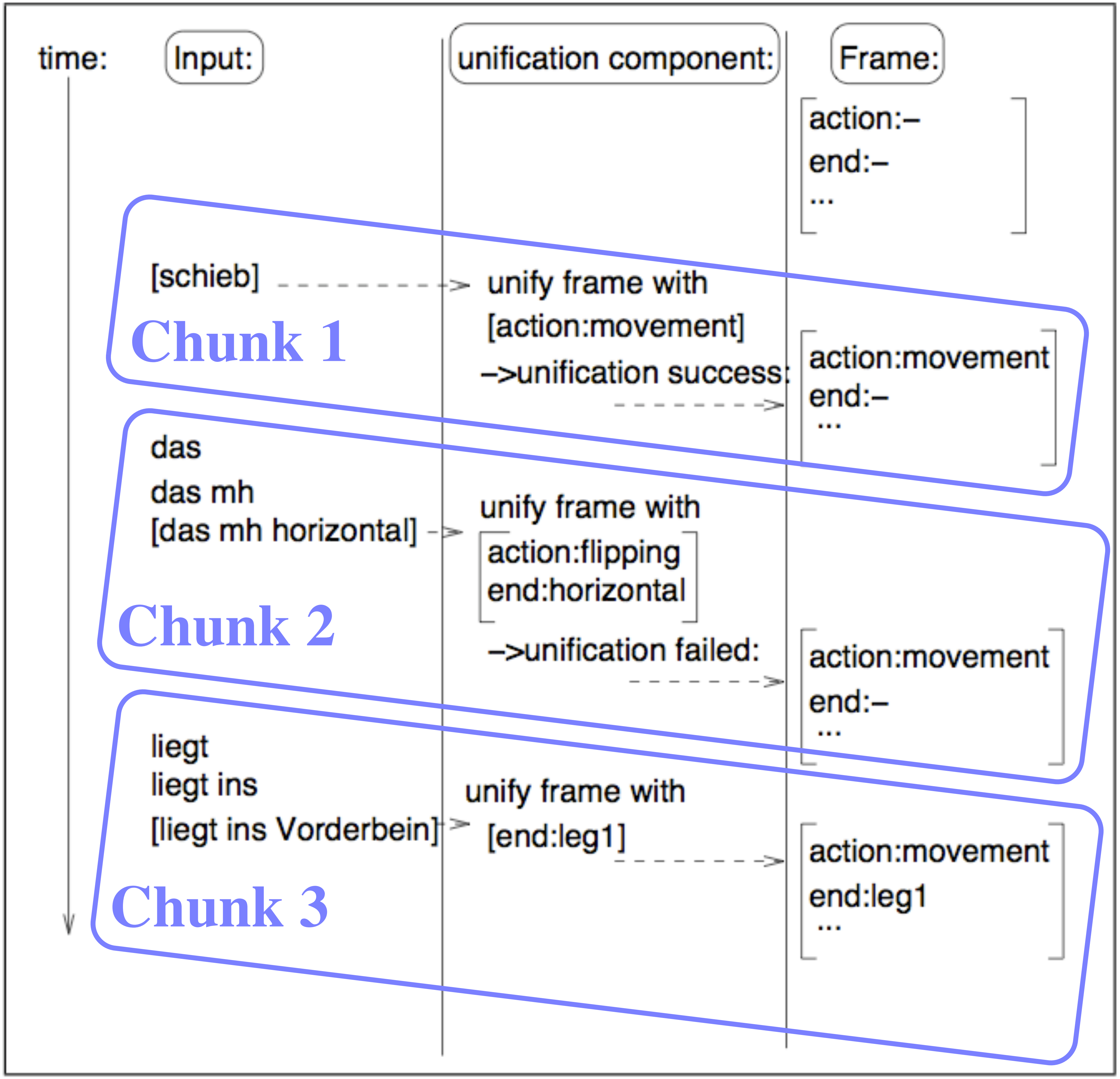
RUBISC

- Unification-based
- Robust
- Incremental

Unification

schieb das mh... das horizontal mh liegt ins Vorderbein

{move that mh which is horizontal into the front leg}



Robustness

- pronunciation variants considered

  spieg(le | el) *{flip}*

- search only for relevant information in a chunk

  in uh der zweiten äh Reihe → zweite Reihe **is found**
  {in uh the second uh row} *{second row}*

- mechanism for jumping over words

  das merkwürdige Kreuz → das Kreuz **is found**
  {the strange cross} *{the cross}*

  ☛ merkwürdig *{strange}* is unknown word

Incrementality

- not strictly incremental
- sometimes more than one word required for a chunk

in das rechte	**waiting...**	Bein
{into the right}		*{leg}*

- first slot filled after 3.5 words
- average length of utterance = 12.4 words

End-of-Sentence Detection

- semantic-pragmatic completeness
- certain slots filled
- in RUBISC
  1. action
  2. end
  3. name or (xpos and ypos)

Evaluation

- 500 utterances manual labeled
- run on 400 unseen utterances
- 100 labelled and 50 other utterances for grammar development

  ! annotators could influence the grammar or the chunker to perform better ⇛ they weren't involved in their development

- used baseline (no slot filling)
  - always returns the empty frame

action:	None
end:	None
entity:name:	None
entity:xpos:	None
entity:ypos:	None

Evaluation

Evaluation results in %	baseline	RUBISC	without unification	without robustness	inter annotator agreement
concept error	83,0	45,5	49,5	73,3	32,3
frame correct	17,0	54,5	50,3	26,8	67,8
slots correct	64,3	86,8	84,6	78,8	92,1
action correct	27,8	90,3	85,8	64,3	89,0
end correct	68,0	85,8	81,0	73,8	95,8
name correct	48,0	86,3	84,5	79,0	86,8
xpos correct	87,5	83,0	83,0	86,5	94,5
ypos correct	89,5	88,8	88,8	90,3	94,5

worst case

best case

Evaluation

Evaluation results in %	baseline	RUBISC	without unification	without robustness	inter annotator agreement
concept error	83,0	45,5	49,5	73,3	32,3
frame correct	**17,0**	**54,5**	**50,3**	**26,8**	**67,8**
slots correct	64,3	86,8	84,6	78,8	92,1
action correct	27,8	90,3	85,8	64,3	89,0
end correct	68,0	85,8	81,0	73,8	95,8
name correct	48,0	86,3	84,5	79,0	86,8
xpos correct	87,5	83,0	83,0	86,5	94,5
ypos correct	89,5	88,8	88,8	90,3	94,5

worst case

best case

Evaluation

Evaluation results in %	baseline	RUBISC	without unification	without robustness	inter annotator agreement
concept error	83,0	45,5	49,5	73,3	32,3
frame correct	17,0	54,5	50,3	26,8	67,8
slots correct	64,3	86,8	84,6	78,8	92,1
action correct	**27,8**	**90,3**	**85,8**	**64,3**	**89,0**
end correct	68,0	85,8	81,0	73,8	95,8
name correct	48,0	86,3	84,5	79,0	86,8
xpos correct	87,5	83,0	83,0	86,5	94,5
ypos correct	89,5	88,8	88,8	90,3	94,5

worst case

best case

Evaluation

Evaluation results in %	baseline	RUBISC	without unification	without robustness	inter annotator agreement
concept error	83,0	45,5	49,5	73,3	32,3
frame correct	17,0	54,5	50,3	26,8	67,8
slots correct	64,3	86,8	84,6	78,8	92,1
action correct	27,8	90,3	85,8	64,3	89,0
end correct	68,0	85,8	81,0	73,8	95,8
name correct	48,0	86,3	84,5	79,0	86,8
xpos correct	**87,5**	**83,0**	**83,0**	**86,5**	**94,5**
ypos correct	**89,5**	**88,8**	**88,8**	**90,3**	**94,5**

worst case (baseline) — best case (inter annotator agreement)

Error Analysis

- 50 utterances
  - vocabulary restrictions
  - restrictions in regular expressions

in den Rücken *{into the back}*	✓	so dass es den Rücken berührt *{such that it touches the back }*	✖

  - start positions confusable with end positions

Intermediate Conclusion

- incremental chunking by sense units suitable for spoken dialogue systems

- expressve power limited to extra coding work
  - lack of generalization
  - suitable for restricted domain

Incremental Parsing, or Incremental Grammar?

Nikolina: Now, I'm going to continue with

Benjamin: The second paper.

Nikolina: the second paper.

Ruth: What did Alex ...

Hugh: Design? A kaleidoscope.

Dialogue Modeling

- to model shared utterances, cross-speaker elipses, alignment
- Need for

  1. incrementality
  2. interaction between parsing and generation processes

Background: Dynamic Syntax

- parsing-directed grammar formalism
- **semantic** interpretation of an utterance represented by decorated tree structure
- grammaticality ≡ parsability

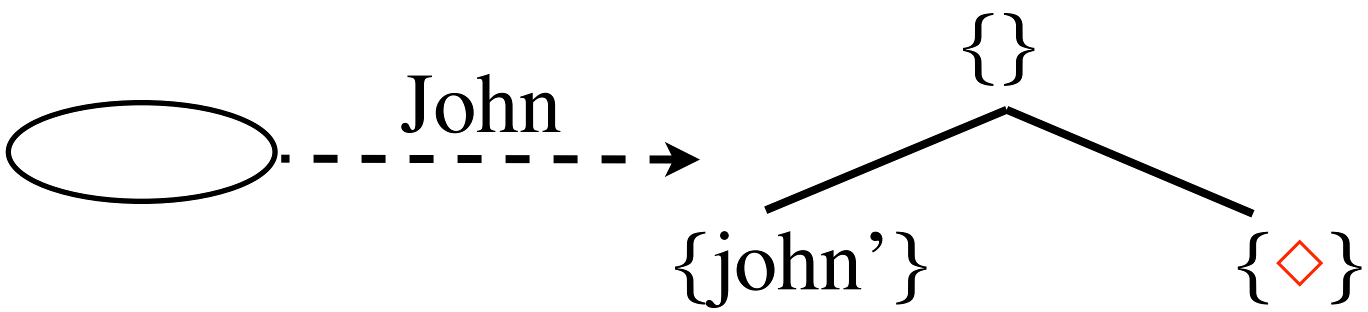
  ➡ successful incremental construction of a tree structure logical form

Parsing with Dynamic Syntax

John likes Mary.

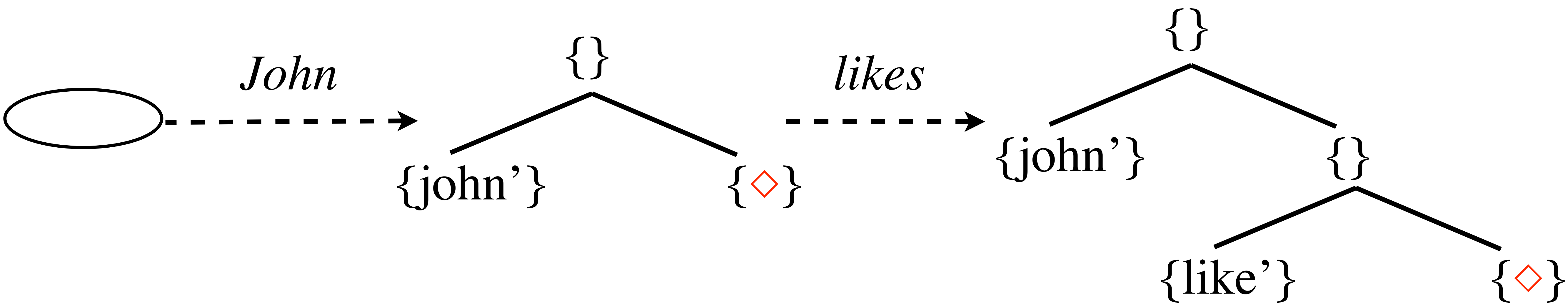
Parsing with Dynamic Syntax

John likes Mary.



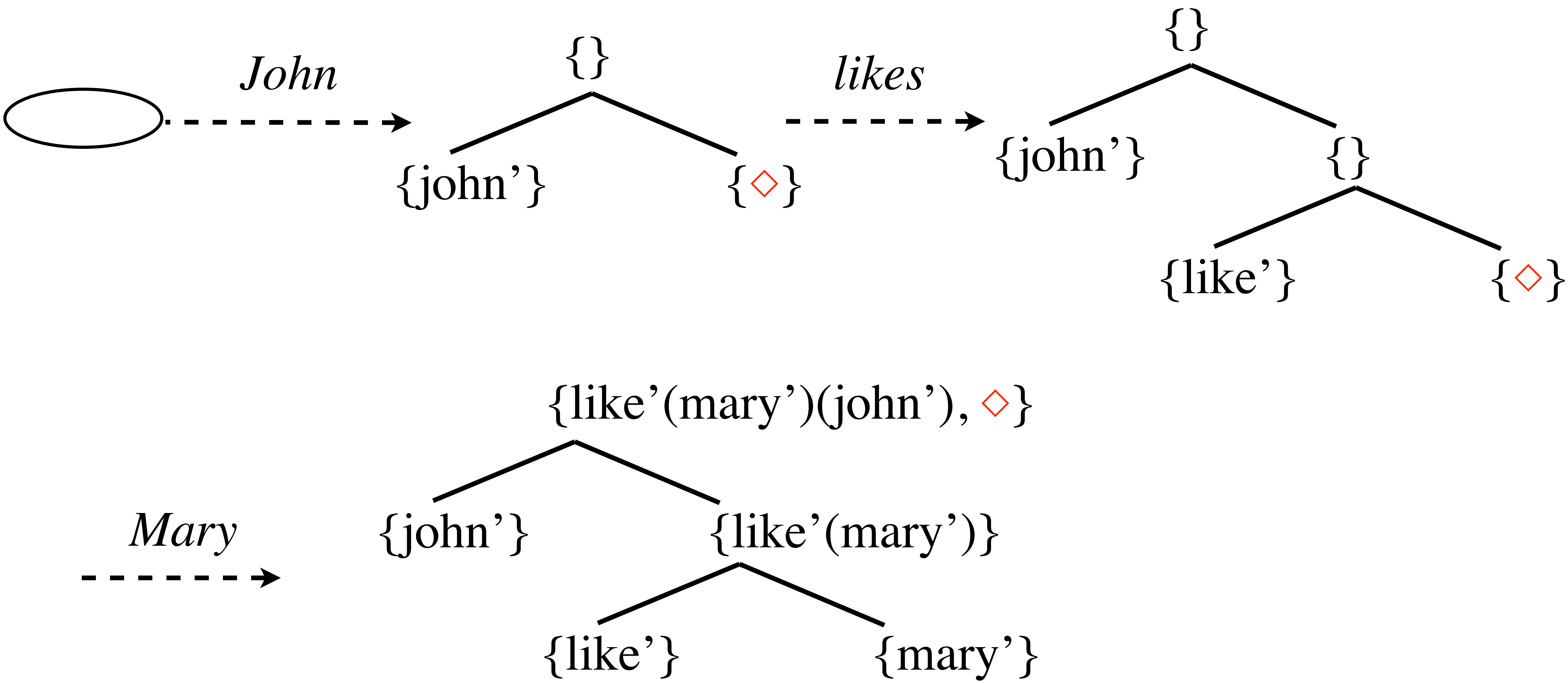
Parsing with Dynamic Syntax

John likes Mary.



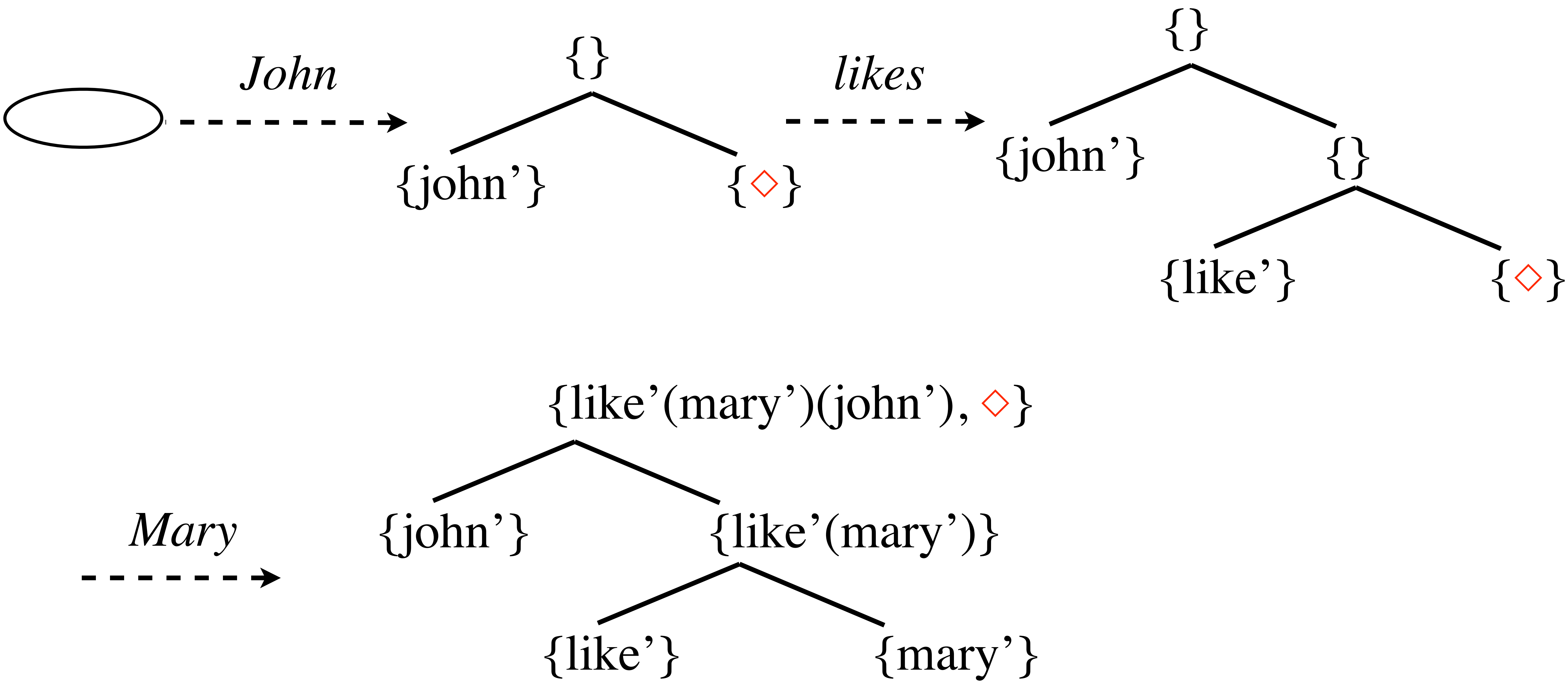
Parsing with Dynamic Syntax

John likes Mary.



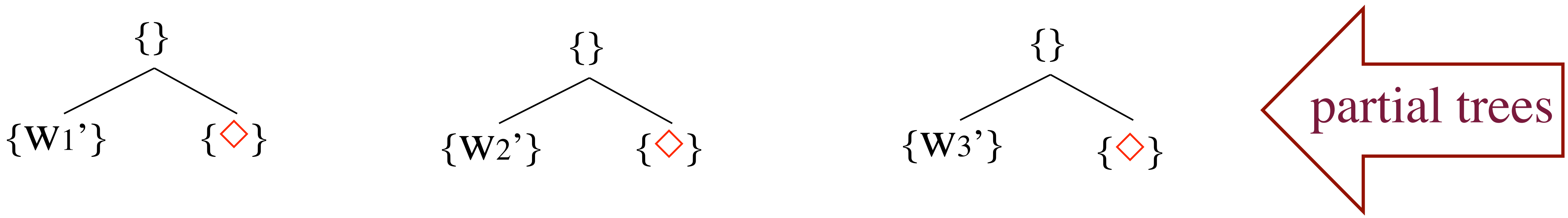
Parsing with Dynamic Syntax

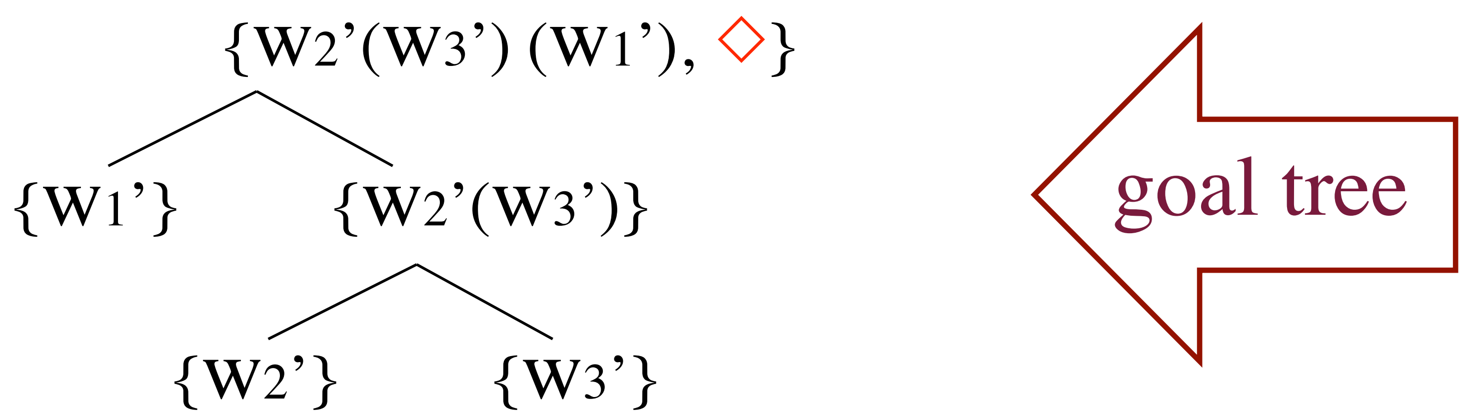
John likes Mary.



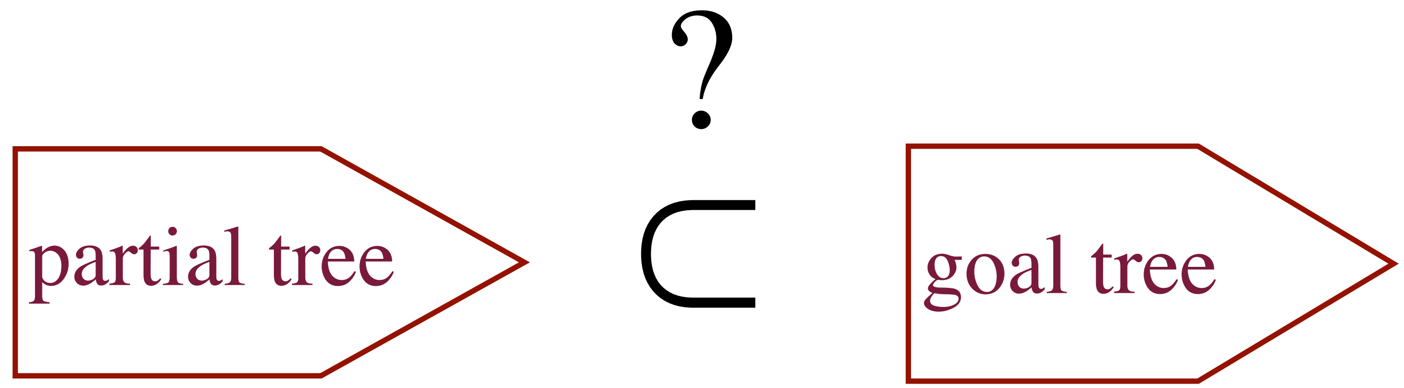
Generation with Dynamic Syntax

W1, W2, W3

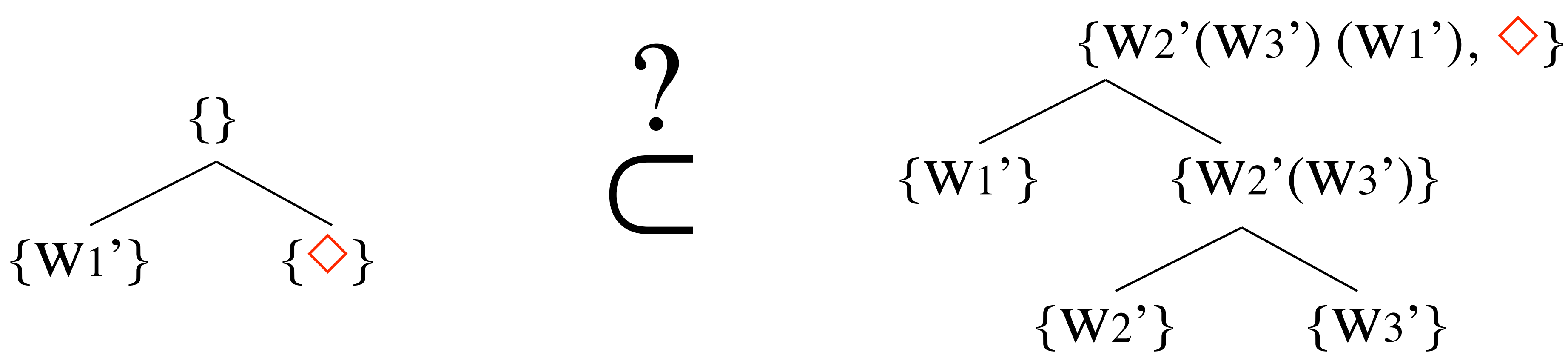




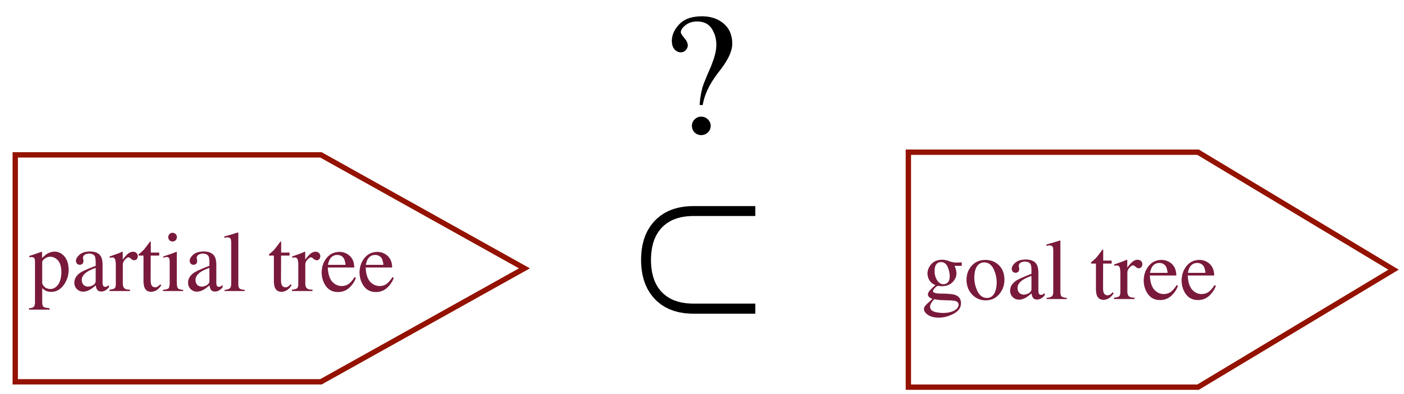
Generation with Dynamic Syntax



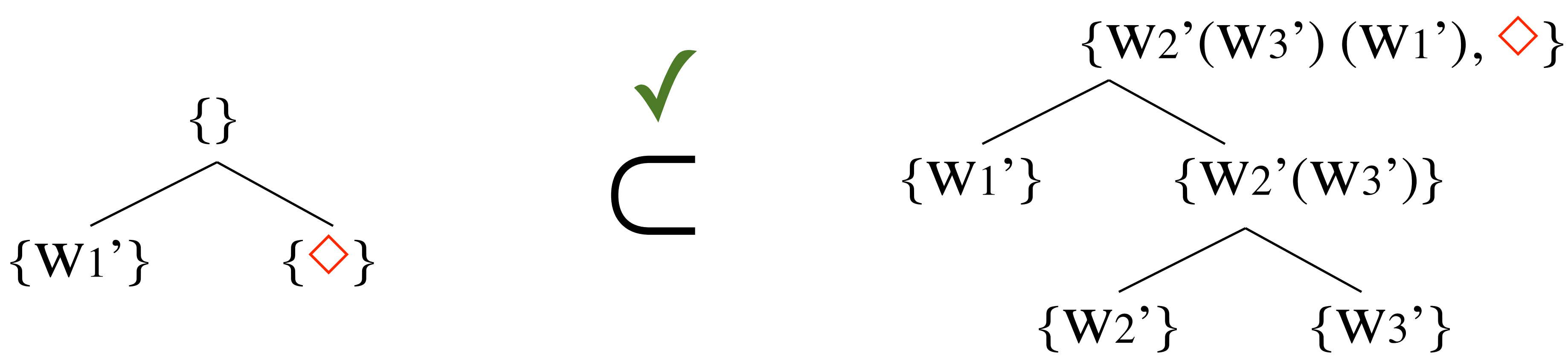
W1, W2, W3



Generation with Dynamic Syntax

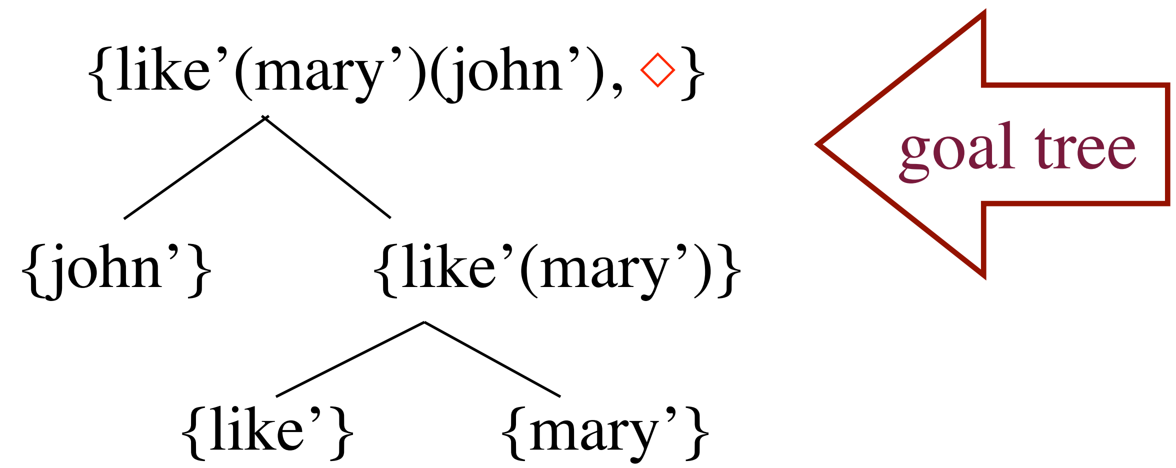


W1, W2, W3



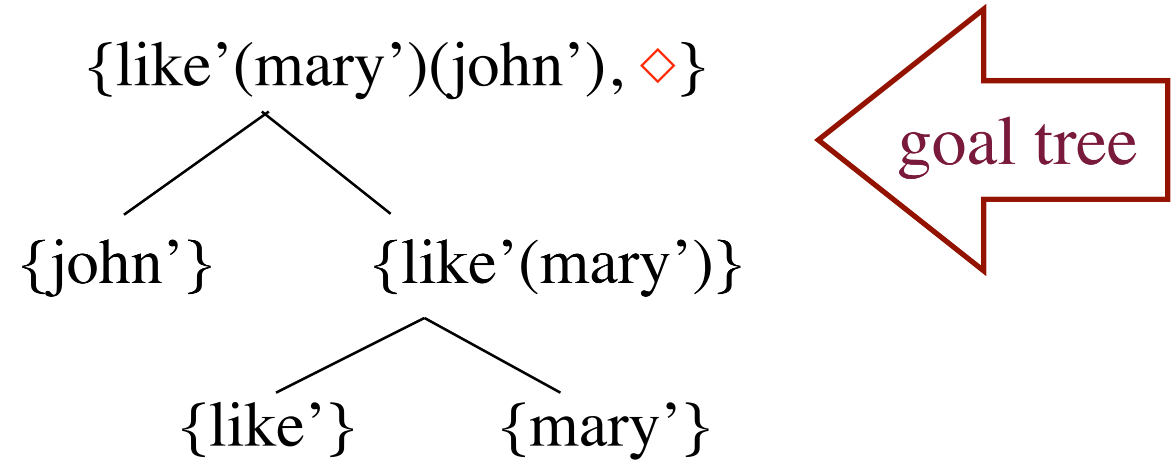
Generation with Dynamic Syntax

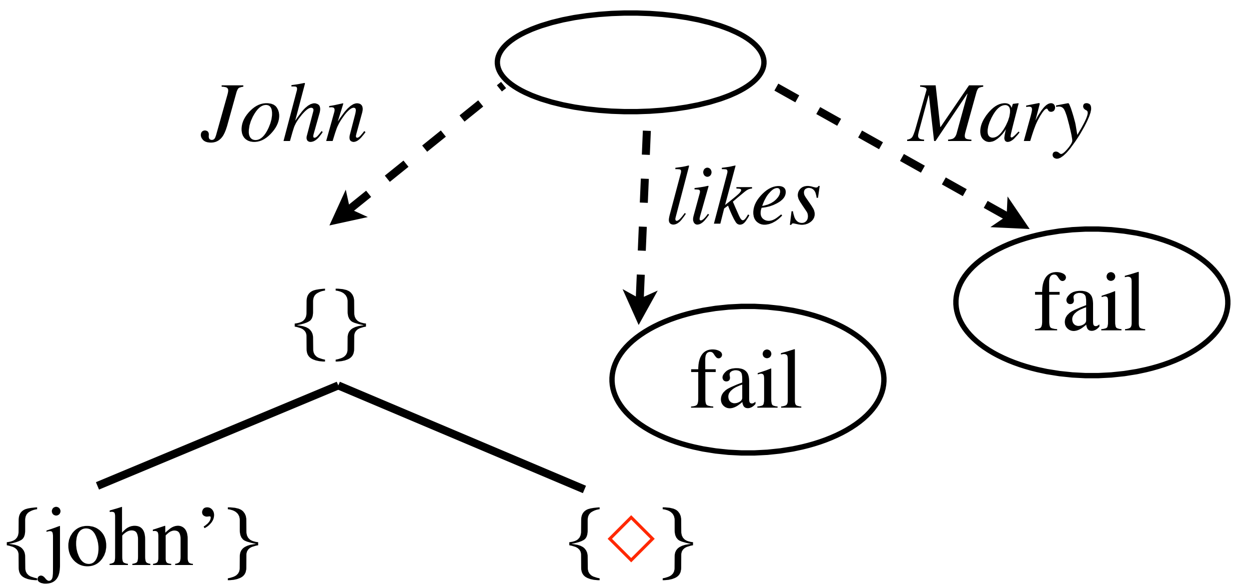
John likes Mary.



Generation with Dynamic Syntax

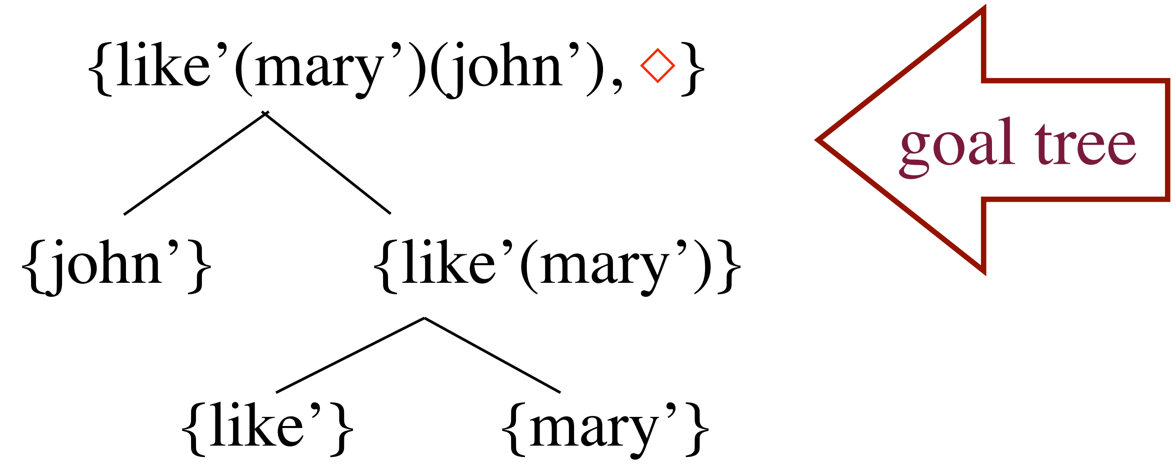
John likes Mary.

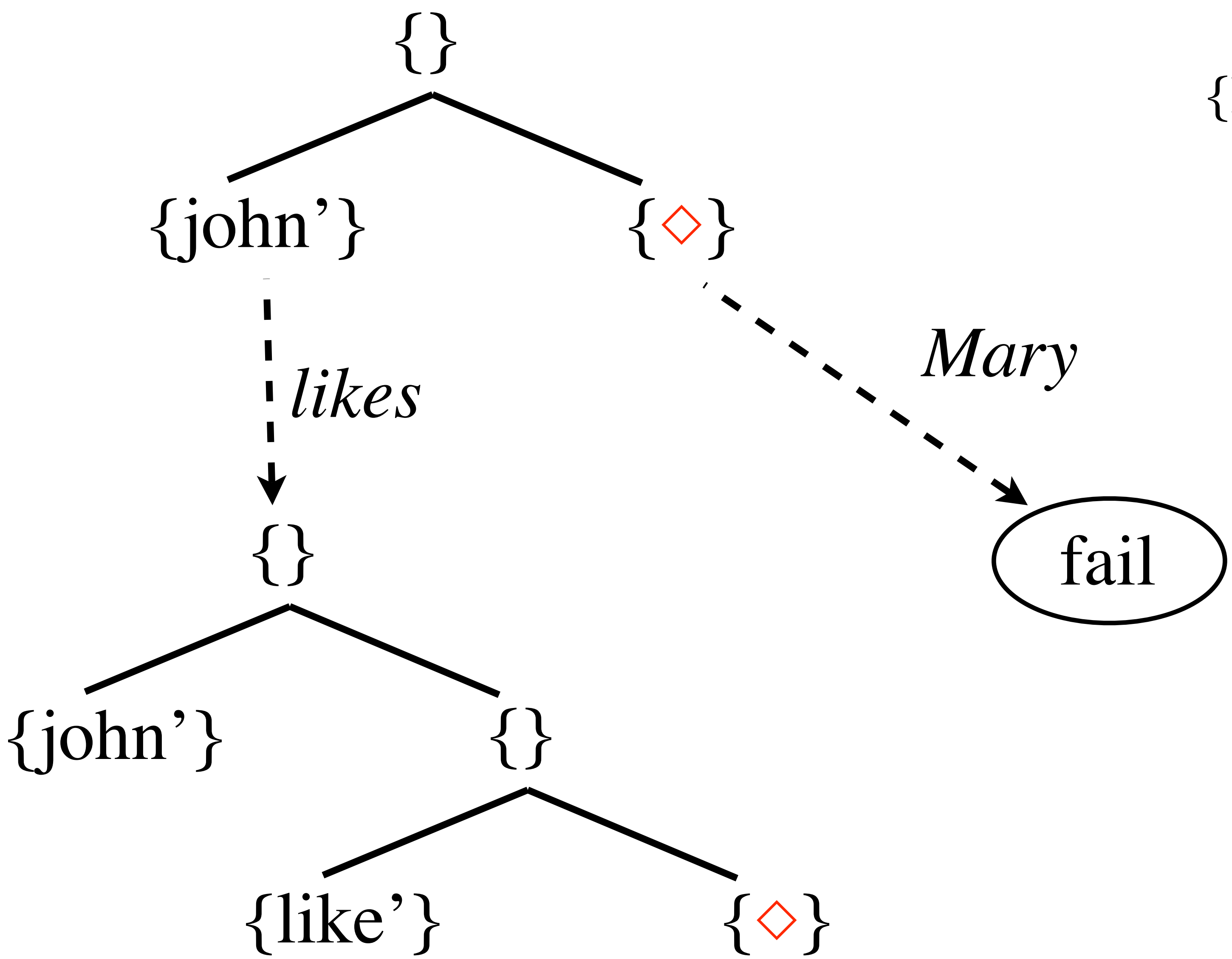




Generation with Dynamic Syntax

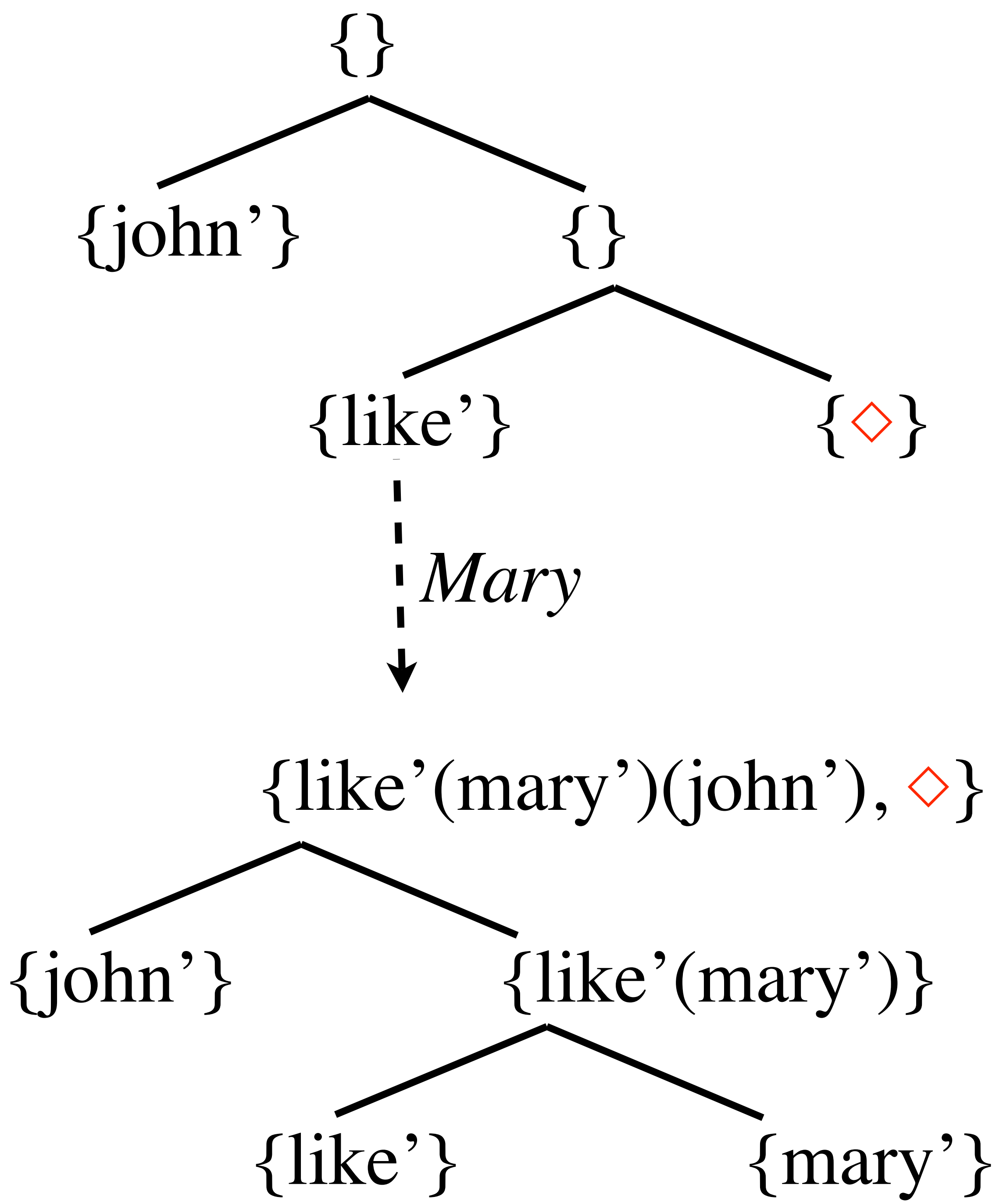
John likes Mary.

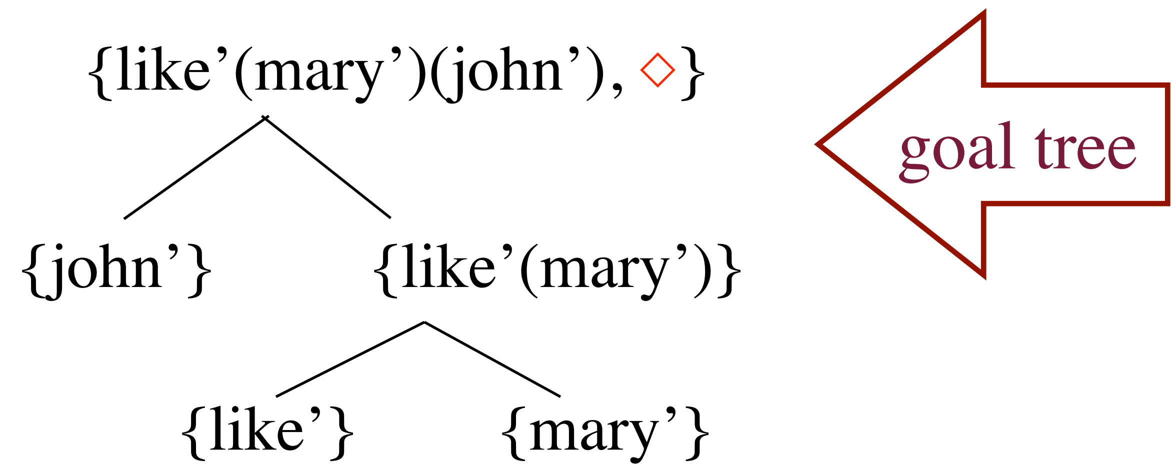




Generation with Dynamic Syntax

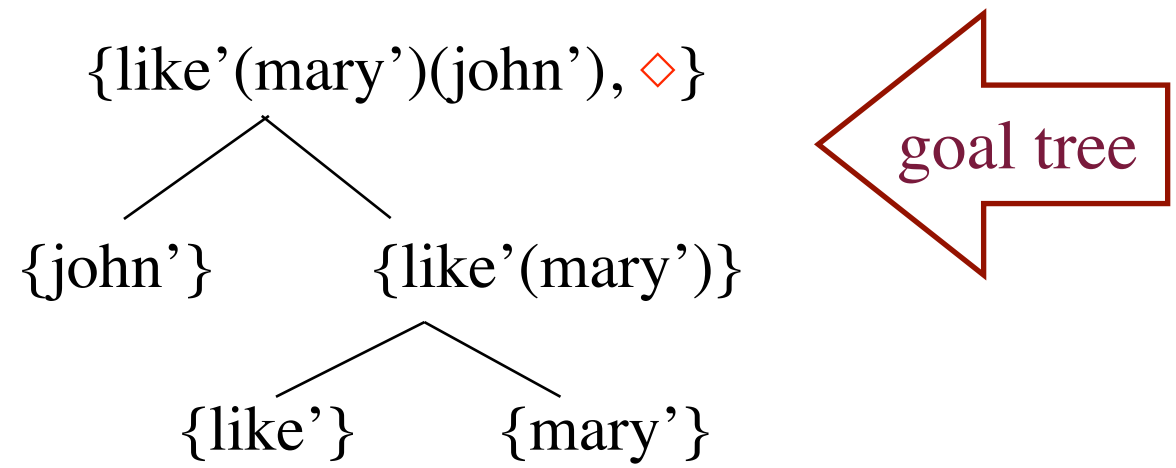
John likes Mary.

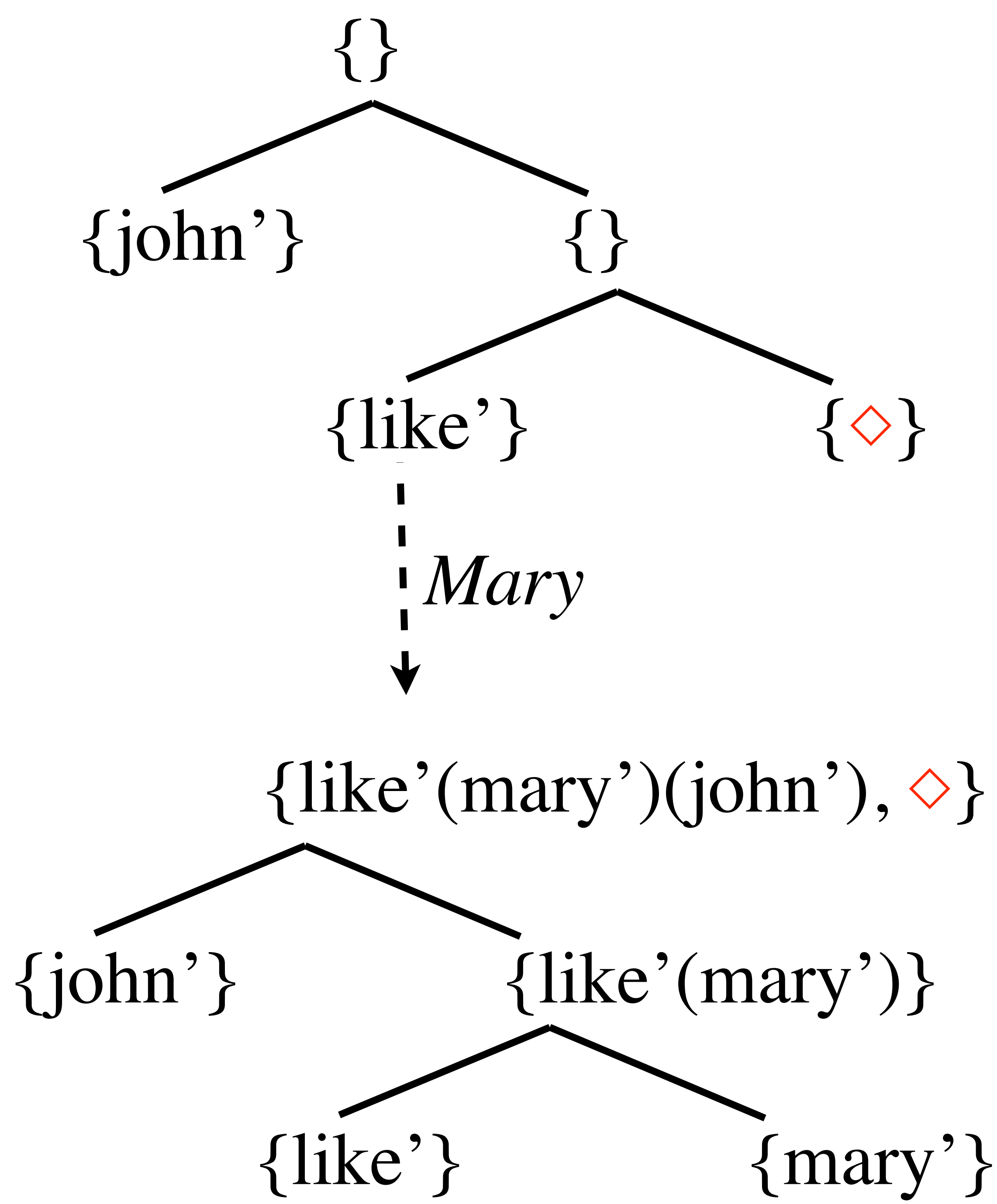




Generation with Dynamic Syntax

John likes Mary.





Contextual Model

1. semantic trees
2. sequences of words
3. associated lexical actions

☛ available to parsing and generation ⇒ enables modeling of complex dialogue phenomena

Parsing in Context

Parser state consists of set of triples $\langle T, W, A \rangle$

- **T** is (partial) semantic tree
- **W** is the sequence of words
- **A** is the set of actions used to create **T** from **W**

initial parser state $:= \{\langle T_a, \emptyset, \emptyset \rangle\}$ /T_a is the basic axiom/

☛ any tree: **context** $:= \{\langle T_0, W_0, A_0 \rangle, \langle T, W, A \rangle\}$

☛ parse completed (disambiguation resolved)

context $:= \{\langle T_1, W_1, A_1 \rangle, \langle T, W, A \rangle\}$

Generation in Context

Generator state consists of pair $\langle T_g, X \rangle$

- $\mathbf{T_g}$ is the goal semantic tree
- **X** is set of pairs (**S**, **P**)
  - **S** is candidate partial string
  - **P** is the analogous parser state

initial generator state $:= \langle T_g, \{(\varnothing, P_0\} \rangle$, where $P_0 := \{\langle T_a, \varnothing, \varnothing \rangle\}$

☛ any tree: **context** $:= \{\langle T_0, W_0, A_0 \rangle, \langle T, W, A \rangle\}$

Shared Utterances

- *necessarily* same intermediate representation shared
- parsing and generation can start from **any** state

Hearer to Speaker

$\langle T_g, \{(\varnothing, P_0\}\rangle \Rrightarrow \langle T_g, \{(\varnothing, P_t\}\rangle$

Requirement: $\exists\ T_a \in P_t \wedge T_a \subset T_g$

Speaker to Hearer

$\langle S_t, P'_t\rangle$

initial speaker needs P'_t

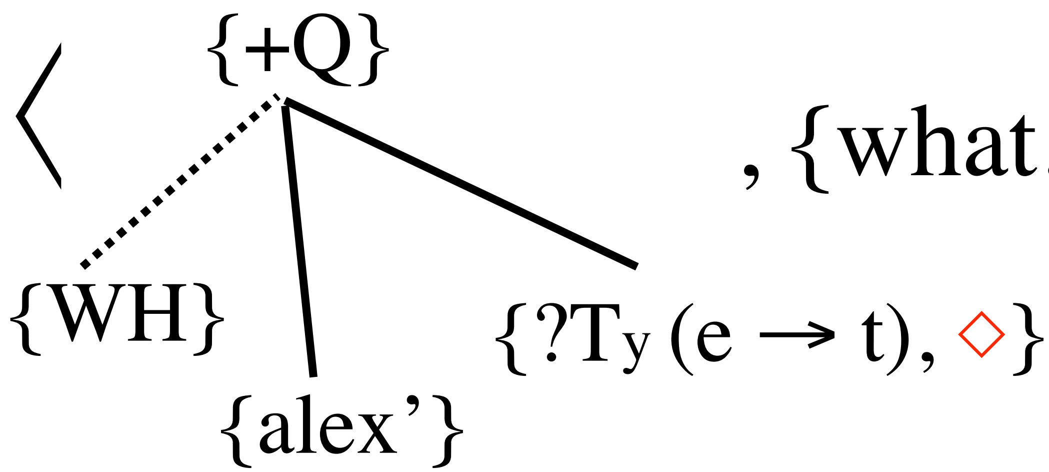
Requirement: $\forall\ T_a \in P'_t \wedge T_a \subset T_g$

$\rightarrow$ keep(T_a)

Shared Utterances: Hearer to Speaker

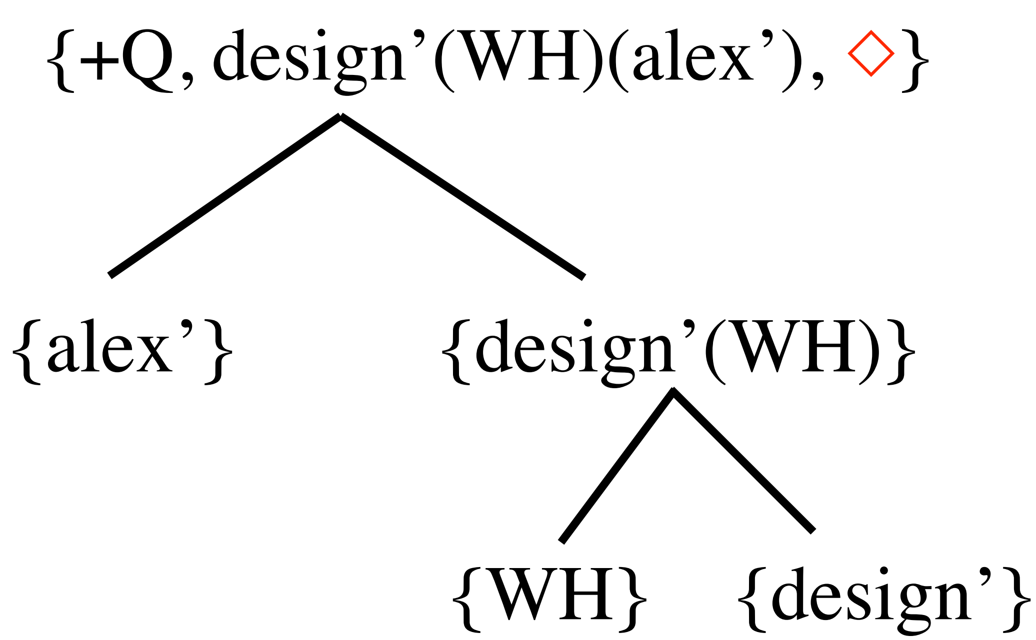
What did Alex ... / ...design?

$P_t = \langle$



, {what, did, alex}, $\{a_1, a_2, a_3\} \rangle$

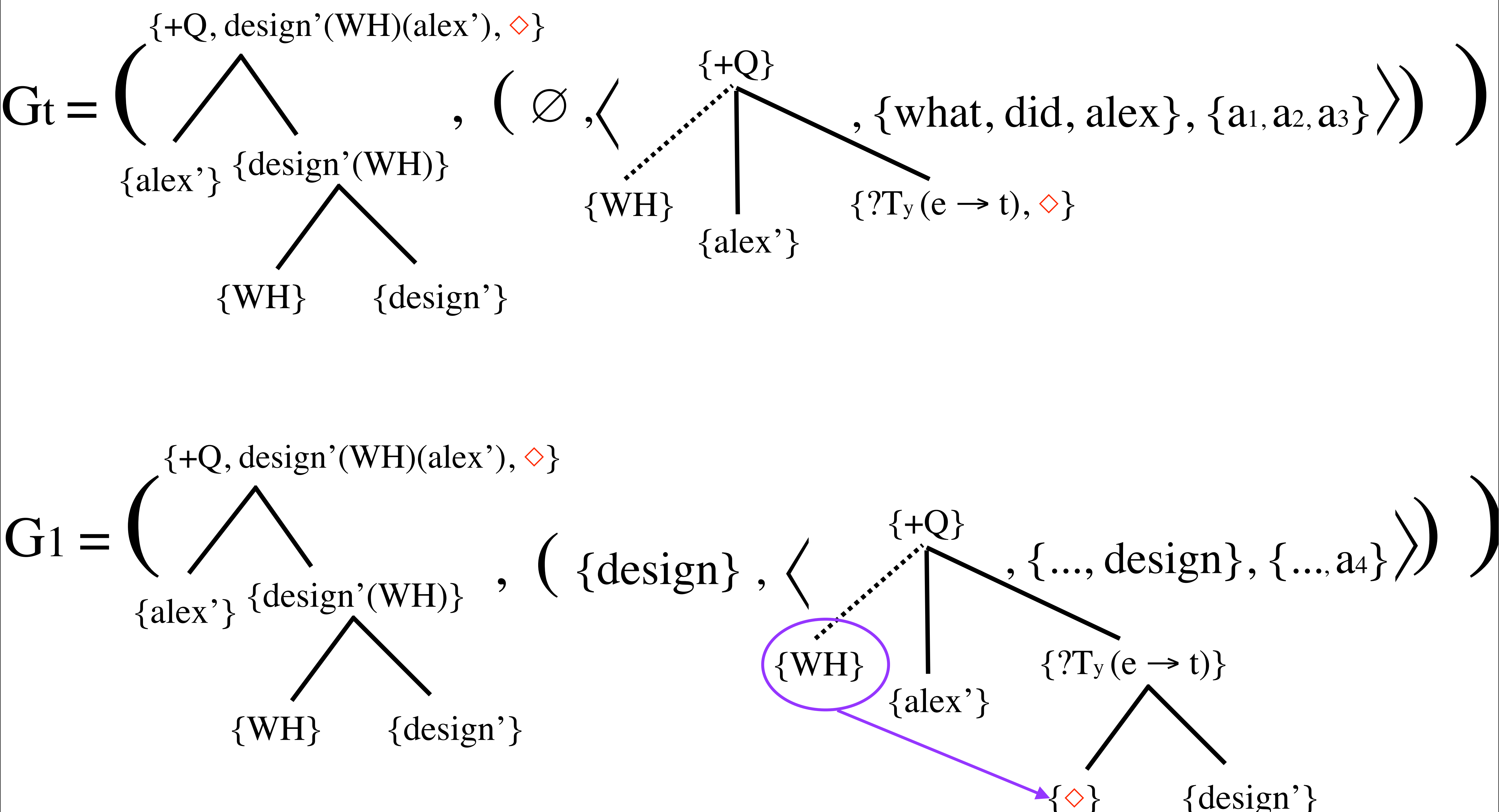
$G_t = ($



, $(\varnothing, P_t)$ $)$

Shared Utterances: Hearer to Speaker

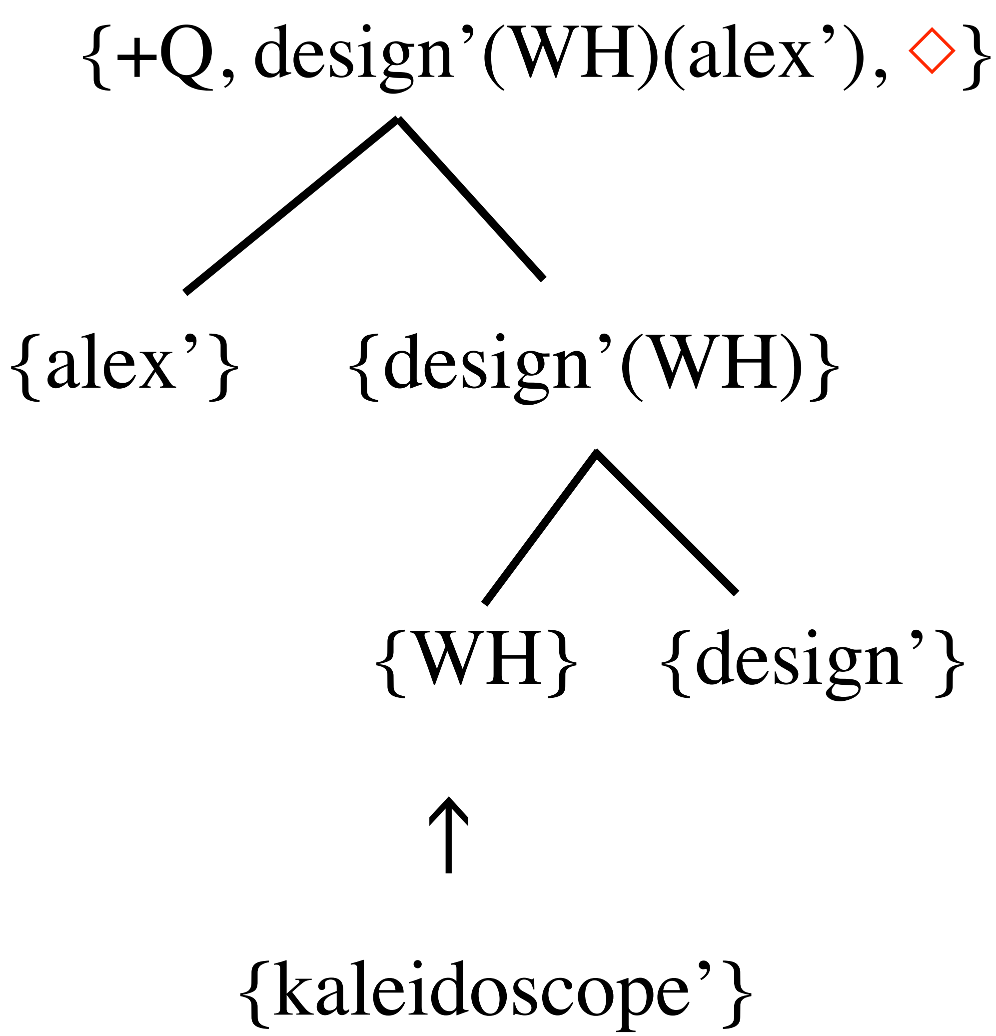
What did Alex … / …design?

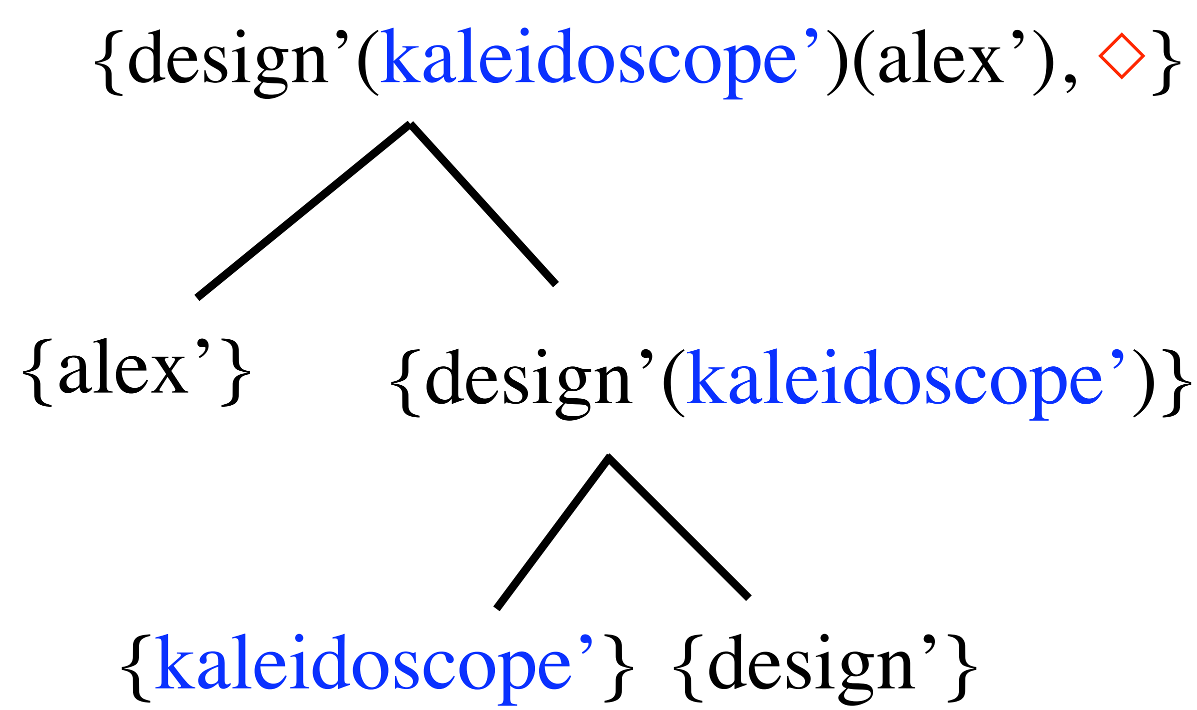


Cross-Speaker Ellipsis

A: *What did Alex design?*

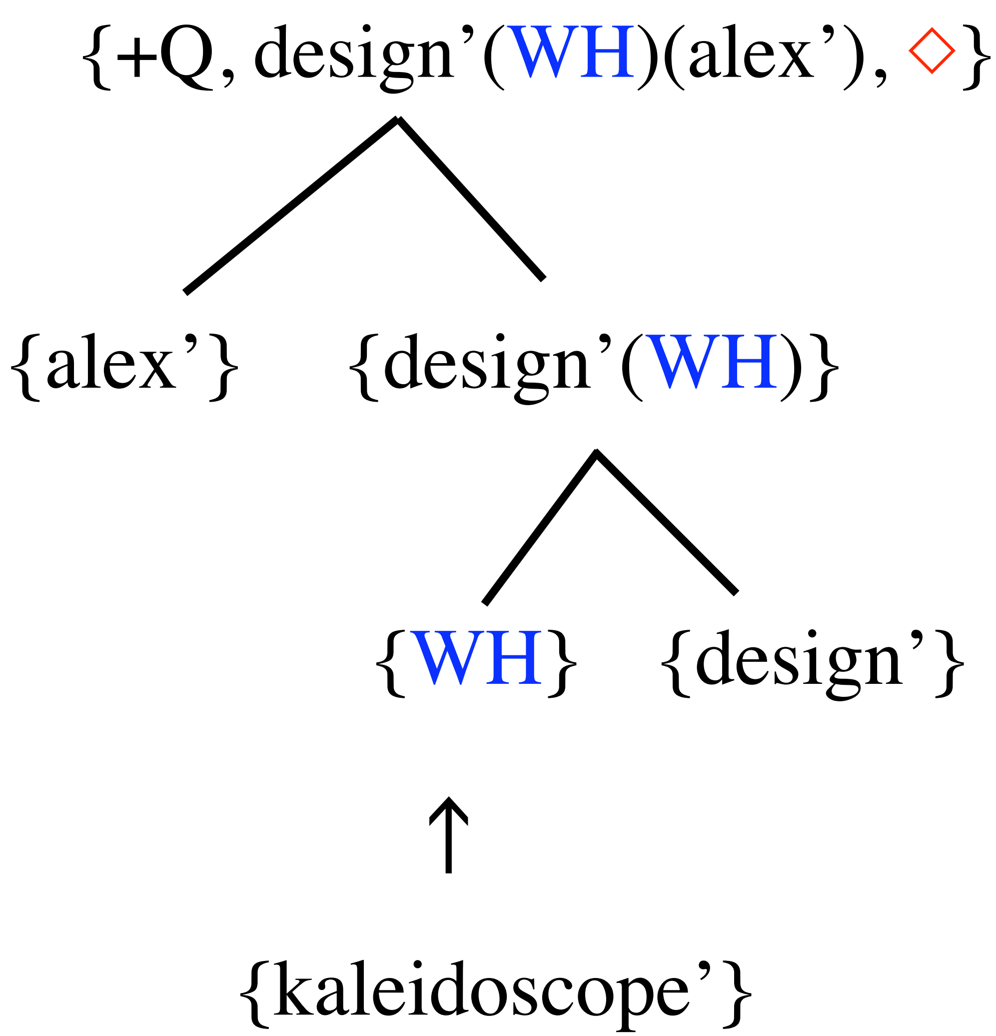
B: *A kaleidoscope*.

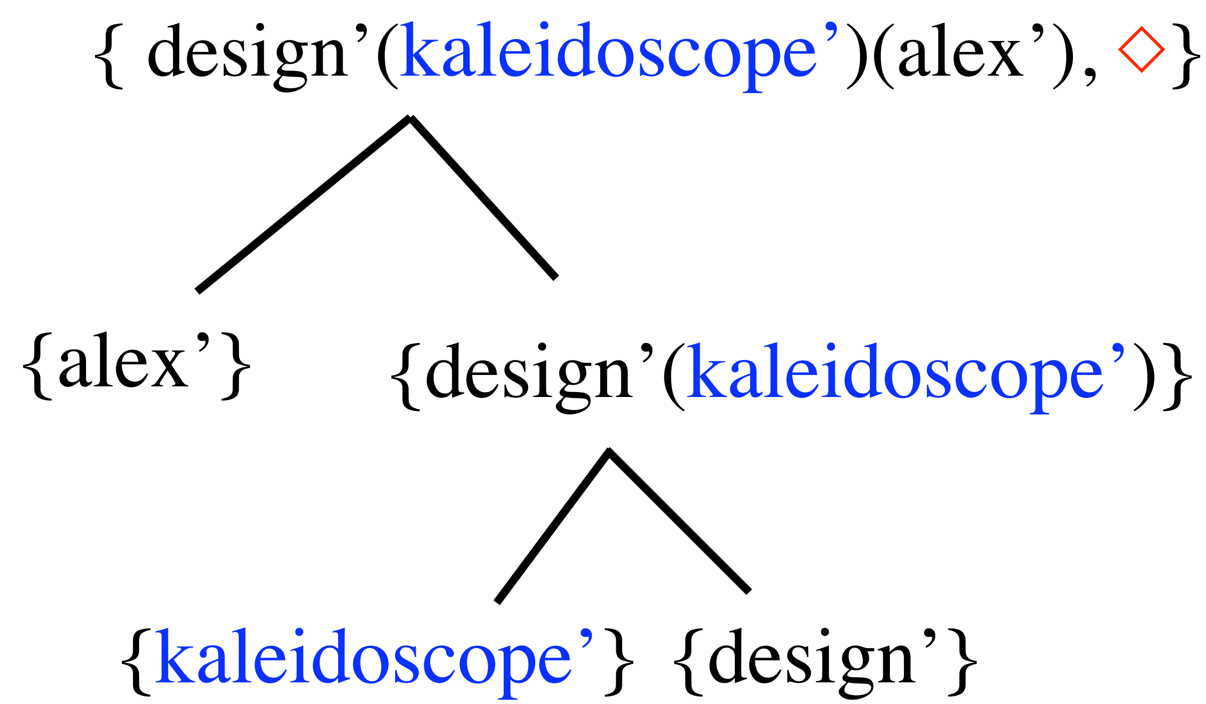




Cross-Speaker Ellipsis

A: *What did Alex design?* B: *A kaleidoscope*.





VP Ellipsis

A: *A policeman who arrested Bill read him his rights*.

B: *The policeman who arrested Tom did too*.

☛ **re-use sequence of actions from the context**

Alignment

- parsing and generation search for suitable entries in the

lexicon

- expensive process

☛ **minimising erffort via re-use sequence of selected actions from the context**

Summary

- semantic parsing
  - suitable for dialogue systems
  - influenced by the current context
  - provides straightforward model of complex dialogue phenomena

References

- Atterer, M. and David Schlangen [2009] **RUBISC -- a Robust Unification-Based Incremental Semantic Chunker** *in Proceedings of 2nd International Workshop on Semantic Representation of Spoken Language 2009* (SRSL 2009), Athens, Greece, March 2009

- Matthew Purver and Ruth Kempson. **Incremental Parsing, or Incremental Grammar?**. In *Proceedings of the ACL Workshop on Incremental Parsing*, pages 74-81, Barcelona, Spain, July 2004.

- http://www.kcl.ac.uk/research/groups/ds/

- http://godel.stanford.edu/mpurver/ds/

Thank you for your attention!

Discussion

- Is robustness always an advantage?
- Is it a disadvantage that the semantic of the first two words isn't put together?