Outline of a formal framework for Role and Reference Grammar

Rainer Osswald & Laura Kallmeyer

Heinrich-Heine-Universität Düsseldorf
SFB 991

Role and Reference Grammar 2013
International Conference

Freiburg, 2.8.2013

Introduction

Why a formal framework for RRG?

- Is this relevant for typological analysis?

  MAYBE NOT, BUT ...
  a formalization can help to eliminate **inconsistencies** and **gaps** of a theory.

- Doesn't RRG already come with a lot of formal elements?

  SURE, BUT ...
  these elements are not defined with **logical** and **mathematical** rigor.

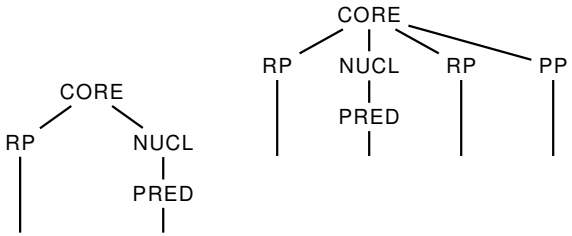
- Any further advantages?

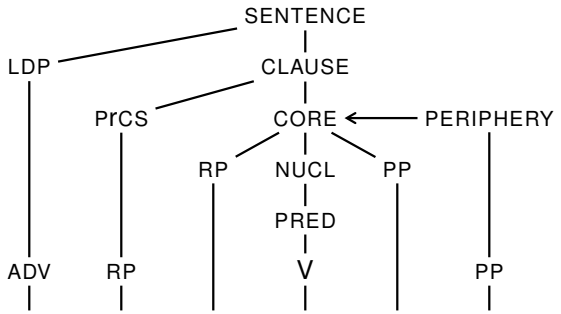
  YES!
  A formalization can serve as a basis for a **computational treatment** of RRG.

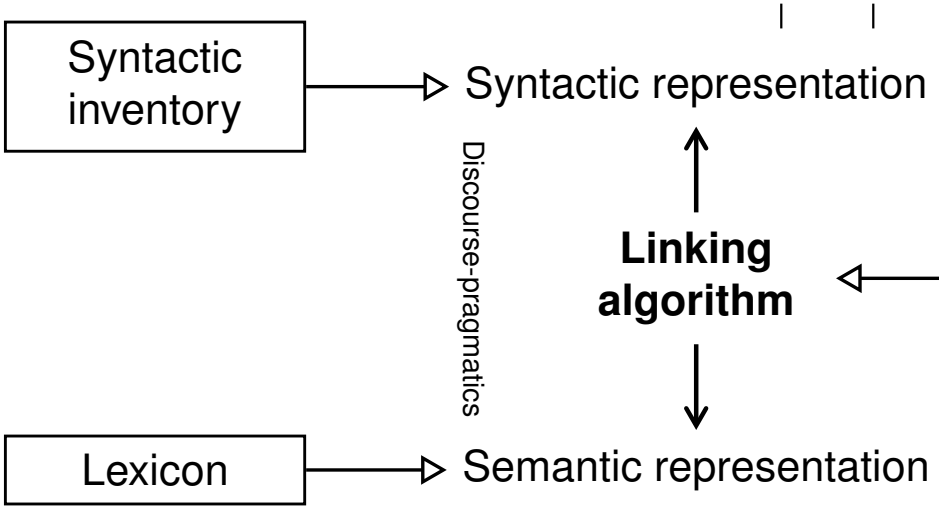
- Is that all?

  NOT AT ALL!
  E.g., a formalization should make it easier to **extend** and **modify** the theory.

The architecture of RRG



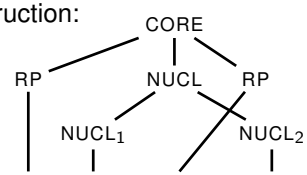




$[\mathbf{do}'(x, \varnothing)]$ CAUSE $[\text{INGR } \mathbf{shattered}'(y)]$

$\langle_{\text{IF}}$ *INT* $\langle_{\text{IF}}$ *PRES* $\langle_{\text{ASP}}$ *PERF PROG* $\langle \mathbf{do}'(\text{Kim}, [\mathbf{cry}'(\text{Kim})]\rangle\rangle\rangle\rangle$

MORPHOLOGY —
SYNTAX Juncture: nuclear
Nexus: cosubordination
Construction:



Linking: default
SEMANTICS $[\text{SEM}_{\text{NUCL}_1}]$ CAUSE $[\text{SEM}_{\text{NUCL}_2}]$
PRAGMATICS unspecified

Introduction

General plan of the formalization

- Take **all explanatory components** of RRG into account.
- Develop a **declarative** (i.e., non-procedural) constraint-based formulation.

Selection of tasks involved

- Syntactic representation

  Formal specification of the syntactic inventory and of the compositional operations on trees

- Semantic representation

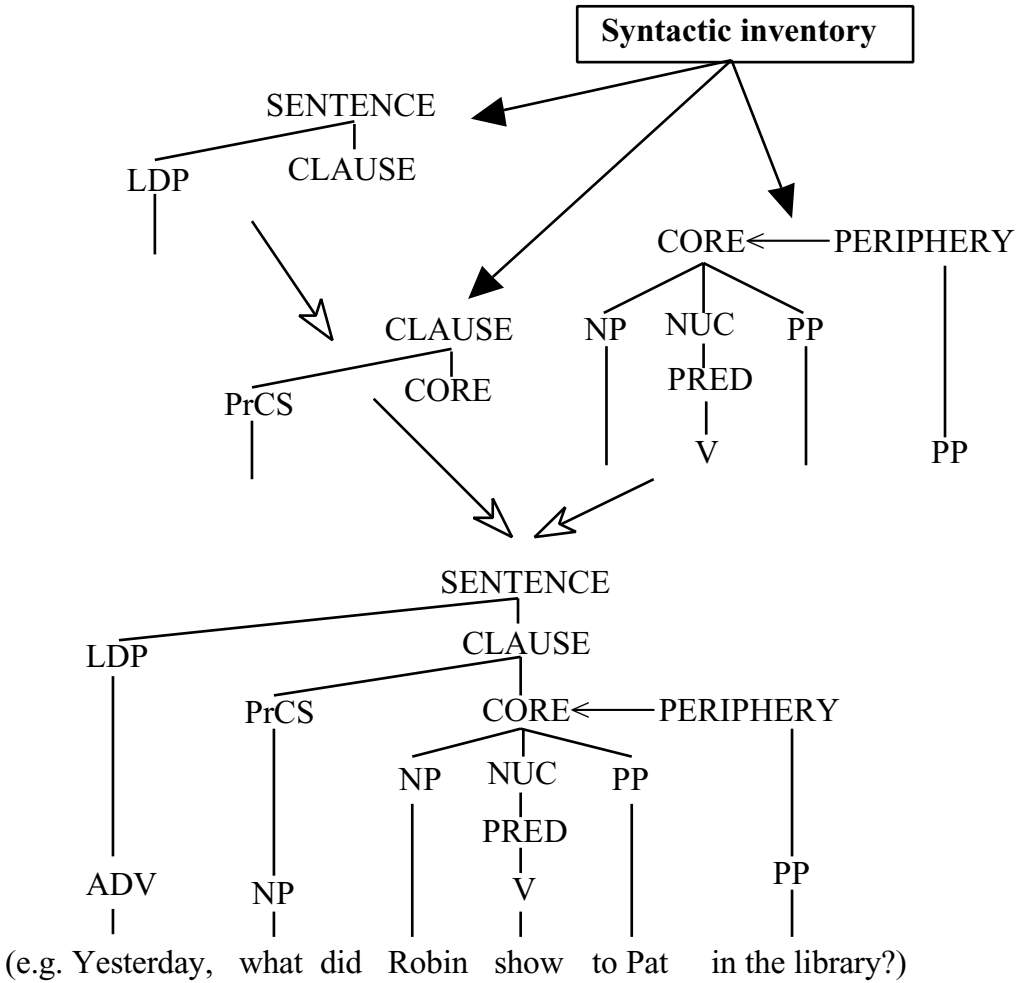
  Clarification of the logical and model-theoretic aspects of RRG's logical structures

- Linking algorithm

  Non-procedural, inherently bidirectional description as a system of constraints

Syntactic representation

The inventory of syntactic templates



[Van Valin 2005, p. 15]

Issues

- How are syntactic templates defined?
- How do they combine?

Proposal

- Use concepts from Tree Adjoining Grammars (TAG)
- Adapt TAG formalism to the syntactic dimension of RRG

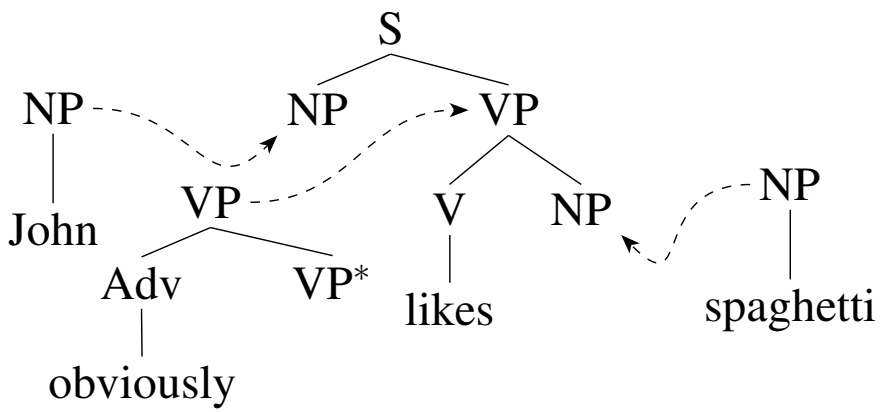
Syntactic representation

Background Lexicalized Tree Adjoining Grammars (LTAG)

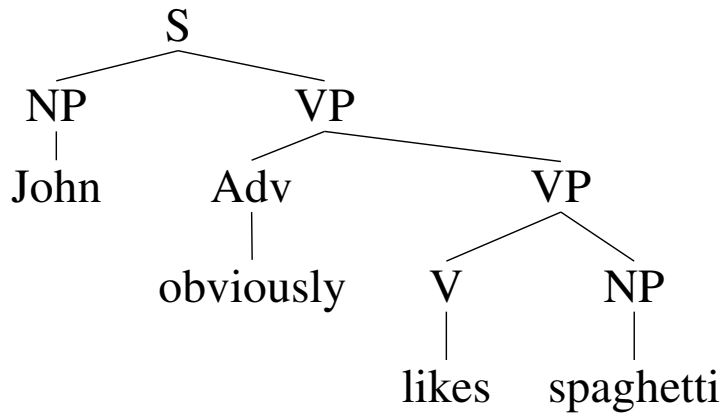
[e.g., Joshi & Schabes 1997]

- Tree rewriting system based on a set of **elementary** (**initial** and **auxiliary**) **trees**
- Two operations: **substitution** of initial trees at leaves
  adjunction of auxiliary trees

Example



two substitutions + one adjunction



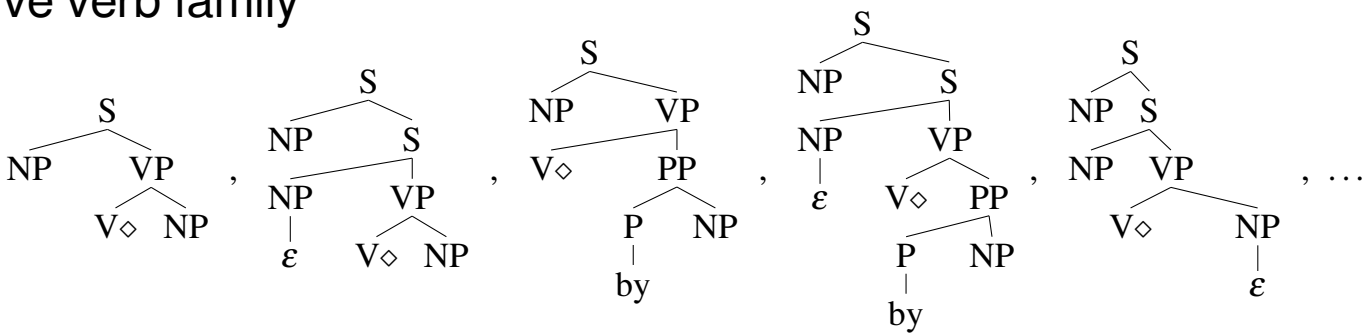
derived tree

Syntactic representation

Background Lexicalized Tree Adjoining Grammars

- Elementary trees are **lexicalized**, i.e., have lexical anchors.
- "**Complicate locally, simplify globally**" [Bangalore & Joshi 2010]
  All predicate-argument dependencies are encoded in elementary trees.
- De-anchored elementary trees are organized in **tree families**, which capture variations in subcategorization frames.

  Example transitive verb family



- Modular characterization of elementary trees in the **metagrammar**, a system of **tree descriptions**. [Crabbé & Duchier 2005]

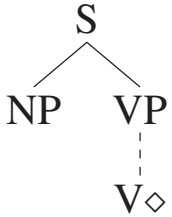
Syntactic representation

Background Metagrammar for LTAGs

- Specification of elementary trees as **minimal models** of tree descriptions (tree classes)

  Example Metagrammar fragment for transitive verb class

Class *CanSubj*

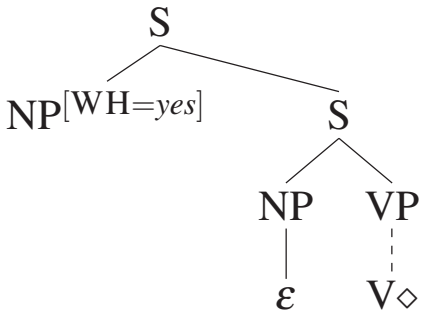


Class *DirObj*

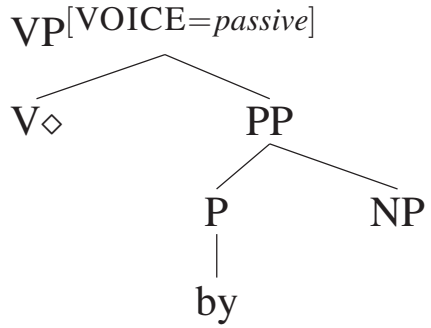
Class *Subj*

CanSubj ∨ *ExtractedSubj*

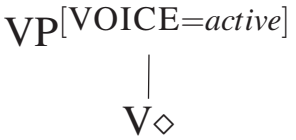
Class *ExtractedSubj*



Class *ByObj*



Class *ActV*



Class *PassV*

Class *Transitive*

((*Subj* ∧ *ActV*) ∨ *ByObj* ∨ *PassV*) ∧ ((*DirObj* ∧ *ActV*) ∨ (*Subj* ∧ *PassV*))

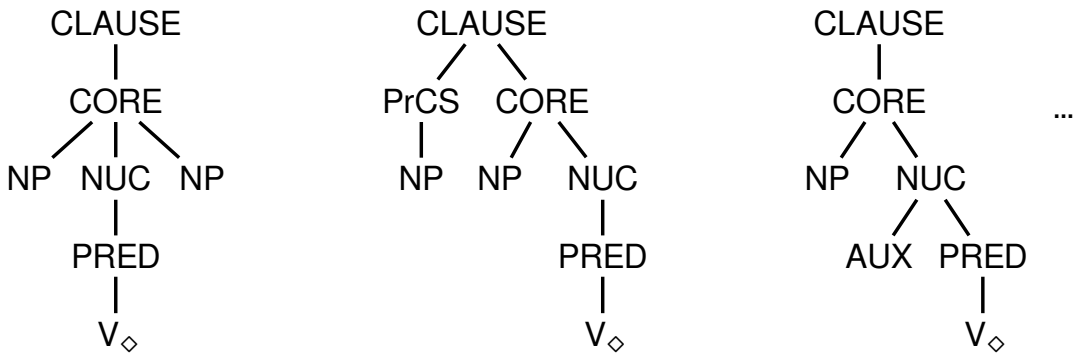
Syntactic representation

Application to the syntactic inventory of RRG

1. What are the elementary trees of RRG?
2. How can they be combined?
3. How can they be characterized as minimal models of metagrammatical specifications?

Possible candidates for elementary trees in RRG

- Basic predication templates and their variants
  e.g.



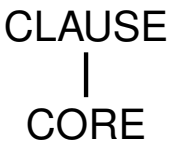
- Constructional schemas (strictly speaking, their syntactic dimension)
  e.g., the nuclear cosubordination templates of resultative constructions

Syntactic representation

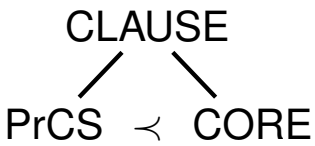
Metagrammar sketches

core-spine

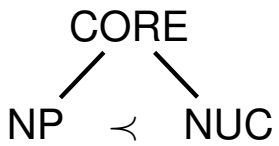
core-clause



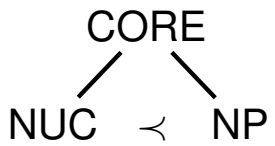
precore-slot



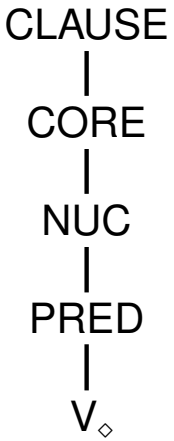
prenuc-np



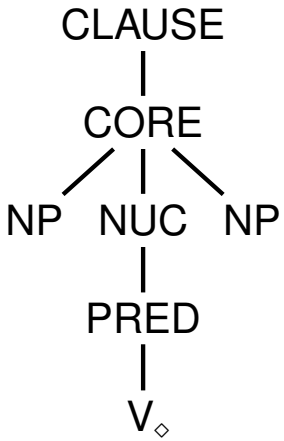
postnuc-np



clause-spine := *core-spine* ∧ *core-clause*



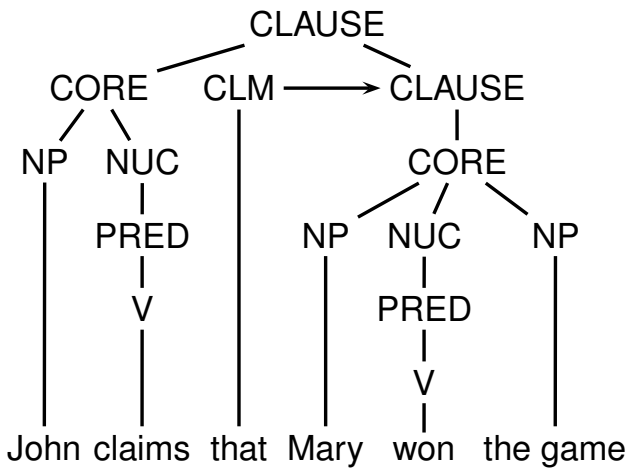
base-transitive := *clause-spine* ∧ *prenuc-np* ∧ *postnuc-np*



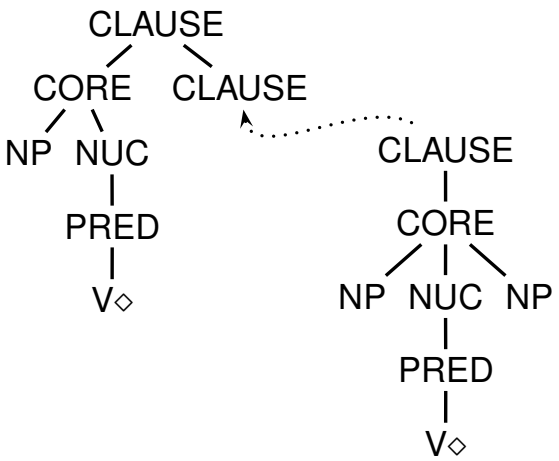
Tree operations for RRG

1. Standard substitution

Derived tree:



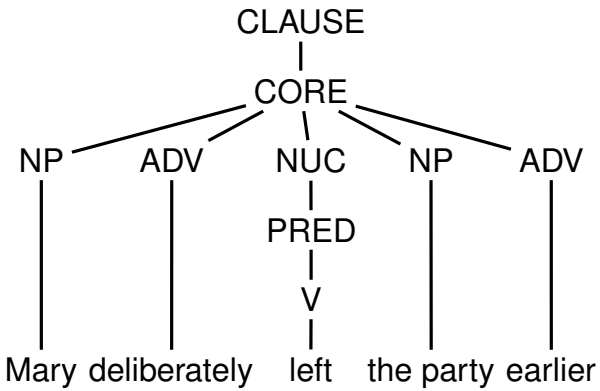
Derivation:



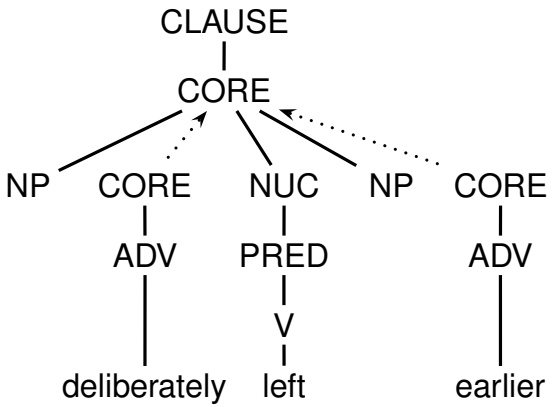
2. Sister adjunction

[Rambow et al. 1995]

Derived tree:



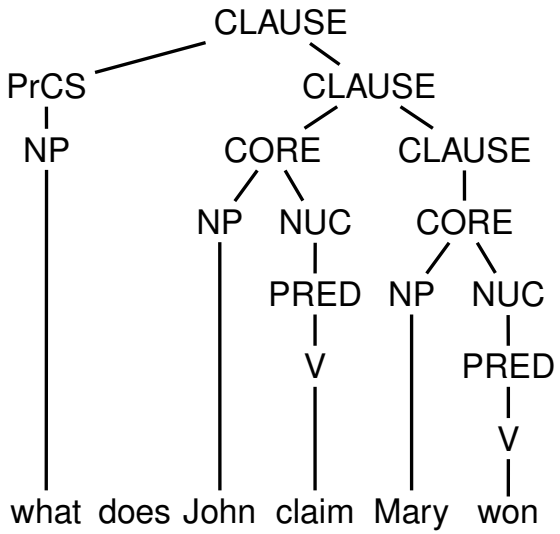
Derivation:



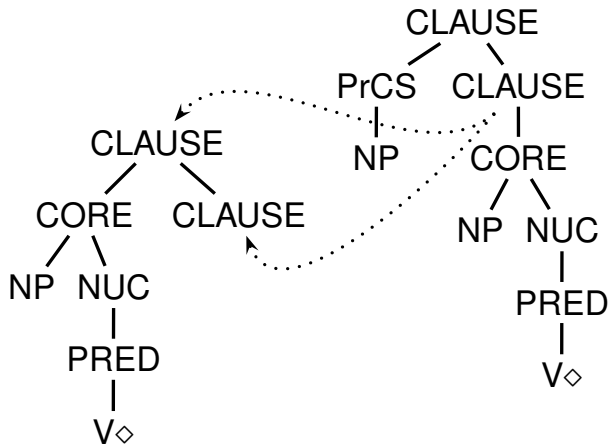
Tree operations for RRG

3. Wrapping substitution

Derived tree:



Derivation:



Tree Wrapping Grammar as a formal grammar framework

- More expressive than context-free grammars (can express cross-serial dependencies)
- CYK parsing algorithm with complexity $\mathcal{O}(n^6)$

[Kallmeyer, Osswald & Van Valin 2013]

Semantic representation

Logical structures in the lexicon and beyond

a. **do**′(*x*, **hit**′(*x*, *y*))
b. INGR **shattered**′(*y*)
c. [**do**′(*x*, ∅)] CAUSE [INGR **shattered**′(*y*)]
d. [**do**′(*x*, **hit**′(*x*, *y*))] CAUSE [INGR **shattered**′(*y*)]

Logical analysis of logical structures

Basic (uncontroversial) assumptions

- RRG's logical structures describe activities, states, changes of state, causations, etc.
- The decompositional structure of logical structures reflects the internal structure of the described events.

  E.g., causative events have a cause and an effect component.

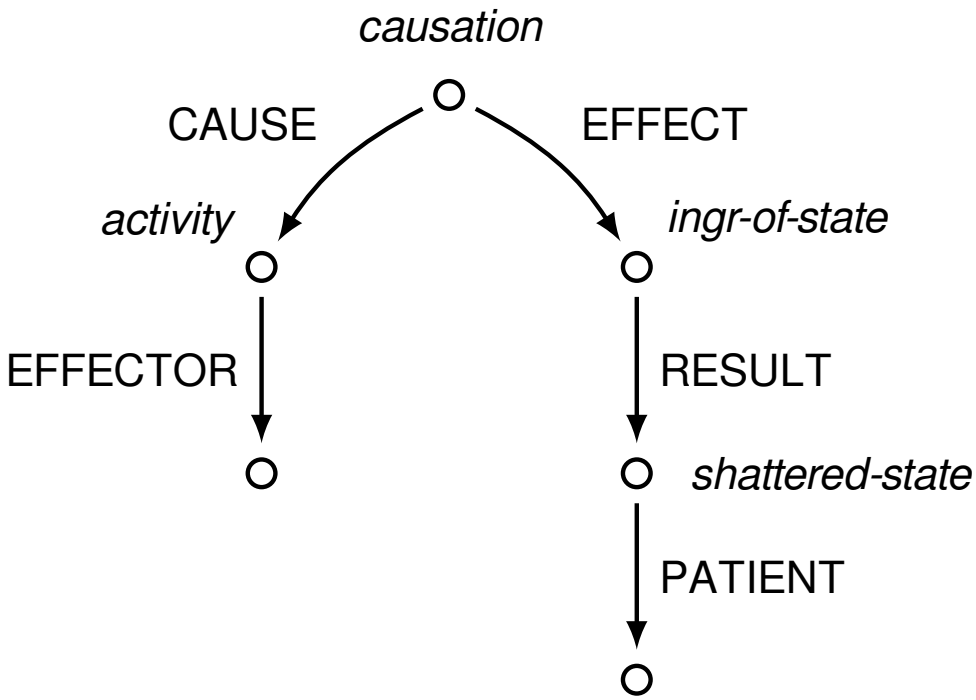
Semantic representation

Logical analysis of logical structures (cont'd)

Example [**do**′(*x*, ∅)] CAUSE [INGR **shattered**′(*y*)]

$$
\begin{aligned}
&\exists e[\textit{causation}(e) \wedge \\
&\quad \exists e' \exists e'' [\text{CAUSE}(e, e') \wedge \\
&\qquad \text{EFFECT}(e, e'') \wedge \\
&\qquad \textit{activity}(e') \wedge \\
&\qquad \exists x[\text{EFFECTOR}(e', x)] \wedge \\
&\qquad \textit{ingr-of-state}(e'') \\
&\qquad \exists s[\text{RESULT}(e'', s) \wedge \\
&\qquad\quad \textit{shattered-state}(s) \wedge \\
&\qquad\quad \exists y[\text{PATIENT}(s, y)]]]
\end{aligned}
$$

logical formula (fist-order logic)



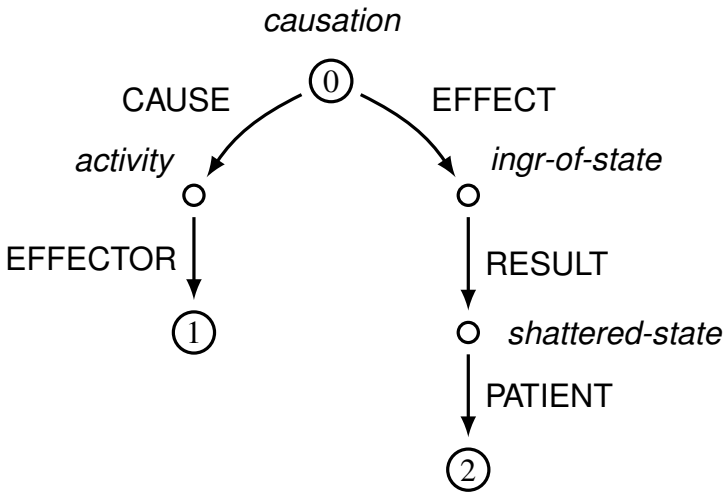
generic model / "**decompositional frame**"

Semantic representation

Decompositional frames $\approx$ multi-base feature structures with sorts and relations

[Kallmeyer & Osswald, submitted]

frame / feature structure



description in attribute-value logic

$\boxed{0} : \textit{causation} \wedge \boxed{0} \cdot \text{CAUSE} : \textit{activity} \wedge \boxed{0} \cdot \text{CAUSE EFFECTOR} \doteq \boxed{1} \wedge$
$\boxed{0} \cdot \text{EFFECT} : \textit{ingr-of-state} \wedge \boxed{0} \cdot \text{EFFECT RESULT} : \textit{shattered-state} \wedge$
$\boxed{0} \cdot \text{EFFECT RESULT PATIENT} \doteq \boxed{2}$

$$\exists e' \exists e'' \exists s (\textit{causation}(\boxed{0}) \wedge \text{CAUSE}(\boxed{0}, e') \wedge \text{EFFECT}(\boxed{0}, e'') \wedge \textit{activity}(e') \wedge \text{EFFECTOR}(e', \boxed{1}) \wedge \textit{ingr-of-state}(e'') \wedge \text{RESULT}(e'', s) \wedge \textit{shattered-state}(s) \wedge \text{PATIENT}(s, \boxed{2}))$$

description in predicate logic

$$\boxed{0} \begin{bmatrix} \textit{causation} & \\ \text{CAUSE} & \begin{bmatrix} \textit{activity} \\ \text{EFFECTOR } \boxed{1} \end{bmatrix} \\ \text{EFFECT} & \begin{bmatrix} \textit{ingr-of-state} & \\ \text{RESULT} & \begin{bmatrix} \textit{shattered-state} \\ \text{PATIENT } \boxed{2} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

attribute-value matrix notation

Semantic representation

Advantages of decompositional frames

Frame representations allow us to combine two key aspects of RRG's template-based structures and genuine logical representations:

- Like decompositional templates they are **concept-centered** and have inherent structural properties.

  I.e., **structural positions** relevant to the **linking** between syntax and semantics are accessible by attribute paths.

- Like logical representations, frame descriptions have a **well-defined** model-theoretic **interpretation**, and they are easily **extensible** by additional subcomponents and constraints.

Moreover:

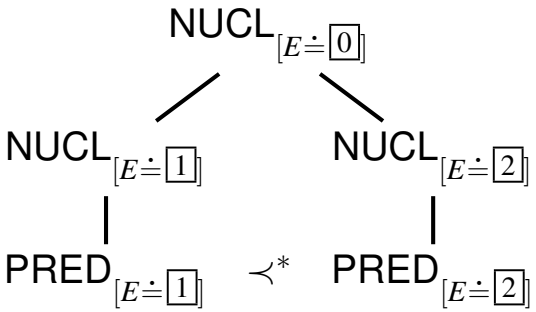
- Subcomponents of frames can be **unified** with other frames (as, e.g., triggered by syntactic substitution) through base label identification.

Semantic representation

Linking sketches

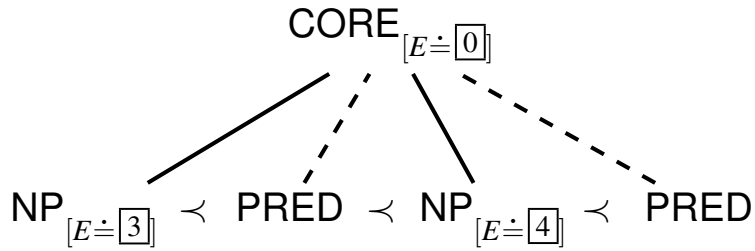
Adjectival resultative construction in English (*wipe clean*, *paint white*, ...)

resultative-nuc-cosubord



$$\boxed{0}\begin{bmatrix} \textit{causation} \\ \text{CAUSE} \quad \boxed{1} \\ \text{EFFECT} \quad \begin{bmatrix} \text{RESULT} \quad \boxed{2}\ \textit{state} \end{bmatrix} \end{bmatrix}$$

serial-pred-core



$$\boxed{0}\begin{bmatrix} \text{ACTOR} & \boxed{3} \\ \text{UNDERGOER} & \boxed{4} \end{bmatrix}$$

$$\text{CAUSE EFFECTOR} : \top \rightarrow \text{CAUSE EFFECTOR} \doteq \text{ACTOR}$$

$$\text{EFFECT RESULT PATIENT} : \top \rightarrow \text{EFFECT RESULT PATIENT} \doteq \text{UNDERGOER}$$

Summary

Outline of a formalization of RRG

- Identify the **elementary syntactic trees** and characterize them as combinations of **tree constraints** in the **metagrammar**.
- Describe the combination of elementary trees by a small set of general **tree operations**.
- Re-analyze the logical structures of RRG as (descriptions of) **decompositional frames**.
- Draw a distinction between **frame constraints** and associated **generic models** similar to what is proposed for the syntax.
- Combine tree operations in the syntactic dimension with **frame unification** in the semantic dimension.
- Characterize the syntax-semantics interface in the metagrammar; (try to) capture **linking constraints** in metagrammar classes.

References

Bangalore, S. & A. K. Joshi (2010). Introduction. In *Supertagging: Using Complex Lexical Descriptions in Natural Language Processing*, pp. 1–31. MIT Press.

Crabbé, B. & D. Duchier (2005). Metagrammar redux. In *Constraint Solving and Language Processing*, pp. 32–47. Springer.

Guest, E. (2008). Parsing for Role and Reference Grammar. In *Investigations of the Syntax-Semantics-Pragmatics Interface*, pp. 435–454. John Benjamins.

Joshi, A. K. & Y. Schabes (1997). Tree-adjoining grammars. In *Handbook of Formal Languages. Vol. 3*, pp. 69–123. Springer.

Kallmeyer, L. & R. Osswald (2012). A frame-based semantics of the dative alternation in Lexicalized Tree Adjoining Grammars. In *Empirical Issues in Syntax and Semantics 9*, pp. 167–184.

Kallmeyer, L. & R. Osswald (2012). An Analysis of Directed Motion Expressions with Lexicalized Tree Adjoining Grammars and Frame Semantics. In *Proceedings of WoLLIC 2012*, pp. 34–55. Springer.

Kallmeyer, L., Osswald R. & R. D. Van Valin, Jr. (2013). Tree Wrapping for Role and Reference Grammar. In *Proceedings of Formal Grammar 2013*, pp. 175–190. Springer.

Kallmeyer, L. & R. Osswald (submitted). Syntax-Driven Semantic Frame Composition in Lexicalized Tree Adjoining Grammars.

Nolan, B. (2004). First steps toward a computational RRG. In *Proceedings of the International Role and Reference Grammar Conference 2004*, pp. 196–223.

Osswald, R. (1999). Semantics for Attribute-Value Theories. In *Proceedings of the 12th Amsterdam Colloquium*, pp. 199–204.

Osswald, R. & R. D. Van Valin, Jr. (in print). FrameNet, Frame Structure, and the Syntax-Semantics Interface. In T. Gamerschlag et al., *Frames and Concept Types*, Springer.

Periñán, C. & R. Mairal (2009). Bringing Role and Reference Grammar to natural language understanding. Procesamiento del Lenguaje Natural, 43:265–273.

Rambow, O., K. Vijay-Shanker & D. Weir (1995). D-Tree Grammars. In *Proceedings of ACL*.

Rounds, W. C. (1997). Feature logics. In *Handbook of Logic and Language*, pp. 475–533. North-Holland.

Van Valin, Jr., R. D. (2005). *Exploring the Syntax-Semantics Interface*. Cambridge University Press.

Van Valin, Jr., R. D. (2006). Semantic macroroles and language processing. In *Semantic Role Universals and Argument Linking*, pp. 263–301. Mouton de Gruyter.