Cascades, TTs, and dot types

Sebastian Löbner

Event semantics 2018
Heidelberg, 09 – 10 Nov. 2018
1. Act-cascades revisited

1.1 Multi-level categorization of action: examples

- ruin one’s night
- wake up the baby
- lighten the room
- turn on the light
- flip the light switch
- get a smile from y
- do y a favor
- let y pass
- keep the door open
- purchase z
- pay for z
- hand money to y
- disappoint y
- decline y’s request
- say “No” to y
1.2 Act-tokens, act-types, act-TTs

An act-token is an instantiation of an act-type.
Example: Amy’s opening the door at 2:03 p.m. is a token of the act-type ‘open the door’.
An act-type can have an open number of act-tokens.
Act-tokens are located in time and space and have a particular agent.

An act-TT is a token of an act-type. Notation: a/A.

➢ Act-TTs are categorized act-tokens.
➢ Whenever we verbally refer to an act, we refer to an act-TT: no reference without some kind of categorizing description
➢ Whenever we think of an act, we think of an act-TT.

In general ‘x/Y’ stands for: “entity-x-under-the-type-description-Y”
1.3 C-constitution

Definition (informal) [Based on the notion of “level-generation” in Goldman (1970)]
Let a1/A1 and a2/A2 be act-TTs with the same agent and the same action time.
Under circumstances C, an act-TT a1/A1 c-constitutes (“level-generates”) an act-TT a2/A2

   a1/A1 c-const a2/A2     or     a1/A1 ↑ a2/A2

iff under the circumstances C, a2/A2 is done by doing a1/A1, or in doing a1/A1.

For example: Under circumstances

   a1/keep the door open ↑ a2/let y pass ↑ a3/do y a favor ↑ a4/get a smile from y

   a1/say “No” to y ↑ a2/decline y’s request ↑ a3/disappoint y
1.4 Cascades

- The relation \( c-\text{const} \) is **irreflexive**: If \( a_1/A_1 \uparrow a_2/A_2 \), then \( a_1/A_1 \) and \( a_2/A_2 \) are different.

- The relation \( c-\text{const} \) is **transitive**: If \( a_1/A_1 \uparrow a_2/A_2 \) and \( a_2/A_2 \uparrow a_3/A_3 \), then \( a_1/A_1 \uparrow a_3/A_3 \).  
  - \( c-\text{constitution forms chains} \)
  - \( \text{several steps of } c-\text{constitution form one (larger) step} \)
  - \( c-\text{constitution may be broken down into finer steps} \)

- The relation \( c-\text{const} \) is **asymmetric**: If \( a_1/A_1 \uparrow a_2/A_2 \), then *not* \( a_2/A_2 \uparrow a_1/A_1 \).

- The relation of \( c-\text{const} \) gives rise to **tree structures**.

**Definition**

A **cascade** is an act-tree generated by \( c-\text{constitution} \).
1.5 Simple conditions

(1) Logical independence
If a₁/A₁ c-const a₂/A₂, then A₁ ⊈ A₂ and A₁ ⊄ A₂:

If a₁/A₁ level-generates a₂/A₂, then A₁ and A₂ don’t subsume each other.

(2) Identical temporal extension

(3) Quasi identity
If a₁/A₁ c-const a₂/A₂, then the A₁ act – quasi – is also an A₂ act.

(4) Contingent dependency at token level
If a₁/A₁ c-const a₂/A₂, then an act of type A₁ is necessary for an act of type A₂ to exist:

No A₁, no A₂.
1.6 Frames

We represent act-types as frames.

- Düsseldorf Barsalou frames are representations of TTs: representations of a type (= a category) by description of a token (= a member of the category)

![Diagram of a frame for the act-type keep a door open](image)

central node: act-token of the type keep a door open

frame for the act-type keep a door open
1.7 Cascades of frames

We represent cascades as trees of frames, built up by c-constitution – *a relation between frames*.

![Diagram of cascades of frames]

Level 3
- A3 do y a favor

Level 2
- A2 let y pass

Level 1
- A1 keep door open

Sebastian Löbner “Cascades, TTs and dot types” Event semantics, Heidelberg 2018
1.8 The cascade for ‘write’
1.9 Reference

• The levels selected by the argument specifications, modifiers, and other adjuncts may not be the same.

• Statements about the writing of somebody may relate to several levels simultaneously.

  *My grandmother used to write her personal letters on her typewriter.*

  Reference is to all cascade levels simultaneously because all that happens in one. The lower levels are conceptually necessary for the higher levels to exist.
1.10 Cascades in cognition and ontology

➤ **Hypothesis**
Whatever we categorize we categorize at potentially more than one cascade level.

- The bits and pieces of what is reality to us as human cognitive individuals always matter in many different contexts.

- There may be macro-levels across action and role concepts, such as
  - the personal level of individual appraisal;
  - levels of social interactions, relationships, and institutions;
  - levels of abstract reasoning

➤ **Hypothesis**
Level-generation is a basic brain mechanism.
2. Roles and the extension of c-constitution to act arguments

2.1 Product specification

- Product specification **selects the level** where it saturates an argument of the verb.
- Saturation at a higher level constrains the argument types at the lower levels.
- Most object specifications are level-selective.
2.2 Roles of objects and c-implementation

Let $a_1/A_1$ c-constitute $a_2/A_2$. Let $x_1$ and $x_2$ are arguments in the same general role, of $a_1/A_1$ and $a_2/A_2$, respectively, then $x_1$ in $a_1/A_1$ c-constitutes $x_2$ in $a_2/A_2$.

For example: The graphemes produced at Level 2 of writing, under circumstances, constitute text.

The text produced at Level 3 of writing, under circumstances, constitutes content.

“$x$ in semantic role $R$ in $e/E$” is an object-under-description: $x/R(e/E)$

- The informal properties of correlates under c-constitution hold here, too. (logical independence, sloppy identity, existential dependence)
2.3 Agent specification

Deviating from Goldman’s notion of act-trees, we may observe: The agents involved in writing are not necessarily the same person.

We may assume that the agent roles in a cascade can be delegated down from the content level agent. (Cf. the notion of “footing” in Goffman 1979)

<table>
<thead>
<tr>
<th>Script</th>
<th>Text</th>
<th>Content</th>
<th>Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>DT writes a letter to Putin</td>
<td>-</td>
<td>-</td>
<td>+</td>
</tr>
<tr>
<td>X writes a letter to Putin</td>
<td>-</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td>Y writes a letter to Putin</td>
<td>+</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
2.4 Roles of agents and c-implementation

If \( a_1/A_1 \) c-constitutes \( a_2/A_2 \), then
the agent of \( a_1/A_1 \) c-implements \([c\text{-constitutes}]\) the agent of \( a_2/A_2 \).

For example:
- Carl = AGENT(a3/write_{text} the speech)
- Joey = AGENT(a4/wake up the baby)
- Sam = BENEFICIARY(a3/do y a favor)

- The informal properties of correlates under c-constitution hold here, too.
  (logical independence, sloppy identity, existential dependence):
  - an act of type A1 need not be an act of type A2 and vice versa
  - in the given constellation, the role incumbent and the role implemented are cotemporal.
  - If \( a_1/A_1 \) c-impl \( a_2/A_2 \), then the A1 act – quasi – is also an A2 act.
  - If there is no p1/AGENT(a1/A1), then there is no p2/AGENT(a2/A2)
2.5 C-constitution as a multi-track relation between act frames
3. Dot types

3.1 Characterization

[following Pustejovsky 2009, Asher 2011]

- “Dot types” are complex types, written “X⦁Y” for things that are of either type or of both types
- Assigned to words, each type within a dot type is the type of one sense of the word.
- The dot is neither conjunction nor disjunction.
- For each sense pair, there is a relation which "connects" the senses in a well-defined way.

- “Dot objects”, a.k.a. “dual aspect objects”, are things of dot types, for example x⦁y of type X⦁Y. The components of dot objects are TTs: x/X and y/Y.

- Evidence for dot types:
  Applicability of predications with incompatible selectional restrictions (e.g. copredication).
3.2 Major types of example

- **book**: physobj
  - information
  - *she burnt the book ...* after reading
  - *physobj contains info* [Pustejovsky 2009]

- **lunch**: event
  - food
  - *the lunch won’t take long* I have my lunch with me
  - *food theme of event* [Pustejovsky 2009]

- **coffee**: fruits
  - liquid
  - *pluck / roast the coffee* pour / drink the coffee
  - *fruits processed into liquid* [Pustejovsky 2009]

- **bronze**: matter
  - object
  - *lump of bronze* a bronze by Bernini
  - *artifact consists of matter* [Asher 2011]

- **keep a promise by dancing**: action
  - *he kept his promise ...* by dancing with him
  - *action realizes action* [Bücking 2014]
3.3 The metonymic subgroup

- Deverbal nouns are considered dot object nouns:
  construction classification examination appointment inflammation belief promise

- Event denoting nouns
  exam lunch laundry concert symphony class lecture

  - The event frame provides a **metonymical relation** between the dotted types,
    in most cases a relation between the event and one of its arguments: the relation is the
    thematic role.

- Other dot nouns with metonymical relation
  book newspaper university library

  - container ↔ content and various other relationships
3.3 The metonymic subgroup (ctd.)

- Metonymy is essentially based on an attribute relation within a frame:
The central node is shifted to a dependent node, i.e. to the value of an attribute. [Löbner 2013, §12]
3.3 The metonymic subgroup (ctd.)

• With a frame, not only the central node is given, but also everything connected to it:
  With a lunch/event  the lunch/food is given – as the value of the THEME attribute.

• The dot relation is the attribute relation.

• The frame for the related sense is the same frame with the referential node shifted.

• Almost all deverbal nouns have a frame resulting from the verbal event frame by a simple such shift of the central node.

• The same type of shift underlies all metonymies, such as
  Honda    producer ↔ product
  newspaper institution ↔ printed copy
  library   book collection ↔ building
  ...

Sebastian Löbner “Cascades, TTs and dot types” Event semantics, Heidelberg 2018
3.4 The processing subgroup

*coffee  chicken  bronze*

• The referents of the two types are related by a process that turns the first into the second

• The two correlates are related by a *processing* frame with attributes for the INPUT and the OUTPUT of processing.

• Unlike with the metonymic type,

  *related tokens of the two types do not co-exist at the same time,*

  but constitute different stages of matter.
4. Dot types and cascades

Are dot types related by c-constitution?

The members of a dot type always fulfil the criterion of logical independence.

4.1 Dot types not related by (generalized) c-constitution

- Dot objects of the processing type are not related by c-constitution, as the correlates do not exist at the same time.

- Dot objects of the various metonymic types have identical temporal extension (roughly), but lack the property of quasi identity:
  the examination/event is not [in any sense] the examination/subject
  the book/physobj is not the book/content,
  and so on
4.2 Dot types related by (generalized) c-constitution

- Bücking’s *action*•*action* dot objects are instances of cascades. The *by* or *inden* relation immediately corresponds to level generation / c-constitution: if y/Y is done by doing x/X, then x/X c-constitutes y/Y.

- Dot objects of the role type are related by c-constitution, “Sam as a lawyer” is Sam implementing a lawyer: Sam and the lawyer he implements form a cascade.

- There are also objects in roles generated by c-constitution: the *burger* as a meal, the *screwdriver* as a chisel, the *car* as a vehicle

- Some types of dot objects are variations of (parts) of Austin’s speech act cascade (generalized):

  phonetic act ↩ rhetic act ↩ illocutionary act
  
  *lecture* *novel* *interview* *question* *music* *concert*
4.2 Dot types related by (generalized) c-constitution (ctd.)

Material constituency can be considered another variant of c-constituency.

- **Mary and John [as a couple]**
  Under circumstances, Mary and John constitute a couple / form a couple (= a social unit)

- **coin money**
  categories of social objects:
  Under circumstances (including social conventions and institutions) certain objects count as social objects, e.g. money. A piece of metal may c-constitute a coin (and thereby money).

- **ring**
  Artefacts
  Under circumstances (i.e. if they are produced accordingly), artefacts constitute a quantity of material constitutes an artefact: a gold ring, a silk cloth.
  Unlike in the processing examples, both coexist.
5. Cascades and dot objects

5.1 All cascades can be considered dot types / dot objects.

- Cascades present simultaneous type assignment to one underlying entity.
- Predications can address a single level within a cascade / a single type.
- The relation connecting the correlates is – invariably – c-constitution.
- The “NP as a N” relation is indicative of a role cascade.
- Cascade dot-objects are restricted by the “simple conditions” mentioned above, including temporal coextensionality.
5.2 Not all dot objects are cascades

- The metonymic type is not.
- The processing type is not.
- There are more types that are not, for example
  the *temperature* type measure•value [Pustejovsky 2005]
References


Acknowledgment

I am very grateful to Sebastian Klinge for collecting a comprehensive survey of examples of dot objects in the literature.

Sebastian Lübner “Cascades, TTs and dot types” Event semantics, Heidelberg 2018