THE EVENT ARGUMENT
and ARGUMENT INTRODUCERS: little ν, and the Applicative Head

\[ \lambda e \langle s, t \rangle \nu \text{ Appl}^\circ \]
OUR ROADMAP

- Review of morphosyntactic function of v
- Adding “events” to our notation
- How little v came to be
- The high and low Applicative° heads
How We Came to Know \( v \) as Our Morphosyntactic Friend

The Accusative Case-Assigner

- **Burzio’s Generalization**: If a verb assigns accusative case, then it assigns an external \( \theta \)-role.
  - No semantic subject \( \theta \)-role \( \rightarrow \) no accusative Case
  - *In essence*: When a verb phrase combines with a head that introduces an external argument, that head assigns accusative case to the internal argument of the verb.

- In b/d there’s only an internal argument, and the internal argument surfaces in subject position.

  a. She fired me.
  b. I was fired. (from Burzio 2000)
  c. They broke the window. (EX26)
  d. The window broke. (EX 2)

- Normal \( v \): the accusative case assigner
- The many flavors of \( v \)
  - \( v \)-ergative
  - \( v \)-dative
  - \( v \)-genitive
Run of the Mill Transitive Sentences

Cherlon cooked a fantastic meal.

$v$ has two functions:
- It assigns accusative case to the object
  - The syntactic job
- It provides a home - via its specifier - for the verb’s external argument
  - The semantic job

NOTE: In Kratzer 1996, there are trees in which the direct object occupies the specifier of VP. We’ll get to that.
“Normal” v’s and “Special” v’s

Icelandic
1. a. Við lásun bókina.
   ‘We read the book.’ (Sigurðsson 1996, Ex 14)
   --Normal v: assigns accusative to the object
   b. Einum málfræðingi líkuðu þessar hugmyndir.
   ‘One linguist liked these ideas.’ (Sigurðsson and Holmberg 2008, Ex 1)
   --Special v: assigns dative to the subject

Gujarati
2. a. Sudha away-i.
   ‘Sudha came.’
   --Normal v: there’s no object for v to assign case to
   b. Sudha-e radio khāridy-o.
   ‘Sudha bought a radio.’ (Woolford 2006, EX 38c/39)
   --Special v: assigns ergative to the subject

Normal

TP
  T'  vP
    T[Nom]
      vP
        DP
          v[Acc]
            VP
              v'
                DP

Special

TP
  T'  vP
    T[Nom/Abs]
      vP
        DP
          v'[Dat] or [Erg]
            VP
              v'
                DP

Ling 340 - Spring 2017 - C. Ussery
**English**
The students gave their professor a necklace.

**Icelandic**
Ég sendi Hildi fiskinn
   ‘I sent Hildur the fish.’

**Greek**
a. Tha mu to stilune fut cl.Gen.sg.1 cl.Acc.sg.3.neut send.3.pl
   ‘They will send it to me.’
b. Tha su to stilune fut cl.Gen.sg.2 cl.Acc.sg.3.masc send.3pl
   ‘They will send him to you.’
c. *Tha su me sistisume fut cl.Gen..sg.2 cl.Acc.sg.1 introduce.3.pl
   ‘They will introduce me to you.’
d. *Tha tu se stilune fut cl.Gen.masc.sg.3 cl.Acc.sg.2 send.3.pl
   ‘They will send you to him.’ (Bonet 1991:182)

---

Memory jogger/Brief Overview of the Person Case Constraint

- Some languages restrict the person combinations of the direct and indirect object.
- In Greek, the direct object clitic cannot be first or second person.
- Analyses of PCC effects (in general, not just particular to Greek) generally argue that because the “special” little \( v \) intervenes between (the normal) \( v \) and the accusative, something goes awry with the relationship between \( v \) and the direct object.
- As a result, the direct object is not allowed to have 1\textsuperscript{st}/2\textsuperscript{nd} person features.
SUMMARY OF MORPHOSYNTACTIC FUNCTION OF v

- Burzio’s Generalization:
  - “Normal” v assigns Accusative case to direct objects.
  - The semantic subject is merged (starts off in) in the specifier of v.
  - No accusative on the semantic objects in passives and unaccusatives.

- Special v’s are proposed to assign case to the DP in their specifier.
  - Dative subjects
  - Ergative subjects
  - Indirect objects of various cases

- ...BUT how did v come to exist???
**Enter Semantics: An Overview of Events and vP**

- **Marantz (1984):** The entire predicate (not just the verb) assigns a θ-role to the subject. The meaning/properties of the object influence the meaning of the verb, which, in turn, determines the semantic properties of the subject.

  - The object is assigned its semantic role by the verb. The subject is assigned its semantic role by the entire predicate.

1. a. throw support behind a candidate  d. throw a party  
   b. throw a baseball  
   c. throw a boxing match (take a dive)  
2. a. take a book from the shelf  d. take an aspirin  
   b. take a bus to New York  
   c. take a nap  
3. a. kill a cockroach  
   b. kill a conversation  
   c. kill an evening watching TV  
   d. kill a bottle (empty it)  
   e. kill an audience (wow them)  

(Kratzer 1996, EX 6-8)
Kratzer argues that we need another syntactic position for the subject. Why? Just how many arguments does a verb have?

From a syntactic perspective, we’re used to thinking about a verb’s arguments as the DPs/CPs/PPs that the verbs subcategorizes for.

From a semantic perspective, verbs refer to events and in order to derive the meaning of a verb, the event that the verb refers to is part of the verb’s meaning.

So, verbs take an argument that is an event.

The semantic function of $v$ is to “introduce” the external argument to the event encoded in the VP.
Kratzer’s Primary Arguments

- Argument/event structure interacts with the syntactic structure (contra previous proposals).
- Verb meanings are functions
- The “agent” is not part of the meaning of a verb
- The “agent” argument is introduced by a functional head that is higher than VP and lower than TP.
- The semantic operation Event Identification combines the “agent” with the VP.

**Severing the external argument from its verb**

\[ \lambda x \lambda e \ [ \text{buy}(x)(e) ] \] or
\[ \lambda x \lambda e \ [ \text{buying}(e) & \text{Theme}(x)(e) ] \]

**The verb has two arguments**
- \( e \): the event of buying
- \( x \): the thing that was bought
Is the head that introduces the external argument lexical or functional? A detour through Malagasy.

- **Hung 1988**: In Malagasy, the agent is introduced by the prefix \(-an\) that resides in a V head that is higher than the V head which hosts the verb.

- \(-an\) does two things:
  - Introduces the subject argument.
  - Assigns case to the object - *ny lamba* ‘the clothes’.

- For Hung, the external argument introducing head is a V - it’s lexical.

**This sounds a lot like** $v$ !!!

**Head Movement**: $V \rightarrow V \rightarrow T$

**Phrasal Movement**: The subject starts in the specifier of the higher VP and moves Spec,IP
Kratzer argues that external argument-introducing head in English is **functional** - not lexical.

**The data argument:** The head that introduces the external argument is not present in *of* gerunds, so that head should not be lexical.
- His rebuilding of the barn took five months.
- *Rebuild* is a V that is nominalized. *The barn* gets case from the preposition here.
- There is no Voice head. Therefore, no external argument.
- *He rebuilding of the barn took five months.*

**The theory argument:** Structural vs lexical case. Structural case is assigned by functional heads.
- Structural case: The normal/expected case. E.g. - nominative on subjects and accusative on objects.
  - The functional head I/T assigns nominative. Therefore, the case on the object should also come from a functional head.
- Lexical case: Cases that are determined by specific lexical items. E.g. - Prepositions in lots of languages - German, Russian, Icelandic - determine the case of their objects.
  - The lexical preposition assigns case to its object.

Fortunately, at the time, there was a functional head hanging around just waiting to be given more work to do.
Enter $\mu$

- Johnson 1991: objects move to the specifier of VP and are assigned (accusative) case by $\mu$. The verb moves and adjoins to $\mu$.
- Only NPs (DPs) move to Spec,VP. Therefore, they precede adverbs and other complements of verbs.
  - Mittie fed the dog quickly.
  - *Mittie fed quickly the dog.
  - Mikey visited his parents quietly.
  - *Mikey visited quietly his parents.
  - Gary told Sam to leave.
  - *Gary told to leave Sam. (EX 20)
- For Johnson, $\mu$’s sole function is case assignment.
  - NOTE: At the time, it was assumed that case was assigned in a very local configuration. Now, it is assumed that the case-assigning and the DP can be farther away in the structure.

For Kratzer, $\mu$ has another role...

“Having acquired semantic content, $\mu$ deserves a meaningful name. I will call it VOICE.” (p. 120)

An Aside:
- Kratzer assumes that since the external argument is an argument of the verb and is generated in Spec,VoiceP, the direct object is generated in Spec,VP, since it is an argument of the verb.
- While Johnson moves the object from the complement of V to Spec,VP, Kratzer starts the object in Spec,VP.
**Fast Forward to Today**

- **Kratzer’s VoiceP is generally referred to as vP.** It’s sometimes handy to posit both VoiceP and vP.
  - E.g. There is one way of thinking about passives in which the agent occupies Spec,vP - just as in actives - and *by* overtly or silently occupies Voice.

- **µ died a long time ago.** (But Kyle Johnson is very much alive!)

- **There are many ν’s,** as we’ve seen. Kratzer actually suggests that there might be more than one Voice head – e.g. an active voice head and a stative voice head. (p.123)

- Some syntacticians/semanticists place the direct object in Spec,VP and others position it as the sister to V. Unless a particular point is being made about object positions, **either way works.**
Technical Details

- **Functional Application:** This is the normal way of composing meaning. There is an open argument slot and the sister node saturates (provides the content for) that slot.

- **Passing Up:** When node doesn’t have a sister, then the meaning of that node travels up to the next highest node.

- **Event Identification:** This operation combines the external argument with the event denoted in the VP.
  - There is an agent and that agent is the agent of the event described in the VP.
  - “Event Identification makes it possible to chain together various conditions for the event described by a sentence.” (p. 122)

The Compositional Operations

Example of Event Identification

\[
\begin{align*}
  f & \quad g \\
  \langle e, <s, d> \rangle & \quad \langle s, d \rangle \\
  \lambda x. \lambda c_s [\text{Agent}(x)(e)] & \quad \lambda c_s [\text{feed}(\text{the dog})(e)] \\
  h & \\
  \langle e, <s, d> \rangle & \\
  \lambda x. \lambda c_s [\text{Agent}(x)(e) \& \text{feed}(\text{the dog})(e)] \\
\end{align*}
\]

Mittie fed the dog.
Tense and the Event Argument

• \( v \) has to be higher than VP because its role is to relate the agent to the event denoted in the VP.

• \( v \) has to be lower than Tense. Kratzer proposes that the role of tense is to “existentially quantify” the event argument.

• Kratzer proposes the meaning in (31) for tense. (The * isn’t relevant here.)

\[
\text{(31) } \text{past} - * = \lambda p_{s,t} \exists e [ P(e) & \text{past}(e) ]
\]

• There is some proposition (a state of affairs), \( \lambda p \), and there exists at least one event, \( \exists e \), such that the proposition encodes the event and the event happened in the past.

• Tense combines with VoiceP and the meaning of VoiceP saturates \( \lambda p \).

\[
T' = \exists e. \text{feed (the dog)}(e) & (\text{agent})(\text{Mittie})(e) & \text{past}(e) \ <s,t>
\]

• There is at least one event of feeding the dog and of which Mittie is the agent and which happened in the past.

• At this point, the real-world situation that maps to the sentence saturates \( \exists e \) and we end up with the meaning of the sentence.

• TP is our truth value.
Mapping the event argument to events

This sentence has multiple interpretations - collective, cumulative, and distributive.

1. Three architects designed four buildings.

In both the collective and cumulative, the total number of buildings designed is four.

Collective: all three of the architects collaboratively design all four buildings.

Cumulative: it is underspecified as to how many architects design each building. Perhaps one architect designed three buildings and the other two designed one building or all three architects collaborated on one building and each of them individually designed the remaining three buildings.

Distributive: there are twelve buildings; each architect designs four buildings.

The events that the collective, cumulative, and distributive interpretations map to are independent of the meaning delivered by the combination of the verb and the object.

In the collective interpretation, we need one event in which all three architects design four buildings. I In the distributive interpretation, we need three events in which each architect designs four buildings.

The cumulative interpretation is more complicated; we don’t know how many events there are. We just know that some subset of architects designed some subset of buildings.

Whatever the event is, it supplies the value for the event argument within the TP projection.

See Kratzer’s chapters The Event Argument and the Semantics of Verbs posted on the Semantics Archive for much more detailed discussion of quantifying over events. http://semanticsarchive.net/Archive/GU1NWM4Z/
Voice was proposed in order to build into the syntax the semantic observation that verbs have an asymmetric relationship with their objects versus their subjects.

- Objects have a closer relationship with the verb and the compositional meaning of the verb plus its object select for the subject.
  - The subject is external to the meaning of the verb.
- Voice builds on $\mu$, which was proposed (by a syntactician) as the accusative case-assigner.

Kratzer proposes Event Identification, which combines the external argument with the denotation of the VP.

In contemporary theory, Kratzer’s voice is generally represented as $v$ and voice is used to encode other kinds of information.

There are many $v$’s in contemporary theory.

- $v$-ergative/dative/genitive all introduce an argument into the structure and they assign case to that DP.
- The tense head combines with vP and the event argument is saturated at the TP level.
Some Practice

Part 1. Draw a tree which includes both type theory and lambda notation up to TP. You can triangle the DPs and label them as type e.

1. Three architects designed four buildings.
2. The professor wore a new necklace.
3. The exhausted athletes sleep.
4. The ice melted.

Part 2. For these trees, just do type theory. NOTE: Think carefully. You will have to propose types for some words.

5. Many contemporary chefs believe people prefer locally-grown food.
6. Mary claims John saw her duck. [Remember this one? It’s ambiguous. Draw two trees.]
7. The delicious sushi made Cherlon happy.
8. Icelandic chefs seem to attend exquisite culinary schools.
9. Cherlon refused to eat pulverized bacon.
10. *It refused Cherlon to eat pulverized bacon. [In terms of type theory, describe what goes wrong with this sentence. You don’t have to draw a tree.]
Argument Structure and the Applicative Head
English and Venda (a Bantu language) both allow *melt* in the same argument structure configurations - (1)/(2).

(1) **English**
   
   a. The ice melted.
   
   b. *John* melted the ice.
   
   c. *John* melted *me* some ice.

(2) **Venda**
   
      snow 3SG.PAST-melt-FV
      ‘The snow melted.’
   
   b. Mukasa o-nok-is-a
      Mukasa 3SG.PAST-melt-CAUSE-FV snow
      ‘Mukasa melted the snow.’
   
   c. Mukasa o-nok-is-el-a
      Mukasa 3SG.PAST-melt-CAUSE-APPL-FV Katonga snow
      ‘Mukasa melted Katonga the snow.’
But, Venda allows some intransitives (unergatives) such as *laugh/speak* to take applicatives, while English does not - (3)/(4).

Pylkkänen (2008) proposes that there is an **applicative head** which introduces the applicative argument.

**(3) Venda**

a. Mukasa o-se-is-a Katonga.
   Mukasa 3SG.PAST-laugh-CAUSE-FV Katonga
   ‘Mukasa made Katonga laugh.’

b. Mukasa o-amb-el-a Katonga.
   Mukasa 3SG.PAST-speak-APPL-FV Katonga
   ‘Mukasa spoke for Katonga.’

**(4) English**

a. *Mary laughed Sue.*
   (Intended meaning: ‘Mary made Sue laugh.’)

b. *Mary spoke Sue.*
   (Intended meaning: ‘Mary spoke for Sue.’)

**High applicatives** (HA): the applicative head attaches above the verb.

There is a relation between an individual/entity and an event. =Venda

**Low applicatives** (LA), the applicative head attaches below the verb.

There is a relation between two individuals/entities. The applicative is either the recipient or the source. =English

LA requires a direct object because the relationship is between the direct object and the applicative argument.
Mary bought John a book.

- The applicative, John, has a relation with the direct object.
- John is the intended recipient of the book.

Wife has a benefactive relation to the event of eating but no relation to the object of eating, food. Likewise, friend benefits from the event of running.

A high applicative head is very much like the external-argument-introducing head: it simply adds another participant to the event described by the verb.” (Pylkkänen 2008:14)
Why Examine Applicatives?

“A comprehensive theory of linguistic representation must minimally:

(i) define the nature of the primitive building blocks that enter into linguistic computation,
(ii) characterize the manner in which the basic units combine into complex representations,
and (iii) identify the ways in which languages may differ with respect to their inventory of possible representations.” (p.1)

Pylkkänen will:

- Argue for specific heads in the syntax (the primitive building blocks);
- Propose denotations for those heads (which encode how those heads combine with other building blocks); and
- Illustrate a typology of those heads (showing how languages differ).

Like Kratzer:

- Pylkkänen assumes a very tight connection between syntax and semantics. “..syntactic structure building is the only mode of structure building in natural language.”(p.5)
- Pylkkänen builds on Kratzer’s work directly: “Thus, one of the main contributions of this book is to provide a new empirical argument for separating the external argument from its verb.” (p.7)
Back to Our Compositional Operations

- **Functional Application:** The standard way of composing meaning. There is an open argument slot and the sister node saturates (provides the content for) that slot.

- **Passing Up:** When node doesn’t have a sister, then the meaning of that node travels up to the next highest node. [Pylkkänen doesn’t use this operation. There aren’t empty nodes in her derivations.]

- **Event Identification:** Combines the external argument - the agent or the high applicative - with the event denoted in the VP.
  
  “Event Identification makes it possible to chain together various conditions for the event described by a sentence.” (Kratzer 1996 p.122)

- **Predicate Modification:** This is new for us. This operation combines two daughters of the same type and returns a value of that type. Like Event Identification, it’s a conjunction operation of sorts.
  
  For the Applicative analysis, the two nodes that are combined both have type \(<e, <s,t>>\). They combine and their mother has type \(<e, <s,t>>\).
Big Difference Between the Denotations for the High and Low Applicative Heads

High Appl... is like Voice in that it adds an entity to an event denoted in the VP and relates that entity to the event in the VP.

(4) Chichewa instrumental
Mavuto a-na-umb-ir-a mpeni mtsuko.
Mavuto SP-PAST-mold-APPL-ASP knife waterpot
‘Mavuto molded the waterpot with a knife.’
(Baker 1988b, 354)

\[ \lambda x. \lambda e. \text{Appl}(e, x) \]
(collapsing \( \text{AppL}_{\text{Ben}}, \text{AppL}_{\text{Instr}}, \text{AppL}_{\text{Loc}}, \text{etc.} \))

✓ This looks a lot like the denotation for Voice!
✓ High Appl has 2 arguments: the applicative \([\lambda x]\) and the event \([\lambda e]\).
Appl...

Low

- is not like Voice. There is no semantic relationship between the Applicative and the event.
- The relationship is between the Applicative and the direct object.

(7) Low recipient applicative: English
a. I wrote John a letter.  
   ‘I wrote a letter and the letter was to the possession of John.’

b. I baked my friend a cake.  
   ‘I baked a cake and the cake was to the possession of my friend.’

c. I bought John a new VCR.  
   ‘I bought a new VCR and the VCR was to the possession of John.’

(15) a. Low-\text{Appl}_{To} (Recipient applicative)  
   $\lambda x.\lambda y.\lambda f_{e,\langle s,t\rangle}.\lambda e. f(e, x) \& \text{theme}(e, x) \& \text{to-the-possession}(x, y)$

b. Low-\text{Appl}_{From} (Source applicative)  
   $\lambda x.\lambda y.\lambda f_{e,\langle s,t\rangle}.\lambda e. f(e, x) \& \text{theme}(e, x) \& \text{from-the-possession}(x, y)$

✓ This does not look like the denotation for Voice!
✓ Low Appl has 4 arguments: the direct object $[\lambda x]$, the applicative $[\lambda y]$, the verb $[\lambda f]$ (a function), and the event $[\lambda e]$. 
High Appl Derivation

The Steps:

- The transitive verb combines with the direct object.

- The Appl head combines with the VP. Just like Voice, the High Appl head introduces a spot for an entity argument to saturate. Here, it’s the applicative.

- The Applicative is merged in the specifier of ApplP and saturates the λx slot.

- The Voice head combines with ApplP and introduces a spot for the subject external argument.

- The subject is merged in in the specifier of VoiceP and saturates the λx slot.

\[
\lambda x.\lambda e.\text{eating}(e) & \land \text{theme}(e, \text{food}) \land \text{benefactive}(e, \text{wife}) \\
\lambda x.\lambda e.\text{agent}(e, x) \\
\text{he} \\
\text{VoiceP} \\
\text{Voice'} \\
\lambda x.\lambda e.\text{eating}(e) & \land \text{theme}(e, \text{food}) \land \text{benefactive}(e, \text{wife}) \\
\text{VP} \\
\text{ApplP} \\
\lambda x.\lambda e.\text{benefactive}(e, x) \\
\text{Appl}\_\text{ben} \\
\lambda x\lambda e.\text{eat}(e, x)
\]

(3) Marantz 1993, in the framework of Kratzer 1996. to appear

\[
\lambda x.\lambda e.\text{eating}(e) & \land \text{theme}(e, \text{food}) \land \text{benefactive}(e, \text{wife}) \\
\text{he} \\
\text{VoiceP} \\
\text{Voice'} \\
\lambda x.\lambda e.\text{agent}(e, x) \\
\text{Voice} \\
\lambda x.\lambda e.\text{eating}(e) & \land \text{theme}(e, \text{food}) \land \text{benefactive}(e, \text{wife}) \\
\text{VP} \\
\text{ApplP} \\
\lambda x.\lambda e.\text{benefactive}(e, x) \\
\text{Appl}\_\text{ben} \\
\lambda x\lambda e.\text{eat}(e, x)
\]

(2) Chaga

\[
\text{FOC-1SG-PRES-eat-APPL-FV} \quad 1\text{-wife} \quad 7\text{-food} \\
\text{He is eating food for his wife.}
\]

Ling 340 ~ Spring 2017 ~ C. Ussery
Low Appl Derivation

The Steps:

- **In the high applicative derivation, the verb combines with its object. NOT SO in the low applicative derivation. The verb combines with ApplP.**

- The direct object combines with the Appl head and saturates the $\lambda x$ slot.

- The applicative is merged in the specifier of ApplP and saturates the $\lambda y$ slot.

- The verb combines with ApplP and saturates the $\lambda f$ slot.

  - $f$ stands for function and, here, the function is necessarily of type $<e, <s, t>>$.

  - Only a verb of type $<e, <s, t>>$ can combine with ApplP.

- The Voice head combines with VP and introduces a spot for the subject external argument.

- The subject is merged in in the specifier of VoiceP and saturates the $\lambda x$ slot.
Low Recipient vs Low Source Applicatives

- English has a low **recipient** applicative.

- Korean has a low **source** applicative.
  - The applicative is the source of the direct object.

- The only difference is that the denotation encodes “from the possession of” instead of “to the possession of”.

(12) Low source applicative: Korean

```
Totuk-i Mary-hanthey punci-lul humchi-ess-ta.
thief-NOM Mary-DAT ring-ACC steal-PAST-PLAIN
‘The thief stole a ring from Mary.’ (Lit.: ‘The thief stole Mary a ring.’)
```
Hypothesized meaning: ‘The thief stole a ring and it was from Mary’s possession.’

(15) a. \( \text{Low-Appl}_{To} \) (Recipient applicative)

\[
\lambda x. \lambda y. \dot{f}_{e, \langle s, t \rangle} . \lambda e. f(e, x) \& \text{theme}(e, x) \& \text{to-the-possession}(x, y)
\]

b. \( \text{Low-Appl}_{From} \) (Source applicative)

\[
\lambda x. \lambda y. \dot{f}_{e, \langle s, t \rangle} . \lambda e. f(e, x) \& \text{theme}(e, x) \& \text{from-the-possession}(x, y)
\]
What about “normal” ditransitives?

- For Pylkkänen, these are the same as low applicatives.

- Pylkkänen argues against a causative/small clause analysis of ditransitives, based on...

  ...Entailment:
  - Resulting states are entailed in causatives. = (10)
  - Resulting states aren’t entailed in double objects. = (9)

  ...Depictives:
  - Small clauses can have depictive modification of the subject. = (11b)
  - Ditransitives can’t have depictive modification of the indirect object. = (11a)

Depictives describe a state that one of the arguments of a verb is in during the event described by the verb. We’ll come back to these.
Diagnostics and Predictions

Transitivity Restrictions: Only High Appl should be able to combine with an unergative. Low Appl requires a direct object.

Verb Semantics: Only High Appl should be able to combine with static verbs. Low Appl involves an (intended) transfer of possession.

Ling 340 ~ Spring 2017 ~ C. Ussery

(20) English
   a. *Unergative verb
      *I ran him.
   b. *Static verb
      *I held him the bag.

(21) Japanese
   a. *Unergative verb
      *Taro-o-ga Hanako-ni hasit-ta.
      Taro-NOM Hanako-DAT run-PAST
      ‘Taro ran for Hanako.’
   b. *Static verb
      *Taro-o-ga Hanako-ni kanojo-no kaban-o mot-ta.
      Taro-NOM Hanako-DAT she-GEN bag-ACC hold-PAST
      ‘Taro held Hanako her bag.’

(22) Korean
   a. *Unergative verb
      Mary-NOM John-DAT run-PAST-PLAIN
      ‘Mary ran to/from John.’
   b. *Static verb
      John-NOM Mary-DAT bag-ACC hold-PAST-PLAIN
      ‘John held Mary her bag.’

(23) Luganda
   a. √Unergative verb
      Mukasa ya-tambu-le-dde Katonga.
      Mukasa 3SG.PAST-walk-APPL-PAST Katonga
      ‘Mukasa walked for Katonga.’
   b. √Static verb
      Katonga ya-kwaant-i-dde Mukasa ensawo.
      Katonga 3SG.PAST-hold-APPL-PAST Mukasa bag
      ‘Katonga held the bag for Mukasa.’

(24) Venda
   a. √Unergative verb
      Nd-i-do-shum-el-a musadzi.
      1SG-FUT-work-APPL-FV lady
      ‘I will work for the lady.’
   b. √Static verb
      Nd-o-far-el-a Mukasa khali.
      1SG-PAST-hold-APPL-FV Mukasa pot
      ‘I held the pot for Mukasa.’

(25) Albanian
   a. √Unergative verb
      I jpgova.
      him.DAT CL ran.1SG
      ‘I ran for him.’
   b. √Static verb
      Agimi i mban D rites çanten time.
      Agim-NOM CL holds Drita.DAT bag.ACC my
      ‘Agim holds my bag for Drita.’
The descriptive generalization:
There is a typology of applicatives. In some applicative constructions, the extra argument is related to an event. In other applicative constructions, the extra argument is related to the direct object.

The analysis:
There is an applicative head (Appl) that mediates the relationship between the applicative argument and the rest of the construction.

In constructions in which the applicative has a relationship with the event denoted in the VP, Appl attaches higher than the verb.
- Appl combines with VP.
- The denotation for High Appl includes argument positions for an entity (the applicative) and an event. It’s like Voice.
- High Appl Languages: Venda, Chaga, Chichewa, Luganda, Albanian

In constructions in which the applicative has a relationship with the direct object, Appl attaches lower than the verb.
- Both the applicative and the direct object are merged inside of ApplP.
- Appl combines with the direct object and Appl’ combines with the applicative.
- The denotation for Low Appl includes argument positions for the direct object, the applicative, the verb, and an event.
- Low Appl Languages: English, Korean, Japanese

Predictions and diagnostics:
- High Applicatives are compatible with unergatives and statives verbs. Low Applicatives are not.
The Descriptive Observations:

- Subjects and direct objects can be modified by a depictive (26a)/(27), but indirect objects can’t (26b).

- Depictives are like adverbs in that “they attribute a property to the event described by the verb.” (p.23)

- The state described by the adjective holds during the event encoded in the verb.

Back to the stage level/individual level distinction:

- Since depictives describe a state that holds during an event, depictives sound odd with individual level adjectives. = (29)

The Proposal:

- Depictives have two parts - (1) the adjective, (2) a depictive head (Dep)
  - In some languages (e.g. Finnish) depictives are morphologically marked. = (32)
We traveled *tired*.

- describes a state that holds during an event
- Again, $\lambda f$ is a function. Here, it’s the adjective, which is of type $<e, <s, t>>$.
- $\lambda x$ is an entity
- $\lambda e$ is an event
- There is some function (an adjective) of type $<e, <s, t>>$ and some entity and some event such that:
- There is at least one state such that the adjective holds of $x$ in that state and that state holds during an event.
**Predicate Modification**: This is new for us. This operation combines two daughters of the same type and returns a value of that type. Like Event Identification, it’s a conjunction operation of sorts.

- For the Applicative analysis, the two nodes that are combined both have type \(<e, <s,t>>\). They combine and their mother has type \(<e, <s,t>>\).

- Pylkkänen proposes that DepP is of type \(<e, <s,t>>\).
- The nodes that DepP combines with are also of type \(<e, <s,t>>\).
  - Object depictives combine with the verb.
  - Subject depictives combine with Voice’.

![Diagram](image)

\(<e, <s,t>>\)
DepP contains the adjective and the depictive head.
- The adjective saturates $\lambda f$ of Dep.
  - DepP is type $\langle e, <s,t> \rangle$.
- The verb combines with DepP.
  - The verb is also $\langle e, <s,t> \rangle$.
- DepP and the verb combine via Predicate Modification.
  - $V'$ is type $\langle e, <s,t> \rangle$.
- The subject of the small clause VP is merged in the specifier and saturates the $\lambda x$ slot. We get:
  - There is some event such that the event is a seeing event and Peter is the theme of that event and there is at least one state which is a state of being tired and Peter is in that state and that state holds during the event.
  - WHEW!!!!!!
- The Voice head combines with VP and introduces a spot for the subject external argument.
- Event Identification applies as normal.
  - Voice is $\langle e, <s,t> \rangle$ and VP is $\langle s,t \rangle$.
- The subject is merged in the specifier of VoiceP and saturates the $\lambda x$ slot.
This one is structurally trickier.

The verb *see* combines with the direct object.

Voice combines with the VP as normal. Event Identification.

Here’s where things get slightly more complex.

Again, DepP contains the adjective and the depictive head. The adjective saturates $\lambda f$ of Dep.

DepP is type $<e, <s, t>>$.

DepP combines with Voice’, which is $<e, <s, t>>$.

Structurally, DepP is in an adjunct position - it’s daughter and sister to a bar-level. Here, there are two Voice’ levels.

Predicate Modification: Voice’ + DepP

There is some entity and some event such that the event is a seeing event and there is an entity that is the agent of that event and Peter is the theme of that event and there is at least one state which is a state of being tired and an entity is in that state and that state holds during the event.

Again...WHEW!!!!!

Unlike with the object depictives, we don’t know who is in the state of being tired at this point in the structure.

The subject is merged in the specifier of VoiceP and saturates the $\lambda x$ slot. Sue is the agent of the seeing and is in the state of being tired.
Depictives and Applicatives

- DepP combines with phrases that are \(<e, <s, t>>\). **Predictions:**
  - Depictives can’t modify low applicatives/indirect objects. \(^{(26b)}\)
    - DepP would have to attach to Appl’ and Appl’ is \(<e, <e, st> <s, t>>\). See (37)
  - Depictives can modify direct objects (even in ditransitive/ applicative constructions). \(^{(26a)}/(38)\)
    - DepP attaches to the verb. (See next page.)

(26) a. I gave Mary the meat raw.
    b. *I gave Mary the meat hungry.

(Baker 1997, (23c,d))

(37) Low applicative

(38) Depictive modification of the direct object in a low applicative construction

a. I bought John the VCR new.
Depictive Modification of Direct Object in an Applicative Construction

Remember the denotation for Low Appl

(15) a. \( Low-\text{Appl}_{\text{Rec}} \) (Recipient applicative)
\[
\lambda x. \lambda y. \lambda f_{\text{(e, x, y)}}. \lambda e. \ell(e, x) \& \text{theme}(e, x) \& \text{to-the-possession}(x, y)
\]
b. \( Low-\text{Appl}_{\text{Src}} \) (Source applicative)
\[
\lambda x. \lambda y. \lambda f_{\text{(e, x, y)}}. \lambda e. \ell(e, x) \& \text{theme}(e, x) \& \text{from-the-possession}(x, y)
\]

\( VP \) means: There is an event such that the event is of buying and the VCR is the theme of that event and there is at least one state which is a state of being new and the VCR is in that state and that state holds during the event and the VCR is to the possession of John.

Predicate Modification: \( V + \text{DepP} \)

- DepP attaches to the verb, just as in the Object Depictive Derivation.
- The verb and DepP combine via Predicate Modification.
- ApplP combines with \( V' \). In syntactic terms, ApplP occupies the specifier of the VP.
- Voice combines with VP and things proceed as expected.

Ling 340 - Spring 2017 - C. Ussery
Both Japanese and English are Low Appl languages and the Depictive can’t modify the applicative.

(39) Basic depictive distribution
   a. Object depictive
      Taroo-ga katuo-o nama-de tabe-ta.
      Taro-NOM bonito-ACC raw eat-PAST
      ‘Taro ate the bonito raw.’
   b. Subject depictive
      Taroo-ga ie-o hadaka-de nut-ta.
      Taro-NOM house-ACC naked paint-PAST
      ‘Taro painted the house naked.’

(40) Depictive cannot modify low applied argument
      Taro-NOM naked Hanako-DAT book-ACC read-PAST
      ‘Taro read Hanako a book while she was naked.’
      (False if Taro isn’t naked)
      Taro-NOM Hanako-DAT naked book-ACC read-PAST
      ‘Taro read Hanako a book while she was naked.’
      (False if Taro isn’t naked)

Naked can’t modify Hanako.
Luganda is a High Appl language and the Depictive can modify the applicative.
From Kratzer:
- The “meaning” of a verb includes its internal argument (if there is one) and an event.
- The semantic job of v is to add the external argument to the event encoded in the VP.

From Pylkkänen:
- Languages vary in how they add “extra” arguments to the structure.
- There is an applicative head (Appl) that mediates the relationship between the applicative argument and the rest of the construction.
- Low Appl attaches below the verb and mediates the relationship between the applicative and the direct object.
- High Appl attaches above the verb and mediates the relationship between the applicative and the event encoded in the VP.
- The distribution of depictives interacts with the typology of applicatives.
  - Depictives can modify high applicatives, but not low applicatives.
REFERENCES