Grammatik verbaler Argumente
Argumentkodierung 4: Ein neuer Ansatz

Gereon Müller

Institut für Linguistik
Universität Leipzig

SoSe 2009

www.uni-leipzig.de/~muellerg
Goal:
A new minimalist analysis of accusative vs. ergative patterns of argument encoding (via case marking or agreement) that meets the criteria postulated before (e.g. no construction-specific rules, no movement, erg=acc, nom=abs, etc.), and that is grounded in Murasugi’s (1992) approach. The proposal is based on the observation that indeterminacies may arise in the application of Merge and Agree (see Chomsky (2000, 2001)), given that they both obey an Earliness requirement (see Pesetsky (1989), Pesetsky & Torrego (2001)).

Basic claim:
A principled resolution of one such indeterminacy (on the vP cycle) in one or the other direction yields an accusative or ergative encoding pattern for arguments.
Syntactic structure is created incrementally, bottom-up, by the elementary operations Merge and Agree, and by Move (which is a special case of Merge: internal vs. external Merge; Chomsky (2005), and which will not play a major role in what follows).

(1) **Two types of features that drive operations:**
   
a. Structure-building features (edge features, subcategorization features) trigger Merge: \([\bullet F \bullet]\)  
   (Svenonius (1994), Collins (2003), Sternefeld (2003), Heck (2004))
   
b. Probe features trigger Agree: \([* F *]\)  
   (Sternefeld (2003))

(2) **Merge Condition:**
Structure-building features (\([\bullet F \bullet]\)) participate in Merge.

(3) **Agree Condition:**
Probes (\([* F *]\)) participate in Agree.
(4) The operation Merge:
\( \alpha \) can be merged with \( \beta \), yielding \( \{\alpha, \{\alpha, \beta\}\} \), if \( \alpha \) bears a structure-building feature \([\bullet F \bullet]\) and \( F \) is the label of \( \beta \).
(4) The operation Merge:
\(\alpha\) can be merged with \(\beta\), yielding \(\{\alpha,\{\alpha, \beta\}\}\), if \(\alpha\) bears a structure-building feature \([\bullet \mathbf{F}\bullet]\) and \(\mathbf{F}\) is the label of \(\beta\).

(5) The operation Agree:
\(\alpha\) agrees with \(\beta\) with respect to a feature bundle \(\Gamma\) iff (a), (b), and (c) hold:

a. \(\alpha\) bears a probe feature \([\ast \mathbf{F}\ast]\) in \(\Gamma\), \(\beta\) bears a matching goal feature \([\mathbf{F}]\) in \(\Gamma\).

b. \(\alpha\) m-commands \(\beta\).

c. There is no \(\delta\) such that (i) and (ii) hold:
   (i) \(\delta\) is closer to \(\alpha\) than \(\beta\).
   (ii) \(\delta\) bears a feature \([\mathbf{F}]\) that has not yet participated in Agree.
Background Assumptions 3

Note:

1. (5-b) permits an Agree relation between a head and its specifier, as seems natural (but see, e.g., Chomsky (2004)).

2. (5-c) presupposes a notion of closeness.

(6) **Closeness:**
\[ \delta \text{ is closer to } \alpha \text{ than } \beta \text{ if the path from } \delta \text{ to } \alpha \text{ is shorter than the path from } \beta \text{ to } \alpha. \]

(7) **Path** (Müller (1998, 130); also cf. Pesetsky (1982, 289), Collins (1994, 56)):
The path from X to Y is the set of categories Z such that (a) and (b) hold:
a. Z is reflexively dominated by the minimal XP that dominates both X and Y.
b. Z dominates X or Y.
The length of a path is determined by its cardinality.

Consequences:
(i) The specifier and the complement of a head qualify as equally close to the head.
(ii) The specifier of a head is closer to the head than a category that is further embedded in the complement of the head.
Further general assumptions (Chomsky (2000, 2001)):

1. **Clause structure:**
   Basic clause structure has CP, TP, vP, and VP.

2. **Numerations:**
   Lexical items that are to participate in derivations are selected from the lexicon pre-syntactically, and assembled in a numeration N (or lexical array).

3. **Workspace** (Frampton & Gutman (1999), Hornstein (2001)):
   The workspace of the derivation comprises items in the numeration and phrases that have been created independently.

4. **Merge of argument DPs:**
   DP$_{int}$ is merged in VP, DP$_{ext}$ is merged in vP, as a specifier.

5. **Argument encoding and functional heads:**
   T and v are involved in the structural encoding of primary arguments (i.e., DP$_{ext}$ and DP$_{int}$ arguments for which no inherent/lexical CASE is specified), by bearing features that act as probes and thus trigger Agree operations.
More specific assumptions about argument encoding:

1. There is one structural argument encoding feature: \textit{CASE}.

2. \textit{CASE} can have two values: ext(ernal) and int(ernal) (determined with respect to vP, the predicate domain).

3. \([\text{CASE:ext}] = \text{nominative/absolutive}, [\text{CASE:int}] = \text{accusative/ergative}\) (Murasugi (1992)).

4. \textit{CASE} features figure in Agree relations involving T/v and DP, as in (8).

(8) **The role of T and v in argument encoding:**

a. T bears a probe \([*\text{CASE:ext}*]\) that instantiates a matching \([\text{CASE:ext}]\) goal on DP.

b. v bears a probe \([*\text{CASE:int}*]\) that instantiates a matching \([\text{CASE:int}]\) goal on DP.
Case and Agreement

Observation:
Case-marking and agreement-marking both depend on an Agree relation between T/v and DP, and thus qualify as two sides of the same coin.

(9) Argument encoding by case or agreement:
   a. Argument encoding proceeds by case-marking if [CASE:α] is morphologically realized on DP.
   b. Argument encoding proceeds by agreement-marking if [*CASE:α*] is morphologically realized on T/v.

(Something extra will have to be said about cases where agreement is not case-based, but, e.g., argument type-based (external vs. internal argument).)
A conspicuous property:

ν (unlike T or V) plays a dual role: It triggers Merge of DP_{ext} (by a \([\bullet \text{D} \bullet]\) feature), and it also triggers an Agree operation (by its \([\ast \text{CASE: int}\ast]\) feature). This dual role has far-reaching consequences for the nature of argument encoding.

An indeterminacy:

Consider a simple transitive context, with two arguments DP_{int}, DP_{ext}. Suppose that the derivation has reached a stage Σ where ν has been merged with a VP containing DP_{int}, with DP_{ext} waiting to be merged with ν in the workspace of the derivation. At this point, an indeterminacy in rule application arises: The next operation could be either Agree(ν,DP_{int}) (see (i)) or Merge(DP_{ext},ν) (see (ii)). The Agree Condition demands the former operation, and the Merge Condition demands the latter.
Merge vs. Agree 2: The Dilemma

Stage $\Sigma$:

$$DP_{[c:\square]} \rightarrow (\text{ii) } v'_{[*c: \text{int*}, [\bullet D \bullet]}}$$

(i) $V$ $DP_{[c:\square]}$
Solution:
Conflicts of this type are real and must be resolved by giving one constraint (the Merge Condition or the Agree Condition) priority over the other in the case of conflict, i.e., by ranking the requirements.

Note:
This is an instance of optimization in syntax, with minimal violability of the lower-ranked requirement; see Prince & Smolensky (2004). However, the optimization involved here is extremely local (competing candidates are derivational steps), which avoids the complexity problems incurred by standard optimization procedures; see Heck & Müller (2000, 2006), Fischer (2004).
Solution:
Conflicts of this type are real and must be resolved by giving one constraint (the Merge Condition or the Agree Condition) priority over the other in the case of conflict, i.e., by ranking the requirements.

Note:
This is an instance of optimization in syntax, with minimal violability of the lower-ranked requirement; see Prince & Smolensky (2004). However, the optimization involved here is extremely local (competing candidates are derivational steps), which avoids the complexity problems incurred by standard optimization procedures; see Heck & Müller (2000, 2006), Fischer (2004).

(11)  Rankings:
   a. Accusative patterns: Agree Condition ≫ Merge Condition
   b. Ergative patterns: Merge Condition ≫ Agree Condition
(12) **Agree before Merge: accusative**

The diagram illustrates the syntactic structure of a sentence, with nodes labeled according to their syntactic categories: TP, T', T[∗c:ext*], vP, v', v[∗c:int*], [•D•], DP_{ext}, DP_{int}, VP, V, and DP. The diagram also includes arrows indicating the direction of Agree before Merge.
The Order of Elementary Operations 2

(13) Merge before Agree: ergative

\[ T' \]
\[ TP \]
\[ T[\ast c:ext\ast] \]
\[ T' \]
\[ vP \]
\[ \text{DP}_{\text{ext}} \]
\[ (i) \]
\[ v[\ast c:int\ast],[\bullet D\bullet] \]
\[ \text{VP} \]
\[ (ii) \]
\[ V \]
\[ \text{DP}_{\text{int}} \]
\[ (iii) \]
Remarks on the accusative pattern:

1 Mechanics:
Given Earliness(Agree) \(\gg\) Earliness(Merge), Agree\( (v, \text{DP}_{int})\) applies first (step (i)) at stage \(\Sigma\). Since \(v\) is marked \([\text{*CASE: int*}]\), this ensures a \([\text{CASE: int}]\) specification on \(\text{DP}_{int}\). \(\text{DP}_{ext}\) is merged in Spec\(v\) in the next step (step (ii)). The derivation continues, merging \(T\) and \(vP\), and then carrying out Agree\( (T, \text{DP}_{ext})\), which instantiates \([\text{CASE: ext}]\) on \(\text{DP}_{ext}\) (step (iii)).

2 Accusative, nominative:
The morphological realization of an internal encoding feature \([(*\text{CASE: int}(*))]\) with Agree\( (v, \text{DP}_{int})\) (by case or agreement) can be called accusative; the morphological realization of an external encoding feature \([(*\text{CASE: ext}(*))]\) with Agree\( (T, \text{DP}_{ext})\) can be called nominative.

3 Language types derived:
This accounts for argument encoding in transitive contexts in accusative languages like Icelandic and Navajo: The internal argument is marked by the internal \text{CASE}, the external argument is marked by the external \text{CASE}. 
Remarks on the ergative pattern:

1. Mechanics:
   Given Earliness(Merge) $\geq$ Earliness(Agree), Merge(DP_{ext},v) must apply first (step (i)) at stage $\Sigma$. DP_{ext} is now closer to v than DP_{int}, and given that Agree relations are subject to a minimality requirement and require only m-command by the probe, the next operation will have to be Agree(v,DP_{ext}), in a specifier/head configuration (step (ii)). This instantiates [CASE:int] on DP_{ext}. Subsequently, T is merged, and Agree(T,DP_{int}) is carried out (step (iii)), with [CASE:ext] for DP_{int}.

2. Ergative, absolutive:
   The morphological realization of an internal encoding feature [(*CASE:int(*))] with Agree(v,DP_{ext}) can be called ergative; the morphological realization of an external encoding feature [(*CASE:ext(*))] with Agree(T,DP_{int}) can be called absolutive.

3. Language types derived:
   This accounts for argument encoding in transitive contexts in ergative languages like Archi and Sierra Popoluca: The internal argument is marked by the external CASE, the external argument is marked by the internal CASE.
Nothing has been said about intransitive contexts so far.

**Problem:**
Unchecked probes lead to a crash of the derivation; hence, [\texttt{*CASE:}\alpha{}] must be absent on either T or v in the derivation if only one DP is present that has a feature [\texttt{CASE:□}]. But on which one?
Nothing has been said about intransitive contexts so far.

Problem:
Unchecked probes lead to a crash of the derivation; hence, [*CASE:α*] must be absent on either T or v in the derivation if only one DP is present that has a feature [CASE:□]. But on which one?

Claim:

(14) Feature Balance (holds of numerations):
For every feature [*F:α*], there must be a matching feature [F:□].

The underlying idea is that a derivation that fails to provide a matching goal feature specification for each probe feature specification that it employs is doomed from the start, and should be excluded as soon as possible, i.e., in the numeration.
Consequence:

[*CASE:ext*] on T or [*CASE:int*] on v must be absent if there is only one D with a CASE feature in the numeration.

Assumption:

There are (at least) two ways to respect Feature Balance in numerations underlying intransitive contexts:

1. Preservation of the unmarked [*CASE*] feature.
2. Preservation of the iconic [*CASE*] feature (that matches the argument type in markedness).
Solution 1: Preservation of the unmarked [*CASE*] feature

(15) Unmarked vs. marked CASE features:

a. \[(*)\text{CASE:ext(*)} \] (nominative/absolutive) $\rightarrow$ unmarked
   (typically default CASE in syntax, and segmentally less complex in
    morphology (often default or zero))

b. \[(*)\text{CASE:int(*)} \] (accusative/ergative) $\rightarrow$ marked
   (typically not default CASE in syntax, and segmentally more complex
   in morphology)

Consequence:
In intransitive contexts, [*CASE:ext*] on T has to be preserved, and [*CASE:int*]
cannot be instantiated on v. Therefore, the sole argument of an intransitive
predicate (DP\(_{ext}\) or D\(_{int}\)) is encoded by \[(*)\text{CASE:ext(*)} \] (nominative/absolutive),
after Agree(T,DP\(_{ext}\)) or Agree(T,DP\(_{int}\)), which captures the situation in the
language types discussed so far.
(16)

a. **Nominative/absolutive with DP_{ext}-V_{i}**

\[
\begin{array}{c}
TP \\
\downarrow \\
T' \\
\downarrow \\
T[*c:ext*] \\
\downarrow \\
DP_{ext} \\
\downarrow \\
(ii) \\
\end{array}
\]

\[
\begin{array}{c}
vP \\
\downarrow \\
v' \\
\downarrow \\
(i) \\
\end{array}
\]

b. **Nominative/absolutive with DP_{int}-V_{i}**

\[
\begin{array}{c}
TP \\
\downarrow \\
T' \\
\downarrow \\
T[*c:ext*] \\
\downarrow \\
v \\
\downarrow \\
\begin{array}{c}
V \\
\downarrow \\
VP \\
\downarrow \\
(iii) \\
\end{array}
\end{array}
\]

\[
\begin{array}{c}
vP \\
\downarrow \\
\begin{array}{c}
\begin{array}{c}
v \\
\downarrow \\
V \\
\downarrow \\
(ii) \\
\end{array}
\end{array}
\end{array}
\]
Solution 2: Preservation of the iconic [*CASE*] feature

(17) Unmarked vs. marked argument types:

   a. $\text{DP}_{\text{int}} \rightarrow \text{unmarked}$
   (merged within its predicate’s projection, not requiring a special ‘externalization’ operation (Williams (1981)) in argument structure)
   
   b. $\text{DP}_{\text{ext}} \rightarrow \text{marked}$
   (not merged within its predicate’s projection, requiring a special ‘externalization’ operation)

Consequence:
(i) A marked feature specification ([*CASE:int*] on $v$) must show up in the numeration in the presence of a $V$ taking an marked argument ($\text{DP}_{\text{ext}}$).
(ii) An unmarked feature specification ([*CASE:ext*] on $T$) occurs in the presence of a $V$ taking an unmarked argument ($\text{DP}_{\text{int}}$).
(18)

a. Ergative with $DP_{ext-Vi}$

b. Absolutive with $DP_{int-Vi}$
Observation:
This makes it possible to account for active ergative argument encoding patterns in languages like Basque, Georgian, Hindi (with case) and Guaraní (with agreement) without invoking the assumption that unergatives are hidden transitives in these languages (but not in others).

(19) **Active ergative case-marking in Basque:**

a. Jon-Ø  etorri  da  
   Jon-ABS come:PTCP.PRF  is:3.SG.INTR  
   ‘Jon came.’

b. Jon-ek  saltatu  du  
   Jon-ERG jump:PTCP.PRF  have:3.SG.TR  
   ‘Jon jumped.’

c. Jon-ek  ardo-a-Ø  ekarri  du  
   Jon-ERG wine-DET-ABS bring:PTCP.PRF  have:3.SG.TR  
   ‘Jon brought the wine.’ (Hualde & Ortiz de Urbina (2003, 364))
Active ergative agreement-marking in Guaraní (Tupí-Guaraní):

a. Še-manu’a
   1.SG.ABS-remember
   ‘I remember.’

b. A-ma.apo
   1.SG.ERG-work
   ‘I work.’

c. Ø-Ai-pete
   3.SG.ABS-1.SG.ERG-hit
   ‘I hit him.’

(Gregores & Suárez (1967))
Anti-Active Patterns

Note:
The present analysis does not per se exclude an ‘anti-active’ pattern, as in (21). Anti-active marking would arise in an accusative system that preserves the CASE feature specification matching the argument type in markedness in intransitive contexts (rather than the CASE feature specification that is unmarked); it differs from the accusative pattern in (1-a) in encoding $DP_{ext}$ of $V_i$ by accusative. This type of encoding pattern does not seem to occur (see, e.g., Sigurðsson (2004)).

(21) Anti-active marking

\[
\begin{array}{c}
DP_{ext-V_i} \\
DP_{ext-V_t}
\end{array} \quad \begin{array}{c}
DP_{int-V_i} \\
DP_{int-V_t}
\end{array}
\]

nom acc

Possible Solution: Such a system is dysfunctional (Bechert (1979), and Lecture 1).
Active Accusative Systems

Languages: Eastern Pomo (extinct; Hokan, California), Acehnese (Austronesian, Northern Sumatra).
Ref.: Bittner & Hale (1996b).

(22) Intransitive and transitive verbs in Eastern Pomo:
   a. Míip míip-al sáaka
      he.NOM him-ACC killed
      ‘He killed him.’
   b. Míip-al xáa baakúma
      him-ACC in the water fell
      ‘He fell in the water (accidentally).’
   c. Míip káluhuya
      he.NOM went home
      ‘He went home.’

Sketch of an analysis:
There is a third possibility in intransitive contexts (next to preservation of the unmarked, or of the iconic case feature): preservation of the internal case feature for internal arguments, and of the external case feature for external arguments. → Prediction for ergative systems?
Active Accusative Systems

Languages: Eastern Pomo (extinct; Hokan, California), Acehnese (Austronesian, Northern Sumatra).
Ref.: Bittner & Hale (1996b).

(22) Intransitive and transitive verbs in Eastern Pomo:

a. Míip míip-al sáaka
   he.NOM him-ACC killed
   ‘He killed him.’

b. Míip-al xáa baakúma
   him-ACC in the water fell
   ‘He fell in the water (accidentally).’

c. Míip káluhuya
   he.NOM went home
   ‘He went home.’

Sketch of an analysis:
There is a third possibility in intransitive contexts (next to preservation of the unmarked, or of the iconic case feature): preservation of the internal case feature for internal arguments, and of the external case feature for external arguments. → Prediction for ergative systems?

anti-active pattern
An Alternative

There is an alternative solution to the Agree/Merge indeterminacy problem on the vP level; one that does not involve constraint violability and constraint ranking (Lennart Bierkandt, p.c.).

Assumption:

- Subcategorization features that trigger Merge ([\(\bullet F \bullet\)]) and probe features that trigger Agree ([\[*F*\]]) are ordered on \(v\).
- Only the highest feature on the feature hierarchy is visible for the Agree Condition and the Merge Condition at any given stage of the derivation.
- A probe or subcategorization features disappears (or becomes inert) after having triggered an operation (Merge or Agree).

(23) Ergative/accusative parameter:

a. \([*\text{CASE}:\text{INT}^*]\) > \([\bullet D \bullet]\) on \(v\): accusative encoding pattern

b. \([\bullet D \bullet]\) > \([*\text{CASE}:\text{INT}^*]\) on \(v\): ergative encoding pattern
Independent Evidence

A simple theory of linking (argument structure $\leadsto$ argument realization):

(24) John$_1$ gave a book$_2$ to Mary$_3$

(25) Hierarchy of subcategorization features on V (follows directly from the Bierwisch/Heim/Kratzer/Wunderlich system in terms of $\lambda$ prefixation laid out in lecture 1):

Agent $>$ Theme $>$ Goal

[●D●] $>$ [●D●] $>$ [●P●]
Syntactic Ergativity 1

So far, a notion like “subject” has been irrelevant. However, there are operations that may have to refer to such a concept, like reflexivization, raising, control, relativization or topic-chaining (pivot-chaining; Dixon (1994)).

Accusative pattern:
In accusative systems, it is often the nominative DP that has subject properties. Typically, the nominative DP is also the highest (or single) argument DP. If the highest argument is a non-nominative DP, as with oblique dative-nominative orders in Icelandic, the oblique DP can have subject properties (see lecture 1).

Ergative pattern:
In ergative systems, there are two possibilities: Either the highest argument DP, or the argument DP that is marked with absolutive case, can exhibit subject properties: morphological ergativity (except for case marking, the syntax treats $DP_{ext/int}^-V_i$ and $DP_{ext}^-V_t$ on a par) vs. syntactic ergativity (as with case marking, the syntax treats $DP_{ext/int}^-V_i$ and $DP_{int}^-V_t$ on a par. as in Mother saw father and – returned, where – is he, not she). The former option is chosen in Archi, Basque, Warlpiri; the latter in Dyirbal (at least as a tendency). Optionality is possible as well (Chukchi). Finally, in a single language, some operations may select the highest argument as the subject, and other operations may select the absolutive argument (Inuit). See Comrie (1989), Bobaljik (1993), Dixon (1994), Bittner & Hale (1996a,b).
Two possible analyses:
(i) In syntactically ergative systems, DP_{int} moves to SpecT (which it Agrees) and becomes the highest argument (cf. Bittner & Hale (1996b)).
(ii) Subject-oriented operations affect prominent arguments. There are two ways for an argument DP to become prominent:

(26) **Prominence:**
An argument DP counts as prominent if it
a. occupies the highest argument position in the clause;
b. undergoes Agree with the highest functional head.

*Consequence*:
The two notions of prominence typically converge on a single argument in accusative systems; the situation is different in ergative systems.
Further Issues: Split Ergativity, Three-Way Systems

Cases of split ergativity
- tense-/aspect-based split ergativity, as in Burushaski, Hindi (see Mahajan (1990))
- clause-type based split ergativity, as in Sierra Popoluca (see Elson (1960)).
- person-based split ergativity, as in Dyirbal (see Dixon (1994))

Co-occurrences of ergative and accusative
(see Dixon (1994), Woolford (1997), Kiparsky (1999), and lectures 1 and 3)
Tense/Aspect-Based Splits

Recall that there are two versions of the present analysis:

- **constraint ranking** (Merge Condition $\gg$ Agree Condition, or vice versa)
- **feature hierarchy** ([●D●] > [*CASE:INT*], or vice versa)

Here the feature hierarchy version might prove superior:

**Sketch of an analysis:**

1. The Hindi lexicon has two v’s:
   - $v_1$ with the feature hierarchy [●D●] > [*CASE:INT*]
   - $v_2$ with the feature hierarchy [*CASE:INT*] > [●D●].
2. $T[+\text{PERF}]$ selects $v_1$, $T[-\text{PERF}]$ selects $v_2$.
3. Morphological realization need not be identical.

Note:
A similar analysis can be developed for **clause-type** based splits, as in Sierra Popocluca.
Three-Way Systems 1

(27) **Antekerrepenhe** (Arandic; Central Australia):

a. Arengke-le aye-nhe ke-ke
   dog-ERG me-ACC bite-PST
   ‘The dog bit me.’

b. Apwerte-le athe arengke-nhe we-ke
   stones-INS I-ERG dog-ACC pelt-PST
   ‘I pelted the dog with stones.’

c. Arengke nterre-ke
   dog-NOM run-PST
   ‘The dog ran.’

(28) **Nez Perce** (Penutian; Oregon):

a. Wewúkiye-ne pée-’wi-ye háama-nm
   elk-ACC 3\textsubscript{i}.3\textsubscript{j}-shoot-PRF man-ERG\textsubscript{i}
   ‘The man shot an elk.’

b. Hi-páayn-a háama
   3\textsubscript{j}-arrive-PRF man-NOM
   ‘The man arrived.’

Analyses:

- morphological realization; or
- \( v \) has two case features (or there are two \( v \)’s)
Three-Way Systems 2: Morphological Account

Sketch of an analysis:

- Three-way systems are basically **accusative systems** (an absolutive of ergative encoding patterns is typically morphologically unmarked on a DP; a nominative of accusative encoding patterns can be marked on a DP).
- There are nominative **allomorphs**, e.g., *le* vs. *Ø* in Antekerrephe.
- Distributed Morphology (Halle & Marantz (1993)): Inflectional morphology is post-syntactic: It realizes morpho-syntactic feature bundles that are present in $X^0$ positions in the syntax. Between syntax and morphology, morpho-syntactic features can be deleted: **impoverishment**.

\[(29)\] An impoverishment rule for N Antekerrephe:

\[\text{[case:ext]} \rightarrow \emptyset/\_V\]  
(note: context = no internal argument present)

Potential conceptual problem:
Functional motivation for ergative patterns in syntax is mimicked in morphology.
Alternative:

- $v$ can exceptionally be equipped with two [*case:int*] features.
- In transitive contexts, there are thus three potential case features ([*case:ext*] on $T$, [*case:int*], [*case:int*] on $v$).
- One of these features has to disappear: constraints on numerations.
- Assumption: [*case:int*] on $v$ can never go away if there are both $\text{DP}_{\text{ext}}$ and $\text{DP}_{\text{int}}$ in the numeration. (See Marantz (1991), Bittner & Hale (1996b), Wunderlich (1997).)
- Consequently, [*case:ext*] is not instantiated on $T$ in transitive contexts.
- In accusative as well as ergative systems, $\text{DP}_{\text{ext}}$ and $\text{DP}_{\text{int}}$ both get [case:int] in transitive contexts.
- In intransitive contexts, only the unmarked case feature is instantiated: [*case:ext*] on $T$. 
Person-Based Splits

Person-based split ergativity in Dyirbal:
In Dyirbal, $\text{DP}_{\text{ext}}$ of $V_t$ is marked ergative if it is a 3rd person pronoun or an item to the right of it on the person/animacy scale in (30). $\text{DP}_{\text{int}}$ of $V_t$ is marked accusative if it is a 1st or 2nd person pronoun. All other types of argument DP are not encoded by an overt marker (see Dixon (1994)).

(30) Person/animacy scale (Silverstein (1976), Aissen (1999)):
1st person pronoun $>$ 2nd person pronoun $>$ 3rd person pronoun $>$ proper name $>$ common noun, human $>$ common noun, animate $>$ common noun, inanimate

Strategies for analysis: A person-based split may in principle be either a syntactic or a morphological phenomenon. Such splits often seem to be functionally motivated in the sense that only unexpected, atypical configurations are marked.

(a) Morphology: There is a zero allomorph that results from impoverishment (an operation from Distributed Morphology that deletes morphosyntactic features post-syntactically).
(Impoverishment rules might be motivated by functional considerations: Impoverishment can be assumed to be brought about by a system of violable and ranked constraints incorporating the Silverstein hierarchy (see Aissen (1999, 2003), where these constraints are automatically derived on the basis of simple hierarchies, via harmonic alignment and constraint conjunction).
(b) Syntax: Instantiation of argument encoding features in the numeration is restricted by the argument type (its place on the person/animacy scale).


