Masaryk University
Faculty of Arts
Department of Czech Language

Lucie Janků

THE NANOSYNTAX OF CZECH NOMINAL DECLENSION

Doctoral Thesis

Supervisor: Pavel Caha

Brno 2022

I hereby declare that the thesis titled The Nanosyntactic Account of Czech Nouns that I have submitted for assessment is entirely my original work, and that no part of it has been taken from the work of others unless explicitly cited and acknowledged within the text of my thesis.

.....
Author

Acknowledgments

I would like to thank my supervisor, Pavel Caha, for his endless patience, support, and all the inspiration not only throughout the time of writing the thesis but throughout my whole Ph.D. studies. Besides always coming up with new and interesting insights, he is also just a nice person to work with.

Big thanks go to Michal Starke, who taught me a lot about organizing my thoughts and creating stories through presentations. His spellout tool gramma helped me check many of the derivations presented here, making my life much easier and my sleep much calmer. Unexpectedly, Michal also showed me that speaking programming languages is almost as fun as speaking natural languages.

I was lucky to have Michal and Pavel organizing the linguistic life in Brno, giving me a chance to spend my Ph.D. years in a lively research community that taught me much in class and outside and made the experience a lot of fun too.

My gratitude extends to Petr Karlík, who was the first one to think I might be material for a linguist. I enjoyed our numerous discussions and appreciated all his encouragement. Thanks to Zdeňka Hladká and other members of the Department of Czech Language, too, for all their support and for creating an environment I was happy to be a part of.

Thanks to MarkétaZiková for patiently answering my phonology questions and helping me get into shape when the wild waves of research kept pushing me in that domain.

Thanks to all the people attending our research classes, whether here in Brno or online, connecting from all over the world, and sharing their wisdom and ideas. Among many others, ordered alphabetically, Tommaso Balsemin, Fenna Bergsma, Hagen Blix, Karen De Clercq, Lucie Taraldsen Medová, Trang Phan, Francesco Pinzin, Tarald Taraldsen, Utku Turk, Bartosz Wiland, and Guido Vanden Wyngaerd. They all influenced the research presented here in one way or another, whether through giving me feedback on my presentations or sharing their own work, which I would then get inspired by.

I am grateful to Marcel Den Dikken for his help in my linguistic beginnings when I came to Budapest for a semester in the hope of learning "some basics". His classes helped me get up to speed, his encouragement allowed me to ask ever more questions.

Last but not least, thanks to Maria Cortiula and Veronika Kloudová, my Ph.D. colleagues, who helped me have a life "outside of the studies", although we always ended up talking about linguistics anyway. A special thanks to Maria for being my roommate throughout the summers at the EGG.

Contents

Abstract

This dissertation proposes an account for suffix allomorphy in the Czech nominal domain in the singular. Czech, among other Slavic languages, is notoriously known for its complex morphological system of nouns. Any attempt for an account has to answer at least two questions. First, what mechanism ensures the correct pairing of the root with its suffix. And second, what the relationship is between paradigm cells with the same form.

The framework of Nanosyntax (Starke 2002) offers tools to handle both questions. Its phrasal spellout and the spellout algorithm suggest that the difference between paradigms is that different roots spell out different subsets of features. Depending on what subset they spell out, they will then be complemented by different suffixes. Importantly, only the exact packaging and the phonological exponents are tied to concrete languages. The features, their ordering, and the spellout algorithm are language-independent and, therefore, well-aligned with the notion of universal grammar. The extensive syncretisms are handled using Caha (2009)'s Case Hierarchy. It suggests a universal containment between cases, leading to adjacent forms being able to share form.

The second part of the account stands on Element Theory (Kaye et al. 1985). It allows us to make sense of the fact that many paradigms in Czech have an apparent relationship with each other despite taking different sets of suffixes. The relationship seems to boil down to the palatalized/non-palatalized distinction. We argue that these paradigms are, in fact, underlyingly the same and that it is phonology concealing their identity. The Element Theory decomposition significantly simplifies the labyrinth of affixes and allows us to elucidate their relationships.

The way the presentation goes is the following. We first propose a syntactic account for the non-palatalized and palatalized paradigms separately. Then, we show what role phonology plays and how we can unite both groups within the same account. That constitutes the largest part of the work. Besides the correct pairing, the work also explains why some paradigms start the same and then split, or vice versa, start different and then unify.

Once the big picture is in place, we tackle additional details, including how two paradigms can be the same in all cases but one in the middle of the functional sequence. We explore the details of the phonological computation, putting together phenomena like the -e-epenthesis and the floating -t. We look at a curious issue of some inanimate paradigms taking what is thought to be a specifically animate syncretism pattern. We also shine a light on a seeming ABA in one of the feminine paradigms. Based on diachronic and dialectal data, we propose that it is homophony, and the paradigm does not have the ABA underlyingly. All this is made possible by the newest developments within the framework, using tools like lexical entries with a complex left branch, private lexical entries, and pointers.

The case suffixes in Czech nouns and their pairing with the correct roots across all the six cases are explained within a single, consistent account without proposing any new or language-specific tools. As such, the account invites to be tested on and extended to data from other, not only Slavic languages.

1 Czech goes Nano

The Czech nominal domain with its three genders, six cases (putting vocative aside), and the animate vs. inanimate distinction seems rather complex. In addition to that, the nouns are split into numerous paradigms with no obvious morphosyntactic correlates. Take, e.g., the nouns *jazyk* 'tongue' and *chleb-a* 'bread'. They are both masculine inanimates, and yet, one takes no suffix in the nominative, while the other takes *-a*:[1]

(1) SURFACE FORMS OF 'TONGUE' AND 'BREAD', NOM

	'tongue'	'bread'
NOM	jazyk	chleb-a

The standard approach to accounting for different paradigms (or "declension classes") is to assign arbitrary class indexes. These indexes are then responsible for the correct root selecting the correct set of suffixes (see, e.g., Halle 1997, Halle and Vaux 1998 or Lampitelli 2014 for Latin).

If we were to follow the same direction, we would suggest arbitrary class indexes I and II for the two roots above, as in (2). In (3), we see how the right suffixes would have to be specified for the correct class as well:

(2) ROOT CLASS FEATURES

jazyk	⇔	[I]
chleb	⇔	[II]

(3) SUFFIX CLASS FEATURES

Ø	⇔	NOM / _ [I]
a	⇔	NOM / _ [II]

The contextual rules in (3) say that Ø gets inserted in the context of the nominative of class I, whereas *-a* gets inserted in the nominative of class II. Such specification not only makes sure that the correct root gets paired with the correct class but at the same time rules out the incorrect pairings, like the nominative **jazyk-a*.

One of the drawbacks of such analysis is that it introduces language-specific features (class I, II...), which goes against the core concept of universal grammar.

Czech nouns also show extensive syncretisms, which means that most of the paradigm cells are the same and could be reduced for a non-redundant account. The syncretisms in Czech are of two types. On one hand, those within a single paradigm (which we call *vertical*, based on their alignment in the tables):

(4) SURFACE FORMS OF 'BREAD', NOM–GEN

	'bread'
NOM	chleb-a
ACC	chleb-a
GEN	chleb-a

And on the other hand, the cross-paradigm (i.e., *horizontal*) syncretisms:

[1] All the nouns used throughout the work are in the singular, unless directly stated otherwise.

(5) SURFACE FORMS OF 'TONGUE' AND 'BREAD', NOM, GEN

	'tongue'	'bread'
NOM	jazyk	chleb-a
GEN	jazyk-a	chleb-a

Consider what happens in the standard theory when the roots that are assigned to two distinct classes in the nominative take the same suffix in, e.g., the genitive, as in (5). It is possible to suggest a rule that says that in the context of the genitive, both class I and class II take *-a*:

(6) CLASS I AND II TAKE -A IN THE GEN

a ⇔ GEN / _ [I, II]

Such rule, however, fails to capture any patterns having to do with which classes unify in the genitive by taking *-a* and which do not.

Müller (2004) takes care of similar cross-class syncretisms in Russian by proposing an analysis along the standard lines but updating it through introducing special equipollent class features +/-α and +/-β (see also Alexiadou and Müller 2008). These features are both arbitrary and language-specific.[2]

Russian has four recognized classes. According to standard descriptions (e.g., Grepl et al. 2012), there are fifteen different declension classes of Czech singular nouns.[3] In this work, we add three more beyond those most often discussed. Each of the eighteen paradigms we tackle has six cases, yielding one hundred and eight surface cells. To account for the wide range of paradigms in Czech, we would need a good number of arbitrary class features to assign them all correctly, and even if we managed, we would still be left with an account that cannot be universally extended to other languages.

Instead, we build on Caha's (2009) Nanosyntactic account of the vertical syncretisms in Czech and extend it to the domain of horizontal syncretisms to show that even those can be implemented without losing the elegance of Caha's original approach.[4] A single lexical entry is used for each suffix, and, in the end, only five lexical entries are needed to derive fifteen of the eighteen used paradigms.[5]

The goal of this work is twofold: to propose an account for declension classes without resorting to language-specific features and, at the same time, to insightfully capture the cross-class syncretisms. We use three tools to achieve this goal: the size of the lexical entries, their shape, and Element Theory to disentangle the phonology.

[2]See Caha (2021c) for a Nanosyntactic discussion of the Russian declension and how we can capture it while still keeping to a universal set of features.

[3]We do not count the paradigms that only differ in their vocative forms since we do not treat the vocative here. We also leave aside variants (like the locative *jazyk-u/jazyc-e* 'tongue'), sticking to just the *-u* suffix for the moment. We briefly come back to the variants later. We also stay away from nouns with adjectival declension, like *hajný* 'game warden', *grizzly* or *muftí* 'Mufti'. Their declension system is completely different, and as such, they constitute a distinct piece of the puzzle.

[4]For a Nanosyntactic work dealing with horizontal syncretisms in Russian and Ossetic, see Caha (2019a).

[5]Three paradigms, *žen-a* 'woman', *růž-e* 'rose' and *koleg-a* 'colleague', are placed in the Loose ends section 6 because more work will need to be done on those.

We will see how surface complexity does not have to entail underlying complexity. Embedding the Czech nominal declension in the Nanosyntactic framework, the underlying system turns out to be much simpler than we might have previously thought.

Before getting to the details, we look at three organizational principles in the Czech nominal system, which will allow us to introduce the basics of Nanosyntax:

1. Case suffixes can be syncretic within a paradigm. Why does grammar make a difference where morphology does not?
2. Two roots with seemingly the same features do not necessarily have the same set of suffixes. Why does morphology make a difference where grammar does not?
3. Despite the fact that even roots with the same features can take a different set of suffixes, the set does not have to differ entirely. Some of the suffixes can be shared across paradigms. How does the sharing happen?

In the following subsection, we start by looking at the second principle, which is ultimately about the issue of why we have the horizontal division within the nominal domain.

1.1 Allomorphy as a root size

As we have already seen in the previous section, the nouns *jazyk* 'tongue' and *chleb-a* 'bread' are both inanimate masculines. They are repeated below, both in the same case and number:

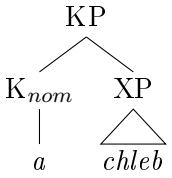
(7) SURFACE FORMS OF 'TONGUE' AND 'BREAD', NOM

	'tongue'	'bread'
NOM	jazyk	chleb-a

Why is it that one of them needs a suffix while the other does not?

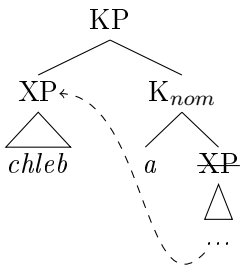
Under anyone's conception, *chleb-a* is going to have a node spelled out by the root (call it XP) and a node spelled out by the suffix (call it K_{nom}). This is like the tree in (8):

(8) SPELLOUT OF 'BREAD', NOM



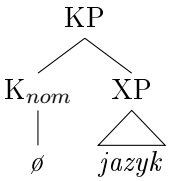
For the moment, let us simplify the picture and not go into the issue of morpheme ordering yet. The only thing relevant here is that *-a* spells out case, and *chleb* spells out XP. The ordering will be later derived through movement, resulting in the structure below:

(9) AFFIX ORDERING THROUGH MOVEMENT

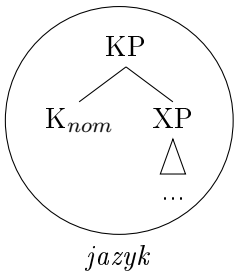


Against this background, there are two different ways how we can analyze *jazyk* – either with a zero allomorph, as in (10), or through phrasal spellout, as in (11):

(10) TERMINAL SPELLOUT OF 'TONGUE', NOM



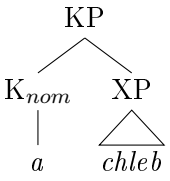
(11) PHRASAL SPELLOUT OF 'TONGUE', NOM



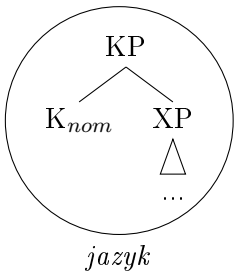
It turns out that if one uses (11), the account looks simpler. Not only is there no zero suffix, but there is also no need for context sensitivity the way we described in the first section.

Since phrases can be of various sizes, one by-product of phrasal spell out is that it allows for different root sizes. We can see that below, in a simplified version of syntactic derivations for both roots from (7):[6]

(12) SPELLOUT OF 'BREAD', NOM, SMALLER ROOT



(13) SPELLOUT OF 'TONGUE', NOM, BIGGER ROOT



[6] Phrasal spell out is at the core of Nanosyntax and both its merits and employment have been shown many times before (see for Starke 2009 and further, e.g. Caha 2009, Taraldsen 2010, Pantcheva 2011, De Clercq 2013, Starke 2018, Caha et al. 2019a). Also see Caha et al. (2019b) for the difference between a syntactic and a morphological root.

In (12)–(13), the root *chleb* 'bread' is smaller than the root *jazyk* 'tongue'. *Chleb* only reaches *under* the nominative feature, and hence the functional sequence needs to be finished by another morpheme. The root *jazyk* does not need any other morpheme, because it spells out the whole sequence.

Under this view, the horizontal division within the nominal domain arises from roots being stored in various sizes in the lexicon. Based on how far up they reach, they will then be complemented by different suffixes, or – if none are necessary – no suffixes at all.[7] There is no need to postulate language-specific classes, like class I, class II, because it is the root size – the packaging of universally shared features – that takes care of the correct root-suffix pairing. The question is: can we use this type of an account to give a coherent characterization of the entire declension?

Let us look at the second piece of the puzzle, which is how such an account takes care of the fact that sometimes a suffix is shared across different cases within a single paradigm.

1.2 Syncretism within a paradigm

The accusative form of *jazyk* 'tongue' is syncretic with its nominative form:

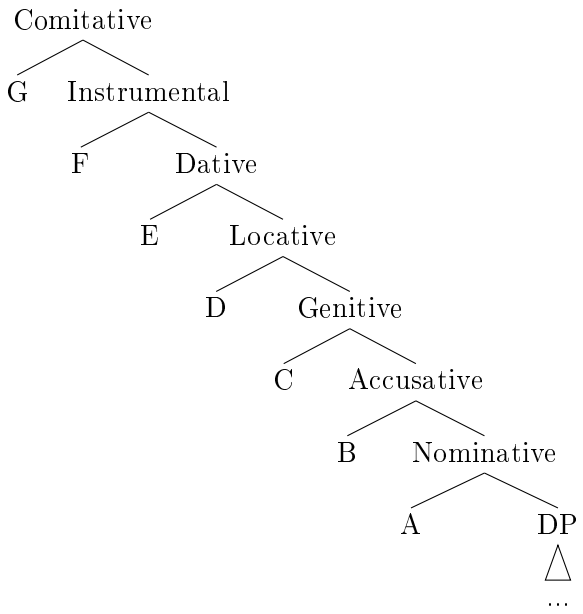
(14) SURFACE FORMS OF 'TONGUE', NOM–ACC

	'tongue'
NOM	jazyk
ACC	jazyk

This is an instance of what we called vertical syncretism before. Caha (2009) took care of these, arguing for containment between cases, as in (15):

[7] Note that nouns that do not take any suffixes throughout the whole declension, like *whisky*, are easily incorporated by suggesting that their roots spell out the whole functional sequence up to the instrumental feature.

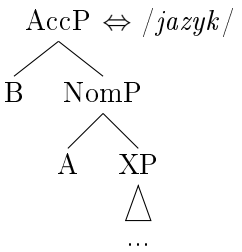
(15) THE CASE HIERARCHY, Caha (2009)



The above came to be known as the Case Hierarchy. Every higher case is built on top of the lower cases, so, e.g., the genitive is the feature C plus all the features contained in the Accusative node.

In the previous section, we accounted for the lack of suffixes in *jazyk* 'tongue' through phrasal spellout, suggesting that the root can spell out the whole functional sequence. The accusative looks the same, so the same logic should apply. According to the case hierarchy above, the accusative is the feature B plus the nominative node. Building on that, we would expect a lexical entry for the accusative of *jazyk* to look like this:

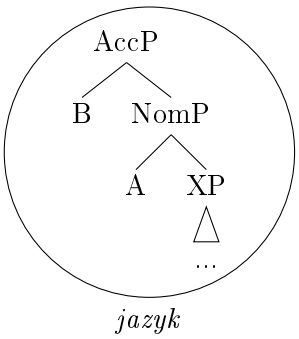
(16) LEXICAL ENTRY OF 'TONGUE'



The root *jazyk* can now spell out the whole sequence up to the accusative phrase and hence still does not need any other suffix.

In the syntactic derivation, the accusative node will be a perfect match for the stored tree – compare the derivation below with the lexical entry in (16):

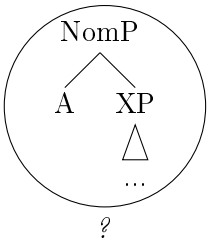
(17) SPELLOUT OF 'TONGUE', ACC



Note that throughout the work, we use bidirectional arrows between a well-formed syntactic structure and a phonological form to mark the lexical entries, as in (16), and a circle to mark a spelled out structure, as in (17).

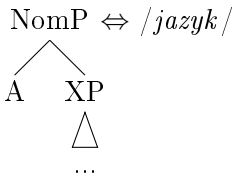
With the lexical entry for *jazyk* 'tongue' containing the accusative node now, something more needs to be said about the derivation of the nominative:

(18) SPELLOUT OF 'TONGUE', NOM

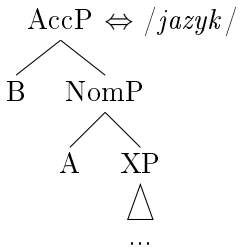


The nominative node is not a perfect match for the lexical entry in (16) since it misses the accusative layer. One way to deal with this would be to have lexical entries for both of the cases stored separately:

(19) LEXICAL ENTRY OF 'TONGUE', NOM SUBSET



(20) LEXICAL ENTRY OF 'TONGUE', FULL



That would, however, fail to say anything about the relationship between the two forms – the relationship that, as Caha (2009) convincingly argues, is not accidental.

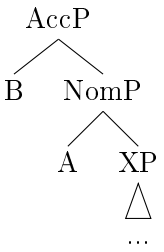
To avoid syncretism randomness of (19)–(20), Nanosyntax uses a matching condition known as the Superset Principle:

(21) THE SUPERSET PRINCIPLE, Starke (2009)

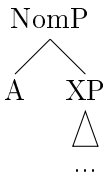
A lexically stored tree matches a syntactic node iff the lexically stored tree contains the syntactic node.

It stands on the idea that if AccP (in (20)) is stored in the lexicon and NomP is a proper subset of AccP, then NomP is also already stored in the lexicon. In practice, the Superset Principle means that the lexical entry in (20) can spell out all the constituents it contains:

(22) FULL ENTRY



(23) NOMP SUBSET

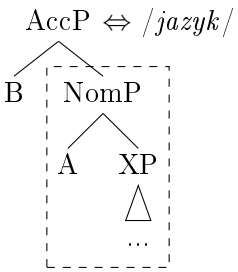


(24) XP SUBSET

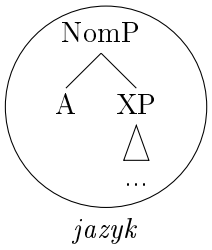
A side-effect of the Superset Principle that can be seen in (22)–(24) is that layers can be only shed from the top, and all the subsets will always share the bottom. This is known as the *foot effect*.

Due to the Superset Principle, we can discard the lexical entry in (19) and keep just the one in (20). The syntactic derivation of the nominative will be matched against the subset of the lexical entry (25) and will yield the correct spell out, *jazyk* (26):

(25) LEXICAL ENTRY OF 'TONGUE', NOMP SUBSET MARKED



(26) SPELLOUT OF 'TONGUE', NOM



As such, the Superset Principle links syncretic forms by storing them within a single lexical entry, relieving the workload of the lexicon.

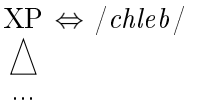
The same will be true for the accusative and genitive of *chleb-a* 'bread', both of which are syncretic with the nominative:

(27) SURFACE FORMS OF 'BREAD', NOM–GEN

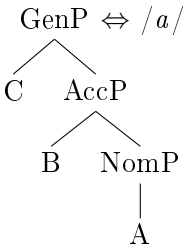
	'bread'
NOM	chleb-a
ACC	chleb-a
GEN	chleb-a

The only difference is that before, it was the root taking advantage of the Superset Principle, and now it is the suffix. As we suggested before, the lexical entry for the root *chleb* only spells out XP (entry repeated in (28)). The suffix *-a* starts right above it, at NomP, and extends to the genitive layer (29):

(28) LEXICAL ENTRY OF 'BREAD'



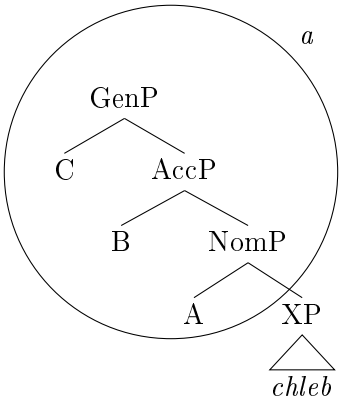
(29) LEXICAL ENTRY OF -A



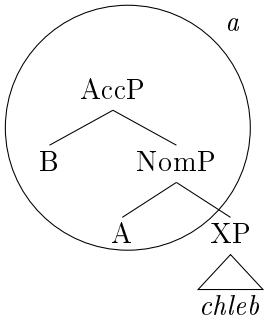
That way, the root *chleb* 'bread' gets automatically finished by *-a*.

For comparison, (30) is a result of the genitive derivation and (31) of the accusative derivation of *chleb-a*:

(30) SPELLOUT OF 'BREAD', GEN



(31) SPELLOUT OF 'BREAD', ACC



Since the accusative node is contained in the genitive node, *-a* can also spell out the accusative. The same will hold for the nominative.[8]

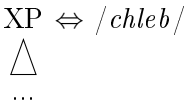
Everything we said up to now works for each root as a separate universe. As we have already seen, though, sometimes paradigms share suffixes. In the next section, we look at that and answer the second part of the issue stated at the beginning, which is how cross-class syncretism can be captured without resorting to language-specific tools.

[8]The Superset Principle is a powerful tool, and it is at the core of Nanosyntax, just as the phrasal spellout is. For literature on its employment, see, e.g., Caha (2009), Starke (2009), Baunaz and Lander (2018). For comparison of the Superset Principle used in Nanosyntax and the Subset Principle of Distributed Morphology, see Caha (2018).

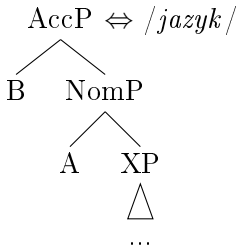
1.3 Syncretism across paradigms

To remind ourselves of where we left off, here are the simplified lexical entries suggested so far:

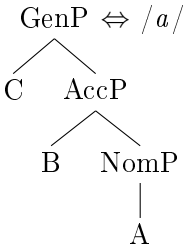
(32) LEXICAL ENTRY OF 'BREAD'



(33) LEXICAL ENTRY OF 'TONGUE'



(34) LEXICAL ENTRY OF -A



The root *chleb* 'bread' only spells out XP and therefore gets picked up by *-a* in the nominative through to the genitive to finish the functional sequence. The root *jazyk* 'tongue' grows all the way up to the accusative phrase.

Now note that the genitive of *jazyk* has the same suffix as *chleb-a*:

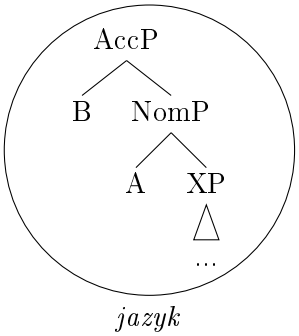
(35) SURFACE FORMS OF 'TONGUE' AND 'BREAD', NOM–GEN

	'tongue'	'bread'
NOM	jazyk	chleb-a
ACC	jazyk	chleb-a
GEN	jazyk-a	chleb-a

The above is an instance of horizontal syncretism, which is very common in Czech. It cannot be easily marked as a coincidence. We will see later that the genitive *-a* stubbornly pops out in other paradigms as well, no matter what the nominative and the accusative forms are.

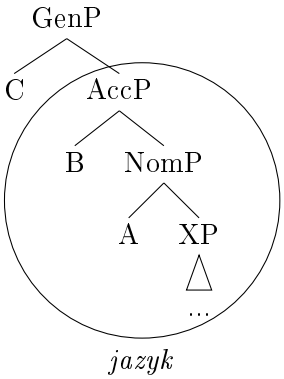
In the derivation, the root *jazyk* grows to AccP, which is as far as it can reach:

(36) SPELLOUT OF 'TONGUE', ACC



Then the syntax adds GenP:

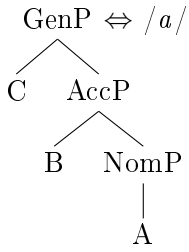
(37) SPELLOUT OF 'TONGUE', GENP MERGED



At this moment, the root cannot spell out GenP because it is not specified for it in the lexicon. The suffix *-a* cannot spell the GenP node either because the GenP node on its own is not a proper subset of the lexical entry for *-a* – remember the foot effect – since it lacks the accusative (and nominative) node (compare below):

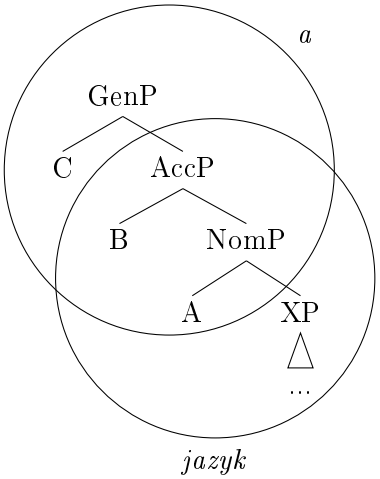
(38) GENP

(39) LEXICAL ENTRY OF -A



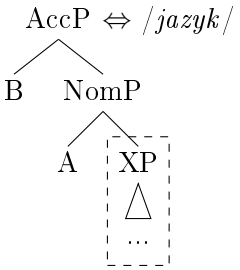
The issue lies in the fact that the two lexical entries overlap:

(40) SPELLOUT OF 'TONGUE', GEN, OVERLAPPING LEXICAL ENTRIES



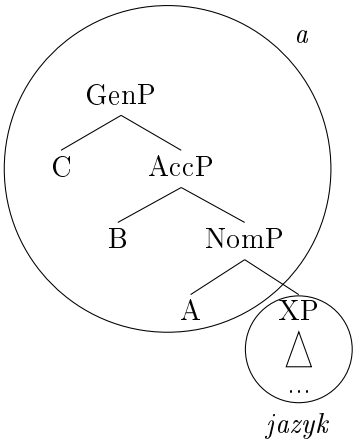
In our system, when two lexical entries overlap, the higher can never shrink enough to solve the problem. The lower one can, though. *Jazyk* can shed the upper layers through the Superset Principle and spell out only XP. The tree in (41) shows a (full) lexical entry of the root *jazyk*. The dashed line indicates a subset spelling out only XP:

(41) LEXICAL ENTRY OF 'TONGUE', XP SUBSET MARKED



This subset can then pick up -*a*, which starts right above XP:

(42) SPELLOUT OF 'TONGUE', GEN



Root shrinking to find someone to finish a functional sequence is called *backtracking* – the affix backtracks to one of the previous derivational stages. It needs to be emphasized that it is not a new tool – it is merely a by-product of the Superset Principle.[9]

As a consequence, our lexicon contains only one lexical entry for all four forms with *-a* (the nominative, accusative and genitive of *chleb-a* 'bread' and the genitive of *jazyk* 'tongue'). More generally, we explain the mechanism behind sharing suffixes – both within and across paradigms while keeping the theory free of language-specific rules. Czech will work just like Russian. Only the concrete packaging of specific lexical entries will distinguish the two.

Phrasal spell-out and the Superset Principle allow us to capture the root-suffix selection and both the vertical and horizontal syncretisms. In the next section, we start by looking at the whole complex system of Czech nouns and see how the tools work in practice. After introducing the Czech nominal system, three main sections follow. Section 3 focuses on the paradigms we can capture purely through root size. Section 4 shows that combining root size with Element Theory can reduce the seemingly large number of paradigms by disentangling the phonology. Finally, section 5 tackles more complex cases using root shape.

[9] For more literature on the use of backtracking, see, e.g., Starke (2018), Caha (2019a), Vanden Wyngaerd et al. (2020). For an alternative proposal to backtracking, see Blix (2021). We come back to the algorithmic implementation of backtracking at a later point.

2 Czech nominal declension and contours of the analysis

Let us now introduce the Czech noun paradigms and see how phonology plays a role in the suffix selection. We sketch the contours of the analysis in this section.

Czech paradigms are split into three main groups based on their gender, which can be masculine, feminine, or neuter. The masculine paradigms are furthermore divided based on their animacy. Each of the gender groups contains several paradigms.

While the affiliation of a noun to a given paradigm is, to a large extent, based on the combination of phonological shape, gender, and lexical semantics, to the best of our knowledge, it still carries a non-trivial level of arbitrariness (Williams et al. 2020, Guzmán Naranjo and Bonami 2021).

Here are the masculine inanimate paradigms (with syncretisms marked by shadowed cells):[10]

(43) SURFACE FORMS OF MASCULINE INANIMATE PARADIGMS 'TONGUE', 'BREAD', 'CASTLE' AND 'ROOM'

	'tongue'	'bread'	'castle'	'room'
NOM	jazyk	chleb-a	hrad	pokoj
ACC	jazyk	chleb-a	hrad	pokoj
GEN	jazyk-a	chleb-a	hrad-u	pokoj-e
LOC	jazyk-u	chleb-u	hrad-u	pokoj-i
DAT	jazyk-u	chleb-u	hrad-u	pokoj-i
INSTR	jazyk-em	chleb-em	hrad-em	pokoj-em

We can see that *jazyk* 'tongue' and *chleb-a* 'bread' share all their suffixes, except for the nominative and accusative. Based on that, *chleb-a* is usually not treated separately (see, e.g., Grepl et al. 2012). From a more descriptive perspective, it is understandable since it is also (to our knowledge) the only word with this exact suffix set. We treat it as a separate paradigm, though, in an attempt to extend the account as much as possible without relying on the concept of "exceptions".

Putting the instrumental aside for the moment, note that the first three paradigms share many suffixes, while the last one seems to be separate. The pattern behind which paradigms can share suffixes and which cannot has to do with the quality of the root-final consonant. Compare the two roots below:

(44) ROOT CHOOSES THE SUFFIX

		'tongue'	'room'
GEN		jazyk-**a**	pokoj-**e**
	*	jazyk-e	pokoj-a

A masculine root ending in *-k*, like *jazyk* 'tongue', never takes *-e* in the genitive, while a root ending in *-j*, like *pokoj* 'room', never ends in *-a* in the genitive.[11] It is not

[10] The masculine animates are added to the picture later, in the section 5.

[11] Although you can find genitive forms like *pokoj-a* 'room' in dialects.

phonologically impossible for a root ending in *-k* to take *-e*, though – the locative form of *jazyk* is *jazyc-e* (*-k* → *-c* is a regular palatalization in Czech). This means there must be more in play behind the restriction on distribution than phonology.

We can see the same split in the neuter paradigms:[12]

(45) SURFACE FORMS OF NEUTER PARADIGMS 'SEA' AND 'JACKET'

	'sea'	'jacket'
NOM	moř-e	sak-o
ACC	moř-e	sak-o
GEN	moř-e	sak-a
LOC	moř-i	sak-u
DAT	moř-i	sak-u
INSTR	moř-em	sak-em

Besides the instrumental, the two paradigms do not share any suffixes. There are again no neuter roots ending in *-ř* that would, e.g., take *-o* in the nominative or accusative. Vice versa for the neuter roots ending in *-k* – they never take *-e* in the two cases.

The split is drawn along the lines of palatalization. The roots ending in palatalized consonants take a different set of suffixes than the roots ending in non-palatalized consonants.[13]

Looking at the feminine paradigms now, they share some suffixes:[14]

(46) SURFACE FORMS OF FEMININE PARADIGMS 'PALM' AND 'BONE'

	'palm'	'bone'
NOM	dlaň	kost
ACC	dlaň	kost
GEN	dlan-ě	kost-i
LOC	dlan-i	kost-i
DAT	dlan-i	kost-i
INSTR	dlan-í	kost-í

Note that contrary to the orthography, the root *dlaň* is palatalized throughout the whole declension. Interestingly, the root *kost* 'bone' seems to be a mix of both palatalization groups. It ends in the non-palatalized *-t* in the nominative and accusative, but in the palatalized *-ť* in the rest of the cases (the table above follows orthography, but the *t* in the genitive through instrumental is indeed pronounced as *ť*). That is why some Czech grammars treat the *kost* paradigm as independent of the palatalized / non-palatalized

[12] There are two other traditional neuter paradigms, *kuře* 'chicken' and *zelí* 'cabbage'. *Kuře* will be discussed in section 4.3.2, *zelí* in section 4.3.3.

[13] The consonants in Czech divide into two main groups: the non-palatalized and palatalized. There are also ambiguous consonants that can go with either of the two groups:
non-palatalized: h, ch, k, r, d, t, n
palatalized: ž, š, č, ř, c, j, ď, ť, ň
ambiguous: b, f, l, m, p, s, v, z

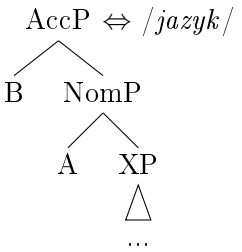
[14] There are two other traditional feminine paradigms, *žen-a* 'woman' and *růž-e* 'rose'. We get to those in section 6.2.

system (cf. Kodýtek 2010). Note, however, that it shares suffixes with *dlaň* 'palm, as a part of a hand' (which belongs to the palatalized group) only when the root has its palatalized form. We will treat it as a palatalized root and get back to the unexpected nominative/accusative form later.

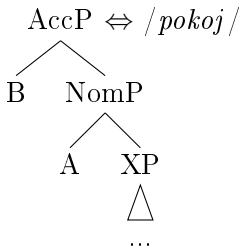
The distinction between the non-palatalized and palatalized group explains why in the masculine paradigms in (43), *pokoj* 'room' shares no endings with *jazyk* 'tongue', *chleb-a* 'bread' and *hrad* 'castle' as *pokoj* is the only palatalized paradigm in the group. It explains why the feminines *dlaň* 'palm' and *kost* 'bone' in (46) share endings as they both have a palatalized root-final consonant (in the relevant cases). And it also explains why the neuters *moř-e* 'sea' and *sak-o* 'jacket' in (45) do not share endings – each of them belongs to a different palatalization group.

We will show that once we understand the phonology, the suffixes are, in fact, the same. Specifically, there is one underlying suffix that is realized in different ways depending on its phonological environment. The argumentation will go as follows: We derive lexical entries for the non-palatalized and palatalized group separately. Upon comparing them, we will see that the same root sizes appear in both groups. For example, compare the non-palatalized *jazyk* 'tongue' and the palatalized *pokoj* 'room' below:

(47) LEXICAL ENTRY OF 'TONGUE'



(48) LEXICAL ENTRY OF 'ROOM'



They both spell out the functional sequence up to the accusative layer.

Once we pair the lexical entries by their root sizes, we will uncover an implicational relationship between the suffixes of the non-palatalized and palatalized paradigms. One such implication is that if *jazyk* has *-a* in a given case, *pokoj* will always have *-e* in the same case:

(49) a → e

Crucially, the left and the right side of (49) share something. In Element Theory (to be introduced later), *-e* is a combination of A + I.[15] So it looks like a more precise implication would be:

(50) A → A + I

The regularity uncovered by Element Theory will show that the palatalized side of the table is whatever suffix was in the non-palatalized paradigm with the addition of I:

[15] A more detailed explanation of how Element Theory works is offered when we get into the details of the account.

(51) V → V + I

We will argue that the only way the palatalized paradigms differ is in having a floating I in the lexical entry of their root:

(52) jazyk

(53) pokojI

This I then changes the quality of the suffix, which is underlyingly the same for both groups:

(54) jazyk + A → jazyka

(55) pokojI + A → pokoje

The end game here is to show that the combination of root size and Element Theory allows us to dispose of any special selection devices to determine which suffix goes where while making sense of the seeming complexity of the Czech nominal declension. In both the palatalized and non-palatalized group, it is simply the root size that decides the outcome of the derivation. And since root size is a tool extendable to different languages, the account stays well in line with the principles of the Universal Grammar.

There is one paradigm in particular, the masculine animate *pán* 'sir' (and its palatalized counterpart *muž* 'man') that presents a challenge to our system. See its forms below, compared to the forms of a masculine inanimate paradigm *jazyk* 'tongue':

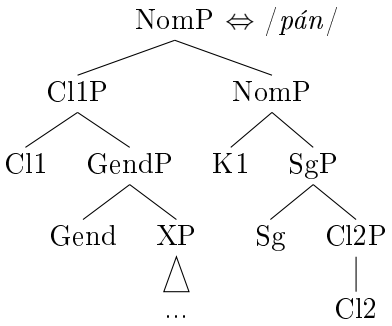
(56) SURFACE FORMS OF MASCULINE ANIMATE 'SIR' AND MASCULINE INANIMATE 'TONGUE'

	'sir' (Ma)	'tongue' (Mi)
NOM	pán	jazyk
ACC	pán-a	jazyk
GEN	pán-a	jazyk-a
LOC	pán-u	jazyk-u
DAT	pán-u	jazyk-u
INSTR	pán-em	jazyk-em

The two are almost identical, except for the difference in the accusative. The masculine animates *pán*/*muž* are well-known for the GEN-ACC syncretism (as opposed to the inanimate NOM-ACC syncretism of the inanimates). What we will see in the section 5 is that the way the spellout is set up, a simple root size account would be deriving the wrong suffix in the accusative of *pán* because a suffix from a different paradigm would be interfering.

That is where we discover the second property of roots – their shape. The roots in the *pán* paradigm will be suggested to look as follows:

(57) DLB LEXICAL ENTRY OF 'SIR'



The shape of the root will not only allow us to incorporate *pán* 'sir' with the rest of the system, but it will also offer a possible explanation for why there are some inanimates with the GEN-ACC syncretism despite it generally being perceived as a marker of animacy.

Now, let us look how this all plays out, starting with the easiest part – the non-palatalized paradigms. We will only need root size to account for them. The next section establishes an important basis for the analysis. For readers particularly interested in the palatalization differences and reducing the number of paradigms, sections 3 and 4 will be important. For readers interested in the syntax side, including the complex lexical entries, sections 3 and 5 should be read, while section 4 can be skipped.

3 Root size: the non-palatalized paradigms

This section proposes lexical entries for the non-palatalized paradigms using the root size account.

The non-palatalized paradigms in Czech can end either in a non-palatalized or an ambiguous consonant:

(58) NON-PALATALIZED AND AMBIGUOUS CONSONANTS IN CZECH

non-palatalized: h, ch, k, r, d, t, n
ambiguous: b, f, l, m, p, s, v, z

We open with three roots, *jazyk* 'tongue', *chleb-a* 'bread' and *sak-o* 'jacket'. Their nominative to genitive derivations give us a minimal example on which we can see how the actual account works. A follow-up to that is completing the locative to instrumental derivations in the subsection 3.2, where we also add a fourth non-palatalized paradigm, *hrad* 'castle'. In the last subsection (3.3), we show detailed tree derivations, and the aim is purely depictive, with no new information presented there.

Here are the surface forms of the paradigms we work with in this section, their syncretisms marked by colored cells:

(59) SURFACE FORMS OF 'BREAD', 'JACKET', 'TONGUE' AND 'CASTLE'

	'bread' (Mi)	'jacket' (N)	'tongue' (Mi)	'castle' (Mi)
NOM	chleb-a	sak-o	jazyk	hrad
ACC	chleb-a	sak-o	jazyk	hrad
GEN	chleb-a	sak-a	jazyk-a	hrad-u
LOC	chleb-u	sak-u	jazyk-u	hrad-u
DAT	chleb-u	sak-u	jazyk-u	hrad-u
INSTR	chleb-em	sak-em	jazyk-em	hrad-em

There is one neuter and three masculine paradigms in this group. All four share some suffixes, including a cross-gender syncretism between the one neuter (*sak-o* 'jacket') and the masculines.

We assume there is a single universal functional sequence. The one used in this work is adopted from Janků and Starke's (2019) work on Romance and extended by the Gend and Cl2 features since the number of declensions in Czech requires a more fine-grained sequence. We leave out the plural feature (found on top of Sg) as we are not looking at the Czech plural in this work.

The functional sequence we use is the following:

(60) FULL FUNCTIONAL SEQUENCE

XP > Gend > Mkd > Cl1 > Cl2 > Sg > K1 > K2 > K3 > K4 > K5 > K6

The XP on the left is a phrase, agnostically labeled as X, containing all the features thought to be located low in the structure, e.g., those related to mass. The details of

what exactly XP contains are irrelevant to the task at hand. Gend and Mkd are features related to the gender domain. While all nouns spell out Gend, Mkd is present only in feminine nouns.[16] Sg marks the singular in the number domain, and we assume it is always present. K1 to K6 are case features in a containment relationship, building the nominative, accusative, genitive, locative, dative, and instrumental.

Finally, Cl1 and Cl2 are a bit of a mystery for now. They are independently needed in Czech and Romance to account for the full variety of endings. Interpretation of these projections could be related to gender right below or number right above them. This is difficult to determine because the analysis is built on the idea that both are always included, which complicates any attempts for a contrastive analysis. The important part is, though, that since they are always present, they are not arbitrary class markers.

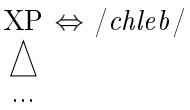
3.1 The idea in a nutshell

3.1.1 Nominative to genitive

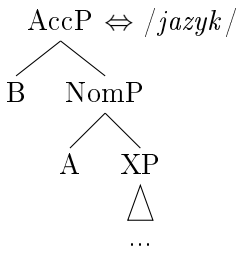
Let us now turn towards deriving the nominative, accusative and genitive of the roots *jazyk* 'tongue', *chleb-a* 'bread' and *sak-o* 'jacket'.

In our toy examples for *jazyk* 'tongue' and *chleb-a* 'bread' in section 1, we suggested the following lexical entries:

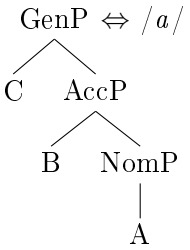
(61) LEXICAL ENTRY OF 'BREAD', OLD



(62) LEXICAL ENTRY OF 'TONGUE', OLD



(63) LEXICAL ENTRY OF -A, OLD

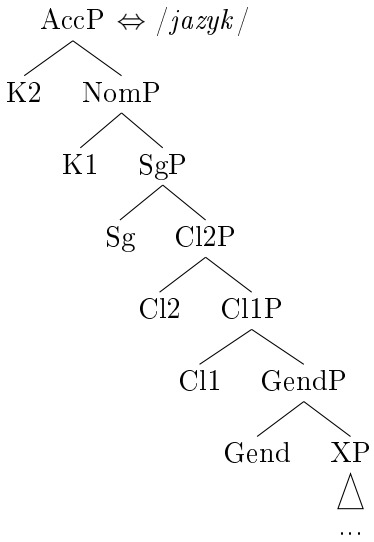


The key part is that the root *chleb* 'bread' needs to be smaller than the root *jazyk* 'tongue'. One of the arguments for that is that *jazyk* does not need any suffix in the nominative

[16] This work does not propose an answer to what distinguishes the masculine and the neuter structures. While there will have to be a syntactic difference somewhere since the two differ in the agreement, the noun roots themselves do not seem to mark any difference. So in the data pool we worked on, there was no reason for an extra feature or extra distinction.

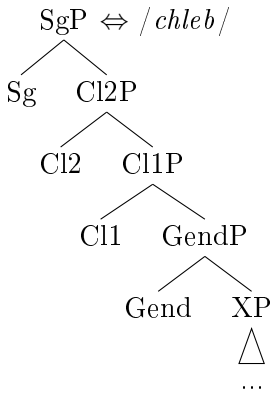
and accusative (while *chleb* does), which suggests that *jazyk*, unlike *chleb*, spells out the whole sequence up to the accusative. Applying this reasoning to the functional sequence in (60), we must update the lexical entry for *jazyk* 'tongue' to fit with our new fseq as follows (note that MkdP is missing because the root is not feminine):

(64) LEXICAL ENTRY OF 'TONGUE', NEW

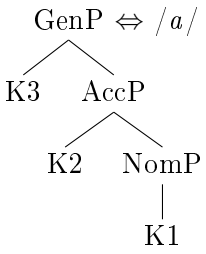


Since *chleb* 'bread' takes *-a* in the nominative, accusative and genitive, the simplest account (to be amended later) is as shown below:

(65) LEXICAL ENTRY OF 'BREAD', NEW



(66) LEXICAL ENTRY OF -A, NEW



The root grows to SgP, and then from NomP up, it gets finished by *-a*. If we compare (64) with (65), we see that *jazyk* spells out more features.

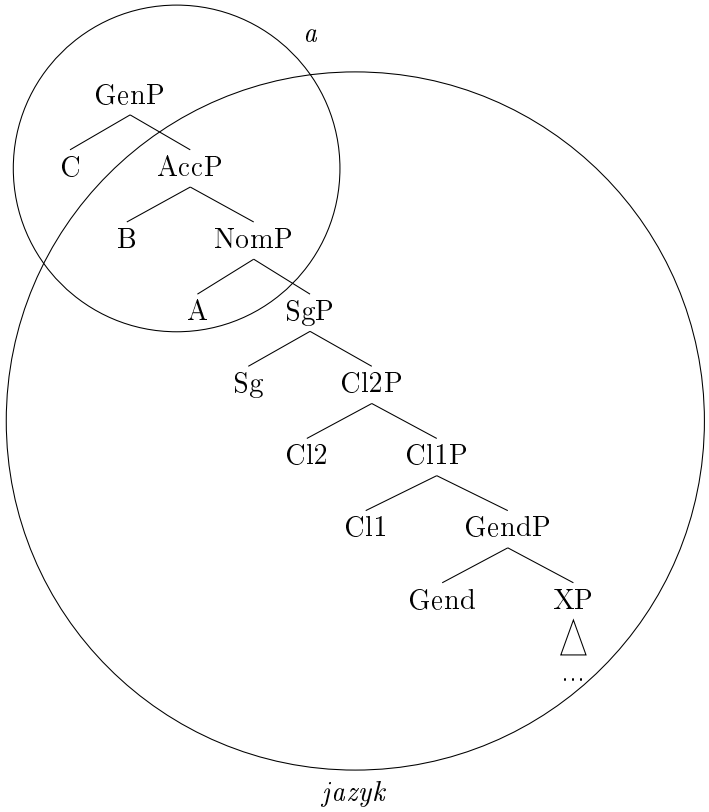
In the genitive, both roots unify in taking the suffix *-a*:

(67) SURFACE FORMS OF 'TONGUE' AND 'BREAD', NOM–GEN, -A SYNCRETISM

	'tongue'	'bread'
NOM	jazyk	chleb-a
ACC	jazyk	chleb-a
GEN	jazyk-a	chleb-a

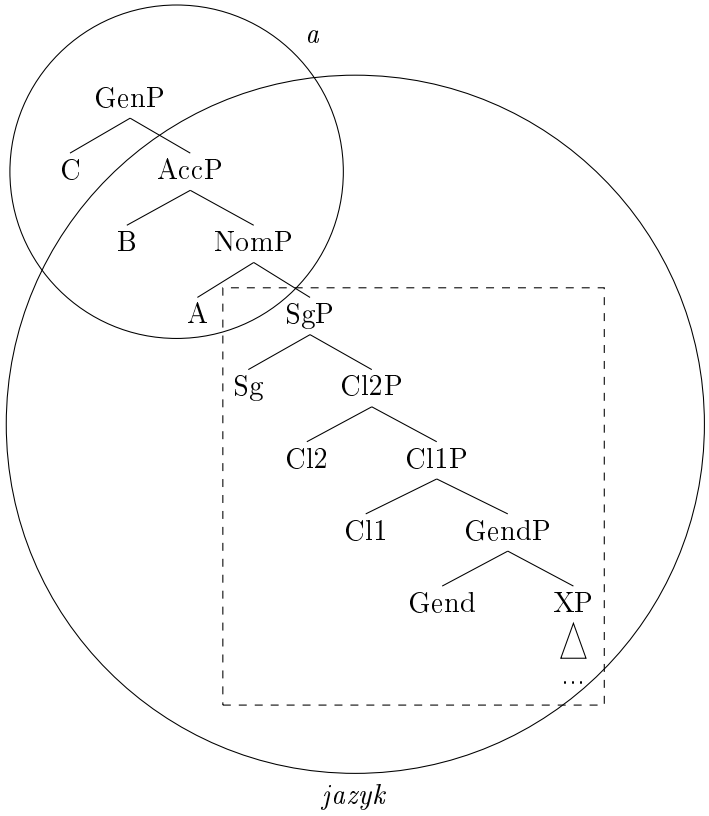
If we keep to the proposal that *jazyk* spells out all the features up to the accusative, then in the genitive, we get two overlapping lexical entries – the same as we have already seen in our simplified examples (refer back to (40)):

(68) SPELLOUT OF 'TONGUE', GEN



Due to the Superset Principle, the lower lexical entry (in this case *jazyk*) can shrink and spell out only the features up to Sg and hence take on *-a* without a problem. Below we can see the relevant subset marked by a dashed line:

(69) SPELLOUT OF 'TONGUE', GEN, USING A SGP SUBSET



What follows is that if in our system one root shows a specific suffix earlier and the other later, then the one that shows it earlier has to be smaller. We delve into a detailed explanation in the following section. For the moment, let us, however, turn back to the rest of the non-palatalized paradigms.

We have seen that *jazyk* has to be bigger than *chleb*:

(70) jazyk > chleb

Now, consider the root *sak* 'jacket'. It also has *-a* in the genitive but a different suffix in the nominative and accusative:

(71) SURFACE FORMS OF 'JACKET', 'TONGUE' AND 'BREAD'

	'jacket'	'tongue'	'bread'
NOM	sak-o	jazyk	chleb-a
ACC	sak-o	jazyk	chleb-a
GEN	sak-a	jazyk-a	chleb-a

Following the logic sketched above, since *sak* 'jacket' gets to *-a* later than *chleb* 'bread', it has to be bigger:

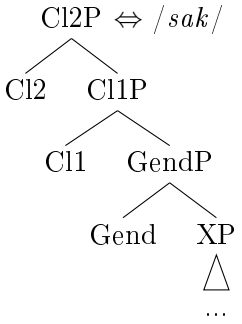
(72) sak > chleb

At the same time, *sak* 'jacket' has to be smaller than *jazyk* 'tongue' because, unlike *jazyk*, it cannot spell out the whole sequence and still needs a suffix even in the two lowermost cases. This implies that *jazyk* 'tongue' is the biggest of the three, *sak* 'jacket' is smaller and *chleb* 'bread' is the smallest:

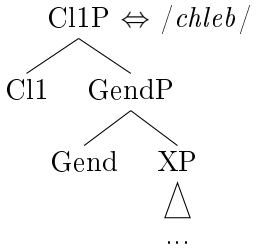
(73) jazyk > sak > chleb

We keep the lexical entry for *jazyk* the same as in (64), i.e., spelling out everything up to the accusative feature. Below is a lexical entry we suggest for *sak* 'jacket' and also an amended lexical entry for *chleb* 'bread', which now needs to end lower to make space for *sak*:

(74) LEXICAL ENTRY OF 'JACKET'



(75) LEXICAL ENTRY OF 'BREAD'



Note that the size of *sak* (and hence then also *chleb*) could be bigger than what we suggest. The root *sak* shows a suffix in the nominative, so it has to be smaller than K1 – to allow the suffix to spell out at least K1. Hence, we can let the root grow to Sg or any feature below Sg. We chose to let it grow one notch less, to Cl2. The reason for that is that the root growing to Sg puts the foot of the *-o* in the case domain. The case domain is above the number domain, and so *-o* would be unspecified for the number. Since we only ever see *-o* in the singular, we want to capture that by having *-o* (and all the other singular suffixes) spell out the singular feature.[17]

From now on, we will sometimes use lexicalization tables instead of trees. It needs to be underscored that the tables are only a tool helping us see the whole sequence at once and quickly compare which morphemes spell out which chunk of the functional sequence. They are still representing tree structures and the system described above. Here is the table representation of the three roots dealt with in this section so far (compare with the trees in (64), (74) and (75)):

[17] Technically speaking, we could also let *sak* grow to Sg and still foot *-o* in Sg, too. The two lexical entries would overlap, and the root would then have to shrink to get the suffix in the same way we saw *jazyk* 'tongue' shrink for *-a* in the genitive. Any shrinking, however, adds complexity to the derivation. Also, from an acquisition view, a shrinking that cannot be detected on the surface would be impossible to acquire. That is why we avoid using it if unnecessary.

(76) LEXICALIZATION TABLE OF 'BREAD', 'JACKET' AND 'TONGUE', ROOTS

XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
chleb										
sak										
jazyk										

For better navigation, we refer to the different paradigms using a naming convention based on the root size. XL for *jazyk* 'tongue', because it is the largest root we will see in this work. M for *sak* 'jacket'. And S for *chleb* 'bread'. We're going for XL for *jazyk* 'tongue' instead of just L because later on, we add a paradigm size in between *jazyk* 'tongue' and *sak* 'jacket':

(77)

XL-sized root:	jazyk
M-sized root:	sak
S-sized root:	chleb

Coming back to the roots in the table in (76) – since the syntactic structure has to contain features up to K1 in the nominative, and since the two smaller roots cannot reach that high, they need to be finished by a suffix. The suffixes need to start "above" the root that takes them, which is why the root selects them in the first place. That leads to the entries below:

(78) LEXICALIZATION TABLE OF 'JACKET' AND 'BREAD', NOM

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	sak				o						
	chleb			a							

The lexical entry for *-o* starts at the Sg feature because that way, it will be automatically picked up by M-roots like *sak*. The lexical entry for *-a* is footed in Cl2 to be picked up by S-roots like *chleb*.

We shall let both suffixes grow above K1. The *-o* must reach to K2 since the accusative is the highest case we see it in. The *-a* must go to K3 because it also shows in the genitive. Below, we see the entries for each of the relevant cases separately:

(79) LEXICALIZATION TABLE OF 'BREAD', NOM–GEN

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	chleb			a							
ACC	chleb			a							
GEN	chleb			a							

LEXICALIZATION TABLE OF 'JACKET', NOM–ACC

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	sak				o						
ACC	sak				o						

It should be remembered that all three instances of *-a* correspond to a single lexical entry, which extends to K3. In the accusative and nominative, a subset of it is spelled out using the Superset Principle. The same goes for *-o*.

The XL-roots like *jazyk* grow all the way to K2 and then shrink in the genitive. Below is a table representation of that:

(80) LEXICALIZATION TABLE OF 'TONGUE', NOM–GEN

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	jazyk										
ACC	jazyk										
GEN	jazyk				a						

Now that we have suggested the first few real lexical entries, let us pause and delve into the details of how the derivation actually proceeds.

3.1.2 Derivations walk-through

We skipped over many details in the preceding sections, including morpheme order. Let us come back to it now and do a detailed walk-through of how the syntactic derivation unfolds in Nanosyntax. Readers well acquainted with spellout-driven movement may want to skip directly to subsection 3.2, where we continue with proposing lexical entries for structurally higher cases.

We start by looking at the derivation of the nominative of S-roots, like *chleb-a* 'bread'. Its nominative to genitive forms are repeated below:

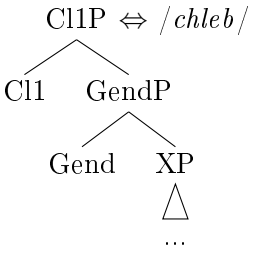
(81) SURFACE FORMS OF 'BREAD', NOM–GEN

	'bread'
NOM	chleb-a
ACC	chleb-a
GEN	chleb-a

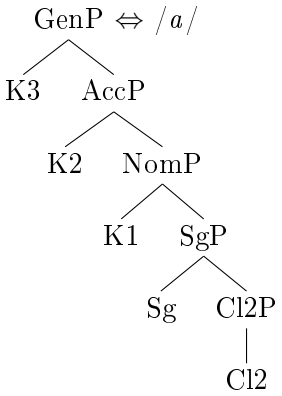
Here are the relevant lexical entries for this derivation:[18]

[18] To save space, we always only show the lexical entries relevant for any particular derivation. However, in the appendix in 8.3, there are all the (final) lexical entries to see the big picture and check that, e.g., no unsolvable competition happens across paradigms.

(82) LEXICAL ENTRY OF 'BREAD'



(83) LEXICAL ENTRY OF -A



We take the moment when XP is built as a starting point (84), staying away from the details of it because it is not relevant to the task at hand.

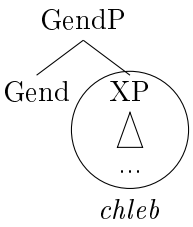
The spellout in Nanosyntax is cyclic, meaning that the structure must be successfully spelled out at every Merge. In order to be spelled out, the node XP has to find a match in the lexicon. Since the root *chleb* can do XP through the Superset Principle (cf. the lexical entry in (82)), that is what our spell out at this stage is, as in (85):

(84) DERIVATION OF 'BREAD', NOM, INTERMEDIATE STEP

(85) DERIVATION OF 'BREAD', NOM, INTERMEDIATE STEP

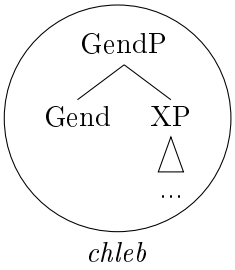
Then we merge XP with the next feature from the universal functional sequence, which is Gend, creating a GendP phrase (86):

(86) DERIVATION OF 'BREAD', NOM, INTERMEDIATE STEP



Spelling out a newly created structure as it is is always preferred to any movement. Hence if the structure can find a lexical entry that matches the GendP node, it has to use it. Because GendP is a proper subset of the lexical entry for *chleb* (82), the structure will still be spelled out by the root:

(87) DERIVATION OF 'BREAD', NOM, INTERMEDIATE STEP



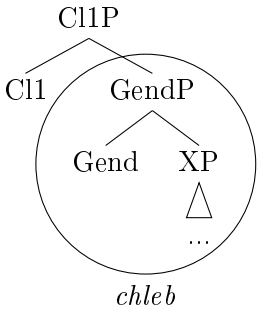
What we just saw is the first part of the spell out algorithm used by Nanosyntax (Caha 2021c, based on Starke 2018):

(88) Merge F and spell out.

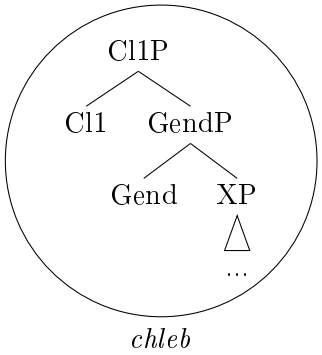
It follows that a root always grows as much as possible before it resolves to take any affix.

Next, in the same manner, we merge Cl1, creating a Cl1P phrase (89). The new phrase can also be spelled out in full by *chleb*, as shown in (90):

(89) DERIVATION OF 'BREAD', NOM, INTERMEDIATE STEP

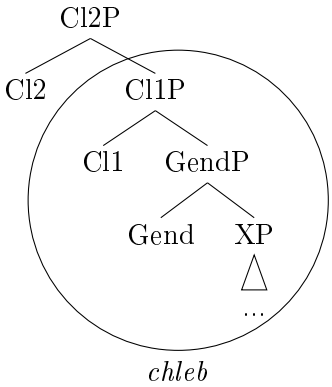


(90) DERIVATION OF 'BREAD', NOM, INTERMEDIATE STEP



Then we merge Cl2, creating a Cl2P phrase:

(91) DERIVATION OF 'BREAD', NOM, INTERMEDIATE STEP



The lexical entry for *chleb* is no longer able to spell out the whole structure because S-roots only reach to Cl1.[19] This is when the movement kicks in.

Nanosyntax uses a so-called spellout-driven movement, named after the fact that it gets triggered to save a spellout of a structure. There is a strict algorithm for it:

(92) THE SPELLOUT ALGORITHM, Caha (2021c), based on Starke (2018)

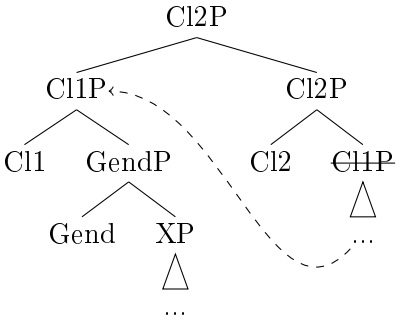
a. Merge F and spell out.

b. If (a) fails, move the Spec of the complement and spell out.

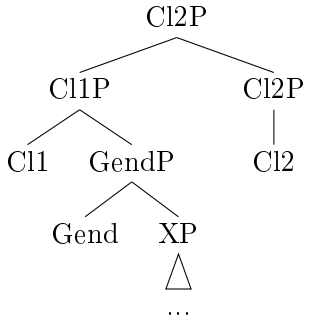
c. If (b) fails, move the complement of F and spell out.

This means that when the lexical entry for *chleb* fails to spell out the whole structure in (91), which is the first option of the spellout algorithm, the second option gets triggered – moving the Spec of the complement. In our case, it would mean Cl1. However, the spellout algorithm operates solely on phrases, and since there is no phrase in the Spec, this option will not be possible. The third step of the algorithm then gets triggered – moving the complement itself, as in (93). The spell out-driven movement leaves no traces, and so the new structure will look like the one in (94):

(93) DERIVATION OF 'BREAD', NOM, INTERMEDIATE STEP



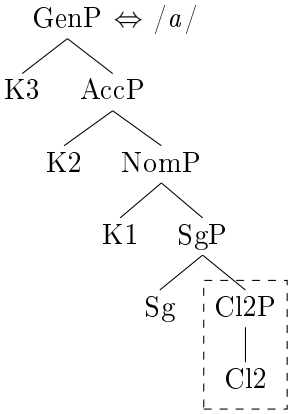
(94) DERIVATION OF 'BREAD', NOM, INTERMEDIATE STEP



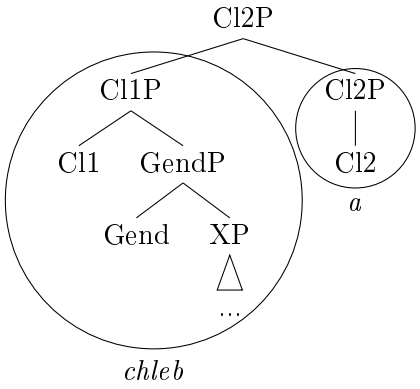
The left branch can still be spelled out by *chleb* as before. The movement has, however, opened up a new spell out option for the Cl2P node. Since it has no complement now, it can be spelled out independently. Only one lexical entry in our current lexicon starts at Cl2. We repeat it in (95), and the dashed line highlights the relevant subset. The spell out we now get is shown in (96):

[19] Note that an M-root, like *sak* 'jacket', could technically spell out the whole structure and hence replace the concept of *chleb*, given all the rules we have presented so far. We stay away from the details of this particular issue, but we work under the assumption that concepts (like *JACKET*, *BREAD* or *KIND*) do not override each other. Rather, they are decided based on "what the speaker wants to say". In this, concepts differ from non-conceptual lexical entries, like case suffixes, which compete with each other. The informal reasoning is simple – we do not want to end up speaking about jackets when the topic on our mind is bread. For the overall logic, see, e.g., Starke (2014). For a suggestion of technical implementation of the issue, see Caha et al. (2019b).

(95) LEXICAL ENTRY OF -A, CL2P SUBSET

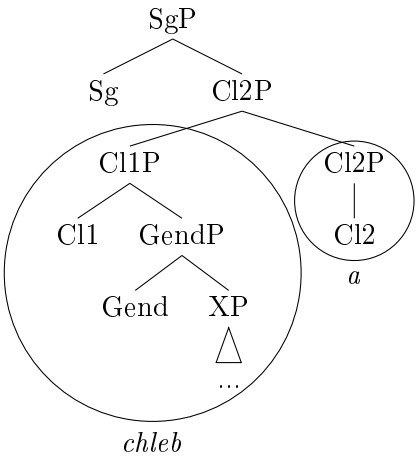


(96) DERIVATION OF 'BREAD', NOM, INTERMEDIATE STEP



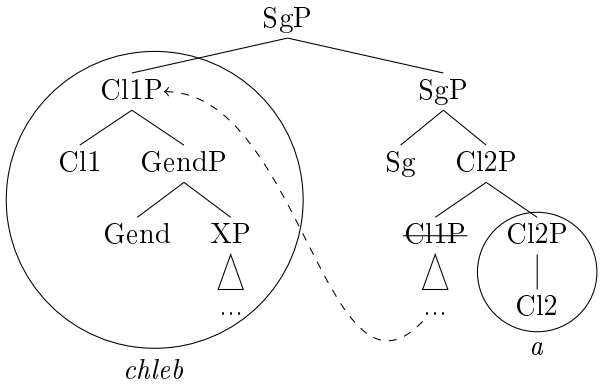
Every feature in the current structure is taken care of, so we can move on to merging the next feature, which is Sg (97):

(97) DERIVATION OF 'BREAD', NOM, INTERMEDIATE STEP

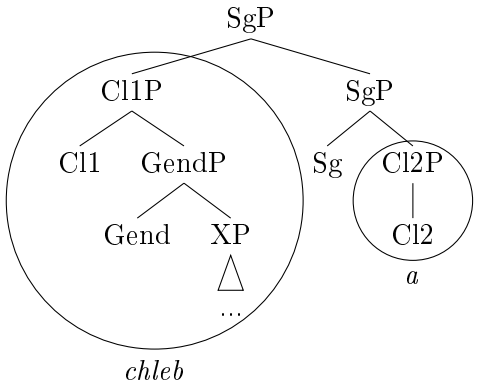


First, we try to spell out the whole structure, but no entry in the lexicon matches it. The second step in the spell out algorithm is to move the Spec of the complement. This time, there is a phrase in that position (Cl1P), so we move it (98), creating the structure in (99):

(98) DERIVATION OF 'BREAD', NOM, INTERMEDIATE STEP

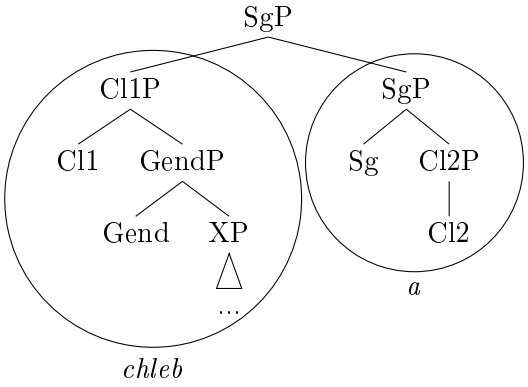


(99) DERIVATION OF 'BREAD', NOM, INTERMEDIATE STEP



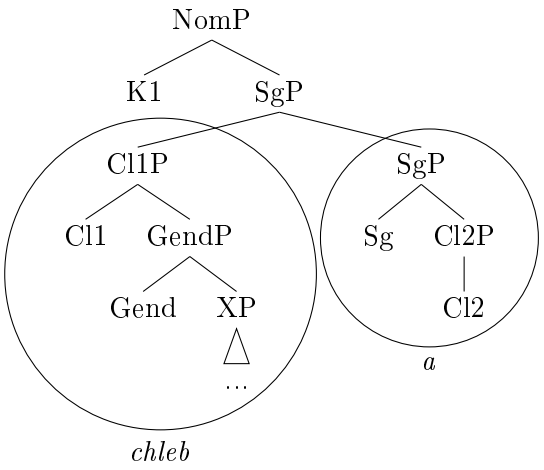
The singular feature still needs to be spelled out, so we go in the lexicon and see if there is a match for the SgP. It turns out that SgP can still be spelled out by the same lexical entry, namely *-a*:

(100) DERIVATION OF 'BREAD', NOM, INTERMEDIATE STEP



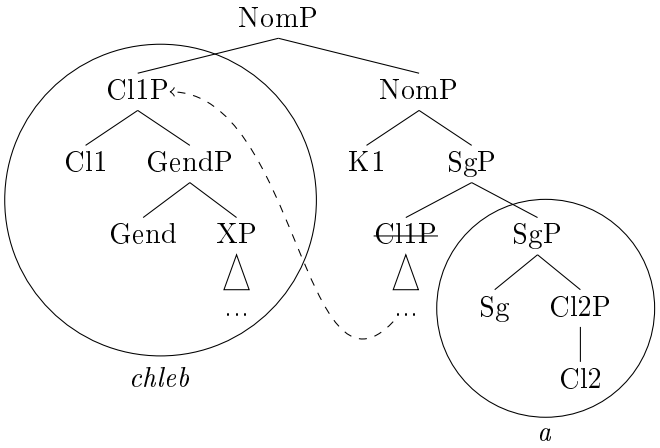
Finally, we merge K1, which is the last feature needed for the spell out of the nominative:

(101) DERIVATION OF 'BREAD', NOM, INTERMEDIATE STEP

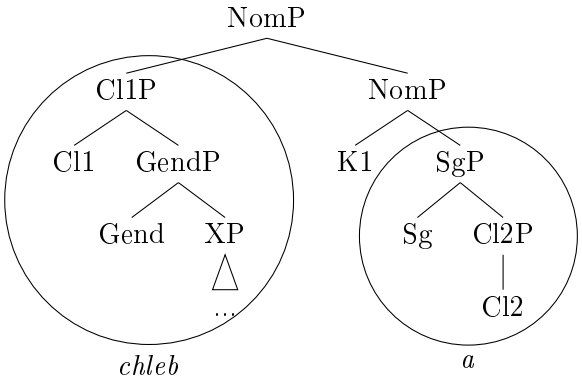


Just as in the preceding cycle, there is no match for the whole NomP. The Spec movement gets triggered (102), resulting in the structure in (103):

(102) DERIVATION OF 'BREAD', NOM, INTERMEDIATE STEP

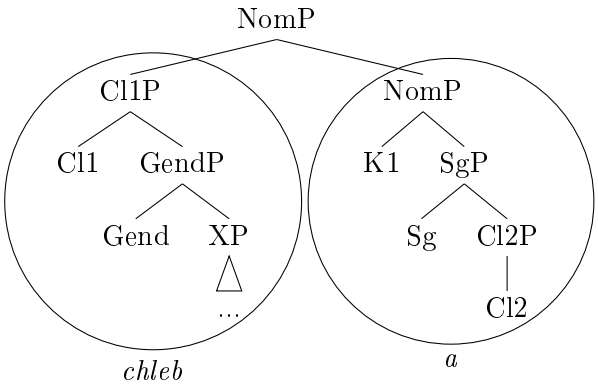


(103) DERIVATION OF 'BREAD', NOM, INTERMEDIATE STEP



The lower NomP goes in the lexicon and finds that it can still be spelled out as *-a*:

(104) DERIVATION OF 'BREAD', NOM, FINAL FORM



At this point, we have successfully derived the nominative of the root *chleb*. Since the accusative and genitive have the same form, their derivation would continue in the same manner by merging K2 and K3, respectively, bringing the Spec of the complement above those features, and spelling out the structure with the same two lexical items as we did in the nominative.

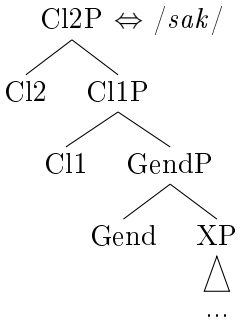
Let us now turn to the derivation of the nominative *sak-o* 'jacket' (M-root). The process will be the same, just the lexical items will differ. Here are the relevant forms repeated:

(105) SURFACE FORMS OF 'JACKET', NOM–GEN

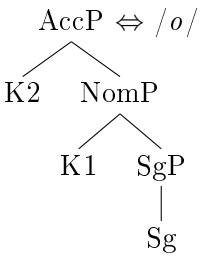
	'jacket'
NOM	sak-o
ACC	sak-o
GEN	sak-a

The lexical entries relevant for this derivation are repeated below:

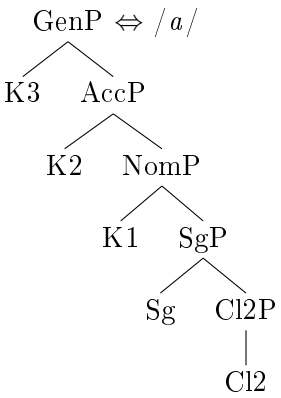
(106) LEXICAL ENTRY OF 'JACKET'



(107) LEXICAL ENTRY OF -O



(108) LEXICAL ENTRY OF -A



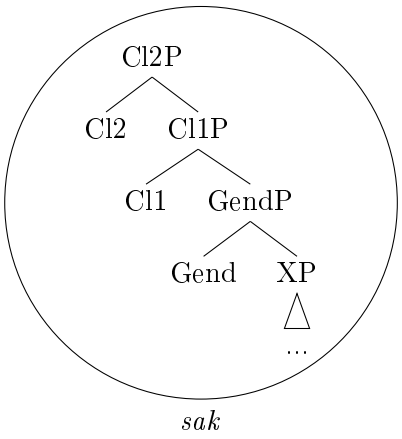
Once again, we start by putting together the XP (109). The root can spell it out, as seen in (110):

(109) DERIVATION OF 'JACKET', NOM, INTERMEDIATE STEP

(110) DERIVATION OF 'JACKET', NOM, INTERMEDIATE STEP

We proceed by merging Gend, Cl1, and Cl2, all of which will be spelled out without any movement. We are saving time and space here, but keep in mind that the derivation would proceed step-by-step, as we saw with *chleb* 'bread', until we would reach this stage:

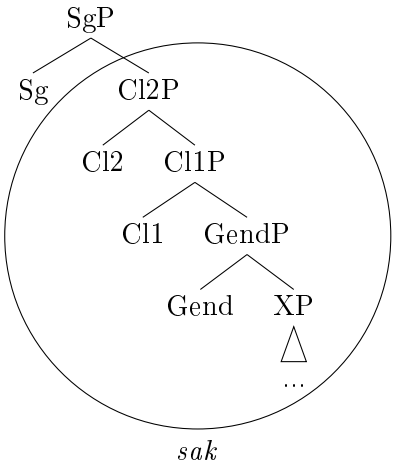
(111) DERIVATION OF 'JACKET', NOM, INTERMEDIATE STEP



In the structure above, we can see the root *sak* growing bigger than *chleb* before. Where *chleb* (S-root) stopped one feature lower, at Cl1, *sak* (M-root) grows to Cl2.

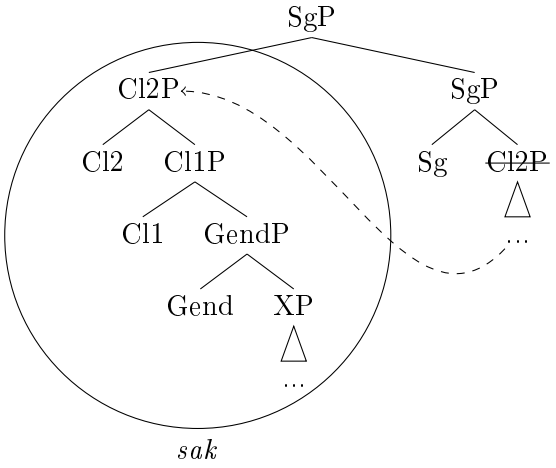
At this point, we merge Sg, as in (112):

(112) DERIVATION OF 'JACKET', NOM, INTERMEDIATE STEP

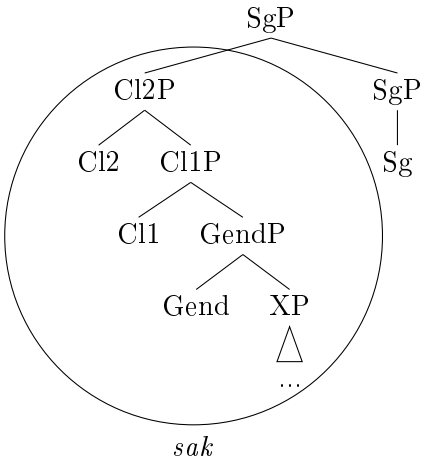


The root cannot spell out the whole structure anymore, so the movement gets triggered. The Spec movement does not apply, so the whole complement moves over SgP, as in (113), resulting in the structure in (114):

(113) DERIVATION OF 'JACKET', NOM, INTERMEDIATE STEP



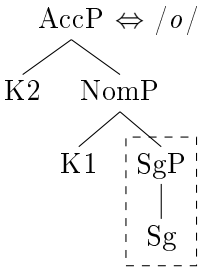
(114) DERIVATION OF 'JACKET', NOM, INTERMEDIATE STEP



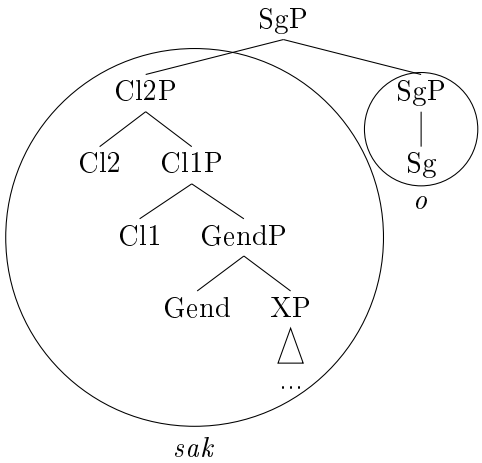
The SgP is freed up to be spelled out on its own now. It goes to the lexicon and sees that it can be spelled out (only) by the lexical entry for *-o*. Note that *-a* as seen in *chleb* would not fit here because the lexical entry for *-a* starts lower (at Cl2). Differing root sizes leave different chunks of functional sequence to be spelled out and hence take different suffixes. This is how we model allomorphy.

In (115), we repeat the lexical entry for *-o*, with a dashed line around the subset our structure uses. In (116), we see the result of the spellout in the syntactic structure:

(115) LEXICAL ENTRY OF -O

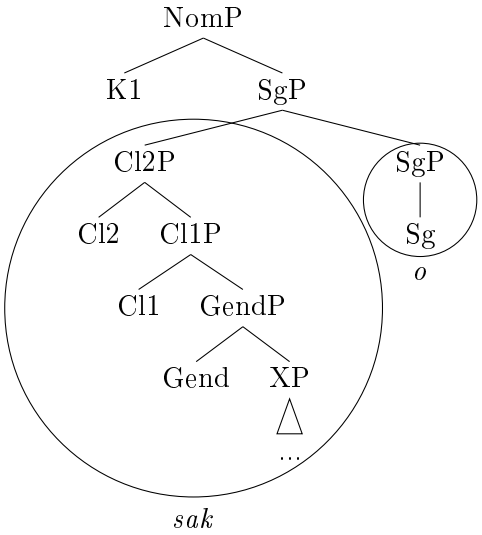


(116) DERIVATION OF 'JACKET', NOM, INTERMEDIATE STEP



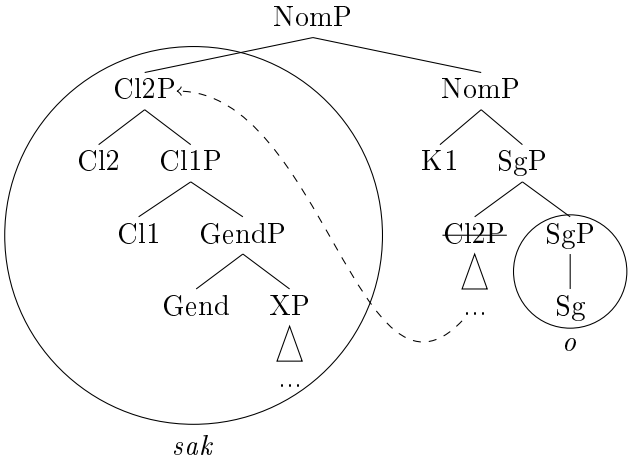
The last step in the nominative is routine by now. We add a K1 feature, as in (117):

(117) DERIVATION OF 'JACKET', NOM, INTERMEDIATE STEP

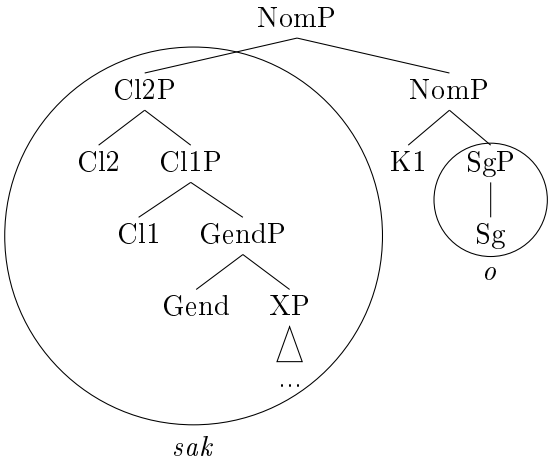


The structure tries to spell out the highest node and fails. Therefore, it proceeds to the second step of the spellout algorithm, which is moving the Spec, as in (118). This results in the structure in (119):

(118) DERIVATION OF 'JACKET', NOM, INTERMEDIATE STEP

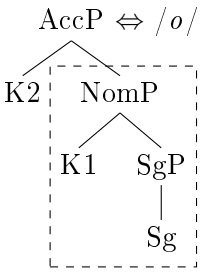


(119) DERIVATION OF 'JACKET', NOM, INTERMEDIATE STEP

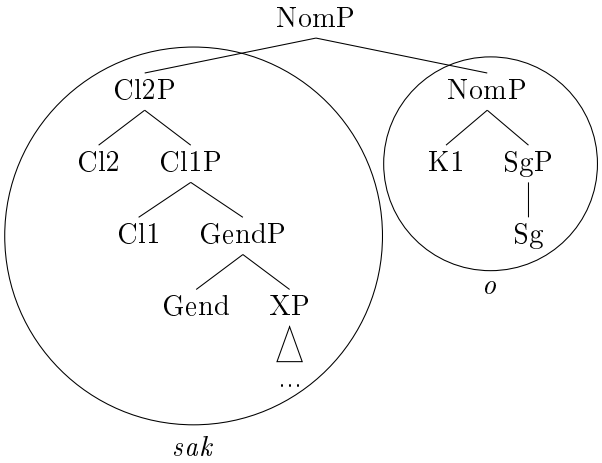


In the next step, the new highest node tries to spell out but fails, so the spellout goes lower. The left daughter is already spelled out by *sak*. The right-hand daughter is not spelled out, so the structure attempts to spell that node. The right branch NomP node finds a match with the same item as the SgP did. We can see that in (120), where the dashed line shows the new subset used for the spell out. In (121) we see the finished spell out and also the finished derivation of the nominative *sak-o*:

(120) LEXICAL ENTRY OF -O, NOMP SUBSET

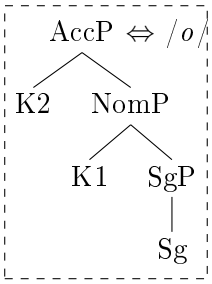


(121) DERIVATION OF 'JACKET', NOM, FINAL FORM

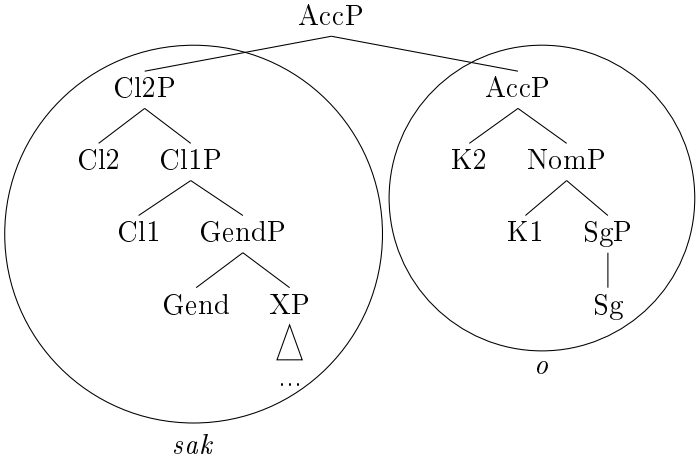


The accusative would continue simply by adding the K2 feature on top and bringing the Spec of the complement over it. The right daughter of the highest node would still be spelled out by *-o* because the lexical entry reaches to the accusative:

(122) LEXICAL ENTRY OF -O

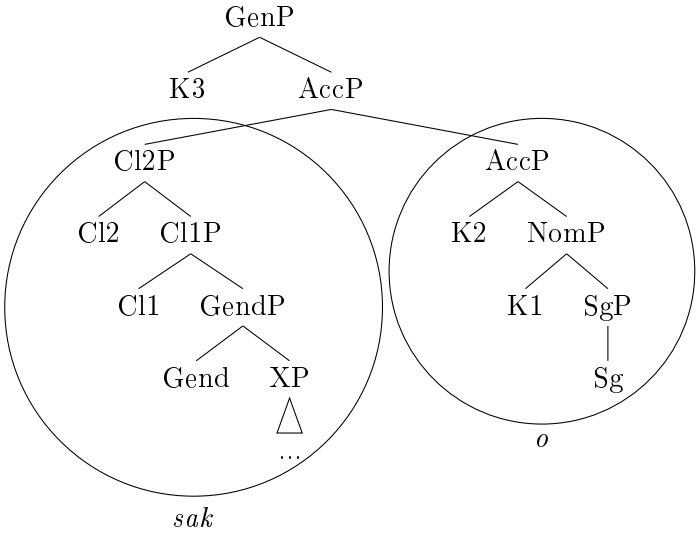


(123) DERIVATION OF 'JACKET', ACC, FINAL FORM



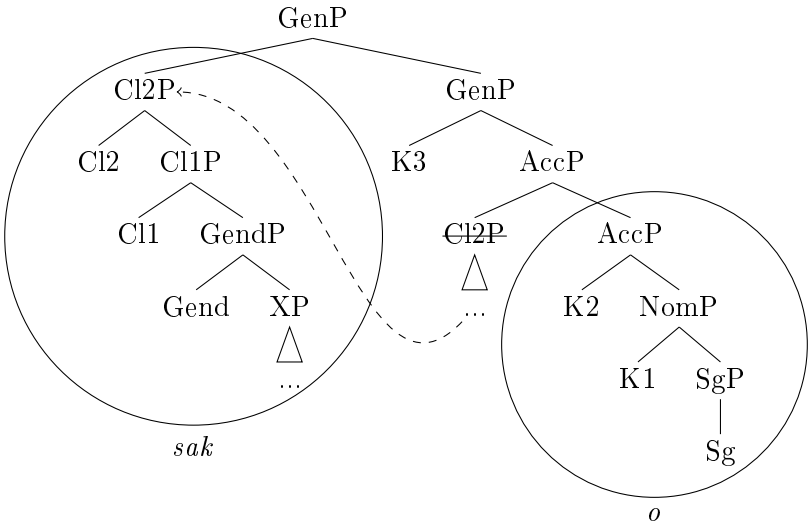
Now, interesting things start to happen when we want to derive the genitive. Consider adding K3 to the structure, as in (124):

(124) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP

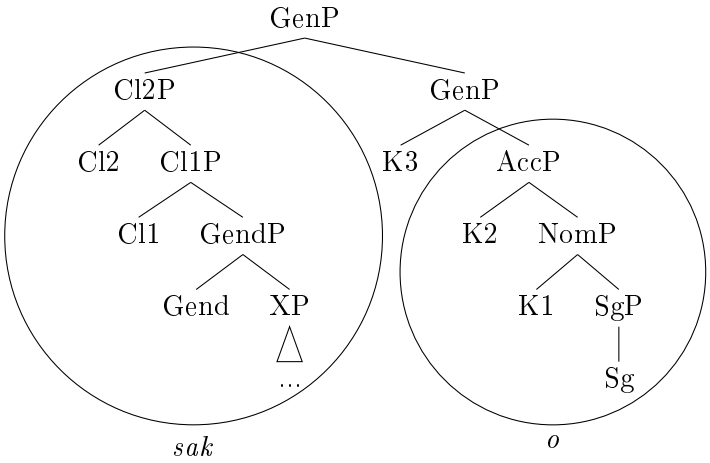


The structure cannot be spelled out as it is, so the Spec movement gets triggered (125), resulting in the structure in (126):

(125) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP

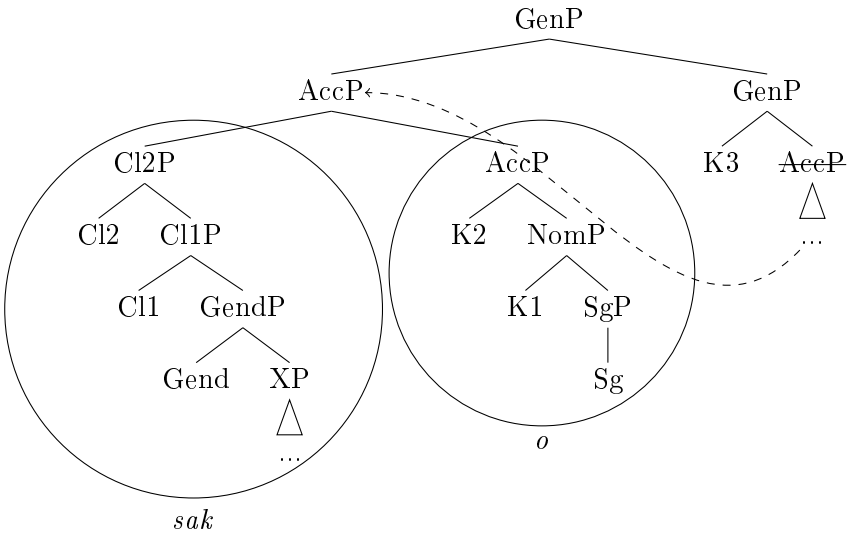


(126) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP

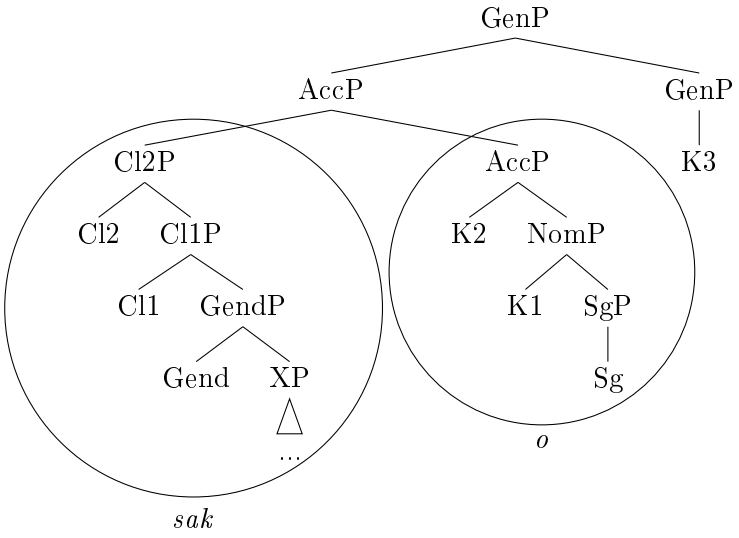


The right branch can no longer be spelled out by the lexical entry for *-o* because *-o* is not specified for K3. In fact, if we refer back to the lexical entries in (106)–(107), no entry starts at SgP and goes up to GenP. That means that moving the Spec did not help. So we undo the movement, which brings us back to the stage in (124), and we try the third step of the spell out algorithm, which is moving the whole complement:

(127) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP



(128) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP



This movement leaves GenP to be spelled out on its own. This attempt to save the structure fails as well because there is no lexical entry that starts at GenP. So even this has to be undone back to the stage in (124).

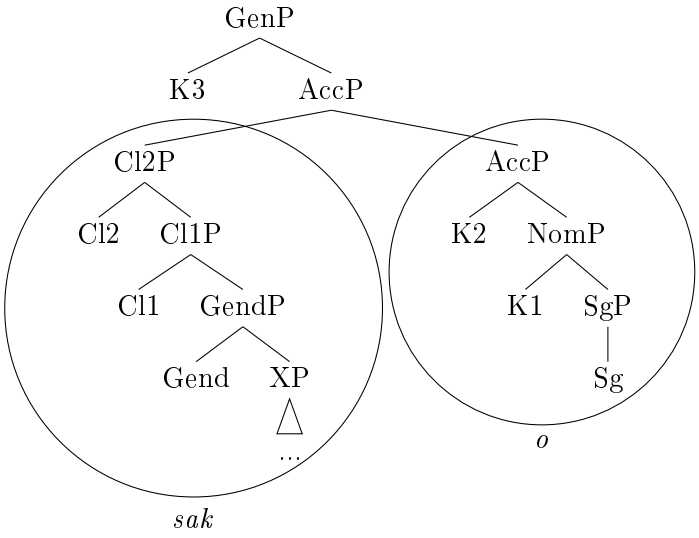
If all the options of the spellout algorithm fail, as we have just seen, a procedure called backtracking gets triggered. So far, we have only mentioned backtracking in terms of applying the Superset Principle, but is it a bit more particular than that. Backtracking, at its core, is undoing the syntactic derivation that already happened, in an attempt to save the spell out:

(129) BACKTRACKING, Caha (2021c), based on Starke (2018)

When spellout fails, go back to the previous cycle, and try the next option for that cycle.

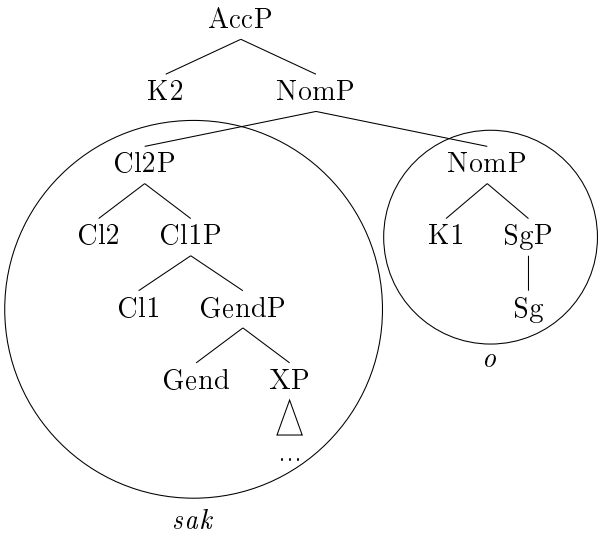
What does it mean for the spell out of the genitive of *sak*? First, let us repeat what the current structure is:

(130) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP, BACKTRACKING



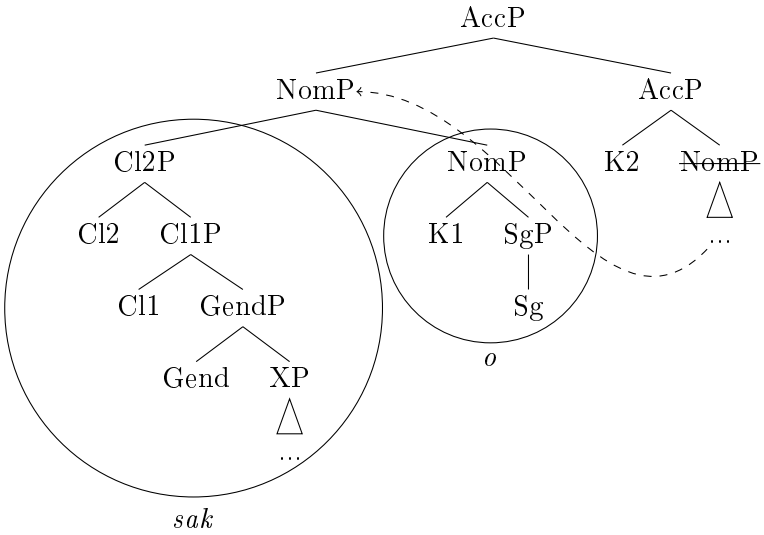
No movement helped to spell K3 out. So we undo the merging of K3, and, as stated in (129), we go to the previous cycle, which is merging K2. We find ourselves back to this structure:

(131) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP, BACKTRACKING

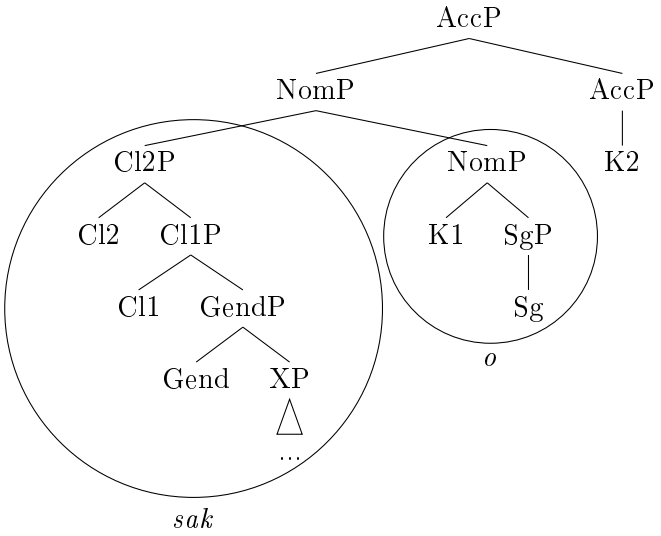


When we were at this stage before, we spelled out AccP by moving the Spec of the complement. That has, however, led to the failed derivation of K3 higher up. So we try the following option for this cycle, which is moving the whole complement (132), resulting in (133):

(132) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP, BACKTRACKING

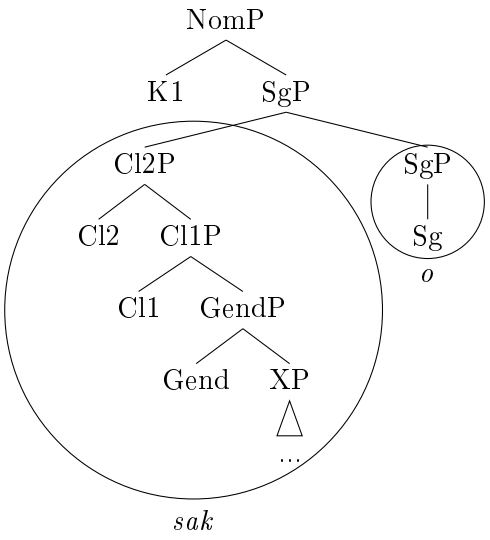


(133) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP, BACKTRACKING



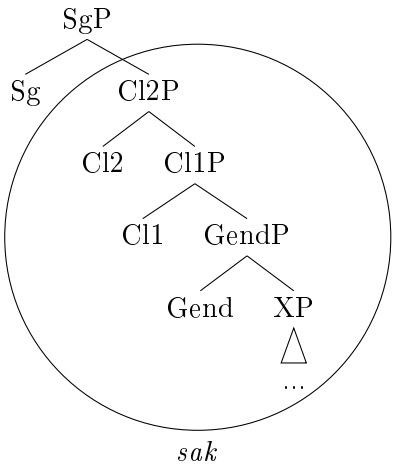
At this point, we try to spell out AccP on the right branch but fail because there is no lexical entry with AccP at its bottom. So the last movement has to be undone back to the stage in (131). There is no other step in the spellout algorithm, so the structure backtracks to an even earlier stage:

(134) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP, BACKTRACKING



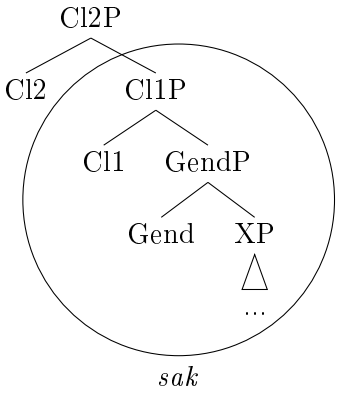
The same happens, as we just did with the accusative stage. When we were at the nominative stage before, we did the Spec movement. That led to a failed spellout at K3, so we try another option for this cycle – moving the whole complement. This movement leaves K1 to be spelled out on its own, and even this fails because there is no lexical entry able to spell out just K1. So we again undo the movement and backtrack still one feature lower, to right after merging Sg:

(135) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP, BACKTRACKING



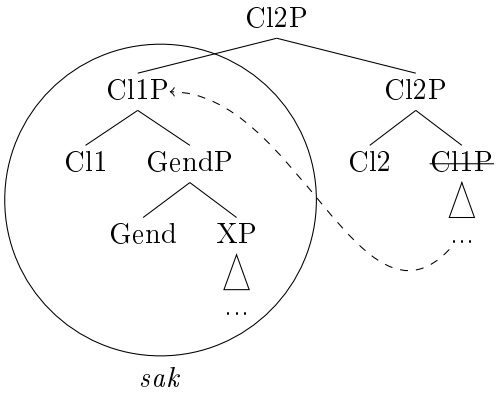
Here, we have gone directly for the complement movement because there is no Spec to be moved, which means there are no more steps to try. Hence we backtrack one step lower:

(136) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP, BACKTRACKING

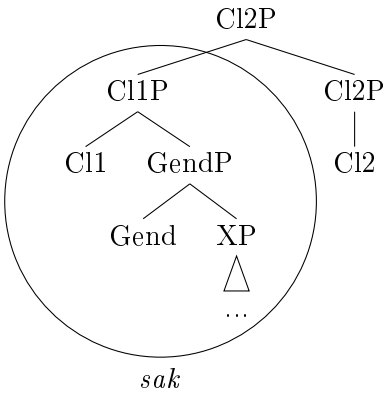


At this stage, we have previously left the structure as it was because the root was able to spell out Cl2P. So this time, we go straight to moving the complement of Cl2P (skipping the Spec movement since there is no Spec):

(137) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP, BACKTRACKING



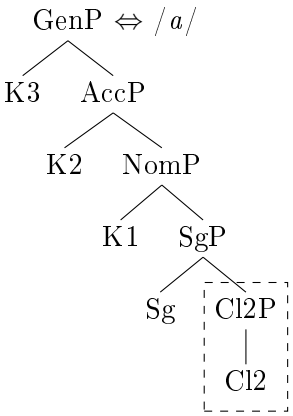
(138) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP, END OF BACKTRACKING



We find a lexical entry to spell out the right branch. The dashed line shows the part of

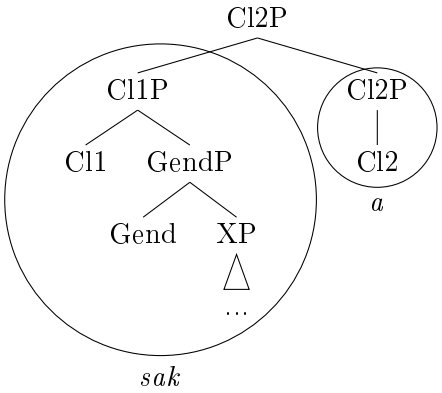
the lexical entry that matches the right branch of our newest syntactic structure:

(139) LEXICAL ENTRY OF -A



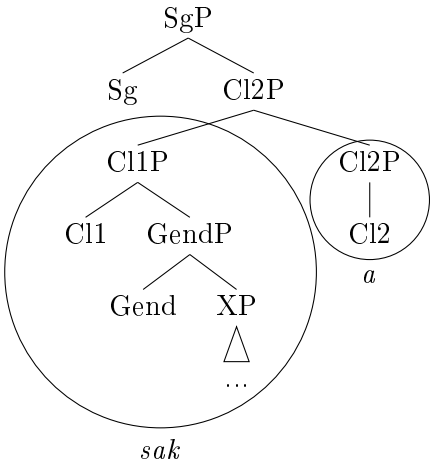
The found match, finally, results in a successful spell out:

(140) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP



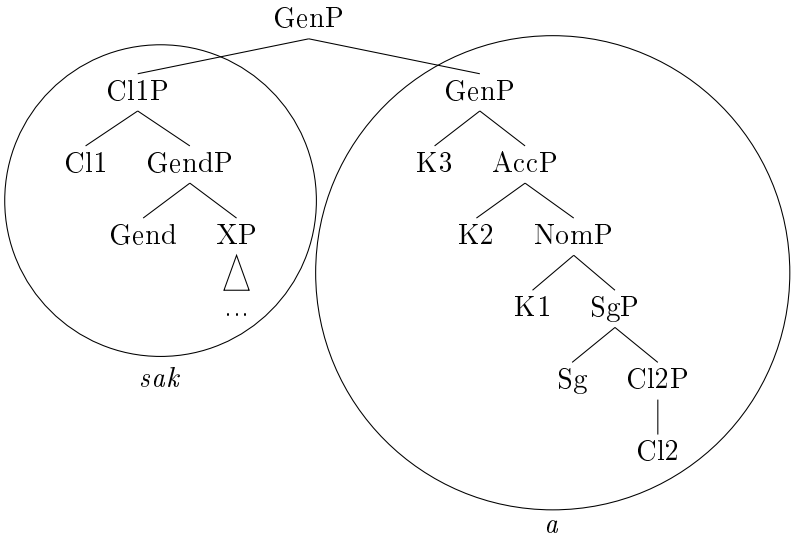
Since everything is spelled out now, we can proceed to merge the next feature, which is Sg:

(141) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP



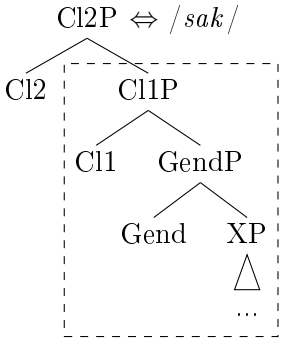
The rest of the derivation up to the genitive proceeds without any surprises. We move Cl1P above SgP and spell out Cl2+Sg as *-a*. We merge K1, move Cl1P above it and spell out Cl2+Sg+K1 as *-a*. The same for K2 and K3, resulting in the structure below:

(142) DERIVATION OF 'JACKET', GEN, FINAL FORM



We have successfully derived the genitive *sak-a* 'jacket'. Two things need to be repeated at this stage. First, the root shrank in the process. While it was spelled out in its full length in the nominative and accusative, it shrank and only spelled out a subset in the genitive. Below, we repeat the full lexical entry. The dashed line shows the subset used in the genitive:

(143) LEXICAL ENTRY OF 'JACKET', Cl1P SUBSET



Second, backtracking allows us to have a single lexical entry for affixes shared across paradigms. In the example above, *sak* 'jacket' shared the genitive *-a* with *chleb* 'bread'. This can be easier grasped through a lexicalization table:

(144) LEXICALIZATION TABLE OF 'BREAD', NOM–GEN

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	chleb			a							
ACC	chleb			a							
GEN	chleb			a							

LEXICALIZATION TABLE OF 'JACKET', NOM–GEN

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	sak				o						
ACC	sak				o						
GEN	sak			a							

There, we can see how *sak* starts bigger, but backtracks to *-a*, and that the same *-a* (starting at the same place and going all the way to the same place) is in *chleb* as well.

The derivation of *jazyk* 'tongue' (XL-roots) would go in the same fashion as *sak*, so let us only repeat a lexicalization table for the lowermost three cases:

(145) LEXICALIZATION TABLE OF 'TONGUE', NOM–GEN

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	jazyk										
ACC	jazyk										
GEN	jazyk			a							

At first, the root extends as much as possible, which means to K2. At K3, it hits a wall and cannot spell the K3 feature out, so it starts to backtrack. When it shrinks to Cl1, it will finally find a way to spell out by using the *-a* lexical entry. Then it again adds all the features up to K3, resulting in a successful spellout of the genitive *jazyk-a*.

3.2 Proof of concept

In this section, we build on the basis established in the previous section, showing that the same principles can derive the rest of the paradigm and, therefore, cover all relevant data. We add the structurally higher cases, the locative, dative, and instrumental, alongside a new paradigm, *hrad* 'castle'.

3.2.1 Locative and dative

There is one more paradigm in the non-palatalized group for which we have not suggested any lexical items yet. That paradigm is represented by a root *hrad* 'castle'. Below, we repeat the table with the forms of the non-palatalized paradigms. Note the distribution of suffixes in *hrad*:

(146) SURFACE FORMS OF 'BREAD', 'JACKET', 'TONGUE' AND 'CASTLE'

	'bread' (Mi)	'jacket' (N)	'tongue' (Mi)	'castle' (Mi)
NOM	chleb-a	sak-o	jazyk	hrad
ACC	chleb-a	sak-o	jazyk	hrad
GEN	chleb-a	sak-a	jazyk-a	hrad-u
LOC	chleb-u	sak-u	jazyk-u	hrad-u
DAT	chleb-u	sak-u	jazyk-u	hrad-u
INSTR	chleb-em	sak-em	jazyk-em	hrad-em

Similarly to the XL-roots like *jazyk*, *hrad* shows no visible suffix in the nominative and accusative. In fact, the only case where the two paradigms differ is the genitive. Where *jazyk* has *-a*, *hrad* has *-u*.

This is a non-trivial situation. If two paradigms have precisely the same features – as is to be expected of *hrad* and *jazyk* – it is impossible to derive the pattern where the paradigms start the same and then, out of the blue, part. (What is more, these two also unite again in the locative.)

To begin to understand the issue, we repeat the lexicalization tables for the nominative to genitive of *jazyk*, as we suggested in the previous section. We also add how *hrad* might look, based on first impressions:

(147) LEXICALIZATION TABLE OF 'TONGUE', NOM–GEN

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	jazyk										
ACC	jazyk										
GEN	jazyk				a						

LEXICALIZATION TABLE OF 'CASTLE', NOM–GEN

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	hrad										
ACC	hrad										
GEN (var 1)	hrad				u						
GEN (var 2)	hrad		u								

Because the root *hrad* shows no suffix in the nominative and the accusative, it would be fair to assume (based on the logic we have used up to now) that its lexical entry should extend up to K2.

The genitive *-u* of *hrad* has to start higher or lower than *-a*. If it were footed in the same place, the two suffixes would be specified for exactly the same features (Cl2 to K3). To solve such a competition, we would have to reach for some specialized mechanism, something on a par with contextual rules. Such a mechanism is not part of the framework – and would be an unnecessary addition. Given the phrasal spell out, we can instead opt to move the foot of one of the lexical entries higher or lower. That is what we show in the table in (147). Variant 1 of the genitive portrays a situation where *-u* has a foot above the foot of *-a*. Variant 2 shows the opposite scenario.

If variant 1 were true, we would correctly derive the form *hrad-u*. The root would grow to K2. Then it would not be able to spell out K3 – not even through movement – and so backtracking would be triggered. The way backtracking is set up is that it accepts the first lexical entry specified for K3 that it finds. Since *-u* has a higher foot, it means that the root would find *-u* through backtracking, and the form would be correctly spelled out as *hrad-u*.

This scenario would, however, go wrong in the genitive of *jazyk* 'tongue' (and the other paradigms with *-a* in the genitive). The reason is this: The root *jazyk* would grow to the accusative without any issues, but then it would not be able to spell out K3. No movement would help, resulting in backtracking. As described before, the root is obliged to take the first lexical entry specified for K3. That would, of course, just as with *hrad* 'castle', be *-u*. So we would, incorrectly, derive the genitive form *jazyk-u*. This is why variant 1 of *-u* does not work.

Variant 2 shows the opposite problem. With *-u* being lower than *-a*, we would derive the genitive *jazyk-a* correctly, because *-a* is the first lexical entry specified for K3 that the root finds through backtracking. However, this time the genitive of *hrad* would be derived incorrectly, as *hrad-a*.

The heart of the problem is that if two roots grow to the same size and then backtrack over the same features, they will always find the same lexical entry. So specifying them for the same size and features will never allow us to account for their difference in the genitive.

Two apparent options follow from this argumentation. First, perhaps the two paradigms do not have the same features. This is both unlikely and would also ultimately still fail because we would not be able to derive the syncretism between *-u* in the genitive, locative, and dative of *hrad* and the locative and dative of the paradigm *jazyk* 'tongue' (and other non-palatalized paradigms).

The second option is that the two roots do not grow to the same size, and there is some invisible suffix in one of them. While we will discard this option at a later stage, after introducing more complicated trees, let us go with it, for now, to show how the analysis would go if we wanted to stick with simple trees without any complex branching.

For the moment, we call the invisible suffix a "yer" (purely as a placeholder name) because, in the older stages of Czech, the predecessors of both paradigms had a yer at the end:

(148) OLD CZECH, Večerka (2006)

rab-ъ ‘slave, servant’ (predecessor of *jazyk* ‘tongue’)
dom-ъ ‘house’ (predecessor of *hrad* ‘castle’)

So it could have been that at least one of the paradigms underlyingly kept some remnant. We will show that it would have to be *hrad*. However, let us go through the opposite argumentation first to see why it could not have been *jazyk*, given the synchronic data.

If it were *jazyk*, the spellout would look something like this:

(149) MOCK LEXICALIZATION TABLE OF ‘TONGUE’, NOM–GEN, WITH A YER

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	jazyk			ъ							
ACC	jazyk			ъ							
GEN	jazyk			a							

MOCK LEXICALIZATION TABLE OF ‘CASTLE’, NOM–GEN, WITHOUT A YER

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	hrad										
ACC	hrad										
GEN	hrad				u						

The main difference from the setups discussed above is that *jazyk* no longer grows to the accusative. Instead, it only grows to Cl1, and then the sequence gets finished by the yer. Since the yer is not specified for K3, it gets overridden by *-a* in the genitive.

The foot of *-u* in *hrad* has to be above the foot of *-a* in *jazyk*. That way, we get the correct distribution of the two genitive suffixes. *Jazyk* now cannot reach *-u* because Cl2 would not be spelled out. On the other hand, *hrad* cannot get *-a* because after the root grows to the accusative and starts to backtrack for the genitive, *-u* will be the first suffix that it encounters. If the paradigms only consisted of the three cases above, this account would work.

The problem arises when we get to the locative. As has been pointed out before, all the non-palatalized paradigms have *-u* in the locative and dative (refer back to (146)). If we followed the idea that this *-u* is the same *-u* as in the genitive of *hrad*, we would get derivations like this:

(150) MOCK LEXICALIZATION TABLE OF 'TONGUE', NOM–LOC, WITH A YER

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	jazyk			ъ							
ACC	jazyk			ъ							
GEN	jazyk			a							
LOC (var 1)	jazyk				u						
LOC (var 2)	jazyk				u						

MOCK LEXICALIZATION TABLE OF 'CASTLE', NOM–LOC, WITHOUT A YER

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	hrad										
ACC	hrad										
GEN	hrad				u						
LOC	hrad				u						

The suffix *-u* starts at Sg as before, but we let it grow to K4 (and, in fact, to K5 in the dative, to which we return later). This works well for *hrad*. It can use *-u* in both the genitive and locative thanks to the Superset Principle.

It is *jazyk* that gets this analysis into trouble. Its locative can now be interpreted in two ways. In variant 1, the Cl2 would be left without spell out, which means the derivation would crash. Variant 2, on the other hand, suggests that the root *jazyk* is bigger than we thought. It grows to Cl2 and takes on *-u* in the locative all right. The problem then is with the previous (structurally lower) cases. If *jazyk can* grow to Cl2, it *has to* grow to Cl2 (as defined by the first step of the spell out algorithm). If it grows to Cl2, it will never take the yer in the nominative and accusative and *-a* in the genitive because their feet are too low. In fact, it will be spelled out by *-u* in all three of these cases, using the Superset Principle. So, as advertised before, setting up *jazyk* for the yer results in a failed derivation.

On the other hand, assigning the yer to *hrad* works well. Consider the lexicalization tables below:

(151) LEXICALIZATION TABLE OF 'TONGUE', NOM–LOC, WITHOUT A YER

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	jazyk										
ACC	jazyk										
GEN	jazyk			a							
LOC	jazyk		u								

LEXICALIZATION TABLE OF 'CASTLE', NOM–LOC, WITH A YER

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	hrad		ъ								
ACC	hrad		ъ								
GEN	hrad		u								
LOC	hrad		u								

The root *jazyk* grows to the accusative. Then, it cannot spell out K3, so it starts to backtrack. The way the lexical items are set up now, the first suffix able to spell out K3 that it finds is, correctly, *-a*. In the locative, it backtracks even lower to spell out K4 and correctly finds *-u*.

The root *hrad* grows only to GendP. The suffix that can finish up the sequence in the nominative and accusative is the yer. The yer is, however, not able to spell out K3 (and K4), and so *-u* picks it up, correctly giving us the form *hrad-u*.

Therefore, we conclude that if one of the two paradigms had an invisible suffix, it would have to be *hrad*. The above argumentation again underscores the logic that has already been presented in the previous sections. If two paradigms show syncretism and one of the paradigms has a particular suffix later, this paradigm must necessarily be of bigger root size. We can see it above: *jazyk* shows *-u* only in the locative, while *hrad* has it in the genitive already. And we have indeed seen that *jazyk* has to be bigger.

In the lexicalization tables for *hrad* above is one part of the theory assumed that we have not mentioned yet. Note that through the Superset Principle, *-u* would be perfectly able to spell out also the nominative and genitive of *hrad* (pushing the yer out of the picture). The reason why that does not happen is the so-called Elsewhere Condition:

(152) THE ELSEWHERE CONDITION, Kiparsky (1973), cited from Caha (2021c)

> When two entries can spell out a given node, the more specific entry wins. The more specific entry is the one that has fewer features.

Under this consideration, the yer wins both in the nominative and accusative. In the accusative, it is a perfect match, i.e., it does not have any superfluous features. In the nominative, the lexical entry for the yer has only one unused feature (K2). That means it will win over *-u*, which is specified for features up to K4 in the table above and hence would have three extra features in the nominative. On the other hand, the yer cannot be used instead of *-u* because it only reaches to K2 and, as such, misses the K3 and K4 features.

As the last step now, let us extend the account to all the instances of *-u* in the non-palatalized paradigms. All of them have *-u* in the locative and the dative. The lexicalization tables below show both paradigms with all the cases up to the dative:

(153) LEXICALIZATION TABLE OF 'TONGUE', NOM–DAT

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	jazyk										
ACC	jazyk										
GEN	jazyk			a							
LOC	jazyk		u								
DAT	jazyk		u								

LEXICALIZATION TABLE OF 'CASTLE', NOM–DAT

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	hrad		ъ								
ACC	hrad		ъ								
GEN	hrad		u								
LOC	hrad		u								
DAT	hrad		u								

We have extensively discussed the derivations of the roots *jazyk* and *hrad*. All that changed in these tables is that *-u* is now also specified for K5 in the dative.

Now, look at the lexicalization tables for the two other paradigms:

(154) LEXICALIZATION TABLE OF 'BREAD', NOM–DAT

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	chleb			a							
ACC	chleb			a							
GEN	chleb			a							
LOC	chleb		u								
DAT	chleb		u								

LEXICALIZATION TABLE OF 'JACKET', NOM–DAT

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	sak				o						
ACC	sak				o						
GEN	sak			a							
LOC	sak		u								
DAT	sak		u								

The first three cases are the same that we saw in the previous section. We only added *-u* in the locative and dative. *Chleb* grows to Cl1 and then gets finished by *-a* in the nominative to genitive. However, the suffix *-a* cannot spell out any higher features (K4 and K5 here), so backtracking is triggered. The first suffix able to spell out K4 (and then K5) that *chleb* finds is *-u*, hence correctly deriving the form *hrad-u* for both cases.

The root *sak* grows higher than *chleb*, to Cl2. In the nominative and accusative, it gets finished by *-o*. The suffix *-o* cannot spell out any higher features, though, and so in the genitive, backtracking gets triggered. The first suffix that the root find is, correctly,

-a. The same process repeats for the locative and dative, where *sak* backtracks still one feature lower and ends up spelling the two cases with the form *sak-u*.

Note that all the differences in the derivations (and hence in the suffixes) boil down to two mechanisms: the different root size and its interaction with the spellout algorithm. The root that grows higher will "encounter" suffixes in a slightly different way than the smaller root.

3.2.2 Instrumental

The instrumental is the last case needed for a complete account of the four non-palatalized paradigms. As might be already clear from the surface forms, it is a straightforward case to incorporate:

(155) SURFACE FORMS OF 'BREAD', 'JACKET', 'TONGUE', 'CASTLE', INSTR

	'bread'	'jacket'	'tongue'	'castle'
INSTR	chleb-em	sak-em	jazyk-em	hrad-em

All the non-palatalized paradigms we treat here share a single suffix *-em*.

In the dative, all the roots backtrack to GendP to take *-u*. The instrumental is built on top of the dative, which means that the foot of the instrumental lexical entry has to start in Cl1 (or lower) since the spellout algorithm does not allow roots to regrow once they shrank through backtracking. So the most straightforward way (which will be amended later, once we look at the palatalized paradigms) is to let the instrumental suffix start at Cl1 (like *-u* does in the dative) and spell out all the features up to K6. Below, we see an example of that for the XL-sized paradigm *jazyk* 'tongue':

(156) LEXICALIZATION TABLE OF 'TONGUE'

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	jazyk										
ACC	jazyk										
GEN	jazyk			a							
LOC	jazyk		u								
DAT	jazyk		u								
INSTR	jazyk		em								

After the syntax merges K6 to create the instrumental, the dative *-u* is no longer applicable and is substituted by perfectly matching *-em*. The process will be the same for all the other non-palatalized paradigms as well.

Remember that while *-em* would also be able to spell out the locative and dative through the Superset Principle, it will not do so. The reason is that the Elsewhere Condition picks the most specific lexical entry – and that will always be *-u* in the dative.

This section has looked at four non-palatalized paradigms in Czech and suggested their lexical entries. The only tool we needed was differing root size – a natural consequence

of phrasal spellout, which allows us to account both for the allomorphy and the intra- and inter-paradigm syncretisms. Before we turn to the other big group, the palatalized paradigms, let us go step by step through the exact derivations for the XL-sized and M-sized group.

3.3 Full derivations

This subsection shows detailed derivations for two of the paradigms discussed above. Instead of presenting new data or tools, it aims to make sure that the algorithmic nature of the spellout-driven movement is well-understood. It needs to be highlighted again that these are all standard tools necessary beyond the task at hand.

3.3.1 The XL-sized group derivation

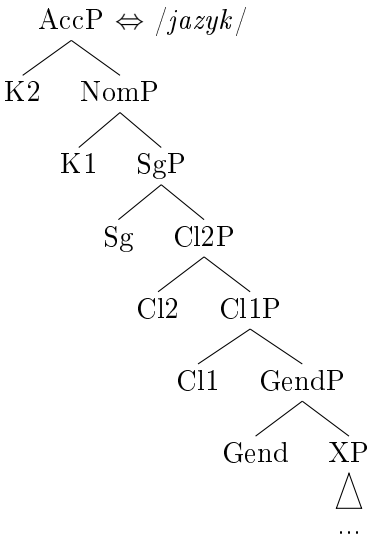
Let us start from the XL-sized group, represented by the root *jazyk* 'tongue' for the non-palatalized paradigms. In (157), there is a table with the surface forms:

(157) SURFACE FORMS OF 'TONGUE' (XL-SIZED)

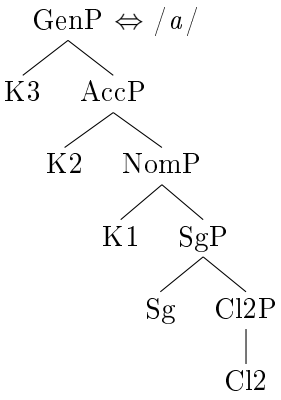
	'tongue'
NOM	jazyk
ACC	jazyk
GEN	jazyk-a
LOC	jazyk-u
DAT	jazyk-u
INSTR	jazyk-em

Here are the relevant lexical entries:

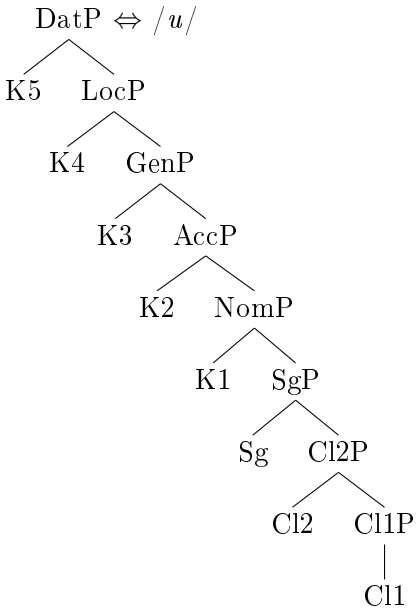
(158) LEXICAL ENTRY OF 'TONGUE'



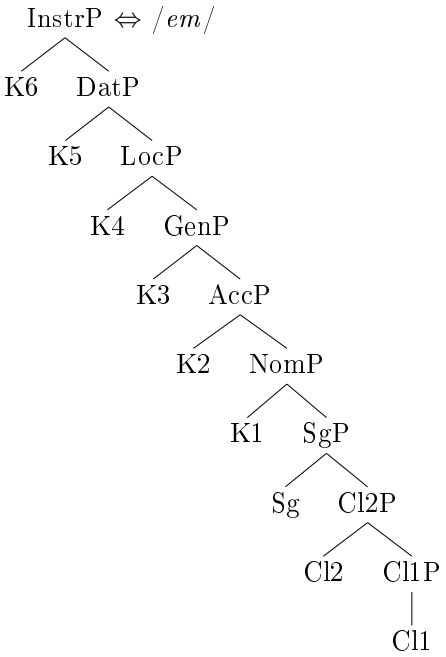
(159) LEXICAL ENTRY OF -A



(160) LEXICAL ENTRY OF -U



(161) LEXICAL ENTRY OF -EM



To remind ourselves, here are the spellout algorithm and the backtracking explanations, repeated:

(162) THE SPELLOUT ALGORITHM, Caha (2021c), based on Starke (2018)

a. Merge F and spell out.

b. If (a) fails, move the Spec of the complement and spell out.

c. If (b) fails, move the complement of F and spell out.

(163) BACKTRACKING, Caha (2021c), based on Starke (2018)

When spellout fails, go back to the previous cycle, and try the next option for that cycle.

The general direction of this derivation will be that the root grows up to K2 and then backtracks in the genitive, locative and instrumental.

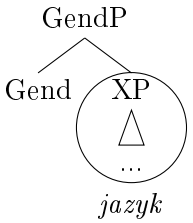
Now onto the derivation. First, XP is created (164). As before, we stay away from its content because it is not relevant for us here. The lexical entry of the root contains the XP node (see (158)), so XP is spelled out as *jazyk* in (165):

(164) DERIVATION OF 'TONGUE', NOM, INTERMEDIATE STEP

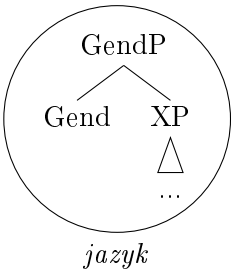
(165) DERIVATION OF 'TONGUE', NOM, INTERMEDIATE STEP

Then, in (166), we merge the next feature up, which is Gend. The new GendP node can also be spelled out by the root, using a bigger subset of the same lexical entry as in the step above. The derivation results in (167):

(166) DERIVATION OF 'TONGUE', ACC, INTERMEDIATE STEP

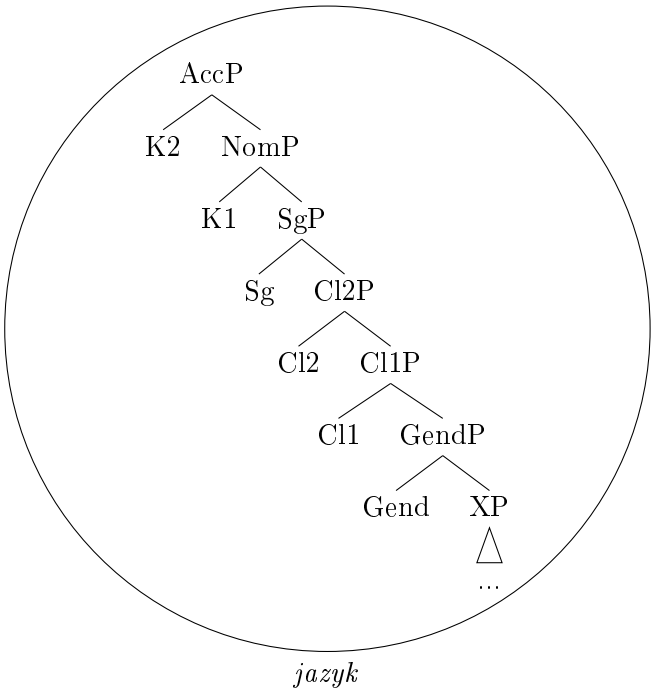


(167) DERIVATION OF 'TONGUE', ACC, INTERMEDIATE STEP



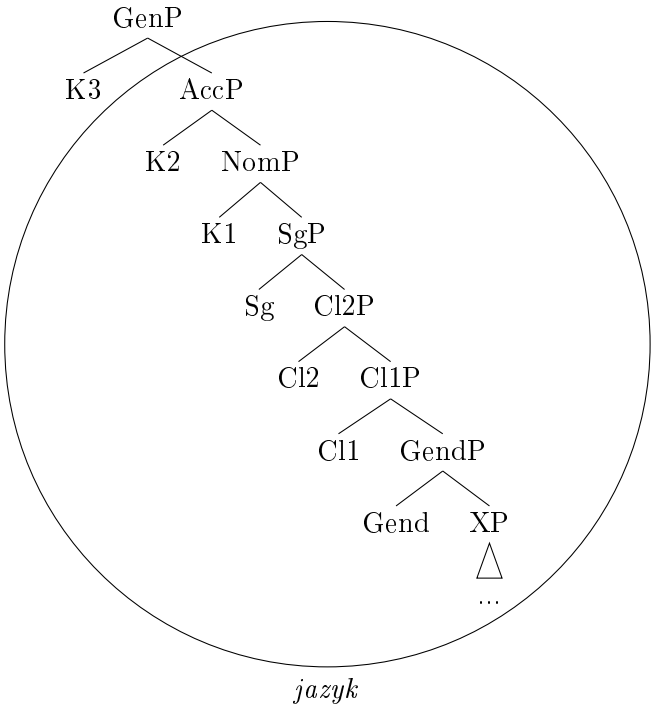
The derivation proceeds in the same way a bit longer – merging a feature after feature and spelling it all out using the lexical entry for the root, until we merge K2 and one more time spell the structure out as *jazyk*:

(168) DERIVATION OF 'TONGUE', ACC, FINAL FORM



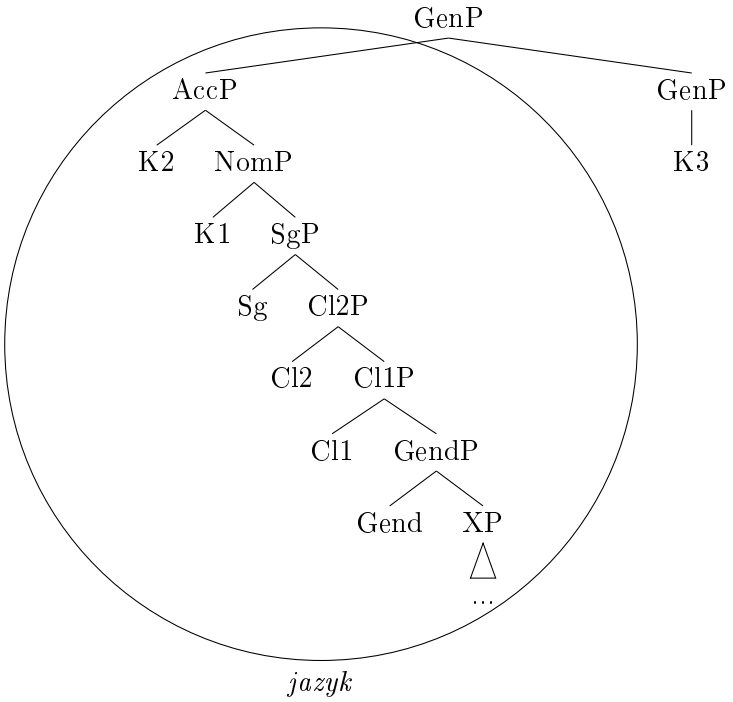
Then we merge K3 for the genitive:

(169) DERIVATION OF 'TONGUE', GEN, INTERMEDIATE STEP



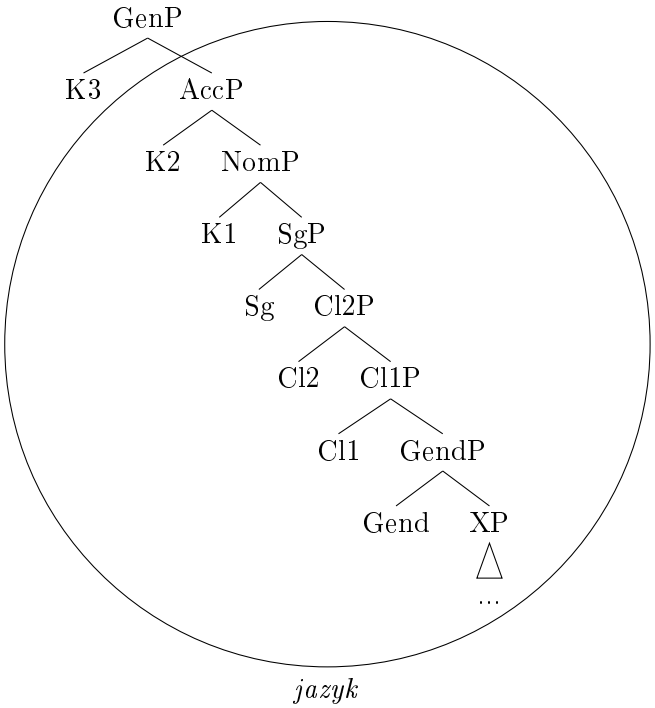
The new structure can no longer be spelled out by the lexical entry of the root because the root does not contain K3. That is when the next step of the spellout algorithm gets triggered. The second step means cyclically moving the phrase in the Spec of the complement of K3. Since there is no phrase in that position, this step does not apply, and we skip to the next step. That is moving the whole complement of K3 up:

(170) DERIVATION OF 'TONGUE', GEN, INTERMEDIATE STEP



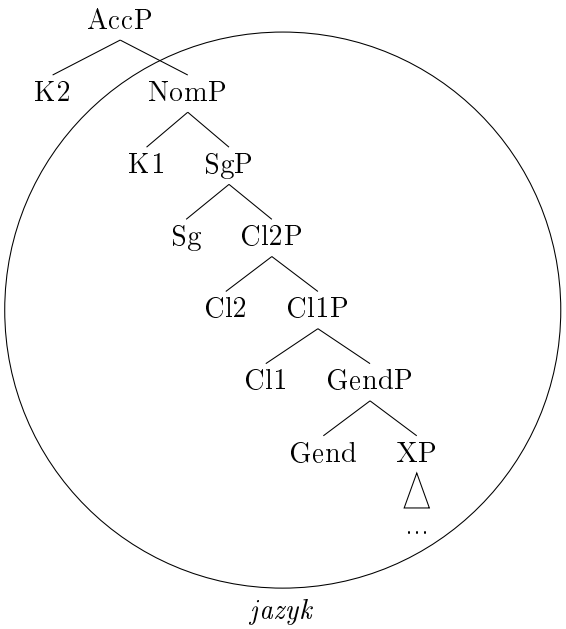
The complement movement does not help since there is no lexical entry in our lexicon that would be able to spell out just the leftover K3. So we undo the last movement, coming back to the structure below:

(171) DERIVATION OF 'TONGUE', GEN, INTERMEDIATE STEP



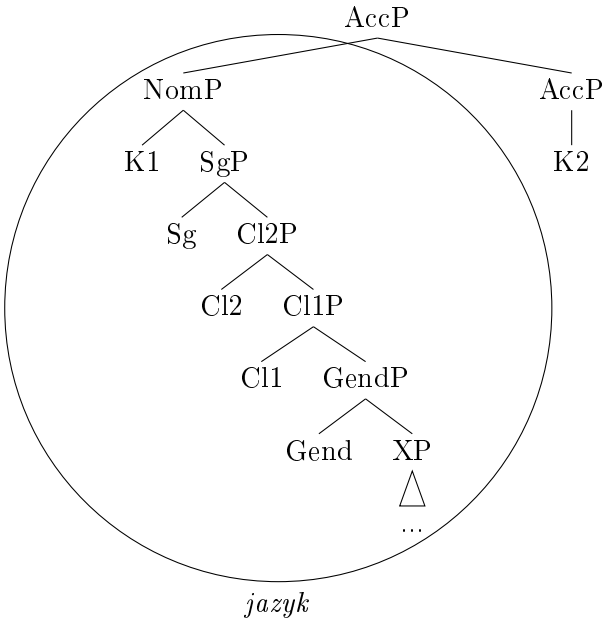
Since neither of the two spellout-driven movements helped, the next option of the algorithm is triggered, which is backtracking. The feature K3 gets deleted, and the previous derivational stage is restored. The previous stage is the moment after merging K2, but prior to its spell out:

(172) DERIVATION OF 'TONGUE', GEN, INTERMEDIATE STEP, BACKTRACKING



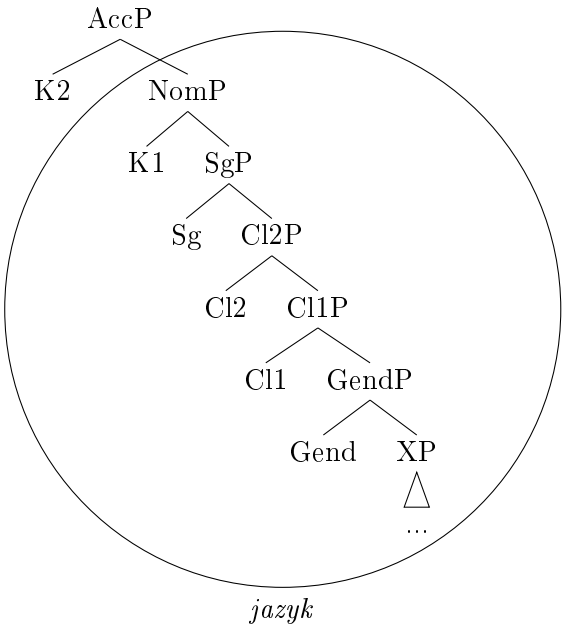
The next step after backtracking is always trying the following option from the spellout algorithm that was not tried at that stage. What we have done at this stage before was merging K2 and spelling out the whole structure without any movement. So the next option is to move the phrase in the Spec of the complement of K2. This step is again not applicable because there is no phrase in that position. So we proceed to the next step, which is moving the whole complement of K2:

(173) DERIVATION OF 'TONGUE', GEN, INTERMEDIATE STEP, BACKTRACKING



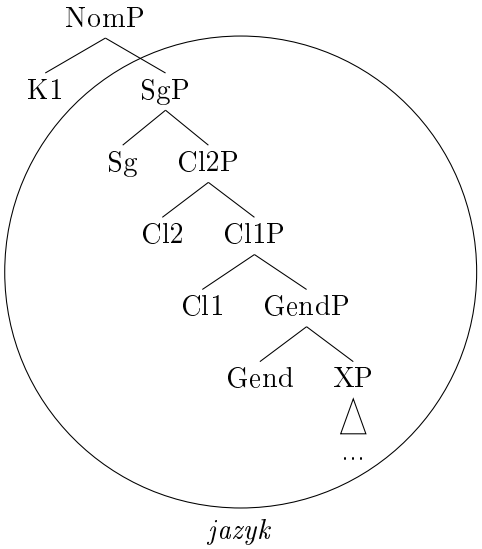
An attempt to find a lexical entry to match against the right side AccP fails again. No lexical entry can spell out just K2. Hence, we undo the last movement, coming back to having K2 on top as the only unspelled out feature:

(174) DERIVATION OF 'TONGUE', GEN, INTERMEDIATE STEP, BACKTRACKING



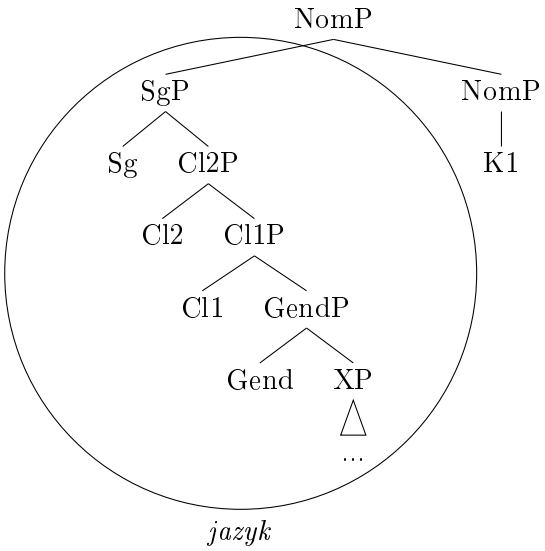
From here, the same process repeats. Since none of the movements worked, the structure backtracks to just after merging K1:

(175) DERIVATION OF 'TONGUE', GEN, INTERMEDIATE STEP, BACKTRACKING



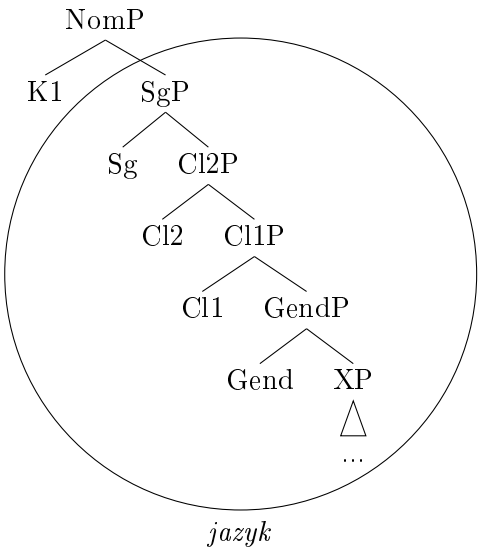
The Spec movement is again not applicable. The complement movement (shown below) does not help because there is no lexical entry able to spell out structure starting from K1 while extending to K3:

(176) DERIVATION OF 'TONGUE', GEN, INTERMEDIATE STEP, BACKTRACKING



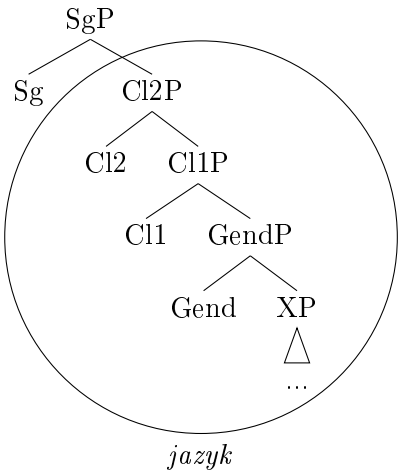
The movement gets undone, and the structure returns to the previous stage:

(177) DERIVATION OF 'TONGUE', GEN, INTERMEDIATE STEP, BACKTRACKING



Once again, because no movement helped, the structure backtracks one step down:

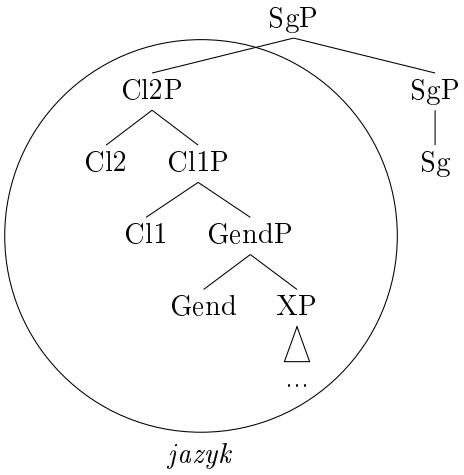
(178) DERIVATION OF 'TONGUE', GEN, INTERMEDIATE STEP, BACKTRACKING



Just as before, the Spec movement cannot be done. The complement movement (below) will not help because there is no lexical entry starting at Sg and extending to K3:[20]

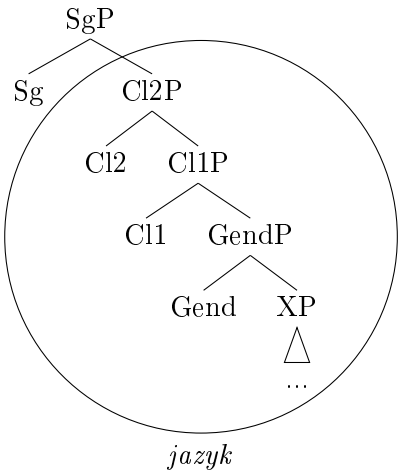
[20] Technically speaking, the derivation is slightly more complex than we show here. From the previous sections, we know that there is a lexical entry starting at the Sg feature: *-o* (see the entire lexicon in the appendix in 8.3). This entry can spell out the feature Sg on its own. That kick-starts a new derivation path where the left side is spelled out as *jazyk*, while the right as *-o*. After merging every new feature (K1 and K2, respectively) and the Spec cyclically moves above them both, the left side is still spelled out by the root, while the right side still matches the lexical extry for *-o*. The moment we merge K3, though, and try to spell it out, it fails. The entry for *-o* cannot extend that far, and there is no other lexical entry starting at Sg. This triggers yet another round of backtracking, bringing the

(179) DERIVATION OF 'TONGUE', GEN, INTERMEDIATE STEP, BACKTRACKING



That means that the movement needs to be undone:

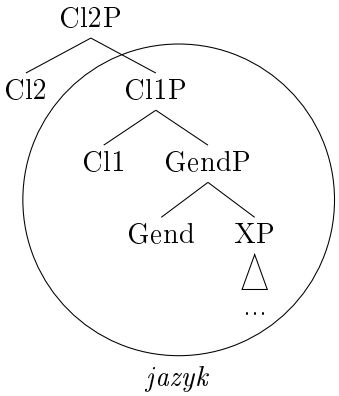
(180) DERIVATION OF 'TONGUE', GEN, INTERMEDIATE STEP, BACKTRACKING



One more backtracking is triggered, resulting in the structure below:

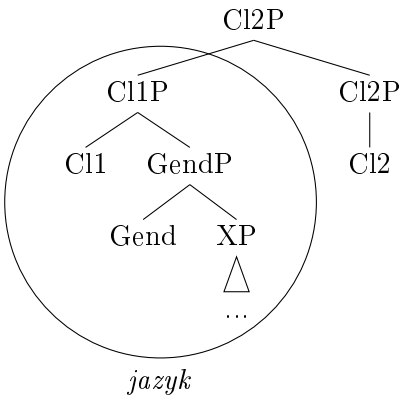
structure to the same point as in (178) above. The rest of the derivation continues the same way as described in the main text.

(181) DERIVATION OF 'TONGUE', GEN, INTERMEDIATE STEP, BACKTRACKING



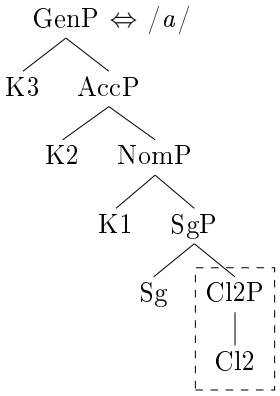
Even here, the Spec movement is not applicable. The complement movement gives us the following structure:

(182) DERIVATION OF 'TONGUE', GEN, INTERMEDIATE STEP, END OF BACKTRACKING

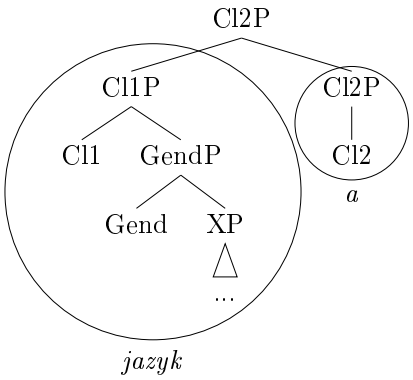


At this moment, we have finally found a structure that does not fail. The right side starts with Cl2, which can be spelled out by the lexical entry for *-a* (repeated in (183)). The dashed rectangle marks the used subset. In (184), we see the fully spelled out structure:

(183) LEXICAL ENTRY OF -A, CL2P SUBSET

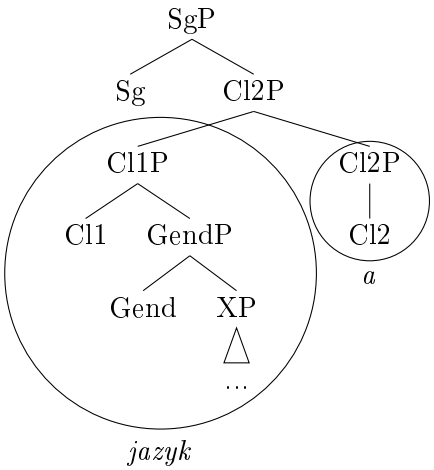


(184) DERIVATION OF 'TONGUE', GEN, INTERMEDIATE STEP



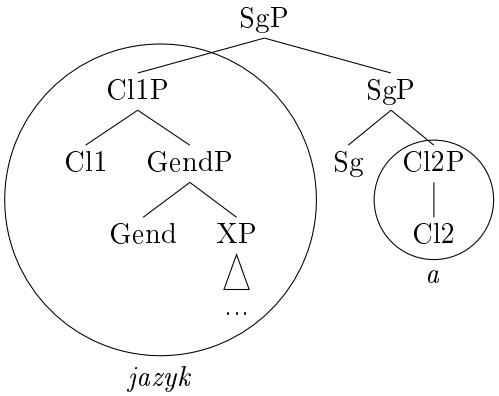
This is, of course, not the end of the derivation because we still need to merge all the features up to K3, which is the genitive specification. So in the next step, we merge Sg:

(185) DERIVATION OF 'TONGUE', GEN, INTERMEDIATE STEP



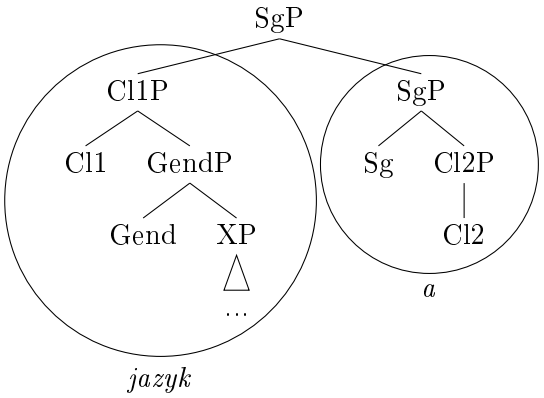
This complex structure cannot be spelled out as it is, so the Spec movement gets triggered, resulting in the structure below:

(186) DERIVATION OF 'TONGUE', GEN, INTERMEDIATE STEP



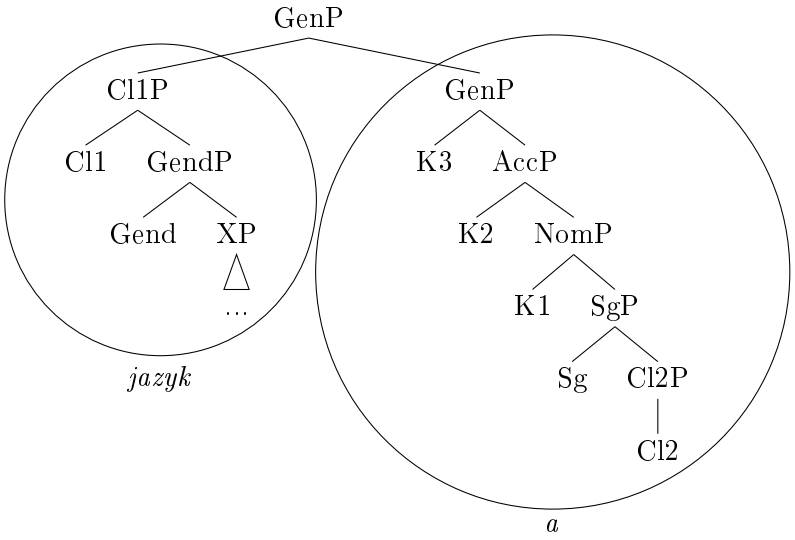
Now that the root is out of the way, the SgP on the right-hand side can be spelled out by the same *-a* we used to spell out Cl2P in the previous step (using a bigger subset now):

(187) DERIVATION OF 'TONGUE', GEN, INTERMEDIATE STEP



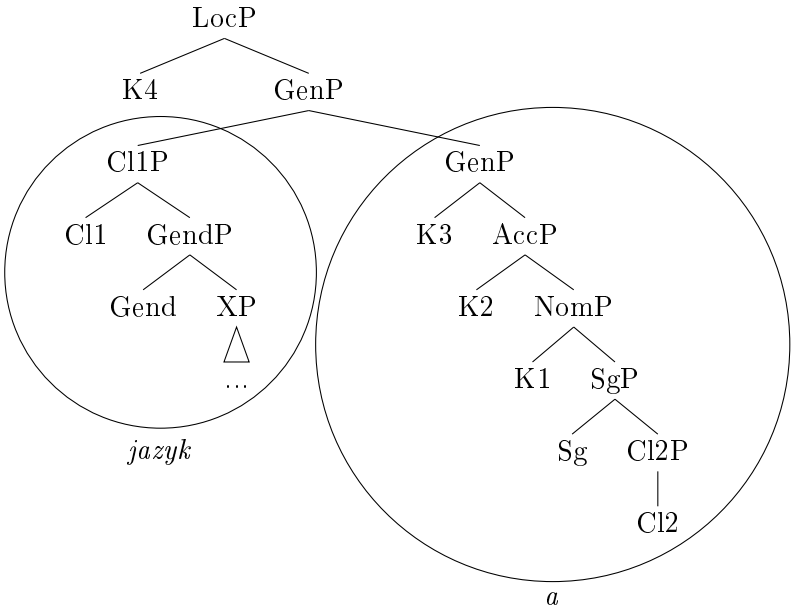
The same happens with merging all the remaining features, K1, K2, and K3. The root in the Spec will always move above the new feature, freeing the structure to spell out the right side by *-a*. The final result, the complete genitive structure, is shown below:

(188) DERIVATION OF 'TONGUE', GEN, FINAL FORM



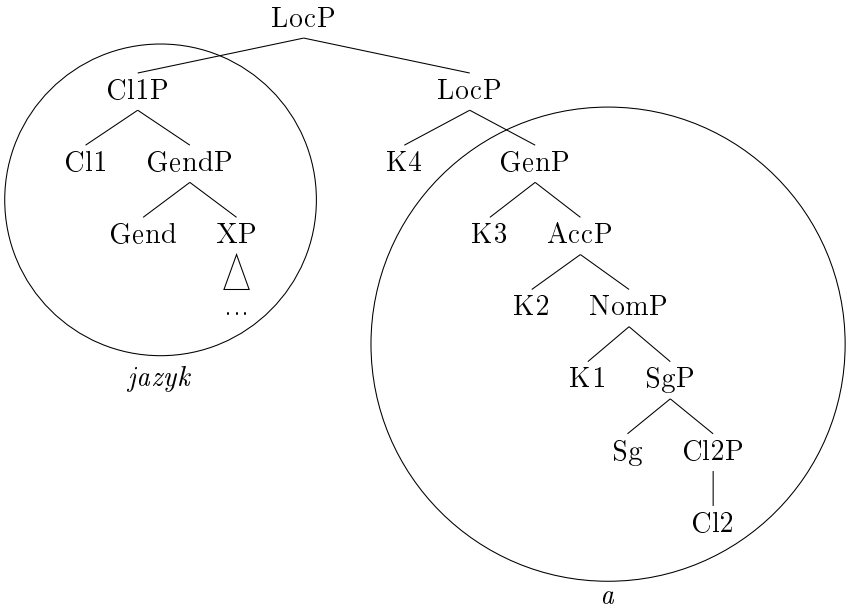
The locative is the feature K4 on top of the genitive, so after getting to the genitive, we merge K4:

(189) DERIVATION OF 'TONGUE', LOC, INTERMEDIATE STEP



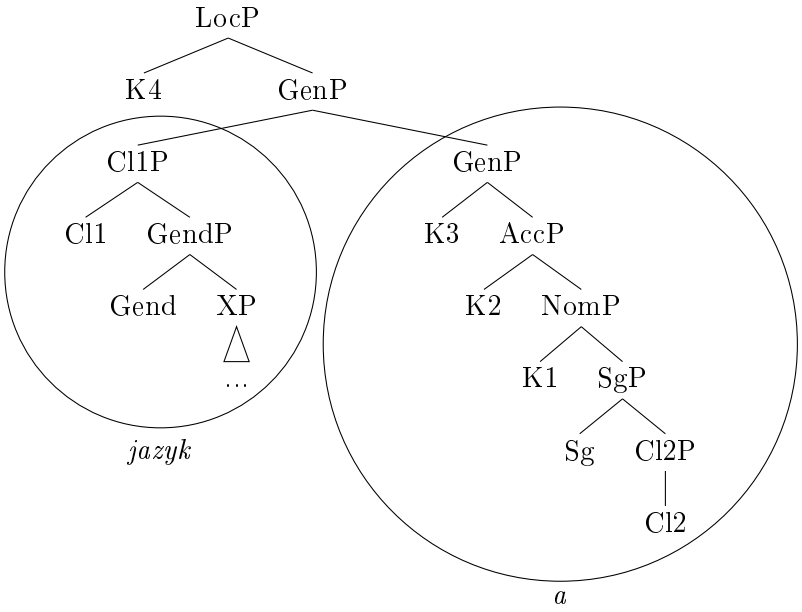
The structure cannot be spelled out as it is, so the Spec movement gets triggered:

(190) DERIVATION OF 'TONGUE', LOC, INTERMEDIATE STEP



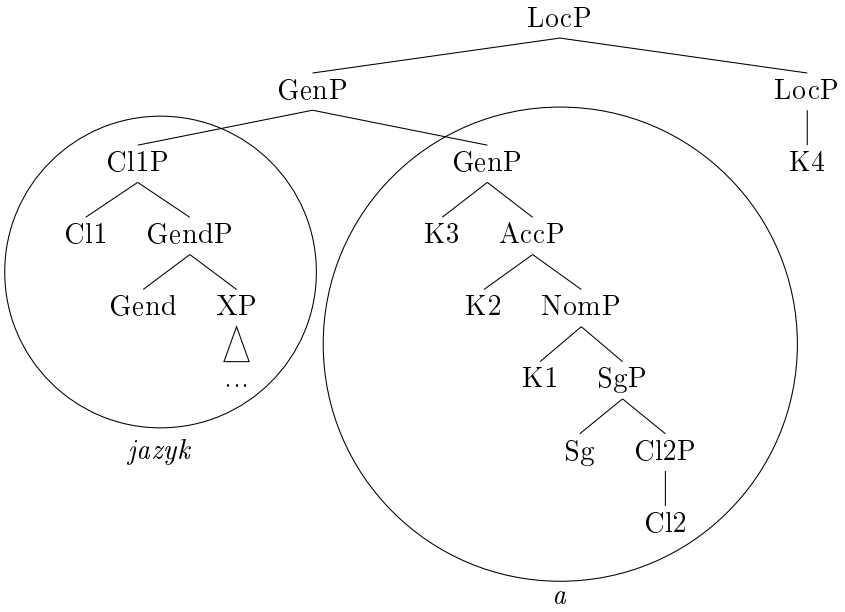
The structure attempts to spell out the whole lower LocP node but fails because there is no lexical entry starting at Cl2 and extending to LocP. So the unsuccessful movement has to be undone:

(191) DERIVATION OF 'TONGUE', LOC, INTERMEDIATE STEP



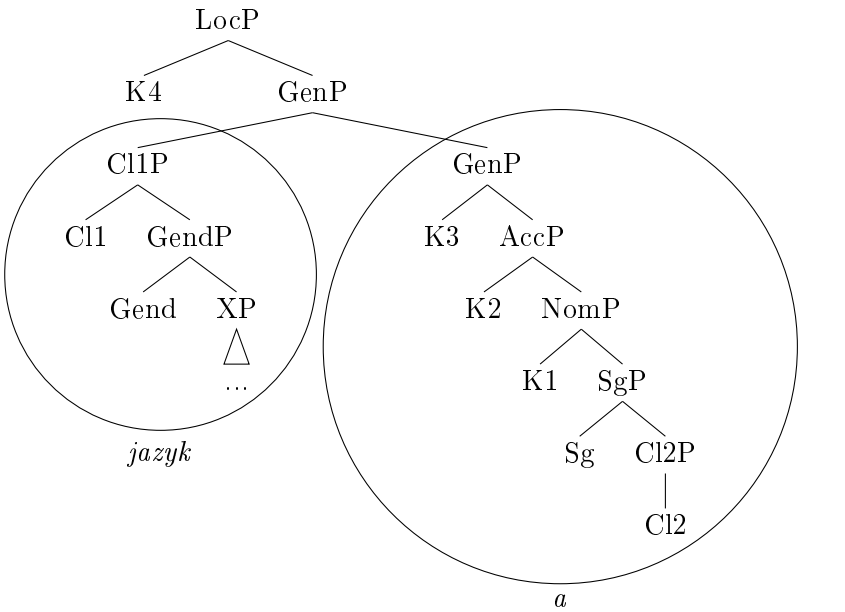
Next in the spellout algorithm is the complement movement. We try it, moving the whole upper GenP. Our attempt results in the following structure:

(192) DERIVATION OF 'TONGUE', LOC, INTERMEDIATE STEP



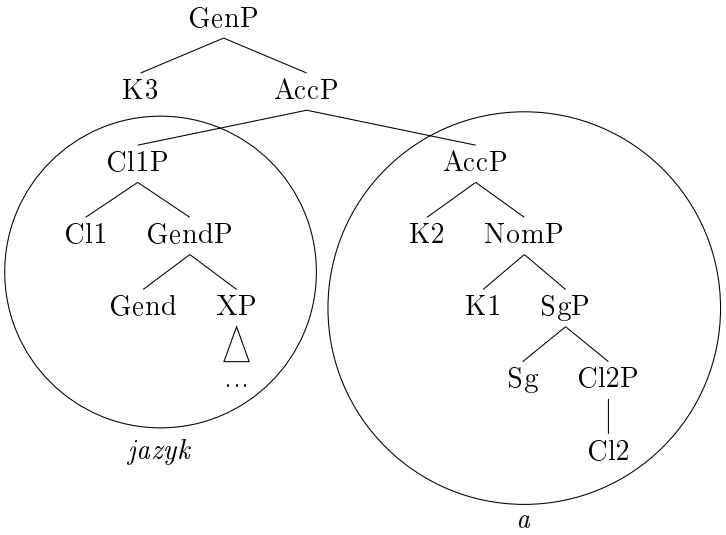
If there were a lexical entry starting with K4, we would get a three-morphemic spellout. However, there is no lexical entry able to spell out just LocP, so we undo this movement:

(193) DERIVATION OF 'TONGUE', LOC, INTERMEDIATE STEP



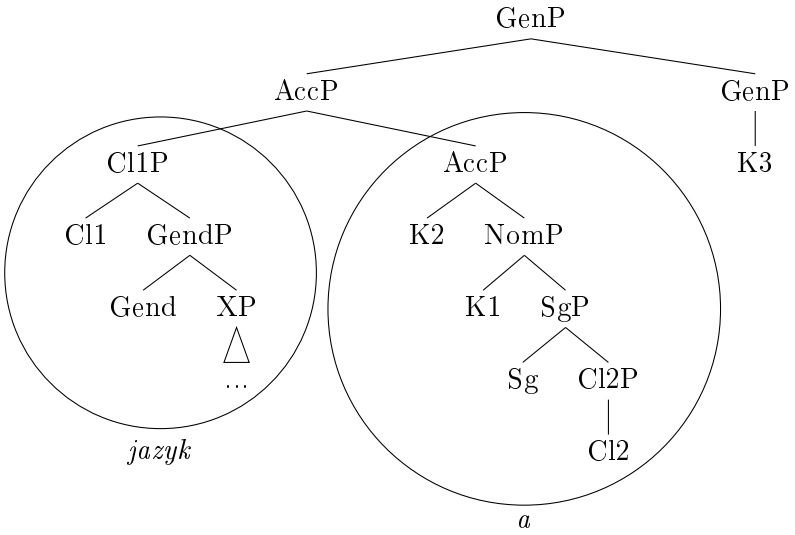
Since neither of the two movements delivered a successful structure, backtracking gets triggered. In this case, backtracking might seem more complex, but it is still the same process – get to the stage just after merging the previous feature (K3 here), but prior to spelling it out:

(194) DERIVATION OF 'TONGUE', LOC, INTERMEDIATE STEP, BACKTRACKING



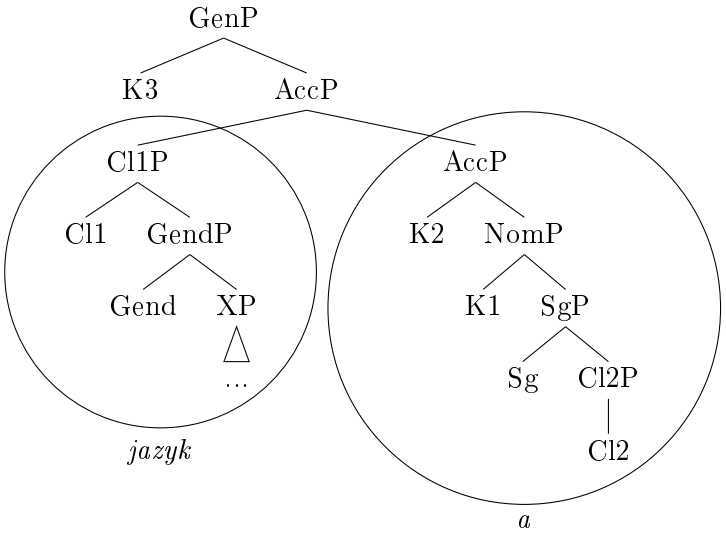
Following backtracking, we do the next step of the spellout algorithm that has not been tried before. Previously, we have moved the Spec here, which brought us to a dead end. So now, we try the complement movement:

(195) DERIVATION OF 'TONGUE', LOC, INTERMEDIATE STEP, BACKTRACKING



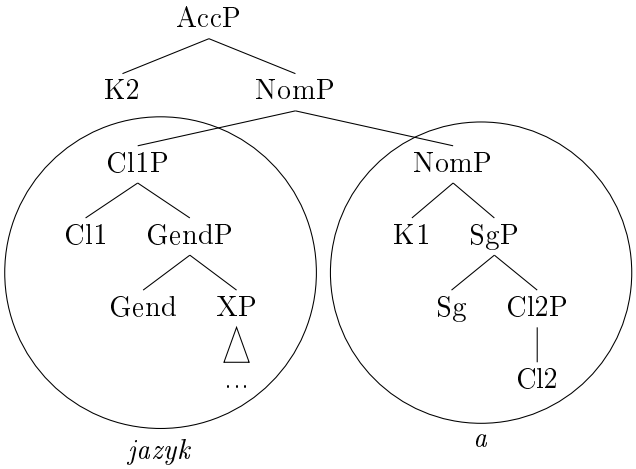
The complement movement does not help, as there is no lexical entry able to spell out the leftover GenP. So the movement gets undone back to the structure below:

(196) DERIVATION OF 'TONGUE', LOC, INTERMEDIATE STEP, BACKTRACKING



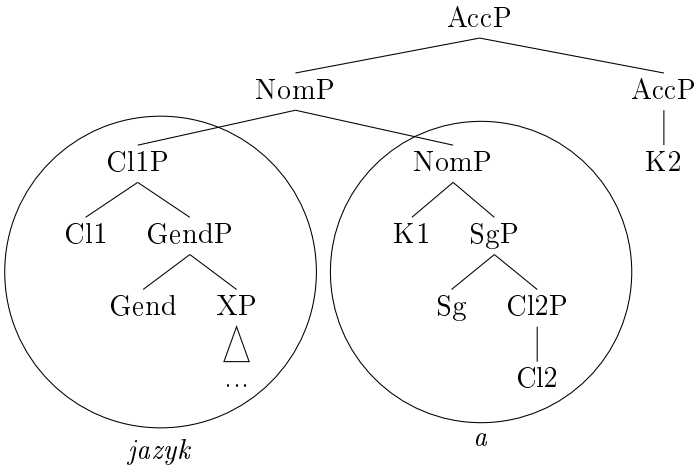
Neither of the two movements worked, so further backtracking is triggered, getting us back to the K2 stage:

(197) DERIVATION OF 'TONGUE', LOC, INTERMEDIATE STEP, BACKTRACKING



We have tried the Spec movement here before, so we go straight to the complement movement:

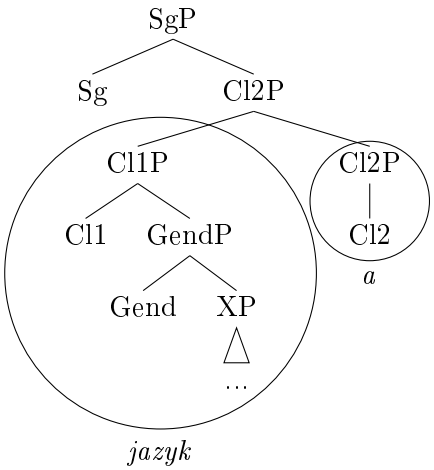
(198) DERIVATION OF 'TONGUE', LOC, INTERMEDIATE STEP, BACKTRACKING



This does not help either because there is no lexical entry able to spell out just AccP. This will go on a bit further, step-by-step backtracking and always trying the complement movement (since the Spec movement has already been tried in the previous stages of the derivation). Notice that what is happening, as we backtrack, is that the structure spelled out by *-a* is getting smaller.

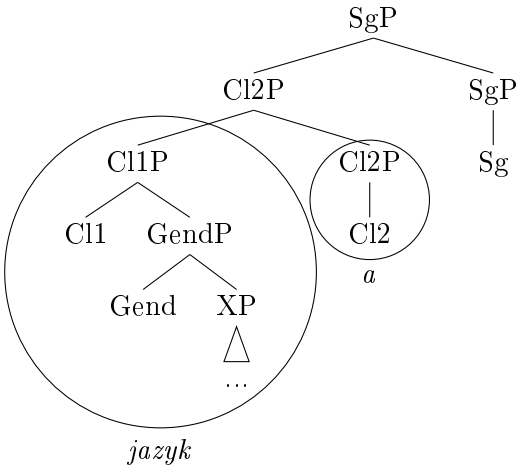
Let us cut to the place just before things become more exciting again. Below, we can see the structure that backtracked to having only Cl2P spelled out by the lexical entry for *-a*. We are attempting to spell out the SgP node on the top of the structure:

(199) DERIVATION OF 'TONGUE', LOC, INTERMEDIATE STEP, BACKTRACKING



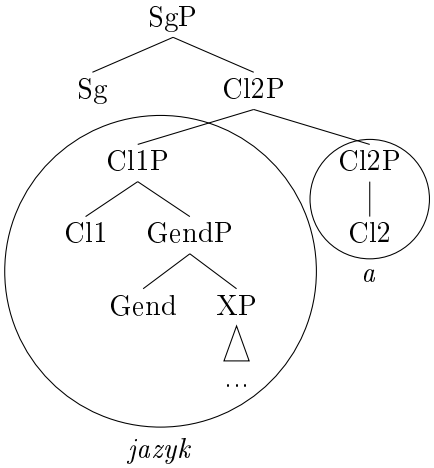
When we were at this stage before, we tried the Spec movement, which led us astray. So we try the complement movement:

(200) DERIVATION OF 'TONGUE', LOC, INTERMEDIATE STEP, BACKTRACKING



The complement movement does not help because no lexical entry can spell out just Sg while extending to K4.[21] So we undo the movement:

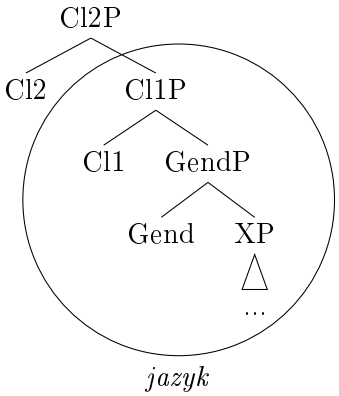
(201) DERIVATION OF 'TONGUE', LOC, INTERMEDIATE STEP, BACKTRACKING



We backtrack one step more, opening the Cl2P to a new spell out option:

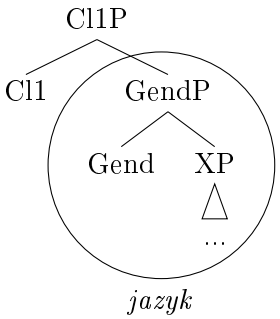
[21] The same comment as in the footnote 20 applies here. The derivation is more complex, pursuing the path to spell Sg as *-o* first. Then it hits a dead end because it cannot spell out K4. At that moment, it backtracks some more, coming back to the Sg stage again. It backtracks below Cl2, just as we are about to show in the simplified version.

(202) DERIVATION OF 'TONGUE', LOC, INTERMEDIATE STEP, BACKTRACKING



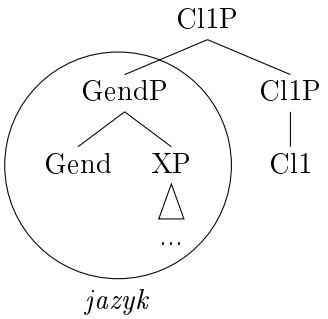
Note that we disrupted the right-hand side, previously spelled out by *-a*. Only the part spelled out by the root is kept. When we were at this stage before, we moved the complement of Cl2 up. Since that did not work out and there is no other movement to be tried, more backtracking gets triggered:

(203) DERIVATION OF 'TONGUE', LOC, INTERMEDIATE STEP, BACKTRACKING



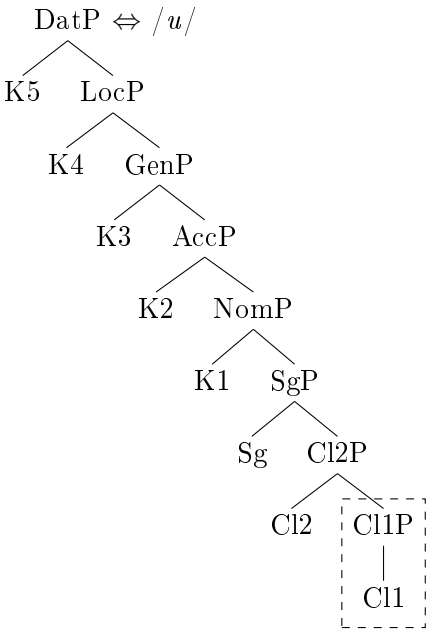
Previously, we have extended the spell out to the Cl1 here without moving anything. That failed later, so the next option is to do the Spec movement. The Spec movement is not applicable because there is no phrase in the Spec of the complement of Cl1. So the next attempt is the complement movement. That results in the following structure:

(204) DERIVATION OF 'TONGUE', LOC, INTERMEDIATE STEP, END OF BACKTRACKING

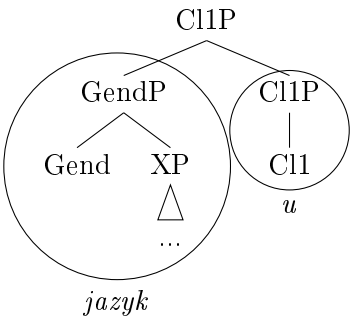


At last, there is a lexical entry able to spell out only the right-hand side (Cl1P), and that is -*u* (in (205)). The dashed line marks the relevant subset. The resulting structure is in (206):

(205) LEXICAL ENTRY OF -U



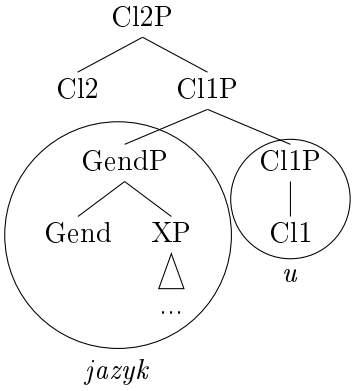
(206) DERIVATION OF 'TONGUE', LOC, INTERMEDIATE STEP



This has opened a new derivation path that will lead us to derive the locative form.

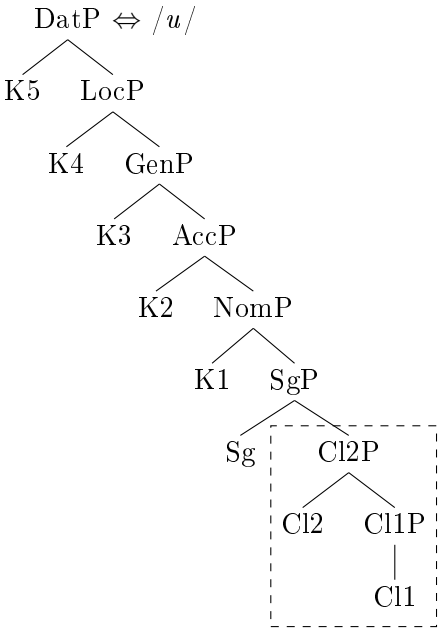
Because everything in the structure in (206) is successfully spelled out, we proceed to merge a new feature, Cl2P:

(207) DERIVATION OF 'TONGUE', LOC, INTERMEDIATE STEP

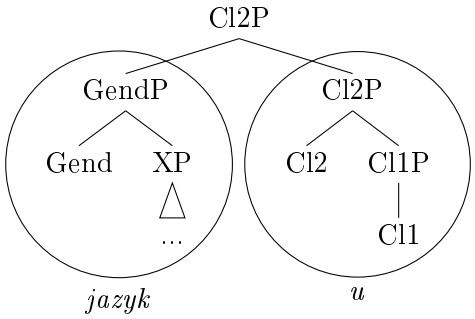


Because no lexical entry can spell out the whole structure, the Spec movement gets triggered. It frees the right-hand side to spell out Cl2P and Cl1P together. In (208), we can see that the lexical entry for -*u* is still able to spell out the right side. In (209), we see the resulting structure:

(208) LEXICAL ENTRY OF -U

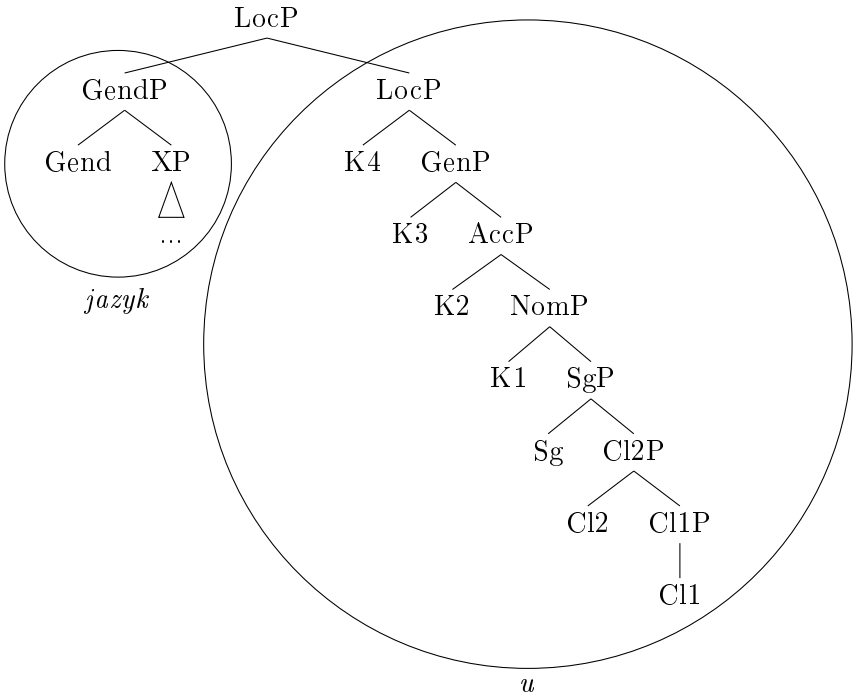


(209) DERIVATION OF 'TONGUE', LOC, INTERMEDIATE STEP



The same process continues. Progressively, we merge Sg, K1, K2, K3, and K4. After each merge, the Spec – in our case, the part spelled out by the root – moves above the new feature. The new features are spelled out together with the other features on the right. Up to the end of the locative derivation, i.e., to K4, the right side can now always be spelled out by the lexical entry for *-u*. The final structure can be seen in (210):

(210) DERIVATION OF 'TONGUE', LOC, FINAL FORM



This is the end of the locative derivation in syntax. Similarly, the dative would be built on top of the locative and instrumental on top of the dative derivation.

3.3.2 The M-sized group derivation

Let us also show a derivation of the M-sized group. The aim is to see how it differs from the XL-sized group. We will point out the places where the two derivations differ.

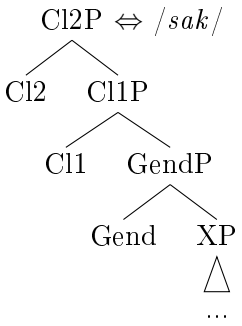
In (211), there are the surface forms of *sak* 'jacket', the root representing the M-sized group:

(211) SURFACE FORMS OF 'JACKET' (M-SIZED)

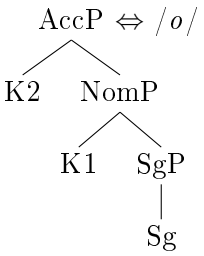
	'jacket'
NOM	sak-o
ACC	sak-o
GEN	sak-a
LOC	sak-u
DAT	sak-u
INSTR	sak-em

We will show two derivations: the nominative and the genitive. Below are the lexical entries needed:

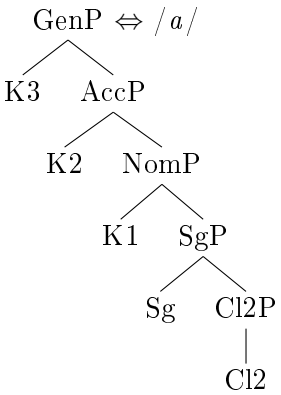
(212) LEXICAL ENTRY OF 'JACKET'



(213) LEXICAL ENTRY OF -O

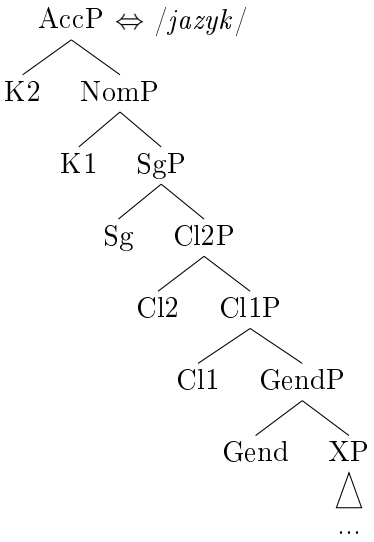


(214) LEXICAL ENTRY OF -A

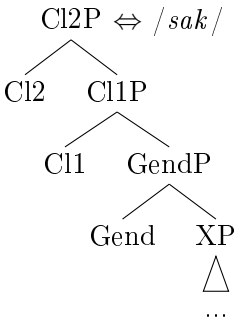


The way this derivation proceeds is that everything up to Cl2 will be spelled out by the root at first, while Sg to K2 will be finished by the suffixal *-o*. In the XL-sized group, the root grew all the way to K2 in the beginning. That is the one difference from which all the rest follows. Compare the two root sizes below:

(215) LEXICAL ENTRY OF 'TONGUE' (XL-SIZED)



(216) LEXICAL ENTRY OF 'JACKET' (M-SIZED)



Due to their different sizes, they take different suffixes at first. After the accusative, the M-sized root will need to backtrack, finding its way to the *-a* suffix in the genitive. At that point, the derivation gets unified with the XL-sized group, which has the same backtracking, only longer. From then, the rest of the derivation of both groups is identical. They backtrack some more for *-u* in the locative and for *-em* in the instrumental.

Knowing the lexical entries and the general schematics, let us proceed to the actual derivation.

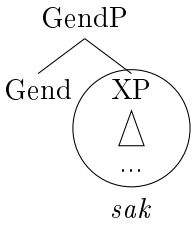
As always, we start by creating XP (217). It gets spelled out by the root (218), using a subset of the root's lexical entry:

(217) DERIVATION OF 'JACKET', NOM, INTERMEDIATE STEP

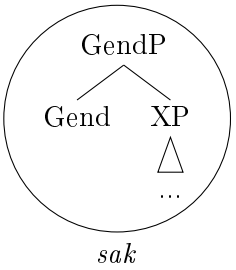
(218) DERIVATION OF 'JACKET', NOM, INTERMEDIATE STEP

Since the spellout has been successful, we move to the merging of the next feature, Gend (219). The new GendP phrase can still be spelled out by a subset of the root's lexical entry (220):

(219) DERIVATION OF 'JACKET', NOM, INTERMEDIATE STEP

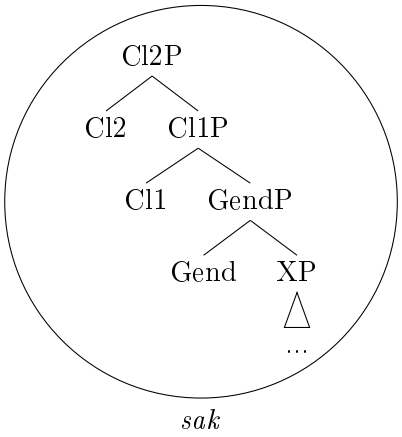


(220) DERIVATION OF 'JACKET', NOM, INTERMEDIATE STEP



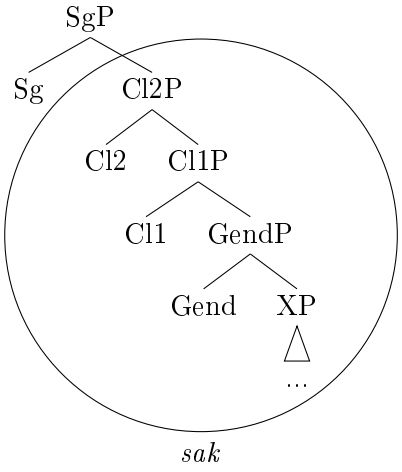
In the same manner, we merge two more features – Cl1 and Cl2, respectively. Each time, the root can spell out the newly created structure, resulting in the tree below:

(221) DERIVATION OF 'JACKET', NOM, INTERMEDIATE STEP



The next feature we merge is Sg (222):

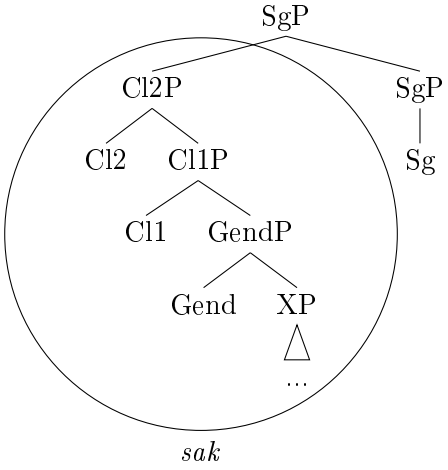
(222) DERIVATION OF 'JACKET', NOM, INTERMEDIATE STEP



At this point, the root can no longer spell out the whole structure because its lexical entry ends with Cl2. This is where the M-sized roots start to deviate from the XL-sized roots. The XL-sized roots continued by merging more features up to K2.

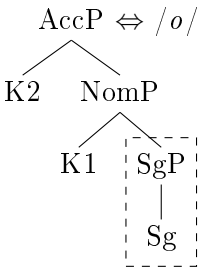
Because the M-sized roots are smaller and cannot spell out the new structure, they need to employ the next step of the spellout algorithm. The Spec movement is not possible, so the complement movement happens:

(223) DERIVATION OF 'JACKET', NOM, INTERMEDIATE STEP

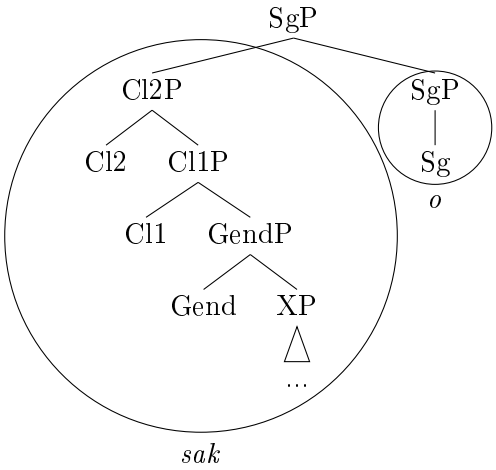


Thanks to the movement, the SgP is freed to be spelled out on its own, and the only lexical entry able to spell it out is *-o*. See (224) for the *-o* lexical entry – a dashed line shows the relevant subset. In (225), there is the spelled-out structure:

(224) LEXICAL ENTRY OF -O, SGP SUBSET

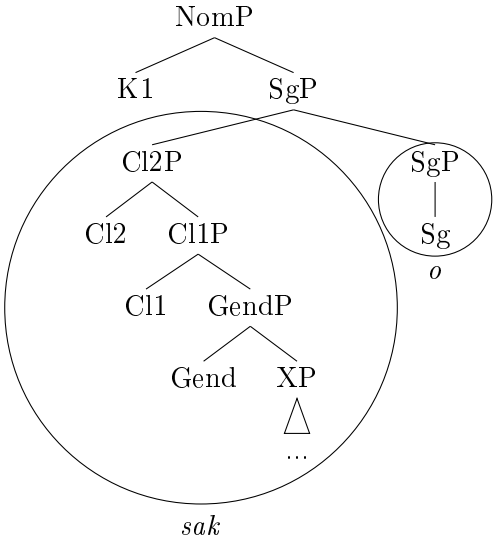


(225) DERIVATION OF 'JACKET', NOM, INTERMEDIATE STEP



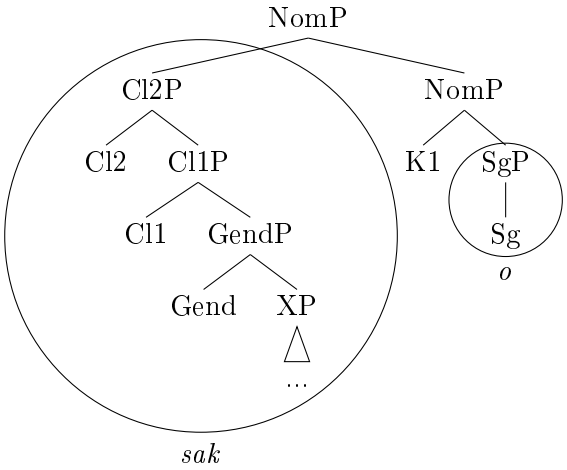
Everything has been successfully spelled out, so the derivation proceeds to merge the next feature, K1 (226):

(226) DERIVATION OF 'JACKET', NOM, INTERMEDIATE STEP



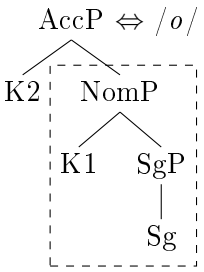
As the structure cannot be spelled out the way it is, the movement gets triggered. This time, the Spec movement is applicable because there is a phrase in the Spec of the complement of K1, namely Cl2P. So this phrase moves above the new feature:

(227) DERIVATION OF 'JACKET', NOM, INTERMEDIATE STEP

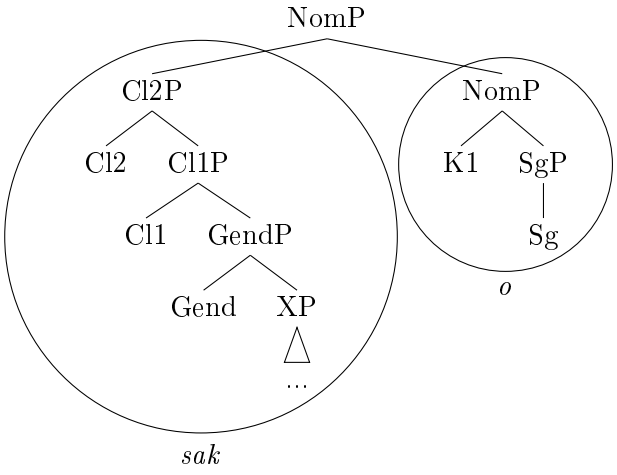


The right-side NomP can be spelled out by *-o* in its entirety. See (228) for the *-o* lexical entry with the relevant subset marked by a dashed line. In (229), we see the final result of the derivation:

(228) LEXICAL ENTRY OF -O, NOMP SUBSET



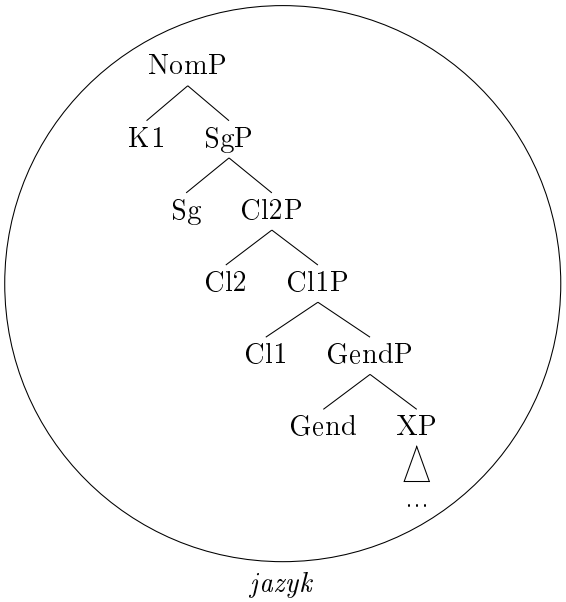
(229) DERIVATION OF 'JACKET', NOM, FINAL FORM (M-SIZED)



Since K1 is the nominative feature, this is the end of the nominative derivation.

Compare the final structure of the M-sized group derivation above (229) with the final structure of the XL-sized group nominative derivation below:

(230) DERIVATION OF 'TONGUE', NOM, FINAL FORM (XL-SIZED COMPARISON)

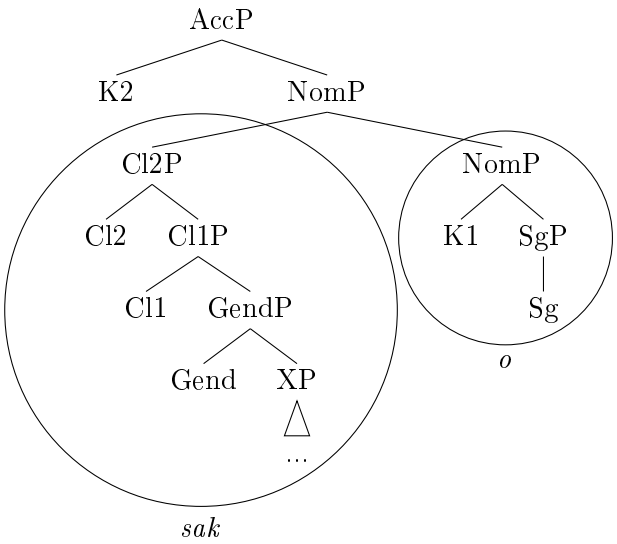


As said before, the XL-sized group roots can spell out the whole nominative structure alone. The M-sized group roots need the help of a suffix, and given that they grow to Cl2, they end up with the suffix *-o*.

Let us show one last derivation – the genitive of the M-sized group. The reason is to see the point where the group unifies its derivation (and therefore suffixes) with the other groups, e.g., the above shown XL-group.

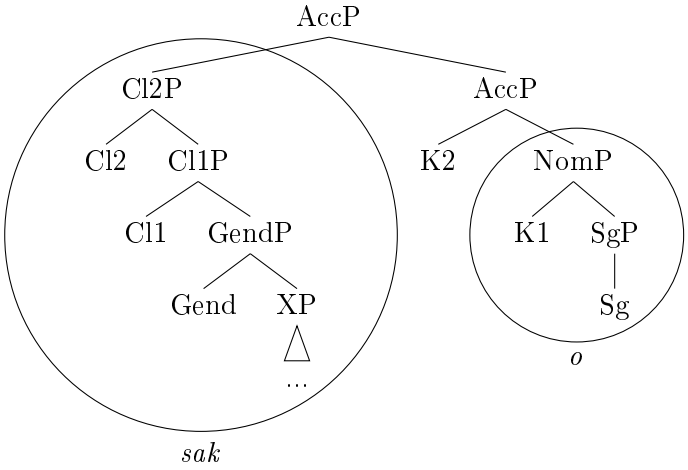
Continuing from the final point of the nominative derivation, we merge the K2 feature:

(231) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP



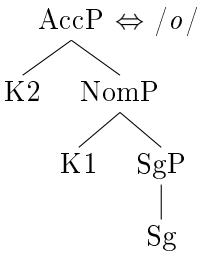
The structure cannot be spelled out in its entirety, so we deploy the Spec movement, bringing the Cl2P over the new K2 feature:

(232) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP

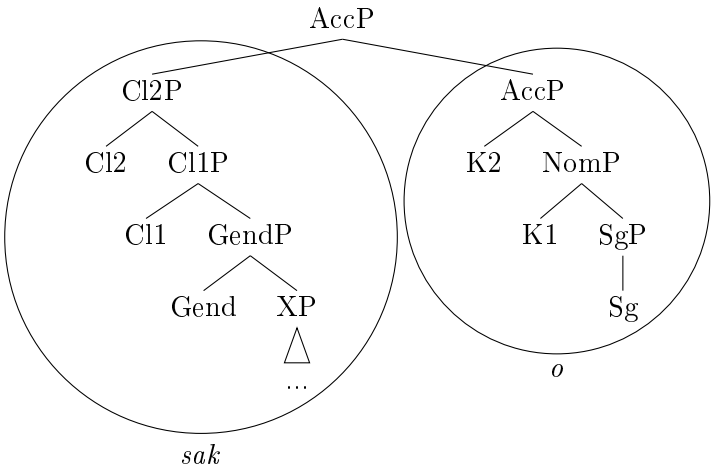


The lower AccP node can now be spelled out together with the NomP underneath it, as the Cl2P is not interfering anymore. It gets spelled out by *-o*. In (233), we see the lexical entry for *-o*. This time we are not using a subset but the whole entry. In (234), we see the spelled out structure:

(233) LEXICAL ENTRY OF -O

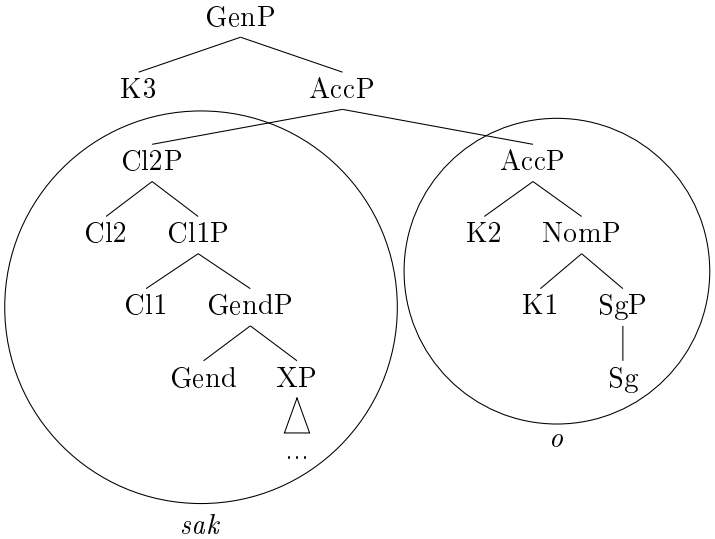


(234) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP



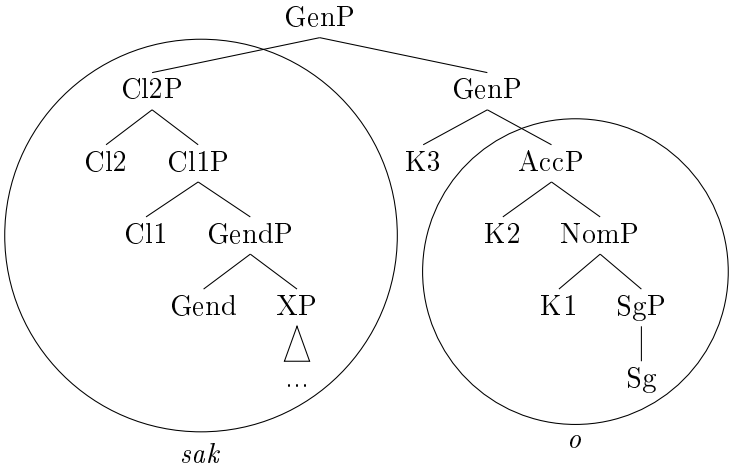
This grants us the accusative form, on top of which the genitive is built. Hence, we continue with the derivation by merging the last needed feature, K3.

(235) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP



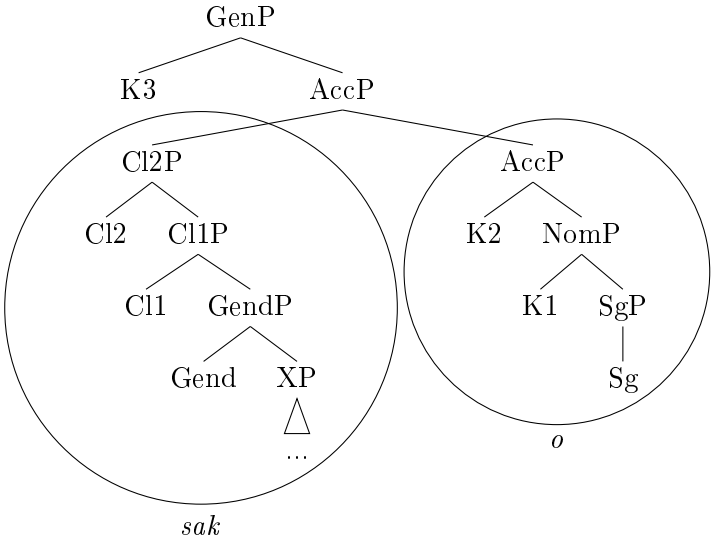
As before, the new structure cannot be spelled out as it is, so the next step of the spellout algorithm kicks in. First comes the Spec movement:

(236) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP



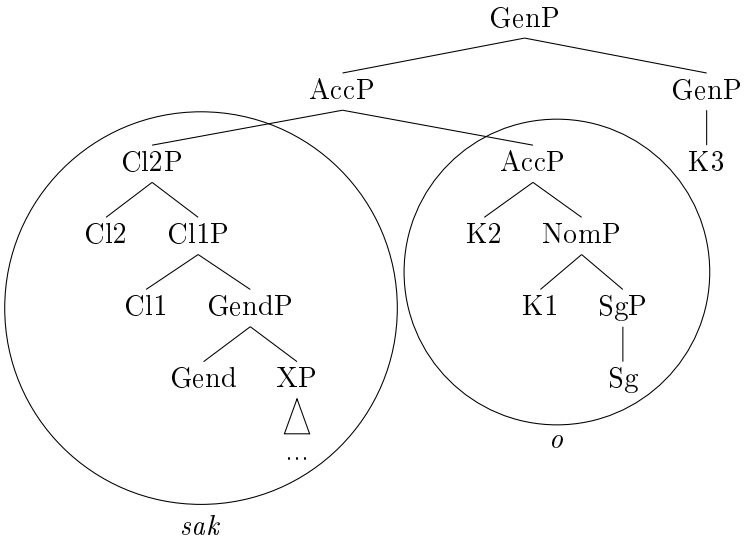
There is no match for the lower GenP in the lexicon. So we need to undo the movement:

(237) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP



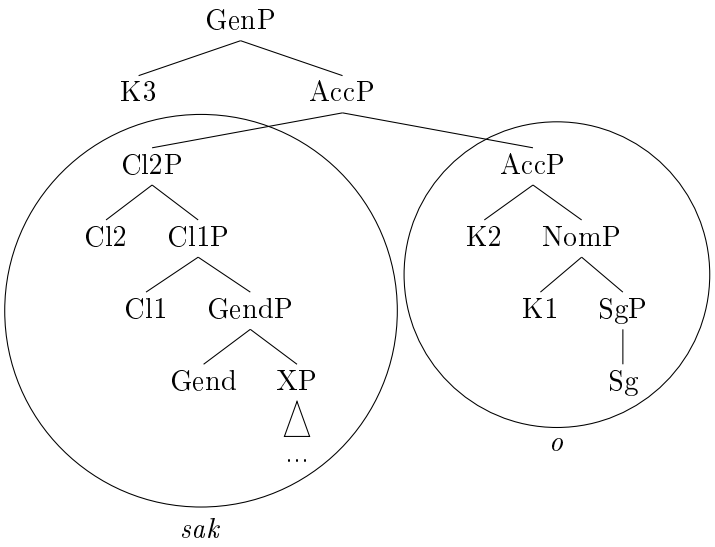
The next option is to try the complement movement:

(238) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP



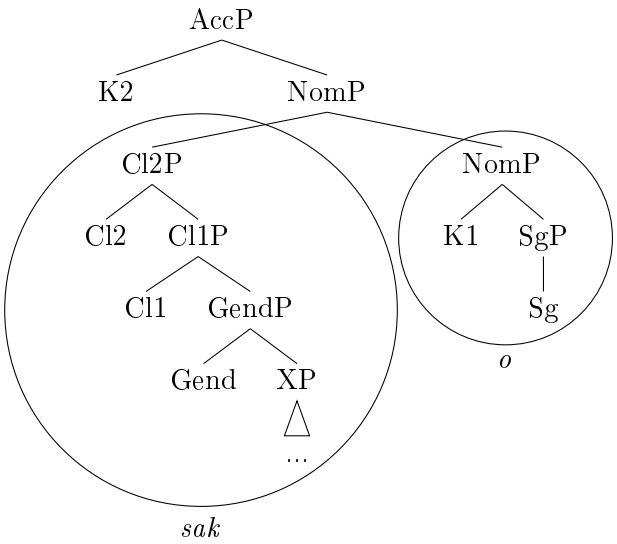
The complement movement has freed the lower GenP to be spelled out on its own. This still does not help because there is no lexical entry starting at K3. So this movement also needs to be undone:

(239) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP



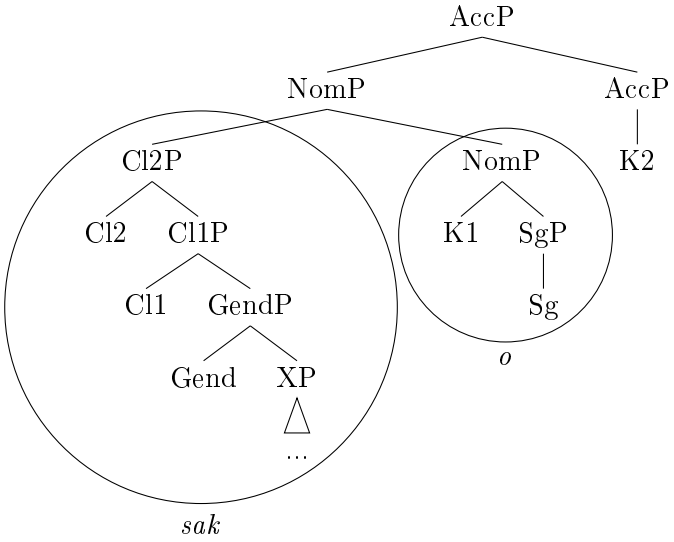
Since neither of the two movements worked, the backtracking gets triggered, bringing us to the previous stage of the derivation. That means to the point just after merging K2:

(240) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP, BACKTRACKING



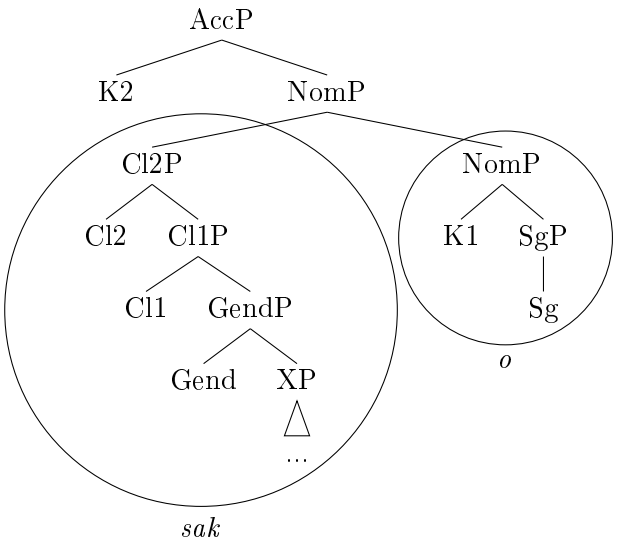
When we were at this stage before, we did the Spec movement, so this time we attempt the complement movement:

(241) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP, BACKTRACKING



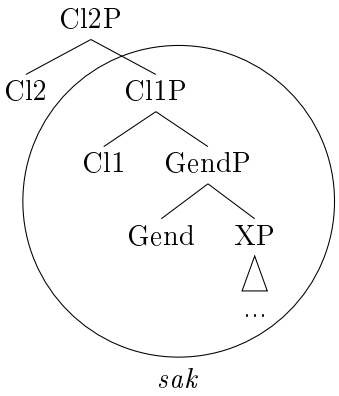
The complement movement does not help either as there is no lexical entry able to spell out just AccP. So also this movement has to be undone:

(242) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP, BACKTRACKING



We need to backtrack some more. Going through the same process described above – backtracking by one feature, trying the complement movement, learning that it does not help, undoing the movement, and backtracking more – we will backtrack to the Cl2 stage:[22]

(243) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP, BACKTRACKING

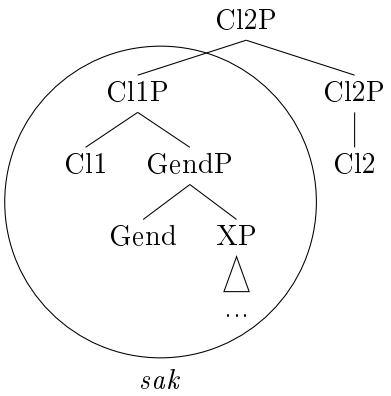


Note that we have entirely undone the suffix, and now we are backtracking into what was spelled out by the root – in other words, the root is "shrinking".

When we were at this stage before, we continued extending the root without movement. The next option is to try the Spec movement, which is, however, not applicable here. So we go straight for the complement movement:

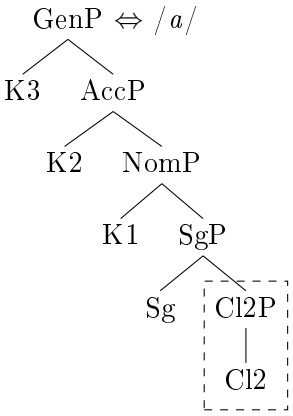
[22]There will be one false path at the Sg stage. For details, see the footnotes 20 and 21.

(244) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP, END OF BACKTRACKING

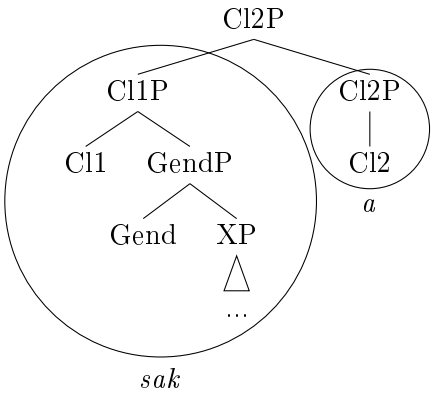


We try to spell out the lower Cl2P on its own and succeed because the *-a* lexical entry contains a matching node. In (245), there is the lexical entry for *-a*. In (246), there is the spelled-out structure:

(245) LEXICAL ENTRY OF -A, CL2P SUBSET



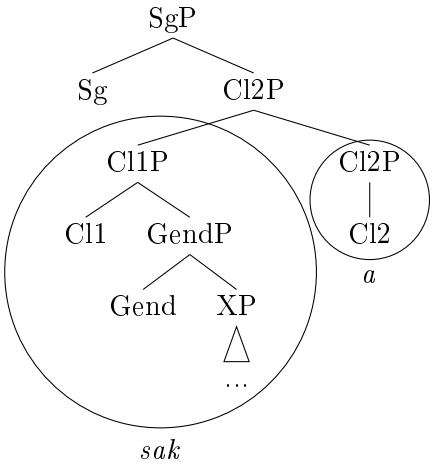
(246) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP



At this point, the derivation of the M-sized root unites with the derivation of the XL-sized group. It is a consequence of both groups backtracking to the same size, Cl1P. Unless they differ in features (which they do not), they are bound to have the same derivation steps and hence take the same morphemes from now on.

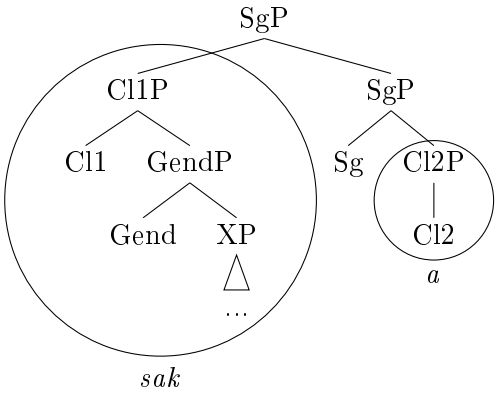
From here on, the derivation goes smoothly. We merge the next feature, Sg:

(247) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP



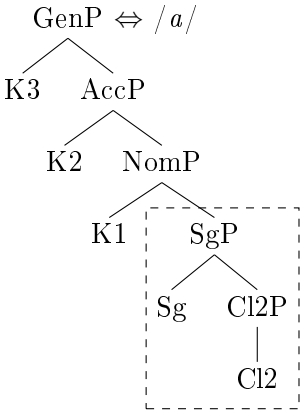
The structure cannot be spelled out as it is, so the Spec movement gets triggered:

(248) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP

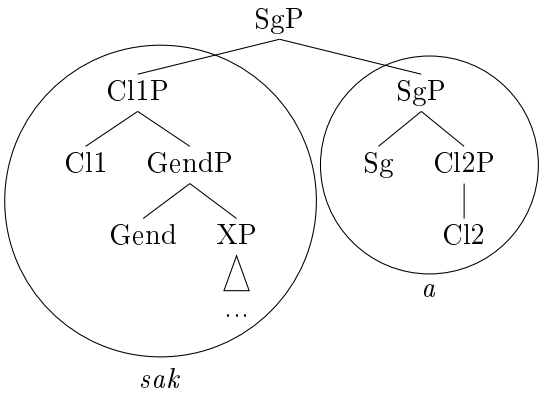


This frees the lower SgP to be spelled out with the right branch. In (249), we see that a bigger subset of the *-a* lexical entry is used this time. In (250), there is the spelled-out structure:

(249) LEXICAL ENTRY OF -A, SGP SUBSET

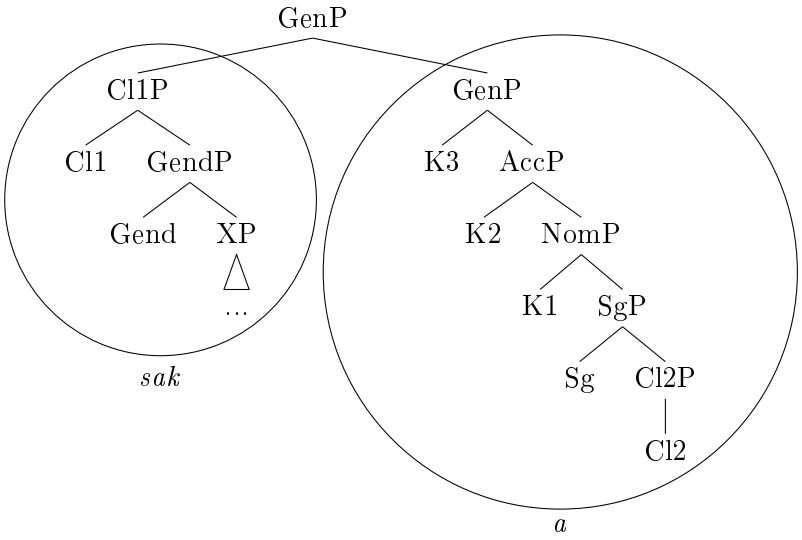


(250) DERIVATION OF 'JACKET', GEN, INTERMEDIATE STEP



The same process happens with the features K1, K2, and K3. Progressively, we merge them and always use the Spec movement to bring the root on top, allowing these new features to be spelled out with the right branch. Below is the final result:

(251) DERIVATION OF 'JACKET', GEN, FINAL FORM



This is the genitive structure. In the same way, we can derive any other case or a paradigm group from those we have presented up to now.

4 Root size + phonology: the palatalized paradigms

In section 2 with contours of the analysis, we described a system according to which Czech noun paradigms split into palatalized and non-palatalized. We discussed the non-palatalized paradigms in the previous section, and now we turn toward the second group.

The roots of palatalized paradigms can end either in a palatalized or in an ambiguous consonant:

(252) PALATALIZED AND AMBIGUOUS CONSONANTS IN CZECH

palatalized: ž, š, č, ř, c, j, ď, ť, ň
ambiguous: b, f, l, m, p, s, v, z

Besides the different root endings, palatalized roots also take a set of suffixes different from the non-palatalized group. Here are the palatalized paradigms we focus on in this section:[23]

(253) SURFACE FORMS OF 'SEA', 'ROOM', 'PALM' AND 'BONE'

	'sea' (N)	'room' (Mi)	'palm' (F)	'bone' (F)
NOM	moř-e	pokoj	dlaň	kost
ACC	moř-e	pokoj	dlaň	kost
GEN	moř-e	pokoj-e	dlaň-e	kosť-i
LOC	moř-i	pokoj-i	dlaň-i	kosť-i
DAT	moř-i	pokoj-i	dlaň-i	kosť-i
INSTR	moř-em	pokoj-em	dlaň-í	kosť-í

The cross-paradigm syncretisms are widespread in the palatalized group, as we can observe through the cell coloring. Our task will again be to suggest lexical entries that would cover them all.

4.1 Deriving the palatalized paradigms

A good place to start is the paradigm *pokoj* 'room' as it is one of the paradigms with no visible suffix in the nominative and accusative. Following the same logic explained in the previous sections, we can infer that the root spells out all the features up to the accusative. Below we can see a lexicalization table with both cases shown separately:[24]

[23] The paradigms *dlaň* 'palm' and *kost* 'bone' are transcribed quasi-phonologically to uncover what orthography conceals – that they both indeed have palatalized roots throughout the paradigm, with the exception of the nominative and accusative *kost*, which we keep for later.

[24] For the moment, we treat the palatalized paradigms as entirely separate from the non-palatalized. This results in postulating allomorphs with the same feature specification we used in the non-palatalized group. Such an account would, of course, ultimately fail to explain the distribution of suffixes between the two groups. We come back to unifying them later in this section.

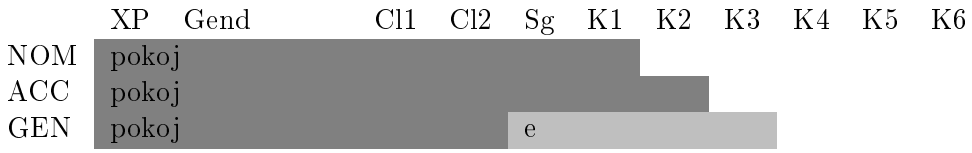
(254) LEXICALIZATION TABLE OF 'ROOM', NOM–ACC

The suggested lexical entry for *pokoj* is of the XL-size – the same as was *jazyk* 'tongue' in the non-palatalized group.

The genitive of this paradigm takes *-e*. Since every morpheme has to spell out at least one feature, the minimal hypothesis would be that *-e* spells out K3. We will, however, follow the same argumentation as with the suffixes in the non-palatalized group: The case features are above the number features. Since all the case suffixes in Czech nouns are number-specific (i.e., suffixes are not shared across the two numbers), we want to capture that in our analysis. Hence we put the foot of all the suffixes at least in Sg. That way, we have them specified for number, and they cannot appear in the plural.

So for the genitive *-e* of *pokoj*, we suggest a foot in Sg (to be amended later):[25]

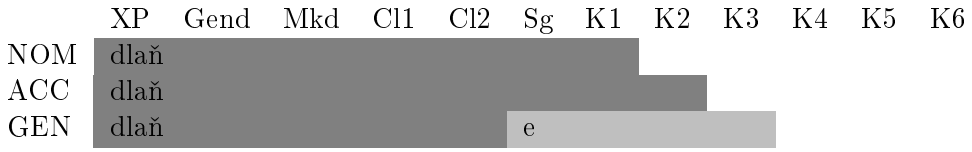
(255) LEXICALIZATION TABLE OF 'ROOM', NOM–GEN



As we can see, the root has to backtrack to spell out the genitive.

The same pattern can be seen in the *dlaň* 'palm' paradigm. The root shows no suffix in the nominative and accusative and then takes *-e* in the genitive. The same suffixes suggest the same size, so these are the entries we suggest for *dlaň*:

(256) LEXICALIZATION TABLE OF 'PALM', NOM–GEN



The only difference between *dlaň* 'palm' and *pokoj* 'room' is that *dlaň* is a feminine, so its functional sequence contains the Mkd feature. This does not affect the suffixal lexical entries above because Mkd is (in the cases relevant here) spelled out by the root low in the functional sequence and does not interact with the suffix. To influence its shape, the feature would have to be local to the suffix (i.e., to be spelled out as its foot or just below). We will see that happening later, in the instrumental.

[25] From now on, we sometimes leave a bigger space in the tables between the features Gend and Cl1 to suggest a missing Mkd feature. We add feminine paradigms in this section that have this feature, and the extra space allows for a quick visual comparison of both derivations. With the table set up like this, it is more evident that the *-e* in the table in (255) is the same as *-e* in the table in (256). This visualization is of no theoretical importance – Gend is a single feature, and there is no Mkd in the non-feminine forms like *pokoj* 'room'.

In the table in (253), one more paradigm shows no suffix in the nominative and accusative: the root *kost* 'bone'. Interestingly, it does not take *-e* in the genitive, as the two paradigms discussed above. In the previous section, we have already seen that if the paradigms have the same features, it is not possible to derive such a difference – and get *-e* with one root, but *-i* with another. The root *pokoj* 'room', as the masculine, lacks Mkd, which makes the sequence different from the sequence of *kost* 'bone'. This is, however, not true for *dlaň* 'palm', which is also feminine and has Mkd. Both *dlaň* and *kost* are likely to have exactly the same features. Therefore, we will set the *kost* paradigm aside and come to it in a moment.

The neuter paradigm *moř-e* 'sea' also shows *-e* in the genitive. In fact, it shows it from the nominative through the genitive. A fair first hypothesis is that these *-e* are all the same, although later, we will see that there are reasons to think otherwise. For the moment, let us proceed with these lexical entries:

(257) LEXICALIZATION TABLE OF 'SEA', NOM–GEN

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	moř				e						
ACC	moř				e						
GEN	moř				e						

Since the lexical entry for *-e* starts in Sg, the root is likely to grow to Cl2. Comparing *moř* with the root *pokoj* 'room', we once again see the pattern that if two roots share a suffix and one of them shows it earlier (like *moř-e* 'sea' here), the root showing it earlier must be smaller. We will assign the M-size to the root *moř*.

Looking back at the surface forms in (253), we see that the locative and dative of all the palatalized paradigms are the same, *-i*. So far, we have no reason to assume more backtracking, so the easiest way is to foot the lexical entry for *-i* in the same feature as we did for *-e* (this will have to be changed later). Below, we show the current locative and dative entries of the paradigm *pokoj* 'room'. It will be the same for the two other paradigms discussed so far:

(258) LEXICALIZATION TABLE OF 'ROOM', NOM–DAT

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	pokoj										
ACC	pokoj										
GEN	pokoj				e						
LOC	pokoj				i						
DAT	pokoj				i						

Before proceeding to the instrumental, let us come back to the paradigm *kost* 'bone' that we have left out. Here are the surface forms of the paradigm repeated (again transcribed quasi-phonologically), together with the paradigm *dlaň* 'palm' to help point out the issue:

(259) SURFACE FORMS OF FEMININE 'PALM' AND 'BONE'

	'palm'	'bone'
NOM	dlaň	kost
ACC	dlaň	kost
GEN	dlaň-e	kosť-i
LOC	dlaň-i	kosť-i
DAT	dlaň-i	kosť-i
INSTR	dlaň-í	kosť-í

Both paradigms have the same suffixes, except for in the genitive, where *dlaň* takes *-e* and *kost* takes *-i*. We have seen the same pattern in the non-palatalized group. We will not go through the details of the logic again but will follow the same resolution. (To remind yourself of the logic, refer back to (146) and the discussion after.) If two paradigms (with likely the same features) both start without any visible suffix and then split to take different suffixes in at least one of the cases, it must mean that one of them has an invisible suffix of some sort.

Following the same logic we had for *hrad* 'castle' in the previous section, we will argue that it must be *kost* 'bone' here. The theoretical basis of the reason has already been repeated several times throughout this work: both paradigms share the suffix *-i*, but *kost* 'bone' shows it earlier, which means that it must be smaller than *dlaň* 'palm'.

Also, similarly to the previous section, we will refer to the invisible suffix as "yer". The paradigm *kost* had a front yer in the Old Church Slavonic:

(260) OLD CHURCH SLAVONIC, Večerka (2006)

kost-ь 'bone'

This does not mean we assume the presence of an ultrashort vowel in modern Czech. The yer is simply a stand-in sign for any invisible suffix on which nature we remain agnostic and which is not of great importance since we will ultimately adopt a different account for both paradigms for which we suggested the yers.

If we were to work with the lexical entry for *-i* suggested for the three other palatalized paradigms above, we would get the following lexical entries for *kost* 'bone':

(261) LEXICALIZATION TABLE OF 'BONE', NOM–DAT

	XP	Gend	Mkd	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	kost					ь						
ACC	kost					ь						
GEN	kost					i						
LOC	kost					i						
DAT	kost					i						

The root *kost* only grows to Cl2 so that it can pick up first the yer and later *-i*, which are both footed in Sg.

On its own, this paradigm would work, but think about how the derivation of the genitive would go in the context where we also have the paradigm *dlaň* 'palm'. Compare the two genitives below:

(262) LEXICALIZATION TABLE OF 'BONE' AND 'PALM', GEN

	XP	Gend	Mkd	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
GEN	kost					i						
	dlaň					e						

The lexical entries for *-i* and *-e* are identical – starting at the same feature and spelling out the same part of the sequence. This creates an unsolvable competition, so our lexical entries must be wrong. What we need is a situation when the root *kost* 'bone' never reaches to *-e*. We can get this by pulling the foot of the yer and *-i* down:

(263) LEXICALIZATION TABLE OF 'BONE', NOM–DAT

	XP	Gend	Mkd	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	kost				ь							
ACC	kost				ь							
GEN	kost				i							
LOC	kost				i							
DAT	kost				i							

The root *kost* now only reaches to Cl1. If *-e* starts at Sg as previously suggested, *kost* will never be able to take it because it is too small and does not reach it.

The fact that we brought the foot of *-i* lower is going to show in the rest of the palatalized paradigms as well. Below we can see an adjusted lexicalization table for the *dlaň* 'palm' paradigm – there will have to be backtracking from the genitive *-e* to the locative/dative *-i*:

(264) LEXICALIZATION TABLE OF 'PALM', NOM–DAT

	XP	Gend	Mkd	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	dlaň											
ACC	dlaň											
GEN	dlaň					e						
LOC	dlaň				i							
DAT	dlaň				i							

Comparing the table above with the table in (263), the issue of unsolvable competition is resolved, and the derivations work correctly. The root *dlaň* grows high, and the first lexical entry it finds through backtracking for the genitive is *-e*. In the locative and dative, it needs to backtrack even more to spell out all the features and gets *-i*. The root *kost*, on the other hand, only grows to Cl1. It takes the yer in the nominative and accusative, and then, since it never manages to reach *-e*, will take on *-i* through the Superset Principle.

Ultimately, the surface forms suggest that the palatalized paradigms also differ in root sizes, just as we saw with the non-palatalized in the previous section. We suggest the following:

(265) pokoj/dlaň > moř > kost

The roots *pokoj* 'room' and *dlaň* 'palm' are the biggest, and for the ease of exposition, we refer to them as XL-sized. The root *moř* 'sea' is smaller, and we refer to it as M-sized. And the root *kost* 'bone' is the smallest of these, and we will assign it the XS-sized tag. For reasons that will become clear later, we are choosing the label XS and not S. Note that the XS-size will be discarded in the end because, as it turns out, there is a better way to analyze this paradigm. We are again skipping size L to leave space for a paradigm we will incorporate later. At this stage, all that is important is the relative difference in size between the four paradigms at hand.

Finally, we repeat the forms of the instrumental of the palatalized paradigms below:

(266) SURFACE FORMS OF 'SEA', 'ROOM', 'PALM' AND 'BONE', INSTR

	'sea' (N)	'room' (Mi)	'palm' (F)	'bone' (F)
INSTR	moř-em	pokoj-em	dlaň-í	kost-í

The instrumental of the masculine and neuter paradigms is the same as the instrumental of the masculine and neuter paradigms in the non-palatalized group: *-em*. The instrumental of the feminine paradigms differs and has the suffix *-í*.

Ideally, we would want to say that the *-em* in both groups is spelled out by the same lexical entry. This is the lexical entry we have suggested previously:

(267) LEXICALIZATION TABLE OF -EM

XP	Gend	Mkd	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
			em								

It has a foot in Cl1, so the palatalized roots like *pokoj* 'room' have to backtrack to get it:

(268) LEXICALIZATION TABLE OF 'ROOM'

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	pokoj										
ACC	pokoj										
GEN	pokoj				e						
LOC	pokoj			i							
DAT	pokoj			i							
INSTR	pokoj		em								

On their own, the lexical entries above work all right. It is the feminine suffix *-í* that offers us a clue that the foot of *-em* must still be adjusted. To see why, consider the feminine paradigm *dlaň*. Below we repeat its lexicalization table suggested so far:

(269) LEXICALIZATION TABLE OF 'PALM', NOM–DAT

	XP	Gend	Mkd	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	dlaň											
ACC	dlaň											
GEN	dlaň					e						
LOC	dlaň				i							
DAT	dlaň				i							

The question now is what part of the structure would the instrumental *-í* spell out.

We know it has to be footed in Cl2 (or lower) because that is the highest place the root reaches at that point (due to backtracking). It cannot start in Cl2, though, because if it did, the masculine and neuters would not be forced to backtrack to Cl1 for *-em* and would incorrectly take *-í* instead. Footing *-í* in Cl1 is also not an option because then both *-em* and *-í* lexical entries would spell out the same part of the sequence, once again giving us an unsolvable competition. Putting the foot of *-í* in Mkd or anywhere lower also fails – the feminine paradigms like *dlaň* 'palm' would then, through backtracking, first find *-em* and would be forced to take it, yielding an incorrect form *dlaň-em*.

Luckily, the split between the roots taking *-em* vs. *-í* gives us a clue: the split is clearly gender-based, putting against each other feminine and non-feminine paradigms. In other words, it seems that some a gender feature is involved in spelling both instrumental suffixes out. Below, we can see how the derivations would go if we footed the lexical items for *-em* and *-í* in the features connected to gender expression. In the non-feminine paradigms, we suggest that *-em* spells out the Gend feature:

(270) LEXICALIZATION TABLE OF 'ROOM'

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	pokoj										
ACC	pokoj										
GEN	pokoj				e						
LOC	pokoj			i							
DAT	pokoj			i							
INSTR	pokoj	em									

In the feminine paradigms, the *-í* can either have a foot in the same place (i.e., in Gend) or even one feature up, in Mkd. We put the foot in Mkd under the attempt to keep the analysis as simple as possible and hence not suggest more backtracking than necessary:

(271) LEXICALIZATION TABLE OF 'PALM'

	XP	Gend	Mkd	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	dlaň											
ACC	dlaň											
GEN	dlaň					e						
LOC	dlaň				i							
DAT	dlaň				i							
INSTR	dlaň		í									

This solution proves to give us the correct derivations. In the locative/dative stage, both *pokoj* 'room' (Masc) and *dlaň* 'palm' (Fem) share the same suffix, *-i*. This suffix, however, cannot spell out K6 in the instrumental, so backtracking is triggered. The first lexical entry able to spell out K6 that *dlaň* finds is *-í*, correctly deriving *dlaň-í*. Crucially, *pokoj* will not find the same suffix because *-í* is specified for Mkd, and there is no Mkd in *pokoj* due to its non-feminine nature. Hence *pokoj* backtracks further to Gend and correctly finds *-em*.

This way, we can derive all the palatalized paradigms. Next, we compare the lexical entries of the palatalized and non-palatalized paradigms to see whether they are mutually inclusive – something we want in order to explain how the two groups co-exist in Czech.

So far, we have been treating the palatalized and non-palatalized groups as if entirely separate. That is, of course, not the case, and any analysis aspiring to account for the complexity of Czech should be able to put them together and explain how the paradigms relate to each other.

In Section 4.2.1, we show how we can unify the two groups morphosyntactically. In Section 4.2.2, we investigate the nature of the phonological process that turns the non-palatalized endings into the palatalized ones.

Two observations suggest that the two big groups underlyingly have the same suffixes.

The first one is that both contain roots of the same size. We call the roots that have the same size but belong to a different palatalization group counterparts. So, e.g., *pokoj* 'room' is a palatalized counterpart of *jazyk* 'tongue'. Let us take this group as an example:[26]

(272) LEXICALIZATION TABLE OF 'TONGUE'

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	jazyk										
ACC	jazyk										
GEN	jazyk			a							
LOC	jazyk		u								
DAT	jazyk		u								
INSTR	jazyk	em									

LEXICALIZATION TABLE OF 'ROOM'

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	pokoj										
ACC	pokoj										
GEN	pokoj				e						
LOC	pokoj			i							
DAT	pokoj			i							
INSTR	pokoj	em									

[26] The palatalized paradigms are also represented by the feminine root *dlaň* 'palm', which – except for in the instrumental – has the same endings as *pokoj* 'room'. To avoid repetitiveness, we leave it out here.

Both paradigms happen to have the same backtracking patterns as well. After growing to K2, they backtrack in the genitive, then again in the locative, and then in the instrumental.

The second observation has to do with an implication relationship between the phonology of the counterpart suffixes. First, let us see the counterpart paradigms next to each other. They are ordered from the biggest root size to the smallest (XL, M, and XS, respectively). Remember also that the instrumental, unlike the rest of the cases, is guided by a gender principle and therefore works slightly differently:[27]

(273) SURFACE FORMS OF 'TONGUE', 'ROOM' AND 'PALM'

	'tongue'	'room'	'palm'
NOM	jazyk	pokoj	dlaň
ACC	jazyk	pokoj	dlaň
GEN	jazyk-a	pokoj-e	dlaň-e
LOC	jazyk-u	pokoj-i	dlan-i
DAT	jazyk-u	pokoj-i	dlan-i
INSTR	jazyk-em	pokoj-em	dlan-í

(274) SURFACE FORMS OF 'JACKET', AND 'SEA'

	'jacket'	'sea'
NOM	sak-o	moř-e
ACC	sak-o	moř-e
GEN	sak-a	moř-e
LOC	sak-u	moř-i
DAT	sak-u	moř-i
INSTR	sak-em	moř-em

(275) SURFACE FORMS OF 'CASTLE', AND 'BONE'

	'castle'	'bone'
NOM	hrad	kost
ACC	hrad	kost
GEN	hrad-u	kost-i
LOC	hrad-u	kost-i
DAT	hrad-u	kost-i
INSTR	hrad-em	kost-í

Seeing the counterpart paradigms placed next to each other, we can formulate a set of exception-less rules that map any suffix of the non-palatalized paradigms onto a suffix of the palatalized paradigm:

(276) IMPLICATIONAL RELATIONSHIP BETWEEN PALATALIZED AND NON-PALATALIZED SUFFIXES

NON-PAL		PAL
a	→	e
u	→	i
o	→	e

If a non-palatalized paradigm has *-a* or *-o* for a suffix, then the palatalized counterpart will have *-e* in the same environment. If a non-palatalized paradigm has *-u*, the palatalized counterpart will have *-i*.

[27] We leave out the paradigm *chleb-a* 'bread' (S-size) because, at this stage, it does not seem to have a counterpart – we come back to this later.

Based on these two observations, we hypothesize that the endings of the palatalized paradigms are derived from the endings of the non-palatalized paradigms by an application of a phonological rule. So, e.g., the genitive *-e* of *pokoj* 'room' is underlyingly the same as the genitive *-a* of *jazyk* 'tongue'. It is the phonology that muddles up their identity.

The following section shows how we can integrate both observations into one system. We first need to see whether the counterpart paradigms can be precisely the same size – which is a prerequisite for them to be syntactically the same. We will see that they can. The only thing needed is to adjust the feet of lexical entries wherever they currently differ.

Second, we need to look deeper into the phonology of the counterpart suffixes and see what processes might make a suffix come out as, e.g., *-a* in one paradigm but *-e* in the other. Here, we will employ Element Theory decomposition.

In the end, we will conclude that the counterpart suffixes are indeed the same and that the only difference is that the palatalized suffixes have a floating I at the end of the root:

(277) non-palatalized: sak-A
palatalized: mořI-A

The floating I in *moř* above will then combine with the suffix -A, with an outcome of *-e*:

(278) I + A = e

This line of argumentation will allow us to put the palatalized and non-palatalized paradigms within the same account and explain the surface differences.

4.2 Incorporating palatalized and non-palatalized paradigms within one system

In this subsection, we turn to integrating both of the big groups, non-palatalized and palatalized, to be able to work within one system. We start by unifying lexical entries of the counterpart paradigms and then decompose their suffixes according to Element Theory.

4.2.1 Unifying the lexical entries

Let us start with adjusting the lexical entries so that each counterpart suffix spells out the same chunk of the functional sequence.

We first look at the M-sized roots as it turns out that we can see all necessary changes through them. Below we repeat the original lexical entries from the sections above:

(279) PALATALIZATION TABLE OF 'JACKET'

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	sak				o						
ACC	sak				o						
GEN	sak			a							
LOC	sak		u								
DAT	sak		u								
INSTR	sak	em									

PALATALIZATION TABLE OF 'SEA'

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	moř				e						
ACC	moř				e						
GEN	moř				e						
LOC	moř			i							
DAT	moř			i							
INSTR	moř	em									

The first place where they differ is the genitive. Since our goal is to unify the non-palatalized *-a* and palatalized *-e*, there are two ways we can go about it. Either we try to liken the non-palatalized suffixes to the palatalized ones, or the other way round.

The former possibility, with bringing the foot of *-a* up, is depicted in table (280) and it will fail. To see that, consider the S-sized paradigm *chleb* 'bread'. The root takes *-a* in the nominative, accusative and genitive. According to our analysis, this is the same *-a* as in *sak* 'jacket'. So once we have adjusted the *-a* in the paradigm *sak-o*, we also need to adjust the root *chleb* to spell out the feature that we took out of *-a*:

(280) PALATALIZATION TABLE OF 'JACKET', NOM–GEN

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	sak				o						
ACC	sak				o						
GEN	sak				a						

PALATALIZATION TABLE OF 'BREAD', NOM–GEN

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	chleb				a						
ACC	chleb				a						
GEN	chleb				a						

However, by doing this, both roots become indistinguishable, and we would expect them to inflect alike, which they do not.

Coming back to the original lexical entries in (279), since adapting the non-palatalized suffixes to the palatalized ones failed, we try the opposite strategy. That means we have to distinguish two *-e* in the paradigm of *moř* 'sea' (one originating from *-o*, the other from *-a*):

(281) PALATALIZATION TABLE OF 'JACKET'

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	sak				o						
ACC	sak				o						
GEN	sak			a							
LOC	sak		u								
DAT	sak		u								
INSTR	sak	em									

PALATALIZATION TABLE OF 'SEA'

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	moř				e1						
ACC	moř				e1						
GEN	moř			e2							
LOC	moř		i								
DAT	moř		i								
INSTR	moř	em									

Having to distinguish two *-e* does not seem ideal. Nevertheless, besides the fact that the counterpart suffixes now spell out precisely the same features, there is another reason to think that this analysis might be on the right track. Look at the old Czech forms:

(282) OLD CZECH, Pleskalová (2001)

	'sea'
NOM	mor'-e
ACC	mor'-e
GEN	mor'-a

In the older stages of Czech, the genitive was different from the nominative and accusative forms.[28] Due to this data, it seems feasible that the two *-e* might, underlyingly, be different even in modern Czech.

The second adjustment we made in the table with the paradigm *moř* 'sea' above is pulling the foot of *-i* one feature down to Cl1. This way, it spells out the same part of the functional sequence as its non-palatalized counterpart *-u*.[29]

[28] The same pattern shows in some modern Czech dialects. Look, e.g., at the paradigm below (Lanžhot, East Moravia; Balhar 2002):

	'field'
NOM	pol-e
ACC	pol-e
GEN	pol-a

The word *pol-e* 'field' belongs to the same paradigm as *moř-e* 'sea'. Just as in the old Czech, the nominative and accusative suffixes differ from the suffix in the genitive.

[29] It is worth noting that by adjusting the foot of the genitive *-e2*, it was necessary to adjust the foot of *-i*. Have we left the foot of *-i* in Cl2, we would, incorrectly, get *-e2* in the genitive of the paradigm *kost* 'bone'. So technically speaking, the only change we need to do on our own is moving the foot of *-e2*, and the rest will be taken care of by the necessities of the spellout algorithm.

These two small adjustments – pushing the genitive *-e* and the locative/dative *-i* one feature down – do not disrupt the analysis. It still works. However, the two changes take care of the rest of the paradigms so that all the counterpart suffixes now spell out the same part of the structure. See this on XL-sized roots below with the newly adjusted suffixes *-e2* and *-i*:

(283) PALATALIZATION TABLE OF 'TONGUE'

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	jazyk										
ACC	jazyk										
GEN	jazyk			a							
LOC	jazyk		u								
DAT	jazyk		u								
INSTR	jazyk	em									

PALATALIZATION TABLE OF 'ROOM'

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	pokoj										
ACC	pokoj										
GEN	pokoj			e2							
LOC	pokoj		i								
DAT	pokoj		i								
INSTR	pokoj	em									

PALATALIZATION TABLE OF 'PALM'

	XP	Gend	Mkd	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	dlaň											
ACC	dlaň											
GEN	dlaň				e2							
LOC	dlaň			i								
DAT	dlaň			i								
INSTR	dlaň		í									

The same goes also for the XS-sized roots, *hrad* 'castle' and *kost* 'bone':

(284) PALATALIZATION TABLE OF 'CASTLE'

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	hrad		ъ								
ACC	hrad		ъ								
GEN	hrad		u								
LOC	hrad		u								
DAT	hrad		u								
INSTR	hrad	em									

PALATALIZATION TABLE OF 'BONE'

	XP	Gend	Mkd	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	kost			ь								
ACC	kost			ь								
GEN	kost			i								
LOC	kost			i								
DAT	kost			i								
INSTR	kost		í									

Just through adjusting *-e2* and *-i*, we get all the counterpart suffixes precisely the same.

The only paradigm we have not mentioned here yet is the S-sized *chleb* 'bread':

(285) PALATALIZATION TABLE OF 'BREAD'

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	chleb			a							
ACC	chleb			a							
GEN	chleb			a							
LOC	chleb		u								
DAT	chleb		u								
INSTR	chleb	em									

At this point, the paradigm does not have a palatalized counterpart. It is theoretically possible that the paradigm *moř* 'sea' could take place if we let go of the two *-e* analysis. It could also be that what we consider the *moř*-like roots are, in fact, roots of two sizes: M and S, and due to the homophony in the genitive, we cannot distinguish them. For the moment, we choose to keep *moř* in the M-sized group, though, because we consider the diachronic evidence for having two *-e* convincing. Note that in the subsection 4.2.4, we add a palatalized paradigm *soudc* 'judge' that will work as a counterpart to *chleb* 'bread'.

Now that we have unified the sizes of the suffixal lexical entries, we will look at the phonology side of the equation.

4.2.2 Undoing the phonology

This section aims to undo the phonology, which will allow us to see that the suffixes of the palatalized paradigms can be derived from those of non-palatalized paradigms. To

reach this goal, we start by introducing Element Theory (Kaye et al. 1985, Harris and Lindsey 1995, Backley 2011) that helps us do just that.

In Element Theory, there are three subphonemic vowel primitives (called elements), A, I, and U. These correspond to the three vowels *a*, *i*, *u* respectively:

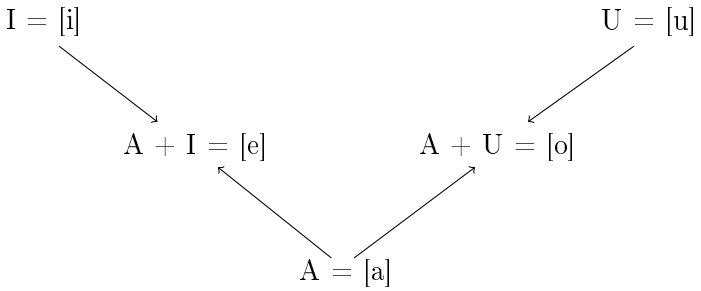
(286) ELEMENT THEORY PRIMITIVES

I = [i] U = [u]

A = [a]

The rest of the vowels (*-e* and *-o* for Czech) arise as a combination of these primitives:

(287) ELEMENT THEORY COMBINATIONS FOR CZECH

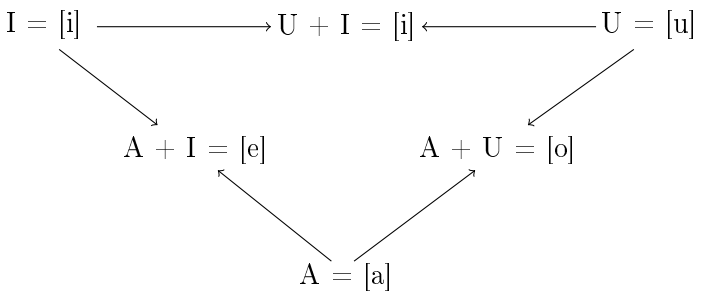


The vowel *-e* is a combination of A + I. The vowel *-o* is a combination of A + U.

As for the combination of U + I, there are opposing views on whether these elements can ever combine.[30] In analyses that allow it, the result is a close front rounded vowel *y*. Czech does not have such a vowel (orthographically, it uses *y* which is, however, pronounced as [*i*]), and we argue that the two elements never combine in Czech. That, however, does not mean that they cannot find themselves in each others' proximity via derivation. When that happens, they get into a competition, and I wins:

[30] Cf. Kaye et al. (1985) arguing that while the impossibility of the combination is an unmarked option across languages, there are some cases when it is possible. Backley (2011) calls it "universally a marked combination". For an opposing view, cf. Scheer (1996) (cited through Passino 2009) whose analysis does not allow the combination in any language.

(288) ELEMENT THEORY COMBINATIONS FOR CZECH, U + I



The argument for I winning over U comes from the French borrowings. Overwhelmingly, when a French word contains the vowel *y* (U + I), the vowel is then adapted to Czech as *i* and not *u* (Pešek 2017, transcription mine):[31]

(289) CZECH BORROWINGS FROM FRENCH WORDS WITH /Y/

FR		CZ	
bureaucratie [byʁokʁasi]	→	byrokracie [bɪrokra͡tsɪjɛ]	'bureaucracy'
sujet [syʒɛ]	→	syžet [sɪʒɛt]	'storyline'

With the elements in place, we now show that the palatalized paradigms are derived from the non-palatalized by adding the element I to the non-palatalized. To see that, consider the relationship between the vowels of the palatalized and non-palatalized paradigms again:

(290) IMPLICATIONAL RELATIONSHIP BETWEEN PALATALIZED AND NON-PALATALIZED SUFFIXES

NON-PAL		PAL
a	→	e
u	→	i
o	→	e

Whenever a non-palatalized paradigm has *-a* as a suffix, its palatalized counterpart will be *-e*, and so on. We will now decompose the above vowels in Element Theory.

The first row of (290) is straightforward. The non-palatalized *-a* stays A as one of the primitives. The palatalized *-e* decomposes into A + I. Note that both sides share the element A. The palatalized side furthermore has an extra I:

(291) A → E IMPLICATION DECOMPOSED

NON-PAL			PAL	
A	[a]	→	A + I	[e]

[31] The argumentation is built on Paradis and Prunet (2000). The paper looks into how French *y* is adapted into several languages lacking the sound. In their set of data, *i* is the prevalent adaptation. Passino (2009) and Lampitelli (2014) argue along similar lines for Italian. Backley (2011) counts with basically the same process in Polish, describing it in terms of I *suppressing* U.

That is an enticing picture since the two sides differ in palatalization, and I is a known palatalizer. So let us explore whether the other two implications could follow the same pattern. Namely, whether the palatalized side could have the same element(s) as the non-palatalized side but with an additional I. If that were true, the picture would look as we show below:

(292) U → I AND O → E IMPLICATIONS DECOMPOSED

NON-PAL			PAL	
U	[u]	→	U + I	[i]
A + U	[o]	→	A + U + I	[e]

The first line works. The non-palatalized paradigms have U. If it is indeed true that U + I cannot combine in Czech and that I wins, we get the correct result for the palatalized paradigms – U is not used, while I surfaces as a suffix *-i*.

In the second line, the suffix *-o* is decomposed into A + U. If we repeat the same combination of elements for the palatalized group and add the extra I, we get the correct results as well. The elements U and I will again compete, I will win and then combine with A, correctly yielding *-e*.

The fact that both sides can repeat the same elements strongly suggests that treating the palatalized and non-palatalized paradigms as the same might be on the right track. So we conclude that the palatalized suffixes are merely the non-palatalized suffixes combined with I:

(293) V → V + I

It is a regular phonological process that conceals the uniformity of the two big groups.

Here is an example of how this looks for concrete paradigms. Below, we see the genitive of the XL-sized counterparts, *jazyk* 'tongue' for the non-palatalized group and *pokoj* 'room' for the palatalized group:

(294) THE XL-SIZED PARADIGMS 'TONGUE' AND 'ROOM' IN THE ET DECOMPOSITION

jazyk		A	→	jazyk-a
pokoj	I	A	→	pokoj-e

They both have the same suffix A. The difference is that the palatalized paradigm *pokoj* 'room' has an additional element I. This I then combines with the element A, creating the surfacing suffix we called *-e2* before.

We have skipped the instrumental suffixes so far. There are two that we have seen up to now: *-em* (in masculines and neuters) and *-í* (in palatalized feminines). To stay cohesive, we decompose them into -AIm (for *-em*) and -I+length (for *-í*). The *-em* can be found both in the palatalized and non-palatalized group. The main point is that since it already contains I in the suffix, adding another I will not change it. We have yet to see the shape of the non-palatalized feminines, though, so we will come back to the details of the instrumental derivations later.

The following section looks at where the element I resides and what its relationship with the palatalized group is.

4.2.3 The nature of the palatalizing I

One last significant puzzle is the nature of the element I that appears in the palatalized paradigms. This section argues that the most natural explanation for its appearance is that it is a floating element attached to the root.

First, let us consider where the floating element is. From its distribution, appearing only with certain roots, it must be next to the root. Since it can combine with the suffix, the only suitable place is between the root and the suffix.

Building upon that, there are two more options. Either it is a self-standing suffix (between the root and the case suffix), or it is a floating element at the end of the palatalized roots:

(295) root-I-
rootI-

In Nanosyntax, only the second option is possible. If the I is a floating element at the end of the palatalized roots, it easily combines with suffixes of different heights. Take, e.g., the genitive of the palatalized paradigm *dlaň* 'palm' (XL-size). In our analysis, it is a combination of the root *dlaň*, the element A and the element I (in this variant, floating at the end of the root):

(296) LEXICALIZATION TABLE OF 'PALM', GEN

	XP	Gend	Mkd	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
GEN	dlaňI				A							

In such a layout, the element I can easily combine with -A to create the surfacing suffix *-e2*.

This works well for roots of other sizes as well. Take, e.g., the genitive of another palatalized paradigm, *kost* 'bone' (XS-size):

(297) LEXICALIZATION TABLE OF 'BONE', GEN

	XP	Gend	Mkd	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
GEN	kostI			U								

Even though the genitive suffix of *kost* starts lower than -A in the previous example, the root-anchored I will have no problem combining with it. As has been argued before, combining I and U will give us a surface *-i*, which is correct for this paradigm.

The second option from (295), in which the element I would be a suffix on its own, does not work. Take the paradigm *dlaň* 'palm' again – this time, we see it with the element decomposition for which we have argued above:

(298) LEXICALIZATION TABLE OF 'PALM'

	XP	Gend	Mkd	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	dlaň											
ACC	dlaň											
GEN	dlaň				A							
LOC	dlaň			U								
DAT	dlaň			U								
INSTR	dlaň		I+length									

Since the element I can be seen both in masculine and feminine paradigms, it has to be footed above the Mkd feature. If it were footed below and hence contained Mkd, we would never be able to get it in the masculine paradigms since these are not specified for Mkd and would not match the lexical entry.

At the same time, it has to end below Cl2 to combine with -A. This would make the I spell out only the Cl1 phrase:[32]

(299) MOCK LEXICALIZATION TABLE OF 'PALM', GEN

	XP	Gend	Mkd	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	dlaň											
ACC	dlaň											
GEN	dlaň			I	A							
LOC	dlaň			U								
DAT	dlaň			U								
INSTR	dlaň		I+length									

This is problematic in several places in the derivation, but let us focus on the locative as an example. Both the elements -I and -U are footed in the same feature, Cl1, and compete. Competing means they are mutually exclusive. In this case, -U would win because it can spell out all the remaining features. We would never get I there, and the expected surface form would, incorrectly, be *dlaň-u*.

Perhaps an even more convincing argument for the element I not being a self-standing affix (if not as easy to show) is that lexical entries always grow as much as possible. This means that I would always grow to the same size, and we would expect anything above it to be of the same size, too. Given that the suffixes of different root sizes start at different places, this is impossible. So the two facts, first, that we have different root sizes, and second, that I must be located between the root and the suffixes, shows that I must not be a self-standing affix.

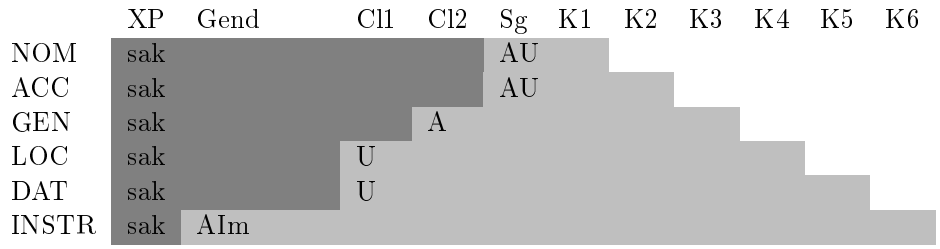
Because of the above, we conclude that the element I is a floating part of the palatalized roots in the following manner:

(300) rootI-

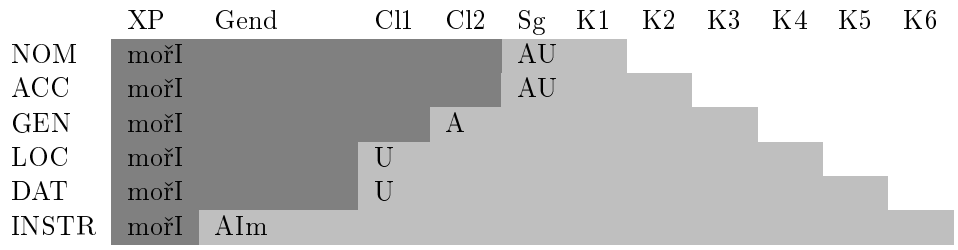
[32] There is a theoretical option of the lexical entry for I growing above Cl2 and then backtracking below it to take A. This would, however, not change anything about the analysis. It would only unnecessarily complicate it, so we do not consider it here.

Below, we can see the complete account of the M-sized paradigms as an example:

(301) LEXICALIZATION TABLE OF 'JACKET'



LEXICALIZATION TABLE OF 'SEA'



Both paradigms grow to the same size, to Cl2. That means they take the same suffixes, too. The root-final floating I creates the superficial impression of a different set of suffixes for the palatalized paradigms. It combines with the suffixal elements, which is how we get, e.g., *-e* in the nominative of the palatalized, but *-o* in the nominative of the non-palatalized paradigms.[33]

Note also the nominative/accusative and the genitive distinction in the palatalized paradigm *moř-e* 'sea'. The decomposition uncovers that the two *-e* on the surface are indeed homophonous – underlyingly, they are different (-AU vs. -A, respectively).

This is a considerable simplification of the seemingly complex system of Czech nouns. For eight different paradigms with forty-eight cells, we need no more than six suffixal lexical entries. All the inter-paradigm and cross paradigm syncretisms are accounted for as well.

4.2.4 Adding 'judge' to the picture

The last paradigm we tackle in this subsection is *soudc-e* 'judge'. We delayed its presentation because its surface forms might be misleading – the paradigm seemingly has the same suffixes as *moř-e* 'sea'. In contrast, we will argue that it is instead a counterpart to the non-palatalized S-sized paradigm *chleb-a* 'bread'.

See the surface forms of *soudc-e* 'judge' below, compared to the palatalized paradigm *moř-e* 'sea':

[33] Note that suggesting AU for the nominative and accusative of *moř-e* is well-aligned with the diachronic facts since its predecessor in Old Church Slavonic belonged to the so-called o-stems that actually had *-o* in these cases (Večerka 2006).

(302) SURFACE FORMS OF 'JUDGE' AND 'SEA'

	'judge' (Ma)	'sea' (N)
NOM	soudc-e	moř-e
ACC	soudc-e	moř-e
GEN	soudc-e	moř-e
LOC	soudc-i	moř-i
DAT	soudc-i	moř-i
INSTR	soudc-em	moř-em

The root *soudc* takes the palatalized set of endings, and is – as already indicated – identical to the *moř-e* 'sea' paradigm. We suggested that *moř-e* is a palatalized counterpart to *sak-o* 'jacket', and the same could be said for *soudc-e* 'judge' (to be amended):

(303) DECOMPOSED FORMS OF 'JUDGE', 'SEA' AND 'JACKET'

	'judge' (Ma)	'sea' (N)	'jacket' (N)
NOM	soudcI-AU	mořI-AU	sak-AU
ACC	soudcI-AU	mořI-AU	sak-AU
GEN	soudcI-A	mořI-A	sak-A
LOC	soudcI-U	mořI-U	sak-U
DAT	soudcI-U	mořI-U	sak-U
INSTR	soudcI-AIm	mořI-AIm	sak-AIm

We will not pursue this line of thinking because unlike for *moř-e* 'sea', there is no good reason to think that *soudc-e* 'judge' has an underlying *-AU/o* suffix. Instead, we will go the easier road and decompose all the *-e* as -A (with the floating I in the root), which means that *soudc-e* ends up as a counterpart to the non-palatalized S-sized paradigm *chleb-a* 'bread':

(304) DECOMPOSED FORMS OF 'JUDGE' AND 'BREAD'

	'judge' (Ma)	'bread' (Mi)
NOM	soudcI-A	chleb-A
ACC	soudcI-A	chleb-A
GEN	soudcI-A	chleb-A
LOC	soudcI-U	chleb-U
DAT	soudcI-U	chleb-U
INSTR	soudcI-AIm	chleb-AIm

Through this, *soudc-e* 'judge' joins the S-sized group of paradigms. Its lexicalization table looks as follows:

(305) LEXICALIZATION TABLE OF 'JUDGE'

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	soudcI			A							
ACC	soudcI			A							
GEN	soudcI			A							
LOC	soudcI		U								
DAT	soudcI		U								
INSTR	soudcI	AIm									

In the first three cases, the root reaches to Cl1 and selects the -A suffix. In the locative, the root backtracks, and the first suffix it finds is -U. Finally, it also backtracks in the instrumental, and since it is a non-feminine paradigm, it gets -AIm.

Generally, the paradigm *soudc-e* is easy to incorporate and does not raise any additional challenges.

4.3 How the suffixes are stored in the lexicon

In this subsection, we tackle two questions. The first one is about how and why the floating I links to the skeleton for the spell out. The second one has to do with the exact phonological representation of the suffixes. Ultimately, we want to understand how the two parts – the floating I and the suffix – interact.

We build the argumentation on several different phenomena: e-zero alternation, consonant-zero alternation, and paradigms with a stem-final long vowel.

All three phenomena have been discussed in the literature before. We try to put them together into one consistent account, also adding the floating palatalizer I. The central part of the proposal is that vocalic suffixes come from the lexicon with both a melody and the skeletal CV but are not linked to it (see below for the genitive -A):[34]

(306) PROPOSITION ON HOW CZECH VOCALIC SUFFIXES ARE STORED

C V

A

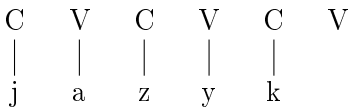
We couch this part of the analysis in the CV theory (Lowenstamm 1996, Scheer 2004).

The CV theory proposes a so-called CV-tier for the phonology module. A minimal building block of the CV-tier is a CV unit, and in the strict CV version, C cannot be inserted without V and vice versa. The Cs and Vs function as slots for melodic segments. Only those segments that are connected to CV slots through association lines are phonetically realized.

As an example, see the root *jazyk* 'tongue' of the non-palatalized XL-sized paradigms in the CV account:

[34] I am indebted to Pavel Caha for suggesting the analysis presented in this section. It will be worked out in more detail in a paper by Caha, Ziková, and Janků.

(307) 'TONGUE' IN CV THEORY, NOM

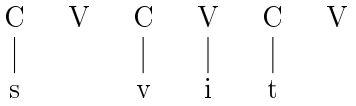


Since we always see the whole root pronounced, we can assume that it already comes connected to the CV slots from the lexicon (instead of being created through a derivation). Due to the strict CV hypothesis, there will be an empty nucleus at the end of each consonant-final root. This will turn out to be essential for the suggested account.

One part of the analysis will also build on the notion of government. Government is a "regressive relation that holds between the syllabic constituents: Nuclei govern either other Nuclei, or their own Onsets. What is important is that only those Nuclei which are not governed display Government" (Ziková 2007). Take the example of *svit* 'shine', which is of the same paradigm as the above *jazyk* 'tongue'.

Unlike *jazyk*, *svit* has a consonant cluster in the root. The CV theory depicts this by a structure with an empty vowel slot in between two consonants:

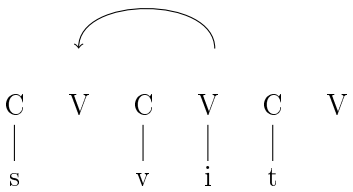
(308) 'SHINE' IN CV THEORY, NOM



If empty nuclei were unconstrained (e.g., if we could have three empty nuclei in a row), then the CV theory would make few (if any) predictions about the restrictions on consonant clusters. However, it has been observed that clusters (in languages that allow them) are cross-linguistically restricted. Therefore, Government Phonology works with the notion of government, limiting the sequences of empty nuclei (thereby indirectly limiting the sequences of consonants).

The standard theory says that in languages like Czech (which allow for consonant clusters), a filled nucleus position will govern the preceding vocalic slot, making the structure possible:

(309) 'SHINE' IN CV THEORY, NOM, GOVERNMENT OF THE EMPTY V SLOT



The arrow above the skeleton depicts a successful government.

If the *-i-* slot were also empty, the structure would fail. Below, we are using a non-existent word *svt* to depict this:

(310) MOCK WORD WITH A FAILED GOVERNMENT OF AN EMPTY V SLOT

C	V	C	V	C	V
|		|		|	
s		v		t	

The crossed arrow above the skeleton shows that this time the government between the same two slots failed – because an empty slot can never govern.

As a last note, in languages that allow a word-final coda, it has been suggested that the domain-final empty nucleus is governed by default. This might be a matter of parametric variation (see, e.g., Passino 2009). In Czech, it has been previously argued to be true (Ziková 2007). That is why the word-final V slot in the representation of *svit* can stay empty without causing any problems (as depicted by the vertical arrow):

(311) 'SHINE', NOM, FINAL V SLOT GOVERNED BY DEFAULT

C	V	C	V	C	V
|		|	|	|	
s		v	i	t	

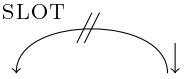
Building on these notions, our analysis will go as follows. The root for the non-palatalized paradigms comes fully linked, while the suffixes come with a CV but are not linked to it. See it on the example of the genitive of *jazyk* 'tongue' (where the genitive suffix is the element -A):

(312) 'TONGUE' + SUFFIX -A COMBINED, GEN

C	V	C	V	C	V	C	V
|	|	|	|	|			
j	a	z	y	k			A

The domain-final V is governed by default. Being empty, however, prevents it from governing the root-final empty V:

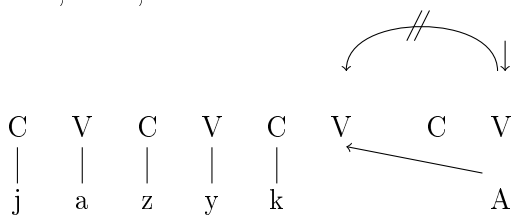
(313) 'TONGUE', NOM, FINAL V SLOT GOVERNED BY DEFAULT, NOT BEING ABLE TO GOVERN THE PREVIOUS V SLOT



C	V	C	V	C	V	C	V
|	|	|	|	|			
j	a	z	y	k			A

Because the root-final V cannot be left ungoverned, the floating A links to that position to save the structure:

(314) 'TONGUE', NOM, SUFFIX -A ATTACHES TO SAVE THE FAILED GOVERNMENT



The above will allow us to account for all the phenomena that will be worked out in the following sections. In the end, we will manage to incorporate three more standardly recognized paradigms into our system.

The following section zooms in on the argumentation for the above account, plus we add the floating I into the picture. As a first step, we look into how floating melody gets realized. We turn to the Czech e-zero alternation analysis to see how previous literature treated floating melody in that context.

4.3.1 The root and suffix interaction

In Czech, certain roots show a root alternation between the nominative/accusative on one side and the rest of the cases on the other side. See it below on the noun *kolek* 'duty stamp':[35]

(315) SURFACE FORMS OF 'DUTY STAMP', ROOT ALTERNATION COLOR-CODED

	'duty stamp'
NOM	kolek
ACC	kolek
GEN	kolk-u
LOC	kolk-u
DAT	kolk-u
INSTR	kolk-em

While in the nominative and accusative, the root form is *kolek*, everywhere else, it is *kolk*.

This is unlikely to be an epenthesis. Where *kolek* 'duty stamp' has the alternating *-e-*, other, phonologically (and morphologically) very similar words, *folk* 'folk (music)' and *úlek* 'scare' do not:

[35]The e-zero alternation appears with some variation across all Slavic languages and has been discussed in Slavic literature many times before. Cf. Scheer (2005), (2006), who highlights that the classic generative analysis based on yers points in the same direction as the analysis of the empty nuclei in Government Phonology, making it a stronger case for both.

(316) SURFACE FORMS OF 'FOLK (MUSIC)', 'DUTY STAMP' AND 'SCARE', ROOT ALTERNATION COLOR-CODED

	'folk (music)'	'duty stamp'	'scare'
NOM	folk	kolek	úlek
ACC	folk	kolek	úlek
GEN	folk-u	kolk-u	úlek-u
LOC	folk-u	kolk-u	úlek-u
DAT	folk-u	kolk-u	úlek-u
INSTR	folk-em	kolk-em	úlek-em

All three roots have the same phonological environment of the word-final *lk* consonants and belong to the same paradigm (the current size XS).

The roots *folk* 'folk (music)' and *kolek* 'duty stamp' have the same bisyllabic genitive that does not contain *-e-* in the middle of the *lk* cluster. However, it is impossible to predict the nominatives from their genitive form. The root *folk* does not have *-e-* even in the nominative, while *kolek* does. The epenthesis story would have a hard time explaining the difference.

Vice versa, it is also not possible to predict the genitive based on the nominative. Both *kolek* 'duty stamp' and *úlek* 'scare' have *-e-* in the consonant cluster in the nominative. Only *úlek* keeps it throughout the rest of the cases, though. The root *úlek* also disproves a potential hypothesis that the *-u* genitives always have to be bisyllabic, and that is what drives the *-e-* insertion.[36]

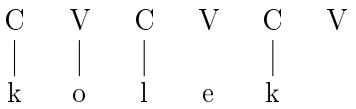
In Czech, the e-zero alternation has been described by so-called Havlík's Law (Havlík 1889), which is based on the diachronic development of yer vocalization.[37] Scheer (2005) reformulates Havlík's Law:

(317) HAVLÍK'S LAW REFORMULATED, Scheer (2005)

> "... given a sequence of consecutive empty nuclei in Common Slavic, every other empty nucleus is vocalized, counting from the right edge."

In strict CV, the yers are exchanged for nuclei with a floating *-e-* (Scheer 2005, Ziková 2007, Scheer and Ziková 2009). Following Ziková 2007, this is how *kolek* 'duty stamp' is stored in the lexicon in her analysis:

(318) 'DUTY STAMP', NOM, FLOATING -E-

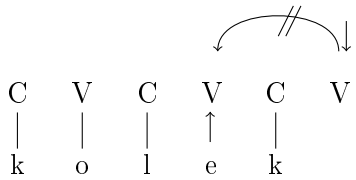


[36] Particularly nice is a comparison of two Czech homonyms – both pronounced as *úlek* in the nominative. One means 'scare', the other 'beehive (diminutive)'. As shown in the text, the one that means 'scare' keeps *-e-* throughout the whole paradigm. The one that means 'beehive (dim.)' has the same pattern as *kolek* 'duty stamp', i.e., with the alternating *-e-*.

[37] Havlík's Law does not apply in all Slavic languages. Cf. Bethin (1998) for an overall picture.

In the nominative (and accusative), *kolek* 'duty stamp' does not take any suffix. If the final empty nucleus is always governed, as traditionally assumed, it cannot govern the preceding vocalic slot. That means no one can govern the alternation site nucleus. Ziková concludes that if the alternation site stays ungoverned, the -*e*- has to attach:

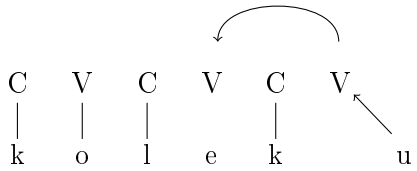
(319) 'DUTY STAMP', NOM, FLOATING -E- ATTACHES



That is how we get the nominative/accusative form with -*e*- surfacing, *kolek*.

In the rest of the cases, however, there is an extra suffix. So, e.g., the genitive singular form *kolk-u* is treated in the following way (Ziková 2007; to be amended later):

(320) 'DUTY STAMP', GEN, FLOAING -E- DOES NOT ATTACH



The floating genitive -*u* attaches to the root-final nucleus. The now filled nucleus can govern the alternation site, as shown by the arrow on top. Since the alternation site is governed, there is no need for -*e*- to attach.

The analysis builds on the ideas that:

1. the domain-final nucleus is always governed, and that
2. the floating melody links when a nucleus is not governed.

We will adopt both of these assumptions and apply them meticulously in all the other phenomena we tackle in the remainder of this section.

Let us now turn back to the phonological representation of the suffix. There are three logical options, and we show this on the genitive element -A.

In (321), we see the -A coming without any skeleton. In (322), the same -A comes attached to the skeleton. Finally, in (323), the lexical entry for -A brings in a skeleton but is not connected to it:[38]

[38] In this part, we will keep alternating between two roots (*jazyk* 'tongue' and *kolek* 'duty stamp') to point to different phenomena. To avoid confusion, keep in mind that while they are both non-palatalized, they belong to different root sizes. That is the reason for their different genitive suffixes – *jazyk* takes -A, and *kolek* takes -U. That being said, we assume that both have the same phonological representation.

(321)

A

(322)

C V
|
A

(323)

C V

A

We will now slowly go through the argumentation for why only the last fits Czech facts, given the assumptions above.

Imagine the first option. Combining the XL-sized root *jazyk* 'tongue' with the floating suffix would result in the structure below:

(324) 'TONGUE', GEN, MOCK VERSION, SUFFIX COMES WITHOUT A SKELETON

C V C V C V
| | | | |
j a z y k A

While this is the option that Ziková (2007)'s analysis chose for the suffixes, we discard it. One of our reasons is that it leads to disparate treatment of floating melody. Remember that one of Ziková (2007)'s conclusions was that the floating alternating *-e-* only connects when the alternation slot is not governed. However, the suffix was also a floating melody. So if the domain-final nucleus is governed by default, there is no reason for the suffix to attach:

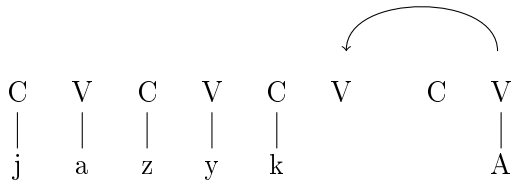
(325) 'TONGUE', GEN, MOCK VERSION, SUFFIX COMES WITHOUT A SKELETON

↓
C V C V C V
| | | | |
j a z y k A

While it is conceptually possible that there are different types of floating melody (with different linking patterns), we do not currently see the need for such distinction here. So for the moment, we discard this option – and in a later section dedicated to the paradigms with a floating consonant, we will show another reason we do not pursue having fully floating suffixes.

The second option for the phonological representation of the suffix was the idea that the suffixal element comes attached to its own CV (recall (322)). On the *jazyk* 'tongue' example, it would look as follows:

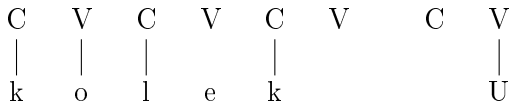
(326) 'TONGUE', GEN, MOCK VERSION, SUFFIX COMES ATTACHED TO A SKELETON



For the example above, it works. The word-final nucleus is filled and therefore can govern the preceding empty root-final nucleus.

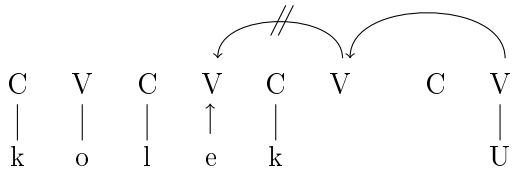
The reason why we discard this option comes back to the e-zero alternation. In the genitive, *kolek* 'duty stamp' combines with the element -U:

(327) 'DUTY STAMP', GEN, MOCK VERSION, SUFFIX COMES ATTACHED TO A SKELETON



The suffixal nucleus governs the empty root-final nucleus. However, being empty, the root-final nucleus can no longer govern the alternation site inside the root. In turn, if the alternation site is not governed, we would expect the *-e-* to link, giving us the (incorrect) form *kolek-u*:

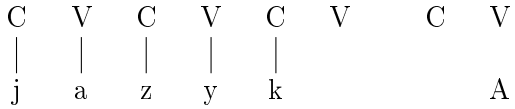
(328) 'DUTY STAMP', GEN, MOCK VERSION, SUFFIX COMES ATTACHED TO A SKELETON



Based on this, we conclude that the suffix does not bring its own CV in the way depicted above.

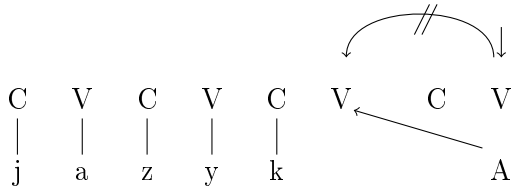
Finally, the third possibility was that the suffixal element comes with a CV but is not attached to it (recall (323)). This would give us the structure below (coming back to the example *jazyk* 'tongue'):

(329) 'TONGUE', GEN, SUFFIX COMES WITH A SKELETON BUT NOT ATTACHED



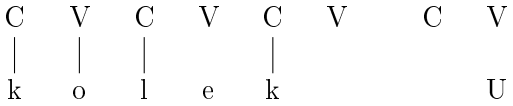
The word-final nucleus is governed by default. As such, it cannot govern the empty root-final nucleus. Since all nuclei must be either filled or properly governed, this triggers the floating melody to connect to the ungoverned spot, and we get the correct form, *jazyk-a*:

(330) 'TONGUE', GEN, SUFFIX COMES WITH A SKELETON BUT NOT ATTACHED



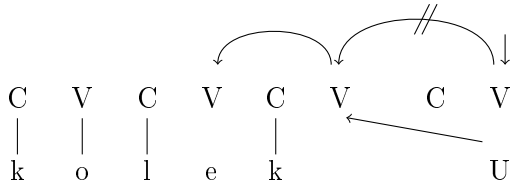
The same suffixal phonological representation works for the e-zero alternations. Consider the example of the genitive of *kolek* 'duty stamp' again:

(331) 'DUTY STAMP', GEN, SUFFIX COMES WITH A SKELETON BUT NOT ATTACHED



The word-final nucleus is governed by default. That means it cannot govern the empty root-final nucleus. The ungoverned root-final nucleus triggers the linking of the element -U. Now the root-final nucleus is filled and can therefore govern the alternation site. So we correctly get the genitive form *kolk-u* (and not kolek-u):

(332) 'DUTY STAMP', GEN, SUFFIX COMES WITH A SKELETON BUT NOT ATTACHED

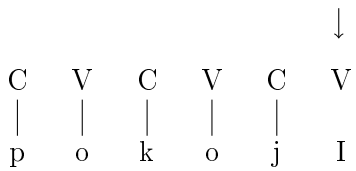


Finally, the same logic will also work for the palatalized paradigms. Take the root *pokoj* (XL-sized, a counterpart to *jazyk* 'tongue'):

(333) 'ROOM', NOM, SUFFIX COMES WITH A SKELETON BUT NOT ATTACHED

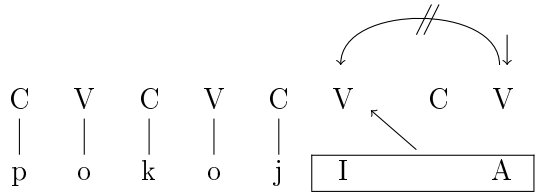
All the palatalized roots have a floating palatalizer at the end. In the nominative and accusative, there is no suffix. The domain-final nucleus is governed, nothing prompts the I to link, and we only get *pokoj* (and not *pokoji*):

(334) 'ROOM', NOM, SUFFIX COMES WITH A SKELETON BUT NOT ATTACHED



In the genitive, however, the syntax also feeds PF the suffixal -A:

(335) 'ROOM', GEN, SUFFIX COMES WITH A SKELETON BUT NOT ATTACHED



The empty nucleus brought by the suffix is governed by default and remains empty. As such, it cannot properly govern the preceding empty nucleus, which must be then filled – and we know it is filled since there is an audible ending in the genitive, unlike in the nominative.

Let us now address the question of which of the floating elements is associated to the empty nucleus. The result that we see on the surface is that the root *pokoj* is followed by an *-e*; therefore, both elements must link. However, from the perspective of government, any of them would do; linking both seems superfluous.

Based on this reasoning, we conclude that the two elements merge into a single segment, *-e*, before linking takes place, and then are linked jointly to the skeleton in a single step, as depicted in the graph in (335) through the rectangle.

The entire logic and all the operations align with how the CV theory works, and it accounts for the Czech data well. If correct, it also offers support for some specifics about the algorithm, suggesting that floating elements group before linking happens.

Note that there is still the question left of why the palatalized paradigms not only have a palatalized set of suffixes – for which we have just proposed an answer – but also why their root-final consonants mostly tend to be palatalized as well. We will not go into this issue in this work. Generally, however, the line of argumentation will probably have to follow along the lines of the floating I doing a dual job, influencing both the suffix and the root.

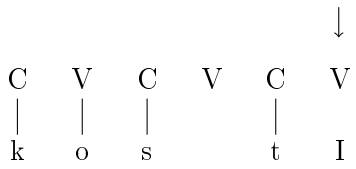
This kind of approach can shed some light on the curious lack of palatalization in the nominative and accusative of the paradigm *kost* 'bone' (the transcription below is rewritten quasi-phonologically):

(336) SURFACE FORMS OF 'BONE'

	'bone'
NOM	kost
ACC	kost
GEN	kosť-i
LOC	kosť-i
DAT	kosť-i
INSTR	kosť-í

Given the suffixes from the palatalized set, the first natural assumption is that there is the floating I at the end of the root. That would give us the following CV structure in the nominative:

(337) 'ROOM', NOM, SUFFIX COMES WITH A SKELETON BUT NOT ATTACHED



If we assume that the floating I also influences the root palatalization, then this works well because in the nominative (and accusative) of *kost* 'bone', the floating I does not attach because, first, *kost* is big enough not to need a suffix in these two cases, and second, the domain-final V slot is governed by default.

The details will have to be more complex, though, because even if it is prevalent, not every root with a palatalized final consonant takes the palatalized set of suffixes (and vice versa). An example of such a noun would be *gejš-a* 'geisha', which takes non-palatalized suffixes, e.g., *-a* in the nominative, despite the root ending in the palatalized consonant *-š*.

That being said, we leave the details of this issue for future work as it is tangential to our immediate goals.

As a recapitulation, the above analysis suggests that the Czech case suffixes come from the lexicon with skeleton but are not linked to it. The approach still keeps most parts of the traditional analysis of the e-zero alternations but also allows us to unify the behavior of floating elements in both cases presented in this section – in the e-zero alternations and the case suffixes.

In the following two sections, we tackle two more phenomena: floating consonants and paradigms with the stem ending in a long vowel. We will see that they also support the phonological representation of suffixes suggested above.

4.3.2 The paradigms with floating consonants

The paradigm *kuře* 'chicken' has the palatalized set of suffixes. They are the same as in the XL-sized *pokoj* 'room':[39]

(338) SURFACE FORMS OF 'CHICKEN' AND 'ROOM'

	'chicken' (N)	'room' (Mi)
NOM	kuře	pokoj
ACC	kuře	pokoj
GEN	kuře-t-**e**	pokoj-**e**
LOC	kuře-t-**i**	pokoj-**i**
DAT	kuře-t-**i**	pokoj-**i**
INSTR	kuře-t-**em**	pokoj-**em**

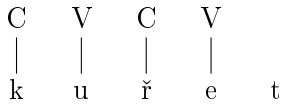
[39] It could also be compared to the M-sized paradigm *moř-e* 'sea'. We would then need to explain how come the -e disappears in the NOM/ACC of *kuře*. Since the stem stays *kuře* (as opposed to *kuř*) throughout the singular, there would be an *-e* missing. Whether, in the end, the *kuře* paradigm goes with the XL- or M-sized group, it does not pose particular problems for our analysis.

Where the paradigm differs from *pokoj* 'room' is the alternating *-t-* in the genitive through instrumental – only appearing when it is not word-final.[40]

In Ziková and Faltýnková (2020), the authors suggest an analysis for the alternating *-t-*. It is more complex than what we show here because, besides the alternating *-t-*, they also deal with allomorphy across the number within the paradigm. For our current purpose, we will do with a simplified version of their proposal.

Their idea is that the *-t-* is a floating consonant attached to the root (in the same sense as our floating I, or French liaison – cf. Encrevé 1988 cited through Newell 2017). The *kuře* 'chicken' root comes from the lexicon in the form below[41]:

(339) 'CHICKEN', NOM, FLOATING -T-



Since no suffix is added in the nominative and accusative, the above is the final form in these two cases. The floating *-t-* has nowhere to attach, and so on the surface, we only see *kuře*.

In the rest of the cases, there is an additional suffix, e.g., *-e* in the genitive. Ziková and Faltýnková (2020)'s analysis suggests that this suffix comes attached to a CV (to be amended later):

(340) 'CHICKEN', GEN, SUFFIX -E COMES ATTACHED TO A SKELETON

Thanks to the fact that the suffix only occupies the V slot, there is now an empty C slot for the *-t-* to attach to:

(341) 'CHICKEN', FLOATING -T- ATTACHES TO AN EMPTY C SLOT

[40] The XL-paradigm is not the only paradigm exhibiting this property. While it is most common there, the same zero-consonant alternation can be seen, e.g., in *drama* 'drama' (GEN. *drama-t-u*), which belongs to the XS-sized group represented by *hrad* 'castle'. A few masculine animate nouns like *hrabě* 'earl' (a group we have yet to discuss) show the same alternation and, in fact, almost the same endings as *kuře*. They differ in one curious aspect – they take masculine agreement in the singular but neuter agreement in the plural. We do not delve into this phenomenon.

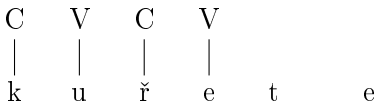
[41] Ziková and Faltýnková (2020) separate the *-e(t)-* from the root and treat it as a stem-building affix. We do not delve into that because the distinction would lead us away from the topic at hand. As far as we can tell, our analysis is reconcilable with the full extent of their views.

This is a straightforward way to account for the consonant-zero alternation. If the analysis is correct, it makes *kuře* 'chicken' a special instance of the XL-sized paradigm (represented by the noun *pokoj* 'room'). They only differ in that the *kuře*-type has a floating *-t-* at the end of the stem.

The above approach stands on the idea that the suffixes bring their own skeleton – and through those, the floating *-t-* has somewhere to link. Notably, the analysis cannot work if the suffixes do not bring the skeleton. This supports our conclusion from the previous section that the case suffixes in Czech cannot be fully floating.

In (342), we explain why the skeleton is necessary for stems like *kuře*. If both the *-t-* and the suffix are floating, we get the following picture in the genitive:

(342) 'CHICKEN', GEN, BOTH -T- AND -E ARE FLOATING



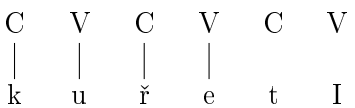
There is simply no skeleton to which the melody can attach. We would always get the form *kuře*. Hence the suffix must come with its own CV.

At the same time, in the previous section, we argued for the suffix coming with a CV but not linked to it, which is in contrast to what Ziková and Faltýnková (2020) suggest for *kuře* 'chicken'. So here, we will try to unify the account for the paradigm with what we proposed in the before.

There are three changes to Ziková and Faltýnková's analysis that need to take place. The first is about the phonological representation of the suffix. Instead of coming linked to a CV, it comes with a CV but is not connected to it. The second change is that the stem of *kuře*-like nouns has an extra empty CV at the end. The third is that the floating *-t-* only attaches when it finds itself in an intervocalic position.

The lexical entry of our proposal looks as follows:

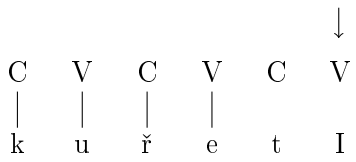
(343) 'CHICKEN', GEN, BOTH -T- AND -I FLOATING, ROOT HAVING AN EXTRA CV



The shared melody that we see in all case forms (i.e., *kuře*) comes linked to the skeleton. The alternating *-t-* is floating, just as in Ziková and Faltýnková (2020)'s account. Since the paradigm is palatalized, it also contains the floating I. The extra CV at the end is new.

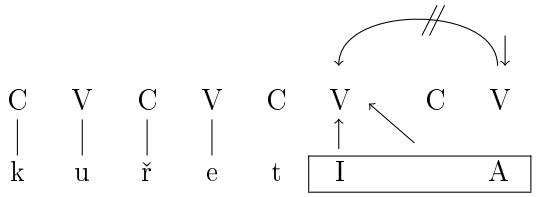
There is no suffix in the nominative or accusative, and the only empty nucleus is domain-final and hence governed by default. The floating *-t-* is not in an intervocalic position and stays unlinked. Consequently, the pronounced melody will correctly be *kuře*:

(344) 'CHICKEN', NOM, FINAL V SLOT GOVERNED



In the genitive, the PF also receives the lexical entry for -A:

(345) 'CHICKEN', GEN, FLOATING ELEMENTS ATTACH TO THE EMPTY ROOT-FINAL V SLOT



The empty word-final nucleus is governed by default, which means that the empty stem-final nucleus is not governed. The two elements, I and A, group and link to the stem-final nucleus to save it. The *-t-* also links to the skeleton because the C slot with a floating melody finds itself in an intervocalic position.

We see the process of linking the *-t-* melody to the consonantal slot to be an instance of hiatus resolution. We assume that the algorithm behind it will need to be more complex, but since it is somewhat orthogonal to the main focus of this work, we leave the topic aside for later research.

The above account allows us to unify the phonological shape of suffixes for the *kuře* 'chicken' paradigm with the e-epenthesis and the floating I discussed in the previous section. At the same time, it offers the first attempt to unify the reasons for linking between vocalic and consonantal floating elements – both connect to save the structure.

Kuře 'chicken' is traditionally considered a separate paradigm (Kodýtek 2010). While it would be certainly possible to incorporate it in our system in other ways, the way suggested above simplifies the workload of the lexicon by unifying the suffixes with the suffixes of the XL-sized paradigm *pokoj* 'room'. Furthermore, other paradigms with the floating *-t-* can be incorporated in the same manner. Only their phonological representation makes them seem different in certain aspects.

In the next section, we look at the paradigms like *zelí* 'cabbage', which end in a long vowel. We unify their account with what has been proposed up to now.

4.3.3 The paradigms ending in a long vowel

There are paradigms in Czech whose stem ends in a long vowel. One such example is the paradigm *zelí* 'cabbage':

(346) SURFACE FORMS OF 'CABBAGE'

	'cabbage' (N)
NOM	zelí
ACC	zelí
GEN	zelí
LOC	zelí
DAT	zelí
INSTR	zelí-m

Besides the stem ending in a long vowel, which itself is atypical for Czech, the paradigm also shows massive syncretisms, with only the instrumental having any clear suffix. Furthermore, the instrumental suffix is consonant-initial, which is also unique for the Czech nominal domain. Even the suffixes that do contain a consonant (like the masculine/neuter singular *-em*) always tend to start with a vowel.

Ziková (2007) argues that the *-í* is not a case marker but a stem-building affix. The main reason is the overwhelming and unprecedented syncretisms.[42] Her analysis stands on the idea that the paradigm (which she represents by the root *dělání* 'making') takes entirely regular suffixes (in line with those we have seen in other paradigms up to now). However, due to the long *-í*, they have no way of surfacing.

Given that they do not surface, we cannot guess what they are. Since all the surfacing Czech case suffixes in nouns are standardly vowels (or vowel-initial), Ziková uses a V to represent that these are likely to be vowels as well:

(347) SURFACE FORMS OF 'CABBAGE', MYSTERIOUS SUFFIXES

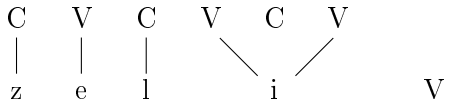
	'cabbage'
NOM	zelí-V
ACC	zelí-V
GEN	zelí-V
LOC	zelí-V
DAT	zelí-V
INSTR	zelí-m

Note that Ziková does not add the mysterious vowel in the instrumental. The reason is that it is difficult to find any clear-cut argument for whether it is there (but the opposite is also true). We are going to go ahead and suggest it there, too, because there are no masculines or neuters that take a different instrumental suffix than *-em*. Therefore, we think it fair to assume that *zelí* 'cabbage' is not unique. If that is true, then the mysterious vowel in the instrumental of *zelí* is *-e-*.

In Ziková's (2007) analysis, the suffixes are floating:

[42] An example of an unprecedented syncretism is the syncretism between "the locative and the nominative singular as well as the genitive and the nominative plural" (Ziková 2007). These are not attested in the Czech declension, and they also do not exist in any other Slavic language.

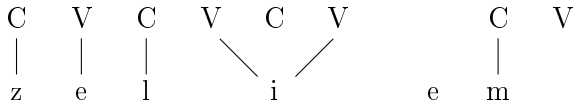
(348) 'CABBAGE', NOM, LONG -I TAKES TWO V SLOTS



As such, they have nowhere to link (and no reason to) since the root-final V is filled by the long *-í*. Due to that, the suffix stays floating.

For the instrumental *-em*, Ziková suggests that it comes with a CV, but only the *-m* is linked, while the *-e-* is floating. Combining it with the root *zelí* would look as follows:

(349) 'CABBAGE', THE INSTRUMENTAL -M COMES ATTACHED TO A C SLOT

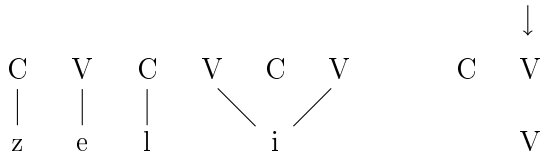


Just as in the other cases, the suffixal vowel has nowhere to attach, and we get the correct instrumental form, *zelí-m*.[43]

We adopt the account of the instrumental *-(e)m* in its fullness and extend it to the roots like *zelí* 'cabbage', as it works for all of the phenomena raised in the past few sections. Where we differ from Ziková's (2007) analysis is the phonological shape of the rest of the cases. To unify even this phenomenon with the rest, we again suggest that other than instrumental suffixes come with an additional CV but are not connected to it.

Here is an example of the genitive for the *zelí* 'cabbage' paradigm:

(350) 'CABBAGE', THE FINAL V SLOT IS GOVERNED



The word-final nucleus is governed by default. There is no ungoverned nucleus in the structure, and hence the floating V has no reason to attach.

[43] If a counter-argument is found for *zelí* only truly having *-m* in the instrumental, it is straightforward how to adjust the account. We would delete the *-e-* from the lexical entry.
There is also a logical possibility of the *-e-* in the instrumental *-em* to be epenthetic. It would surface with roots like *pán* to save the structure from forms like **pánm*, which are not valid in Czech. The root *zelí* does not need an epenthetic *-e-* because the cluster *-lím* works well. We do not pursue this line of inquiry because there are scarce noun roots in Czech, that end in a (short) vowel, like *kaka-o* 'cocoa' ending in *-a-*, and these still take the *-em* suffix (*kaka-em*), even though the (non-existent) form **kakam* would be perfectly sound for the Czech phonological system.

Due to the lack of overt suffixes, it is impossible to say what root size *zelí* belongs to (or if it is a group on its own). The suffix choice and syncretisms are the primary tools to identify root sizes, and without them, not much more can be said.

A question follows from the analysis above. Do we ever see a feminine paradigm that would end in a long vowel and therefore lend itself to the same account? In such a case, what would be predicted would be a complete syncretism from the nominative through to the instrumental because feminines tend to take purely vocalic instrumentals.

The feminine paradigm *paní* 'lady' seems to be precisely that. Here are its surface forms:

(351) SURFACE FORMS OF 'LADY'

	'lady' (F)
NOM	paní
ACC	paní
GEN	paní
LOC	paní
DAT	paní
INSTR	paní

Just as in the *zelí* 'cabbage' paradigm, the nominative through dative only ever shows the long-vowel stem without any suffixes. The instrumental differs from the one we saw with *zelí*. Where *zelí* had the form with *-m* (*zelí-m*), *paní* still only shows the stem.

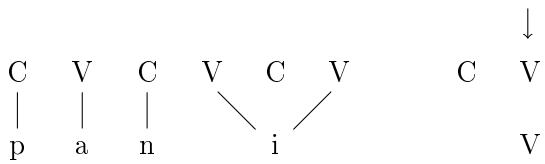
This is what the analysis predicts. All the suffixal vowels were argued to come not linked to the skeleton. It was only the instrumental consonant *-m* that came linked from the lexicon. Since the feminines only show vocalic suffixes, it is fair to assume that even the instrumental V in *paní* 'lady' is floating and hence never surfaces. Compare the suggested structure for the masculine (352) and the feminine (353) instrumental:

(352)

(353)

Below, we show the whole picture of the *paní* 'lady' instrumental:

(354) 'LADY', MYSTERIOUS INSTRUMENTAL VOWEL



The word-final nucleus is governed by default, and there is no other empty nucleus. Hence the floating V never gets the chance to surface.

If the above is correct, then *paní* 'lady' works on a similar principle as *zelí* 'cabbage', each representing their respective genders. This does not mean they have to be of the

same root size. As pointed out with *zelí*, there is no way for us to know its root size, and the same is true for *paní*.

To sum up this section, we have looked at three different PF issues: e-zero alternations (*kotel* 'boiler'), consonant-zero alternations (*kuře* 'chicken'), and paradigms with stem-final long vowel (*zelí* 'cabbage').

Different analyses have been suggested for these phenomena before and have not always been fully compatible. We have suggested a way to put them all within the boundaries of a single account. The main idea is that the vocalic suffixes bring a CV from the lexicon but are not linked to it. In consonant-zero alternations, the roots also bring an extra CV at the end of their skeleton. Since the detached CV in suffixes can unify all the phenomena mentioned above, we take it as a sign of this analysis being on the right track.

The rest of the analysis follows the standard CV theory. We work with the strict CV approach, the empty root-final nuclei are always governed, and melody only attaches when triggered by the derivation.

Furthermore, we have incorporated three additional paradigms recognized in the traditional descriptions: *kuře* 'chicken', *zelí* 'cabbage' and *paní* 'lady'.

5 Root shape

5.1 Lexical entries with a complex left branch: the masculine animates 'sir' and 'man'

This section zooms in on a subset of paradigms we have been putting aside so far. We address two (surface) paradigms here – *pán* 'sir' and *muž* 'man'.[44] Both of these pose some challenges to an easy account. We offer three logically possible analyses, each looking at the paradigms from a slightly different perspective and using different tools. Ultimately, we opt for trees with a complex left branch (CLB, sometimes also referred to as movement-containing trees, or trees with a detachable left branch), which seems to capture the data the best.

In the nominal domain, Czech only makes animacy distinction within the masculine group. One well-known difference between the masculine animates and inanimates is that while the inanimates (and also most feminines and all neuters) have a NOM.SG = ACC.SG syncretism, the animates have an ACC.SG = GEN.SG one. Below, compare two animate-inanimate pairs of masculine paradigms. The pair in the table (355) is non-palatalized, the pair in the table (356) is palatalized):[45]

(355) SURFACE FORMS OF 'SIR' AND 'TONGUE'

	'sir' (Ma)	'tongue' (Mi)
NOM	pán	jazyk
ACC	pán-a	jazyk
GEN	pán-a	jazyk-a
LOC	pán-u	jazyk-u
DAT	pán-u	jazyk-u
INSTR	pán-em	jazyk-em

(356) SURFACE FORMS OF 'MAN' AND 'ROOM'

	'man' (Ma)	'room' (Mi)
NOM	muž	pokoj
ACC	muž-e	pokoj
GEN	muž-e	pokoj-e
LOC	muž-i	pokoj-i
DAT	muž-i	pokoj-i
INSTR	muž-em	pokoj-em

Note that all the endings besides the difference in the accusative mentioned above are the same. This syncretism contrast makes any simple account challenging – assuming

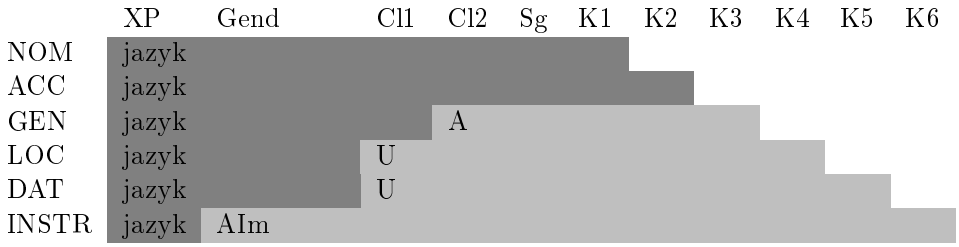
[44] There are two more traditionally recognized masculine animate paradigms, *soudc-e* 'judge' and *koleg-a* 'colleague'. We incorporated *soudc-e* in section 4.2.4. *Koleg-a* will be discussed in section 6.1, which is dedicated to paradigms ending in *-a* in the nominative.

[45] The paradigm *pán* 'sir' has a variation in the locative and dative. Besides *-u*, there is also a specifically animate suffix *-ovi*. To start the analysis, we stick with *-u* as the ending that unifies the paradigm with what we already know from previous sections. We will discuss *-ovi* later, in section 6.1.

we want to keep the suffixes syncretic across the animate and inanimate paradigms and not suggest that they are but accidental syncretisms despite looking the same.

To see the issue, let us repeat the lexicalization table with the account for the non-palatalized inanimate paradigm *jazyk* 'tongue':

(357) LEXICALIZATION TABLE OF 'TONGUE'



Based on this, we suggest the lexicalization table for the masculine animate *pán* 'sir' below. Given the widespread syncretism with *jazyk*, we want it to have identical lexical entries for its endings. We skip the differing accusative for the moment:

(358) LEXICALIZATION TABLE OF 'SIR' (TO BE REVISED)

Now, what happens in the accusative? Since, on the surface, we see *-a*, there are two obvious options. Either the *-a* is the same as in the genitive, or it is different. First, let us show how the account would go if the two -As were different. While it seems technologically simpler, we will ultimately discard it for an account that allows us to keep a simpler lexicon by unifying the two -As.

The idea that the -As are different works pretty straightforwardly. The root *pán* 'sir' spells out features up to K1 and then needs someone else to spell out K2. We could suggest a lexical entry -A to spell out only K2. As can be seen below, this -A (which we will call -A1) is different from the genitive -A (which we will call -A2):[46]

[46] The entry for -A1 could also start lower and spell out K1 + K2, and the account would work the same. It cannot go lower than that, though, as it would start to interact with other suffixal entries in unwanted ways.

(359) LEXICALIZATION TABLE OF 'SIR' (TO BE REVISED)

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	pán										
ACC	pán						A1				
GEN	pán			A2							
LOC	pán		U								
DAT	pán		U								
INSTR	pán	AIm									

Since the accusative -A1 cannot spell out more features above K2, the root is forced to backtrack in the genitive and finds -A2. From then on, the derivation continues the same as for the inanimate *jazyk* 'tongue'.

Suggesting the -A1 suffix does not cause any issues for the inanimate paradigm *jazyk* 'tongue' because that one has a root able to spell out features up to K2 and therefore does not interact with the -A1 suffix.

The -A1 entry also does not cause any issues for paradigms of a smaller root size because they never reach it. The current second biggest size, the M-sized group, has roots only growing to Cl2, so -A1 starting in K2 is never a candidate for them. Therefore the above account of adding an extra -A to the set of lexical entries works.

At the same time, we could wonder whether there is a way to unify -A1 and -A2 into one lexical entry. Below, we show two accounts that are more complex but make this possible.

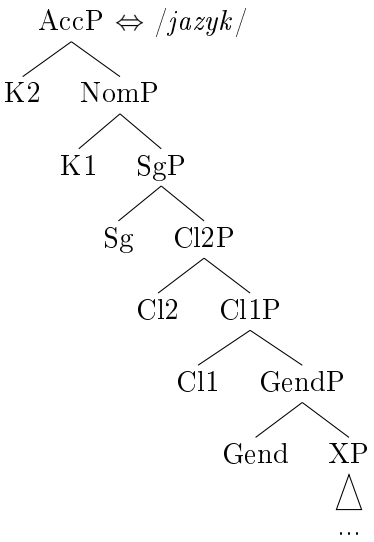
First, however, let us show why no *simple* account can unify the two -As and why we need to reach for more complicated tools. The relevant animate-inanimate pattern is repeated below:

(360) SURFACE FORMS OF 'SIR' AND 'TONGUE', NOM–GEN

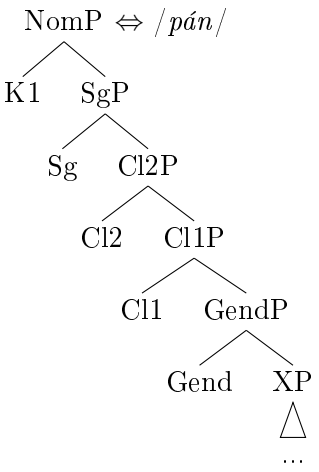
	'sir' (Ma)	'tongue' (Mi)
NOM	pán	jazyk
ACC	pán-a	jazyk
GEN	pán-a	jazyk-a

The pattern where one root reaches a particular suffix earlier and is, later on, joined by a different root is a pretty common one. We have suggested above that the difference between the two roots at hand is that while *jazyk* 'tongue' spells out all the features in the relevant functional sequence up to K2, the root for *pán* 'sir' is smaller and only spells out features up to K1:

(361) LEXICAL ENTRY OF 'TONGUE'

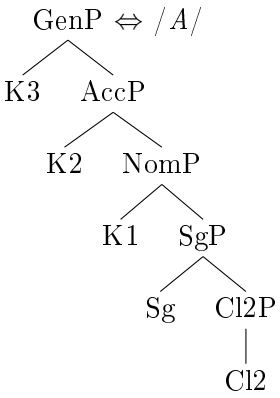


(362) LEXICAL ENTRY OF 'SIR'



We will keep the lexical entry for -A the same as we did so far:

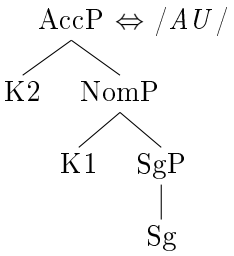
(363) LEXICAL ENTRY OF -A



With lexical entries above, we can derive the pattern in (360). Both paradigms need to backtrack to get -A. The roots like *pán* first backtrack in the accusative derivation, while the roots like *jazyk* only start to backtrack for -A in the genitive as they can spell out the accusative on their own.

However, this only works in a vacuum with just these two paradigms present. The issue is not with the pattern itself but with the interference of other lexical entries – namely, the entry for -AU, which we used in the nominative and accusative of the M-sized group (sak-o 'jacket'):

(364) LEXICAL ENTRY OF -AU



The entry for -AU has a higher foot (in Sg) than -A has (in Cl2P). So when *pán* 'sir' starts to backtrack in search of a suffix able to spell out K2, -AU will be the first found candidate – and there will be no reason to backtrack more to find -A. Hence, the form we would predict for the accusative of *pán* is *pán-o* (← *pán-AU*), which is incorrect.

We cannot just change the footing of neither *-o* nor *-a* to make this work. Due to how the *sak-o* 'jacket' paradigm looks like and how it interacts with the *chleb-a* 'bread' paradigm, -AU *has* to have a higher foot than -A (refer back to (78)). Hence, a simple approach to unifying the two -As cannot work.

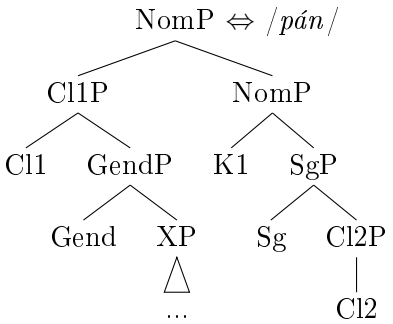
In the remainder of this section, we present two accounts that manage to unify the two -As in a more elaborate way. One uses complex left-branching trees, the other a more fine-grained case hierarchy.

The first account – the one we will ultimately want to champion – is based on the trees with a complex left branch (CLB; Blix 2021, Caha 2021a).

It builds on the idea of branching lexical entries and the fact that the standard Superset Principle allows syntax to use a subset of any entry. The DLB trees are helpful precisely in situations where we want to "skip" suffixes that would otherwise pop up.

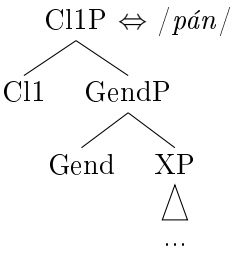
We suggest the following entry for the roots of the *pán* 'sir' paradigm – the root keeps the same size as in the previous account (growing up to NomP), but its shape is different. Given that it grows lower on the functional sequence than the XL-sized roots like *jazyk* 'tongue' but at the same time higher than the M-sized *sak-o* 'jacket', we will refer to them as the L-sized group:

(365) DLB LEXICAL ENTRY OF 'SIR' (L-SIZED)

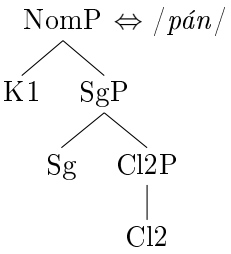


Due to the Superset Principle, the root will not only be able to spell out the whole structure but also either of the branches separately (or any of their subsets):

(366) DLB LEXICAL ENTRY OF 'SIR', LEFT BRANCH

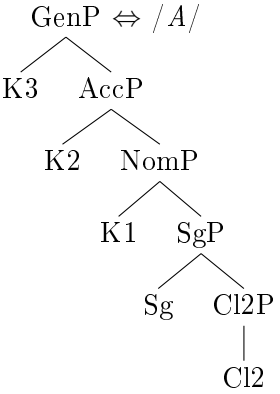


(367) DLB LEXICAL ENTRY OF 'SIR', RIGHT BRANCH

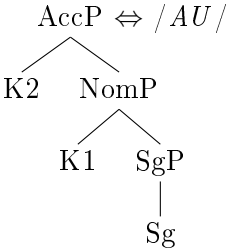


Before proceeding to show the derivation, let us repeat the two lexical entries for the suffixes in question: -A, which we want to get both in the accusative and genitive of *pán* 'sir', and -AU, which we want to avoid in the accusative of the same paradigm:

(368) LEXICAL ENTRY OF -A

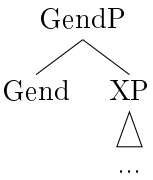


(369) LEXICAL ENTRY OF -AU

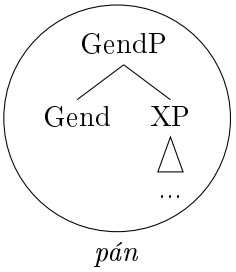


In the derivation, we first create XP (i.e., merge whatever features are in it) and merge it with Gend (370). The new GendP node is spelled out by the root (371) because a match is found in the subset of the left branch of the lexical entry for *pán*:

(370) DERIVATION OF 'SIR', NOM, INTERMEDIATE STEP

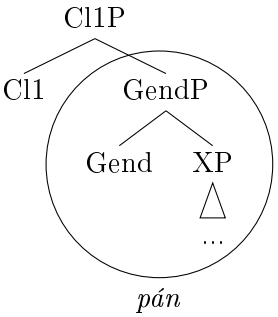


(371) DERIVATION OF 'SIR', NOM, INTERMEDIATE STEP

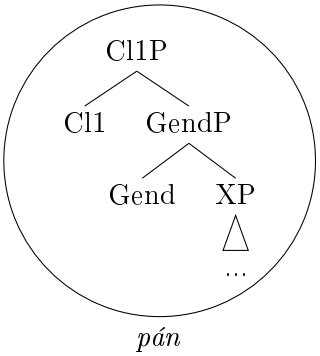


Next, we merge Cl1P (372), which is still spelled out by the entry for *pán* (373):

(372) DERIVATION OF 'SIR', NOM, INTERMEDIATE STEP

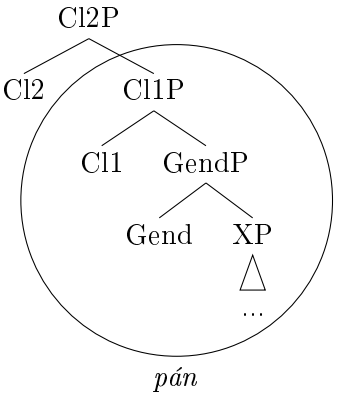


(373) DERIVATION OF 'SIR', NOM, INTERMEDIATE STEP



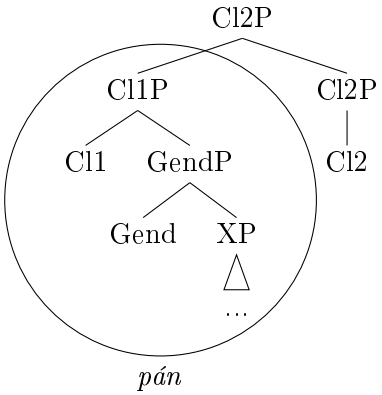
Then we merge Cl2P:

(374) DERIVATION OF 'SIR', NOM, INTERMEDIATE STEP



We try to match, but the lexical entry for *pán* is not able to match the structure because, in it, the Cl2P node does not contain Cl1P. The Spec movement does not apply here, so the complement movement gets triggered:

(375) DERIVATION OF 'SIR', NOM, INTERMEDIATE STEP

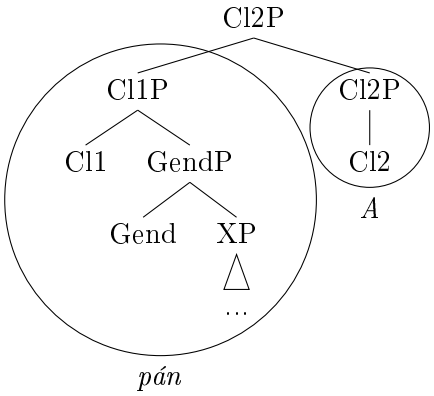


Syntax attempts to match the highest node but finds no candidate. While the left branch

is a perfect match for the left branch of the lexical entry for *pán*, and the Cl2P node on the right branch is a part of *pán*'s right branch as well, the Superset Principle is not satisfied because there is no proper containment. The right branch in (375) misses the singular and nominative features.

So the matching moves to the daughter nodes. As has been said, the left branch can still be spelled out by *pán*. The right branch on its own can now be spelled out by -A (refer back to (183)):[47]

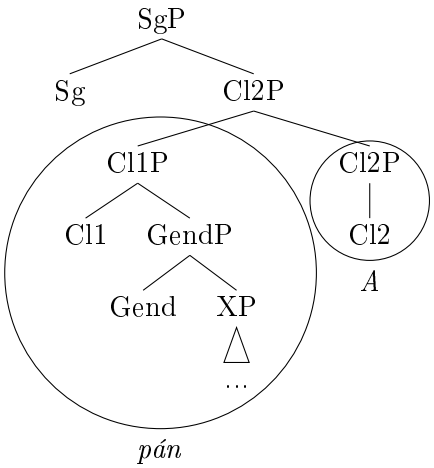
(376) DERIVATION OF 'SIR', NOM, INTERMEDIATE STEP



The crucial part here is that the complex nature of the lexical entry of *pán* allowed us to use just its left branch, resulting in the form *pán-a*.

Next, we merge the singular feature:

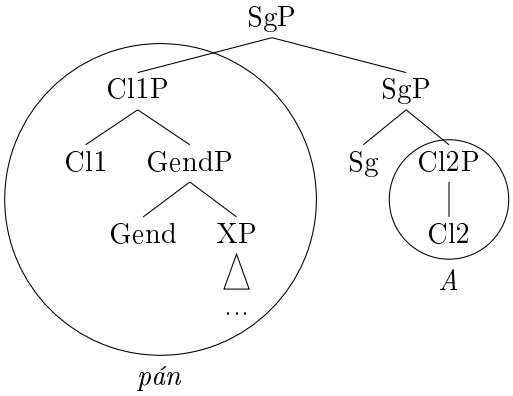
(377) DERIVATION OF 'SIR', NOM, INTERMEDIATE STEP



[47] Note that the root could spell out the right branch as well, leading to a doubling: *pán-pán*. This will always be a marked option because the complex lexical entries have a lot of superfluous features and therefore lose in the competition against simpler suffixal entries. However, since we see a doubling in some languages, this might be an intriguing theoretical possibility to account for it.

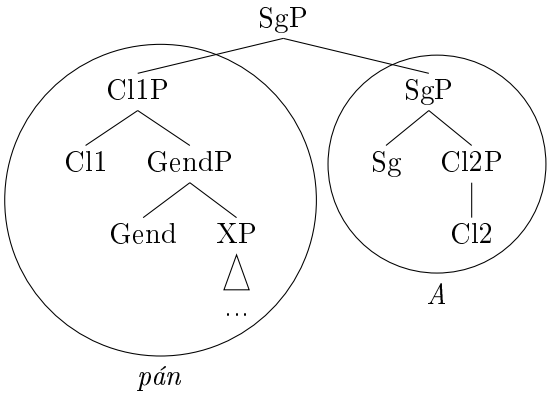
There is no entry able to spell the structure out as it is, so the Spec movement gets triggered:

(378) DERIVATION OF 'SIR', NOM, INTERMEDIATE STEP



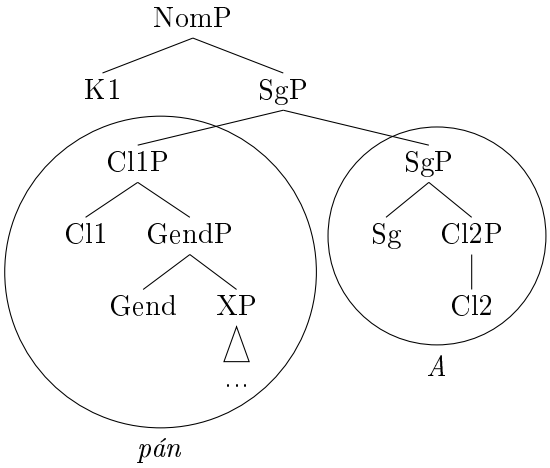
There is no match for the whole structure, but the left branch can still be spelled out as *pán* and the right branch as -A:

(379) DERIVATION OF 'SIR', NOM, INTERMEDIATE STEP



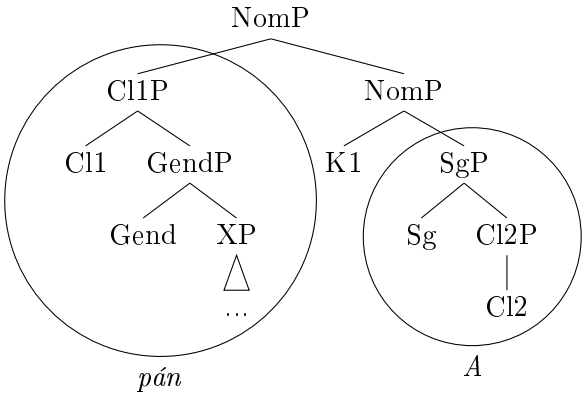
Finally, we merge K1 – the last feature needed for the nominative:

(380) DERIVATION OF 'SIR', NOM, INTERMEDIATE STEP



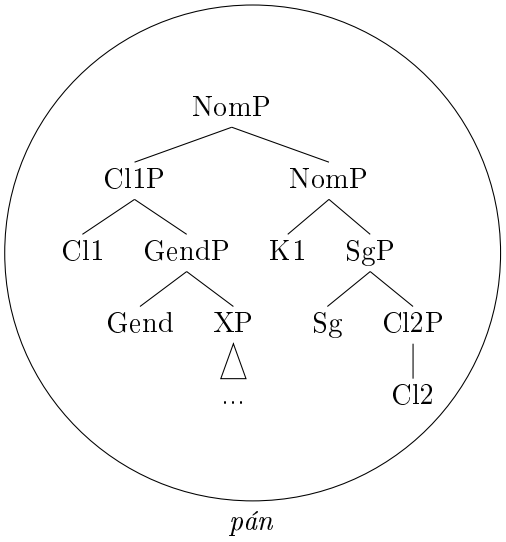
No lexical entry can spell the structure out, and the Spec movement gets triggered:

(381) DERIVATION OF 'SIR', NOM, INTERMEDIATE STEP



Since we added the two missing features, the singular and the nominative, the structure in its entirety is now a perfect match for the lexical entry of *pán*:

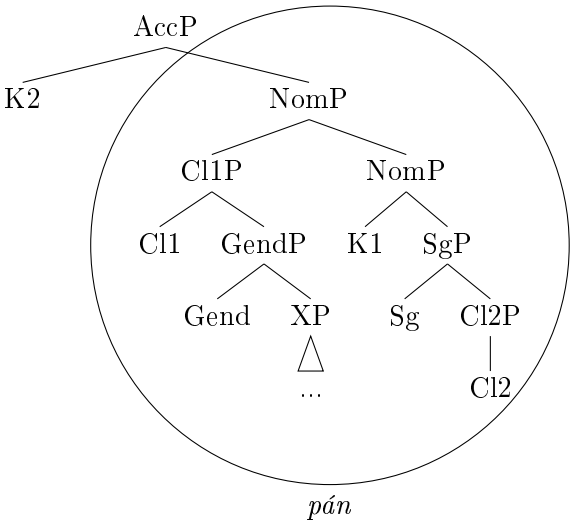
(382) DERIVATION OF 'SIR', NOM, FINAL FORM



That is how we get the nominative. Let us proceed to the accusative, which is the troublesome one to derive if we want to unify the accusative and genitive -A.

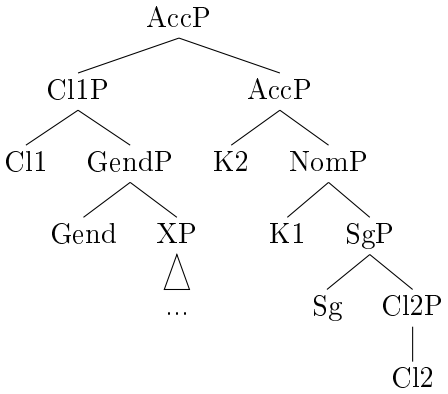
We merge K2 on top of the nominative structure:

(383) DERIVATION OF 'SIR', ACC, INTERMEDIATE STEP



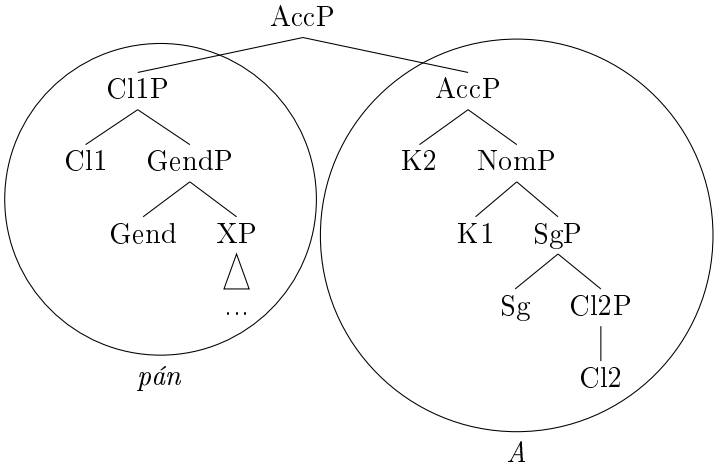
No entry can spell the structure out, so the Spec movement gets triggered, raising above K2:

(384) DERIVATION OF 'SIR', ACC, INTERMEDIATE STEP



The full entry of the root *pán* is no longer able to spell out the whole structure because it does not contain the K2 feature. Hence, the syntax attempts to spell out each branch separately. This brings us back to the situation we already saw – the left branch here matches the left branch of the entry for *pán*, while the right branch matches the entry for -A:

(385) DERIVATION OF 'SIR', ACC, FINAL FORM

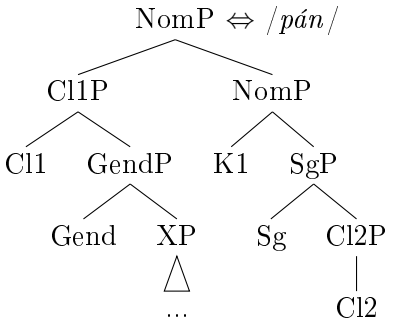


The derivation of the genitive will then go the same as in the accusative – we merge the next feature, K3, do the Spec movement and spell the resulting structure as *pán-a*.

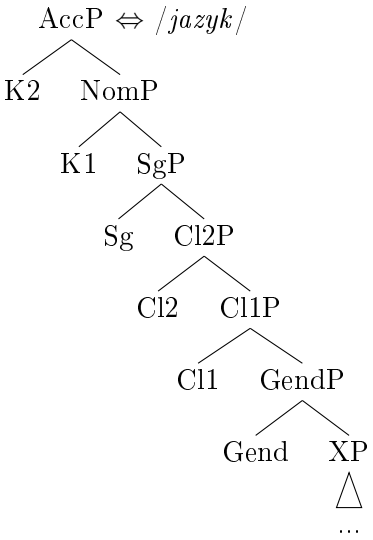
The account comes with a prediction about the nature of the GEN-ACC syncretism. While the syncretism pattern is traditionally understood as connected to animacy, the DLB account suggests the connection may be looser, possibly (synchronically) non-existent in Czech.

Notice that there is no point in the derivation where we reference any animacy feature. The only difference between the masculine animates and inanimates is the shape of their lexical entry (and in these two particular examples, in how many features they spell out):

(386) DLB LEXICAL ENTRY OF 'SIR'



(387) LEXICAL ENTRY OF 'TONGUE'



The masculine animate paradigms, like *pán* 'sir', have complex, branching entries, while the inanimates, like *jazyk* 'tongue', do not.

Since nothing prevents it, the account predicts there may be other nouns than masculine animates that can have the complex lexical entry as in (386). On the surface, this would result in having the GEN-ACC syncretism. This prediction turns out to be correct.

There is a small set of masculine inanimates that show the GEN-ACC syncretism (see, e.g., Šulc 2001 or Najbrtová 2013):[48]

(388) GEN-ACC SYNCRETISM OF MASC.INANIM 'SHOT'

a. Na stole stál **panák**
on table stood shot.NOM
'There was a shot (glass) on the table'

b. Dám si **panák-a**
have.1SG POSS shot-ACC
'I'll have a shot' (in drinking)

c. Bez **panák-a** nikam ne-jdu
without shot-GEN nowhere NEG-go.1SG
'I'm not going anywhere without a shot' (in drinking)

Using the "inanimate" NOM-ACC pattern with this noun – while not possible to completely exclude – does not seem to work that well:[49]

[48] Czech is not the only Slavic language with the unexpected GEN-ACC in the inanimates. It has also been noted, e.g., in Polish, Ukrainian and Slovak (Swan 1988, Pugh and Press 1999, Dvonč et al. 1966).

[49] While Internetová jazyková příručka allows for both forms, there are only the a-forms in the accusative in the Czech National Corpus (228 tokens, searched in the SyD tool, Cvrček and Vondřička 2011). All native speakers we polled had the same general intuition, preferring *-a* in the accusative. That being

(390) ?Dám si **panák**
take.1SG POSS shot.ACC
'I'll take a shot' (in drinking)

The group of inanimate nouns allowing for GEN-ACC syncretism does not seem to be semantically delimited. There are words from many different areas, e.g., drinks (rum 'rum'), mushrooms (hřib 'boletus'), sweets (indiánek 'whipped egg white mousse covered in chocolate with a sponge base'), organizations (skaut 'the Scout association'), or cars (Fiat), to name just a few (Internetová jazyková příručka).

This phenomenon, sometimes also referred to as facultative animacy, is rather challenging to account for as it is not clear why nouns that are clearly semantically inanimates would take what is supposed to be the animate syncretism pattern.

Things get even more interesting once we look at the possessive forms. Only (semantically) masculine animates can take *-ův/ova/ovo* suffixes in the possessives in the singular (the exact form depending on the agreement), e.g.:[50]

(391) **Petr-ov-a** hlav-a
Peter-POSS-FEM head-FEM
'Peter's head'

However, the animate possessive suffixes are unavailable to the above mentioned special inanimates like *panák* 'shot' (392-a), and one has to use the genitive construction (typical for inanimates) instead (392-b):

(392) a. ***panák-ov-o** dn-o
shot-POSS-NEUT bottom-NEUT
'the bottom of a shot (glass)'
b. dn-o **panák-u**
bottom-NEUT shot-GEN
'the bottom of a shot (glass)'

---

said, this sub-class is probably still internally differentiated as many inanimate nouns that allow for the GEN-ACC pattern (389-a) can still easily have the NOM-ACC pattern (389-b):

(389) GEN-ACC SYNCRETISM OF PANÁK 'SHOT'
a. Dám si **rum**
take.1SG POSS shot.NOM
'I'll take a shot' (in drinking)
b. Dám si **rum-a**
take.1SG POSS shot-ACC
'I'll take a shot' (in drinking)

Note that the accusative *-a* in the inanimates is often perceived as "expressive" in grammars (e.g., Havránek and Jedlička 1992), although see Šulc (2001) for an opposing view. Some of these forms are substandard and are more used in the spoken language. We are not aware of any in-depth research on the phenomenon in spoken Czech and therefore lack data to make any definite conclusions.

[50] We are putting aside a few technically inanimate nouns, like *sněhulák* 'snowman' (and others of humanoid shape), which are for all grammatical purposes animate.

There are a few semantically animate nouns that cannot take the possessive *-ův* suffix (Karlík and Veselovská 2017), e.g., neuters (*dítě* 'child' – **dítětova hlava* 'the child's head'). However, the inanimate GEN-ACC nouns do not seem to fit in any of the recognized groups. Furthermore, we can see a minimal example to suggest that it is not their phonology standing in the way. The noun *panák* can have two readings. We saw one above, where it meant 'shot (glass)'. The other is 'dummy, figurine'. When we use the noun with the second reading (which falls among animates again probably due to its humanoid shape), suddenly, the use of the *-ův* suffix seems to be grammatical:

(393) **panák-ov-a** hlav-a
dummy-POSS-FEM head-FEM
'dummy's head'

Based on these data, there are two things to consider. First, how come there are semantically inanimate nouns, like *panák* 'shot (glass)', that take the supposedly animate syncretism pattern of GEN-ACC. And second, how come these do not take the possessive animate ending and pair with the inanimates instead.

The DLB tree account shows a way to capture these data by proposing that the GEN-ACC syncretism is not actually about animacy. It is caused merely by the shape of lexical entries – the complex, branching entry naturally leads to the syncretism pattern.

The fact that it is predominantly animate masculines showing the GEN-ACC syncretism may be seen as a leftover of a historic change. The GEN-ACC syncretism for animates started to enter the Czech declension in the second half of the 12th century. Around the 15th/16th century, the pattern also expanded to animal names of the masculine gender (Pleskalová 2001). It is not clear to us when the special group of inanimate nouns like *panák* 'shot (glass)' acquired it.

One thing to be pointed out is that the DLB trees allow for a specific type of the ABA pattern. In fact, with how our lexical items are set up, we get a double-ABA. The table below shows individual steps of the derivation we just went through and their spellout:

(394) SPELLOUT STEPS OF THE 'SIR' DERIVATION IN THE DLB ACCOUNT, XP TO ACCP

node	spellout
XP/GendP/Cl1P	pán
Cl2P/SgP	pán-a
NomP	pán
AccP	pán-a

The above table showcases two types of ABA: A-Ax-A (counting from the first to the third line) and Ax-A-Ax (counting from the second to the fourth line). Several conditions constrain these patterns.

For the A-Ax-A pattern, the A needs to be the root. The lexical entry of the root needs to have a complex left branch, and it needs to be able to spell out all three cells

of the pattern. For the Ax-A-Ax pattern, the same rules apply, except that this time, it is the suffix that needs to be able to spell out all three cells.

Here, not only are all conditions met, but they chain behind each other, creating the double-ABA effect. One of the entries in question is an entry of a root, it has a complex left branch, and both the root and the *-a* suffix can spell out all three cells of the pattern. So while an ABA is derived, it is restricted by clear conditions and only happens with roots. We do not immediately see that it is an issue for the theory, but more data will need to be considered to take a stance on that.[51]

The ABA aside and coming back to the problematic derivation of the accusative *-a*, we derive the correct form *pán-a* thanks to two facts. First, we allow the entry for *pán* to spell out just the left branch (which is but the standard Superset Principle). Second, we set the entry up so that the right branch of *pán*'s lowest feature is the same lowest feature as in A. The entry for -AU is no longer interfering (as was the main catch before) because its foot is higher (in SgP) and will not be considered throughout this derivation.

All things considered, the DLB trees account is a promising way to incorporate masculine animates, *pán* 'sir' and *muž* 'man', while unifying the accusative and genitive -A.

That being said, Nanosyntax has a way to deal with the GEN-ACC syncretism, even if it turns out that it is better accounted for by tying it to animacy. Let us quickly sketch an account that focuses on a more fine-grained version of Caha (2009)'s case hierarchy, as suggested by Starke (2017).

In his paper, Starke looks at Spanish (among other data), which has two ways of expressing the accusative:

(395) SPANISH, Rodríguez-Mondoñedo (2009) through Starke (2017)

a. María quiere a un abogado [+animate, +specific]
Mary wants PREP a lawyer
'Mary wants a (specific) lawyer'

b. María quiere un abogado [+animate, –specific]
Mary wants a lawyer
'Mary wants a lawyer (any lawyer)'

The two sentences above only differ in the preposition *a*, which marks the animate + specific + accusative context. Starke argues that given the different morphosyntax, we should identify these as two different cases.

Depending on which of these two we call "accusative" – the bare noun or the PP – there is (or is not) an ABA pattern, a possible challenge for Caha's original hierarchy. If we choose to call "accusative" the bare noun, there is no ABA violation:

(396) SPANISH, Starke (2017)

ACC	Ø
GEN	de
DAT	a

[51] See Blix (in print) for a note on how the inverse pattern Ax-A can help derive a tripartite number-based system of nominal classification in Kipsigis.

However, if we choose to call "accusative" the PP form, we get the *ABA:

(397) SPANISH, Starke (2017)

ACC	a
GEN	de
DAT	a

Starke distinguishes the two ways of marking the accusative by calling the bare noun (395-b) a small accusative (sACC) and the PP (395-a) a big accusative (bACC).

Based on syncretism data, he suggests the following placement of the two accusatives in the case hierarchy:[52]

(398) Nom > sAcc > Gen > bAcc > Dat

This shows that the *ABA issue does not arise if we distinguish between a regular accusative and a special DOM case. Under this view, the big accusative and the dative are adjacent, and therefore their syncretism is not surprising:

(399) SPANISH

sACC	Ø
GEN	de
bACC	a
DAT	a

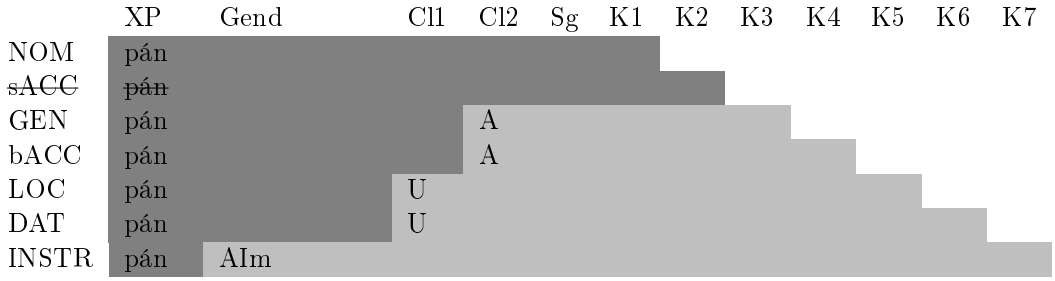
With this background, let us turn back to Czech. We could suggest that, like Spanish, Czech employs both the big and the small accusative, and their distribution is decided through a particular morphosyntactic context. The masculine inanimates (and the feminines and neuters) use the small accusative, while the masculine animates use the big accusative.

What this translates to, in terms of lexical entries, is keeping the masculine inanimates the same and adding a new case, bACC, for the masculine animates:[53]

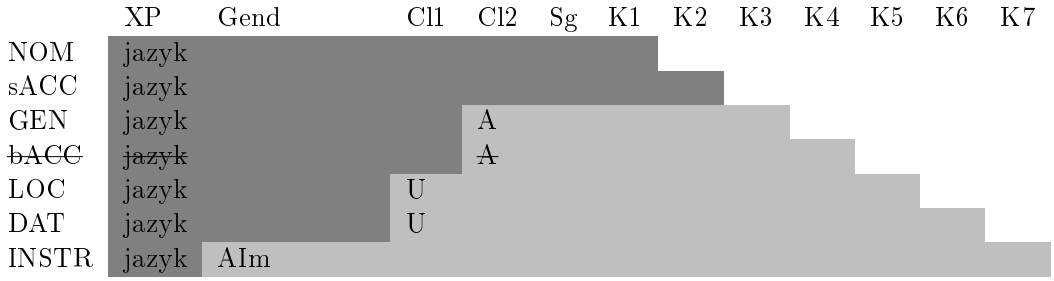
[52] In the same paper, Starke also identifies two different datives in Germanic languages. Since those are not our focus, we leave them aside.

[53] We cross the unused case in each table to make it visually easier to navigate.

(400) LEXICALIZATION TABLE OF 'SIR', WITH bACC

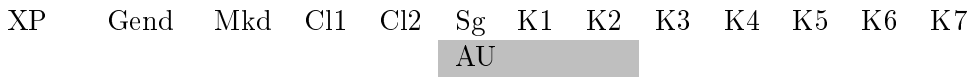


(401) LEXICALIZATION TABLE OF 'TONGUE', WITH sACC



Splitting the accusative into two re-shuffles the syncretisms, which leads to the -AU suffix not popping up in the accusative of *pán* 'sir'. See it repeated below:

(402) LEXICAL ENTRY OF -AU



Now, after the root *pán* 'sir' grows up to K1, its derivation goes straight to the genitive. The root starts to backtrack to find a suffix able to spell out K3 – and the first that it finds is -A. The -AU suffix is not applicable because it cannot spell out K3. The same -A found for the genitive can extend to K4, which now marks the (animate) big accusative, and that is how we get the typically animate GEN-ACC syncretism.

Note that the small accusative for *pán* 'sir' and the big accusative for *jazyk* 'tongue' are never used, and hence if we were to pursue this account, we would have to explain why. It will likely be a blocking effect of some sort due to the difference in animacy, but we do not delve into the issue here.

While the morphosyntactic context of using the big accusative is not precisely the same between Czech and Spanish, both languages use the big accusative for the animate nouns (which seems well compatible with the animacy/definiteness scales in Aissen 2003). So there might be a chance to tie the big accusative with animacy somehow, which would then differentiate this type of an account from the one with DLB trees, as that one treats the animacy connection of the GEN-ACC pattern as an accident.

Still, given the existence of Czech inanimate nouns that can take the supposedly animate GEN-ACC pattern, we argue that the pattern is not presently tied to animacy and is merely a matter of declension class instead. We see no obvious way how these nouns could be incorporated into an account that connects them with animacy, in which way the DLB account seems superior.

The same account can be extended for the palatalized paradigm *muž* 'man'. It has the same GEN-ACC syncretism as *pán* 'sir' does but different surface suffixes:

(403) SURFACE FORMS OF 'MAN' AND 'SIR'

	'man'	'sir'
NOM	muž	pán
ACC	muž-e	pán-a
GEN	muž-e	pán-a
LOC	muž-i	pán-u
DAT	muž-i	pán-u
INSTR	muž-em	pán-em

On a par with our treatment of other palatalized paradigms in this work, we suggest that their syntax (their lexical entry) is the same, and the surface difference between the two is caused by the fact that *muž* 'man' has the root-final floating I, just as all the other palatalized paradigms. The I then combines with the suffixal -A in PF (for details, refer back to section 4):

(404) ACC mužI + A → muž-e

In this part, we incorporated the masculine animate roots *pán* 'sir' and *muž* 'man' into our system. They differ from the masculine inanimates like *jazyk* 'tongue' only in having a GEN-ACC syncretism. Our main proposal is rooted in the idea that these roots have complex branching lexical entries that can – just through the standard Nanosyntactic algorithm – derive the syncretism.

The advantage of this account is freeing the GEN-ACC pattern from its standardly-viewed tie to animacy, which is hard to maintain once we consider that there are also some inanimates with the same pattern. That being said, we will still add an animacy feature into our functional sequence in the section 6.1.

5.2 Height of the complex left branch: back to the castle

In this section, we return back to the paradigm *hrad* 'castle' that we introduced in section 3.2. While we suggested a working analysis there, we had to propose that the paradigm has an invisible suffix in the nominative and accusative.

The paradigm shows no overt suffix in the two structurally lowermost cases, just as another paradigm with the same features, the XL-sized *jazyk* 'tongue'. Despite seemingly having the same root size, though, they split ways in the genitive, where *hrad* 'castle' takes *-u*, while *jazyk* 'tongue' takes *-a*:

(405) SURFACE FORMS OF 'CASTLE' AND 'TONGUE'

	'castle' (Mi)	'tongue' (Mi)
NOM	hrad	jazyk
ACC	hrad	jazyk
GEN	hrad-u	jazyk-a
LOC	hrad-u	jazyk-u
DAT	hrad-u	jazyk-u
INSTR	hrad-em	jazyk-em

It is impossible in our system for two nouns to have the same features, the same size (and shape), and yet somehow start to differ throughout the derivation. Based on that, we suggested the following lexical entries for *hrad* 'castle'. We used a yer as a placeholder for whatever silent suffix is in the nominative and accusative of the *hrad* 'castle' paradigm:

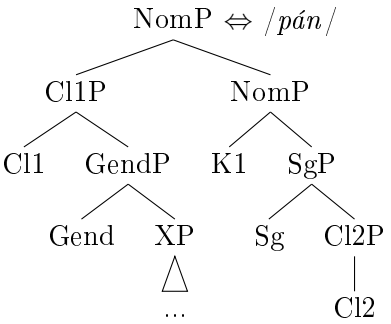
(406) LEXICALIZATION TABLE OF 'CASTLE'

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	hrad		ъ								
ACC	hrad		ъ								
GEN	hrad		U								
LOC	hrad		U								
DAT	hrad		U								
INSTR	hrad	AIm									

In the previous subsection, we have introduced a new tool, the DLB trees, and saw how the size of a lexical entry and its shape determines which suffixes a root takes throughout the derivation. We will look at *hrad* through the same lense now, making a more complex shape a part of the entry, and we will see how it allows us to capture the paradigm cleanly without a silent suffix or a need to postulate anything new.

What we suggested for the paradigm *pán* 'sir' was a lexical entry whose complex left branch sits on top of the structure (repeated below):

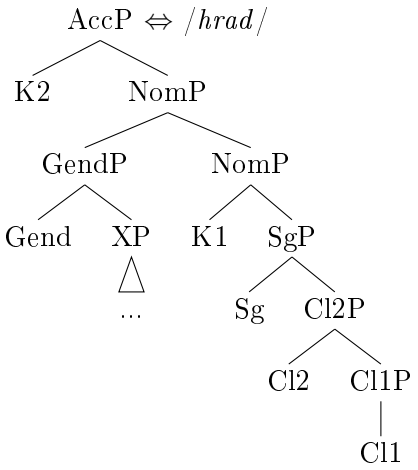
(407) DLB LEXICAL ENTRY OF 'SIR'



However, nothing prevents the complex left branch from being anywhere in the structure, with simplex branch(es) on top of it. That is exactly what we propose for *hrad* 'castle'.

The structure we ultimately adopt is below. Note that compared to *pán* 'sir', not only does it have K2 on top, but also the size of the left branch changed. The Cl1 feature migrated from the left branch to the right. We will discuss the reason behind this choice later on:

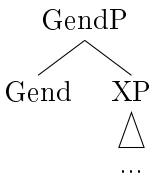
(408) DLB LEXICAL ENTRY OF 'CASTLE'



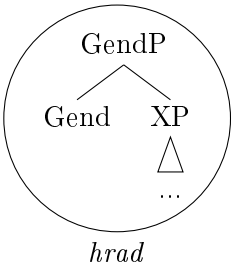
Structuring the lexical entry like this works because putting the complex left branch between K1 and K2 (i.e., the nominative and accusative layers) allows us to spell out both cases. It spells out the accusative using the whole tree and the nominative discarding the K2 layer under the standard Superset Principle.

Let us now go through a derivation to show how it proceeds step-by-step and point out a few vital decisions we made in suggesting the lexical entry. Our starting point is merging the XP with Gend (409). The phrase is spelled out as *hrad* 'castle' because it matches a subset of that entry, concretely its complex left branch (410):

(409) DERIVATION OF 'CASTLE', NOM, INTERMEDIATE STEP

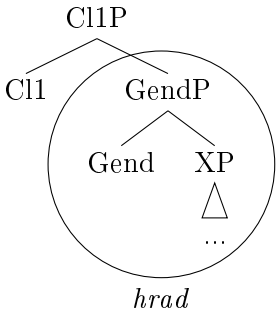


(410) DERIVATION OF 'CASTLE', NOM, INTERMEDIATE STEP

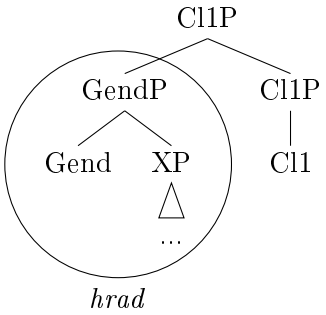


In next step, Cl1 is merged (411). The root cannot spell the structure out the way it is and hence does an evacuation movement – and since there is no eligible Spec, the whole complement gets moved (412):

(411) DERIVATION OF 'CASTLE', NOM, INTERMEDIATE STEP

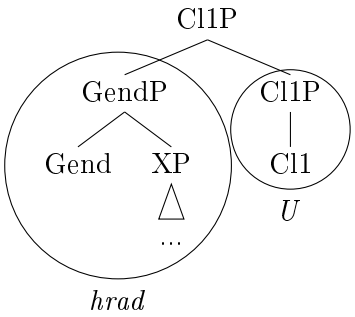


(412) DERIVATION OF 'CASTLE', NOM, INTERMEDIATE STEP



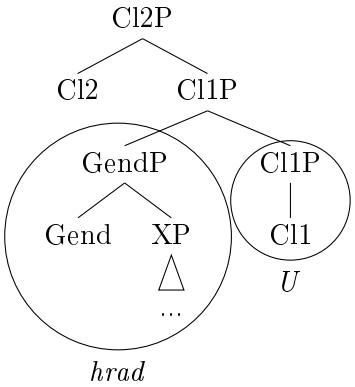
The root cannot spell out the new structure in its entirety like this either because to match it with the entry for *hrad*, there are missing features on the right branch. On the other hand, it can still spell out the left branch, as before, and another entry can spell out the newly freed-up Cl1P. Previously, there were two entries with a foot in Cl1: -U and the yer. However, since we discarded the yer, the only entry left is -U. So the spell out of both of the branches combined will be *hrad-U*:

(413) DERIVATION OF 'CASTLE', NOM, INTERMEDIATE STEP

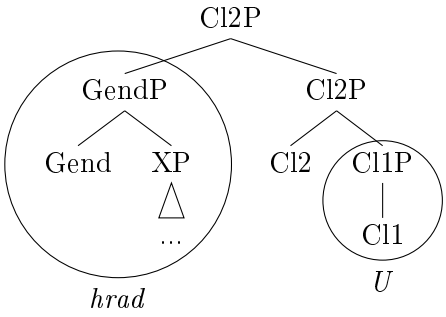


Next, we merge Cl2 (414), and since the root cannot spell out the whole structure, an evacuation movement gets triggered. This time there is an eligible Spec, so we move it (415):

(414) DERIVATION OF 'CASTLE', NOM, INTERMEDIATE STEP

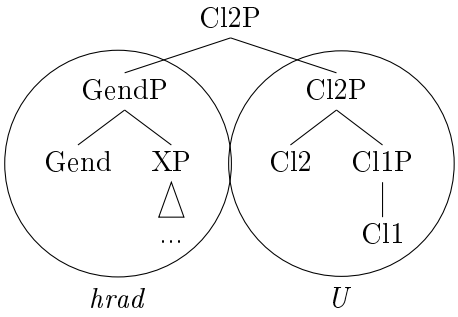


(415) DERIVATION OF 'CASTLE', NOM, INTERMEDIATE STEP



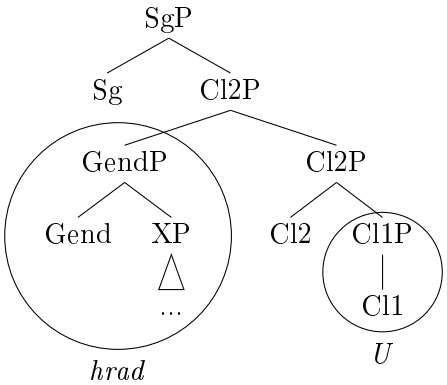
The right branch can still be spelled out by -U:

(416) DERIVATION OF 'CASTLE', NOM, INTERMEDIATE STEP

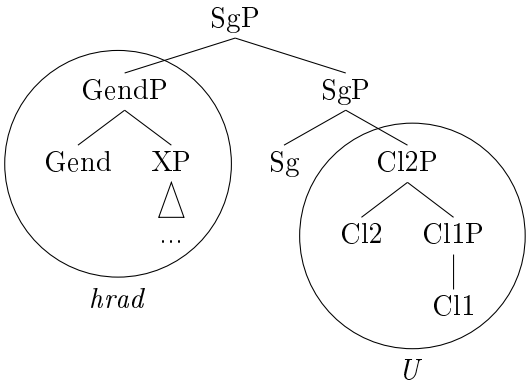


The next step follows the same pattern. We merge Sg (417) and then cyclically move the Spec (418). The root spells out the left branch, -U the right branch (419):

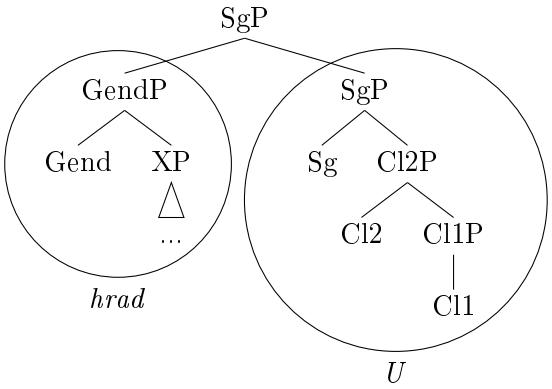
(417) DERIVATION OF 'CASTLE', NOM, INTERMEDIATE STEP



(418) DERIVATION OF 'CASTLE', NOM, INTERMEDIATE STEP

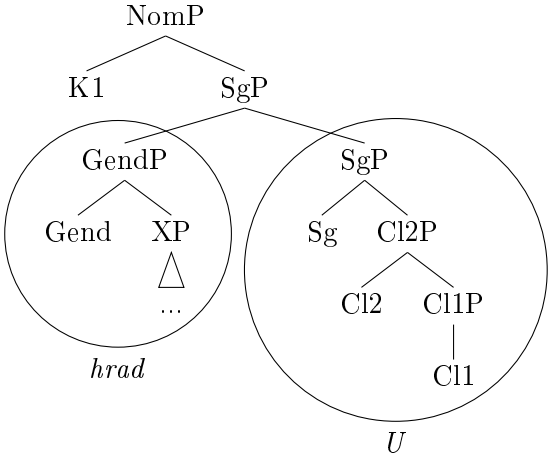


(419) DERIVATION OF 'CASTLE', NOM, INTERMEDIATE STEP



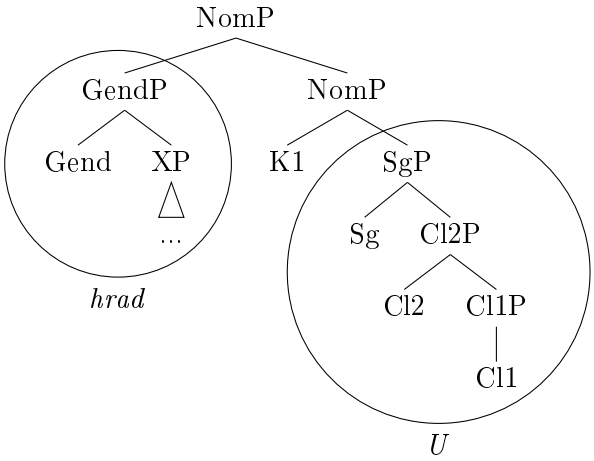
Finally, for the nominative derivation, we only miss one last feature, K1. So we merge it:

(420) DERIVATION OF 'CASTLE', NOM, INTERMEDIATE STEP



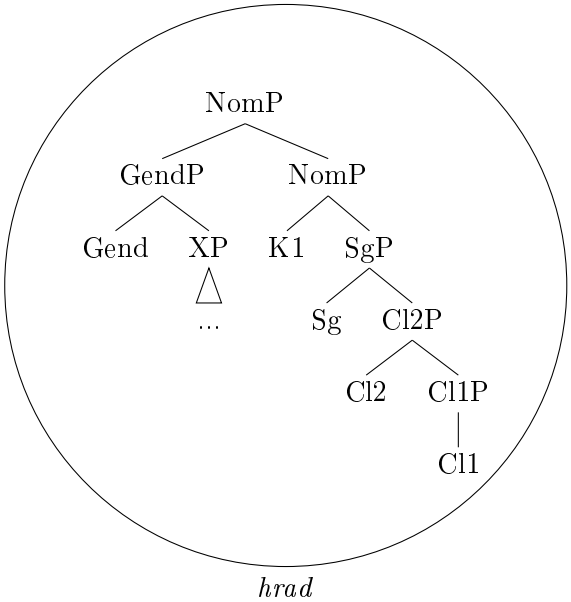
The root cannot spell out the whole structure, so it again cyclically moves the Spec on top:

(421) DERIVATION OF 'CASTLE', NOM, INTERMEDIATE STEP



The note-worthy thing that happens is that the root can now spell out the whole structure – because the whole structure, with the complex left branch and everything, is a proper subset of the lexical entry for *hrad* 'castle' (refer back to (408)). So instead of a split spellout of the root and the suffix, we only get the root:

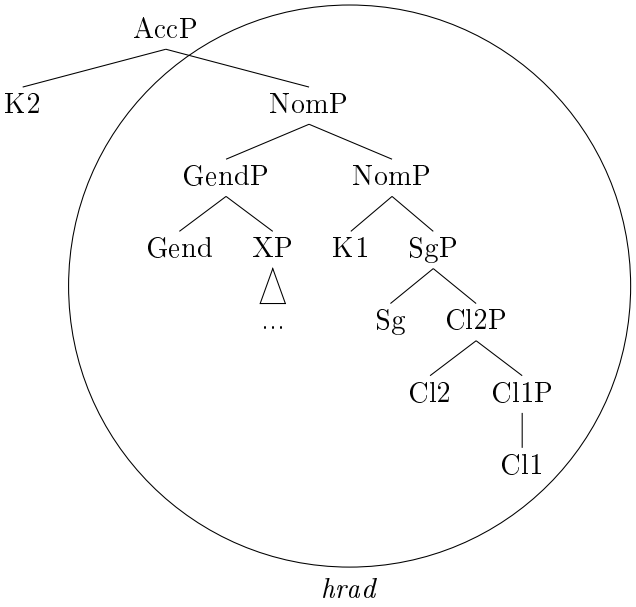
(422) DERIVATION OF 'CASTLE', NOM, FINAL FORM



What changed was that the structure reached a stage where both branches are a subset of the lexical entry of the root.

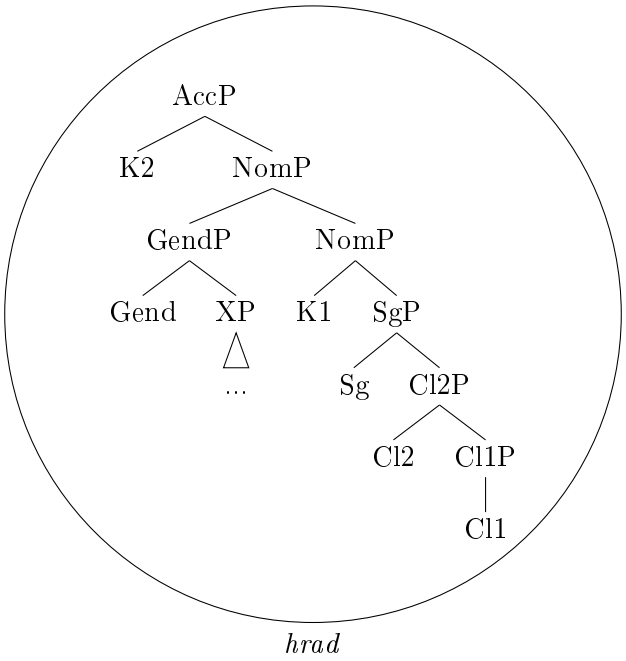
The accusative derivation is simple, then. We merge K2:

(423) DERIVATION OF 'CASTLE', ACC, INTERMEDIATE STEP



And since the structure is a perfect match for the root as it is, we spell it out as *hrad*, and that is it:

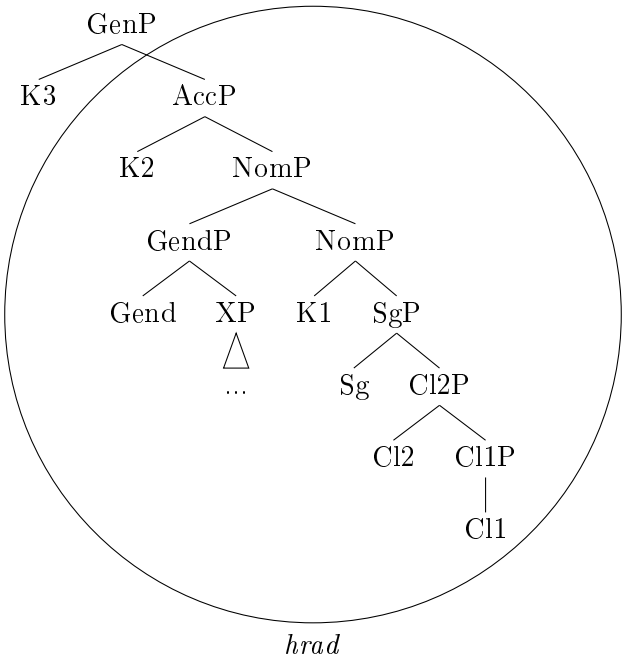
(424) DERIVATION OF 'CASTLE', ACC, FINAL FORM



The genitive is, of course, the crucial case for us because it is there where we need *hrad* 'castle' to derive -U (whereas a root of the same size but with a simple lexical entry, like *jazyk* 'tongue', derives -A).

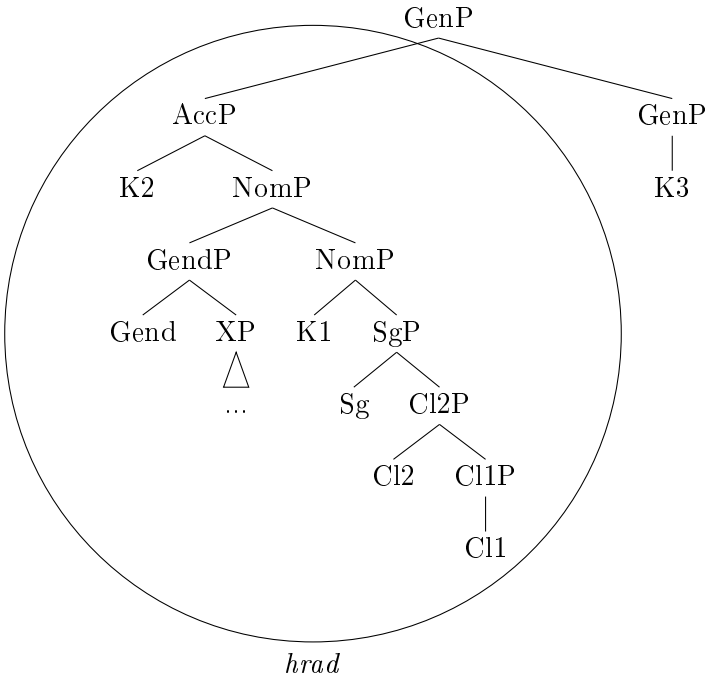
For a genitive derivation, we merge K3 to the accusative structure:

(425) DERIVATION OF 'CASTLE', GEN, INTERMEDIATE STEP



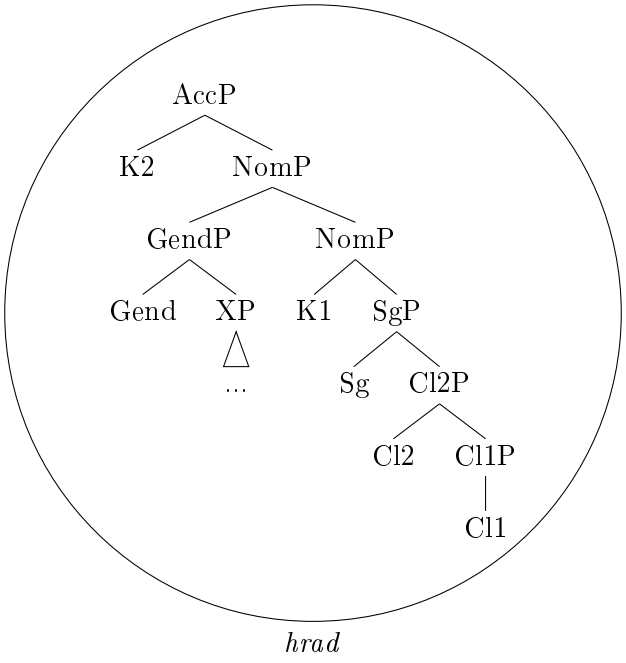
Since the root cannot spell the structure out as it is, the evacuation movement gets triggered – and because there is no eligible Spec for the move (as the only eligible Spec would be a specifier of the daughter of the root node), the whole complement moves across K3:

(426) DERIVATION OF 'CASTLE', GEN, INTERMEDIATE STEP



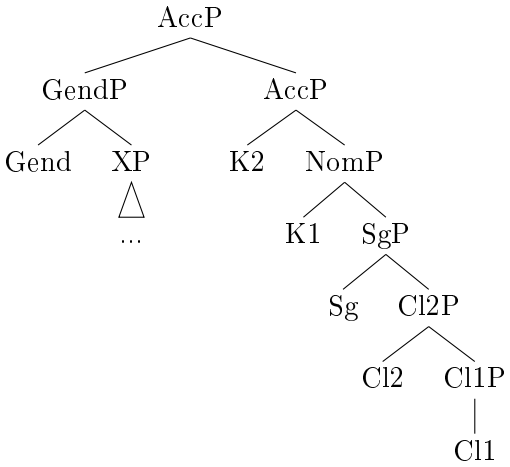
Freeing up K3 does not help because no entry in our lexicon can spell out just K3. So the K3 stage gets undone, and, since there is no other movement to try, the structure backtracks to the accusative stage:

(427) DERIVATION OF 'CASTLE', GEN, INTERMEDIATE STEP



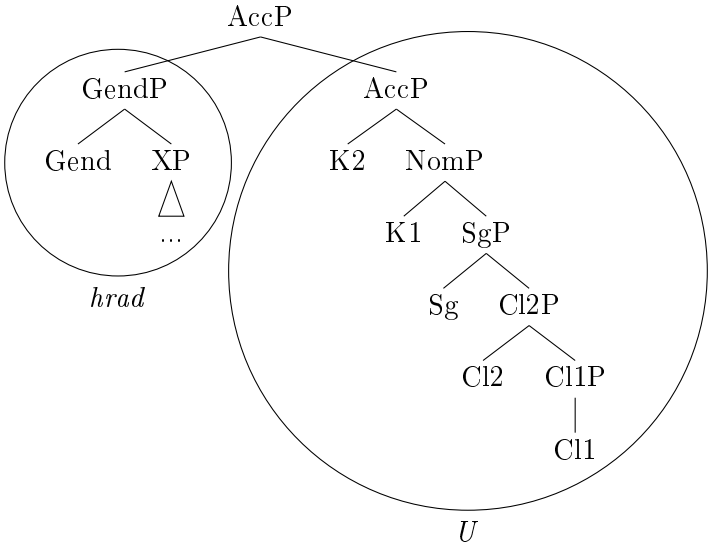
When a structure backtracks, it always tries the next eligible option of the previous stage that has not been tried before. In our case, we did not move at all on the K2 stage because the root could spell out the whole structure. Therefore, the next option is the Spec movement. We move the complex left branch on top:

(428) DERIVATION OF 'CASTLE', GEN, INTERMEDIATE STEP, BACKTRACKING



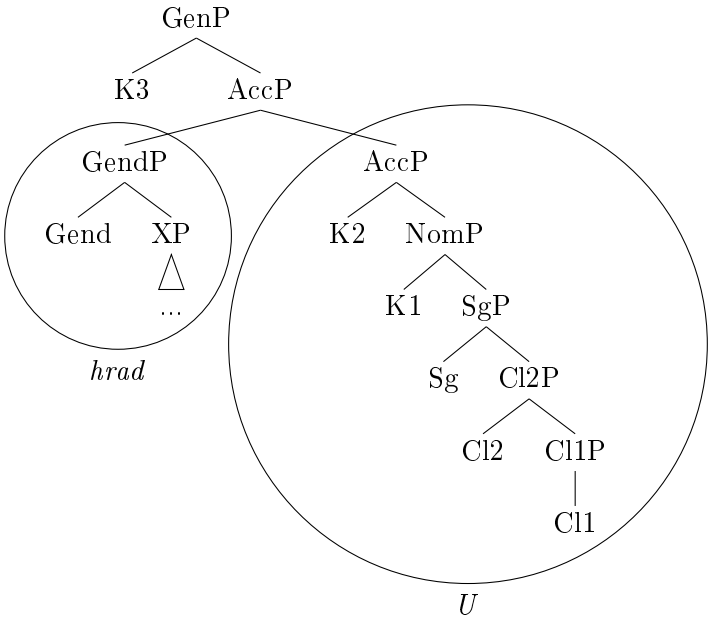
This frees up the right branch for a separate spellout, and this time, it does find a match, -U. The left branch can be again spelled out by the root:

(429) DERIVATION OF 'CASTLE', GEN, INTERMEDIATE STEP, END OF BACKTRACKING



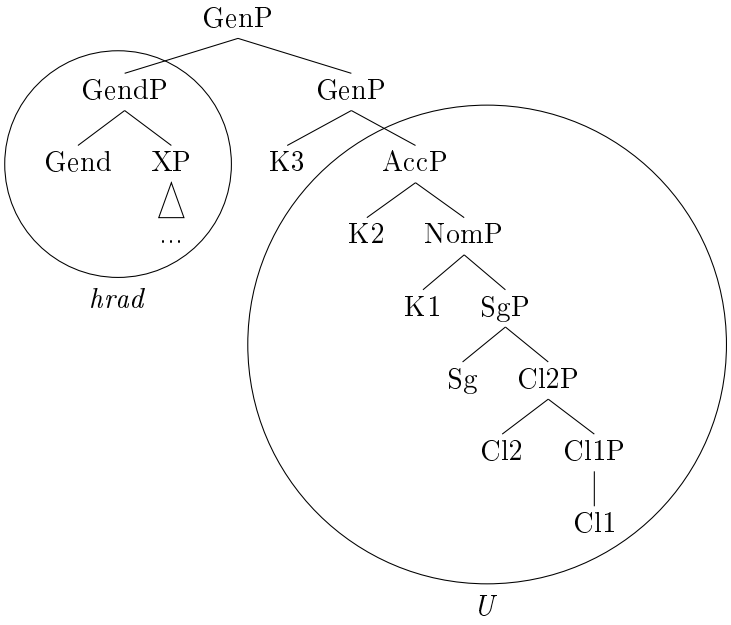
Next, we merge the K3 feature back to see whether all the new evacuation movement helped in finding the genitive spellout:

(430) DERIVATION OF 'CASTLE', GEN, INTERMEDIATE STEP



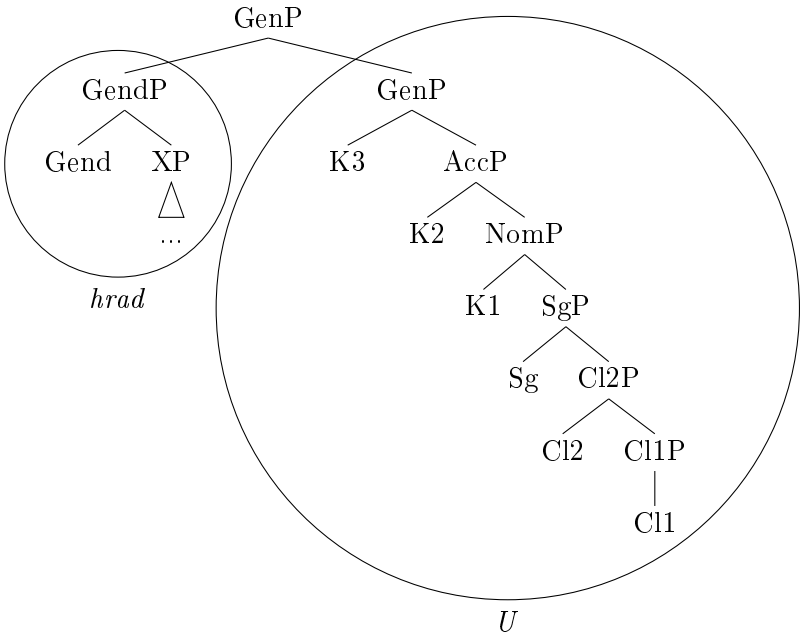
The structure cannot be spelled out as it is, so the Spec movement gets triggered, bringing the complex left branch across the K3 feature:

(431) DERIVATION OF 'CASTLE', GEN, INTERMEDIATE STEP



It turns out that the backtracking indeed helped find a proper path to derive the genitive. Since the lexical entry for -U can also spell out K3, the right branch will still be spelled out by it and gives us precisely the right genitive spellout, *hrad-u*:

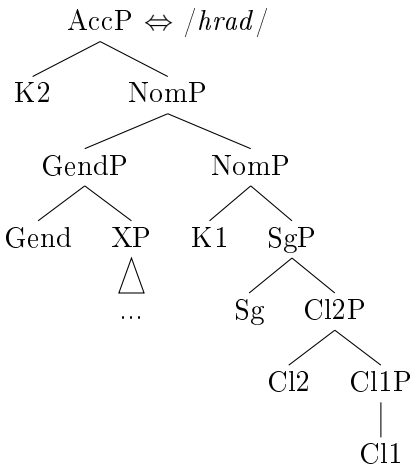
(432) DERIVATION OF 'CASTLE', GEN, FINAL FORM



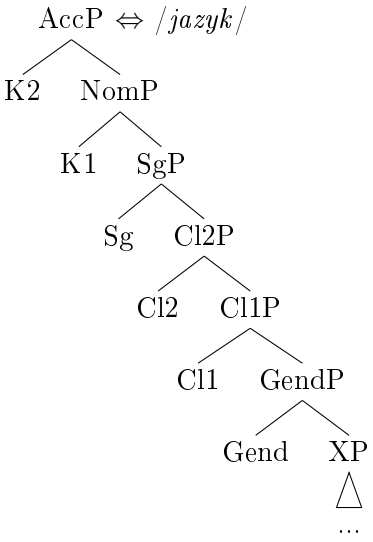
That is how we get the correct genitive structure. From then on, the rest of the cases are derived in the same way as the XL-paradigm *jazyk* 'tongue' since they share the same suffixes from the locative up.

Compare how the shape of the lexical entry for *hrad* 'castle' differs from the one for *jazyk* 'tongue':

(433) DLB LEXICAL ENTRY OF 'CASTLE'



(434) LEXICAL ENTRY OF 'TONGUE'



There are three takeaways we would like to highlight:

1. The complex nature of the lexical entry of *hrad* 'castle' allows it to take a different derivational path compared to the standard simple lexical entry of *jazyk* 'tongue'.

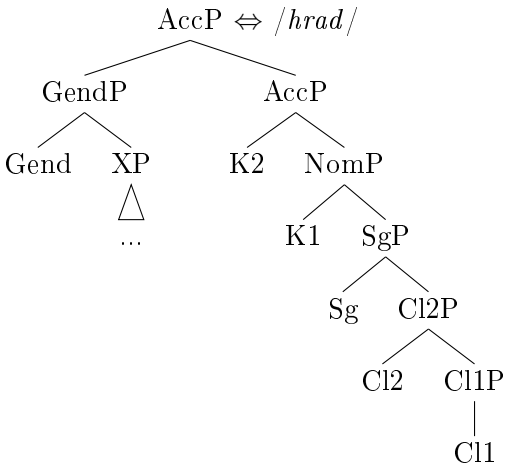
   Where *jazyk* 'tongue' keeps extending up to the nominative or accusative without any movement, *hrad* 'castle' is forced to move out XP+Gend cyclically up. Upon reaching the accusative stage, both structures backtrack for the genitive, but their different shape forces them to go for different movement options, and therefore they find a different suffix. This is how we achieve having the same size for both roots – XL, all the way up to K2, spelling out the same features – but still take a slightly different derivation.

2. The root *hrad* 'castle' can spell out both the nominative and the accusative due to having the complex left branch in between the two projections and therefore being able to interact with both of them.

   The structure would also work if we put the complex left branch below K1 (or lower) because it would still allow the syntax to shed top layers via the Superset Principle.

   At the same time, putting the complex left branch on top of the structure (the way we did for *pán* 'sir'), i.e., above K2, would fail. Consider the entry below:

   (435) MOCK LEXICAL ENTRY OF 'CASTLE', WITH DLB ON TOP OF THE STRUCTURE



   The accusative would be spelled out using the whole tree, but there would be no way to use this structure to spell out the nominative because there is no node containing both the left branch and the NomP branch without the AccP node.

3. It is crucial for *hrad* that its right branch is footed in the same feature (Cl1) as the suffix -U we want to derive in the genitive. The reason is that once the left branch extracts and spells out the root on its own, the right branch automatically "picks up" whatever entry best fits it.

Note that it is not the case that the right branch always has to match the desired suffix. The foot of the right branch must either match or be above the foot of the desired suffix. If it is above, it can backtrack to it. The reason why we need the exact match for *hrad* 'castle' here is that if we put the foot of its right branch anywhere higher, other lexical entries start to interact with it in the derivation in unwanted ways. If the foot were, e.g., in Cl2, the right branch would be automatically spelled out by -A, which is footed in Cl2.

As has been already mentioned, redesigning the lexical entry for *hrad* 'castle' the way we did above has the consequence of us being able to let go of the silent suffix. It also means that *hrad* (and its palatalized counterpart *kost* 'bone') is no longer of the XS-size, but of the XL-size together with *jazyk* 'tongue' (and *pokoj* room' and *dlaň* 'palm'), because it now spells out all the features up to K2.

Employing DLB trees can challenge certain derivation patterns we presented earlier. Take again the surface forms of *hrad* 'castle' and *jazyk* 'tongue', focusing on the *-u* syncretism:

(436) SURFACE FORMS OF 'TONGUE' AND 'CASTLE', -U SYNCRETISM

	'tongue'	'castle'
NOM	jazyk	hrad
ACC	jazyk	hrad
GEN	jazyk-a	hrad-u
LOC	jazyk-u	hrad-u
DAT	jazyk-u	hrad-u
INSTR	jazyk-em	hrad-em

Remember that when we were only working with simple structures without complex left branches, whenever we would see two roots taking the same suffix but one root having it earlier, it necessarily meant that that root was smaller. In the table above, we see *hrad* 'castle' taking *-u* earlier, which in the simple trees system meant, it was smaller than *jazyk* 'tongue'. That was because all the simple structures backtrack the same and always first find the entry with a higher foot. Smaller entries have a shorter way to find a lower-foot entry when backtracking.

However, once we employ DLB trees, the matter becomes slightly more complex. Both *jazyk* 'tongue' and *hrad* 'castle' are of the same size now and yet *hrad* 'castle' reaches *-u* earlier, which we achieve via its complex left branch. This seemingly goes against the "reaches earlier, therefore smaller" rule.

The rule still works, though. Only it is no longer about the total size of the root but rather the size of the left branch, if applicable, since it is the left branch and left branch only that combines with -U.[54] Since the extracting left branch of *hrad* 'castle' is smaller than the entry for *jazyk* 'tongue', we expect it to take the suffixal entry with a lower foot before *jazyk* 'tongue' does.

[54] From a different perspective, it is about the size of the branch with the lowermost features, which is the left branch for *hrad* 'castle' but the right branch for *jazyk* 'tongue'.

6 Loose ends: pointers

There are three paradigms in the Czech nominal domain that prove to be rather tricky to account for. Two of them are feminine, *žen-a* 'woman' and *růž-e* 'rose', and one is masculine, *koleg-a* 'colleague'.[55] This section shows two issues with their incorporation. Interestingly, both share the same pattern.

The first issue has to do with the paradigms *žen-a* 'woman' and *koleg-a* 'colleague' and is about how the paradigms can reunite after they have already split:

(437) SURFACE FORMS OF 'WOMAN' AND 'COLLEAGUE (MASC)'

	'woman' (F)	'colleague' (Ma)
NOM	žen-a	koleg-a
ACC	žen-u	koleg-u
GEN	žen-y	koleg-y
LOC	žen-ě	koleg-ovi
DAT	žen-ě	koleg-ovi
INSTR	žen-ou	koleg-ou

The two paradigms have the same suffixes from the nominative to the genitive. Then they split for the locative and dative, only to reunite again in the instrumental. These two roots are likely to have slightly different functional sequences, at least in gender specification and/or possibly in animacy, as Czech feminine nouns never show an animacy distinction, while masculines do.

In abstract, it comes down to the lexicalization tables in (438) and (439). The root $\sqrt{1}$ has an extra feature M that the root $\sqrt{2}$ lacks. This extra feature is located in between the features 1 and 2:

(438)

ROOT	1P	MP	2P	3P
$\sqrt{1}$	X			
$\sqrt{1}$	W			
$\sqrt{1}$	Z			

(439)

ROOT	1P		2P	3P
$\sqrt{2}$	X			
$\sqrt{2}$	Y			
$\sqrt{2}$	Z			

Both roots are of the same size, so when they derive 1P, they take the same lexical entry. During the derivation of 2P, the suffix grows over the M feature position that only one of the roots contains. This makes the root $\sqrt{1}$ select a lexical entry for W because W is specified for the feature M. On the other hand, the root $\sqrt{2}$ has to select Y as it needs an entry not specified for M.

This is a straightforward way to make two paradigms split after starting the same – they grow (or backtrack) over a feature that distinguishes them.[56]

The problem comes when we now want to derive the third line of each of the roots. As we can see from the Z marking, the paradigms reunite again. So the whole pattern

[55] The feminine version of the root *koleg* is *koleg-yn-ě* and as such falls under the paradigm *růž-e* 'rose'.

[56] We already saw a similar case when we dealt with the instrumental. Backtracking over a Mkd feature that only feminines are specified for caused the feminine and non-feminine instrumentals to split. Refer back to the subsection 4.1 for details.

is: start the same – split – reunite. Due to its shape of sharing the top and the bottom, we call this "the conjoined twin" pattern:

(440) THE CONJOINED TWIN PATTERN

X	X
Y	W
Z	Z

Deriving the conjoined twin pattern is not straightforward. There are ways to make paradigms start the same and then split. We saw that roots can grow (or backtrack) over a feature that distinguishes them, which causes the split. There are also ways to make the paradigms start different and then unify – which could be explained by them having a different root size but ultimately finding a way to unite through backtracking. However, combining both faces the issue that if they split over a different feature, the feature will always remain present in further derivations and cause problems with the unification.

The second pair we look at in this section poses an even bigger challenge. The two feminines, *žen-a* 'woman' and *růž-e* 'rose', seem to start the same, then split and the unify again, showing off the same conjoined twin pattern:[57]

(441) SURFACE FORMS OF 'WOMAN' AND 'ROSE', ET-DECOMPOSED

	'woman'	'rose'
NOM	žen-A	růžI-A
ACC	žen-U	růžI-U
GEN	žen-I	růžI-AI
LOC	žen-AI	růžI-I
DAT	žen-AI	růžI-I
INSTR	žen-U+length	růžI-U+length

What makes this issue different from the woman–colleague one is that *žen-a* 'woman' and *růž-e* 'rose' are likely to have the same features. They are both feminines, and the roots in each of the paradigms do not show any apparent semantic grouping or any other clear distinction to justify a featural difference.

In the abstract, this translates to the lexicalization tables below:

(442)

ROOT	1P	2P	3P
$\sqrt{1}$	X		
$\sqrt{1}$	Y		
$\sqrt{1}$	Z		

(443)

ROOT	1P	2P	3P
$\sqrt{2}$	X		
$\sqrt{2}$	W		
$\sqrt{2}$	Z		

Without a difference in features, it is not immediately apparent how the two paradigms can start the same and then split. What would be driving it? And how they unify again after the mysterious split?

[57] For simplicity, we are directly showing the Element Theory decomposition forms here. In the following subsection, we come back to the surface forms and the decomposition steps, including a few of its issues.

The following subsection goes through ways how we can deal with the conjoined twin pattern. We start from the woman–colleague pair.

6.1 The wondrous reunion of a woman and her colleague

6.1.1 The conjoined twin pattern: theoretical options

As has been mentioned already, *žen-a* 'woman' and *koleg-a* 'colleague' differ in their gender specification. *Žen-a* is a feminine, while *koleg-a* is a masculine. *Koleg-a* is the only masculine paradigm that can take *-a* in the nominative singular, a suffix generally regarded as feminine.

Below, the two paradigms are repeated and compared side by side:

(444) SURFACE FORMS OF 'WOMAN' AND 'COLLEAGUE (MASC)'

	'woman' (F)	'colleague' (Mi)
NOM	žen-a	koleg-a
ACC	žen-u	koleg-u
GEN	žen-y	koleg-y
LOC	žen-ě	koleg-ovi
DAT	žen-ě	koleg-ovi
INSTR	žen-ou	koleg-ou

The syncretisms are color-coded and show the conjoined twin pattern of sharing the same suffixes in the nominative, accusative, and genitive, then splitting for the locative and dative, and then unifying for the instrumental again.

Since the first two steps – starting the same and then splitting – are pretty straightforward, let us focus on the last part of the puzzle, which is how the paradigms unify again. There are at least four ways to deal with paradigm leveling in Nanosyntax:

1. backtracking
2. simple pointers
3. pointer siege
4. backtracking + simple pointers

Let us now go through each in an abstracted way to see their benefits and drawbacks.[58]

Backtracking

In the lexicalization table below, we see a situation we have touched upon in the introduction to this section. It shows the first two parts of the derivation:

[58] Readers, who wish to skip the abstract discussion and go straight to real data and suggested accounts for the three paradigms at hand, should go directly to the example (491) and the text that follows it.

(445)

-3P	-2P	-1P	1P	MP	2P	3P
	$\sqrt{1}$		X			
	$\sqrt{1}$		W			

(446)

-3P	-2P	-1P	1P		2P	3P
	$\sqrt{2}$		X			
	$\sqrt{2}$		Y			

In deriving 1P, both roots are of the same size and therefore select the same morpheme. In deriving 2P, their ways split because the derivation goes over MP. Only the root $\sqrt{1}$ is compatible with the meaning of M, so it selects W, while the root $\sqrt{2}$ is not and therefore selects Y.[59] What happens in the 3P derivation if we only use backtracking?

(447)

-3P	-2P	-1P	1P	MP	2P	3P
	$\sqrt{1}$		X			
	$\sqrt{1}$		W			
$\sqrt{1}$		Z				

(448)

-3P	-2P	-1P	1P		2P	3P
	$\sqrt{2}$		X			
	$\sqrt{2}$		Y			
$\sqrt{2}$		Z				

No matter how far we backtrack, there will always be a difference in spelling out the M feature, which will be part of the 3P derivation. So the root $\sqrt{2}$ will always have a gap and will therefore be unable to spell out the same lexical entry as $\sqrt{1}$.

Therefore, backtracking alone can never account for the conjoined twin pattern.

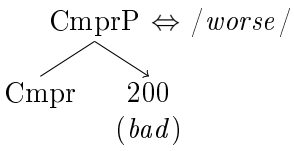
Simple pointers

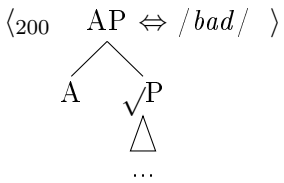
Pointers are a Nanosyntactic device where one lexical entry points to another existing, lexical entry.[60]

Here is an example from likely the most developed area for pointers, the root suppletion (adapted from Vanden Wyngaerd et al. 2021):

[59] We can think of it alongside the lines of how the Anim feature will only be relevant for animate derivations and will be missing in the inanimate roots.

[60] This tool, originally suggested by Starke in unpublished work, has been so far used in at least three types of contexts – idioms, root suppletion (Starke 2014, Caha et al. 2019a, Vanden Wyngaerd et al. 2021), and affix selection and affix syncretism in multidimensional paradigms (Caha and Pantcheva 2012, Caha 2019a, DeClercq and Vanden Wyngaerd 2019, Taraldsen 2019, Blix 2021, Caha 2021b). For work discussing a potential problem of the pointer device – that they might allow certain ABA patterns – see, e.g., Vanden Wyngaerd (2018).

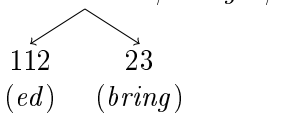
(449) 

(450) 

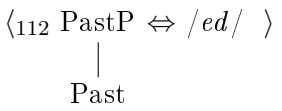
The notation in (449) informally reads as "if the cycle before merging the comparative head has been spelled out by the lexical entry 200 (= *bad*, cf. (450)), then the cycle after merging Cmpr will be spelled out as *worse*". This way, we can put concrete forms in a relationship with each other.

The part that will be relevant for us is that both *bad* and *worse* in the examples above are discrete lexical entries, and as such, they succumb to the Superset Principle separately. Let us take a different example to show how that works.

The English verb *bring* has an irregular past form *brought*, which lacks the regular past tense morpheme *-ed*. This can be dealt with well using the pointer device by suggesting lexical entries like the ones below (adapted from Vanden Wyngaerd 2018):[61]

(451) 

(452) 

(453) 

The lexical entry in (451) says that if a structure spelled out as *-ed* (452) finds itself just above a structure spelled out as *bring* (453), then the mother node XP should be spelled out as *brought*.

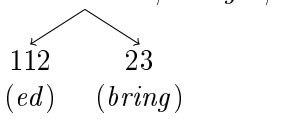
The Superset Principle comes into play once we look more closely at the syntactic structure of the morpheme *-ed*. Besides in simple past (454-a), *-ed* can also be found in (past) perfect (454-b) and (past simple) passive (454-c) forms (Vanden Wyngaerd 2018):

(454) a. They elected George
b. They have elected George
c. George was elected

Given the syncretisms, the suffix *-ed* must be internally more complex. In (457), see an updated version of the *-ed* lexical entry based on the structure that Vanden Wyngaerd (2018) suggests:

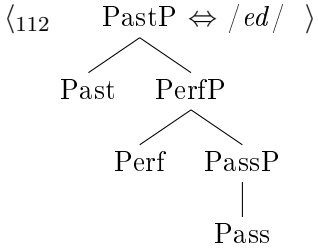
[61] One place where we adapted Vanden Wyngaerd's (2018) approach is in having *bring* below *-ed* in the lexical entry for *brought*. Both options are possible and only differ in whether the syntactic structure finds a matching entry pre- or post-movement. The reason we chose the pre-movement version here is for expository purposes since our own structures later in this section work with the pre-movement versions as well.

(455)



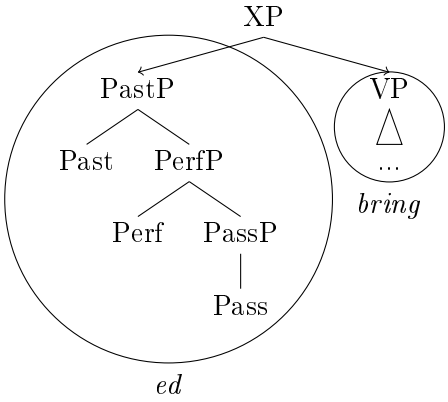
(456)

(457)



Combining the more fine-grained structure of *-ed* with the structure of the verbal root *bring* now leads to the derivation below[62]:

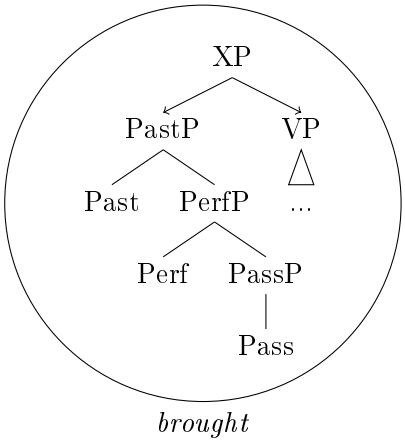
(458)



The highest node (XP) of the structure in (458) now matches the lexical entry in (455) and therefore can be spelled out as *brought*:

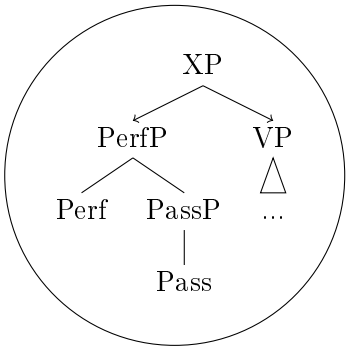
[62]From the derivation in (458) onward, we choose to draw pointers also in syntax. This is technically an incorrect representation as pointers are a device that is only used in the lexicon and never in syntax. However, to make the logic more visual, we use the arrows in syntactic derivations as well so that it is immediately clear where a pointer from the lexicon interacts with the structure.

(459)

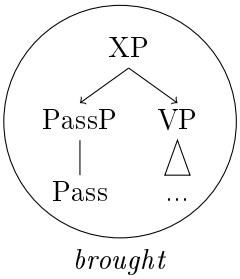


The interesting part is that thanks to the pointer, we can also spell out the structures (460) and (461), which have a gap in the middle. The example in (460) lacks the PastP projection, while the example in (461) misses both PastP and PassP. This kind of gapping would be impossible without the pointer "restarting" the spellout:

(460)



(461)



Anything under a pointer is always a discrete lexical entry, which allows the pointers to spellout subsets in those contexts. We will use this pointer attribute in an attempt to account for the conjoined twin pattern.

Let us now look at the use of a simple lexical entry with a pointer and see whether this tool would be able to account for the 3P reunion, despite the difference in the two functional sequences.

The first two rows of both derivations are the same as in the backtracking example. It is the third line that differs:

(462)

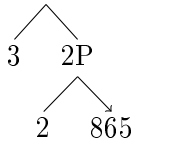
-3P	-2P	-1P	1P	MP	2P	3P
	$\sqrt{1}$		X			
	$\sqrt{1}$		W			
	$\sqrt{1}$		?		← Z	

(463)

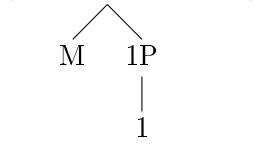
-3P	-2P	-1P	1P		2P	3P
	$\sqrt{2}$		X			
	$\sqrt{2}$		Y			
	$\sqrt{2}$		?		← Z	

In the third line, we have set up a lexical entry with a pointer that, in the tree format, looks as follows:

(464)



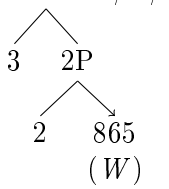
(465)



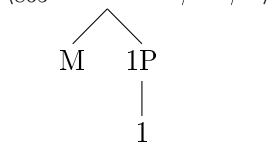
The pointer resides just above M, a feature in which the two functional sequences in question differ. That allows us to "restart" the lexicalization there and hence use a subset of the lexical entry below. In simpler words, it allows us to skip the feature M when it is not part of the structure. The question now is, what is below the pointer.

We know that in the lexicon, the entry needs to be W:

(466)



(467)



That is because only W can spell out both MP+1P as needed for $\sqrt{1}$, and at the same time just 1P as needed for $\sqrt{2}$.

Whether we get the correct entries in the actual spellout depends on our assumptions of who "counts" under the pointer. More precisely, whether it has to be a winner of that stage of the derivation (Taraldsen 2019), or whether all candidates count (Caha 2021b).

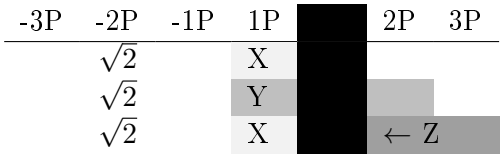
The root $\sqrt{1}$ is again not a problem either way because it will derive the needed W, as it is the only lexical entry able to spell out M:

(468)

-3P	-2P	-1P	1P	MP	2P	3P
	$\sqrt{1}$		X			
	$\sqrt{1}$		W			
	$\sqrt{1}$		W		← Z	

The issue is with the root $\sqrt{2}$. The winner of the stage under the pointer is X, not W, because X is the perfect match for 1P:

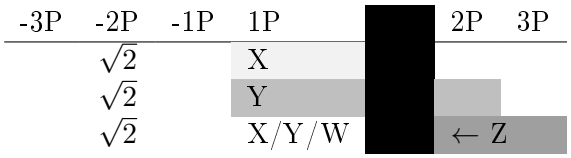
(469)



Hence if we assume that, it has to be the winner under the pointer, this account will not work. The $\sqrt{1}$ derives W there, while the root $\sqrt{2}$ derives X, and therefore they do not unify in 3P.

However, if we go the other way and say that it is enough for the lexical entry to match one of the competitors, we get the correct result. That is because W (together with X and Y) is a competitor below the pointer:

(470)



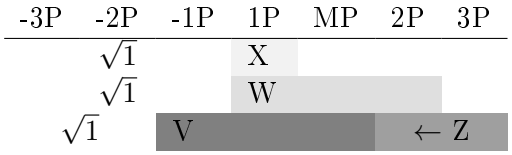
We will call this a pointee issue and, for the moment, we stay agnostic on its resolution.

To summarize, what we call a "simple pointer" approach can work for the conjoined twin pattern but requires us to make concrete assumptions about the pointee issue. We set up the pointer just above the feature that distinguishes the two paradigms, which allows us to use the Superset Principle to shed the extra feature. And we need to allow that the entry below the pointer does not have to be a winner of the derivation but can be just one of the candidates.

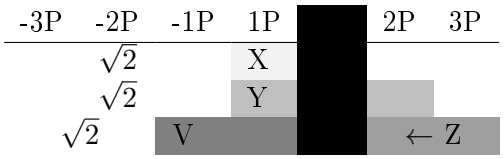
Backtracking + simple pointers

The third approach combines the two from above: backtracking and simple pointers. It can be seen in the abstractions below:

(471)



(472)



The first two rows are again derived the same as before, taking care of the first two stages of the conjoined twin pattern: starting the same and then splitting when growing over the M feature.

The third row of both roots is the one that adopts the combination of backtracking and a pointer. As before, the foot of the pointer is set up just above M, i.e., the feature in which the two roots differ. This time, however, the derivation needs to backtrack to find a lexical entry below the pointer.

After deriving 2P and adding the final feature, 3, no one can spell out the sequence from 1P to 3P. This triggers backtracking to -1P, where the V entry is found. Once V grows back to 2P, the pointer kicks in, and the whole -1P to 3P sequence will be spelled out as Z in both roots.

Furthermore, we do not run into the pointee issue that the simple pointer approach faced because W/X/Y are not in competition with Z due to Z having a lower foot. The combination of backtracking and a pointer can derive the conjoined twin pattern in our abstracted model well.

Pointer siege

Finally, the fourth possible way to go about the conjoined twin pattern is what we will call a "pointer siege". Technically speaking, this is a more complex version of the simple pointer approach above. Pointer siege is a setting where pointers enclose a lexical entry without a pointer, as can be seen in (473) and (474):

(473)

-3P	-2P	-1P	MP	1P	2P	3P
$\sqrt{1}$		α		$\leftarrow$ X		
$\sqrt{1}$		W				
$\sqrt{1}$		α		$\leftarrow$ Z		

(474)

-3P	-2P	-1P		1P	2P	3P
$\sqrt{2}$		α		$\leftarrow$ X		
$\sqrt{2}$		Y				
$\sqrt{2}$		α		$\leftarrow$ Z		

In the abstractions above, the first and third rows for each of the two roots have pointers. The second lines do not.

The way this derivation starts is that both roots are of the same size -2P, and so when they reach -1P, they both combine with the lexical entry for α:

(475)

-3P	-2P	-1P	MP	1P	2P	3P
$\sqrt{1}$		α				

(476)

-3P	-2P	-1P		1P	2P	3P
$\sqrt{2}$		α				

After merging M in the $\sqrt{1}$ derivation, the root still keeps the suffix spelled out by α. That is possible due to the Superset Principle, which can discard layers from the top.

The second root has a gap in that place. In the lexicalization tables, this will look as follows:

(477)

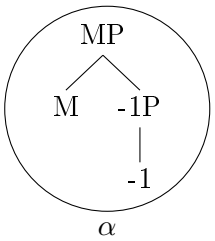
-3P	-2P	-1P	MP	1P	2P	3P
$\sqrt{1}$		α				

(478)

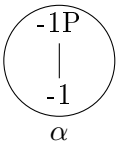
-3P	-2P	-1P		1P	2P	3P
$\sqrt{2}$		α				

For demonstration, compare the lexicalization tables to the tree structures below. The α from the lexicalization in (477) translates to the tree structure in (479), whereas the one in (478) translates to (480). They are both the same lexical entry, but the smaller structure uses the Superset Principle to shed the MP layer:

(479)



(480)



Continuing with the derivations, once reaching 1P, both roots hit a pointer saying that if the entry below is α, then the whole -1P to 1P stretch will be spelled out as X:

(481)

-3P	-2P	-1P	MP	1P	2P	3P
$\sqrt{1}$		α		$\leftarrow$ X		

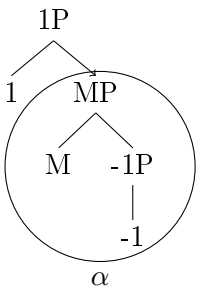
(482)

-3P	-2P	-1P		1P	2P	3P
$\sqrt{2}$		α		$\leftarrow$ X		

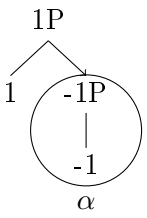
The derivation is still the same for both roots despite differing in the specification for MP because the pointer is set up just above M and allows us to shed the top layer through the Superset Principle.

In the tree structures, this stage looks as follows. First, we merge 1P to each of the derivations:

(483)

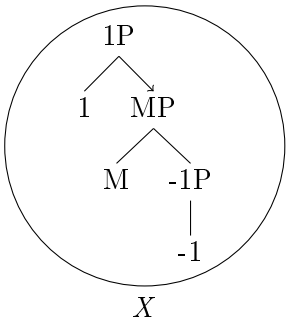


(484)

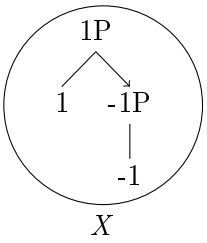


And since we arrived at the environment for which a pointer is specified (where the environment has 1P above a structure spelled out as α), both entries are spelled out as X in their entirety:

(485)



(486)



This is the final form for the first line of our lexicalization tables.

The 2P stage builds on the 1P stage above. However, when the spellout reaches 2P, X can no longer spell all the features out. At that point, the perfect match for the root $\sqrt{1}$ is W, whereas for the root $\sqrt{2}$ is it Y:

(487)

-3P	-2P	-1P	MP	1P	2P	3P
$\sqrt{1}$		α		$\leftarrow$ X		
$\sqrt{1}$		W				

(488)

-3P	-2P	-1P		1P	2P	3P
$\sqrt{2}$		α		$\leftarrow$ X		
$\sqrt{2}$		Y				

The difference is due to the gap in MP. Every time before, M was a top layer of a separate lexical entry, α, so we were able to discard it. But since MP is in the middle of the functional sequence of W and Y and there is no pointer, this is not possible here.

Note that W would be able to spell out -1P to 1P, i.e., what X spelled out in the derivation of 1P for the root $\sqrt{1}$. However, X wins over W because it is a better match. In the same manner, Y would be able to spell out the first line for the root $\sqrt{2}$, but again, X will win.

Finally, in the derivation of 3P, neither W nor Y can spell the structure out anymore, and that is when Z comes into play as the only candidate:

(489)

-3P	-2P	-1P	MP	1P	2P	3P
$\sqrt{1}$		α		$\leftarrow$ X		
$\sqrt{1}$		W				
$\sqrt{1}$		α		$\leftarrow$ Z		

(490)

-3P	-2P	-1P		1P	2P	3P
$\sqrt{2}$		α		$\leftarrow$ X		
$\sqrt{2}$		Y				
$\sqrt{2}$		α		$\leftarrow$ Z		

By getting Z for both roots, we manage to derive the conjoined twin pattern. So it seems that the pointer siege account works for us.

There is one issue, though, and it has to do with competition. While Z is the only candidate for both of the third rows, it can also spell out both of the second rows. And, once again, the result depends on our assumptions about the details of the matching procedure. The assumption concerns what we will call a competition issue, and it has to do with who wins when a pointer lexical entry competes with a non-pointer lexical entry.

If we assume that pointers always win, then Z should win over W/Y in the second lines. This would fail to derive the conjoined twin pattern because we would get X – Z – Z instead of the expected X – W/Y – Z. On the other hand, if our assumption is that pointers are irrelevant for a competition and it is, instead, a lexical entry with less junk that wins, we derive the right pattern. Both W and Y are better matches in the second rows due to not having any superfluous features.[63]

The conclusion here is that the pointer siege can also derive the conjoined twin pattern, assuming that in the competition issue, we go for the Elsewhere Condition over the "pointers win" view.

To sum up, we have seen four approaches to deal with the conjoined twin pattern. Backtracking on its own cannot work because there will always be a gap in the place of the feature in which the two structures differ. The other three approaches work, although we need to adopt specific assumptions for two of them. In the simple pointer approach, we need to say that pointers can point to all competitors in the spellout and not just to winners. For the pointer siege approach, we have to assume that in a competition between an entry with a pointer and without it, it is the one with fewer superfluous features that wins. The combination of a simple pointer with backtracking seems to work without any issues.

[63]The third logical option is that pointers always lose in competition with non-pointer lexical entries. This possibility is explored (in slightly differing ways) in Taraldsen (2019) and Caha (2021b). We will not delve into it here because it would further complicate the exposition. However, note that it would not derive the conjoined twin pattern in the pointer siege approach, as sketched in (489) and (490). The assumption would cause W/Y to spell out also the 1P derivations in the first lines, never allowing X to surface. Therefore the final pattern would be W-W-Z, and Y-Y-Z, respectively.

In the remainder of this section, we look at the issues that arise once we try to incorporate the above approaches with actual data. First, let us repeat the two paradigms in focus:

(491) SURFACE FORMS OF 'WOMAN' AND 'COLLEAGUE (MASC)'

	'woman'	'colleague'
NOM	žen-a	koleg-a
ACC	žen-u	koleg-u
GEN	žen-y	koleg-y
LOC	žen-ě	koleg-ovi
DAT	žen-ě	koleg-ovi
INSTR	žen-ou	koleg-ou

It would seem that the easiest way to account for the pattern is with the lexical entries below:

(492) LEXICALIZATION TABLE OF 'WOMAN' (TO BE REVISED)

	XP	Gend	Mkd	M	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	žen			a									
ACC	žen			u									
GEN	žen			y									
LOC	žen		ě										
DAT	žen		ě										
INSTR	žen	β		← ou									

In (492), we needed to add a new feature to the functional sequence, which we call M for now. Without it, we have no place to set up the foot of the genitive suffix *-y*. If we set it up in Cl1, we would get a wrong derivation for the genitive of *hrad* 'castle'. There, we use a subset of the lexical entry for -U, which is, however, specified for features up to K5:

(493) LEXICALIZATION TABLE OF -U

XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
		U								

If *-y* were also footed in Cl1 and would end in K3, it would be a better match in the genitive, and we would, incorrectly, derive *hrad-y*. Putting the foot of *-y* anywhere higher up, the suffixes would also crash with other lexical entries we already proposed.

At the same time, at first blush, we do not want to pull the foot of *-y* in Mkd (or lower) because we specified Mkd for feminine roots and *koleg* 'colleague' is masculine. The Mkd feature would prevent us from having the suffixes shared between the two paradigms. Hence, an extra feature is needed. This is just the first, most apparent reasoning. We will return to it and adjust it once we see more data.

The idea for the rest of the derivation is that in the locative, *-y* can no longer spell the structure out and backtracks to find *-ě*. It backtracks even more in the instrumental, ultimately taking the pointer lexical entry *-ou*.

The entries for the masculine root *koleg* 'colleague' would then look as follows:

(494) LEXICALIZATION TABLE OF 'COLLEAGUE' (TO BE REVISED)

	XP	Gend	Mkd	M	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	koleg			a									
ACC	koleg			u									
GEN	koleg			y									
LOC	koleg	ovi											
DAT	koleg	ovi											
INSTR	koleg	β		← ou									

Most of the suffixes are shared with *žen* 'woman', and therefore have the same lexical entries. The only difference is in the locative and dative. While the root *žen* only backtracks to Mkd and takes *-ě*, the root *koleg* is not specified for Mkd and therefore backtracks further to Gend and takes *-ovi*. In other words, the gender feature in which they differ is the one that makes them split. In the instrumental, they unify again because the only lexical entry able to spell the instrumental out is *-ou*, and the pointer allows us to leave Mkd out in masculines.

Since the lexical entry for the instrumental *-ou* can also spell out the locative and dative, there is a potential issue with the competition between the locative/dative and the instrumental. We saw the same when talking about a simple pointer approach, and there are at least two ways to deal with it. Either we specify that when lexical entries with and without pointers compete, it is the one with fewer superfluous features that wins, or we introduce a new feature M0 below Gend and pull the foot of the instrumental β there:

(495) LEXICALIZATION TABLE OF 'COLLEAGUE' (TO BE REVISED), LOC–INSTR

	XP	M0	Gend	Mkd	M	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
LOC	koleg		ovi											
DAT	koleg		ovi											
INSTR	koleg	β			← ou									

That way, *-ou* is not specified for the same features as the locative/dative *-ě*/*-ovi* and therefore, the competition does not arise.

We do not commit to either of the two ideas because, as it turns out, this approach does not work. While the lexical entries suggested above work independently, they fail once we incorporate them with the rest of the paradigms. Let us repeat the entries for *žen-a* 'woman':

(496) LEXICALIZATION TABLE OF 'WOMAN' (TO BE REVISED)

	XP	Gend	Mkd	M	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	žen			a									
ACC	žen			u									
GEN	žen			y									
LOC	žen		ě										
DAT	žen		ě										
INSTR	žen	β		← ou									

The tricky part is how to enforce backtracking in the locative. Given the other lexical entries we suggested so far, this will never happen. Remember that one of the entries in our tentative lexicon is -U – we just saw it a few paragraphs above and repeat it again below. The -U can be found in all the paradigms we worked with in the main section, e.g., in the M-sized *sak-o* 'jacket'. The entry for -U starts in Cl1:

(497) LEXICALIZATION TABLE OF -U

XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
		U								

The way the spellout algorithm is designed, we first try to merge F, then the Spec movement, then the complement movement, and only if all this fails do we opt for backtracking. What this comes to, on the surface, is that entries are extended as long as possible. Once they cannot extend, i.e., once the structure cannot have a monomorphemic spellout, morpheme stacking is tried – and if that does not work, the structure backtracks.

In the nominative, accusative, and genitive in the *žena* 'woman' paradigm, we have been able to keep growing the suffix (while pulling the root up through the Spec movement after every Merge). This is suddenly not an option in the locative because there is no lexical entry in our lexicon that would spell out the M to K4 stretch. So the next attempt is stacking. And that is possible. What the spellout algorithm derives is having *-a* to spell out M and then -U all the way up to K4. Hence the derivation we would expect is this:

(498) LEXICALIZATION TABLE OF 'WOMAN' (TO BE REVISED)

	XP	Gend	Mkd	M	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	žen			a									
ACC	žen			u									
GEN	žen			y									
LOC	žen			a	U								
DAT	žen			a	U								
INSTR	žen	β		← ou									

So the backtracking for *-ě* in the locative never gets triggered, and we derive the wrong suffixes. Analogically, the same happens for the root *koleg* 'colleague' and its attempt to

backtrack for *-ovi*.[64]

As far as we can tell, there is no easy, obvious way to remedy the problem. Even if we set up more mysterious features, the result of this kind of design would always be the same, namely stacking in the locative and dative.

There is also a second issue with how we suggested the lexical entries for *žen* and *koleg*. Interestingly, the second issue might give us just the right clue for adjusting some of our assumptions.

This issue has to do with the lexical entry for the instrumental. In Czech, the instrumental is gender-based. Below, we can see a representative set of examples where it shows that the feminines take *-ou*, while non-feminines take *-em*. The only paradigm that does not follow the above split is *koleg-a*, which takes the *-ou* despite being masculine:[65]

(499) -OU VS. -EM INSTRUMENTALS

	'woman'	'colleague'	'sir'	'castle'	'jacket'
	F	M.Anim	M.Anim	M.Inanim	N
INSTR	žen-ou	koleg-ou	pán-em	hrad-em	sak-em

This poses a challenge for the account because the way we set up our system is that *koleg* 'colleague' has the same functional sequence as *sak* 'jacket', or any other masculine or neuter noun. So when any non-feminine root backtracks in the instrumental, it should always find the same lexical entry, and that includes *koleg*. Below, we have just one example of a possible setup of the instrumental entries, but any other similar setup would also go wrong:

(500) LEXICALIZATION TABLE OF 'COLLEAGUE' AND 'JACKET', INSTR

	XP	M0	Gend	Mkd	M	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
INSTR	koleg		β		← ou									
	sak	em												

With *-ou* and *-em* specified for the features as suggested above, both roots would find *-ou* in an actual derivation due to its higher foot, and *sak* 'jacket' would never be able to select *-em*. This issue persists no matter what we do with the feet of *-ou* or *-em* because of the same feature specification.

However, the unexpected instrumental of the *koleg-a* 'colleague' paradigm offers us a clue about where to take the account to at least partially salvage it. We need two things: first, the roots in the *koleg-a* 'colleague' paradigm also need to be specified for the Mkd

[64] Note that one theoretical option is to set up a lexical entry that only spells out MP and is pronounced as elements AI. This would lead to the Element Theory computation of A + I with the following U. Since I and U cannot combine, U would "disappear", and on the surface, we would get (a potentially palatalizing) *-e* as a result of A + I. The logic of combining the elements is the same as we used in deriving the paradigm *moř* 'sea'. The nice outcome is that it would possibly derive the correct DAT/LOC form *žen-ě*. We do not pursue this option, though, because the above setup would mean we get the same *-ě* in the paradigm *koleg* 'colleague', which is incorrect.

[65] Feminines can also take *-í*, as in *dlaň-í* 'palm'. Later on, we will suggest that both *-í* and *-ou* have the same underlying representation, U+length, and therefore we treat them as the same morpheme.

feature, together with feminines, and second, the masculine animates need to be specified for an animacy feature.

The first part allows us to solve the instrumental puzzle. Allowing *koleg* to have Mkd in its functional sequence means it will be featurally different from other non-feminines, and hence we can suggest instrumental entries like the ones below:

(501) LEXICALIZATION TABLE OF 'WOMAN', 'COLLEAGUE' AND 'JACKET', INSTR

	XP	Gend	Mkd	M	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
INSTR	žen		ou										
	koleg		ou										
	sak	em											

Both *žen* and *koleg* roots share Mkd, so they select the same suffix, *-ou*. Other non-feminines, like *sak*, lack Mkd and therefore need to backtrack even lower to find an instrumental entry, and end up with *-em*.

The data supporting the idea of Mkd being relevant for the *koleg-a* paradigm comes from older stages of Czech and modern Czech dialects. In the old Czech (from the 12th to the 15th century), all roots taking *-a* in the nominative singular were fully unified – not differing in any case, not even in the dative and locative:[66]

(502) OLD CZECH SURFACE FORMS OF THE A-STEMS, 'WOMAN' AND 'DUKE', Pleskalová (2001)

	'woman'	'duke'
NOM	žen-a	vévod-a
ACC	žen-u	vévod-u
GEN	žen-y	vévod-y
LOC	žen-ě	vévod-ě
DAT	žen-ě	vévod-ě
INSTR	žen-ú	vévod-ú

Most of the other paradigms were declinable depending on their gender but not a-stems (including their variants ja-stems and -ьja-stems) – these contained both masculines and feminines.

Furthermore, even in the old Czech, there was a gender-based difference in the instrumental outside of a-stems. The masculines took *-em* or *-ím*, while the feminines took *-ú*. And since the masculine a-stems took the otherwise feminine ending, we go with the explanation that they unexpectedly had the Mkd feature – which persisted in standard Czech to this day.

If *koleg-a* 'colleague' lacked Mkd, we would expect it to take the *-em* ending. This indeed happens in some modern Czech dialects. An example would be the Brno dialect (Balhar 2002). There it looks like the old Czech state of the features was altered and – given its masculine suffix – *koleg-a* lost Mkd.

[66] The specifically masculine suffixes, like the modern Czech *-ovi* for the locative and dative singular, started to differentiate the masculine a-stems from the feminine a-stems in the 15th century.

To sum up the instrumental discussion, we can solve the puzzle of why *koleg* and *sak* are taking different instrumentals by positing that *koleg*, unlike other non-feminine paradigms, has the Mkd feature.[67]

Specifying the root *koleg* for Mkd has a drawback as well – and it will lead us to the second part of the puzzle, which is adding the animacy feature. The thing is that now it is *koleg* 'colleague' and *žen* 'woman' who have the same featural specification. This is an issue since they are supposed to start the same (taking *-a*, *-u* and *-y* from the nominative to genitive) but then somehow split. Previously, we handled the split through backtracking over Mkd, so they had to take a different suffix. However, since they both have Mkd now, this is no longer an option.

Remember that the suffix in the dative and locative is *-ě* for the feminines and *-ovi* for masculines:

(503) SURFACE FORMS OF 'WOMAN' AND 'COLLEAGUE', LOC–DAT COMPARISON

	'woman'	'colleague'
NOM	žen-a	koleg-a
ACC	žen-u	koleg-u
GEN	žen-y	koleg-y
LOC	žen-ě	koleg-ovi
DAT	žen-ě	koleg-ovi
INSTR	žen-ou	koleg-ou

We take a clue from the form *-ovi*.

Masculine animates can take two suffixes in the locative and dative: *-ovi* and/or *-u*. The *-u* suffix can also be found in masculine inanimates and neuters (and, as we argued, underlyingly also in the feminines), but *-ovi* is strictly tied to masculine animates:

(504) -OVI VS. -U LOCATIVES AND DATIVES

'colleague' Ma	'sir' Ma	'castle' Mi	'jacket' N
koleg-ovi	pán-ovi		
	pán-u	hrad-u	sak-u

We can also see the *-ov-* marker in masculine possessives:

(505) syn- ov- a lžičk- a
son- MASC.POSS- FEM.AGR spoon- FEM
'son's spoon'

We can playfully use it with masculine inanimates, too, but only as a personification. This can be seen, e.g., in stories for kids. Take the word *burák* 'peanut', which is under

[67] At some point, we will need to come back to the potential issue of how come *koleg* still takes the masculine agreement, as in *dobr-ý koleg-a* 'good-MASC colleague'. Ultimately, the answer will have to be that it is not the presence or absence of the Mkd feature that controls the agreement. However, we leave this question aside for further research.

normal circumstances inanimate and as such only ever takes *-u* in the locative and dative singular. However, if made animate by using it as a name of an animate entity (like in the stories, where "Burák" is a name of a lively breakdown truck on Veselepohadky.cz), it suddenly gets the *-ov-* suffix:

(506) Burák- ov- y povídačk- y
peanut- MASC.POSS- PL.AGR tale- PL
'Peanut's tales' (personification)

What it points to is that *-ov-* is firmly tied to (masculine) animacy. Based on this, we will suggest that there is also an animacy feature somewhere in the structure and that only masculines have it. This will create the necessary featural difference between *koleg* and *žen* and will allow us to explain why they split in the locative.[68]

The question now is where the Anim feature sits in the functional sequence. While we do not have a fully functional account of the two paradigms at hand, we will show two directions in which we could take it. In the first one, the Anim feature is low in the sequence, just above Mkd, and the account takes the pointer siege approach. In the second one, Anim is high in the structure, in the middle of the case domain, and explores the pointers + backtracking approach:

		low-Anim					**high-Anim**				
XP	Gend	Mkd	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
		pointer siege					**pointer + backtracking**				

[68] Note that feminines do not express animacy in the nominal domain of Czech, which allows us to suggest that only masculines take the Anim feature there. However, at some point, we will need to come back to the fact that there is one place in Czech grammar where feminines show animacy, and that is in possessives. A feminine possessive marker *-in-* can only be used with animate roots, whereas inanimates have to use a genitive construction:

(507) žen- in- a noh- a
woman- FEM.POSS- FEM.AGR leg- FEM
'woman's leg'

(508) *židl- in- a noh- a
chair- FEM.POSS- FEM.AGR leg- FEM
'the leg of the chair'

The direction we could take is to look into different types of animacy, like virility in Upper Sorbian (Baltoslav: Upper Sorbian Grammar). That might lead to decomposing the animacy feature into several projections, each further specifying/restricting the meaning of animacy. Then, we could, e.g., suggest that the feminines have the simplest animacy feature, which is hidden in their root in the nominal domain, and therefore never influences the morphology but gets exposed in possessives. The masculines would have the virility feature on top of the simple animacy feature, and it would be this virility feature that makes the difference in the locative and dative singular. We leave the details and technical implementation for further research.

6.1.2 Low-animacy approach

We will first explore the low Animacy approach. Below, we show our best guess at the lexical entries. We will point out a few issues with them later on:

(509) LEXICALIZATION TABLE OF 'WOMAN', LOW-ANIMACY APPROACH

	XP	Gend	Mkd	Anim	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	žen		*β*		← a								
ACC	žen		*β*		← u								
GEN	žen		*β*		← y								
LOC	žen		ě										
DAT	žen		ě										
INSTR	žen		*β*		← ou								

(510) LEXICALIZATION TABLE OF 'COLLEAGUE', LOW-ANIMACY APPROACH

	XP	Gend	Mkd	Anim	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	koleg		*β*		← a								
ACC	koleg		*β*		← u								
GEN	koleg		*β*		← y								
LOC	koleg		ovi										
DAT	koleg		ovi										
INSTR	koleg		*β*		← ou								

The Anim feature is low here, just above the Mkd feature. The setup of the lexical entries makes use of the pointer siege approach, with the locative/dative *-ě* and *-ovi* being encircled by entries with pointers. Let us take the root *žen* to show how the derivation proceeds.

To derive the nominative, the root *žen* spells out everything up to Gend. Then, after merging Mkd, the root can no longer spell the structure out, and the best candidate to spell out only MkdP is *β*:

(511) LEXICALIZATION TABLE OF 'WOMAN', NOM, INTERMEDIATE STEP

	XP	Gend	Mkd	Anim	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	žen		*β*										

The other two candidates, *-ě* and *-ovi*, have too many superfluous features, which is why they lose the competition.

The Anim feature gets skipped in the feminine root, so in the next step, Cl1 is merged. At that point, *β* can no longer spell out the structure. However, *all* suffixal lexical entries that we see in tables (509) and (510) are candidates here (except for *-ovi* which contains the Anim feature). Given the features to spell out, *-a* will win the competition. Out of all the competing lexical entries, it has the fewest superfluous features:

(512) LEXICALIZATION TABLE OF 'WOMAN', NOM, INTERMEDIATE STEP

	XP	Gend	Mkd	Anim	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	žen		*β*		← a								

After that, the structure keeps merging more features until it reaches K1, and the suffix that the root *žen* derives in the nominative is *-a*:

(513) LEXICALIZATION TABLE OF 'WOMAN', NOM, FINAL FORM

	XP	Gend	Mkd	Anim	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	žen		*β*		← a								

In the accusative, the derivation goes the same, only at the end, K2 is also merged into the structure. At that point, *-a* can no longer spell it out, and *-u* wins. The same for *-y* in the genitive:

(514) LEXICALIZATION TABLE OF 'WOMAN', NOM–GEN

	XP	Gend	Mkd	Anim	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	žen		*β*		← a								
ACC	žen		*β*		← u								
GEN	žen		*β*		← y								

Now, in the locative, K4 gets merged, and *-y* can no longer spell it out. At this point, there are only two candidates, *-ě* or *-ou*. This is precisely the situation we talked about in the part about the pointer siege approach. The account can work, but we have to posit that in the competition between entries with and without pointers, it is those with fewer superfluous features that win. Adopting such an assumption helps to decide the *-ě* vs. *-ovi* competition in favor of *-ě*, allowing us to derive the correct structure in the locative and dative:

(515) LEXICALIZATION TABLE OF 'WOMAN', NOM–DAT

	XP	Gend	Mkd	Anim	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	žen		*β*		← a								
ACC	žen		*β*		← u								
GEN	žen		*β*		← y								
LOC	žen		ě										
DAT	žen		ě										

Finally, in the instrumental, there is no longer any other candidate besides *-ou* after merging K6. So we get the correct form *žen-ou*:

(516) LEXICALIZATION TABLE OF 'WOMAN', NOM–INSTR

	XP	Gend	Mkd	Anim	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	žen		*β*		← a								
ACC	žen		*β*		← u								
GEN	žen		*β*		← y								
LOC	žen		ě										
DAT	žen		ě										
INSTR	žen		*β*		← ou								

Note that all the pointer entries are shared between the roots *žen* 'woman' and *koleg* 'colleague'. Only the two entries without pointers are unique for each of the two paradigms. This allows us to create the conjoined twin pattern because, with pointers, we can shed the Anim feature in which the paradigms differ, while the entries without pointers make that impossible.

Let us briefly come back to the instrumental entries to show that this account works well for it even after adding the Anim feature.

First, let us focus on the difference between the two roots at hand:

(517) LEXICALIZATION TABLE OF 'WOMAN' AND 'COLLEAGUE', INSTR

	XP	Gend	Mkd	Anim	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
INSTR	žen		*β*		← ou								
	koleg		*β*		← ou								

There are two ingredients to why they both can take *-ou*. First, they both contain Mkd in their functional sequence. The second ingredient has to do with the pointer setup. Having its foot just above Anim, in Cl1, means that the Animacy feature can be forgone through the Superset Principle. That is how we can spell both structures out despite them differing in their featural specification.

Something new happens in the instrumental of masculine nouns now. Before adding Anim, we did not need the *-em* instrumental to have a pointer because all masculine and neuter nouns had the same features. With adding Anim, this changes as a gap is created in Anim for the inanimate roots like *jazyk* 'tongue'. So a lexical entry without a pointer could not spell out both the masculine animate and inanimate *-em*:

(518) LEXICALIZATION TABLE OF 'WOMAN', 'COLLEAGUE', 'SIR' AND 'TONGUE', INSTR

	XP	Gend	Mkd	Anim	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
INSTR	žen		*β*		← ou								
	koleg		*β*		← ou								
	pán	*α*			← em								
	jazyk	*α*			← em								

Using a pointer makes it easy. As we have done with the instrumental *-ou* (and in other places where we used pointers), we only need to set the pointer up above the feature that

goes missing. That is why we put the foot of the pointer just above Anim. Below Anim, we set up a new lexical entry, α, which needs to start below -em. It has to be footed in a feature that both paradigms contain. That can be Gend here.

The entries in (518) work well for deriving both the *-ou*/*-em* suffixes and the animacy distinction. The roots *žen* 'woman' and *koleg* 'colleague' will take *-ou*, because – unlike *pán* 'sir' and *jazyk* – they contain the Mkd feature. On the other hand, the non-feminine paradigms like *pán* 'sir' and *jazyk* 'tongue' will not be able to take it (due to the lack of the Mkd feature), so they will backtrack more, ultimately landing at *-em*.

Below is the currently explored account repeated:

(519) LEXICALIZATION TABLE OF 'COLLEAGUE'

	XP	Gend	Mkd	Anim	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	koleg		β		← a								
ACC	koleg		β		← u								
GEN	koleg		β		← y								
LOC	koleg		ovi										
DAT	koleg		ovi										
INSTR	koleg		β		← ou								

The positives of the account are a natural animacy placement (on top of the gender domain) and having only the features we have already posited previously, without the need for any new, mysterious features. Some potential drawbacks are having a forest of pointers to derive two fairly common, productive paradigms, and the need to posit so-called "phantom" lexical entries, i.e., entries that never surface, like our α and β. We also need to state the assumption that in the competition between entries with and without pointers, it is those with fewer superfluous features that win. Whether this is correct and therefore belongs to the positives or negatives is yet to be seen.

The most serious issue is something that has not been mentioned yet, and that is that the entry for *-ovi* is now specified for Mkd. Factually, this is wrong. We already saw that *-ov(i)* is a suffix used strictly in the context of masculine animates. Having it footed in Mkd prevents any possible reuse for other masculine animates because those are not specified for Mkd. Our account of the paradigm *koleg-a* 'colleague' violates this. What follows is that if we wanted to make *-ovi* as specified here work with our account for the paradigm *pán* 'sir' (see section 5), we would have to propose there are two different *-ovi* suffixes.

Unfortunately, we currently do not see any possible way around this. Hence, we put the approach aside, unable to make it fit with the data concerning *-ov(i)*.

6.1.3 High-animacy approach

The second approach we will present has the Anim feature high up, in the middle of the case domain, and takes the backtracking + a simple pointer route:

(520) LEXICALIZATION TABLE OF 'WOMAN', HIGH-ANIMACY APPROACH

	XP	Gend	Mkd	M	Cl1	Cl2	Sg	K1	K2	K3	Anim	K4	K5	K6
NOM	žen			a										
ACC	žen			u										
GEN	žen			y										
LOC	žen			ě										
DAT	žen			ě										
INSTR	žen			β								← ou		

(521) LEXICALIZATION TABLE OF 'COLLEAGUE', HIGH-ANIMACY APPROACH

	XP	Gend	Mkd	M	Cl1	Cl2	Sg	K1	K2	K3	Anim	K4	K5	K6
NOM	koleg			a										
ACC	koleg			u										
GEN	koleg			y										
LOC	koleg			ovi										
DAT	koleg			ovi										
INSTR	koleg			β								← ou		

Besides the Anim feature placement and the backtracking + simple pointer approach, an extra difference from the previous setup is that here we need a new feature M, in between Mkd and Cl1. The reason is that if we put the foot of *-y* in Cl1, we would incorrectly derive *-y* in the genitive of *hrad* 'castle' (instead of -U), as already discussed under (496).

We also cannot pull *-y* one feature down because we would get into the same issue that broke the neck of the previous account. Having the foot in Mkd would mean that the foot of *-ovi* also has to be in Mkd (or lower), which is incompatible with the Czech nominal data. Hence we posit a new feature that we call M for the moment.

Besides this, the derivation is pretty simple. Let us show it on the root *žen* 'woman' again.

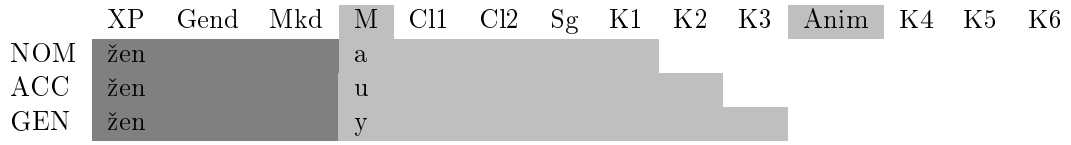
The way the nominative derivation of *žen* 'woman' goes is that the root grows up to Mkd, after which it can no longer extend, and so from M on, it takes *-a*, since *-a* is the best match in that context. The entries for *-u*, *-y*, *-ě* and *-ovi* are also candidates but have many more superfluous features which makes them worse matches. The *-a* entry keeps extending up to K1 in the nominative, and we derive *žen-a*:

(522) LEXICALIZATION TABLE OF 'WOMAN', NOM

	XP	Gend	Mkd	M	Cl1	Cl2	Sg	K1	K2	K3	Anim	K4	K5	K6
NOM	žen			a										

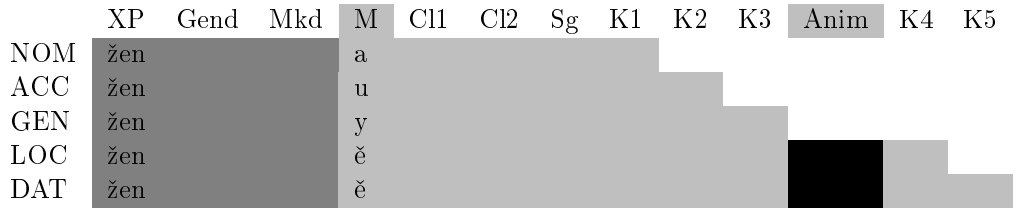
In the accusative, we merge K2 on top, which means *-a* can no longer spell the structure out, and *-u* takes over. In the genitive, the same happens for K3 and *-y*:

(523) LEXICALIZATION TABLE OF 'WOMAN', NOM–GEN



The next step is merging K4 (remember, the root *žen* does not have the Anim feature). Neither *-a*, *-u*, nor *-y* are candidates anymore because they are too small to spell the structure out. At that stage, the only candidate is *-ě*:

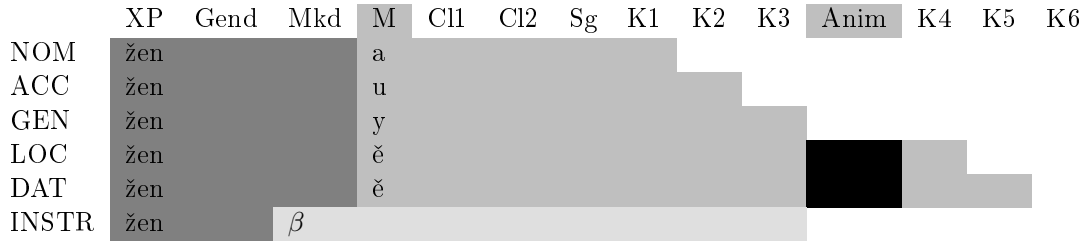
(524) LEXICALIZATION TABLE OF 'WOMAN', NOM–DAT



The reason why *-ovi* is not a candidate is that it is specified for Anim.

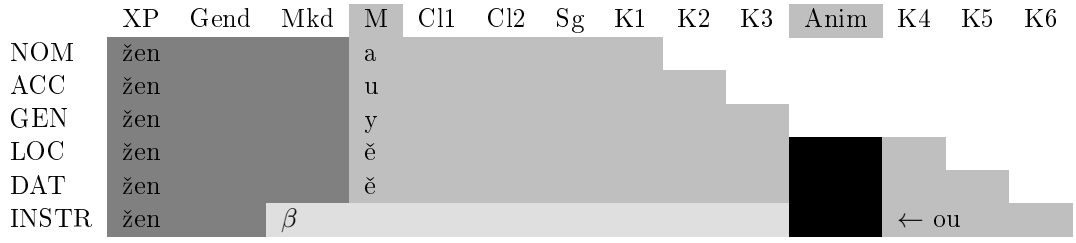
After merging K6 for the instrumental, we find ourselves in the situation when no one can spell the M to K6 stretch, so backtracking gets triggered. In Mkd, the derivation finds β that extends up to K3:

(525) LEXICALIZATION TABLE OF 'WOMAN', INSTR, INTERMEDIATE STEP



After β can no longer grow, the structure gets finished by a lexical entry with a pointer that says that if K4 points to β, the whole structure will be spelled out as *-ou*:

(526) LEXICALIZATION TABLE OF 'WOMAN', INSTR, FINAL FORM



The derivation of all the cases of the root *koleg* 'colleague' go the same, except in the locative and dative. There, *koleg* differs from *žen* in being specified for the Anim feature, and therefore chooses *-ovi* over *-ě*.

Note that the instrumental works well here, too. The *žen* vs. *koleg* feature difference is solved by having the pointer just above Anim. That allows us to shed Anim in the derivation of the instrumental of *žen*:

(527) LEXICALIZATION TABLE OF 'WOMAN' AND 'COLLEAGUE', INSTR

	XP	Gend	Mkd	M	Cl1	Cl2	Sg	K1	K2	K3	Anim	K4	K5	K6
INSTR	žen		β								■	← ou		
	koleg		β									← ou		

At the same time, it causes no issues with the non-feminine instrumental *-em*. The non-feminines backtrack even lower because they are not specified for Mkd. They find α in Gend. Their pointer is also set up just above Anim, and that way, we can easily model the animacy difference between *pán* 'sir' and *jazyk* 'tongue' as well, just as the fact that the animacy difference does not reflect in selecting different instrumental markers:

(528) LEXICALIZATION TABLE OF 'WOMAN', 'COLLEAGUE', 'SIR' AND 'TONGUE', INSTR

	XP	Gend	Mkd	M	Cl1	Cl2	Sg	K1	K2	K3	Anim	K4	K5	K6
INSTR	žen		β								■	← ou		
	koleg		β									← ou		
	pán	α	■									← em		
	jazyk	α	■								■	← em		

The positives of this account are that there is just one pointer in each paradigm and perhaps, that we do not need to commit to any assumptions about the competition between entries with pointers and without them. The competition issue never arises in the backtracking + a simple pointer account because the entries with pointers and without them never get into the competition due to having different feet.

The possible success of this approach is its treatment of the *-ovi* morpheme. In the low Animacy account, we found ourselves in the situation when *-ovi* was unexpectedly (and wrongly) specified for Mkd. This does not happen here because the foot of *-ovi* is above Mkd.

There is also some chance to capture the *-ovi*/*-u* distribution in the masculine animates. Remember that while generally any paradigm of any gender can take *-u* in the locative and dative, most masculines animates can also take *-ovi*. However, the paradigm *koleg-a* is exceptional in the masculine animate realm in only taking *-ovi* in these two cases, and never *-u*:

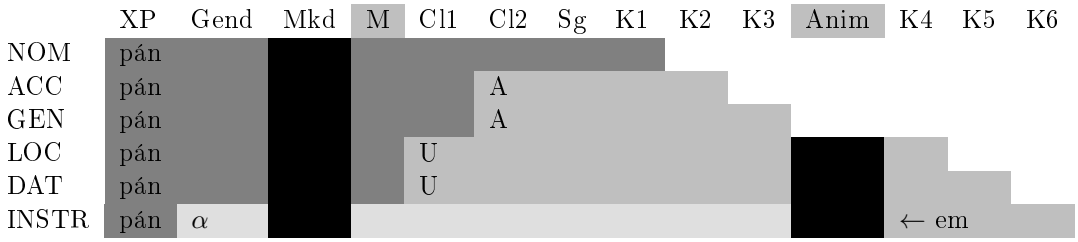
(529) -OVI VS. -U LOCATIVES AND DATIVES (REPEATED)

'colleague'	'sir'	'castle'	'jacket'
Ma	Ma	Mi	N
koleg-ovi	pán-ovi		
	pán-u	hrad-u	sak-u

While the high Animacy account still does not offer a full account of the distribution, it offers a possible direction that could shine a light on two questions. First, how come that most masculine animates, like *pán* 'sir', can take both *-u* and *-ovi*. And second, why *koleg* 'colleague' can only take the latter.

First, let us look at how *pán* 'sir' can take both suffixes. Below, we can see the lexical entries for the paradigm (its complex root shape does not play a role here):

(530) LEXICALIZATION TABLE OF 'SIR'



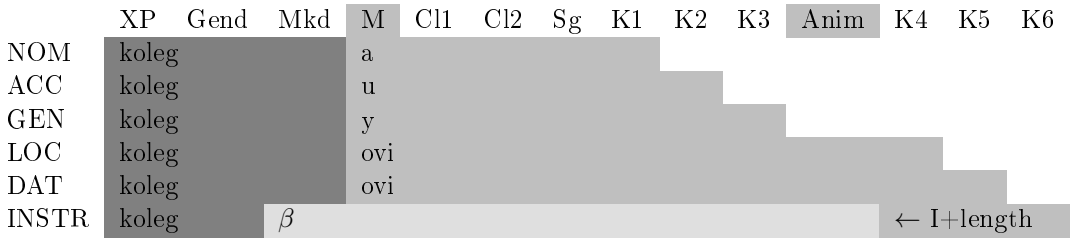
The way *pán* gets -U in the locative and dative is through backtracking from -A while at the same time lacking the Anim feature. Now look at the lexical entry for *-ovi*:

(531) LEXICALIZATION TABLE OF -OVI

In cases where the root *pán* is specified for Anim, the derivation up up to the genitive goes the same, but then Anim is inserted, and -U can no longer spell the structure out due to lacking the Anim feature. So the structure backtracks even more, ultimately hitting *-ovi* in M. Since *-ovi* can spell out the stretch from M to K4/K5, we derive *pán-ovi*.

The root *koleg* 'colleague' is too small for this, though:

(532) LEXICALIZATION TABLE OF 'COLLEAGUE'



It only grows to Mkd and therefore will never be able to reach Cl1 where -U is footed since M would not be spelled out:

(533) LEXICALIZATION TABLE OF -U

	XP	Gend	Mkd	M	Cl1	Cl2	Sg	K1	K2	K3	Anim	K4	K5	K6
DAT					U									

One other part that comes out well is that the account more or less explains why only masculines take *-ovi*. It is because *-ovi* is specified for Anim, and the only paradigms that have the Anim feature are the masculine paradigms.[69]

Of course, one issue is that now *koleg* differs from *pán* in that *pán* can sometimes come with Anim and sometimes without, while *koleg* cannot, and we have no idea why. If *koleg* also came without Anim, we would not be able to use -ovi in the locative and dative, and the structure that would be ultimately derived is *koleg-a-U*. This would give the surface form *koleg-o*, which is incorrect in this context.

The few issues and an exact account of the *-ovi*/*-u* distribution aside, it is still true that, unlike the low Animacy approach, this at least gives the possibility of one because *-ovi* is no longer specified for Mkd.

The drawbacks of the high Animacy account are having an extra, mysterious feature M and a very unexpected Anim placement. We need to treat Anim as part of the case domain. It only gets inserted once the structure grows beyond K3, which itself has three potential shortcomings. First, we need a "phantom" lexical entry in the instrumental, which is less than in the low-Anim approach but still not ideal. Second, it is very substandard to treat animacy this way. And third, what follows from the setup is that now we cannot make animacy distinction in the cases below the current place for the Anim feature. The question then is whether we need to.

One place where Czech seems to make the animacy distinction in the lower cases is the singular. We see a nominative-accusative syncretism in the masculine inanimates but an accusative-genitive syncretism in the masculine animates, as already discussed in detail in section 5.1:

[69] Let us briefly come back to our account of the special inanimates that take the GEN-ACC syncretism, as proposed in the section 5. We suggested that the unexpected GEN-ACC pattern has nothing to do with animacy but with the shape of the lexical entries. However, proposing that *-ovi* spells out the animacy feature opens up a new issue. While there are inanimate nouns with the GEN-ACC pattern that do not seem to combine with *-ovi* (e.g., *Mluvila o svém ??**koníčk-ovi*** 'She talked about her hobby'), there are some that do combine with it (e.g., *Půjdu až po **panák-ovi*** 'I'll only go after a shot (in drinking)'). This does not invalidate the DLB tree account. Instead, it opens up a new question about the exact nature of the special inanimates. We leave it aside for the time being.

(534) (possibly) animacy-driven difference in structurally lower cases in the singular

SG	Ma	Mi
	'sir'	'tongue'
NOM	pán	jazyk
ACC	pána	jazyk
GEN	pána	jazyka

However, we suggested that this is not an actual animacy distinction but merely a result of the shape of lexical entries spelling out the animate roots. If that is on the right path, this example does not constitute a counterpoint against the high-Anim account.

Another commonly accepted place of animacy distinction in lower cases of nouns is in the plural. There, the inanimates syncretize between the nominative and accusative, whereas the animates do not:

(535) (possibly) animacy-driven difference in structurally lower cases in the plural

PL	Ma	Mi
	'sir'	'tongue'
NOM	páni	jazyky
ACC	pány	jazyky

It will require more research to see whether this is an actual animacy distinction or whether it can be likened to the situation with the singular example above.

If we conclude we need the animacy distinction in the lower cases, the high-Anim approach might fail to capture it. We keep it for later to see whether that is the conclusion to be drawn.

To sum up this section, we showed the conjoined twin pattern in the paradigms *žen-a* 'woman' and *koleg-a* 'colleague'. The issue had to do with the fact that the two paradigms start the same, then split, and then reunite again. We argued that *koleg-a* has the Mkd feature in its functional sequence, despite being a masculine animate. This allowed us to suggest an account where both paradigms at hand can share a suffix *-ou* in the instrumental without it causing trouble for the non-feminine instrumental *-em*. After this, we needed to add the Anim feature that allowed us to derive different suffixes for *žen* and *koleg* in the locative and dative.

We presented two possible accounts. One had the Anim feature low, just above Mkd, and took the pointer siege route. The other had the Anim feature high, above K3, and adopted the backtracking + simple pointer approach. Both accounts, however, had some issues preventing us from incorporating them into our system in a wholly satisfying manner. The low-Anim account faced the problem of having *-ovi* specified for Mkd and therefore being unable to account for it in other masculine animate paradigms. The high-Anim account predicted that there should not be any animacy distinctions in nouns in the nominative, accusative, and genitive, which might be incorrect.

For the moment, we do not see a way to make the *žen-a* and *koleg-a* paradigms fully work, so we leave it as an open issue.

6.2 Doubly mysterious case of the woman with the rose

This section touches upon the other pair of paradigms mentioned in the beginning, *žen-a* 'woman' and *růž-e* 'rose'. They are both feminine, and the first one has a set of non-palatalized suffixes. The other belongs to the palatalized group.

We will point out an unexpected ABA pattern in *růž-e* 'rose' and suggest a solution using the Element Theory decomposition, basing ourselves on diachronic and dialectal data. The decomposition will uncover that also these two paradigms show the conjoined twin pattern and that this one is even trickier because it seemingly operates without any distinction in the functional sequence. We will show two possible ways to deal with it in an attempt to incorporate both paradigms into our system.

Let us start from the *růž-e* 'rose' paradigm alone. As can be seen through the cell color in the table below, it has what seems like an instance of a (double) ABA pattern, one in the nominative and genitive, the other in the accusative, locative, and dative:

(536) SURFACE FORMS OF 'ROSE'

	'rose'
NOM	růž-e
ACC	růž-i
GEN	růž-e
LOC	růž-i
DAT	růž-i
INSTR	růž-í

The fact that languages seem to avoid the ABA pattern has been pointed out many times for various domains (see, e.g., Bobaljik 2012 for comparatives, or Caha 2009 for the case domain). The observation led to formulating the so-called *ABA constraint, which refers to a concept of overriding that rules out syncretism across non-adjacent items:

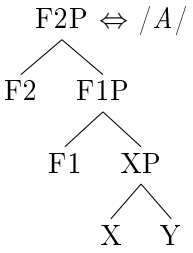
(537) *ABA VISUALIZATION

A
B
A

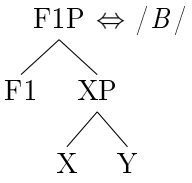
The explanation for the *ABA constraint in Nanosyntax falls out from its core principles, more concretely from the combination of the Superset Principle and the Elsewhere Condition.

Imagine that our lexicon has (only) the two lexical entries below (adapted from Caha 2019b):

(538) LEXICAL ENTRY OF A



(539) LEXICAL ENTRY OF B



Spelling out F2P is straightforward. We get A, as the lexical entry in (538) is a perfect match. Spelling out F1P is also unproblematic, as the lexical entry in (539) is an exact match. The structure gets spelled out as B. This creates the AB part of the pattern.

Spelling out only the XP now, no lexical entry in our lexicon is a perfect match. Thanks to the Superset Principle, both lexical entries above are candidates because both contain the XP node. The Elsewhere Condition then chooses the one with fewer superfluous features. In this case, it is the structure spelled out by B, creating the ABB pattern. More importantly, due to the containment (and assuming fixed functional sequence), it will always be the structure "closer" to the new structure that wins, and hence we can never create the ABA pattern.

Occasionally, we can see what seems like ABA patterns in languages. Often, the explanation boils down to accidental syncretisms (Caha 2009), which is a route we take here as well.

The *růž-e* 'rose' paradigm offers a particularly intriguing situation because – as we already saw above – it shows not one but two ABAs. So what can be said about it while still keeping the observation of the overwhelming cross-linguistic *ABA pattern?

In (540), notice the shape of the endings of the paradigm in old Czech (*duš-a* 'soul' represents the *růž-e* paradigm):

(540) 'SOUL' (RŮŽE 'ROSE' PARADIGM) IN THE MODERN STANDARD VS. OLD CZECH, Pleskalová (2001)

	'rose (modern Czech)'	'soul (old Czech)'
NOM	růž-e	duš-a
ACC	růž-i	duš-u
GEN	růž-e	duš-ě
LOC	růž-i	duš-i

There are two things to be noticed. First, the nominative and accusative suffixes of *duš-a* (*růž-e*) differ between old Czech and modern standard Czech. The modern standard Czech has *-e* and *-i* respectively, while the old Czech has *-a* and *-u*.

And second, in old Czech, there is no ABA in the *duš-a* (*růž-e*) paradigm yet. The nominative is not syncretic with the genitive. The accusative is not syncretic with the locative.[70]

[70]We take the palatalization difference in the genitive to be a regular historical change of *ě* > *e*, which

The same shape of the *růž-e* 'rose' paradigm can be seen in Czech dialects even nowadays (e.g., Velké Bílovice, East Moravia):

(541) 'ROSE' IN THE VELKÉ BÍLOVICE DIALECT, Marie Skoupilová, p.c.

	'rose'
NOM	růž-a
ACC	růž-u
GEN	růž-e
DAT	růž-i

Taking this data as a hint, we argue that while surface-wise the same, the syncretisms we see in the modern paradigm of *růž-e* do not exist underlyingly. They are only a result of different elements merging, creating accidental homophony.

Concretely, we suggest that *růž-e* has the same floating I element at the end of the root as we proposed for all the other palatalized paradigms:

(542) růžI-

With -A as the nominative and -U as the accusative element, precisely as in the diachronic and dialectal data, we get the right results:

(543) DECOMPOSITION OF NOM/ACC OF 'ROSE'

	'rose'	
NOM	růžI-A →	růž-e
ACC	růžI-U →	růž-i

Decomposing the suffixes into different sets of elements helps us explain away the double-ABA in *růž-e*. We suggest the following decomposition of the paradigm:

(544) FULL DECOMPOSITION OF THE 'ROSE' PARADIGM

	surface		decomposition
NOM	růž-e	→	růžI-A
ACC	růž-i	→	růžI-U
GEN	růž-e	→	růžI-AI
LOC	růž-i	→	růžI-I
DAT	růž-i	→	růžI-I
INSTR	růž-í	→	růžI-U+length

All the underlying suffixes are precisely what we see in the diachronic and dialectal data. Under this decomposition, both ABAs vanish. The only syncretism left is the ABA-compatible locative/dative. Hence treating *růž-e* this way suggests that the apparent ABA pattern might indeed be a case of accidental homophony.

happened in several stages in the second half of the 14th century and in the beginning of the 15th century. In this period, the short *ě* lost its palatalization and unified with *e* in front of the "soft" consonants *j*, *ň*, *ď*, *ť*, *ž*, *š*, *č*, *z*, *s*, *c*, *ř* (Pleskalová 2001).

One place worth pausing at is the decomposition in the instrumental. We have seen two different instrumental endings in the feminines: *-ou* (in *žen-a* 'woman') and *-í* (in *růž-e* 'rose', among others):

(545) THE INSTR OF 'WOMAN' AND 'ROSE'

žen-ou
růž-í

Note that both suffixes are lengthened. In the first instance, the length is marked by the acute accent over *-i*. In the second instance, the falling diphthong *-ou* is a long version of *-u* (see, e.g., Ziková 2018).[71] Since *žen-a* is non-palatalized, we take U+length as the basis for the (feminine) instrumental suffix. Combining this suffix with the floating I in the palatalized paradigms, we once again get U + I next to each other. They cannot merge, and only I surfaces and combines with the length:

(546)

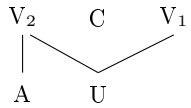
žen	+	U+length	→			žen-ou
růžI	+	U+length	→	růžI + length	→	růž-í

For the moment, we do not delve into questions about the exact phonological computation of the diphthong *-ou* in the instrumental.[72]

The decomposition uncovers a certain regularity. The paradigm *růž-e* 'rose' now shares some suffixes with the paradigm *žen-a*, suggesting they might be palatalization counterparts. However, they do not share all the suffixes, and looking at the syncretisms, we see the conjoined twin pattern appearing again. See the surface forms in (547) and the decomposed forms in (548):[73]

[71] Czech has two long counterparts of the monophthong *u*. One is a monophthong written as *ú* or *ů*, the difference between them only orthographical. The other is the diphthong *ou*.

[72] Ziková (2018) argues for the CV representation of *-ou* as shown below (simplified for our purposes):



If we are to keep the analysis that *-í* and *-ou* are counterpart suffixes, the element A has to delink (or not link at all) at some point during the derivation, otherwise, it will interfere with the result.

[73] It is not completely obvious to us how to decompose *y* in Czech. The language has two homophones, *i* and *y* – the only difference between them being that unlike *i*, *y* never causes palatalization. That is why we choose to treat them both as the element I, but *i* will have to have an extra PAL feature or palatalizing properties. The vowels *e* and *ě* could be treated in a similar fashion. Here, we leave palatalization aside as it does not seem crucial for the task at hand.

(547) SURFACE FORMS OF 'WOMAN' AND 'ROSE'

	'woman'	'rose'
NOM	žen-a	růž-e
ACC	žen-u	růž-i
GEN	žen-y	růž-e
LOC	žen-ě	růž-i
DAT	žen-ě	růž-i
INSTR	žen-ou	růž-í

(548) DECOMPOSED FORMS OF 'WOMAN' AND 'ROSE'

	'woman'	'rose'
NOM	žen-A	růžI-A
ACC	žen-U	růžI-U
GEN	žen-I	růžI-AI
LOC	žen-AI	růžI-I
DAT	žen-AI	růžI-I
INSTR	žen-U+length	růžI-U+length

Both paradigms start with having -A in the nominative, then both switch to -U in the accusative. They part ways in the middle for the genitive, locative, and dative, and then unify again in the instrumental. Note that even in the cases where the two paradigms differ, they have the same syncretism shape. So although each paradigm takes a different suffix in the "middle" cases, they both have a unique genitive, while the locative and dative are syncretic.[74]

Incorporating *žen-a* 'woman' and *růž-e* 'rose' in our system proves even more difficult because unlike with the previous pair (*žen-a* 'woman' and *koleg-a* 'colleague', section 6.1), these two do not seem to differ in their functional sequence. They are both feminines, both contain animate and inanimate nouns alike, and neither of the two paradigms shows any clues of any obvious semantic grouping. If that is correct, there is no feature over which they could split – so how come they do split after starting the same?

We will present two theoretical options for incorporating them into our system. The first option treats *růž-e* 'rose' and *žen-a* 'woman' as counterparts and works with the

[74] Another possible hint that the two paradigms might be counterparts comes from words like *šarí-a* 'sharia' which have variants. Except in the nominative, they show both the endings of the *žen-a* 'woman' and the *růž-e* 'rose' paradigms:

(549) 'SHARIA' PARADIGM, VARIANTS

	'sharia'	
	woman paradigm	rose paradigm
NOM	šarí-a	
ACC	šarí-u	šarí-i
GEN	šarí-i	šarí-e
LOC	šarí-e	šarí-i
DAT	šarí-e	šarí-i
INSTR	šarí-ou	šarí-í

concept of private lexical entries. These allow individual roots to use pointers towards lexical entries no one without the pointer can see. The second option turns away from treating the two paradigms as counterparts and makes use of the trees with a complex left branch as we saw it with *pán* 'sir' and *hrad* 'castle' in section 5.

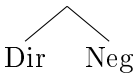
6.2.1 Private lexical entries account

A private lexical entry is a concept developed by Starke around 2012 that has been initially used in deriving the unproductive morphology of verbal suffixes like *-en*, e.g., in the past participle *brok-en*.

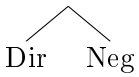
The idea is that unproductive morphology is stored in a separate part of the lexicon, inaccessible for standard syntactic derivation. For expository purposes, let us take an example from DeClercq and Vanden Wyngaerd's (2019) research of negative prefixes in French gradable adjectives. Among other, they look at the allomorphs *iN-*, which is productive (e.g., *injuste* 'unjust'), and *dé(s)*, which is unproductive and only combines with specific roots (e.g., *déloyal* 'disloyal').

De Clercq and Vanden Wyngaerd suggest the following lexical entries relevant to the *déloyal* 'disloyal' form where the adjectival root combines with an unproductive prefix *dé(s)* (we took the liberty to slightly adjust the format of the entries to fit with our own):

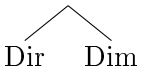
(550) $\langle_{765}$ DirP $\Leftrightarrow$ /*iN-*/ $\rangle$



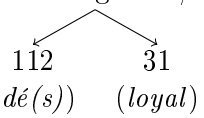
(551) $\langle_{112}$ DirP $\Leftrightarrow$ /*dé(s)*/ $\rangle$



(552) $\langle_{31}$ DirP $\Leftrightarrow$ /*loyal*/ $\rangle$



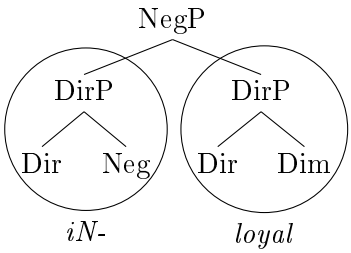
(553) $\langle_{599}$ NegP $\Leftrightarrow$ /–/ $\rangle$



Notice that both prefixes – the productive and the unproductive – spell out the same part of the functional sequence. Then there is an entry for the root and a pointer entry that combines the root with the unproductive prefix.

During the derivation, there will be a point when the NegP stage is reached, and the first spell out will be *iN-* + *loyal* because *iN* is the productive spellout of the left branch:

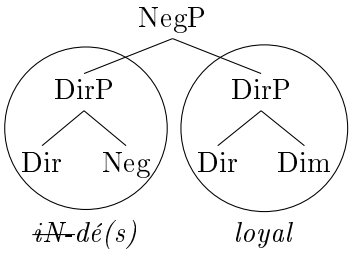
(554)



However, once syntax tries to find a match for the highest node, NegP, it will find the lexical entry <599>, which puts together the root and the unproductive prefix. This

entry does not have phonology of its own, but its left branch points to a private entry <112>, spelling out *dé(s)*. Since the spellout of the left branch of the derivation deviates from the spellout within the lexical entry, the left branch in the derivation swaps *iN-* for *dé(s)*:

(555)



The crucial part for us here is that the only way to the private entry for *dé(s)* is through a different lexical entry with a pointer. As such, the entry for *dé(s)* is never a competitor to *iN-* in a standard derivation, even though it has the same syntax.
We will use the same logic for the mysterious middle part of the conjoined twin pattern of the *žen-a* 'woman' and *růž-e* 'rose' paradigms.

For the root *žen* 'woman', we will reuse the second account from the previous section (see (520)), which was built on the idea of pointers combined with backtracking. Choosing this account over the one with pointer siege is merely practical – since it has fewer pointers, it simplifies the exposition. Below is the account repeated with the suffixes decomposed according to Element Theory:

(556) LEXICALIZATION TABLE OF 'WOMAN'

	XP	Gend	Mkd	M	Cl1	Cl2	Sg	K1	K2	K3	Anim	K4	K5	K6
NOM	žen			A										
ACC	žen			U										
GEN	žen			I										
LOC	žen			AI										
DAT	žen			AI										
INSTR	žen			β								← U+length		

The lexical entries we suggest for the paradigm *růž-e* 'rose' are below:

(557) LEXICALIZATION TABLE OF 'ROSE'

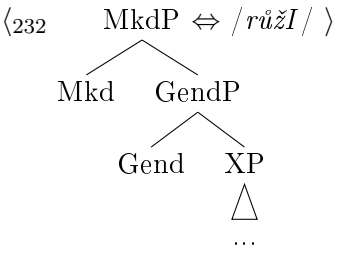
	XP	Gend	Mkd	M	Cl1	Cl2	Sg	K1	K2	K3	Anim	K4	K5	K6
NOM	růžI			A										
ACC	růžI			U										
GEN	růžI			↔ (AI)										
LOC	růžI			↔ (I)										
DAT	růžI			↔ (I)										
INSTR	růžI			β								← U+length		

As pointed out already, both paradigms share the nominative, accusative, and instrumental. Where they differ is the middle. The root *žen* 'woman' has standard lexical entries found

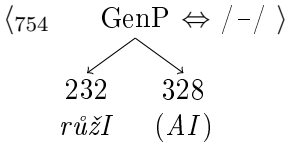
through the usual syntactic derivation. On the other hand, the root *růž* uses pointers towards private lexical entries (their private status marked by brackets).

Let us go through how the derivation proceeds. Below, see the lexicon we need to derive the nominative, accusative, and genitive of the root *růžI* 'rose'. There is a lexical entry of the root, an entry for the suffixes in the nominative and accusative, two different suffixes for the genitive (one standard and one private), and finally an entry that puts together the root *růž* with the private lexical entry for the genitive:

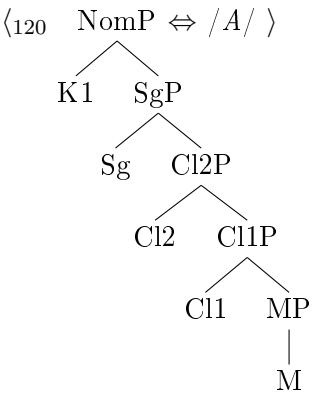
(558) LEXICAL ENTRY OF 'ROSE'



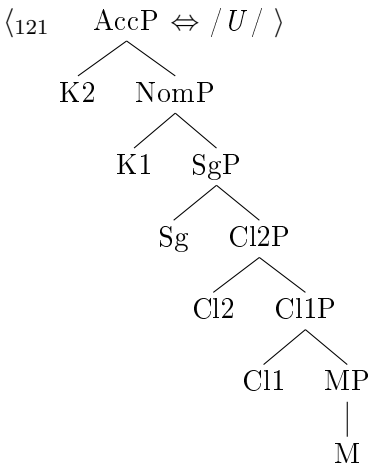
(559) LEXICAL ENTRY OF 'ROSE' POINTING TO THE PRIVATE ENTRY -AI



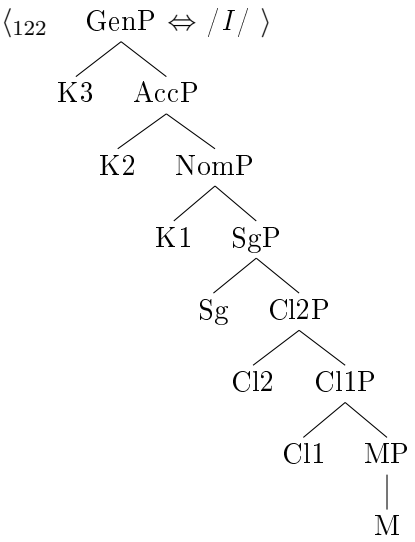
(560) LEXICAL ENTRY OF -A



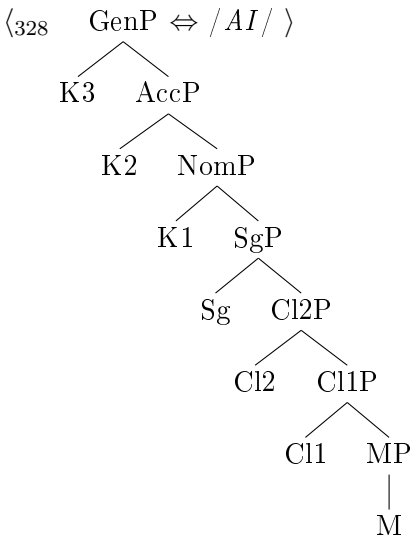
(561) LEXICAL ENTRY OF -U



(562) LEXICAL ENTRY OF -I

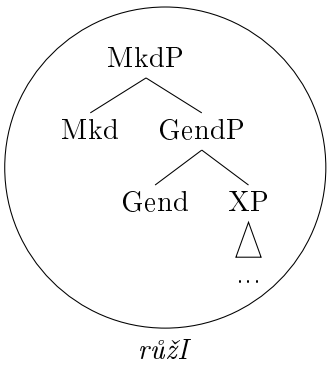


(563) PRIVATE LEXICAL ENTRY OF -AI



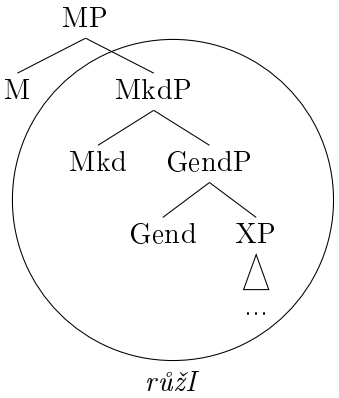
In the derivation, syntax first builds up the root by merging feature after feature and matching at each cycle. Ultimately, it arrives at the MkdP stage:

(564) DERIVATION OF 'ROSE', NOM, INTERMEDIATE STEP



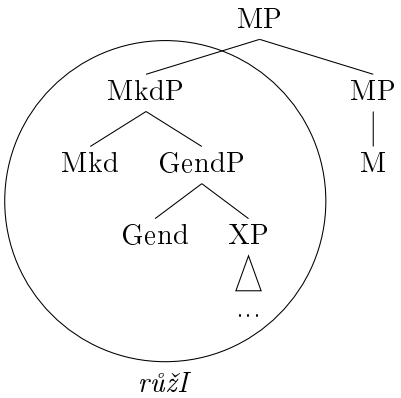
Next, the M feature is merged:

(565) DERIVATION OF 'ROSE', NOM, INTERMEDIATE STEP



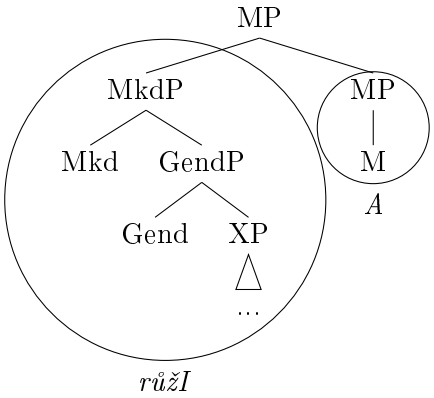
The root can no longer spell the whole structure out, so the complement movement gets triggered:

(566) DERIVATION OF 'ROSE', NOM, INTERMEDIATE STEP



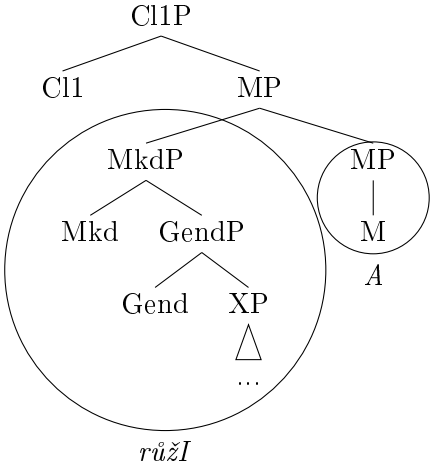
The root spells out the left branch. The right branch gets matched against the lexical entry for -A, which is the closest match:

(567) DERIVATION OF 'ROSE', NOM, INTERMEDIATE STEP



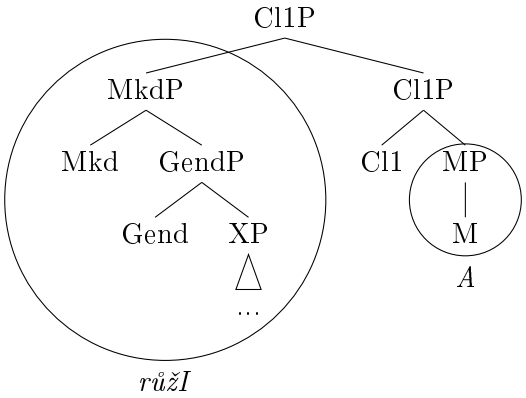
Next, the Cl1 feature gets merged in the structure:

(568) DERIVATION OF 'ROSE', NOM, INTERMEDIATE STEP



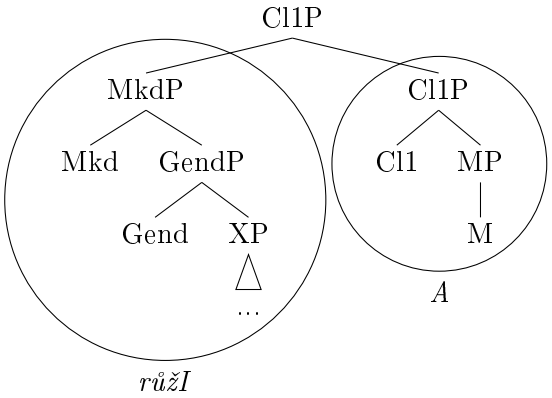
Since there is no lexical entry that could spell out the whole structure as it is, the Spec movement gets triggered:

(569) DERIVATION OF 'ROSE', NOM, INTERMEDIATE STEP



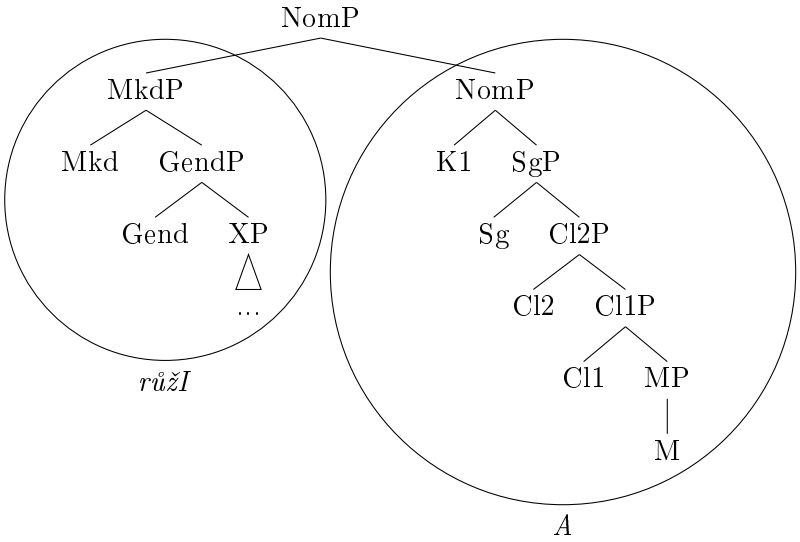
The right branch can now be again spelled out by -A:

(570) DERIVATION OF 'ROSE', NOM, INTERMEDIATE STEP



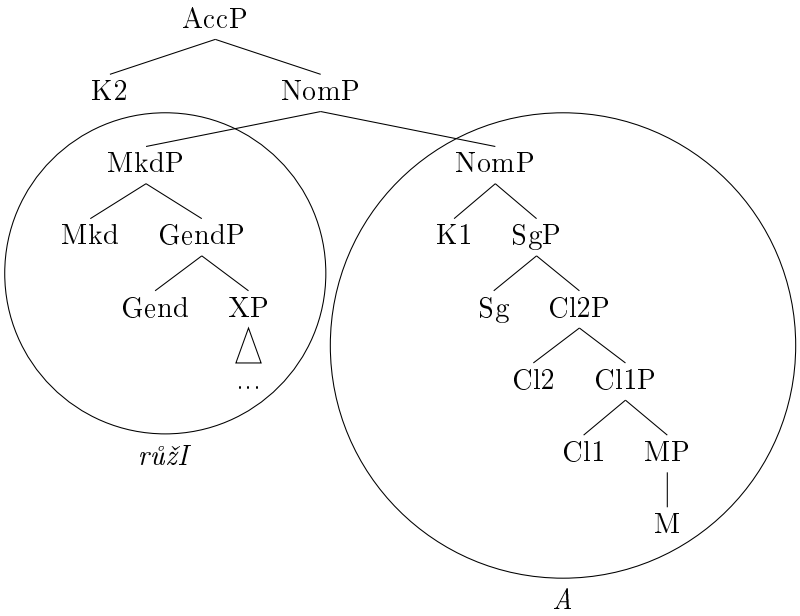
The same happens a few more times – merging, doing the Spec movement, and spelling out as -A – until we arrive at the complete derivation of the nominative:

(571) DERIVATION OF 'ROSE', NOM, FINAL FORM



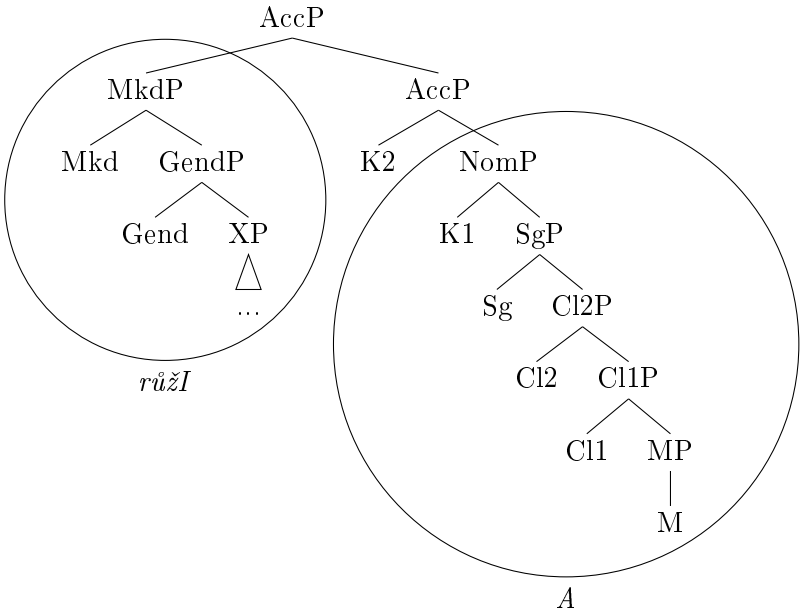
In the accusative, the K2 feature gets merged to the nominative structure:

(572) DERIVATION OF 'ROSE', ACC, INTERMEDIATE STEP



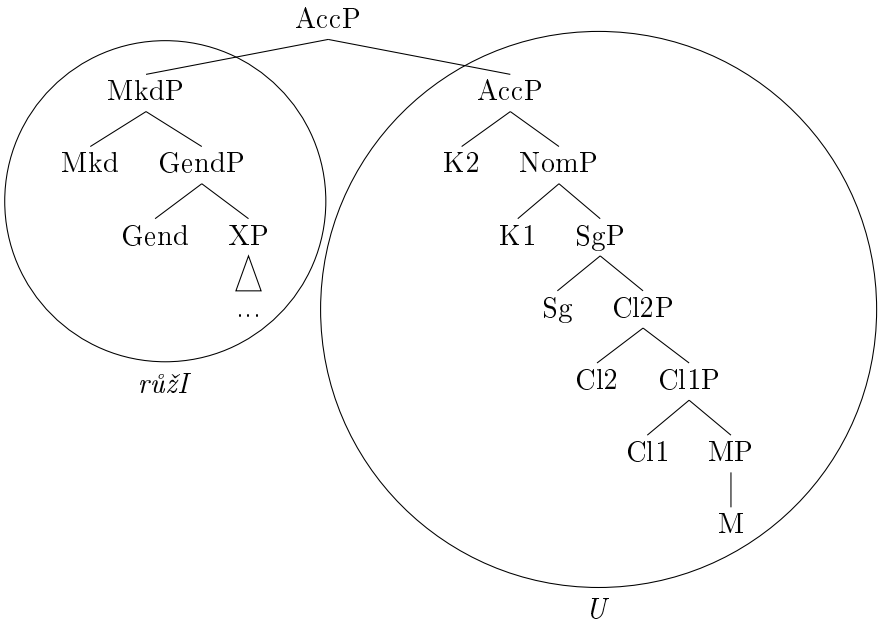
Since no entry can spell the whole structure out, the Spec movement gets triggered again:

(573) DERIVATION OF 'ROSE', ACC, INTERMEDIATE STEP



The entry for -A can no longer spell out the right branch. However, the branch is now a perfect match for the accusative -U:

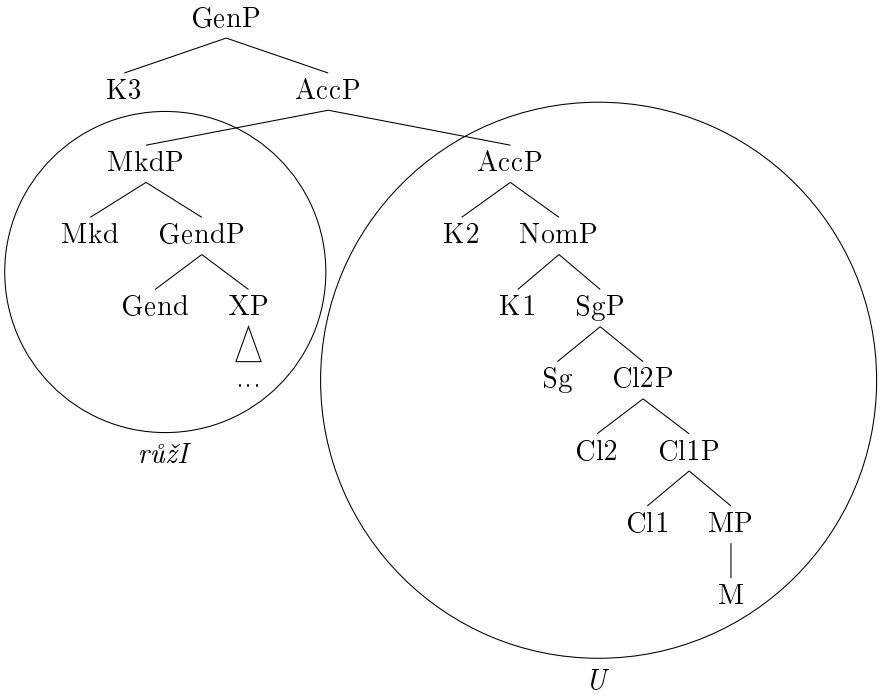
(574) DERIVATION OF 'ROSE', ACC, FINAL FORM



Finally, in the genitive derivation, we get to see the core of this account, the private lexical entry -AI.

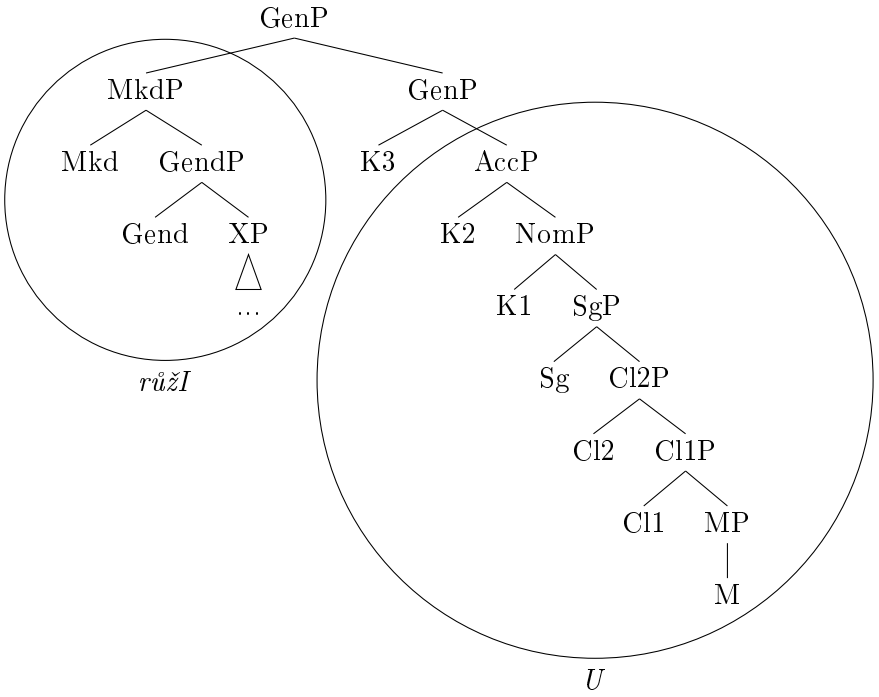
In the first step, we merge K3 to the accusative structure:

(575) DERIVATION OF 'ROSE', GEN, INTERMEDIATE STEP



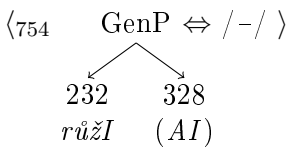
No lexical entry can spell out the whole tree, so the Spec movement gets triggered, bringing the root over K3:

(576) DERIVATION OF 'ROSE', GEN, INTERMEDIATE STEP



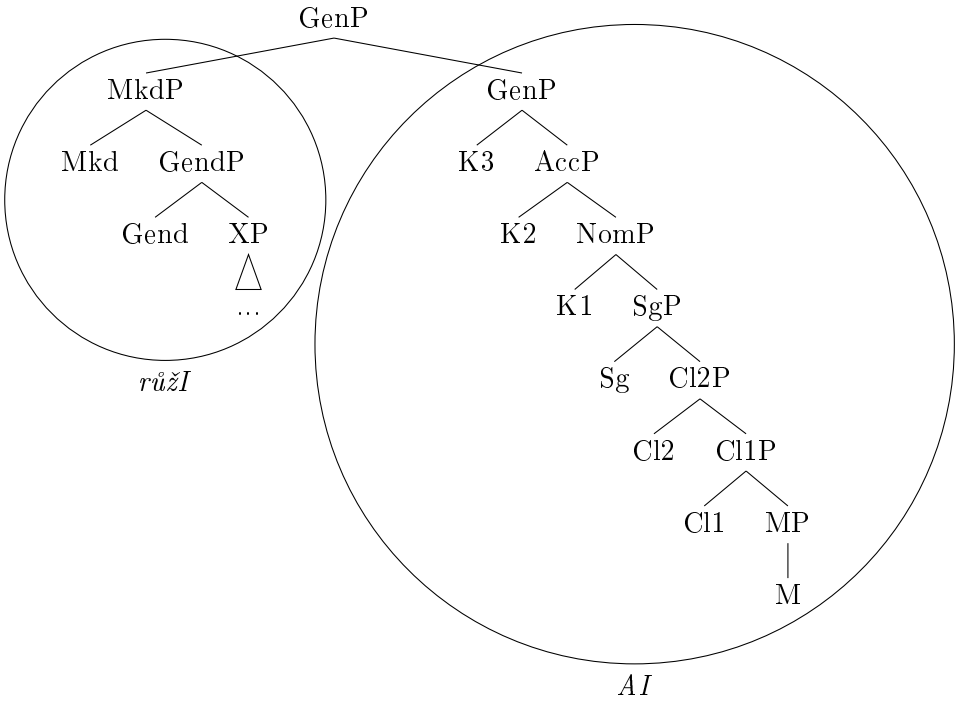
When syntax tries to match now, it finds that the topmost node is a perfect match for the entry <754> (repeated below):

(577) LEXICAL ENTRY OF 'ROSE' POINTING TO THE PRIVATE ENTRY -AI



This entry has no phonology, but its right branch points to the private entry <328> (*-AI*) and therefore makes it available for spellout. It requires the spellout of the right branch to be -AI, so -AI overrides the previous winner. That is how we get the genitive form with the private lexical entry:

(578) DERIVATION OF 'ROSE', GEN, FINAL FORM



The private lexical entry for the locative and dative would work the same – merging the following features, always bringing the root up through the Spec movement, then matching the highest node against the lexical entry with a pointer towards a private lexical entry on the right branch, and rewriting it with the spellout of the private lexical entry.

This is how we can get the middle of the conjoined twin pattern different, using private lexical entries.

There are three potential issues with this account. The first one has to do with productivity. It is a fact that there are fewer roots in the *růž-e* 'rose' paradigm than in the *žen-a* 'woman' paradigm. A quick corpus search (Křen et al. 2020) shows that there are over 11 thousand (feminine) lemmas with the pattern NOM:a – ACC:u (like *žen-a*), but only under 4 thousand (feminine) lemmas with the pattern NOM:e – ACC:i (like *růž-e*). Hence, if any of these should have unproductive private lexical entries, we would expect it from *růž-e*. At the same time, having under 4 thousand lemmas is not little. Moreover, we are not aware of any source that would suggest that the paradigm is no longer productive – although the opposite is also true. More data will be needed to determine whether the paradigm *růž-e* 'rose' is still productive and, therefore, qualifies as a candidate for the private lexical entries account.

The second drawback is a potential constituency/selection issue. The account is based on the idea that it is the individual roots, like *růž* 'rose', specifying that they take the private lexical entries. However, some morphologically complex nouns in Czech take the same case suffixes as *růž-e*. See, e.g., *lv-ic-e* 'lioness' below:

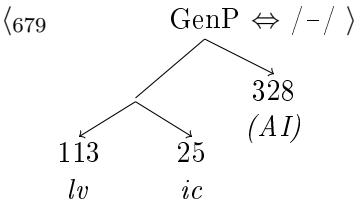
(579) SURFACE FORMS OF 'LIONESS' AND 'ROSE'

	'lioness'	'rose'
NOM	lv-ic-e	růž-e
ACC	lv-ic-i	růž-i
GEN	lv-ic-e	růž-e
LOC	lv-ic-i	růž-i
DAT	lv-ic-i	růž-i
INSTR	lv-ic-í	růž-í

The morpheme *-ic-* is (among other) a frequent derivational morpheme for female animal names (*lev* 'lion' – *lv-ic-e* 'lioness', *slon* '(bull) elephant' – *slon-ic-e* '(cow) elephant' etc.; cf. Findejsová 2017). Crucially, it always seems to select the suffixes of the *růž-e* 'rose' paradigm.

Since the root and the *-ic-* morpheme are closer in the structure than *-ic-* and the case suffix, we cannot easily say that it is *-ic-* selecting the case suffix. We would have to say that it is the whole complex left branch, spelled out as *lv-ic*, that has a pointer towards the suffix:

(580) POTENTIAL LEXICAL ENTRY OF 'LIONESS'



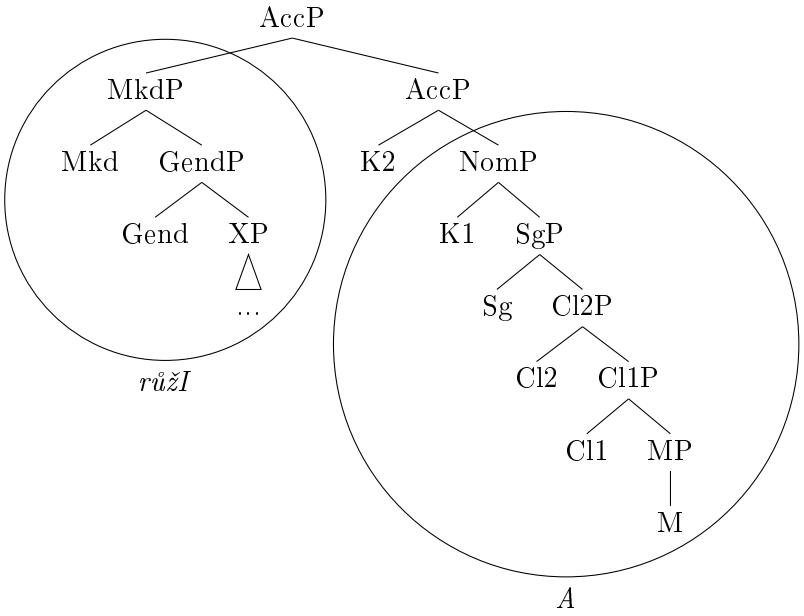
While technically possible, this solution fails to capture the generalization that all the nouns with *-ic-* go for the *růž*-type suffixes. It should be this morpheme and not the whole stem selecting them.

The third issue of this account has to do with the matching procedure. If we were to follow the standard Nanosyntactic matching strictly, we would get the form *růž-e* (⇐ růžI-AI) and not *růž-i* (⇐ růžI-U) in the accusative.[75] The reason for that is that the matching procedure always targets the highest possible node.

When deriving the accusative, we get to the stage where the root moves across the K2 feature, and the structure searches for a matching candidate for the new structure (repeated from (573)):

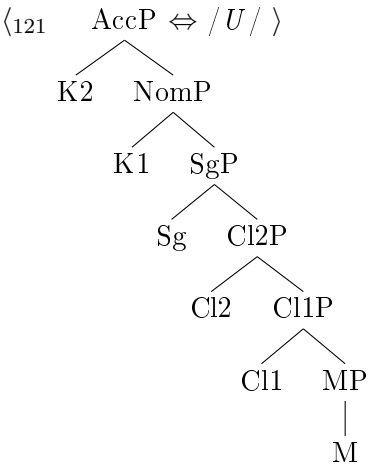
[75] The same form would be derived in the nominative. There it is less clear it would be an issue since the correct surface form is the same, *růž-e*. The paradigm would no longer be a counterpart to *žen-a*, though, since the nominative-accusative suffixes would be AI-AI and not A-U, and hence it would not be a way to capture the conjoined twin pattern.

(581) DERIVATION OF 'ROSE', ACC, INTERMEDIATE STEP



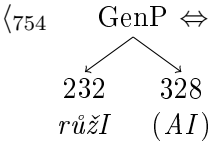
The solution we presented counted with finding the accusative -U:

(582) LEXICAL ENTRY OF -U



However, the private lexical entry for the genitive is, strictly speaking, a better match here:

(583) LEXICAL ENTRY OF 'ROSE' POINTING TO THE PRIVATE ENTRY -AI



It is a better match because it can spell out the root node in (581) through the Superset Principle since the root + the accusative structure are a subset of it.

The reason for having the highest node to win over the daughter nodes is that languages seem to favor synthetic spellout over the analytic one. However, since the entry <754> has zero phonology, using this entry and the entry in <121> results in the same amount of morphemes. So perhaps, there is a clue in that. With how the spellout algorithm works at the moment, though, we are mispredicting the accusative (and possibly the nominative) form.

These issues aside, the private lexical entry account offers one way to deal with the conjoined twin pattern, where two paradigms start and end the same but differ in the middle.

6.2.2 Trees with a complex left branch account

The second type of an account we explore makes use of the trees with a complex left branch, a tool already presented in the section 5, where we zoomed in on the masculine paradigms *pán* 'sir' and *hrad* 'castle'.

Here the *žen-a* 'woman' and *růž-e* 'rose' pair is approached differently as it does not assume the two being counterparts. Below, we can see the complete suggestion for the lexical entries under this approach. The only suffixal entry the two paradigms share is the instrumental -U+length. Besides that, *žen-a* has a unique set of suffixes from the nominative to the dative. It never interacts with suffixes from other paradigms, including *růž-e* 'rose', because the root is too small:

(584) LEXICALIZATION TABLE OF 'WOMAN'

	XP	Gend	Mkd	M	Cl1	Cl2	Sg	K1	K2	K3	Anim	K4	K5	K6
NOM	žen			A										
ACC	žen			U										
GEN	žen			I										
LOC	žen			AI										
DAT	žen			AI										
INSTR	žen		β									← U+length		

On the other hand, *růž-e* 'rose' shares the nominative suffix with the M-sized *sak/moř* 'jacket/sea' paradigm. The genitive and dative here are also unified with the M-sized paradigm (among others):

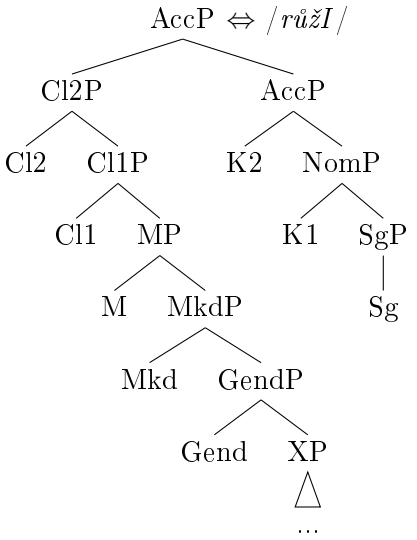
(585) LEXICALIZATION TABLE OF 'ROSE'

	XP	Gend	Mkd	M	Cl1	Cl2	Sg	K1	K2	K3	Anim	K4	K5	K6
NOM	růžI						AU							
ACC	růžI													
GEN	růžI					A								
LOC	růžI				U									
DAT	růžI				U									
INSTR	růžI		β									← U+length		

The unique part of this account shows in the accusative. The entire lexical entry is used for the spellout, unlike in the other cases where we only use the left branch. We suggest that the *-i* we see in the accusative there is the root-final floating I.

See the tree-form of the *růž-e* 'rose' entry below:

(586) DLB LEXICAL ENTRY OF 'ROSE'



A big advantage of this account is that it captures the shared suffixes of *růž-e* with other paradigms. See, e.g., the surface forms comparison with *moř-e* 'sea':

(587) SURFACE FORMS OF 'ROSE' AND 'SEA'

	'rose' (F)	'sea' (N)
NOM	růž-e	moř-e
ACC	růž-i	moř-e
GEN	růž-e	moř-e
LOC	růž-i	moř-i
DAT	růž-i	moř-i
INSTR	růž-í	moř-em

Unifying *růž-e* 'rose' and *moř-e* 'sea' suffixes does not seem possible when pursuing the idea that *žen-a* 'woman' and *růž-e* 'rose' are counterparts, as we did in the private lexical entry account.

We will show the derivation in a moment, but first, let us say that just like the private lexical entries account, this one also has its drawbacks. First, using -AU as the lexical entry for the nominative of *růž-e* 'rose' raises some questions. It seems reasonable to suggest the -AU elements for *moř-e*, since it has emerged from the so-called o-stems (Večerka 2006) and, in fact, originally had an *-o* suffix. So the idea is simple – it still has the *-o* suffix underlyingly, but it gets hidden through combining it with the root-final floating I. The paradigm *růž-e* did not develop from the o-stems and historically did not

have a suffix *-o* (and had *-a* instead). So connecting it with the -AU ending is not as obvious as for *moř-e* 'sea'.

The second issue with this account is that the accusative relies on the root-final floating I to be the suffix *-i* we see on the surface:

(588) 'ROSE' IN CV THEORY, FLOATING I

C	V	C	V
|	|	|	
r	ů	ž	I

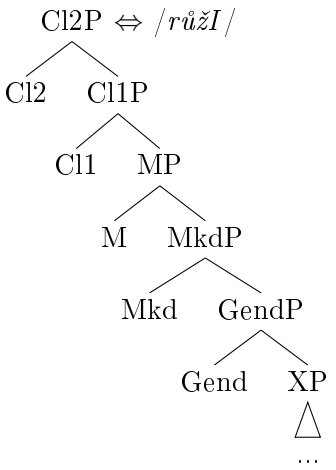
(589) 'ROSE' IN CV THEORY, FLOATING I ATTACHES

C	V	C	V
|	|	|	↑
r	ů	ž	I

This is not well compatible with the proposal in the section 4.3, where we operate with the assumption that the domain-final empty nucleus is governed by default. This would mean there is no reason for the floating I to link in the accusative of *růžI*.

There are at least two ways to think about it. One is that perhaps the I in the root of *růžI* is not floating (unlike in the other palatalized roots) and is always linked. We would still have to explain, what motivates the suffixes in other cases to connect to the I. A second possibility is that the accusative has a suppletive root:

(590) LEXICAL ENTRY OF 'ROSE', ALTERNATIVE PROPOSAL OF A SUPPLETIVE ROOT

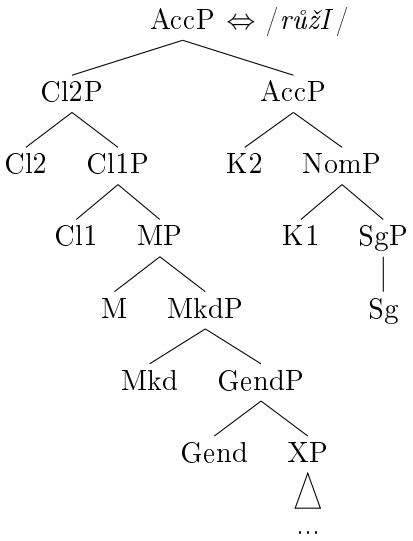


The accusative would still be spelled out by the complex lexical entry in (586), except that the I there would not be floating and would always be linked, giving us the correct surface form *růž-i*. The other cases would be spelled out by the suppletive root in (590), where the I would be floating and therefore would combine with the case suffixes in the same way we described in the section 4.3.

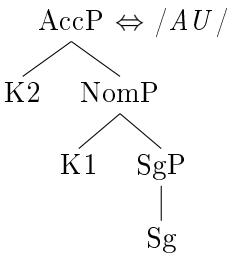
Putting these issues aside, let us go step by step through the derivation of *růž-e* 'rose' to depict how the DLB trees work here.

We will need the following lexical entries:

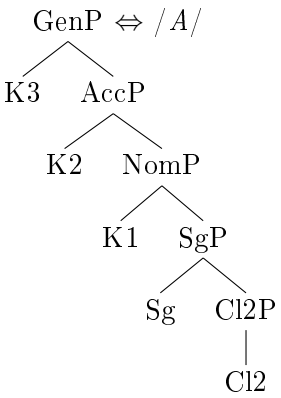
(591) DLB LEXICAL ENTRY OF 'ROSE'



(592) LEXICAL ENTRY OF -AU

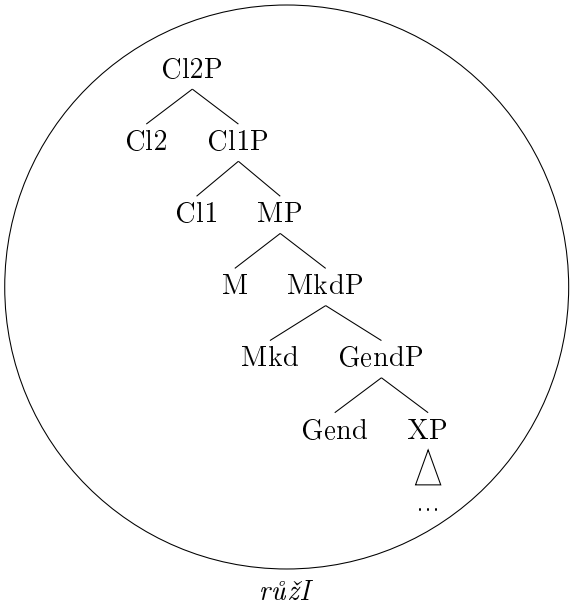


(593) LEXICAL ENTRY OF -A



Now, onto the derivation. We skip the first few steps that merely merge feature by feature and spell out the whole structure by the root. This brings us to the Cl2 stage:

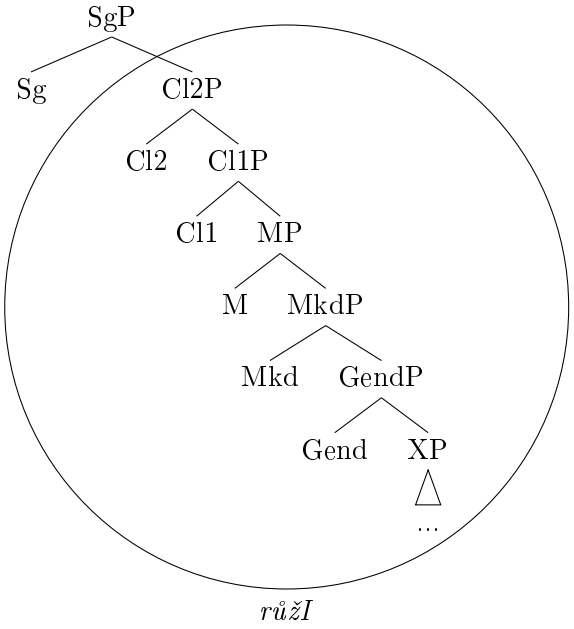
(594) DERIVATION OF 'ROSE', NOM, INTERMEDIATE STEP



Note that this spellout is possible due to the Superset Principle that can spell out any subset of a lexical entry – in this case, the left branch of the *růžI* lexical entry.

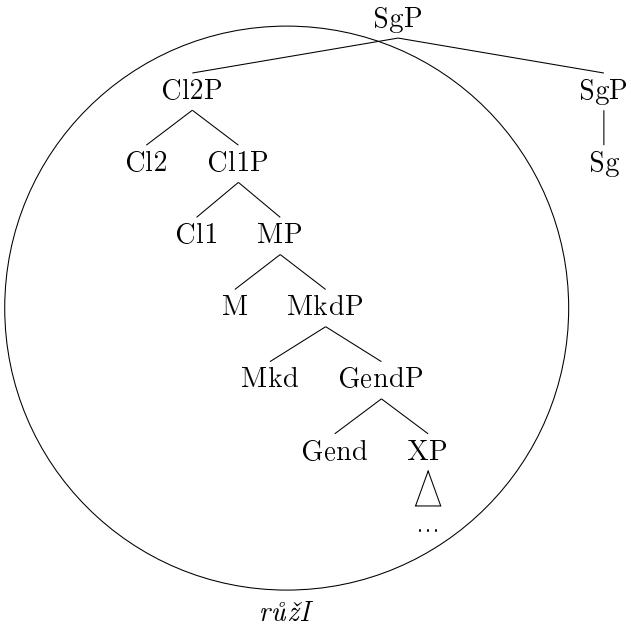
Next, we merge Sg and try to spell it out:

(595) DERIVATION OF 'ROSE', NOM, INTERMEDIATE STEP



The root can no longer spell the structure out as it is, which triggers the complement movement, bringing the Cl2P node above SgP:

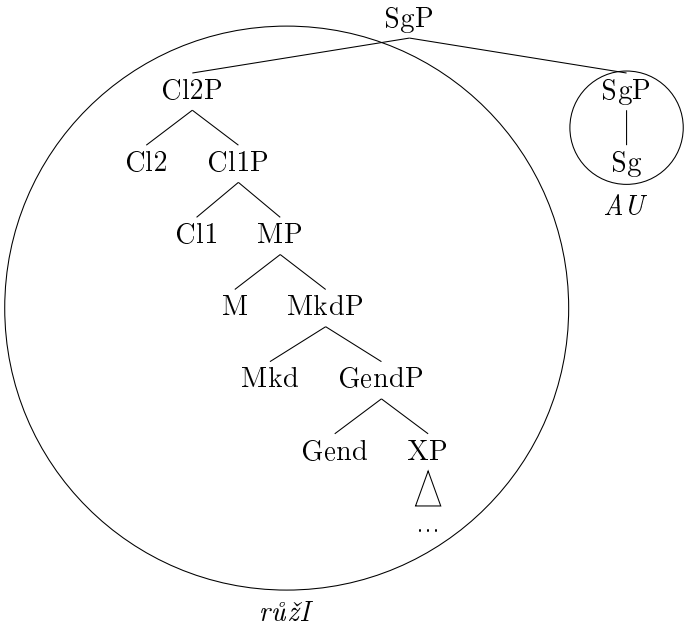
(596) DERIVATION OF 'ROSE', NOM, INTERMEDIATE STEP



Our lexicon has no entry that can spell out the highest node. Particularly notice it for the entry of *růžI* – it cannot spell it out because it misses the nominative and accusative layers in the middle of the right branch.

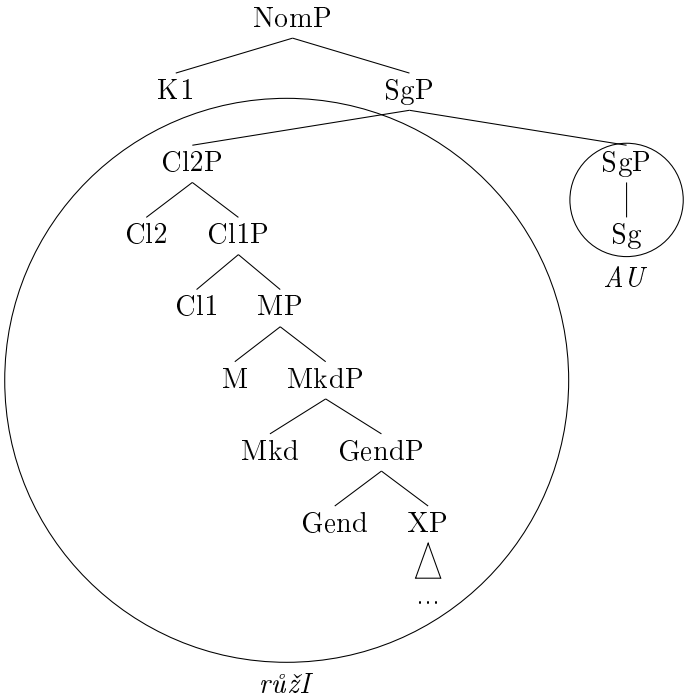
That being said, the left branch is already spelled out by the root, so the only thing we need to make the derivation proceed is to find a match for the right branch (for the lower SgP node). The only match for SgP in our lexicon is the entry -AU:

(597) DERIVATION OF 'ROSE', NOM, INTERMEDIATE STEP



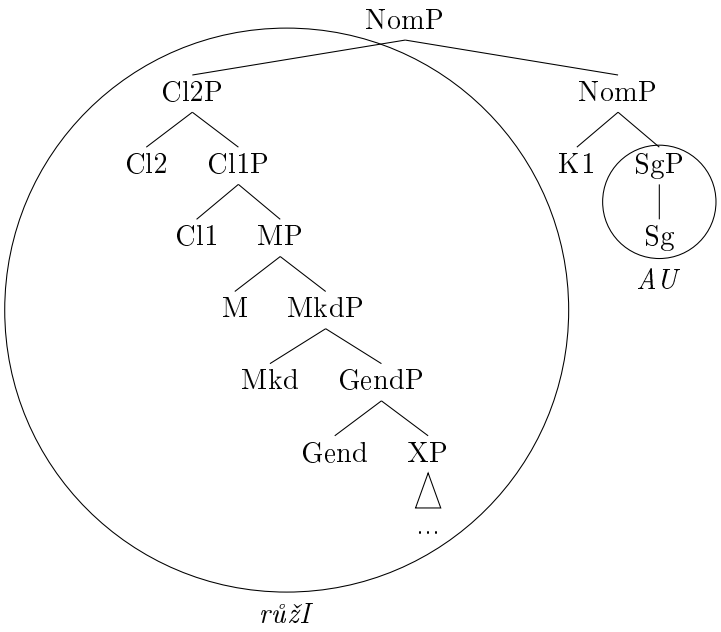
Everything is matched, so we merge K1, the last feature needed to derive the nominative:

(598) DERIVATION OF 'ROSE', NOM, INTERMEDIATE STEP



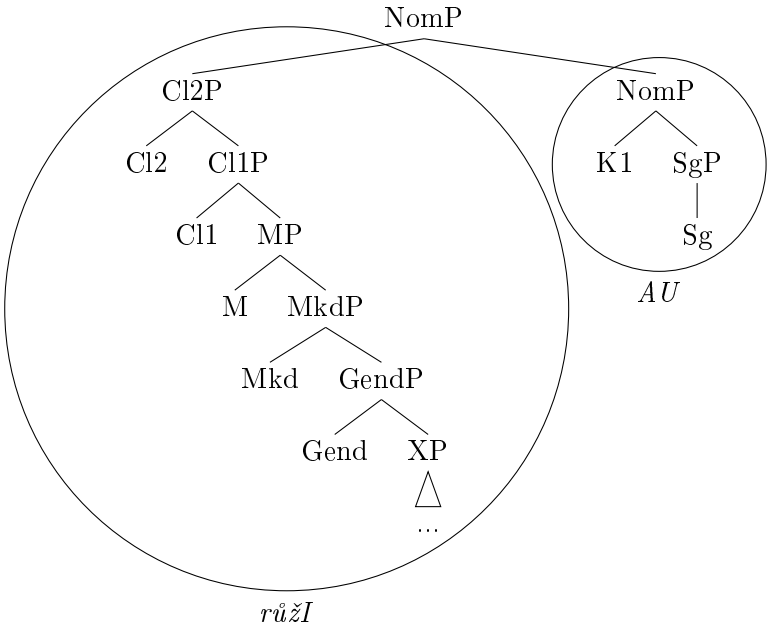
Again, no lexical entry can spell the highest node with everything below as it is, so the Spec movement gets triggered:

(599) DERIVATION OF 'ROSE', NOM, INTERMEDIATE STEP



The lexical entry for -AU can spell out the whole right branch:

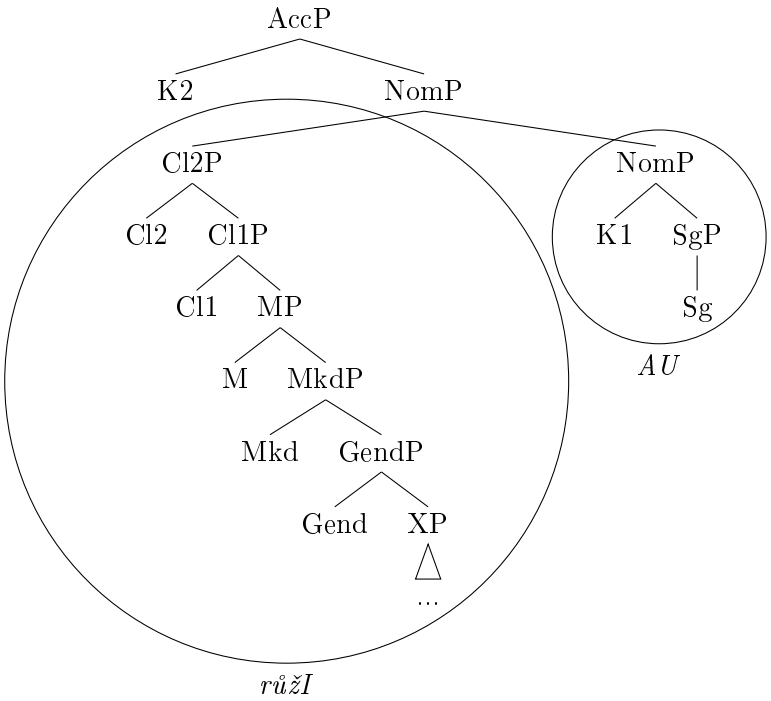
(600) DERIVATION OF 'ROSE', NOM, FINAL FORM



That is how we get the nominative. Then in the phonology module, the elements attempt to combine, but since I and U cannot combine, only I and A merge, and we get *-e* on the surface.

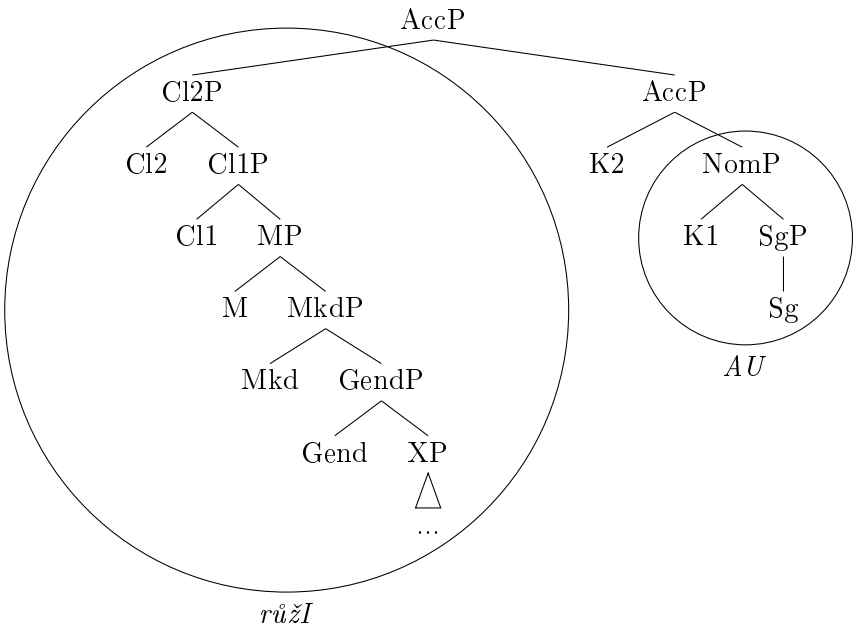
Next is the accusative. For that, we need to merge K2 on top of the nominative structure:

(601) DERIVATION OF 'ROSE', ACC, INTERMEDIATE STEP



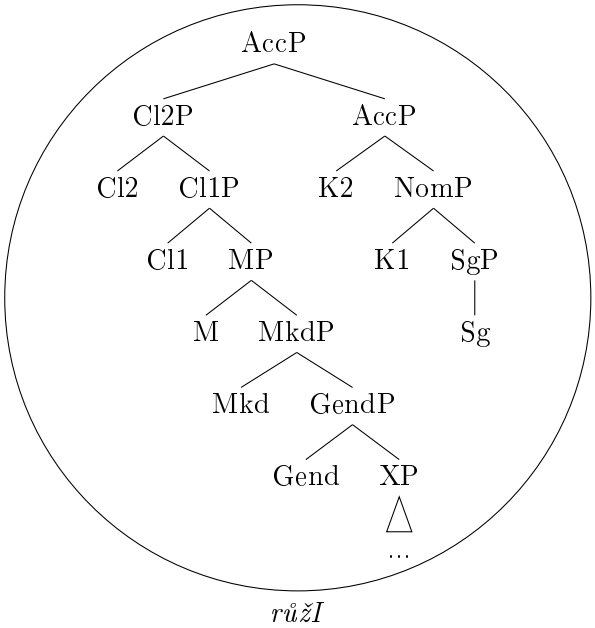
No lexical entry can spell out the structure in its entirety, so the Spec movement gets triggered:

(602) DERIVATION OF 'ROSE', ACC, INTERMEDIATE STEP

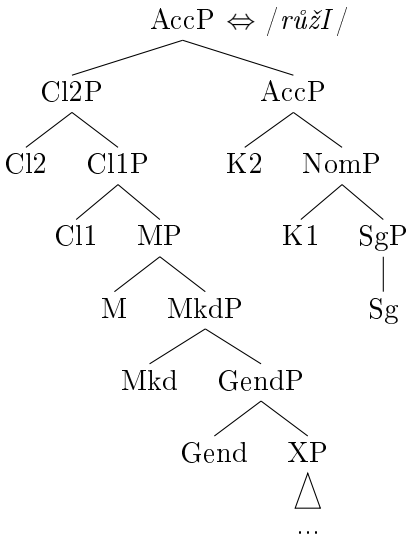


Since the matching proceeds top-down in our model, the syntax first tries to match the current highest node. Unlike in the nominative, it finds it. This time, the full lexical entry for the root *růžI* 'rose' can spell the structure out. Compare the syntax in (603) with the lexical entry repeated in (604):

(603) DERIVATION OF 'ROSE', ACC, FINAL FORM



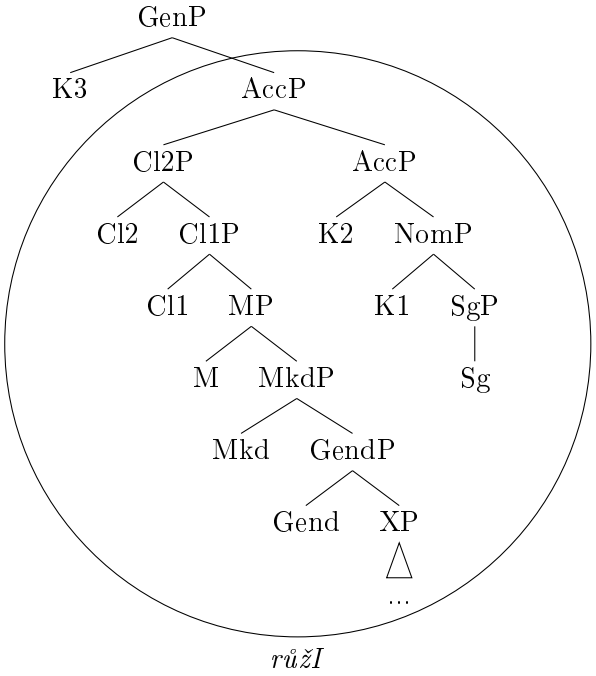
(604) DLB LEXICAL ENTRY OF 'ROSE'



The entry for *růžI* 'rose' is now a perfect match for the accusative structure, and therefore everything can and will be spelled out just by the root without a need for any suffix.

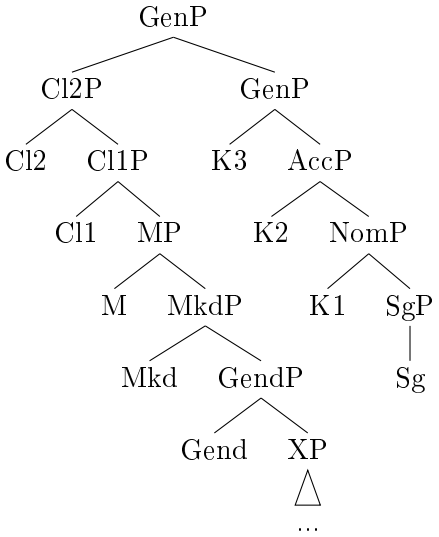
Let us look at the derivation of the genitive and see how the root stops being able to spell out the whole structure and therefore returns to the suffixal spellout.
After deriving the accusative, we merge one more feature, K3:

(605) DERIVATION OF 'ROSE', GEN, INTERMEDIATE STEP



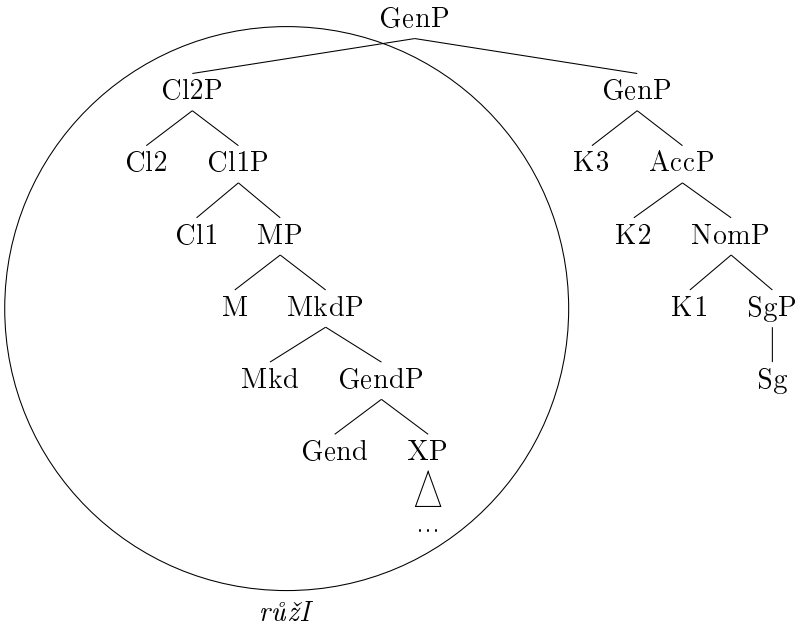
No lexical entry can spell out the whole structure as it is, so the Spec movement gets triggered:

(606) DERIVATION OF 'ROSE', GEN, INTERMEDIATE STEP



The syntax tries to find a match for the highest node but finds none. The root *růžI* can no longer spell the structure out because its entry does not contain the genitive phrase. However, the root can still spell out the left branch as that is a proper subset of it:

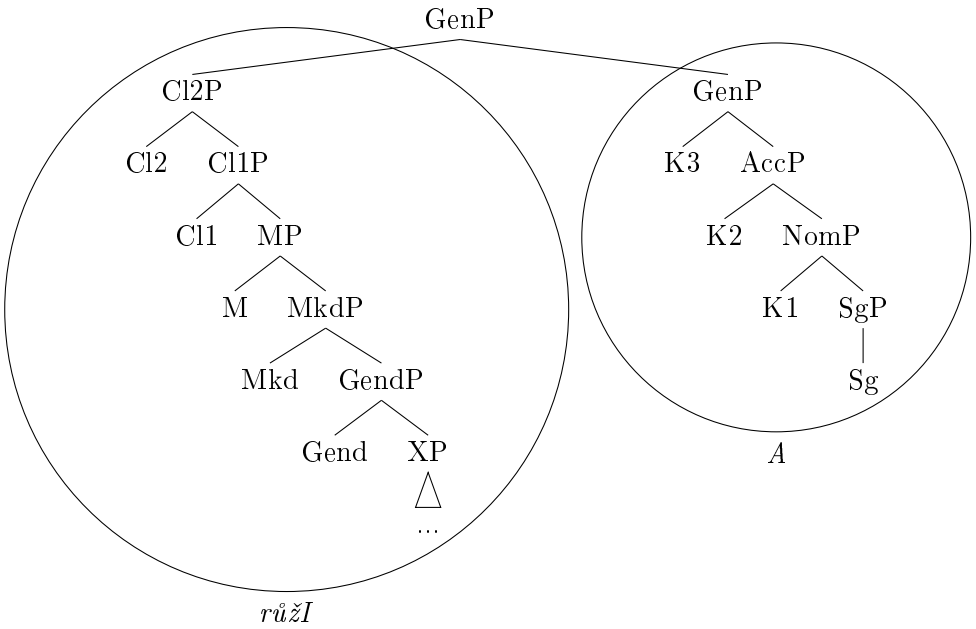
(607) DERIVATION OF 'ROSE', GEN, INTERMEDIATE STEP



So only the right branch is left for the spellout, and that one perfectly matches the lexical

entry for -A:

(608) DERIVATION OF 'ROSE', GEN, FINAL FORM



That is how we derive the genitive of *růž-e* 'rose' in the DLB trees model. The derivation of the rest of the cases would then continue the same as for the M-sized roots.

To sum this up, we have offered two possible accounts for incorporating the *žen-a* 'woman' / *růž-e* 'rose' pair. One considers the two paradigms counterparts and treats the conjoined twin pattern using private lexical entries. These entries are not accessible during a computation unless a specific lexical entry with pointers points to them.

The second account keeps away from the idea of counterparts and instead highlights the similarity of the *růž-e* paradigm with other paradigms we have seen. It uses complex trees for lexical entries and works because we can use just the left branch of an entry using the Superset Principle.

Both accounts have some drawbacks, so these two paradigms stay as dots to be fully connected later.

6.3 The 'album' paradigm

There is one last paradigm we would like to discuss, and that is the neuter *album*. This paradigm is unremarkable in its genitive to instrumental suffixes as they are identical to the suffixes of the XL-sized paradigm *jazyk* 'tongue'. However, it has one potentially intriguing attribute, and that is the status of the *-um* ending in the nominative and accusative. See its forms below, compared to the *jazyk* 'tongue' paradigm:

(609) SURFACE FORMS OF 'ALBUM' AND 'TONGUE'

	'album' (N)	'tongue' (Mi)
NOM	album	jazyk
ACC	album	jazyk
GEN	alb-a	jazyk-a
LOC	alb-u	jazyk-u
DAT	alb-u	jazyk-u
INSTR	alb-em	jazyk-em

The analysis can unfold in different directions depending on whether we treat the *-um* as a self-standing suffix or as a part of the root:

(610) album
alb-um

It is not clear which alternative we should prefer. On the one hand, Czech does not have a productive *-um* suffix, which would speak in favor of the undecomposed root *album*. With this option, we would, however, have to explain where the *-um* disappears in the other cases.

At the same time, the roots with the *album* pattern are not particularly scarce. A quick corpus search (Křen et al. 2020) suggests there might be up to one thousand lemmas with it. So having to propose a special entry for each of the lemmas separately – as would be required have we decomposed the form and used an allomorphy account – does not sound economical.[76]

The good news is that whichever version we choose in the end, both can have a straightforward Nanosyntactic analysis using the tool introduced in this section, namely pointers. For the time being, we present both options and remain fairly agnostic on which one should be preferred. However, at the end of this section, we offer a few reasons for possibly adopting the root-allomorphy account in the future.

6.3.1 Suffix -um as a private lexical entry

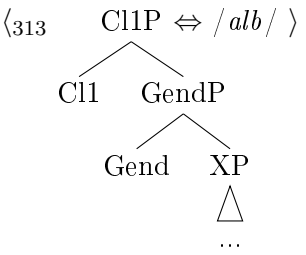
First, let us propose an analysis for the decomposed version, *alb-um*, which treats *-um* as a separate morpheme. As has been already mentioned, this is not self-evident because there is no productive suffix *-um* in Czech, and proposing a standard lexical entry for it would lead to the possibility of other roots interacting with it in unwanted ways.

However, in section 6.2.1, we have introduced a tool that allows us to capture morpheme decomposition for unproductive items – the private lexical entries. The same can be used here to treat *-um*.

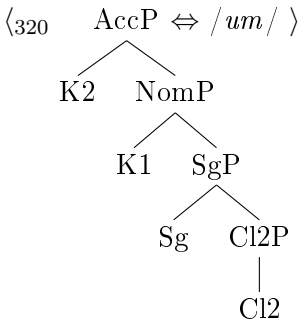
For this alternative line of analysis, we would need the following lexical entries:

[76] Furthermore, roots with *-um* are not the only ones with such a pattern. There are others, e.g., *cyklus*, that show *-us* in the nominative and accusative and then lose it in the structurally higher cases. Everything we will propose for the *-um* roots can be applied to the *-us* roots as well.

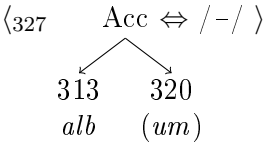
(611) LEXICAL ENTRY OF 'ALBUM'



(612) PRIVATE LEXICAL ENTRY OF -UM



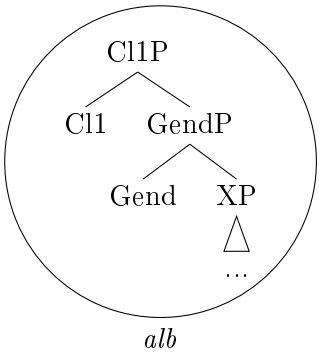
(613) LEXICAL ENTRY OF 'ALBUM' POINTING TO THE PRIVATE ENTRY -UM



The entry in (611) shows the size of the root (S-size, on a par with *chleb-a* 'bread'). The entry in (612) shows the private lexical entry *-um*. Since it is private, it will never be a candidate in a normal derivation and will only ever be accessed if there is another lexical entry pointing to it. That is the job of the complex entry with pointers in (613), where the private status of *-um* is marked by round brackets around it.

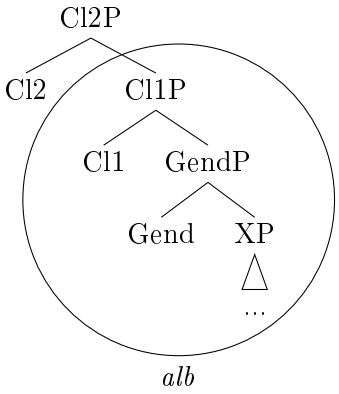
The derivation proceeds standardly. First, we merge all the lowest features, spelling the structure out by the root at every cycle, until we reach the Cl1 stage:

(614) DERIVATION OF 'ALBUM', NOM, INTERMEDIATE STEP



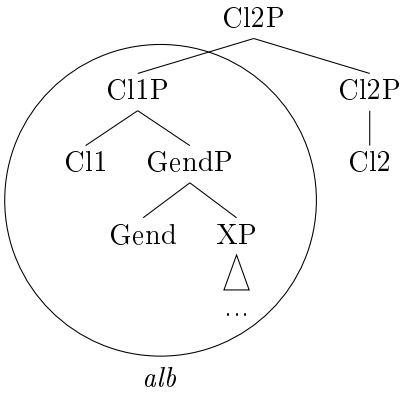
Then Cl2 is merged:

(615) DERIVATION OF 'ALBUM', NOM, INTERMEDIATE STEP



The root can no longer spell the entire structure out, so the complement movement gets triggered:

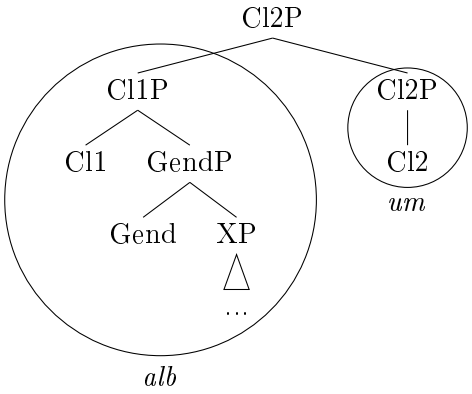
(616) DERIVATION OF 'ALBUM', NOM, INTERMEDIATE STEP



When we try to spell out now, there are two candidates. One is the morpheme -A (see (663)), which can spell out the right branch on its own, and the other is the pointer lexical entry <327>. The pointer entry wins because it can spell out the root node, which is always preferred. It matches the syntactic structure above because the structure is its subset (and the pointer allows us to shed top layers on the right branch).

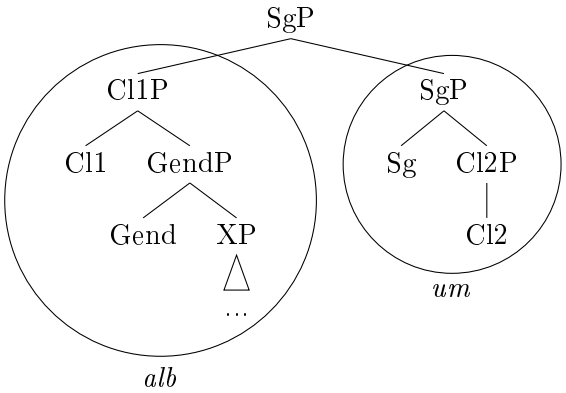
The winning entry does not have phonology assigned to it. Instead, it says that the daughter nodes need to be spelled out as *alb* and *-um*. The left branch is already spelled out correctly, but the right branch needs to be spelled out as *-um*:

(617) DERIVATION OF 'ALBUM', NOM, INTERMEDIATE STEP



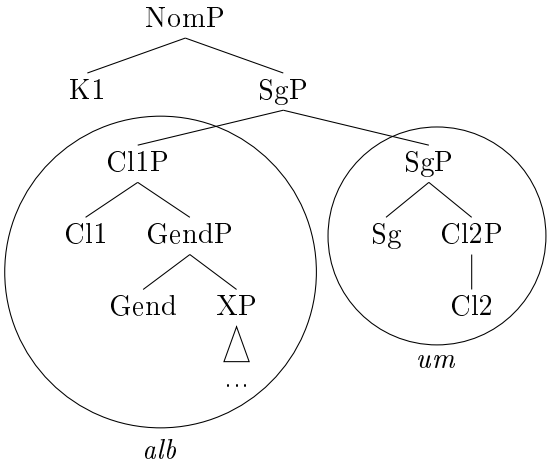
From then on, the derivation continues in the same fashion. We merge Sg, do the Spec movement to bring the root-spelled left branch on top, and again spell out using the pointer lexical entry:

(618) DERIVATION OF 'ALBUM', NOM, INTERMEDIATE STEP



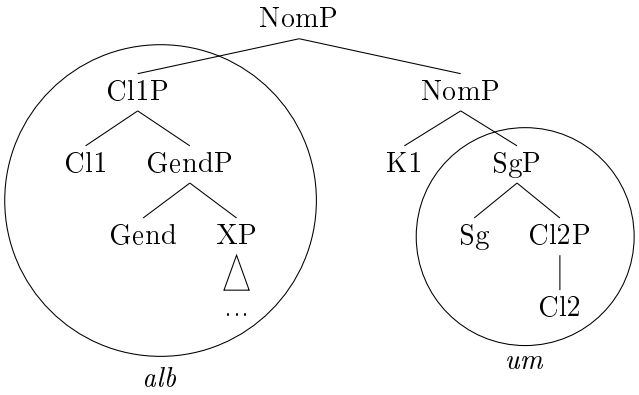
The last feature merged for the nominative is K1:

(619) DERIVATION OF 'ALBUM', NOM, INTERMEDIATE STEP



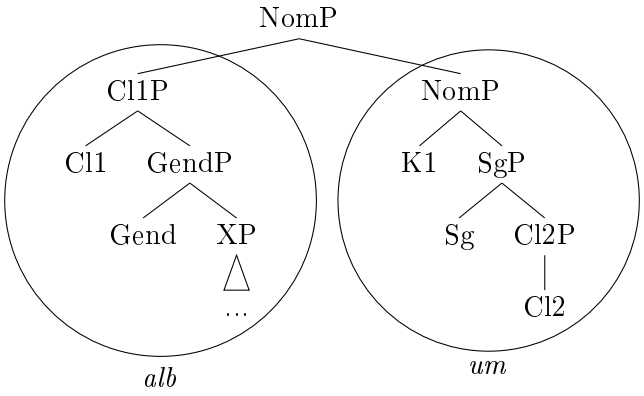
The left branch moves across K1 through the Spec movement:

(620) DERIVATION OF 'ALBUM', NOM, INTERMEDIATE STEP



The structure gets again spelled out by the pointer entry, creating the correct nominative surface form *alb-um*:

(621) DERIVATION OF 'ALBUM', NOM, FINAL FORM



The accusative is then spelled out by the same lexical entries. The standard, non-private suffix -A wins the competition in the genitive because the pointer entry can no longer spell the structure out. From there, the derivation continues the same as for the XL-sized *jazyk* 'tongue'.

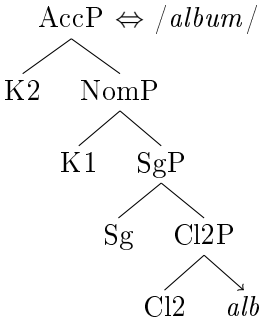
That is how we can derive *alb-um* where the root and the suffix are separate morphemes but still keeping to the idea that *-um* is not a productive morpheme.[77]

6.3.2 Root-allomorphy done through simple pointers

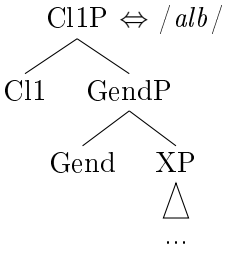
The second decomposition option, *album*, looks away from the private lexical entries and instead treats *-um* as a part of the root. This path would mean there is a root allomorphy within the paradigm. That is an interesting take because we have only seen single-root paradigms so far. However, phrasal spellout naturally gives us the option of having two (or more) root forms.

Consider the lexical entries below:[78]

(622) ALBUM, ALLOMORPHY ENTRY



(623) ALB, ALLOMORPHY ENTRY



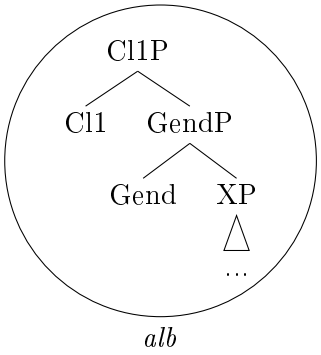
There are two things to note in these entries. First, a pointer is at the bottom of (622). Second, the pointer points to another lexical entry, the one in (623). The fact that the entry for *album* points to the entry for *alb* connects them and assures that *album* is an allomorph to *alb* and not to any other root.

Let us quickly sketch a derivation where we will see how this account works. We start by first merging all the features up to Cl1. This is spelled out as *alb* because the lexical entry in (623) is a perfect match for the structure:

[77] Note that there is also the theoretical option of decomposing *alb-um* to the root and the suffix and treating *-um* as a regular (non-private) entry. It would require us to design the lexical entry for *-um* carefully so that it never interacts with other roots. While theoretically possible, the private lexical entry account seems more likely given the unproductivity of *-um*.

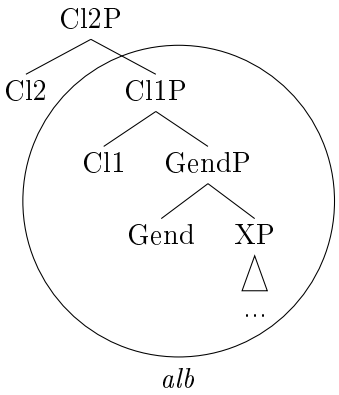
[78] Thanks to Guido Vanden Wyngaerd (one of the official readers for this dissertation) for suggesting the solution below. The original account contained a complex left branch in one of the entries, and since the two lexical entries were not connected in any way, it led to the so-called mouse-elephant issue (cf. Caha 2021c).

(624) DERIVATION OF 'ALBUM', NOM, INTERMEDIATE STEP



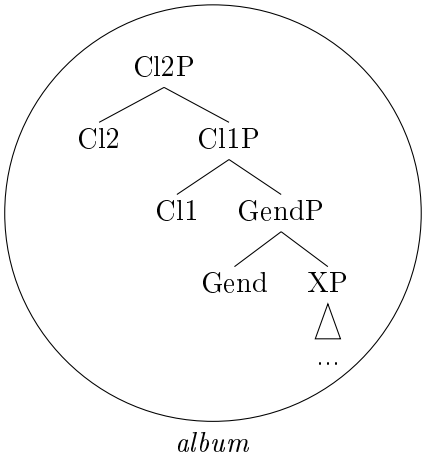
Then we merge the Cl2 feature:

(625) DERIVATION OF 'ALBUM', NOM, INTERMEDIATE STEP



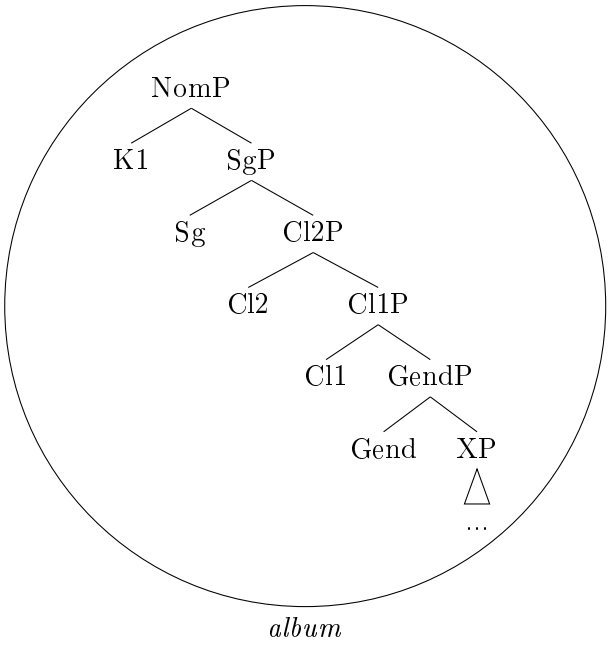
The root *alb* can no longer spell the structure out. However, the match is now found in the entry for *album* because that one has a Cl2P above Cl1P, pointing towards a structure previously spelled out as *alb*:

(626) DERIVATION OF 'ALBUM', NOM, INTERMEDIATE STEP



The rest of the nominative derivation continues the same, by merging first Sg and then K1 and spelling it out as *album* each time:

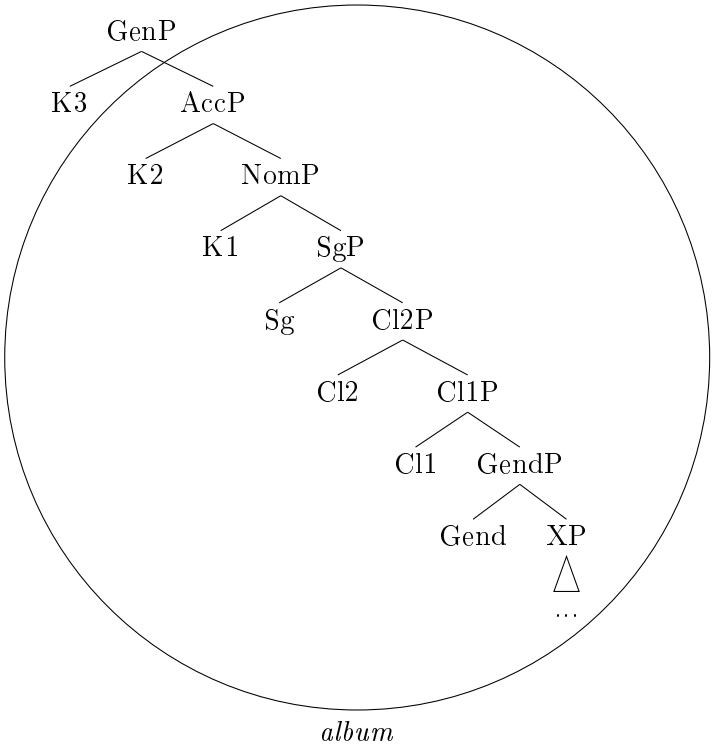
(627) DERIVATION OF 'ALBUM', NOM, FINAL FORM



The accusative will be derived by merging K2 on top and spelling the structure out as it is, also by the entry *album*.

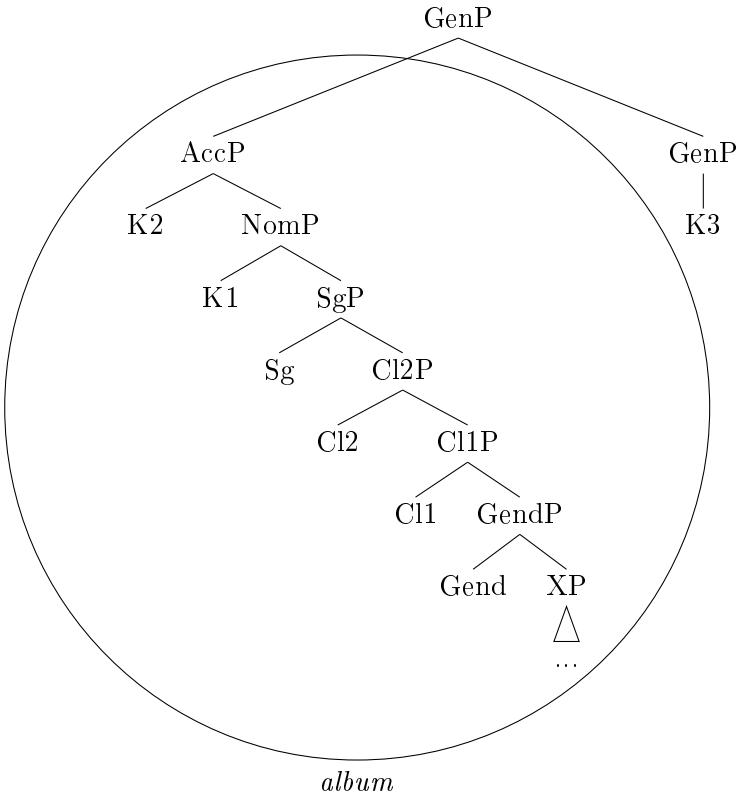
Let us see what happens when we merge the genitive feature K3:

(628) DERIVATION OF 'ALBUM', GEN, INTERMEDIATE STEP



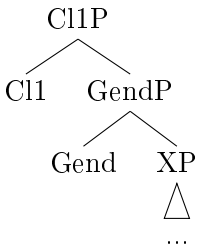
No lexical entry can spell out the structure above, so the complement movement gets triggered:

(629) DERIVATION OF 'ALBUM', GEN, INTERMEDIATE STEP



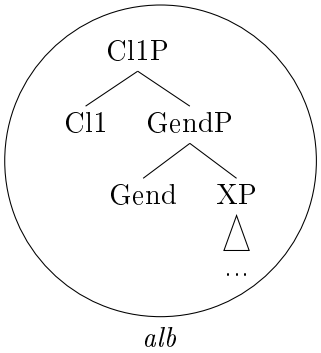
No lexical entry can spell out the new structure in its entirety, and, at the same time, no lexical entry can spell out the right branch on its own. The derivation is more complex than what we are going to show here, but the crucial part is that, ultimately, the structure will backtrack to the Cl1P stage to find an entry that can spell out the K3 feature:

(630) DERIVATION OF 'ALBUM', GEN, INTERMEDIATE STEP



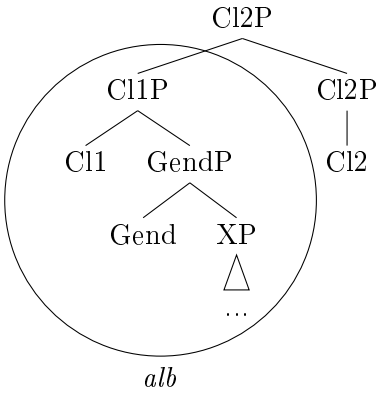
The best fit for this structure is again the smaller lexical entry, *alb*:

(631) DERIVATION OF 'ALBUM', GEN, INTERMEDIATE STEP



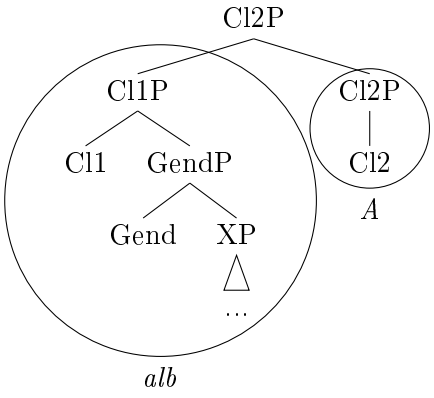
After merging Cl2P again, we try the next movement that was not tried before, and we move the complement over Cl2:

(632) DERIVATION OF 'ALBUM', GEN, INTERMEDIATE STEP



The right branch can now be spelled out by A:

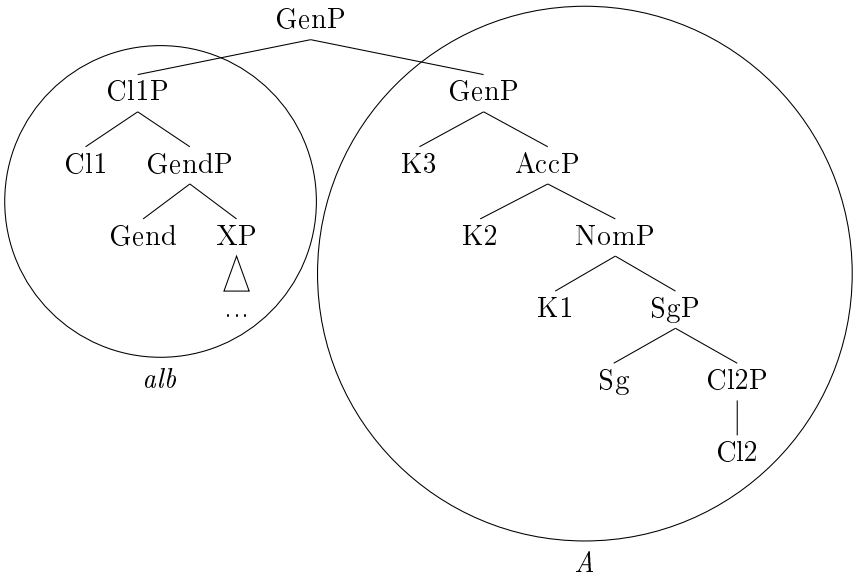
(633) DERIVATION OF 'ALBUM', GEN, INTERMEDIATE STEP



The rest of the derivation proceeds by merging feature after feature, bringing the Spec

on top of the structure and always spelling the left branch as *alb* and the right branch as *A* until we again reach the genitive stage:

(634) DERIVATION OF 'ALBUM', GEN, INTERMEDIATE STEP



Comparing the nominative/accusative derivation with the genitive one shows that having two lexical entries for a root can capture root allomorphy if we decide that *album* is an example of that.

It is not completely clear to us which option covers the data better – whether the root allomorphy or the private lexical entries. However, we need a root allomorphy account for Czech regardless, which is why we are inclined to go with that one for *album* as well.

One of the clearest examples calling for a root allomorphy account is the allomorphy for the root *Zeus* 'Zeus (the Greek god)'. Suffix-wise, it is the same paradigm as the masculine animate *pán* 'sir'. Its uniqueness lies in the fact that in any other case besides the nominative, the shape of its root changes to *Di-* in standard Czech:

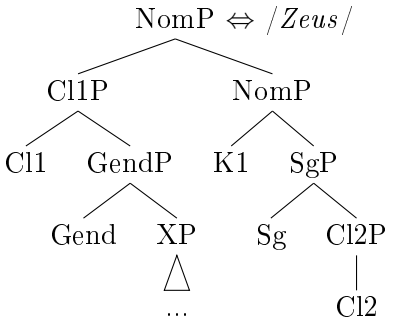
(635) SURFACE FORMS OF 'ZEUS'

	'Zeus' (Ma)
NOM	Zeus
ACC	Di-a
GEN	Di-a
LOC	Di-ovi
DAT	Di-ovi
INSTR	Di-em

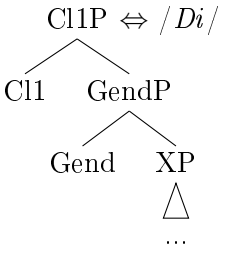
Remember that the paradigm *pán* 'sir' has the complex left branch on top of its lexical entry structure, so we would expect the same for *Zeus* (see (636)). We could suggest

that the alternating root *Di-* is specified exactly for features that match the left branch of the complex lexical entry for *Zeus* (637):

(636) 'Zeus', allomorphy entry



(637) 'Zeus', allomorphy entry



In the derivation, the nominative will be spelled out by the complex structure in (636). However, after merging the accusative layer K2, the entry for *Zeus* will not be able to spell out the whole tree, so the left branch in its Spec will move out. The left branch on its own will find a better candidate in the smaller simple entry in (637), spelled out as *Di-*, while the right branch will be spelled out by the suffixal element *A*.

Unlike for *album*, there is probably no way around such an account. Given the complete allomorphy of *Ze(us)* and *Di-*, both roots need to be stored separately. This might be a clue towards choosing the root allomorphy account for *album* as well.

The main message here is that whatever account we end up choosing for *album*, the one with root-allomorphy, or with the private lexical entries, both can work well with the rest of the system suggested in this work and, as such, extend its explanatory power.

7 Conclusions

To sum up, we proposed an account for the Czech case suffix allomorphy in nouns in the singular. The central part of the account builds on the intuition that, in the jungle of Czech nominal morphology, some paradigms seem to be in a relationship. Take, e.g., the paradigms *jazyk* 'tongue' and *pokoj* 'room'. On the surface, they show different sets of suffixes, sharing only the instrumental:

(638) SURFACE FORMS OF 'TONGUE' AND 'ROOM'

	'tongue' (Mi)	'room' (Mi)
NOM	jazyk	pokoj
ACC	jazyk	pokoj
GEN	jazyk-a	pokoj-e
LOC	jazyk-u	pokoj-i
DAT	jazyk-u	pokoj-i
INSTR	jazyk-em	pokoj-em

At the same time, they both have the same syncretism patterns – no visible suffix in the nominative and accusative, then a unique suffix for the genitive, a syncretism between the locative and dative, and then a unique suffix for the instrumental.

If we pair paradigms based on their syncretism patterns, their suffixes seem to hold a one-directional implicational relationship. If a paradigm has *-a* or *-o* in a certain case, the other will have *-e* in the same morphosyntactic context. If it has *-u*, the other will have *-i*:

(639) IMPLICATIONAL RELATIONSHIP BETWEEN PALATALIZED AND NON-PALATALIZED SUFFIXES

NON-PAL		PAL
a	→	e
u	→	i
o	→	e

The Element Theory decomposition uncovers even more regularity. It seems to hold that, abstractly, the implication is: whatever vowel there is in the base paradigm, the other paradigm will have the same vowel combined with -I (see (640)). A concrete example for the genitive of the *jazyk* → *pokoj* paradigms can be seen in (641):

(640) V → V + I

(641) A [a] → A + I [e]

The difference between the paradigms in such a relationship boils down to the presence or absence of the extra I, which we argue to be a floating element at the end of the some roots. And since I is a known palatalizer and the roots with it show root-final palatalization, we talk about the palatalized and non-palatalized counterparts. This goes well with the traditional division of paradigms into palatalized and non-palatalized,

also called "hard" and "soft" in Czech literature. One novel part of our account is a technical implementation of this traditional intuition.

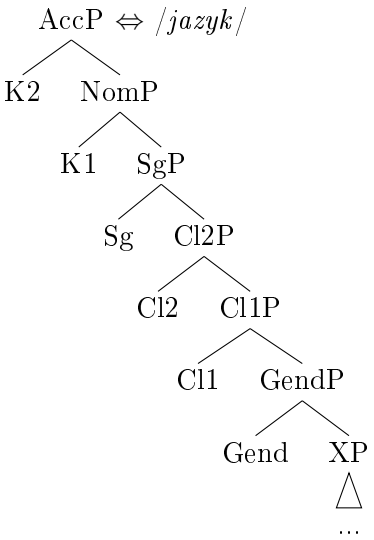
The unification brings a considerable simplification to the system. See below how instead of twelve separate paradigm types, we can now have only five (note that these are not all that we work with):

(642) PALATALIZATION COUNTERPARTS

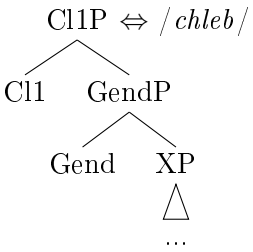
1	jazyk 'tongue'	pokoj 'room'	dlaň 'palm'	kuře 'chicken'
2	hrad 'castle'	kost 'bone'		
3	pán 'sir'	muž 'man'		
4	sako 'jacket'	moře 'sea'		
5	chleba 'bread'	soudce 'judge'		

These paradigm types are then accounted for by suggesting different root sizes. So, e.g., the fact that the root *jazyk* 'tongue' takes no suffix in the nominative but *chleb* 'bread' takes *-a* is argued to be a difference in their respective sizes:

(643) LEXICAL ENTRY OF 'TONGUE'



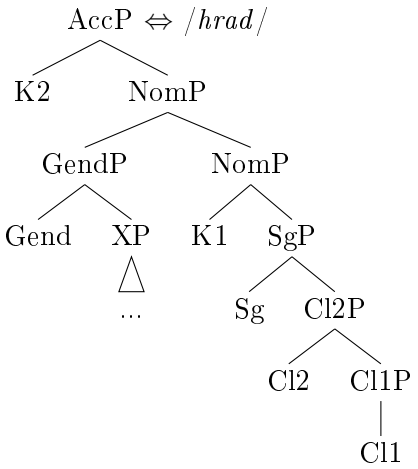
(644) LEXICAL ENTRY OF 'BREAD'



The root *jazyk* 'tongue' spells out all the relevant features in the nominative and therefore does not need to be finished by an affix. On the other hand, *chleb* 'bread' does not spell out all the relevant structure, and for a successful derivation, it will be completed by the most fitting morpheme, in this case will be *-a*. In Nanosyntax, all of this is taken care of by the phrasal spellout and the spellout algorithm.

Careful consideration of all the paradigms leads us to propose four root sizes, which we call XL, L, M, and S. The XL size is, furthermore, split into two based on the shape of their lexical entries. There are simple lexical entries like we already saw with *jazyk* 'tongue' in (643). The other group, with complex entries, has a branching shape, like with the example of *hrad* 'castle' below:

(645) LEXICAL ENTRY OF 'CASTLE'



Branching lexical entries have one noteworthy property. Given how the matching algorithm is set up in Nanosyntax, it is possible to spell out a syntactic structure using any subset of a matching lexical entry. For an entry with a complex left branch, it is enough if the left branch (or its proper subset) matches the structure.

This property comes in handy when comparing *hrad* 'castle' to another paradigm that has the same suffixes except for the genitive:

(646) SURFACE FORMS OF 'TONGUE' AND 'CASTLE'

	'tongue' (Mi)	'room' (Mi)
NOM	jazyk	hrad
ACC	jazyk	hrad
GEN	jazyk-a	hrad-u
LOC	jazyk-u	hrad-u
DAT	jazyk-u	hrad-u
INSTR	jazyk-em	hrad-em

On the one hand, their nominative forms seem to hint at the two roots being of the same size, since they both seem to spell out all the features up to the accusative layer. On the other hand, they need to differ in something, otherwise, they would not take different endings in the genitive. Note that the difference is unlikely to be phonological or featural. The difference in shape captures the suffix distribution well.

The proposed account shows how we can make sense of a complex morphological system with extended syncretisms using just standard tools of Nanosyntax, combined with the Element Theory in places where a phonological insight is necessary. The root size and the root shape, i.e., the concrete packaging of the universal functional sequence, are at the heart of the matter.

8 Appendix

In this appendix, we summarize the final account of the paradigms dealt with in sections 3 to 5. The few missing paradigms and open-ended details are left out as they require further research.

8.1 Paradigms

(647) SURFACE FORMS OF THE XL-SIZED GROUP WITH SIMPLE LEXICAL ENTRIES

	'tongue' (Mi)	'room' (Mi)	'palm' (F)
NOM	jazyk	pokoj	dlaň
ACC	jazyk	pokoj	dlaň
GEN	jazyk-a	pokoj-e	dlan-ě
LOC	jazyk-u	pokoj-i	dlan-i
DAT	jazyk-u	pokoj-i	dlan-i
INSTR	jazyk-em	pokoj-em	dlan-í

(648) SURFACE FORMS OF THE XL-SIZED GROUP WITH COMPLEX LEXICAL ENTRIES

	'castle' (Mi)	'bone' (F)
NOM	hrad	kost
ACC	hrad	kost
GEN	hrad-u	kost-i
LOC	hrad-u	kost-i
DAT	hrad-u	kost-i
INSTR	hrad-em	kost-í

(649) SURFACE FORMS OF THE L-SIZED GROUP

	'sir' (Ma)	'man' (Ma)
NOM	pán	muž
ACC	pán-a	muž-e
GEN	pán-a	muž-e
LOC	pán-u	muž-i
DAT	pán-u	muž-i
INSTR	pán-em	muž-em

(650) SURFACE FORMS OF THE M-SIZED GROUP

	'jacket' (N)	'sea' (N)
NOM	sak-o	moř-e
ACC	sak-o	moř-e
GEN	sak-a	moř-e
LOC	sak-u	moř-i
DAT	sak-u	moř-i
INSTR	sak-em	moř-em

(651) SURFACE FORMS OF THE S-SIZED GROUP

	‘bread’ (Mi)	‘judge’ (Ma)
NOM	chleb-a	soudc-e
ACC	chleb-a	soudc-e
GEN	chleb-a	soudc-e
LOC	chleb-u	soudc-e
DAT	chleb-u	soudc-e
INSTR	chleb-em	soudc-e

8.2 Lexicalization tables

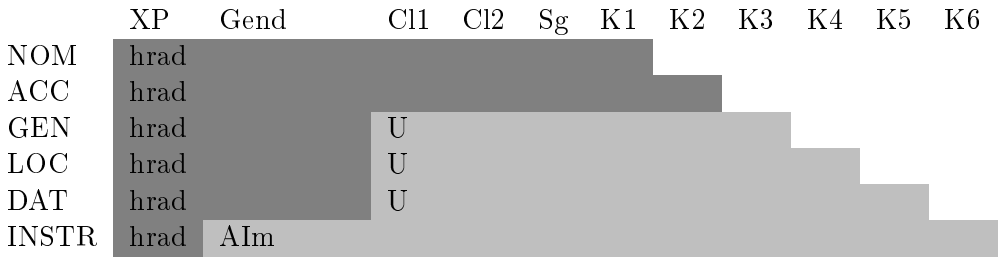
(652) LEXICALIZATION TABLES OF THE XL-SIZED GROUP WITH SIMPLE LEXICAL ENTRIES, ‘TONGUE’, ‘ROOM’ AND ‘PALM’

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	jazyk										
ACC	jazyk										
GEN	jazyk			A							
LOC	jazyk		U								
DAT	jazyk		U								
INSTR	jazyk	AIm									

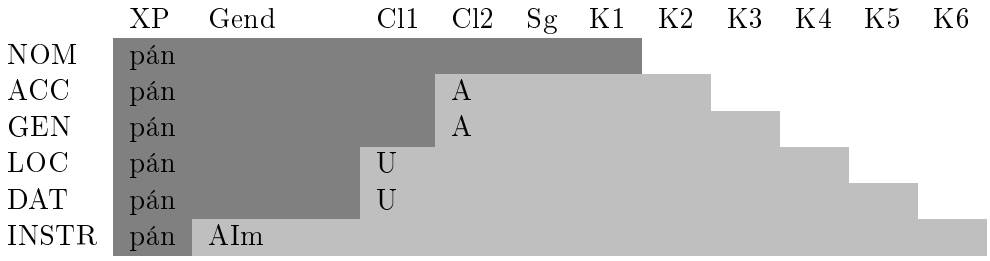
	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	pokojI										
ACC	pokojI										
GEN	pokojI			A							
LOC	pokojI		U								
DAT	pokojI		U								
INSTR	pokojI	AIm									

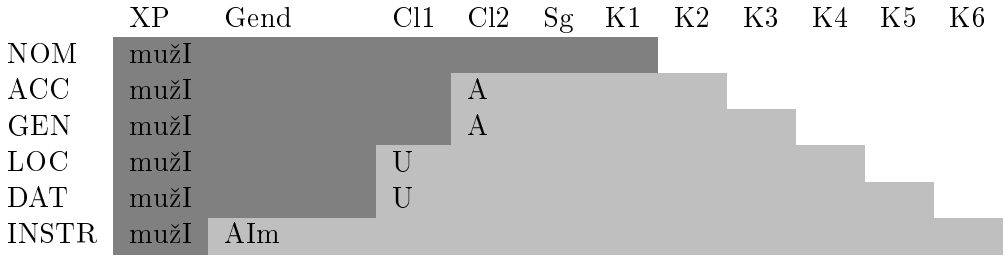
	XP	Gend	Mkd	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	dlaňI											
ACC	dlaňI											
GEN	dlaňI				A							
LOC	dlaňI			U								
DAT	dlaňI			U								
INSTR	dlaňI		U+length									

(653) LEXICALIZATION TABLES OF THE XL-SIZED GROUP WITH COMPLEX LEXICAL ENTRIES, 'CASTLE', 'BONE'



(654) LEXICALIZATION TABLES OF THE L-SIZED GROUP, 'SIR' AND 'MAN'





(655) LEXICALIZATION TABLES OF THE M-SIZED GROUP, 'JACKET' AND 'SEA'

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	sak				AU						
ACC	sak				AU						
GEN	sak			A							
LOC	sak		U								
DAT	sak		U								
INSTR	sak	AIm									

	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	mořI				AU						
ACC	mořI				AU						
GEN	mořI			A							
LOC	mořI		U								
DAT	mořI		U								
INSTR	mořI	AIm									

(656) LEXICALIZATION TABLES OF THE S-SIZED GROUP, 'BREAD' AND 'JUDGE'

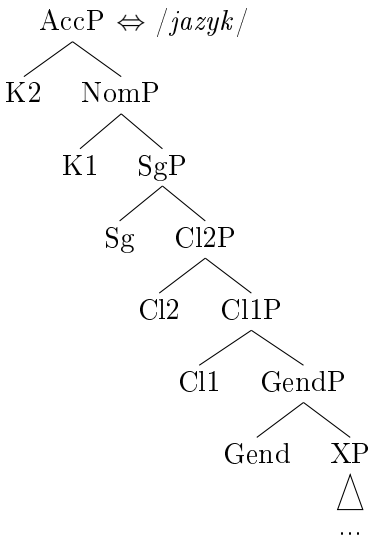
	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	chleb			A							
ACC	chleb			A							
GEN	chleb			A							
LOC	chleb		U								
DAT	chleb		U								
INSTR	chleb	AIm									

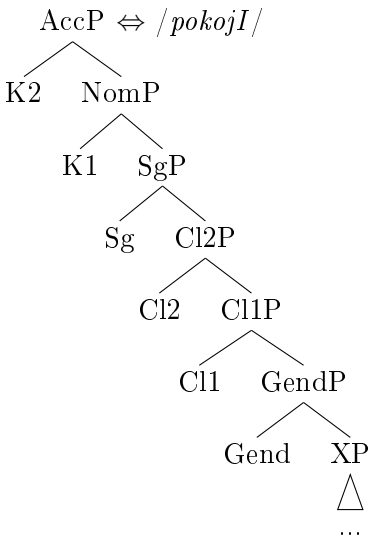
	XP	Gend	Cl1	Cl2	Sg	K1	K2	K3	K4	K5	K6
NOM	soudcI			A							
ACC	soudcI			A							
GEN	soudcI			A							
LOC	soudcI		U								
DAT	soudcI		U								
INSTR	soudcI	AIm									

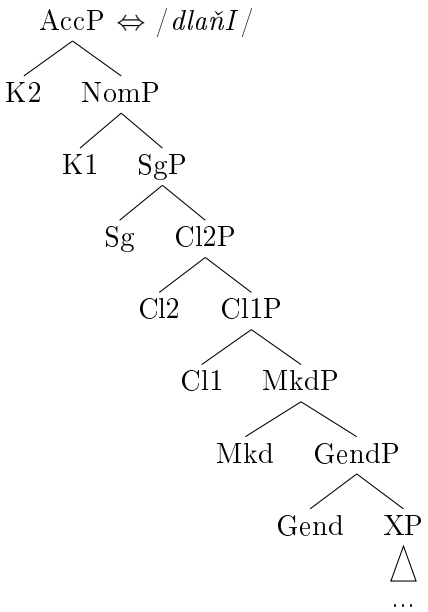
8.3 Lexical entries

8.3.1 Roots

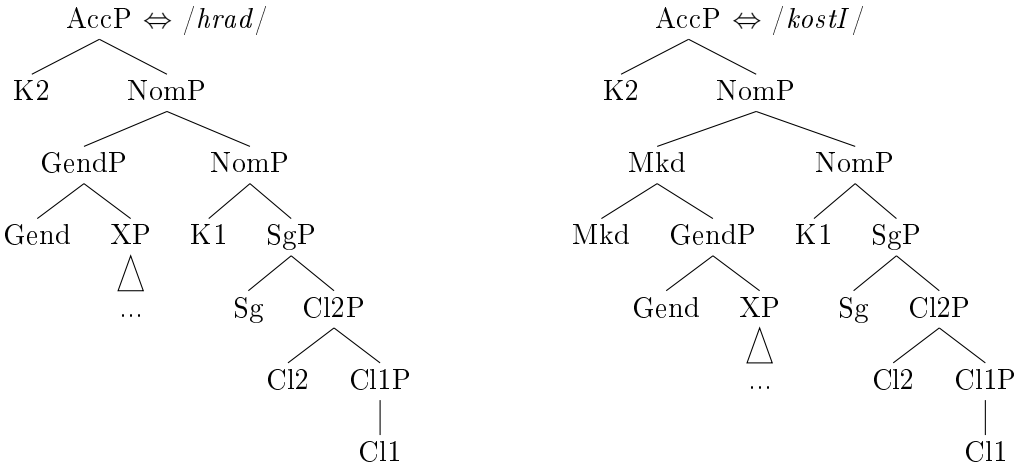
(657) LEXICAL ENTRIES OF THE XL-SIZED ROOTS WITH SIMPLE ENTRIES, 'TONGUE', 'ROOM', AND 'PALM'



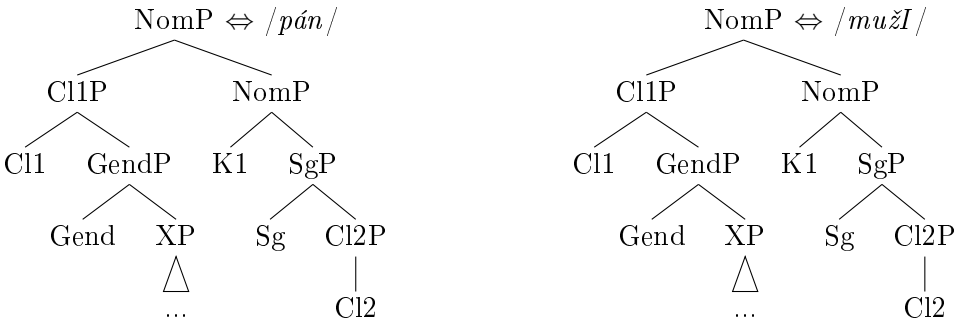




(658) LEXICAL ENTRIES OF THE XL-SIZED ROOTS WITH COMPLEX ENTRIES, 'CASTLE' AND 'BONE'

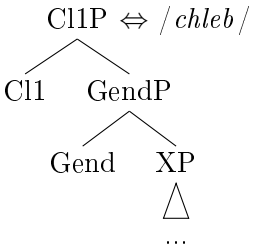


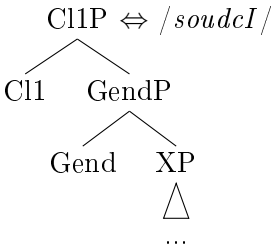
(659) LEXICAL ENTRIES OF THE L-SIZED ROOTS, 'SIR' AND MUŽ 'MAN'



(660) LEXICAL ENTRIES OF THE M-SIZED ROOTS, 'JACKET' AND 'SEA'

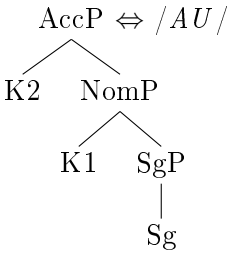
(661) LEXICAL ENTRY OF THE S-SIZED ROOT, 'BREAD' AND 'JUDGE'



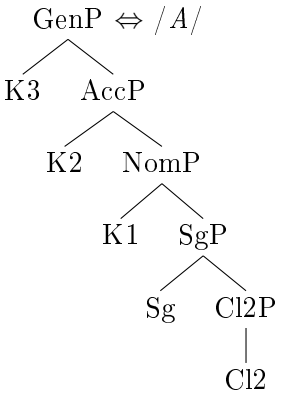


8.3.2 Case suffixes

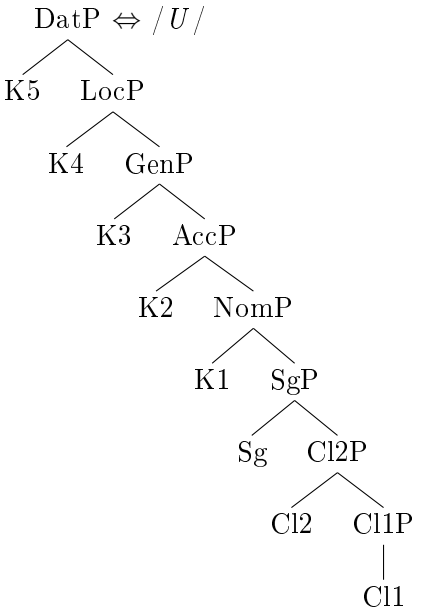
(662) LEXICAL ENTRY OF -AU



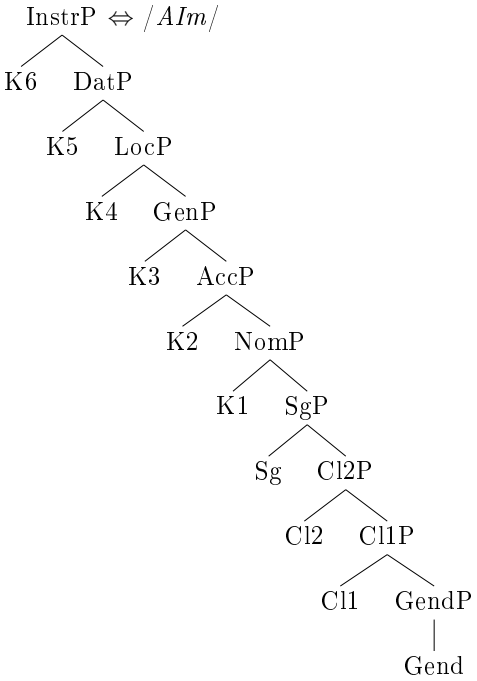
(663) LEXICAL ENTRY OF -A



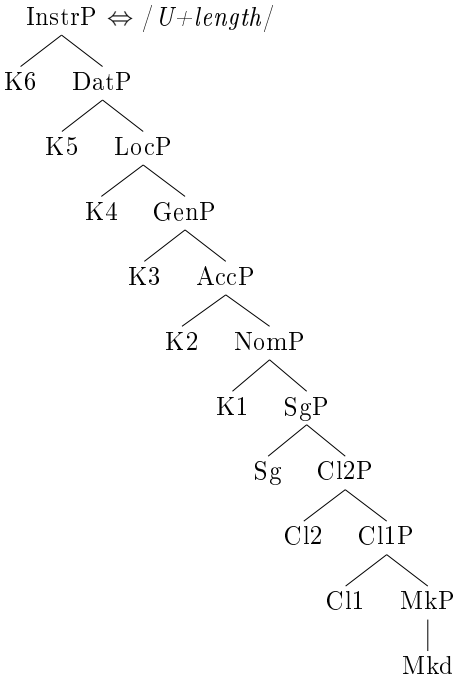
(664) LEXICAL ENTRY OF -U



(665) LEXICAL ENTRY OF -AIM



(666) LEXICAL ENTRY OF -U+LENGTH



8.4 Principles of Nanosyntax

(667) THE SUPERSET PRINCIPLE, Starke (2009)

A lexically stored tree matches a syntactic node iff the lexically stored tree contains the syntactic node.

(668) THE ELSEWHERE CONDITION, Kiparsky (1973), cited from Caha (2021c)

When two entries can spell out a given node, the more specific entry wins. The more specific entry is the one which has fewer features.

(669) SPELLOUT ALGORITHM, Caha (2021c), based on Starke (2018)

a. Merge F and spell out.

b. If (a) fails, move the Spec of the complement and spell out.

c. If (b) fails, move the complement of F and spell out.

(670) BACKTRACKING, Caha (2021c), based on Starke (2018)

When spellout fails, go back to the previous cycle, and try the next option for that cycle.

References

Aissen, J. (2003). Differential Object Marking: Iconicity vs. Economy. Natural Language and Linguistic Theory, 21(3), 435–483.

Alexiadou, A., and Müller, G. (2008). Class Features as Probes. In A. Bachrach, and A. Nevins (Eds.) Inflectional Identity, (pp. 101–155). Oxford: Oxford University Press.

Backley, P. (2011). Introduction to Element Theory. Edinburgh University Press.

Balhar, J. (2002). Český jazykový atlas 4. Praha: Academia.
URL https://cja.ujc.cas.cz/CJA4/

Baltoslav (2019). Baltoslav: Upper Sorbian Grammar. Accessed on January 4, 2022.
URL https://baltoslav.eu/hsb/index.php?b=rzem&mova=en

Baunaz, L., and Lander, E. (2018). Nanosyntax: The Basics. In L. Baunaz, K. De Clercq, L. Haegeman, and E. Lander (Eds.) Exploring Nanosyntax, (pp. 3–56). Oxford: Oxford University Press.

Bethin, C. Y. (1998). Slavic Prosody: Language Change and Phonological Theory. Cambridge: Cambridge University Press.

Blix, H. (2021). Phrasal Spellout and Partial Overwrite: On an Alternative to Backtracking. Glossa: A Journal of General Linguistics, 6(1).
URL https://ling.auf.net/lingbuzz/005793

Blix, H. (in print). Interface Legibility and Nominal Classification: A Nanosyntactic Account of Kipisigis Singulatives. Glossa: A Journal of General Linguistics.

Bobaljik, J. (2012). Universals in Comparative Morphology. MIT Press: Cambridge, MA.

Caha, P. (2009). The Nanosyntax of Case. Ph.D. thesis, University of Tromsø.
URL https://ling.auf.net/lingbuzz/000956

Caha, P. (2018). Notes on Insertion in Distributed Morphology and Nanosyntax. In L. Baunaz, K. De Clercq, L. Haegeman, and E. Lander (Eds.) Exploring Nanosyntax, (pp. 57–87). Oxford: Oxford University Press.
URL https://ling.auf.net/lingbuzz/002855

Caha, P. (2019a). Case Competition in Nanosyntax: A Study of Numeral Phrases in Ossetic and Russian. Berlin: Language Science Press.
URL https://ling.auf.net/lingbuzz/004875

Caha, P. (2019b). Syncretism as Merge F: The Nanosyntax of Case Ten Years On. In P. Caha, K. De Clercq, and G. Vanden Wyngaerd (Eds.) The Unpublished Manuscript, (pp. 19–38).
URL https://ling.auf.net/lingbuzz/004340

Caha, P. (2021a). Comparatives in Czech. Class taught in the fall 2021 at the Masaryk University, Brno.

Caha, P. (2021b). The Marking of Mass, Count and Plural Denotations in Multi-Dimensional Paradigms. Studia Linguistica.
URL https://ling.auf.net/lingbuzz/006084

Caha, P. (2021c). Modeling Declensions without Declension Features. The Case of Russian. Acta Linguistica Academica, 68(4), 385–425.
URL https://ling.auf.net/lingbuzz/005537

Caha, P., De Clercq, K., and Vanden Wyngaerd, G. (2019a). The Fine Structure of the Comparative. Studia Linguistica, 73(3), 470–521.
URL https://ling.auf.net/lingbuzz/003790

Caha, P., De Clercq, K., and Vanden Wyngaerd, G. (2019b). On the Difference between a $\sqrt{}$ and a Root.
URL https://ling.auf.net/lingbuzz/004391

Caha, P., and Pantcheva, M. (2012). Contiguity beyond linearity. Talk at Decennium: The first 10 years of CASTL.

Cvrček, V., and Vondřička, P. (2011). SyD – korpusový průzkum variant. Praha: FF UK. Accessed on June 26, 2021.
URL http://syd.korpus.cz

De Clercq, K. (2013). A Unified Syntax of Negation. Ph.D. thesis, Universiteit Gent, Gent.
URL https://ling.auf.net/lingbuzz/001973

DeClercq, K., and Vanden Wyngaerd, G. (2019). On the Idiomatic Nature of Unproductive Morphology. In Linguistics in the Netherlands 2019. Benjamins.
URL https://ling.auf.net/lingbuzz/004764

Dvonč, L., Horák, G., Miko, F., Mistrík, J., Oravec, J., Ružička, J., and Urbančok, M. (Eds.) (1966). Morfológia slovenského jazyka. Bratislava: Vydavateľstvo Slovenskej akadémie vied.

Encrevé, P. (1988). La liaison avec et sans enchaînement. Paris: Seuil.

Findejsová, L. (2017). Názvy zvířecích druhů a jejich zástupců: vzorce a jejich frekvence. Bc., Masaryk University.

Grepl, M., Karlík, P., Nekula, M., and Rusínová, Z. (2012). Příruční mluvnice češtiny. Praha: NLN.

Guzmán Naranjo, M., and Bonami, O. (2021). Overabundance and Inflectional Classification: Quantitative Evidence from Czech. Glossa: A Journal of General

Linguistics, 6(1), 1–31.
URL https://doi.org/10.5334/gjgl.1626

Halle, M. (1997). Distributed Morphology: Impoverishment and Fission. In B. Bruening, Y. Kang, and M. McGinnis (Eds.) MITWPL 30: Papers at the Interface, (pp. 425–449). Cambridge: MIT.
URL https://ling.auf.net/lingbuzz/000084

Halle, M., and Vaux, B. (1998). Theoretical Aspects of Indo-European Nominal Morphology: The Nominal Declensions of Latin and Armenian. In J. Jasanoff, H. Melchert, and L. Olivier (Eds.) Mir Curad: Studies in Honor of Clavert Watkins, (pp. 223–240). Innsbruck: Institut für Sprachwissenschaft der Universität Innsbruck.

Harris, J., and Lindsey, G. (1995). The Elements of Phonological Representation. In J. Durand, and F. Katamba (Eds.) Frontiers of Phonology: Atoms, Structures, Derivations. Harlow: Longman.

Havlík, A. (1889). K otázce jerové v staré češtině. Listy Filologické, 16.

Havránek, B., and Jedlička, A. (1992). Stručná mluvnice česká. Praha: Státní pedagogické nakladatelství.

Janků, L., and Starke, M. (2019). Class Meets Gender in Italian and Spanish: A Nanosyntactic Account. Talk at Going Romance 2019, Leiden.

Karlík, P., and Veselovská, L. (2017). Posesivní -ův-/-in- adjektivum. In P. Karlík, M. Nekula, and J. Pleskalová (Eds.) CzechEncy – Nový encyklopedický slovník češtiny. Accessed on February 1, 2022.
URL https://www.czechency.org/slovnik/POSESIVN%C3%8D%20-%C5%AEV-/-IN-ADJEKTIVUM

Kaye, J., Lowenstamm, J., and Vergnaud, J.-R. (1985). The Internal Structure of Phonological Elements: A Theory of Charm and Government. In C. Ewen, and J. Anderson (Eds.) Phonology Yearbook, vol. 2, (pp. 305–328). Cambridge University Press.

Kiparsky, P. (1973). 'Elsewhere' in Phonology. In P. Kiparsky, and S. R. Anderson (Eds.) A Festschrift for Morris Halle. New York: Holt, Rinehart and Winston.

Kodýtek, V. (2010). Podstatná jména (substantiva). In V. Cvrček (Ed.) Mluvnice současné češtiny. Praha: Univerzita Karlova v Praze, Nakladatelství Karolinum.

Křen, M., Cvrček, V., Henyš, J., Hnátková, M., Jelínek, T., Kocek, J., Kováříková, D., Křivan, J., Milička, J., Petkevič, V., Procházka, P., Skoumalová, H., Šindlerová, J., and Škrabal, M. (2020). Syn2020: reprezentativní korpus psané češtiny. Accessed on January 26, 2022.
URL http://www.korpus.cz

Lampitelli, N. (2014). The Romance Plural Isogloss and Linguistic Change: A Comparative Study of Romance Nouns. Lingua, 140, 158–179.
URL https://ling.auf.net/lingbuzz/001514

Lowenstamm, J. (1996). CV as the Only Syllable Type. In J. Durand, and B. Laks (Eds.) Current Trends in Phonology. Models and Methods, (pp. 419–441). Salford, Manchester: ESRI.

Müller, G. (2004). A Distributed Morphology Approach to Syncretism in Russian Noun Inflection. Proceedings of FASL, 12, 353–373.
URL https://ling.auf.net/lingbuzz/000093

Najbrtová, K. (2013). Zavolal jsem si taxík/taxíka – životná koncovka neživotných maskulin v akuzativu singuláru. In B. Bednaříková, and P. Hernandezová (Eds.) Od slova k modelu jazyka. Sborník z 13. mezinárodního setkání mladých lingvistů, (p. 266–281).

Newell, H. (2017). English Lexical Levels Are Not Lexical, But Phonological.
URL https://ling.auf.net/lingbuzz/003898

Pantcheva, M. (2011). Decomposing Path : The Nanosyntax of Directional Expressions. Ph.D. thesis, University of Tromsø.
URL https://ling.auf.net/lingbuzz/001351

Paradis, C., and Prunet, J.-F. (2000). Nasal Vowels as Two Segments: Evidence from Borrowings. Language, 76(2), 324–357.

Passino, D. (2009). An Element-Based Analysis of Italian Nominal Inflection. In F. Montermini, G. Boyé, and J. Tseng (Eds.) Selected Proceedings of the 6th Décembrettes. Morphology in Bordeaux, (pp. 63–75). Cascadilla.

Pešek, O. (2017). Galicismy v českém lexiku. In P. Karlík, M. Nekula, and J. Pleskalová (Eds.) CzechEncy – Nový encyklopedický slovník češtiny. Prague: NLN.
URL https://www.czechency.org/slovnik/GALICISMY%20V%20%C4%8CESK%C3%89M%20LEXIKU

Pleskalová, J. (2001). Stará čeština pro nefilology. Brno: Masarykova univerzita.

Pugh, S. M., and Press, I. (1999). Ukrainian: A Comprehensive Grammar. Routledge.

Rodríguez-Mondoñedo, M. (2009). The Acquisition of Differential Object Marking in Spanish.
URL http://ling.auf.net/lingbuzz/000350

Scheer, T. (1996). Une théorie de l'interaction directe entre consonnes. Ph.D. thesis, University of Paris 7.

Scheer, T. (2004). A Lateral Theory of Phonology. Vol. 1: What Is CVCV, and Why Should It Be?. Berlin: Mouton de Gruyter.

Scheer, T. (2005). Slavic Vowel-Zero Alternations and Government Phonology: Two Approaches, One Solution. In S. Franks, F. Gladney, and M. Tasseva-Kurktchieva (Eds.) Formal Approaches to Slavic Linguistics, 13, (pp. 300–311). Ann Arbor: Michigan Slavic Publications.
URL https://ling.auf.net/lingbuzz/000073

Scheer, T. (2006). How Yers Made Lightner, Gussmann, Rubach, Spencer and Others Invent CVCV. In P. Bański, B. Łukaszewicz, and M. Opalińska (Eds.) Studies in Constraint-Based Phonology, (pp. 133–207). Warsaw: Wydawnictwo Uniwersytetu Warszawskiego.

Scheer, T., and Ziková, M. (2009). The Havlík Pattern and Directional Lower. In W. Browne, A. Cooper, A. Fisher, E. Kesici, and N. Predolac (Eds.) Formal Approaches to Slavic Linguistics. The Second Cornell Meeting, 2009, (pp. 471–486). Ann Arbor: Michigan Slavic Publications.

Starke, M. (2002). The day syntax ate morphology. Class taught at the EGG summer school, Novi Sad.

Starke, M. (2009). Nanosyntax: A Short Primer to a New Approach to Language. Nordlyd, 36, 1–6.
URL https://ling.auf.net/lingbuzz/001230

Starke, M. (2014). Cleaning Up the Lexicon. Linguistic Analysis, 39, 245–256.
URL https://ling.auf.net/lingbuzz/004309

Starke, M. (2017). Resolving (DAT = ACC) ≠ GEN. Glossa: A Journal of General Linguistics, 2(1).
URL https://ling.auf.net/lingbuzz/003783

Starke, M. (2018). Complex Left Branches, Spellout, and Prefixes. In L. Baunaz, K. De Clercq, L. Haegeman, and E. Lander (Eds.) Exploring Nanosyntax, (pp. 239–249). Oxford: Oxford University Press.

Šulc, M. (2001). Životná koncovka -a v akuzativu singuláru neživotných maskulin. Jazykovědné aktuality, (38).
URL https://www.jazykovednesdruzeni.cz/wp-content/uploads/2020/04/2001_3.pdf

Swan, O. E. (1988). Facultative Animacy in Polish: A Study in Grammatical Gender Formation. The Carl Beck Papers in Russian and East European Studies, (606).
URL https://carlbeckpapers.pitt.edu/ojs/index.php/cbp/article/view/104/105

Taraldsen, T. (2010). The Nanosyntax of Nguni Noun Class Prefixes and Concords. Lingua, 120(6), 1522–1548.
URL https://ling.auf.net/lingbuzz/000876

Taraldsen, T. (2019). Nanosyntax and Syncretism in Multidimensional Paradigms. Linguistics Vanguard, 5(1).

Ústav pro jazyk český AV ČR (2022). Internetová jazyková příručka [online] (2008–2021). Accessed on January 23, 2022.
URL https://prirucka.ujc.cas.cz/

Vanden Wyngaerd, G. (2018). The Feature Structure of Pronouns: A Probe into Multidimensional Paradigms. In L. Baunaz, K. De Clercq, L. Haegeman, and E. Lander (Eds.) Exploring Nanosyntax, (pp. 277–304). Oxford: Oxford University Press.
URL https://ling.auf.net/lingbuzz/003166

Vanden Wyngaerd, G., De Clercq, K., and Caha, P. (2021). Late Insertion and Root Suppletion. Revista Virtual de Estudos da Linguagem - ReVEL.
URL https://ling.auf.net/lingbuzz/006043

Vanden Wyngaerd, G., Starke, M., De Clercq, K., and Caha, P. (2020). How to Be Positive. Glossa: A Journal of General Linguistics, 5(1).
URL https://ling.auf.net/lingbuzz/004806

Veselepohadky.cz (2022). Burákovy povídačky. Accessed on April 17, 2022.
URL https://www.veselepohadky.cz/burakovy-povidacky/

Večerka, R. (2006). Staroslověnština v kontextu slovanských jazyků. Olomouc: Univerzita Palackého v Olomouci.

Williams, A., Pimentel, T., Blix, H., McCarthy, A. D., Chodroff, E., and Cotterell, R. (2020). Predicting Declension Class from Form and Meaning. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, (p. 6682–6695). Association for Computational Linguistics.
URL https://aclanthology.org/2020.acl-main.597/

Ziková, M. (2007). Why Czech Case Markers Sometimes Get Lost. In M. Dočekal, P. Karlík, and J. Zmrzlíková (Eds.) Czech in Generative Grammar, (pp. 193–203). Munich: Lincom Europa.

Ziková, M. (2018). Licensing of Vowel Length in Czech. The Syntax-Phonology Interface. Berlin: Peter Lang.
URL https://ling.auf.net/lingbuzz/004328

Ziková, M., and Faltýnková, M. (2020). How to Derive Allomorphy: A Case Study from Czech. The Linguistic Review, 37.