Language technology for endangered languages: Sámi as a case study

Trond Trosterud

January 25, 2008

Seen from a computational point of view, all languages pose the same challenges

There is no reason to treat endangered and non-endangered differently

but...

Within language technology,

- there are economical demands for solutions for only a handful of languages.
- for most languages the basic tools for making lg tech are not readily available

For minority languages,

- there are not large amounts of texts and speech in electronic format
- reference grammars may be incomplete
- dictionaries and other semantic resources may be too small or lacking

On the positive side,

- we need not repeat costly mistakes made by the lg tech pioneers
- projects starting today may build clean, modern systems

Why language technology in the first place?

- Without well-developed language technology resources, no language will in the future be able to:
  - function as an administrative language
  - .... in a bilingual administration, or
  - be stored in digital archives

Language technology for administrative languages:

- Text must be proofread
- Different types of schemes and fill-in forms must be generated
- People will need summaries and abstracts

All this will be done automatically, with the help of language technology tools.

... for bilingual administrations:

- Multilingual versions of the same text — generated by machine translation
- Consistent use of multilingual terminology — checked automatically

... for text storage:

- Source engines will rely upon language technology in order to classify and retrieve text.
- Language-independent content search

Different types of text-based language technology

Technical Localisation

Grammatical

Semantical

Multilingual

Text-to-speech

Localisation

... is everything that makes the computer aware of

- what language the user is using
- what country he or she lives in
- character sets, keyboard layout, sorting, day-and-time, currency symbol...

Character sets – that give each letter and symbol a unique number

- There used to be 8-bit character sets for different regions and languages
- Now: Unicode – *the largest revolution since Gutenberg* – which makes all the letters in the world unambiguously available on every computer

Grammatical analysers

... take text or words as input and deliver a grammatical analysis, or vice versa.

- Analysis — based upon wordform lists or morphological transducers?
- Disambiguation of grammatical homonymy — based upon linguistics or statistics?

Wordform lists or transducers?

- List approaches can be used for languages with small paradigms and evenly-used wordforms
- For larger inflectional paradigms, there is no guarantee that all forms will be covered
- ==> Such languages will need morphological transducers

Test: wordform lists or transducers?

Check of coverage for high- and mid-frequency verb:

	Corpus type	Words	High-frequency verb	Mid-frequency verb
Sámi	New Testament	145282	*dadjat*	*bálvalit*
Danish	New Testament	177051	*sige*	*tjene*
English	New Testament	188616	*say*	*serve*

	Danish						English			
Inf	**sige**	272	**tjene**	37		Inf	**say**	422	**serve**	33
Prs	**siger**	627	**tjener**	16		Pst	**said**	1058	**served**	5
Pst	**sagde**	1289	**tjente**	9		Ger	**saying**	408	**serving**	6
PftPtc	**sagt**	125	tjent	0		2Sg			**serveth**	5
Imp	**sig**	860	tjen	0						
PrsPrt	**sigende**	4	**tjenende**	2						

Legend:

black = found in corpus (number of hits), grey = *not* found in corpus.

Northern Sámi *dadjat* 'to say', in the NT

	Ind.Present		Ind.Past		Cond.Pres		Pot.Pres		Imperative	
Sg1	**dajan**	4	dadjen		**dajašin**	1	dajažan		dadjon	
Sg2	**dajat**	6	**dadjet**	48	dajašit		dajažat		**daja**	=
Sg3	**dadjá**	42	**dajai**	207	**dajašii**	4	**dajaš(a)**	1	dadjos	
Du1	**dadje**	=	dajaime		dajašeimme		dajažetne		daddju	
Du2	**dadjabeahtti**	1	dajaide		dajašeidde		dajažeahppi		daddji	
Du3	dadjaba		**dajaiga**	7	dajašeigga		dajažeaba		dadjoska	
Pl1	**dadjat**	=	**dajaimet**	1	**dajašeimmet**		dajažit		dadjot	
Pl2	**dadjabehtet**	27	dajaidet		**dajašeiddet**	2	dajažehpet		daddjet	
Pl3	**dadjet**	=	**dadje**	183	dajaše(dje)		dajažit		dadjoset	
Neg	**daja**	4			**dajaše**	1	dajaš		**daja**	=
	Inf		PrsPrc		PrfPrc		VerbAbessive		Gerund	
	dadjat	47	**daddji**	4	**dadjan**	15	dajakeahttá		dajadettiin	

Northern Sámi *bálvalit* 'to serve', in the NT

	Ind.Present		Ind.Past		Cond.Pres		Pot.Pres		Imperative	
Sg1	**bálvalan**	=	Bálvalin		bálvaleaččan		bálvalivččen		bálvalehkon	
Sg2	bálvalat		**bálvalit**	=	bálvaleaččat		bálvalivččet		**bálval**	=
Sg3	**bálvala**	10	**bálvalii**	3	bálvaleažžá,-aš,-š		bálvalivččii		**bálvalehkos**	2
Du1	bálvaletne		**bálvaleimme**	1	bálvaležže		bálvalivččiime		bálvaleadnu,hkku	
Du2	bálvaleahppi		Bálvaleidde		bálvaleažžabeahtti		bálvalivččiide		bálvaleahkki	
Du3	bálvaleaba		Bálvaleigga		bálvaleažžaba		bálvalivččiiga		bálvalehkoska	
Pl1	**bálvalit**	=	bálvaleimmet		bálvaleažžat		bálvalivččiimet		bálvaleadnot	
Pl2	**bálvalehpet**	4	**bálvaleiddet**	1	bálvaleažžabehtet		bálvalivččiidet		**bálvaleahkket/ot**	2
Pl3	**bálvalit**	=	**bálvaledje**	2	bálvaležžet		bálvalivčče		bálvalehkoset	
Neg	**bálval**	5			bálvale(a)š, -čča,		bálvalivčče		**bálval**	=
	Inf		PrsPrc		PrfPrc		VerbAbessive		Gerund	
	bálvalit	39	**bálvaleaddji**	72	**bálvalan**	13	bálvalkeahttá		bálvalettiin	

Finnish *palvella* 'to serve' on the Internet

(among the 1000 most common Finnish words)

(N = 3.5 billion words, october 2004)

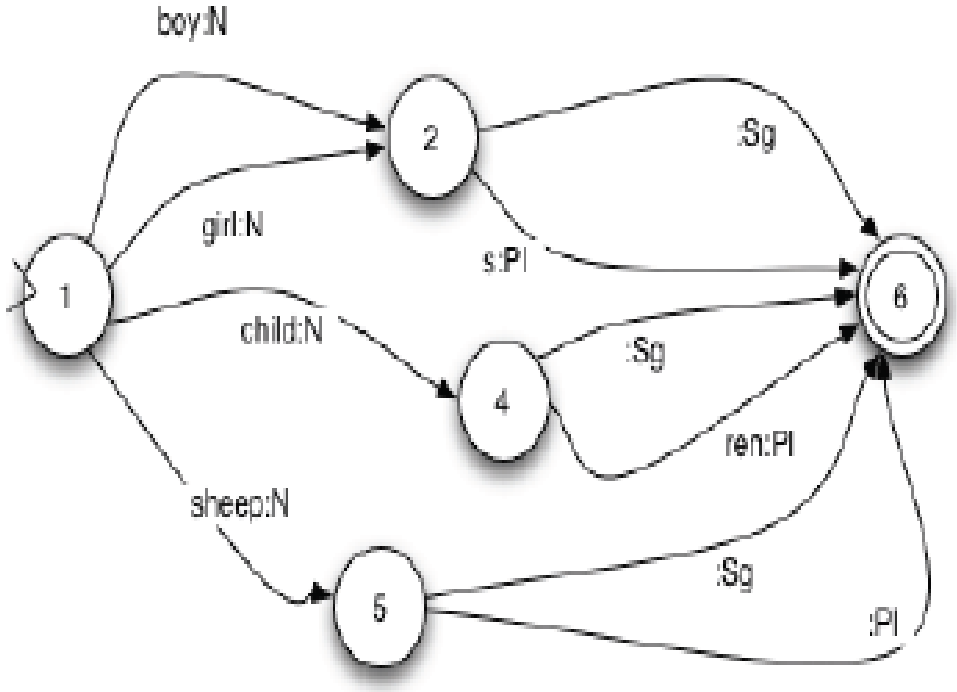
	Ind.Present		Ind.Past		Cond.Pres		Pot.Pres		Imperative	
Sg1	palvelen	931	palvelin	56300	palvelisin	69	palvellen	2060		
Sg2	palvelet	423	palvelit	63	palvelisit	22	palvellet	11	palvele	8370
Sg3	palvelee	94300	palveli	8650	palvelisi	6090	palvellee	44	palvelkoon	199
Pl1	palvelemme	28100	palvelimme	114	palvelisimme	160	palvellemme	3	palvelkaamme	85
Pl2	palvelettr	413	palvelittr	88	palvelisittr	37	palvellettr	2	palvelkaa	960
Pl3	palvelevat	38700	palvelivat	2340	palvelisivat	1500	palvellevat	9	palvelkoot	94

vapisuttaa ‘make shake, shiver’, on the Internet:

(among the 12000 most common Finnish words)

11536	Ind.Present		Ind.past		Cond.Pres		Pot.Pres		Imperative	
Sg1	**vapisutan**	2	vapisutin	0	vapisuttaisin	0	vapisuttanen	0		
Sg2	vapisutat	0	vapisutit	0	vapisuttaisit	0	vapisuttanet	0	**vapisuta**	7
Sg3	**vapisuttaa**	122	**vapisutti**	83	**vapisuttaisi**	1	vapisuttanee	0	vapisuttakoon	0
Pl1	vapisutamme	0	vapisutimme	0	vapisuttaisimme	0	vapisuttaneme	0	vapisuttakaamme	0
Pl2	vapisutatte	0	vapisutitte	0	vapisuttaisitte	0	vapisuttanette	0	vapisuttakaa	0
Pl3	**vapisuttavat**	6	**vapisuttivat**	13	vapisuttaisivat	0	**vapisuttanevat**	1	vapisuttakoot	0

The answer is a morphological transducer



dictionary with declension class info + morphological transducer

=

grammatical analyser

Language with more than a rudimentary morphology need morphological transducers

Such transducers can be written within a year or so (or one might use machine learning)

Semantic resources

- Word sense disambiguation
  - *bank* vs. *bank*, *Roma* (city, Italy's government, football team, ...)
- Semantic classification
  - Some words denote companies, other food, or cars, or ...

  Such resources are already part of commercial products

  When they are included in common search engines people will see the effect of having a search engine that "understands what you mean"

Multilingual language technology

- Minority languages coexist with majority languages ==> *multilingual language technology is crucial*
- The dominating activity with minority language communities is dictionary making, this work should be connected to language technology:
  - dictionaries + morphological analysers
  - parallel corpora for terminology extraction and control
  - syntactic analysers for computer-assisted translation

Text-to-speech

Speech technology is more expensive than lext-based technology, but...

Text-to-speech is not that hard to make:

- One key component is a transducer that translates from text to phonetic representation
- Task: formalise the chapter on prononciation rules in your reference grammar (include all exceptions)
  - The result may be connected to a component for a phonologically similar language
  - ... and we have a cheap text-to-speech component talking with a foreign accent

Some case studies

- Greenlandic
- Sámi
- Yoruba

Greenlandic

- Ongoing project: to build a West Greenlandic spell-checker
  - Wordform list approach:
    - 350000 wordforms recognises 40 % of running text.
  - Morphological approach for Greenlandic:
    - 8 cases, 3 persons, 2 numbers, verbs inflecting for subject and object
    - a derivational component, where a medium-sized set of derivational affixes attach to lexical stems
  - ==> The spell-checker should rather be built like a transducer

Sámi There are 6 Sámi written languages

Lg	Speakers	Alph	extra letters
Southern	500	latin	1
Lule	2000	latin	2
Northern	16600	latin	7
Inari	300	latin	4
Skolt	300	latin	12
Kildin	600	cyrillic	13

Skolt Sámi

Alphabet:

A a, Â â, B b, C c, Č č, ʒ ʒ, ǯ ǯ, D d, Đ , E e, F f, G g, Ǧ ǧ,
G g, H h, I i, J j, K k, Ǩ ǩ, L l, M m, N n, Ŋ ŋ, O o, Õ õ, P p,
[Q q], R r, S s, Š š, T t, U u, V v, [W w], [X x], [Y y], Z z,
Ž ž, Å å, Ä ä, [Ö ö], ´

Kildin Sámi

Alphabet:

А а (Ā ā), Ӓ ӓ (Ӓ ӓ), Б б, В в, Г г, Д д, Е е (Ē ē), Ё ё (Ё̄ ё̄), Ж ж, З з, И и (Ӣ ӣ), Й й, Ӈ ӈ, К к, Л л, Ԯ ԯ, М м, Ӎ ӎ, Н н, Ң ң, Ӊ ӊ, О о (Ō ō), П п, Р р, Ҏ ҏ, С с, Т т, У у (Ӯ ӯ), Ф ф, Х х, Ц ц, Ч ч, Ш ш, Щ щ, Ъ ъ, Ы ы, Ь ь, Ҍ ҍ, Э э (Э̄ э̄), Ӭ ӭ (Э̄̈ э̄̈), Ю ю (Ю̄ ю̄), Я я (Я̄ я̄), Ј ј, Ҍ ҍ, ’

Sámi Localisation – a success story

Keyboard layout is now included, out of the box, no matter where you buy your computer,

in Linux KDE 3.0, Mac OS 10.3, Win XP SP2, due to:

- a decade of hard work, involving experts and language users
- conferences among users in order to arrive at a consensus
- standardisation work in ISO, CEN, national organisations
- lobbying work and explicit pressure from our state administrations upon the OS vendors
- volunteer work within the Linux movement.

Keyboards and graphical user interfaces for many different languages

Out-of-the-box on 3 different platforms

OS	keyboard	GUI
Windows XP	51	33
Mac OS X	42	-
Linux KDE	-	88

12 largest lgs with limited support				12 smallest lgs with basic support or more			
Rank	Speakers	Name	Country	Rank	Speakers	Name	Country
26	41.0	Bhojpuri	India	2108	0.014	Inuktitut	Canada
33	30.0	Siraki	Pakistan	1971	0.017	North. Sámi	Nordic
35	24.0	Maithili	India	1752	0.022	Cherokee	USA
37	23.0	Oriya	India	1344	0.047	Greenlandic	Greenland
39	22.0	Burmese	Myanmar	1343	0.047	Faroese	Denmark
40	22.0	Hausa	Nigeria	1304	0.050	Maori	NZ
44	20.3	Awadhi	India	991	0.940	Gaelic	Scotland
47	20.0	Yoruba	Nigeria	601	0.250	Icelandic	Iceland
51	17.0	Sindhi	Pakistan	517	0.330	Maltese	Malta
53	16.0	Nepali	Nepal	407	0.500	Breton	France
55	15.0	Amharic	Ethiopia	370	0.580	Welsh	UK
59	13.7	Assamese	India	292	0.910	Basque	Spain
60	13.0	Haryanvi	India	130	4.000	Georgian	Georgia

Languages with marginal or no IT support (6400 lgs)

- African languages
- Indian languages other than the official state lgs
- Languages without official status in an independent country, especially in former British and French colonies

Languages with IT support (the remaining 100 lgs)

- Languages with official status in an independent country, and rich and monolingual speakers
- (Most) official state languages of India
- Minority languages with a strong government backing them up (W Europe, Canada, NZ)

Future perspectives for massive multilingual localisation

- Minority language activists turn to the open source movement, where they can localise whatever language they want
- Microsoft and Apple are more restrictive, and only localise when they see a reason for it

The desktop war, as I see it

Microsoft...	Linux...
has a dominant market pos	comes for free
has 3-party software providers	does not crash
has better plug-and-play	is open source
has far better language technology (spell checkers & grammar checkers)	**can be localised to any lg with speakers who care to do the job**

Both parties will probably try to match the competitor

- I expect Microsoft to want to extend both the localisation and the spell checkers to more languages (reacting to critique from the open source community)
  - I expect them to choose a statistically based approach to the spell checkers (Muriel Nolde, Microsoft)

- I expect the Linux community to take the lg tech challenge more seriously
- Problem: lg tech projects cannot be done the Linux way (global volunteer hacking)
  - they need teams of several programmers, lexicographers, philologists and computational and theoretical linguists, to work together for years
- On the other hand: more basic tools become open source
  - Stuttgart *sfst* finite transducer, Odense *vislcg* disambiguator, ...
  - Publically funded dictionary projects now start to make their lexica accessible to open source projects

Yoruba

- 20 million speakers
- Official status
- á, à, é, è, ẹ, ẹ́, ẹ̀, í, ì, ó, ò, ọ, ọ́, ọ̀, ṣ, ú, ù
- No official support on any OS, but Linux work in progress
- Now, there is a discussion on skipping the diacritics

My view: The computer should adjust to humans, and not vice versa

- Orthographies and keyboard layouts should be designed according to linguistic and ergonomic principles
- We linguists invented these diacritic signs – we should help the speakers out

Samuel Olamijulo on the A12n-forum

- Typing ALL Yoruba Letters, undermarks and tonal signs included, is now easy with practice, using free Arial Unicode MS font and ABD Yoruba Keyboard .
- Yoruba Desktop Publishing is making tremendous progress with inputs from many good contributors all over the world.
- One important persisting user headache is in communication accross e-mail , yahoogroups, other web forums, websites and other Internet applications even when Arial Unicode MS or other Unicode Compatible fonts are used in the creation of the message.
- Arial Unicode MS font in MS Word 2003 and ABD Yoruba Keyboard available for free.

What we did for Sámi

Many layouts available We compared them to each other

Letters that had the same positions in all former keyboards kept their positions

The layouts were based on different keyboards We made one keyboard for each country

The users could find symbols like the @, the §, etc. in the positions they would assume them to be placed.

Placement of Sámi letters varied

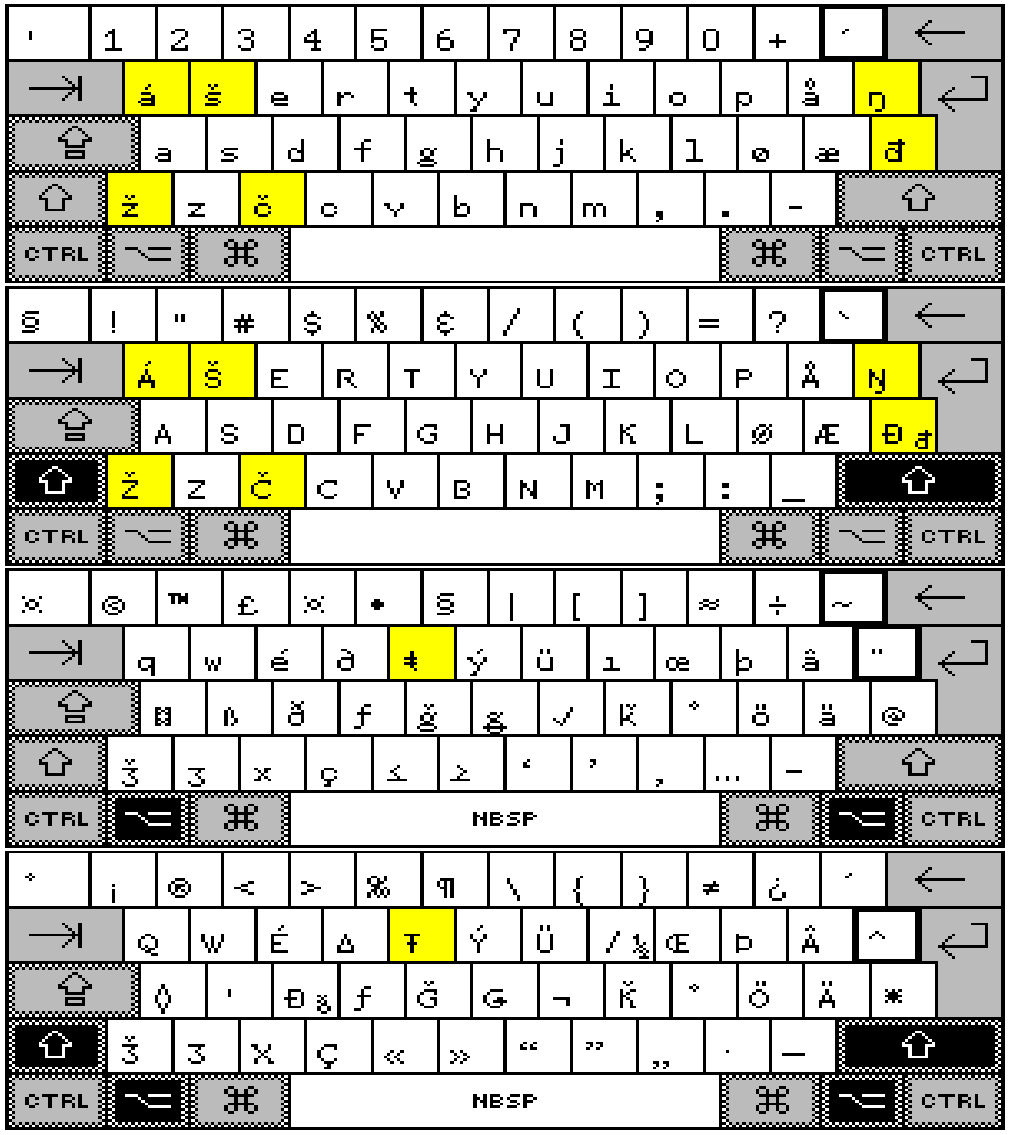
- which keys to use for Sámi letters?
  - Strategy: Keep both maj lg letters and Sámi, and sacrifice non-Nordic *q*, *w*, *x*
- How to place the Sámi letters?
  - According to text frequency
  - The most common letters were given more prominent positions.

- Where to place the replaced letters?
  - As a rule, we put the replaced letters one level up.
  - So, when the key **W** gives *š*, then, in order to get *w*, you press **option-w**, etc..

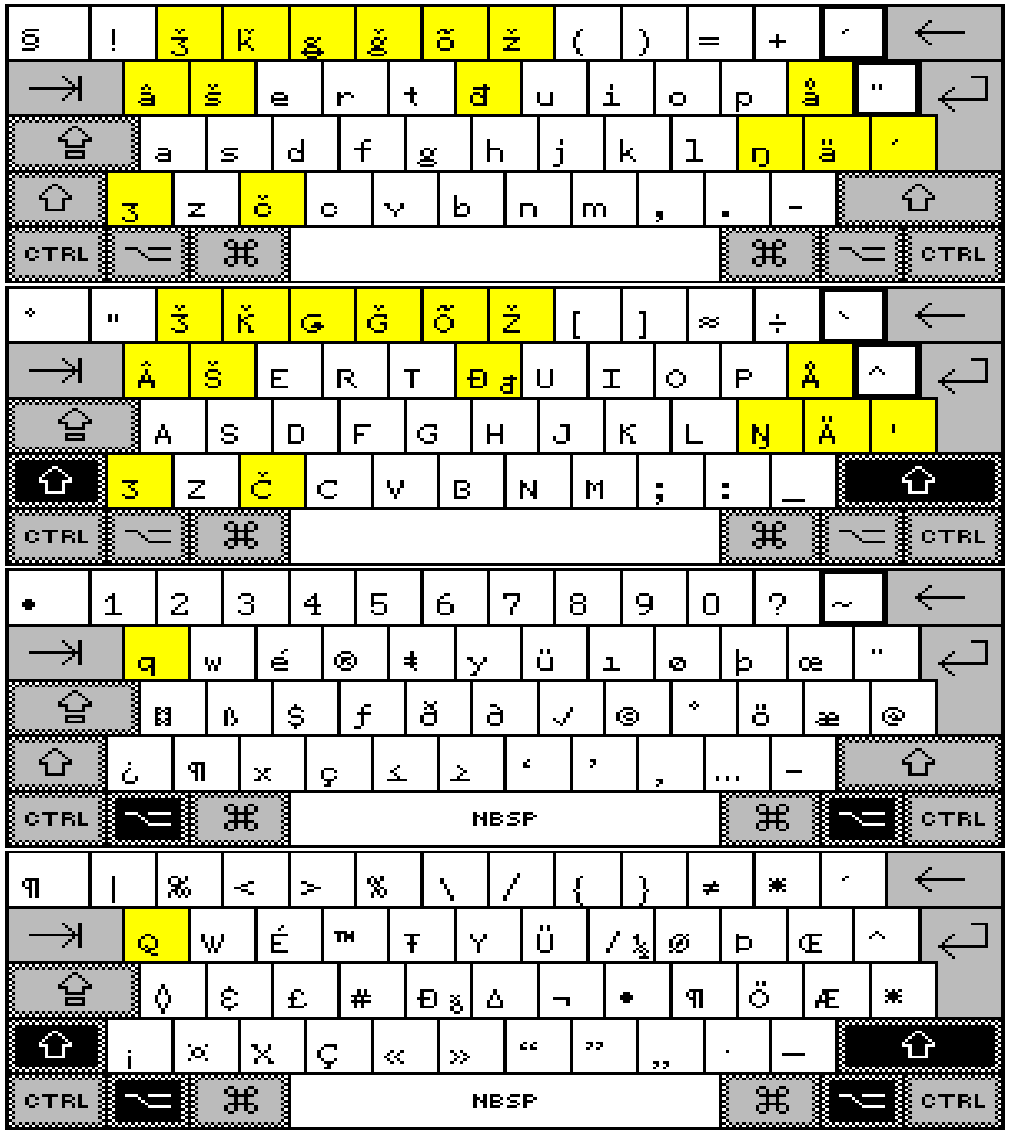
Keyboard layouts should be tested by skilled typists

- For Northern Sámi, we did not carry out systematic testing
  - As a result, low-frequency Sámi letters have a more prominent position on the keyboard than non-Sámi letters frequent in Sámi text

Northern Sámi keyboard for Macintosh, Norway



Skolt Sámi keyboard for Macintosh



Northern Sámi language technology

- Basic tools: Parser and disambiguator
- Spell checker
- Pedagogical programs
- Terminological database
- Encoded mono- and bilingual corpora

The Sámi languages

Eurasian Turkish-type languages (adverbial cases, morphological suffixation, no declension classes), but with

heavy influence from the neighbouring Germanic languages,

non-segmental inflectional processes such as stem-internal diphthong and consonant alternation.

Each lexeme can have several tens of inflected forms, verbs and adjectives have over 100 inflected forms.

The stem modulations make stemming inappropriate as a method for information retrieval.

Parser and disambiguator

Áhčči lea oastán munnje divrras sabehiid 'Father has bought me an expensive pair of skis'

- Morphological analysis
- Disambiguation

```
"<Áhčči>"
        "áhčči" N Sg Nom
"<lea>"
        "leat" V Ind Prs Sg3
"<oastán>"
        "oastit" V PrfPrc
        "oastit" V* N Actor Sg Nom PxSg1
        "oastit" V* N Actor Sg Gen PxSg1
        "oastit" V* N Actor Sg Acc PxSg1
        "oasti" N Sg Nom PxSg1
        "oasti" N Sg Gen PxSg1
        "oasti" N Sg Acc PxSg1
"<munnje>"
        "mun" Pron Pers Sg1 Ill
"<divrras>"
        "divrras" A Attr
        "divrras" A Sg Nom
"<sabehiid>"
        "sabet" N Pl Gen
        "sabet" N Pl Acc
```

```
"<Áhčči>"
        "áhčči" N Sg Nom @SUBJ
"<lea>"
        "leat" V Ind Prs Sg3 @+FAUXV
"<oastán>"
        "oastit" V PrfPrc @-FMAINV
"<munnje>"
        "mun" Pron Pers Sg1 Ill @ADVL
"<divrras>"
         "divrras" A Attr @AN>
"<sabehiid>"
        "sabet" N Pl Acc @OBJ
"<.>"
```

The parser and disambiguator for Sámi have been used for several applications already

Interactively on the web http://giellatekno.uit.no

Spell checker 3-year project for Northern and Lule Sámi (The sámi parliament). – 14 man-years, including work on the Sámi standard

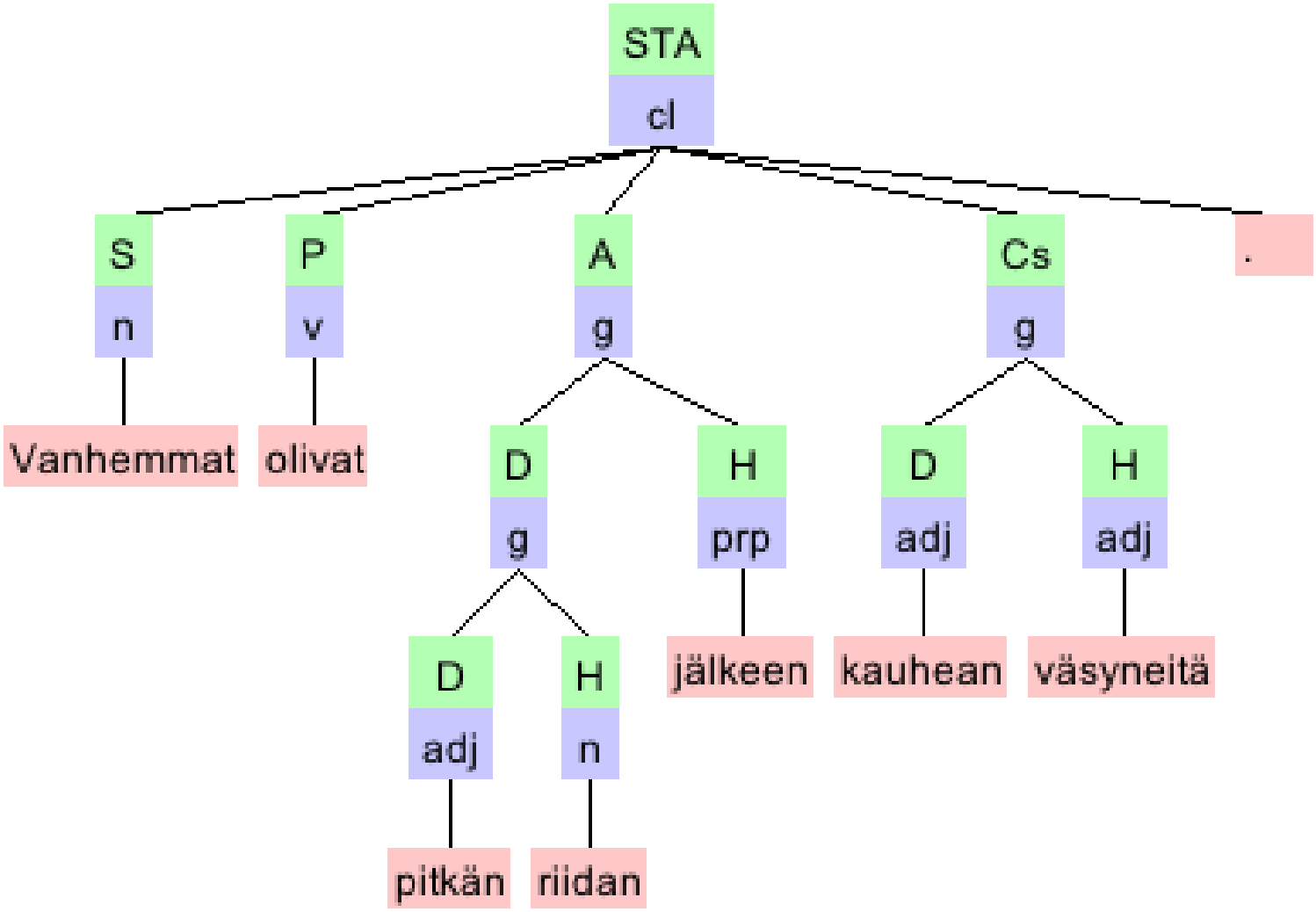
Hyphenator morphologically based hyphenator, that incorporates a morphological analyis in order to find the word boundary, with a set of phonotactic rules in order to find possible hyphen insertion points. In cases of conflict, the word boundary wins.

Pedagogical programs We use the parser to make interactive pedagogical programs (in coop. with the University of Southern Denmark)

Icelandic is part of this project as well, but since there is no Icel. parser available, work on Icelandic is conducted manually

The Northern Sámi sentence translated into underlying ped-format

```
S:n('áhčči',sg,nom) Áhčči
P:g
=D:v('leat',ind,pr,3sg) lea
=H:v('oastit',pcp2) oastán
A:pron('mun',<pers>,1sg,ill) munnje
Od:g
=D:adj('divrras',attr)  divrras
=H:n('sabet',pl,acc) sabehiid
```


STA
cl
S
n
Vanhemmat
P
v
olivat
A
g
D
g
D
adj
pitkän
H
n
riidan
H
prp
jälkeen
Cs
g
D
adj
kauhean
H
adj
väsyneitä
.

Future plans: Portability

Goal: Port solutions for Northern Sámi to other languages

- Large costs go into setting up infrastructure.
- Commercial companies naturally keep this infrastructure to themselves, as this is part of their competitive advantage
- In Tromsø, we will publish our infrastructure as part of an open-source *how-to* for language technology projects.

Lg tech as academic projects? Lg tech for vanishing languages?

- For university lingustics, languages with few speakers are as interesting as languages with many speakers
- Even more so: Languages where you may be a pioneer may be more attractive

But:

- The increasing commercialisation of academia may change this

When languages are about to vanish, we want basic documentation:

- It is not obvious that resources should be geared towards making transducers etc.
- but...
  - lexicographical work should be conducted in a structured way
  - if large corpora are available, they could be annotated by a parser
  - a parser is a handy tool for checking the validity of the rules of the reference grammar

Conclusion: Language technology solutions are

- a *sine qua non* for minority languages needing a written language
- necessary tools for reference work.
- Linguists, programmers and language activists should co-operate on making the necessary tools
- The work costs time, not money, so it can be conducted by any language society
- I am optimistic on behalf of language technology for the 6500 languages of the world
- But I would like to see more of my colleague linguists to join in