

---

GF Resource Grammar Summer School,  
Chalmers University, Gothenburg, Sweden  
August 17-28, 2009

# Status of Icelandic in GF

Martha Dís Brandt  
M.Sc. in Language Technology  
Reykjavík University, Iceland



HÁSKÓLINN Í REYKJAVÍK  
REYKJAVÍK UNIVERSITY

---

# Overview

- ❖ What I've accomplished so far
- ❖ Code examples: the good and the bad
- ❖ Demo... (will hopefully work)
- ❖ Intentions / schedule

# Accomplished

- ❖ Copied an existing language (twice)
- ❖ Got the basic regular nouns to work
- ❖ Began working on verbs
- ❖ Completed the Foods grammar
- ❖ Worked on the miniature grammar, added a gender and 3 cases (incomplete modification)

# Code examples: adjectives

```

adjective : (x1,_,_,_,_,_,_,_,x9 : Str) -> {s : Gender => Number => Defini => Str} =
  \ferskur,fersk,ferskt,ferskir,ferskar,fersk_pl,ferski,ferska,fersku ->
  s = \\g,n,t => case <g,n,t> of {
    < Masc, Sg, Ind > => ferskur ;
    < Masc, Pl, Ind > => ferskir ;
    < Fem, Sg, Ind > => fersk ;
    < Fem, Pl, Ind > => ferskar ;
    < Neutr, Sg, Ind > => ferskt ;
    < Neutr, Pl, Ind > => fersk_pl;
    < Masc, Sg, Def > => ferski ;
    < Fem, Sg, Def > | < Neutr, Sg, Def > => ferska ;
    < _ , Pl, Def > => fersku
  }
  } ;

-- Have yet to add case

```

# Code examples: nouns1

```
mkN_reg_Hestur  : (hestur : Str) -> Gender -> Noun =  
  \hestur -> let hest = Predef.tk 2 hestur in  
    mkN (hest + "ur") hest (hest + "i") (hest + "s")  
      (hest + "ar") (hest + "a") (hest + "um") (hest + "a") ;
```

```
mkN_reg_Grautur : (grautur : Str) -> Gender -> Noun =  
  \grautur -> let graut = Predef.tk 2 grautur in  
    mkN (graut + "ur") graut graut (graut + "ar")  
      (graut + "ar") (graut + "a") (graut + "um") (graut + "a") ;
```

```
mkN_reg_Gestur  : (gestur : Str) -> Gender -> Noun =  
  \gestur -> let gest = Predef.tk 2 gestur in  
    mkN (gest + "ur") gest (gest + "i") (gest + "s")  
      (gest + "ir") (gest + "i") (gest + "um") (gest + "a") ;
```

```
mkN_reg_Fundur  : (fundur : Str) -> Gender -> Noun =  
  \fundur -> let fund = Predef.tk 2 fundur in  
    mkN (fund + "ur") fund (fund + "i") (fund + "ar")  
      (fund + "ir") (fund + "i") (fund + "um") (fund + "a")
```

# Code examples: nouns2

```
-- what about the other mkN_reg_xxxx's ?

mkN = overload {
  mkN : (bord : Str) -> Gender -> N
      = \n -> lin N (mkN_reg_Bord n) ;

  mkN : (x1,_,_,_,_,_,_,x8 : Str) -> Gender -> N
      = \x1,x2,x3,x4,x5,x6,x7,x8,g
      -> lin N (mkNoun x1 x2 x3 x4 x5 x6 x7 x8 g) ;

} ;
```

# Code examples: verbs1

```
param VForm =
  VInf          -- nafnháttur
  | VPres      Number Person -- nútíð
  | VPast      Number Person -- þátíð
  | VImper     Number Person -- boðháttur
  | VPresInd   Number Person -- framsöguháttur
  | VPresSubj  Number Person -- viðtengingarháttur
  | VSup       -- sagnbót
  | VPresPart  -- lýsingarháttur nt
  | VPPart     Gender Number Case -- lýsingarháttur þt
;
```

```
-- Can be 103 forms of a verb..
```

# Code examples: verbs2

```

VInf Active      => ganga ;          -- Infinitive / nafnháttur
VInf Middle     => gangast ;       -- Infinitive / nafnháttur

VImper Sg       => gakktu ;       -- Imperative / boðháttur
VImper Pl       => gangið ;       -- Imperative / boðháttur

VSup Active     => gengið ;       -- Supine / sagnbót
VSup Middle     => gengist ;      -- Supine / sagnbót

VPresPart       => gangandi ;     -- Present participle / lh.nt

VPres Sg P1     => geng ;         -- Present tense / nútíð
VPres Sg P2     => gengur ;
VPres Sg P3     => gengur ;

.....
VPastPart Neutr Sg Dat => gengna ;
VPastPart Neutr Sg Gen => gengna ;
VPastPart _ Pl _      => gengnu   -- Past part (weak decl.) / lh.pt

```



# Future work

- ❖ Main objective is to complete the morphology (by Dec 2009)
  - ❖ Get the mkN overloading to work
  - ❖ Add some Irreg noun forms
  - ❖ Get verbs to work (first present tense, then past tense, other tenses later)
  - ❖ Add pronouns, determiners, numerals, adjectives...

# Thank you...

 Questions?

 Comments?

 Advice???