

Participants:

Bettina

Max

John

Guilia

Stefania

Maria Pia

Example: Italian verbs

An example from

<https://docs.google.com/document/d/1ZH0e3oyQY15qM2bM33k0VxyAj18TGqvsG-luz8HT3r/edit?usp=sharing>

```
## Lexical entry and its "main" form(s)
:lex_accorgersi a ontolex:LexicalEntry ;
  lexinfo:partOfSpeech lexinfo:mainVerb ;
  ontolex:canonicalForm :form_accorgersi .

:form_accorgersi a ontolex:Form ;
  lexinfo:finiteness lexinfo:nonFinite ;
  lexinfo:tense lexinfo:present ;
  lexinfo:verbFormMood lexinfo:infinitive ;
  ontolex:writtenRep "accorgersi"@it .
  morph:inflects :it-refVerb_2_type . # This is new

:it-refVerb_2_type_accorg-stem a morph:StemMorph ;
[...] .

## Paradigm
:it-refVerb a morph:Paradigm ;
  rdfs:comment "some metadata about this paradigm" .

:it-refVerb_2_type a morph:SubParadigm ;
  morph:paradigm :it-refVerb ;
  morph:isSubParadigmOf :it-refVerb_2_type_infin,
:it-refVerb_2_type_indic_1p_sg .

#infinitive rule - output: accorgersi
:it-refVerb_2_type_infin a morph:Rule ;
  morph:subParadigm :it-refVerb_2_type ;
  morph:inflectsFor [ lexinfo:tense lexinfo:present ;
    lexinfo:mood lexinfo:infinitive ] ;
  rdfs:label "-0"@it ;
  morph:prototype "accorgersi" ;
  morph:replacement [ morph:source "$"; morph:target "" ] ;
  morph:generatesMorph [ a morph:StemMorph ] , :it-refVerb_2_type_infin .

#indicative present rule - output: mi accorgo
```

```

:it-refVerb_2_type_indic_1p_sg a morph:Rule ;
  morph:subParadigm :it-refVerb_2_type ;
  morph:inflectsFor [ lexinfo:tense lexinfo:present ;
    lexinfo:mood lexinfo:indicative ] ;
  rdfs:label "mi -o"@it ;
  morph:prototype "mi accorgo"@it ; # optional
  morph:replacement [morph:source "^(.)ersi$" ; morph:target "mi \1o"] .

```

This is before generation. Here's the data after generation:

```

:it-refVerb_2_type_accorgersi-morph a ontolex:StemMorph ;
  morph:generatedBy :it-refVerb_2_type_infin ;
  lexinfo:tense lexinfo:present ;
  lexinfo:mood lexinfo:infinitive .

```

```

:it-refVerb_2_type_infin-form a ontolex:Form ;
  lexinfo:tense lexinfo:present ;
  lexinfo:mood lexinfo:infinitive ;
  ontolex:writtenRep "accorgersi" ;
  morph:generatedBy :it-refVerb_2_type_infin ;
  morph:consistsOf :it-refVerb_2_type_accorgersi-morph .

```

```

:it-refVerb_2_type_indic_1p_sg-form a ontolex:Form ;
  lexinfo:tense lexinfo:present ;
  lexinfo:mood lexinfo:indicative ;
  ontolex:writtenRep "mi accorgo" ;
  morph:generatedBy :it-refVerb_2_type_indic_1p_sg ;
  morph:consistsOf :it-refVerb_2_type_accorg-stem,
:it-refVerb_2_type_indic_1p_sg-morph .

```

```

:it-refVerb_2_type_indic_1p_sg-morph a ontolex:Morph ;
  morph:generatedBy :it-refVerb_2_type_infin ;
  lexinfo:tense lexinfo:present ;
  lexinfo:mood lexinfo:indicative .

```

Questions:

1. Second form (mi accorgo): one rule but how many morphs?
2. CliticMorph?

Italian Inflection grammar example (input data)

####

####Based on G. Speranza, M.P. di Buono #cli_morph_based.nof#

```

#PartMoprh dic
mi, PartMorph1ps
ti, PartMorph2ps
si, PartMorph3ps
ci, PartMorph1pp
vi, PartMorph2pp

```

si, PartMorph3pp

#AffixMorph dic
o, AffixMorphPR1s
i, AffixMorphPR2s
a, AffixMorphPR3s
iamo, AffixMorphPR1p
ate, AffixMorphPR2p
ano, AffixMorphPR3p

#Stem dic
am, StemMorph+FLX=V3
inginocchi, StemMorph+FLX=PRV1

#ARE (e.g., amare)
V3=<StemMorph>are/INF |
#Simple present
<StemMorph><AffixMorphPR1s>/PR+1+s |
<StemMorph><AffixMorphPR2s>/Pr+2+s |
<StemMorph><AffixMorphPR3s>/Pr+3+s |
<StemMorph><AffixMorphPR1p>/Pr+1+p |
<StemMorph><AffixMorphPR2p>/Pr+2+p |
<StemMorph><AffixMorphPR3p>/Pr+3+p;

#PRONOMINAL 1 (e.g., inginocchiarsi)
PRV1=<V3><Bl><Cli>/Inf+Pr |
#Simple present
<LW><PartMorph1ps>" "<V3+Pr+1+s >/Ind+Pres+1+s |
<LW><PartMorph2ps>" "<V3+Pr+2+s >/Ind+Pres+2+s |
<LW><PartMorph3ps>" "<V3+Pr+3+s >/Ind+Pres+3+s |
<LW><PartMorph1pp>" "<V3+Pr+1+p>/Ind+Pres+1+p |
<LW>P<artMorph2pp>" "<V3+Pr+2+p>/Ind+Pres+2+p |
<LW><PartMorph3pp>" "<V3+Pr+3+p>/Ind+Pres+3+p;

#####