**Participants:**

Bettina

Max

Fahad

Stefania

Maria

Julia

# 

# Morph generation definitions (suggestion)

## Classes

### morph:Paradigm

A class that represents a theoretically motivated type of declination, e.g.

* “a” stem declension in Latin
* First declension in Russian

*May* contain metadata information about this type of declination.

**Book analogy**: a full paradigm table with possible allomorphy/alternative variants

### morph:InflectionType

A class that represents a single slot for a single grammatical category for all its possible values (e.g. all the cases)

**Book analogy**: a column from a paradigm table *without* allomorphy/alternative variants for just a single morpheme

### morph:Rule

A class containing necessary information to add **one morpheme** to **a single word form**. It *must* contain either morph:example or morph:replacement (or both). “Tabular” value of a morpheme *must* be stored in rdfs:label (e.g. “-s”@en for usual PL in English)

## Properties

### morph:paradigm

**Domain**: morph:InflectionType

**Range**: morph:Paradigm

A link to the paradigm for the inflection type

### morph:example

**Domain**: morph:Rule

**Range**: string literal

A single generated form that was generated using this rule

### morph:next

**Domain**: morph:InflectionType

**Range**: morph:InflectionType

Links two consecutive inflection types (“slots”), e.g. number and case in Finnish

### morph:inflects

**Domain**: ontolex:Word

**Range**: morph:InflectionType

A link to the first “slot” (inflection type), e.g. an inflection type for number for English nouns

### morph:inflectionType

**Domain**: morph:Rule

**Range**: morph:InflectionType

### morph:replacement

**Domain**: morph:Rule

**Range**: [morph:source, morph:target, both are string literals]

### morph:generates

**Domain**: morph:Rule

**Range**: unrestricted?

# Stefania’s data represented according to this model

:lex\_abbondante a ontolex:LexicalEntry ;

lexinfo:partOfSpeech lexinfo:adjective ;

ontolex:canonicalForm [ ontolex:writtenRep "abbondante"@it ] ;

morph:inflects :it-adj\_002 .

:it-adj a morph:Paradigm ;

rdfs:comment "some metadata about this paradigm" .

:it-adj\_002 a morph:SubParadigm ;

morph:paradigm :it-adj .

:it-adj\_002\_01 a morph:Rule ;

morph:subParadigm :it-adj\_002 ;

morph:generates [ lexinfo:degree lexinfo:positive ;

lexinfo:gender lexinfo:feminine,

lexinfo:masculine ;

lexinfo:number lexinfo:singular ];

rdfs:label ""@it ;

morph:replacement [morph:source "$"; morph:target ""] .

:it-adj\_002\_02 a morph:Rule ;

morph:subParadigm :it-adj\_002 ;

morph:inflectsFor [ lexinfo:degree lexinfo:positive ;

lexinfo:gender lexinfo:feminine,

lexinfo:masculine ;

lexinfo:number lexinfo:plural ];

rdfs:label "-i"@it ;

morph:replacement [morph:source "e$"; morph:target "i"] .

:it-adj\_002\_03 a morph:Rule ;

morph:subParadigm :it-adj\_002 ;

morph:inflectsFor [ lexinfo:degree lexinfo:superlative ;

lexinfo:gender lexinfo:feminine ;

lexinfo:number lexinfo:singular ];

rdfs:label "-issima"@it ;

morph:replacement [morph:source "e$"; morph:target "issima"] .

:it-adj\_002\_04 a morph:Rule ;

morph:subParadigm :it-adj\_002 ;

morph:inflectsFor [ lexinfo:degree lexinfo:superlative ;

lexinfo:gender lexinfo:feminine ;

lexinfo:number lexinfo:plural ];

rdfs:label "-issime"@it ;

morph:replacement [morph:source "e$"; morph:target "issime"] .

:it-adj\_002\_05 a morph:Rule ;

morph:subParadigm :it-adj\_002 ;

morph:inflectsFor [ lexinfo:degree lexinfo:superlative ;

lexinfo:gender lexinfo:masculine ;

lexinfo:number lexinfo:plural ];

rdfs:label "-issimi"@it ;

morph:replacement [morph:source "e$"; morph:target "issimi"] .

:it-adj\_002\_06 a morph:Rule ;

morph:subParadigm :it-adj\_002 ;

morph:inflectsFor [ lexinfo:degree lexinfo:superlative ;

lexinfo:gender lexinfo:masculine ;

lexinfo:number lexinfo:singular ];

rdfs:label "-issimo"@it ;

morph:replacement [morph:source "e$"; morph:target "issimo"] .

MP & Giulia's example - pronominal verbs

### standard ###

: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 .

### paradigma ###

:lex\_accorgersi a ontolex:LexicalEntry ;

lexinfo:partOfSpeech lexinfo:verb ;

ontolex:canonicalForm [ ontolex:writtenRep "accorgersi"@it ] ;

ontolex:morphologicalPattern morph:it-verb\_101 .

:it-verb\_101 a ontolex:morphologicalPattern ;

morph:rule

[ lexinfo:finiteness lexinfo:nonFinite ;

lexinfo:tense lexinfo:present ;

lexinfo:mood lexinfo:infinitive ;

morph:replacement [ morph:source "$" ] ],

[ lexinfo:finiteness lexinfo:nonFinite ;

lexinfo:tense lexinfo:present ;

lexinfo:mood lexinfo:indicative ;

decomp:constituent \*\*\* ,

:clitic\_1-sg ;

morph:example "mi accorgo"@it ; # optional

morph:replacement [ morph:source "ersi$" ; morph:target "o" ; morph:source "^" ; morph:target "mi " ] ]

[ lexinfo:finiteness lexinfo:nonFinite ;

lexinfo:tense lexinfo:present ;

lexinfo:mood lexinfo:gerund ;

decomp:constituent \*\*\* ,

:clitic\_1-sg ;

morph:example "accorgendomi"@it ; # optional

Morph:replacement [ morph:source “rsi$” ; morph:target “ndomi”]]

[...] .

### clitics ###

:clitic\_1-sg a ontolex:Clitic ;

lexinfo:case lexinfo:accusative ;

lexinfo:number lexinfo:singular ;

lexinfo:person lexinfo:firstPerson

ontolex:writtenRep "mi"@it.

[...] .

**Separable verbs *(Trennbare Verben)***

:lex\_anfangen a ontolex:LexicalEntry ;

lexinfo:partOfSpeech lexinfo:verb ;

ontolex:canonicalForm [ ontolex:writtenRep "anfangen"@de ] ;

ontolex:morphologicalPattern morph:de-verb\_88 .

:de-verb\_88 a ontolex:morphologicalPattern ;

morph:rule

[ lexinfo:finiteness lexinfo:nonFinite ;

lexinfo:tense lexinfo:present ;

lexinfo:mood lexinfo:infinitive ;

morph:replacement [ morph:source "$" ] ],

[ lexinfo:finiteness lexinfo:nonFinite ;

lexinfo:tense lexinfo:present ;

lexinfo:mood lexinfo:indicative ;

decomp:constituent \*\*\* ,

:trenn\_verb\_1-sg ;

morph:example "fange an"@de ; # optional

morph:replacement [ morph:source "en$" ; morph:target "e" ; morph:source "^an" ; morph:target "" ] ]

Morph:displacement [ morph:**source** “$” ; morph:target “ an” ]]

Summary of issues to be discussed next time:

* Representing the generation of analytic wordforms (Bettina’s proposal: interconnect the morph module with MMoOn Core)
* Naming of “morph:generates” (Bettina’s proposal morph:inflectsFor)
* Revise naming of “InflectionType”

# Morph generation definitions (refined)

## Classes

### morph:Paradigm

A class that represents a theoretically motivated type of declination, e.g.:

* “a” stem declension in Latin
* First declension in Russian

*May* contain metadata information about this type of declination.

**Book analogy**: a full paradigm table with possible allomorphy/alternative variants

### morph:SubParadigm

A class that represents a single slot for a single grammatical category for all its possible values (e.g. all the cases). There’s no possible allomorphy on this level (even orthographic variants).

**Book analogy**: a column from a paradigm table without allomorphy/alternative variants for just a single morpheme

### morph:Rule

A class containing necessary information to add one morpheme to a single word form. It must contain either morph:example or morph:replacement (or both). “Tabular” value of a morpheme must be stored in rdfs:label (e.g. “-s”@en for usual PL in English)

### morph:Replacement

A class that contains a source and a target for replacement.

## Properties

### morph:paradigm

**Domain**: morph:InflectionType

**Range**: morph:Paradigm

A link to the paradigm for the inflection type

### morph:prototype

**Domain**: morph:Rule

**Range**: string literal

A single generated form that was generated using this rule

### morph:next

**Domain**: morph:SubParadigm

**Range**: morph:SubParadigm

Links two consecutive inflection types (“slots”), e.g. number and case in Finnish

### morph:inflects

**Domain**: ontolex:Word

**Range**: morph:SubParadigm

A link to the first “slot” (inflection type), e.g. an inflection type for number for English nouns

### morph:subParadigm

**Domain**: morph:Rule

**Range**: morph:InflectionType

### morph:replacement

**Domain**: morph:Rule

**Range**: morph:Replacement

### morph:inflectsFor

**Domain**: morph:Rule

**Range**: unrestricted?