Morph telco 2022-05-17, 13:15 CEST

**Link:** [**https://meet.google.com/nsj-tbcy-yop**](https://meet.google.com/nsj-tbcy-yop) **[CHECK HERE FOR UPDATED LINK(S)]**

**Latest Definitions:** [**https://github.com/ontolex/morph/blob/master/draft.md**](https://github.com/ontolex/morph/blob/master/draft.md)

**Nexus:** [**https://nexuslinguarum.eu/the-action/join-us**](https://nexuslinguarum.eu/the-action/join-us)

**Participants [please add yourself]:**

Christian Chiarcos (CC) (excused for being 10 min late)

Max Ionov (MI)

Katerina Gkirtzou (KG)

Besim Kabashi (BK)

Fahad Khan (FK)

Khadija Ait ElFqih (KAE)

Matteo Pellegrini (MP)

Ciprian-Octavian Truică (CT)

Penny Labropoulou (PL)

Elena Simona Apostol (ESA)

Sina Ahmadi (SA)

Elena Benzoni (EB)

Petra Steiner (PS)

Theodorus Fransen (TF)

Ranka Stanković (RS)

Gilles Sérasset (GS)

Mike Rosner (MR)

Table of Contents

[0. Module draft (4.17)](#_v4addnpz2vtu)

[1. Datathon results](#_3b2yols64ffs)

[2. Clitics](#_av83soukzc3q)

[3. Character/sound classes](#_3c1slk2we2i3)

[Maltese example](#_bgy7leujpwjo)

[Defining classes](#_ttwrnvbsp5y3)

[Other examples](#_1f6l1qle86kx)

[4. W3C day presentation](#_845uu386qlpd)

[Next time](#_nk5j7w1igusi)

# 0. Module draft (4.17)



# 1. Datathon results

Two projects that successfully applied Morph for modelling derivation:

1. PIE etymological root database (+ Old Irish, but no Morph was needed there)
2. Aspectual database for Serbian, Croatian and Bosnian

Seemed to work really well, no issues there

**TODO@MI, CC, KG**: Add links to project results

# 2. Clitics

See [minutes from the last call](https://docs.google.com/document/d/1x71lXb6EFaby1v7rF3_Tda_4b-Z6atraN9VjybIXHAo/edit#heading=h.av83soukzc3q), seemed to be a consensus on modelling them as wordforms with a space.

Still some todos:

**TODO@MI**: add a pull request with an example of this to the documentation

**TODO**: check what people think about cliticization

# 3. Character/sound classes

Brief reminder: MI and MR argue for adding an element for representing a group of characters (or sounds) to use in replacement rules. This can be helpful for many languages in a lot of situations, making rules reusable and understandable.

## Maltese example

*kiteb* → *ktibt* (PERF.1SG)

With character/sound classes:

<kiteb\_perf\_1sg> a morph:InflectionRule ;

morph:paradigm <kiteb\_paradigm> ;

morph:involves <suffix\_t\_perf\_1sg> ;

morph:replacement [

a morph:Replacement ;

morph:source "(C)(V)(C)(V)(C)" ;

morph:target "\1\3i\5t" ;

morph:replacementClass [ rdfs:label "V" ;

rdfs:value "[aeiou]", "e", "i", "o", "u" . ],

[]

] .

Modification Christian:

<kiteb\_perf\_1sg> a morph:InflectionRule ;

morph:paradigm <kiteb\_paradigm> ;

morph:involves <suffix\_t\_perf\_1sg> ;

morph:replacement [

a morph:Replacement ;

morph:source "(C)(V)(C)(V)(C)" ;

morph:target "\1\3i\5t" ;

morph:replacementTable :r1 ] .

:r1 a morph:ReplacementTable ;

morph:replacements “{ ‘V’ : ‘[aeiou]’, ‘C’: ‘[bcdghjklmnpqrstvwxz]’ }“ .

Without character/sound classes:

<kiteb\_perf\_1sg> a morph:InflectionRule ;

morph:paradigm <kiteb\_paradigm> ;

morph:involves <suffix\_t\_perf\_1sg> ;

morph:replacement [

a morph:Replacement ;

morph:source "(ċ|d|n|r|s|t|x|ż|z|b|f|ġ|g|għ|h|ħ|j|k|l|m|p|q|v|w)(a|e|i|o|u|ie)(ċ|d|n|r|s|t|x|ż|z|b|f|ġ|g|għ|h|ħ|j|k|l|m|p|q|v|w)(a|e|i|o|u|ie)(ċ|d|n|r|s|t|x|ż|z|b|f|ġ|g|għ|h|ħ|j|k|l|m|p|q|v|w)" ;

morph:target "\1\3i\5t" ;

] .

Technically similar albeit bulky, visually much less clear — difficult to interpret or to see differences between different rules.

## Defining classes

**Model change**: adding one class

<vowels> a <CharacterClass> ; # this is not a part of the model, but a part of the dataset

rdfs:label "V" ;

rdfs:member "a", "e", "i", "o", "u" .

For form generation: adding a replace for each class.

If decided not to use this in the model this will probably be used ad-hoc → rules in the datasets will be **not interoperable**.

## Other examples

* Vowel harmony rules in Turkic and Finno-Ugric languages (e.g. Turkish, Finnish)
  + Vowels in the replacement depend on vowels in the root
  + An affix can be -lla or -llä for a Finnish case

morph:replacement [

a morph:Replacement ;

morph:source "(.\*FRONT\_V.\*)$" ;

morph:target "\1llä" ;

],

a morph:Replacement ;

morph:source “(.\*BACK\_V.\*)$” ;

morph:target “\1lla” ;

] . # Finnish Adessive case

* Rules like “If the stem of the noun ends in a vowel, the buffer consonant *y* is added”

morph:replacement [

a morph:Replacement ;

morph:source "(V)$" ;

morph:target "\1ya" ;

],

a morph:Replacement ;

morph:source “(C)$” ;

morph:target “\1a” ;

] . # Turkish Accusative and Dative

* Umlaut mutations in German (provided by CC, motivated by historical process description)

rule:umlaut a morph:WordFormationRule;

morph:replacement

[ a morph:Replacement;

morph:source "a([^aeiouöü ]\*)$";

morph:target "ä\1" ];

[ a morph:Replacement;

morph:source "o([^aeiouöü ]\*)$";

morph:target "ö\1" ];

[ a morph:Replacement;

morph:source "u([^aeiouöü ]\*)$";

morph:target "ü\1" ] .

vs.

rule:umlaut a morph:WordFormationRule;

morph:replacement

[ a morph:Replacement;

morph:source "a(C\*)$";

morph:target "ä\1" ];

[ a morph:Replacement;

morph:source "o(C\*)$";

morph:target "ö\1" ];

[ a morph:Replacement;

morph:source "u(C\*)$";

morph:target "ü\1" ] .

**Note**: in all these cases we will apply two alternative regexes in the same rule

## Pros and cons of extending the model:

Pro: rules are more readable but at the same time interoperable

Pro: easier conversion from resources providing rules

Con: these properties are not connected to any other elements in the model

Con: there is no *morphological* meaning behind this class (but it won’t be the first time)

# 4. W3C day presentation

* <https://www.w3.org/community/ontolex/wiki/W3c_community_day_@_LDK2023>
* Last time we presented the status of the model back then. Should we present the way it is again, focusing on the latest additions and changes?
* ~40 minutes, but how long should be the discussion? Last time there was not that many questions
* Usually people who care are in our calls
* But: this time we can also mention that the model works for Semitic languages

FK: Be clear about the state of the model: are we happy, if there still something we are unhappy with

* Also, choose if we are going to say if we still have some time to implement new things or just say if it is what it is

CC: We can’t come up with a list of requirements right now because we still have open questions

* **TODO@MI**: go through all the requirements and give an overview of what is left
* **TODO@CC, MI**: discuss what is left

Chat

# Next time

* Probably the last call before September.
* Going through the [requirements](https://docs.google.com/document/d/1s6uY-74Z9dPMau96AmPEHH_ivRvvWTK5-HZKIPTKyoA/edit#heading=h.b77jzkoaql9r)?
* Looking at one last topic / dataset before going on vacation? Anything urgent?
* Paradigm discussion?
* Finite state terminology?
* Open questions from LDL?