Morph telco 2023-10-18, 13:00 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) [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. Quick vote: Replacement → RegexReplacement](#_qlc64wlfho9e)

[2. MP: Another standard for morphology representation](#_wqqmxofrbqs7)

[3. CC: Inuktikut data and problems with LexicalEntries](#_8pfoi21gwxpp)

# 0. Module draft (4.17)



# 1. MP presents Paralex: Another standard for morphology representation

MP presents the new standard for morphology, its compatibility with Morph.

<https://www.paralex-standard.org/standard/>

Based on Unimorph with the hope to overcome Unimorph limitations

Complementation with Morph:

* This is much more specific than Morph
* The idea is to be able to automatically convert this to Morph
* Paralex makes it interoperable with UD-family resources

# 2. Quick vote: Replacement → RegexReplacement

**Suggestion** — having morph:Replacement underdefined, a new subclass morph:RegexReplacement which has the properties morph:Replacement has now.

Conclusion: removing morph:source and morph:target. Moving RegexReplacement as a subclass to a companion vocabulary

NOTE: in the guidelines, we can recommend using rdf:value for string representations of replacements, eg as in Perl or Sed, s/SOURCE/TARGET

# 3. CC: Inuktikut data and problems with LexicalEntries

<https://github.com/acoli-repo/morph-addenda/tree/master/data/polysynthetic>

The problem: ambiguity in derivation — <https://github.com/acoli-repo/morph-addenda/tree/master/data/polysynthetic#5-encode-ambiguity-in-derivation>

CC will explain next time (01.11.2023)