GraphLang: DMRS graph description language

Alexander Kuhnle
University of Cambridge (United Kingdom)
aok25@cam.ac.uk

Version: 11th July 2016

General
- Lower-case (if not mentioned otherwise)
- Variable-length spaces
- #/### stands for an arbitrary value or special value
- % stands for a special value (see below)

DMRS graphs
- [Line] → [Node] (ContLine)
- [ContLine] → [Link] [Node] (ContLine)
- [ContLine] → ; (Line)
- [ContLine] → \n (Line) (\n: new line)

Nodes
- [Node] → (Id:)node(sort)% (underspecified node)
- [Node] → (Id:)[Pred] (Sortinfo)
- [Node] → (:Id:) (predicate name is default id)
- [Pred] → (") [Predicate] ("r[rel](Carg)
- [Id]: Arbitrary name, potentially with node type marker:
  [###] anchor, (###) optional, {###} subgraph

Predicates
- pred(sort)%: Underspecified predicate
- #.#.#: RealPred
- #.#: RealPred with unspecified sense
- #.#.#: RealPred with underspecified POS
- #.#.unknown: RealPred with underspecified sense
- #.: POS RealPred with underspecified lemma and sense
- #: GPred

Carg
- In brackets, (###) or ("###"), lower-case not required

Sortinfos
- i: Fully underspecified sortinfo
- e, x: Unspecified event-instance sortinfo
- e?, x?: Underspecified event-instance sortinfo
- e[...], x[...]: Specified event-instance sortinfo
- [###]: Full shortform value specification (length exactly 5, _ for unspecified, ? for underspecified)
- [###,...]: Partial shortform key-value specification
- [######,...]: Partial key-value specification
- _: Underspecified value

Event values (shortform in square brackets)
- Sentence force ([s]f): [prop], [q]ues, prop-[o]r-ques, [c]omm
- Mood ([m]ood): [i]ndicative, [a]ubjunctive
- Perfect tense ([p]erf): +, -
- Progressive form (p[rogress]): +, -, [b]ool

Instance values (shortform in square brackets)
- Person ([p]ers): 1, 2, 3, 1-[o]r-3
- Number ([n]um): [a]l, [p]l
- Gender ([g]end): f, m, n, m-[o]f-f
- Individuated ([i]nd): +, -
- Pronoun type (p[t]): [a]ld, [r]efl, [z]ero

Links

Link head Either starts with <... or ends with ...>
Link line Either - or = on both sides of the label

Implicit label (if length ≤ 2)
- Argument structure (with 'x' either 1-4 or 'n' for underspecified ARG)
  - -x->: ARGx/NEQ
  - +x->: ARGx/EQ
  - +x-: ARGx/H
  - =x-: ARGx/HEQ
- Quantifier:
  - -->: RSTR/H
- Coordination (with 'x' either 'l' or 'r'):
  - -x->: x-INDEX/NEQ
  - +x->: x-INDEX/EQ
  - -x-: x-HNDL/H
  - +x-: x-HNDL/HEQ
- EQ link:
  - ==>: None/EQ
- Underspecified links (with 'x' either one character or empty, either - or =):
  - -?=>: All

Explicit label (if length ≥ 3, no -*/- distinction)
- -#->: Only rargname
- -#/->: Rargname and post

Special values
- Syntax: Symbol followed by an optional identifier
- Underspecification: ?
- Query: ? and identifier
- Equality constraint: =

Example

"Every big angry dog barks loudly."

_every_q -- subj:dog,1 x[3a_+] <=1_ bark_v,1 e[ppi_];
_bark_v,1 <=1_ loud_a,1 e[ppi_];
_big_a,1 e[ppi_] =>1_ subj <=1_ angry_a_at e[ppi_]