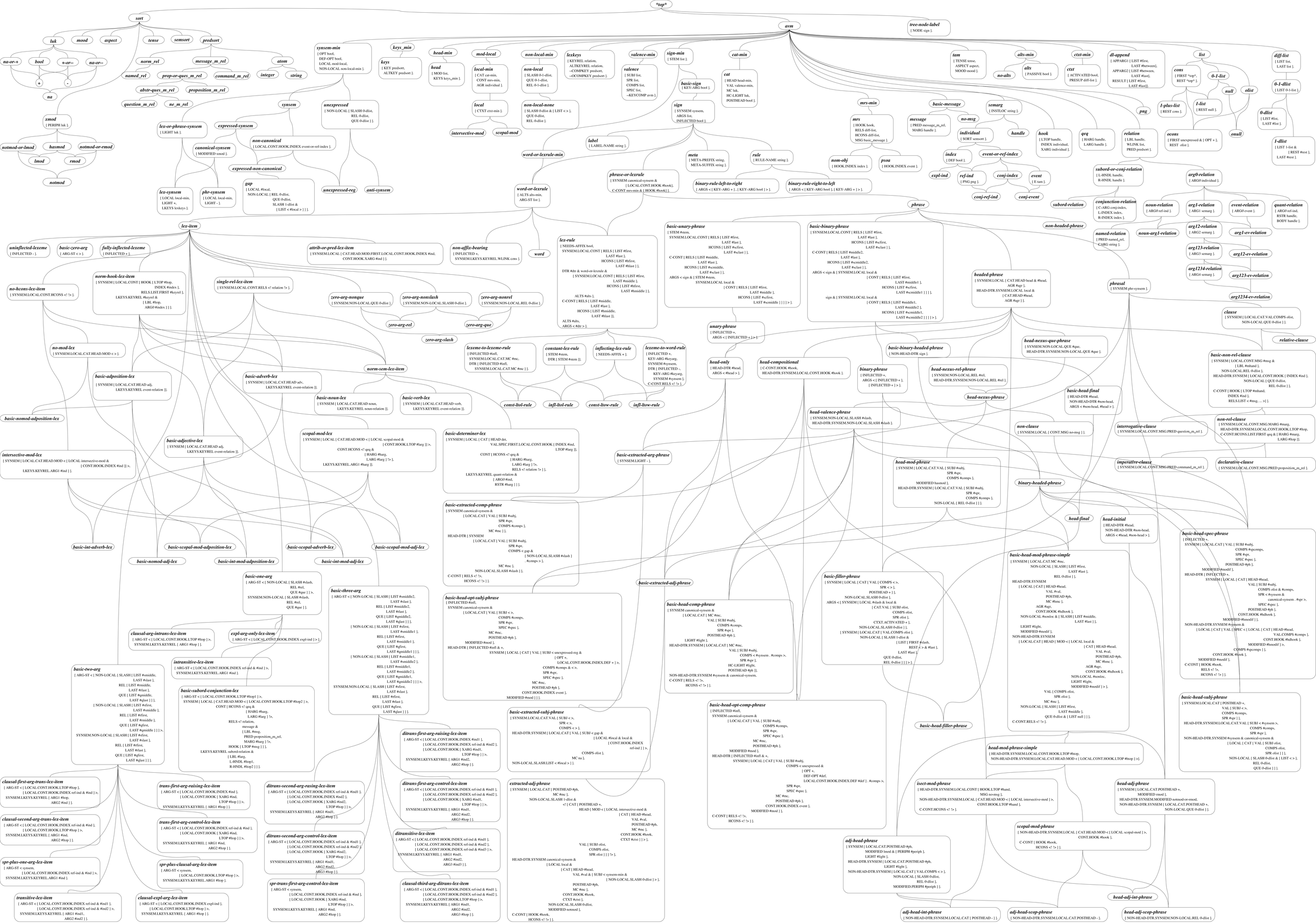
top
sort
avm
tree-node-label
[NODE sign].
luk
mood
aspect
tense
semsort
predsort
na-or-+
bool
+-or--
na-or--
+
-
na
xmod
[PERIPH luk].
notmod-or-lmod
hasmod
notmod-or-rmod
lmod
rmod
notmod
norm_rel
message_m_rel
atom
integer
string
named_rel
prop-or-ques_m_rel
command_m_rel
abstr-ques_m_rel
proposition_m_rel
question_m_rel
ne_m_rel
synsem-min
[OPT bool,
DEF-OPT bool,
LOCAL mod/cat,
NON-LOCAL non-local-min].
keys_min
keys
[KEY predsort,
ALTKEY predsort].
head-min
head
[MOD list,
KEYS keys_min].
mod-local
local-min
[CAT cat-min,
CONT mrs-min,
AGR individual].
local
[CTXT ctxt-min].
intersective-mod
scopal-mod
non-local-min
non-local
[SLASH 0-1-dlist,
QUE 0-1-dlist,
REL 0-1-dlist].
non-local-none
[SLASH 0-dlist & [LIST < >],
QUE 0-dlist,
REL 0-dlist].
lexkeys
[KEYREL relation,
ALTKEYREL relation,
--COMPKEY predsort,
--OCOMPKEY predsort].
valence-min
valence
[SUBJ list,
SPR list,
COMPS list,
SPEC list,
--KEYCOMP avm].
sign-min
[STEM list].
basic-sign
[KEY-ARG bool].
sign
[SYNSEM synsem,
ARGS list,
INFLECTED bool].
cat-min
cat
[HEAD head-min,
VAL valence-min,
MC luk,
HC-LIGHT luk,
POSTHEAD bool].
tam
[TENSE tense,
ASPECT aspect,
MOOD mood].
alts-min
alts
[PASSIVE bool].
no-alts
ctxt-min
ctxt
[ACTIVATED bool,
PRESUP diff-list].
dl-append
[APPARG1 [LIST #first,
LAST #between],
APPARG2 [LIST #between,
LAST #last],
RESULT [LIST #first,
LAST #last]]
list
cons
[FIRST *top*,
REST *top*].
0-1-list
null
olist
1-plus-list
[REST cons].
1-list
[REST null].
ocons
[FIRST unexpressed & [OPT +],
REST olist].
onull
diff-list
[LIST list,
LAST list].
0-1-dlist
[LIST 0-1-list].
0-dlist
[LIST #list,
LAST #list].
1-dlist
[LIST 1-list &
[REST #rest],
LAST #rest].
mrs-min
mrs
[HOOK hook,
RELS diff-list,
HCONS diff-list,
MSG basic_message].
nom-obj
[HOOK.INDEX index].
psoa
[HOOK.INDEX event].
basic-message
message
[PRED message_m_rel,
MARG handle].
no-msg
semarg
[INSTLOC string].
individual
[SORT semsort].
handle
index
[DEF bool].
event-or-ref-index
expl-ind
ref-ind
[PNG png].
conj-index
event
[E tam].
conj-ref-ind
conj-event
hook
[LTOP handle,
INDEX individual,
XARG individual].
qeq
[HARG handle,
LARG handle].
png
relation
[LBL handle,
WLINK list,
PRED predsort].
subord-or-conj-relation
[L-HNDL handle,
R-HNDL handle].
arg0-relation
[ARG0 individual].
subord-relation
conjunction-relation
[C-ARG conj-index,
L-INDEX index,
R-INDEX index].
noun-relation
[ARG0 ref-ind].
arg1-relation
[ARG1 semarg].
event-relation
[ARG0 event].
quant-relation
[ARG0 ref-ind,
RSTR handle,
BODY handle].
noun-arg1-relation
arg12-relation
[ARG2 semarg].
arg1-ev-relation
arg123-relation
[ARG3 semarg].
arg12-ev-relation
arg1234-relation
[ARG4 semarg].
arg123-ev-relation
arg1234-ev-relation
lex-or-phrase-synsem
[LIGHT luk].
expressed-synsem
synsem
unexpressed
[NON-LOCAL [SLASH 0-dlist,
REL 0-dlist,
QUE 0-dlist]].
canonical-synsem
[MODIFIED xmod].
non-canonical
[LOCAL.CONT.HOOK.INDEX event-se-ref-index].
expressed-non-canonical
gap
[LOCAL #local,
NON-LOCAL [REL 0-dlist,
QUE 0-dlist,
SLASH 1-dlist &
[LIST < #local >]]].
lex-synsem
[LOCAL local-min,
LIGHT +,
LKEYS lexkeys].
phr-synsem
[LOCAL local-min,
LIGHT -].
unexpressed-reg
anti-synsem
label
[LABEL-NAME string].
meta
[META-PREFIX string,
META-SUFFIX string].
rule
[RULE-NAME string].
binary-rule-left-to-right
[ARGS < [KEY-ARG +] , [KEY-ARG bool] >].
binary-rule-right-to-left
[ARGS < [KEY-ARG bool] , [KEY-ARG +] >].
word-or-lexrule-min
phrase-or-lexrule
[SYNSEM canonical-synsem &
[LOCAL.CONT.HOOK #hook],
C-CONT mrs-min & [HOOK #hook]].
word-or-lexrule
[ALTS alts-min,
ARG-ST list].
lex-item
uninflected-lexeme
[INFLECTED -].
basic-zero-arg
[ARG-ST < >].
fully-inflected-lexeme
[INFLECTED +].
attrib-or-pred-lex-item
[SYNSEM.LOCAL [CAT.HEAD.MOD.FIRST.LOCAL.CONT.HOOK.INDEX #ind,
CONT.HOOK.XARG #ind]].
non-affix-bearing
[INFLECTED +,
SYNSEM.LKEYS.KEYREL.WLINK cons].
word
lex-rule
[NEEDS-AFFIX bool,
SYNSEM.LOCAL.CONT [RELS [LIST #first,
LAST #last],
HCONS [LIST #hfirst,
LAST #hlast],
DTR #dtr & word-or-lexrule &
[SYNSEM.LOCAL.CONT [RELS [LIST #first,
LAST #middle],
HCONS [LIST #hfirst,
LAST #hmiddle]],
ALTS #alts],
C-CONT [RELS [LIST #middle,
LAST #last],
HCONS [LIST #hmiddle,
LAST #hlast]],
ALTS #alts],
ARGS < #dtr >].
no-hcons-lex-item
[SYNSEM.LOCAL.CONT.HCONS <! !>].
norm-hook-lex-item
[SYNSEM [LOCAL.CONT [HOOK [LTOP #top,
INDEX #index],
RELS.LIST.FIRST #keyrel],
LKEYS.KEYREL #keyrel &
[LBL #top,
ARG0 #index]]].
single-rel-lex-item
[SYNSEM.LOCAL.CONT.RELS <! relation !>].
zero-arg-nonque
[SYNSEM.NON-LOCAL.QUE 0-dlist].
zero-arg-nonslash
[SYNSEM.NON-LOCAL.SLASH 0-dlist].
zero-arg-nonrel
[SYNSEM.NON-LOCAL.REL 0-dlist].
zero-arg-rel
zero-arg-que
zero-arg-slash
no-mod-lex
[SYNSEM.LOCAL.CAT.HEAD.MOD < >].
basic-adposition-lex
[SYNSEM [LOCAL.CAT.HEAD adj,
LKEYS.KEYREL event-relation]].
basic-adverb-lex
[SYNSEM [LOCAL.CAT.HEAD adv,
LKEYS.KEYREL event-relation]].
basic-noun-lex
[SYNSEM [LOCAL.CAT.HEAD noun,
LKEYS.KEYREL noun-relation]].
basic-verb-lex
[SYNSEM [LOCAL.CAT.HEAD verb,
LKEYS.KEYREL event-relation]].
norm-sem-lex-item
lexeme-to-lexeme-rule
[INFLECTED #infl,
SYNSEM.LOCAL.CAT.MC #mc,
DTR [INFLECTED #infl,
SYNSEM.LOCAL.CAT.MC #mc]].
constant-lex-rule
[STEM #stem,
DTR [STEM #stem]].
inflecting-lex-rule
[NEEDS-AFFIX +].
lexeme-to-word-rule
[INFLECTED +,
KEY-ARG #keyarg,
SYNSEM #synsem,
DTR [INFLECTED -,
KEY-ARG #keyarg,
SYNSEM #synsem],
C-CONT.RELS <! !>].
const-itol-rule
infl-itol-rule
const-itow-rule
infl-itow-rule
basic-nomod-adposition-lex
basic-adjective-lex
[SYNSEM [LOCAL.CAT.HEAD adj,
LKEYS.KEYREL event-relation]].
scopal-mod-lex
[SYNSEM [LOCAL [CAT.HEAD.MOD < [LOCAL scopal-mod &
[CONT.HOOK.LTOP #larg]] >,
CONT.HCONS <! qeq &
[HARG #larg,
LARG #larg] !>],
LKEYS.KEYREL.ARG1 #larg]].
basic-determiner-lex
[SYNSEM [LOCAL [CAT [HEAD det,
VAL.SPEC.FIRST.LOCAL.CONT.HOOK [INDEX #ind,
LTOP #larg]],
CONT [HCONS <! qeq &
[HARG #larg,
LARG #larg] !>],
LKEYS.KEYREL quant-relation &
[ARG0 #ind,
RSTR #larg]]].
intersective-mod-lex
[SYNSEM [LOCAL.CAT.HEAD.MOD < [LOCAL intersective-mod &
[CONT.HOOK.INDEX #ind]] >,
LKEYS.KEYREL.ARG1 #ind]].
basic-int-adverb-lex
basic-scopal-mod-adposition-lex
basic-scopal-adverb-lex
basic-scopal-mod-adj-lex
basic-nomod-adj-lex
basic-int-mod-adposition-lex
basic-int-mod-adj-lex
basic-one-arg
[ARG-ST < [NON-LOCAL [SLASH #slash,
REL #rel,
QUE #que]],
SYNSEM.NON-LOCAL [SLASH #slash,
REL #rel,
QUE #que]].
basic-three-arg
[ARG-ST < [NON-LOCAL [SLASH [LIST #middle2,
LAST #last],
REL [LIST #middle2,
LAST #first],
QUE [LIST #qmiddle2,
LAST #qlast]],
NON-LOCAL [SLASH [LIST #sfirst,
LAST #smiddle1],
REL [LIST #rfirst,
LAST #rmiddle1],
QUE [LIST #qfirst,
LAST #qmiddle1]]],
[NON-LOCAL [SLASH [LIST #sfirst,
LAST #smiddle1],
REL [LIST #rfirst,
LAST #rmiddle1],
QUE [LIST #qfirst,
LAST #qmiddle1]]],
SYNSEM.NON-LOCAL [SLASH [LIST #sfirst,
LAST #last],
REL [LIST #rfirst,
LAST #last],
QUE [LIST #qfirst,
LAST #qlast]]].
clausal-arg-intrans-lex-item
[ARG-ST < [LOCAL.CONT.HOOK.LTOP #top] >,
SYNSEM.LKEYS.KEYREL [ARG1 #top]].
expl-arg-only-lex-item
[ARG-ST < [LOCAL.CONT.HOOK.INDEX expl-ind] >].
basic-two-arg
[ARG-ST < [NON-LOCAL [SLASH [LIST #smiddle,
LAST #slast],
REL [LIST #rmiddle,
LAST #rlast],
QUE [LIST #qmiddle,
LAST #qlast]],
[NON-LOCAL [SLASH [LIST #sfirst,
LAST #smiddle],
REL [LIST #rfirst,
LAST #rmiddle],
QUE [LIST #qfirst,
LAST #qmiddle]]],
SYNSEM.NON-LOCAL [SLASH [LIST #sfirst,
LAST #slast],
REL [LIST #rfirst,
LAST #rlast],
QUE [LIST #qfirst,
LAST #qlast]]].
intransitive-lex-item
[ARG-ST < [LOCAL.CONT.HOOK.INDEX ref-ind & #ind] >,
SYNSEM.LKEYS.KEYREL.ARG1 #ind].
basic-subord-conjunction-lex
[ARG-ST < [LOCAL.CONT.HOOK.LTOP #top1] >,
SYNSEM [LOCAL [CAT.HEAD.MOD < [LOCAL.CONT.HOOK.LTOP #top2] >,
CONT [HCONS <! qeq &
[HARG #larg,
LARG #larg] !>,
RELS <! relation,
message &
[LBL #msg,
PRED proposition_m_rel,
MARG #larg] !>,
HOOK [LTOP #msg]]],
LKEYS.KEYREL subord-relation &
[LBL #larg,
L-HNDL #top1,
R-HNDL #top2]]].
clausal-first-arg-trans-lex-item
[ARG-ST < [LOCAL.CONT.HOOK.LTOP #top],
[LOCAL.CONT.HOOK.INDEX ref-ind & #ind] >,
SYNSEM.LKEYS.KEYREL [ARG1 #top,
ARG2 #ind]].
trans-first-arg-raising-lex-item
[ARG-ST < [LOCAL.CONT.HOOK.INDEX #ind],
[LOCAL.CONT.HOOK [XARG #ind,
LTOP #top]] >,
SYNSEM.LKEYS.KEYREL [ARG1 #top]].
clausal-second-arg-trans-lex-item
[ARG-ST < [LOCAL.CONT.HOOK.INDEX ref-ind & #ind],
[LOCAL.CONT.HOOK.LTOP #top] >,
SYNSEM.LKEYS.KEYREL [ARG1 #ind,
ARG2 #top]].
trans-first-arg-control-lex-item
[ARG-ST < [LOCAL.CONT.HOOK.INDEX ref-ind & #ind],
[LOCAL.CONT.HOOK [XARG #ind,
LTOP #top]] >,
SYNSEM.LKEYS.KEYREL [ARG1 #ind,
ARG2 #top]].
spr-plus-one-arg-lex-item
[ARG-ST < synsem,
[LOCAL.CONT.HOOK.INDEX ref-ind & #ind] >,
SYNSEM.LKEYS.KEYREL.ARG1 #ind].
spr-plus-clausal-arg-lex-item
[ARG-ST < synsem,
[LOCAL.CONT.HOOK.LTOP #top] >,
SYNSEM.LKEYS.KEYREL.ARG1 #top].
transitive-lex-item
[ARG-ST < [LOCAL.CONT.HOOK.INDEX ref-ind & #ind1],
[LOCAL.CONT.HOOK.INDEX ref-ind & #ind2] >,
SYNSEM.LKEYS.KEYREL [ARG1 #ind1,
ARG2 #ind2]].
clausal-expl-arg-lex-item
[ARG-ST < [LOCAL.CONT.HOOK.INDEX expl-ind],
[LOCAL.CONT.HOOK.LTOP #top] >,
SYNSEM.LKEYS.KEYREL [ARG1 #top]].
ditrans-second-arg-rasing-lex-item
[ARG-ST < [LOCAL.CONT.HOOK.INDEX ref-ind & #ind1],
[LOCAL.CONT.HOOK.INDEX ref-ind & #ind2],
[LOCAL.CONT.HOOK [XARG #ind2,
LTOP #top]] >,
SYNSEM.LKEYS.KEYREL [ARG1 #ind1,
ARG2 #top]].
ditrans-second-arg-control-lex-item
[ARG-ST < [LOCAL.CONT.HOOK.INDEX ref-ind & #ind1],
[LOCAL.CONT.HOOK.INDEX ref-ind & #ind2],
[LOCAL.CONT.HOOK [XARG #ind2,
LTOP #top]] >,
SYNSEM.LKEYS.KEYREL [ARG1 #ind1,
ARG2 #top]].
spr-trans-first-arg-control-lex-item
[ARG-ST < synsem,
[LOCAL.CONT.HOOK.INDEX ref-ind & #ind],
[LOCAL.CONT.HOOK [XARG #ind,
LTOP #top]] >,
SYNSEM.LKEYS.KEYREL [ARG1 #ind,
ARG2 #top]].
ditrans-first-arg-raising-lex-item
[ARG-ST < [LOCAL.CONT.HOOK.INDEX #ind1],
[LOCAL.CONT.HOOK.INDEX ref-ind & #ind2],
[LOCAL.CONT.HOOK [XARG #ind1,
LTOP #top]] >,
SYNSEM.LKEYS.KEYREL [ARG1 #ind2,
ARG2 #top]].
ditrans-first-arg-control-lex-item
[ARG-ST < [LOCAL.CONT.HOOK.INDEX ref-ind & #ind1],
[LOCAL.CONT.HOOK.INDEX ref-ind & #ind2],
[LOCAL.CONT.HOOK [XARG #ind1,
LTOP #top]] >,
SYNSEM.LKEYS.KEYREL [ARG1 #ind1,
ARG2 #ind2,
ARG3 #top]].
ditransitive-lex-item
[ARG-ST < [LOCAL.CONT.HOOK.INDEX ref-ind & #ind1],
[LOCAL.CONT.HOOK.INDEX ref-ind & #ind2],
[LOCAL.CONT.HOOK.INDEX ref-ind & #ind3] >,
SYNSEM.LKEYS.KEYREL [ARG1 #ind1,
ARG2 #ind2,
ARG3 #ind3]].
clausal-third-arg-ditrans-lex-item
[ARG-ST < [LOCAL.CONT.HOOK.INDEX ref-ind & #ind1],
[LOCAL.CONT.HOOK.INDEX ref-ind & #ind2],
[LOCAL.CONT.HOOK.LTOP #top] >,
SYNSEM.LKEYS.KEYREL [ARG1 #ind1,
ARG2 #ind2,
ARG3 #top]].
phrase
basic-unary-phrase
[STEM #stem,
SYNSEM.LOCAL.CONT [RELS [LIST #first,
LAST #last],
HCONS [LIST #scfirst,
LAST #sclast]],
C-CONT [RELS [LIST #middle,
LAST #last],
HCONS [LIST #scmiddle,
LAST #sclast]],
ARGS < sign & [STEM #stem,
SYNSEM.LOCAL.local &
[CONT [RELS [LIST #first,
LAST #middle],
HCONS [LIST #scfirst,
LAST #scmiddle]]]] >].
basic-binary-phrase
[SYNSEM.LOCAL.CONT [RELS [LIST #first,
LAST #last],
HCONS [LIST #scfirst,
LAST #sclast]],
C-CONT [RELS [LIST #middle2,
LAST #last],
HCONS [LIST #scmiddle2,
LAST #sclast]],
ARGS < sign & [SYNSEM.LOCAL.local &
[CONT [RELS [LIST #first,
LAST #middle1],
HCONS [LIST #scfirst,
LAST #scmiddle1]]],
sign & [SYNSEM.LOCAL.local &
[CONT [RELS [LIST #middle1,
LAST #middle2],
HCONS [LIST #scmiddle1,
LAST #scmiddle2]]]] >].
non-headed-phrase
named-relation
[PRED named_rel,
CARG string].
phrasal
[SYNSEM phr-synsem].
headed-phrase
[SYNSEM.LOCAL [CAT.HEAD head & #head,
AGR #agr],
HEAD-DTR.SYNSEM.LOCAL.local &
[CAT.HEAD #head,
AGR #agr]].
clause
[SYNSEM [LOCAL.CAT.VAL.COMPS olist,
NON-LOCAL.QUE 0-dlist]].
relative-clause
unary-phrase
[INFLECTED +,
ARGS < [INFLECTED +] >].
head-only
[HEAD-DTR #head,
ARGS < #head >].
head-compositional
[C-CONT.HOOK #hook,
HEAD-DTR.SYNSEM.LOCAL.CONT.HOOK #hook].
basic-binary-headed-phrase
[NON-HEAD-DTR sign].
head-nexus-que-phrase
[SYNSEM.NON-LOCAL.QUE #que,
HEAD-DTR.SYNSEM.NON-LOCAL.QUE #que].
basic-non-rel-clause
[SYNSEM [LOCAL.CONT.MSG #msg &
[LBL #nhand],
NON-LOCAL.REL 0-dlist],
HEAD-DTR.SYNSEM [LOCAL.CONT.HOOK [INDEX #ind],
NON-LOCAL [QUE 0-dlist,
REL 0-dlist]],
C-CONT [HOOK [LTOP #nhand,
INDEX #ind],
RELS.LIST < #msg, ... >]].
binary-phrase
[INFLECTED +,
ARGS < [INFLECTED +],
[INFLECTED +] >].
head-nexus-rel-phrase
[SYNSEM.NON-LOCAL.REL #rel,
HEAD-DTR.SYNSEM.NON-LOCAL.REL #rel].
head-nexus-phrase
basic-head-final
[HEAD-DTR #head,
NON-HEAD-DTR #non-head,
ARGS < #non-head, #head >].
head-valence-phrase
[SYNSEM.NON-LOCAL.SLASH #slash,
HEAD-DTR.SYNSEM.NON-LOCAL.SLASH #slash].
non-clause
[SYNSEM.LOCAL [CONT.MSG no-msg]].
interrogative-clause
[SYNSEM.LOCAL.CONT.MSG.PRED question_m_rel].
non-rel-clause
[SYNSEM.LOCAL.CONT.MSG.MARG #marg,
HEAD-DTR.SYNSEM.LOCAL.CONT.HOOK.LTOP #top,
C-CONT.HCONS.LIST.FIRST qeq & [HARG #marg,
LARG #top]].
basic-extracted-arg-phrase
[SYNSEM.LIGHT -].
head-mod-phrase
[SYNSEM [LOCAL.CAT.VAL [SUBJ #subj,
SPR #spr,
COMPS #comps],
MODIFIED hasmod],
HEAD-DTR.SYNSEM [LOCAL.CAT.VAL [SUBJ #subj,
SPR #spr,
COMPS #comps],
NON-LOCAL [REL 0-dlist]]].
imperative-clause
[SYNSEM.LOCAL.CONT.MSG.PRED command_m_rel].
declarative-clause
[SYNSEM.LOCAL.CONT.MSG.PRED proposition_m_rel].
binary-headed-phrase
basic-extracted-comp-phrase
[SYNSEM canonical-synsem &
[LOCAL.CAT [VAL [SUBJ #subj,
SPR #spr,
COMPS #comps],
MC #mc]],
HEAD-DTR [SYNSEM
[LOCAL.CAT [VAL [SUBJ #subj,
SPR #spr,
COMPS < gap &
[NON-LOCAL.SLASH #slash]
. #comps >],
MC #mc],
NON-LOCAL.SLASH #slash]],
C-CONT [RELS <! !>,
HCONS <! !>]].
head-final
head-initial
[HEAD-DTR #head,
NON-HEAD-DTR #non-head,
ARGS < #head, #non-head >].
basic-head-spec-phrase
[INFLECTED +,
SYNSEM [LOCAL.CAT [VAL [SUBJ #subj,
COMPS #comps,
SPR #spr,
SPEC #spec],
POSTHEAD #ph],
MODIFIED #modif],
HEAD-DTR [INFLECTED +,
SYNSEM [LOCAL [CAT [HEAD #head,
VAL [SUBJ #subj,
COMPS olist & #comps,
SPR < #synsem &
canonical-synsem . #spr >,
SPEC #spec],
POSTHEAD #ph],
CONT.HOOK #hdhook],
MODIFIED #modif],
NON-HEAD-DTR.SYNSEM #synsem &
[LOCAL [CAT [VAL [SPEC < [LOCAL [CAT [HEAD #head,
VAL.COMPS #comps],
CONT.HOOK #hdhook],
MODIFIED #modif] >,
COMPS #comps]],
CONT.HOOK #hook],
MODIFIED #modif],
C-CONT [HOOK #hook,
RELS <! !>,
HCONS <! !>]].
basic-head-mod-phrase-simple
[SYNSEM [LOCAL.CAT.MC #mc,
NON-LOCAL [SLASH [LIST #first,
LAST #last],
REL 0-dlist]],
HEAD-DTR.SYNSEM
[LOCAL [CAT [HEAD #head,
VAL #val,
POSTHEAD #ph,
MC #mc],
AGR #agr,
CONT.HOOK #hdhook],
NON-LOCAL #nonloc & [SLASH [LIST #middle,
LAST #last]],
LIGHT #light,
MODIFIED #modif],
NON-HEAD-DTR.SYNSEM
[LOCAL.CAT [HEAD [MOD < [LOCAL local &
[CAT [HEAD #head,
VAL #val,
POSTHEAD #ph,
MC #mc],
AGR #agr,
CONT.HOOK #hdhook],
NON-LOCAL #nonloc,
LIGHT #light,
MODIFIED #modif] >],
VAL [COMPS olist,
SPR olist],
MC #mc]],
NON-LOCAL [SLASH [LIST #first,
LAST #middle],
QUE 0-dlist & [LIST null]],
C-CONT.RELS <! !>].
basic-filler-phrase
[SYNSEM [LOCAL [CAT [VAL [COMPS < >,
SPR < >],
POSTHEAD +]],
NON-LOCAL.SLASH 0-dlist],
ARGS < [SYNSEM [LOCAL #slash & local &
[CAT.VAL [SUBJ olist,
COMPS olist,
SPR olist],
CTXT.ACTIVATED +],
NON-LOCAL.SLASH 0-dlist]],
[SYNSEM [LOCAL.CAT [VAL.COMPS olist],
NON-LOCAL [SLASH 1-dlist &
[LIST [FIRST #slash,
REST < > & #last],
LAST #last],
QUE 0-dlist,
REL 0-dlist]]] >].
basic-extracted-adj-phrase
basic-head-opt-subj-phrase
[INFLECTED #infl,
SYNSEM canonical-synsem &
[LOCAL.CAT [VAL [SUBJ < >,
COMPS #comps,
SPR #spr,
SPEC #spec],
MC #mc,
POSTHEAD #ph],
MODIFIED #mod],
HEAD-DTR [INFLECTED #infl & +,
SYNSEM [LOCAL [CAT [VAL [SUBJ < unexpressed-reg &
[OPT +,
LOCAL.CONT.HOOK.INDEX.DEF +] >,
COMPS #comps & < >,
SPR #spr,
SPEC #spec],
MC #mc,
POSTHEAD #ph],
CONT.HOOK.INDEX event],
MODIFIED #mod]]].
basic-head-comp-phrase
[SYNSEM canonical-synsem &
[LOCAL.CAT [MC #mc,
VAL [SUBJ #subj,
COMPS #comps,
SPR #spr],
POSTHEAD #ph],
LIGHT #light],
HEAD-DTR.SYNSEM [LOCAL.CAT [MC #mc,
VAL [SUBJ #subj,
COMPS < #synsem . #comps >,
SPR #spr],
HC-LIGHT #light,
POSTHEAD #ph]],
NON-HEAD-DTR.SYNSEM #synsem & canonical-synsem,
C-CONT [RELS <! !>,
HCONS <! !>]].
basic-head-filler-phrase
basic-head-subj-phrase
[SYNSEM.LOCAL.CAT [POSTHEAD +,
VAL [SUBJ < >,
COMPS #comps,
SPR #spr]],
HEAD-DTR.SYNSEM.LOCAL.CAT.VAL [SUBJ < #synsem >,
COMPS #comps,
SPR #spr],
NON-HEAD-DTR.SYNSEM #synsem & canonical-synsem &
[LOCAL [CAT [VAL [SUBJ olist,
COMPS olist,
SPR olist]],
NON-LOCAL [SLASH 0-dlist & [LIST < >],
REL 0-dlist,
QUE 0-dlist]]].
basic-extracted-subj-phrase
[SYNSEM.LOCAL.CAT.VAL [SUBJ < >,
SPR < >,
COMPS < >],
HEAD-DTR.SYNSEM [LOCAL.CAT [VAL [SUBJ < gap &
[LOCAL #local & local &
[CONT.HOOK.INDEX
ref-ind]] >,
COMPS olist],
MC na],
NON-LOCAL.SLASH.LIST < #local >]].
basic-head-opt-comp-phrase
[INFLECTED #infl,
SYNSEM canonical-synsem &
[LOCAL.CAT [VAL [SUBJ #subj,
COMPS #comps,
SPR #spr,
SPEC #spec],
MC #mc,
POSTHEAD #ph],
MODIFIED #mod],
HEAD-DTR [INFLECTED #infl & +,
SYNSEM [LOCAL [CAT [VAL [SUBJ #subj,
COMPS < unexpressed &
[OPT +,
DEF-OPT #def,
LOCAL.CONT.HOOK.INDEX.DEF #def] . #comps >,
SPR #spr,
SPEC #spec],
MC #mc,
POSTHEAD #ph],
CONT.HOOK.INDEX event],
MODIFIED #mod]],
C-CONT [RELS <! !>,
HCONS <! !>]].
head-mod-phrase-simple
[HEAD-DTR.SYNSEM.LOCAL.CONT.HOOK.LTOP #top,
NON-HEAD-DTR.SYNSEM.LOCAL.CAT.HEAD.MOD < [LOCAL.CONT.HOOK.LTOP #top] >].
isect-mod-phrase
[HEAD-DTR.SYNSEM.LOCAL.CONT [HOOK.LTOP #hand,
MSG no-msg],
NON-HEAD-DTR.SYNSEM.LOCAL [CAT.HEAD.MOD < [LOCAL intersective-mod] >,
CONT.HOOK.LTOP #hand],
C-CONT.HCONS <! !>].
head-adj-phrase
[SYNSEM [LOCAL.CAT.POSTHEAD +,
MODIFIED mod],
HEAD-DTR.SYNSEM.MODIFIED notmod-or-rmod,
NON-HEAD-DTR.SYNSEM [LOCAL.CAT.POSTHEAD +,
NON-LOCAL.QUE 0-dlist]].
extracted-adj-phrase
[SYNSEM [LOCAL.CAT [POSTHEAD #ph,
MC #mc],
NON-LOCAL.SLASH 1-dlist &
<! [CAT [POSTHEAD +,
HEAD [MOD < [LOCAL intersective-mod &
[CAT [HEAD #head,
VAL #val,
POSTHEAD #ph,
MC #mc],
CONT.HOOK #hook,
CTXT #ctxt]] >,
VAL [SUBJ olist,
COMPS olist,
SPR olist]]] !>],
HEAD-DTR.SYNSEM canonical-synsem &
[LOCAL.local &
[CAT [HEAD #head,
VAL #val & [SUBJ < synsem-min &
[NON-LOCAL.SLASH 0-dlist] >],
POSTHEAD #ph,
MC #mc],
CONT.HOOK #hook,
CTXT #ctxt],
NON-LOCAL.SLASH 0-dlist,
MODIFIED notmod],
C-CONT [HOOK #hook,
HCONS <! !>]].
scopal-mod-phrase
[NON-HEAD-DTR.SYNSEM.LOCAL [CAT.HEAD.MOD < [LOCAL scopal-mod] >,
CONT.HOOK #hook],
C-CONT [HOOK #hook,
HCONS <! !>]].
adj-head-phrase
[SYNSEM [LOCAL.CAT.POSTHEAD #ph,
MODIFIED bmod & [PERIPH #periph],
LIGHT #light],
HEAD-DTR.SYNSEM [LOCAL.CAT.POSTHEAD #ph,
LIGHT #light],
NON-HEAD-DTR.SYNSEM [LOCAL.CAT [VAL.COMPS < >],
NON-LOCAL [SLASH 0-dlist,
REL 0-dlist],
MODIFIED.PERIPH #periph]].
head-adj-int-phrase
adj-head-int-phrase
[NON-HEAD-DTR.SYNSEM.LOCAL.CAT [POSTHEAD -]].
adj-head-scop-phrase
[NON-HEAD-DTR.SYNSEM.LOCAL.CAT.POSTHEAD -].
head-adj-scop-phrase
[NON-HEAD-DTR.SYNSEM.NON-LOCAL.REL 0-dlist].