HPSG Processing of Japanese

Melanie Siegel

JACY

Outline

- Motivation
- Framework, tools, project context
- Fundamentals of JACY
- Basic phrase structures
- The treatment of subcategorization
- Inflection and derivation
- Auxiliaries
- ChaSen integration

Motivation

- **Applications** that rely on deep linguistic processing, such as message extraction systems, machine translation and dialogue understanding systems are becoming **feasible**.
- **Requirement** for rich and highly precise information, well-defined output structures.
- **Requirement** for robustness: wide coverage, large and extensible lexica, interfaces to preprocessing.

Motivation

- **Requirement** for exensibility to multiple languages.
- **Requirement** for efficient processing.
- *The JACY Japanese HPSG has been developed for and used in real-world applications that require the handling of phenomena at the edge position of the language.*

Framework

- Head-Driven Phrase Structure Grammar
  - Feature structures
  - Type hierarchy
  - Efficient processing
- Minimal Recursion Semantics
  - Flat semantic formalism
  - Works well with typed feature structures
  - Structures are underspecified for scopal information (compact representation of ambiguities)

The Tools

- LKB grammar development system (Copestake 2002)
- PET efficient processing system for HPSG grammars (Callmeier 2000)
- ChaSen tokenizer and POS tagger (Asahara & Matsumoto 2000)
- [incr tsdb] grammar testing tool (Oepen and Carroll 2000)
- Heart-of-Gold architecture for the combination of shallow and deep processing (Callmeier et al. 2004)

The JACY grammar: Project context

- 1998-2000
  - Verbmobil: Machine translation of application-oriented spoken dialogues.

    http://verbmobil.dfki.de/
- 2001-2002
  - Co-operation with YY Technologies (CA, USA): Automatic email response (Co-operation with Stephan Oepen, Ulrich Callmeier, Monique Sugimoto, Atsuko Shimada, Dan Flickinger)

    http://www.dfki.de/~siegel/jacy/jacy.html
- 2002-2004
  - EU project DEEP THOUGHT: Hybrid and shallow methods for knowledge-intensive information extraction

    http://www.project-deepthought.net

Japanese open-source HPSG

- JACY is an open-source HPSG grammar for Japanese.
- JACY homepage: www.dfki.de/~siegel/grammar-download/JACY-grammar.html

Multilingual grammar development

- Available HPSG grammars in the DeepThought project:
  - German (50.000 lexical entries)
  - English (12.300 lexical entries)
  - Japanese (35.000 lexical entries)
  - Norwegian (84.240 lexical entries)
  - Italian (4.850 lexical entries)
- A Grammar Matrix allows the efficient implementation of new grammars with compatible and correct output.
- RMRS as the common semantic formalism allows usability of NLP modules for applications.

Multilingual grammar development

- Delph-In includes open-source resources:
  - LKB grammar development system (incl. generation)
  - PET grammar processing system
  - [incr tsdb()] grammar profiling system
  - ERG English HPSG
  - JACY Japanese HPSG
  - NorSource Norwegian HPSG
  - Modern Greek Resource Grammar
  - Lingo Grammar Matrix
  - Redwoods treebank
  - DeepThought Heart of Gold

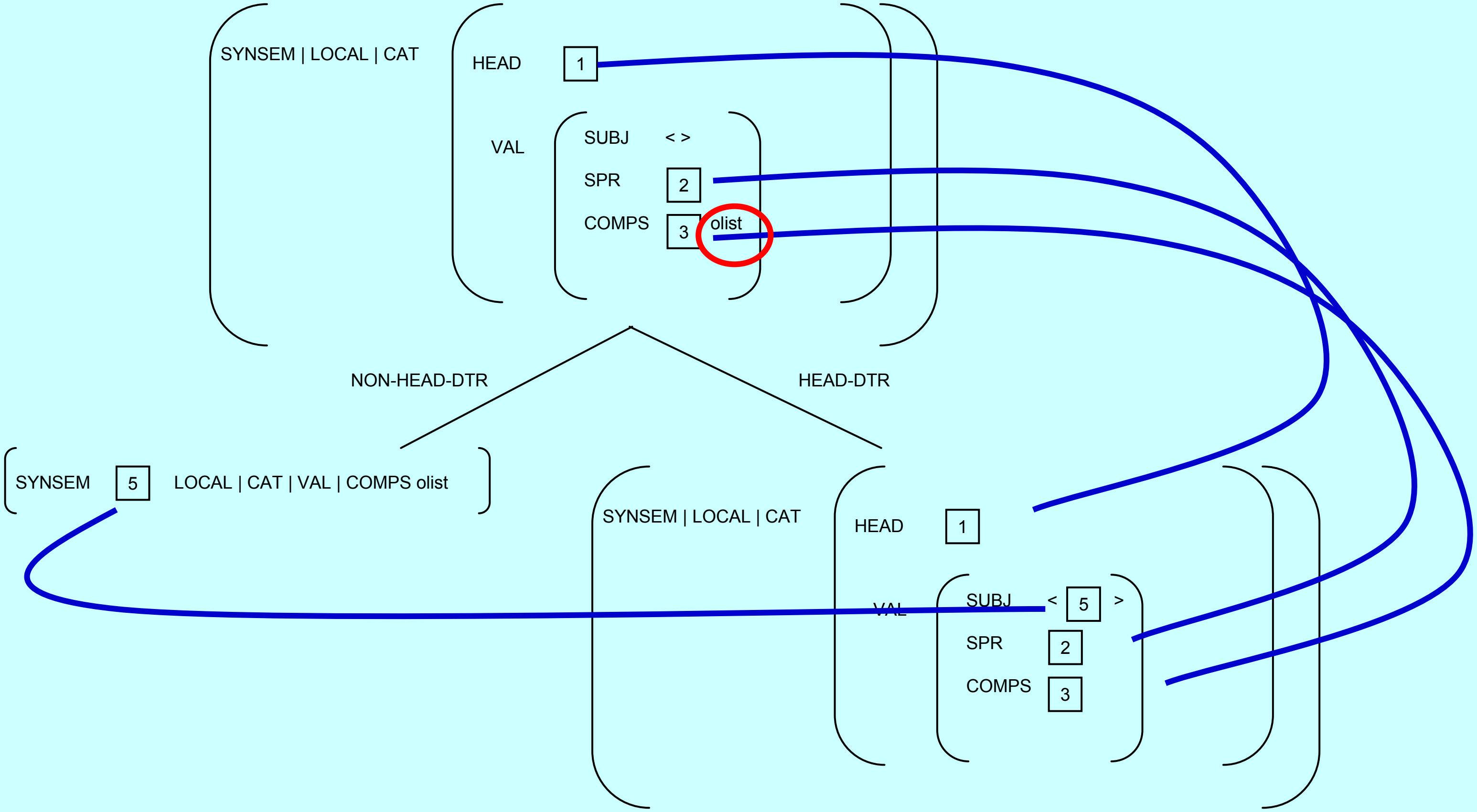
Fundamental notion: sign

- Sign:
  - complex feature structure representing information of different linguistic levels of a phrase or lexical item.
  - similar to a sign in the LinGO English Resource Grammar:
    - orthographical realization of the lexical sign in ORTH .
    - syntactic and semantic information in SYNSEM
    - lexical status in LEX
    - nonlocal information in NON-LOCAL
    - head information that goes up the tree in HEAD
    - information about subcategorization in VAL.

Fundamental notion: types

- The grammar implementation is based on a system of types.
- 990 lexical types define the syntactic, semantic and pragmatic properties of the Japanese words
- 188 types that define the properties of phrases and lexical and inflectional rules.
- 50 rules for inflectional and derivational morphology and lexical rules.
- 47 phrase structure rules (instances of rule types).

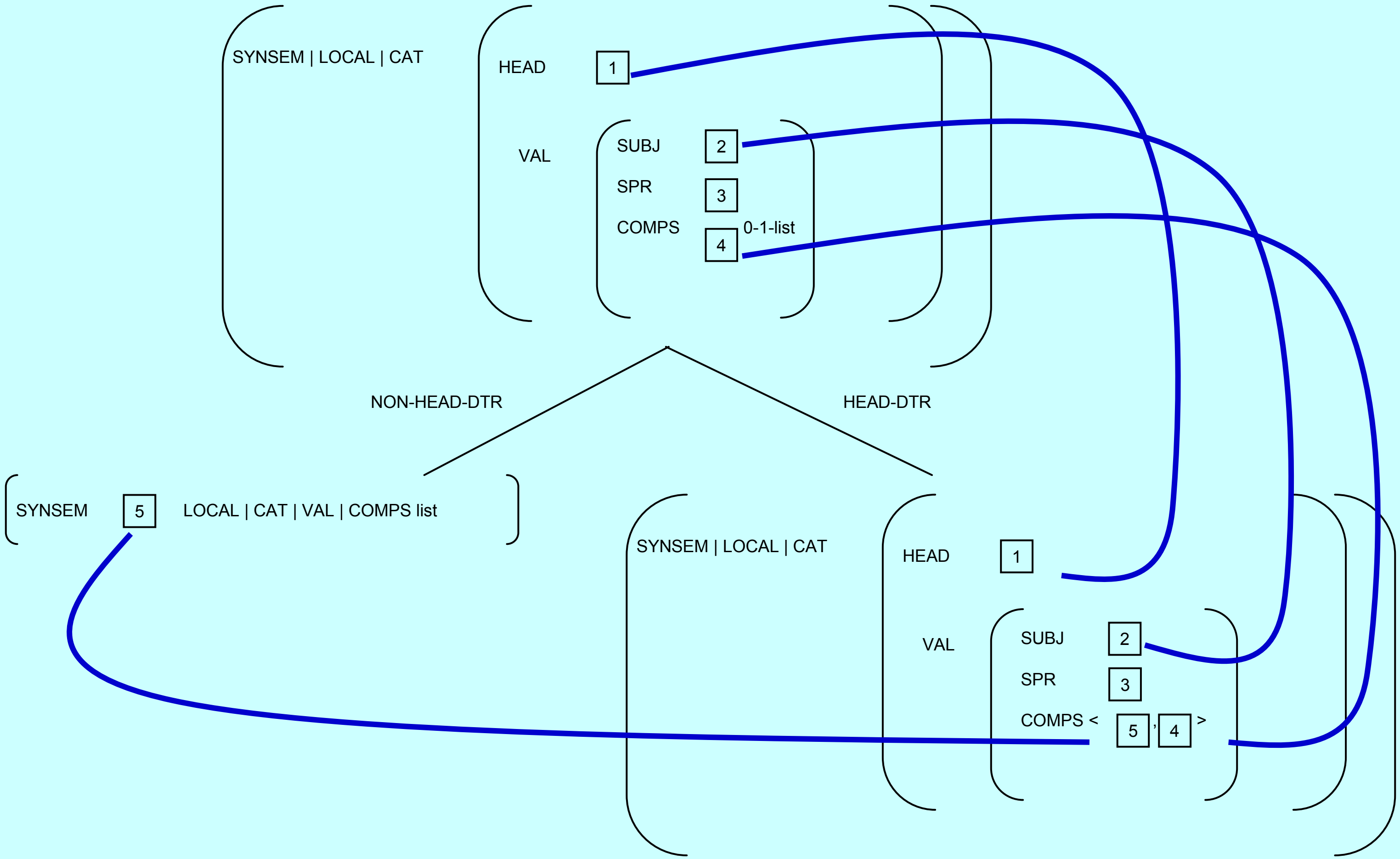
Phrase structure: head-subject



Differences to Pollard/Sag 1994

- The subcategorization value is not a single list, but a complex structure containing three different kinds of lists and an UNSAT value.
- The complement list is not necessarily empty when binding the subject.
- SLASH is not used in the Japanese structure.

Phrase structure: head-complement



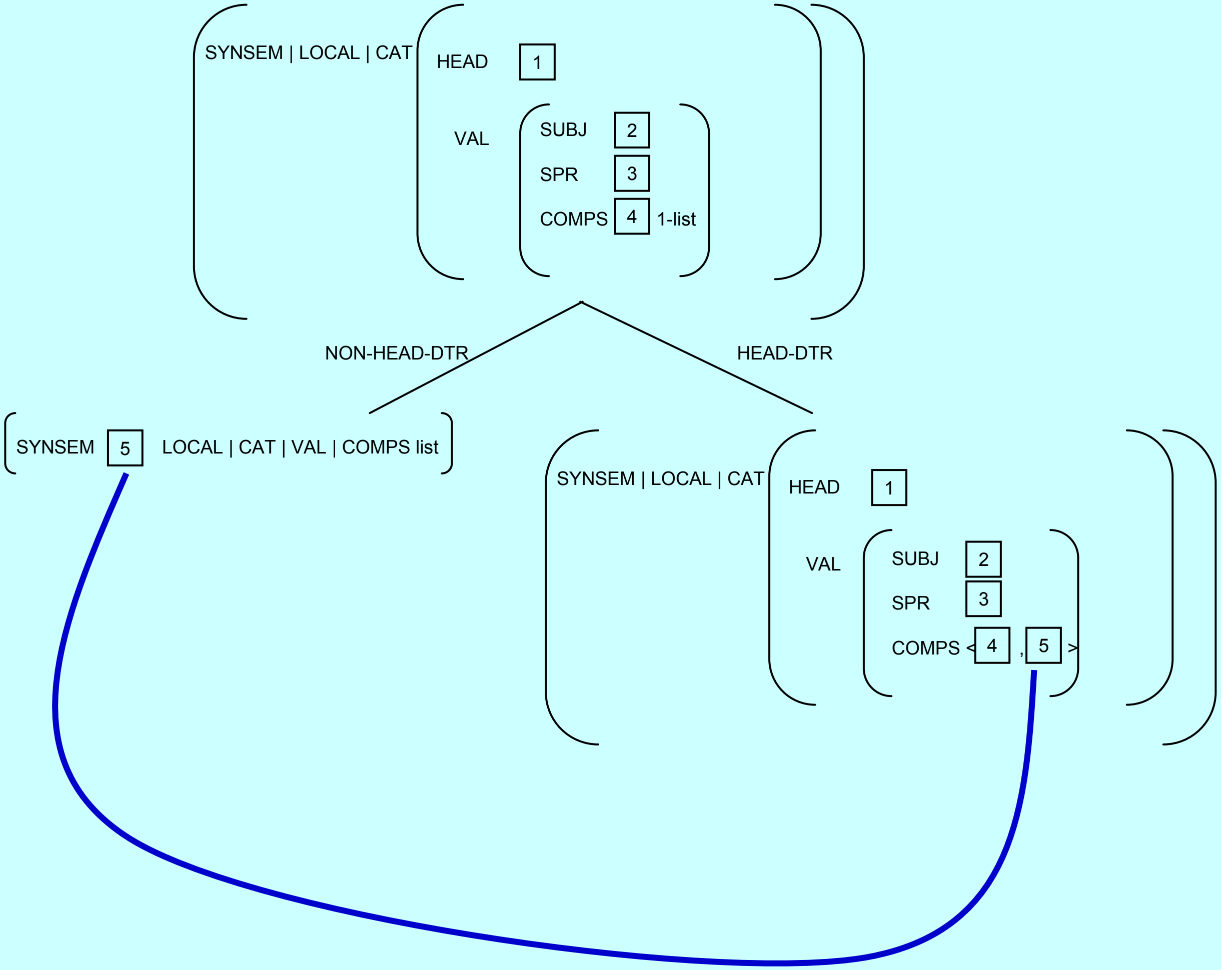
Differences to Pollard/Sag 1994

- JACY’s basic head-complement rule type does not constrain its Head-Daughter value to be a word, as the notion of word is in principle problematic in Japanese language processing.

Rule instances of head-complement-rule

- head-complement-hf-type for head-final complementation.
- head-complement-hi-type for head-initial complementation.

Phrase structure: head-complement2



SYNSEM | LOCAL | CAT
HEAD 1
VAL
SUBJ 2
SPR 3
COMPS 4 1-list

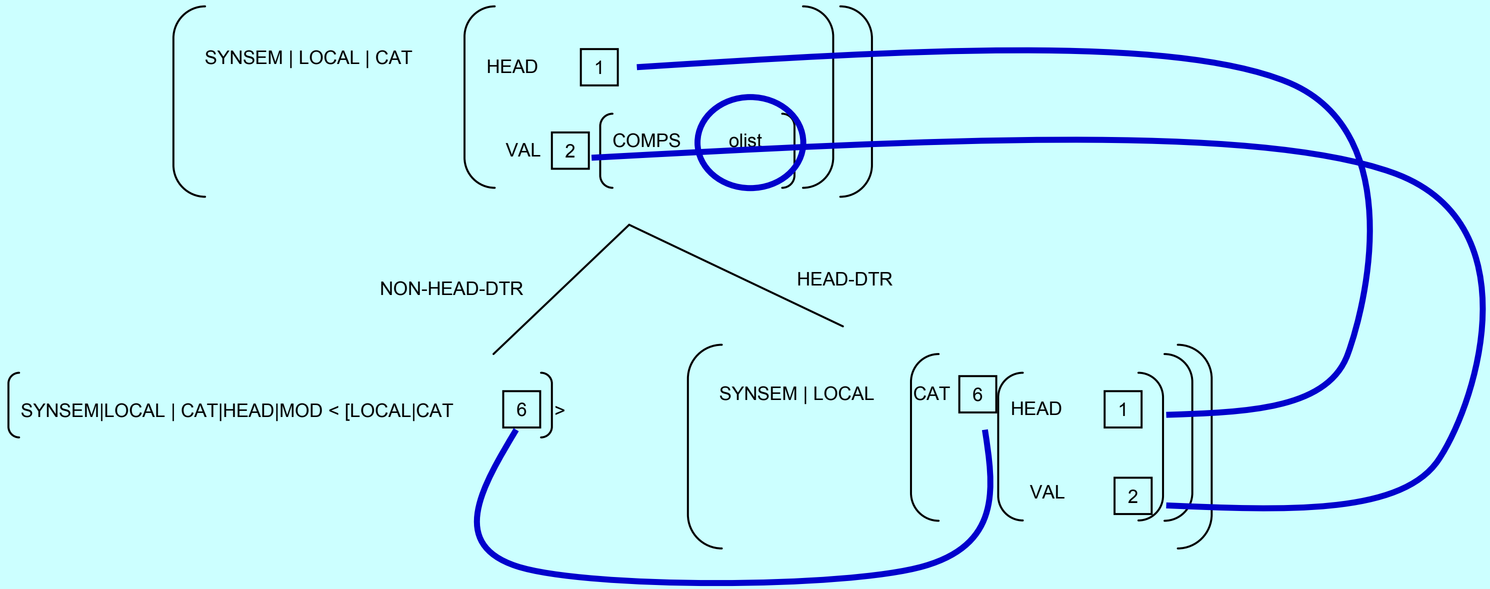
NON-HEAD-DTR

HEAD-DTR

SYNSEM 5 LOCAL | CAT | VAL | COMPS list

SYNSEM | LOCAL | CAT
HEAD 1
VAL
SUBJ 2
SPR 3
COMPS < 4 , 5 >

Phrase structure: head-adjunct



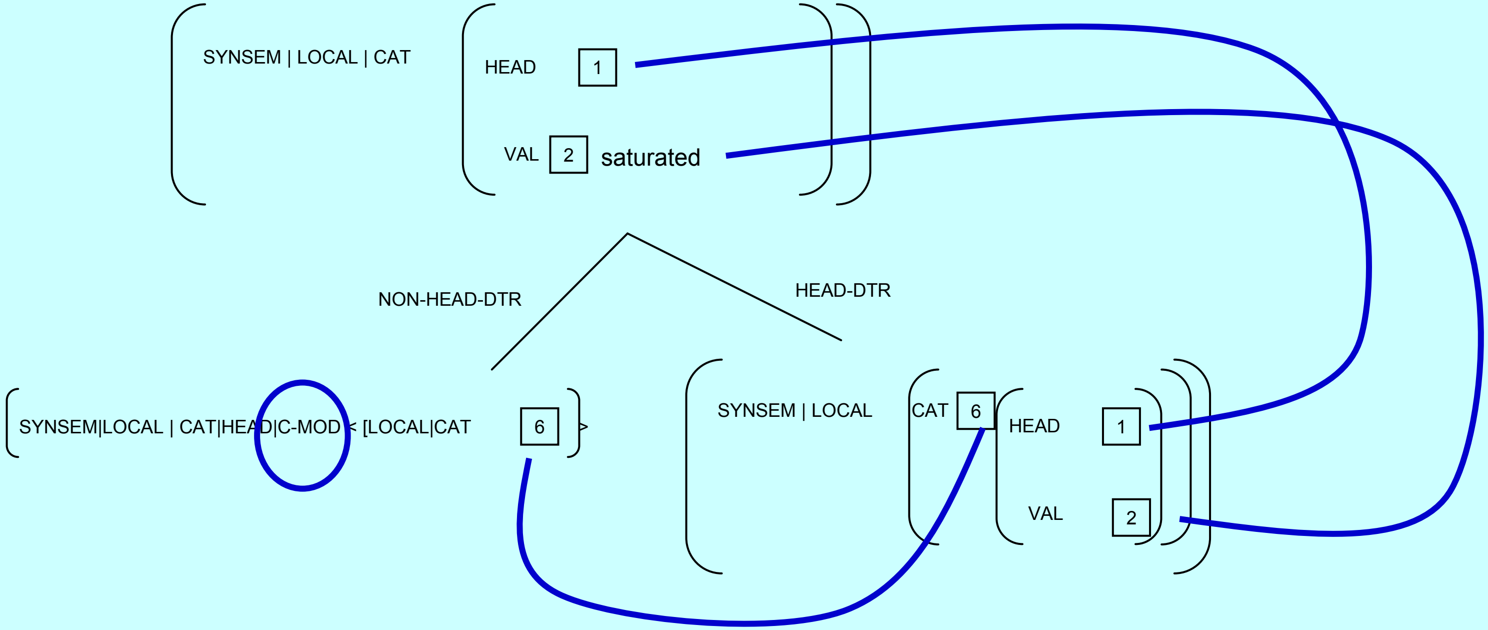
Differences to Pollard/Sag 1994

- As the SLASH mechanism is not used for scrambling in JACY, the COMPS list in VAL is restricted to *olist*, a list of optional arguments. This ensures that no adjacent arguments are allowed to be on the valence list, when adjuncts are found.

Sentence coordination

花子	が	ご飯	を	食べて、	早く	寝た
Hanako	ga	gohan	wo	tabete,	hayaku	neta
Hanako	*NOM*	*rice*	*ACC*	*eat,*	*quickly*	*slept*

Phrase structure: sentence-coordination



SYNSEM | LOCAL | CAT [HEAD 1, VAL 2 saturated]

NON-HEAD-DTR

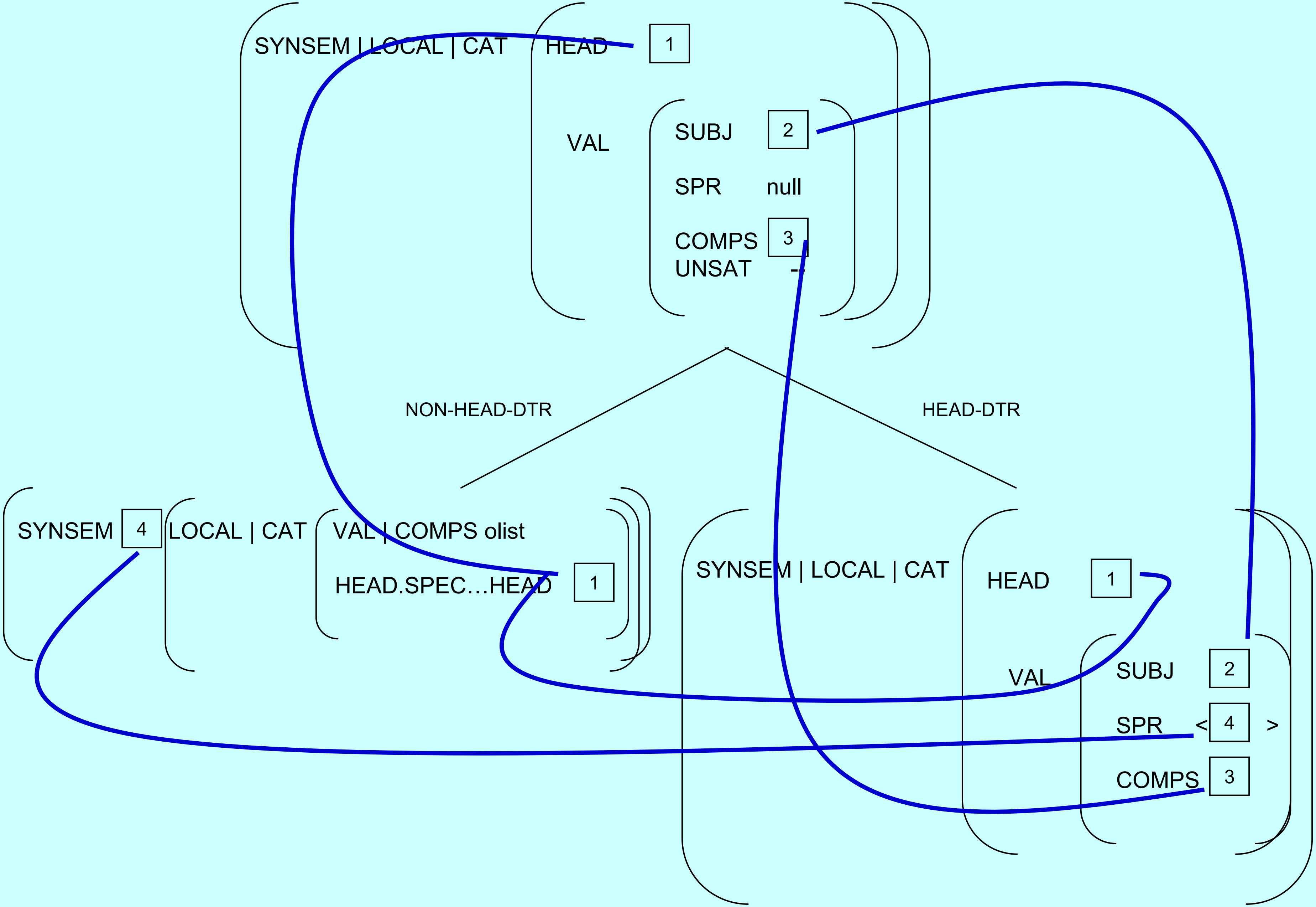
HEAD-DTR

SYNSEM|LOCAL | CAT|HEAD|C-MOD < [LOCAL|CAT 6] >

SYNSEM | LOCAL [CAT 6 [HEAD 1, VAL 2]]

Phrase structure: head-specifier constructions

- Combination of head and valence information of the daughters:
  - determiner-noun (その人)
  - surname-title (田中さん)
  - noun-title (学生さん)
  - nominalizations (ご飯を食べられないこと)
    - A predicative nominalization subcategorizes for a verb, while the verbal endings on the other hand determine the SPEC behaviour of the verb.
  - auxiliary-verb constructions (ご飯を食べています)



SYNSEM | LOCAL | CAT HEAD 1

VAL SUBJ 2

SPR null

COMPS 3

UNSAT --

NON-HEAD-DTR

HEAD-DTR

SYNSEM 4 LOCAL | CAT VAL | COMPS olist

HEAD.SPEC…HEAD 1

SYNSEM | LOCAL | CAT HEAD 1

VAL SUBJ 2

SPR < 4 >

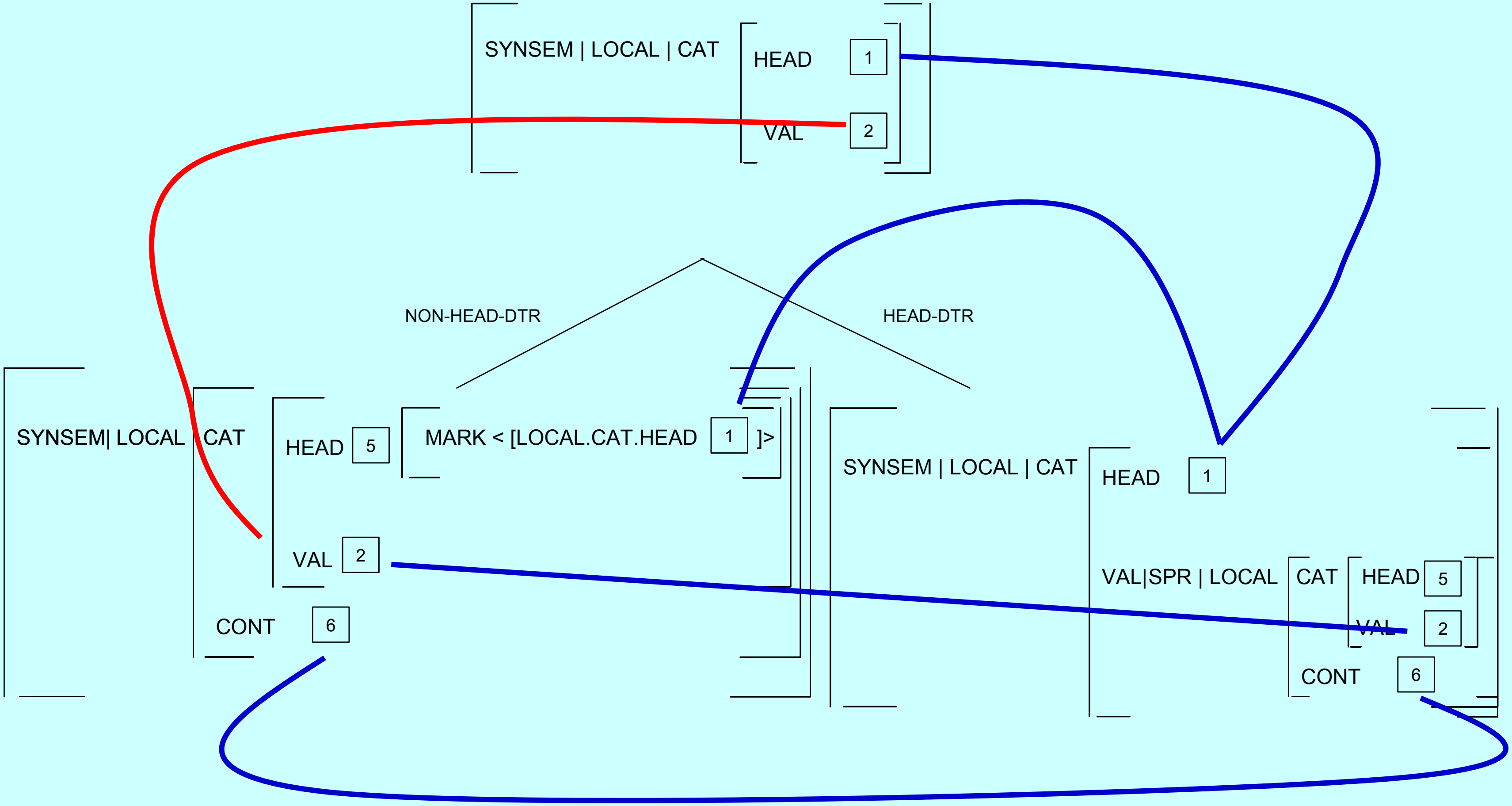
COMPS 3

Phrase structure: head-marker constructions

- verbal noun + light verb (勉強した)
  - The verbal noun *benkyou* contains subcategorization information (transitive), as well as semantic information (the *benkyou*-relation and its semantic arguments).
  - The light verb *shita* supplies tense information (past).
  - Pragmatic information can be supplied by both parts of the construction.
- verbal ending + verbal ending (させられた)

Sub-syntactic phenomena.

Head-marker type



SYNSEM | LOCAL | CAT HEAD 1 VAL 2

NON-HEAD-DTR

HEAD-DTR

SYNSEM| LOCAL CAT HEAD 5 MARK < [LOCAL.CAT.HEAD 1]>

VAL 2

CONT 6

SYNSEM | LOCAL | CAT HEAD 1

VAL|SPR | LOCAL CAT HEAD 5

VAL 2

CONT 6

Facts about Japanese Subcategorization

- Verbal arguments scramble.
- Verbal arguments are frequently omitted.
- Obligatory arguments are also adjacent.
- The Japanese subject is special:
  - It is never obligatory.
  - It restricts subject honorification.
  - It restricts reflexive binding.

Arguments and adjuncts

- Subjects are arguments.
  - They are always optional.
  - They are the goal of subject honorification.
  - They are nominative case.
- Entities that are obligatory are always arguments.
- Entities that are marked by wo (accusative) are arguments.
  - They can be optional or obligatory/adjacent.
- Entities that can be passivized are arguments.
  - It has to be shown, whether this is valid in the other direction as well, such that things that cannot be passivized are adjuncts.
- Things that get a semantic restriction from the head are arguments.

The (traditional) HPSG Approach

- The SPR list contains determiners and subjects.
- The COMPS list contains objects.
- The ARG-ST list contains all arguments (for the statement of binding conditions).
  - →No division of adjacent/obligatory and optional/scrambling arguments.
  - →Cannot account for scrambling, because the lists are sorted.
  - →Cannot account for the speciality of the Japanese subject.

The JPSG Approach

- An un-ordered set of categories instead of a list.
- ADJACENT contains a set of adjacent complements.
  - →But the TDL formalism does not allow sets.
  - →We want to restrict a complement of a lexical type, underspecified whether it is adjacent or optional/scrambled in an actual lexical item: We want to define general subcategorization patterns.

How to cope with the problem?

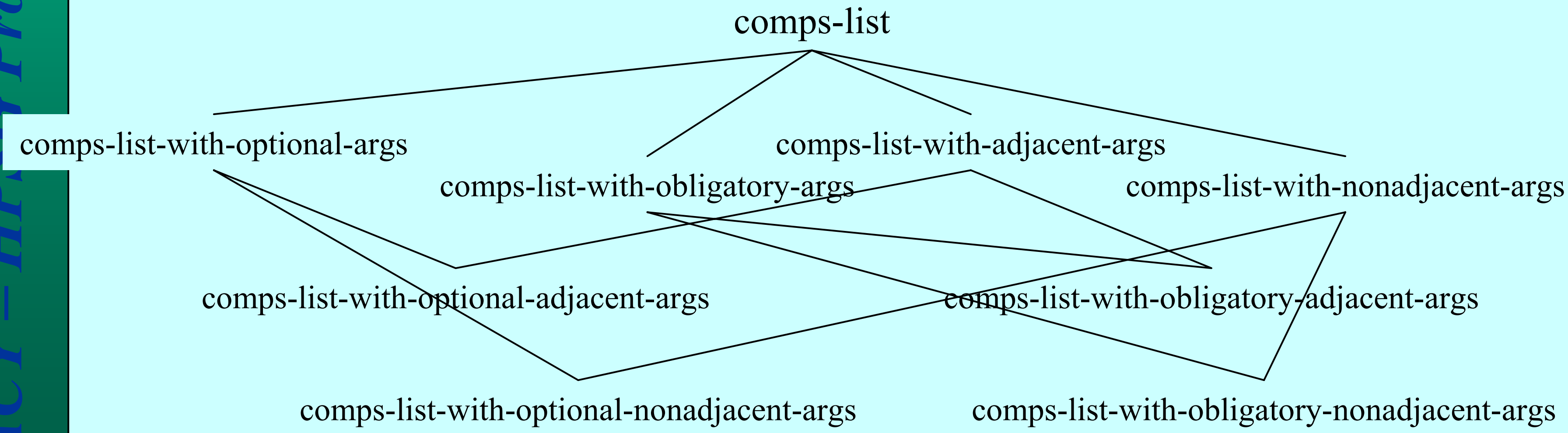
- Use grammatical functions (earlier version of Japanese grammar):
  - naming is not obvious, approach cannot easily be applied to other languages.
- Build several lexicon entries for one lexeme, where each entry represents one possible argument structure (strategy in the Verbmobil German HPSG):
  - the lexicon explodes, especially for a language like Japanese.
  - the approach lacks generality and modularity.

How to cope with the problem?

- Divide the COMPS into COMPS and ADJACENT (JPSG approach)
  - generalizations about, e.g. object control, cannot be easily stated.
  - not applicable to languages where arguments can be optional and adjacent.
- Use a scrambling lexical rule that takes a lexicon entry and produces COMPS lists in various order
  - explosion of grammar processing?

The Matrix Idea!

- Stay with the COMPS list. Add a SUBJ list.
- Add constraints about optionality and adjacency.
- Order types of possible argument structures in a type hierarchy.


comps-list
comps-list-with-optional-args
comps-list-with-adjacent-args
comps-list-with-obligatory-args
comps-list-with-nonadjacent-args
comps-list-with-optional-adjacent-args
comps-list-with-obligatory-adjacent-args
comps-list-with-optional-nonadjacent-args
comps-list-with-obligatory-nonadjacent-args

The Matrix Idea

- Use different head-complement structures that pick up the first, second or third argument of the COMPS list and are not ordered in their application.

Application to the Japanese Grammar

Add the principle of adjacency to the grammar theory:

In a headed phrase, the VALENCE of the non–head daughter must contain only arguments of the type comps-list-with-nonadjacent-arguments.

In a head–complement structure, the VALENCE of the head daughter must contain only arguments of the type comps-list-with-nonadjacent-arguments besides the non–head daughter.

In a head–adjunct structure, the VALENCE of the head daughter must contain only arguments of the type comps-list-with-nonadjacent-arguments.

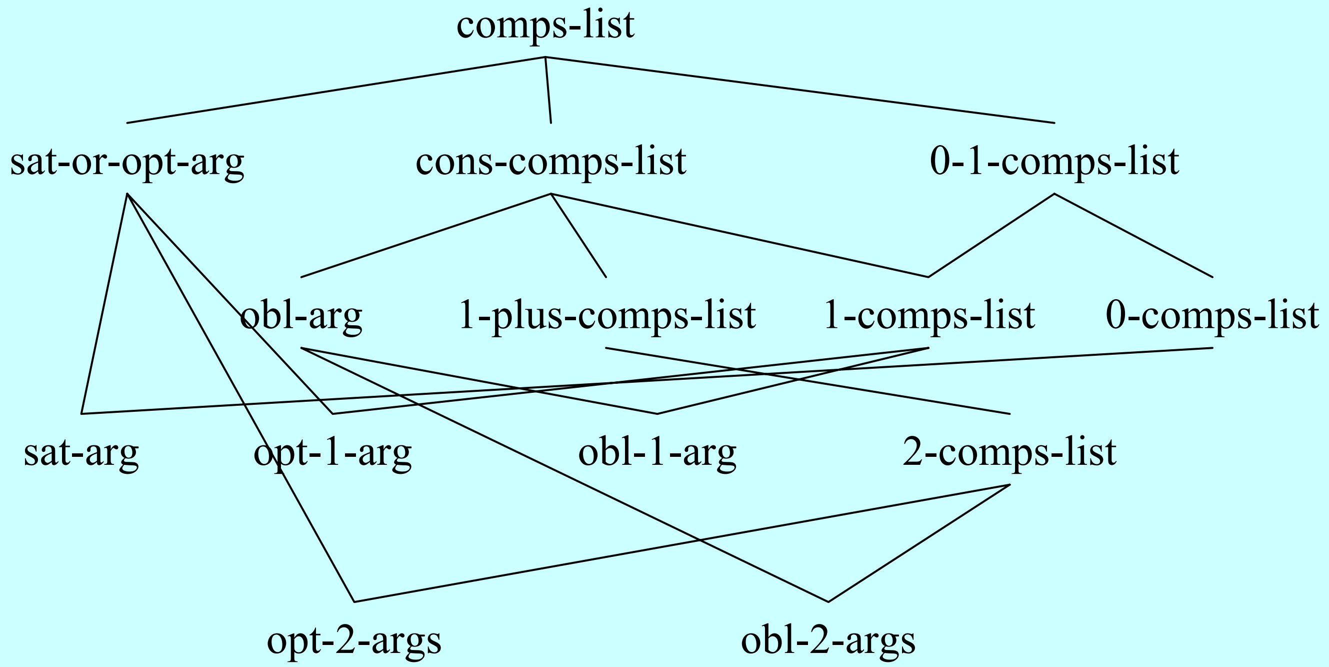
Application to the Japanese Grammar

- Valence feature of a typical transitive verb:

$$\left(\text{VAL}\left(\begin{array}{ll} \multicolumn{2}{l}{\textit{ga-wo-transitive}} \\ \text{UNSAT} & \textit{plus} \\ \text{SUBJ} & \textit{opt-1-arg} \ \& < [\text{LOCAL.CAT.HEAD.CASE } \textit{ga}]> \\ \text{COMPS} & \textit{opt-1-arg} \ \& < [\text{LOCAL.CAT.HEAD.CASE } \textit{wo}]> \\ \text{SPR} & \textit{0-comps-list} \end{array}\right)\right)$$

Application to the Japanese Grammar

- A type hierarchy of complement list types:



Application to the Japanese Grammar

- The necessary **distinction of argument types** for Japanese, optional versus obligatory/adjacent, can be described in a type system.
- The problem of **scrambling** of verbal arguments is solved. There are two head-complement-rules.
- **Zero pronouns** are accounted for by lexical rules. These insert semantic information to the verbal MRS and empty the valence list.

Application to the Japanese Grammar

- The **subject** has a special status and can be restricted. The formulation of empathy and honorification relating the subject argument is possible through direct access to the subject.
- It is easy to express **generalizations**, as for example the fact that verbal subjects are always optional in Japanese. The type *subj-arg* contains as value of SUBJECT the type *opt-1-arg*.

Verbal subcategorization

- There are intransitive (subj-arg), transitive (subj-comps-arg) and ditransitive subcategorization types.

verbal type	subcategorization pattern	example
intrans-stem-lex	SBJ(P,ga)	太る
to-intrans-stem-lex	COMP(P,to,opt)	言う
v1-stem-lex	SBJ(P,ga), COMP(P,wo, opt)	見守る
v2-stem-lex	SBJ(P,ga), COMP(P,ni)	乗る
v2a-stem-lex	SBJ(P,ga), COMP(ADV,obl)	なる
v2b-stem-lex	SBJ(P,ga), COMP(P,ni or to, opt)	なる
v3-stem-lex	SBJ(P,ga), COMP(P,to, obl)	言う
v4-stem-lex	SBJ(P,ga), COMPS (P,wo, opt; P,ni,opt)	置く
v5-stem-lex	SBJ(P,ga), COMP(P,to, opt)	付き合う
v5a-stem-lex	SBJ(P,ga), COMP(P,to, obl)	思い出す
v6-stem-lex	SBJ(P,ga), COMP(P,ni or to, opt)	入れ替わる
v8-stem-lex	SBJ(P,ga), COMP(N,obl)	書き送る
cop-id-stem-lex	SBJ(P,ga-or-coparg), COMP(N,obl)	です

From stem to word

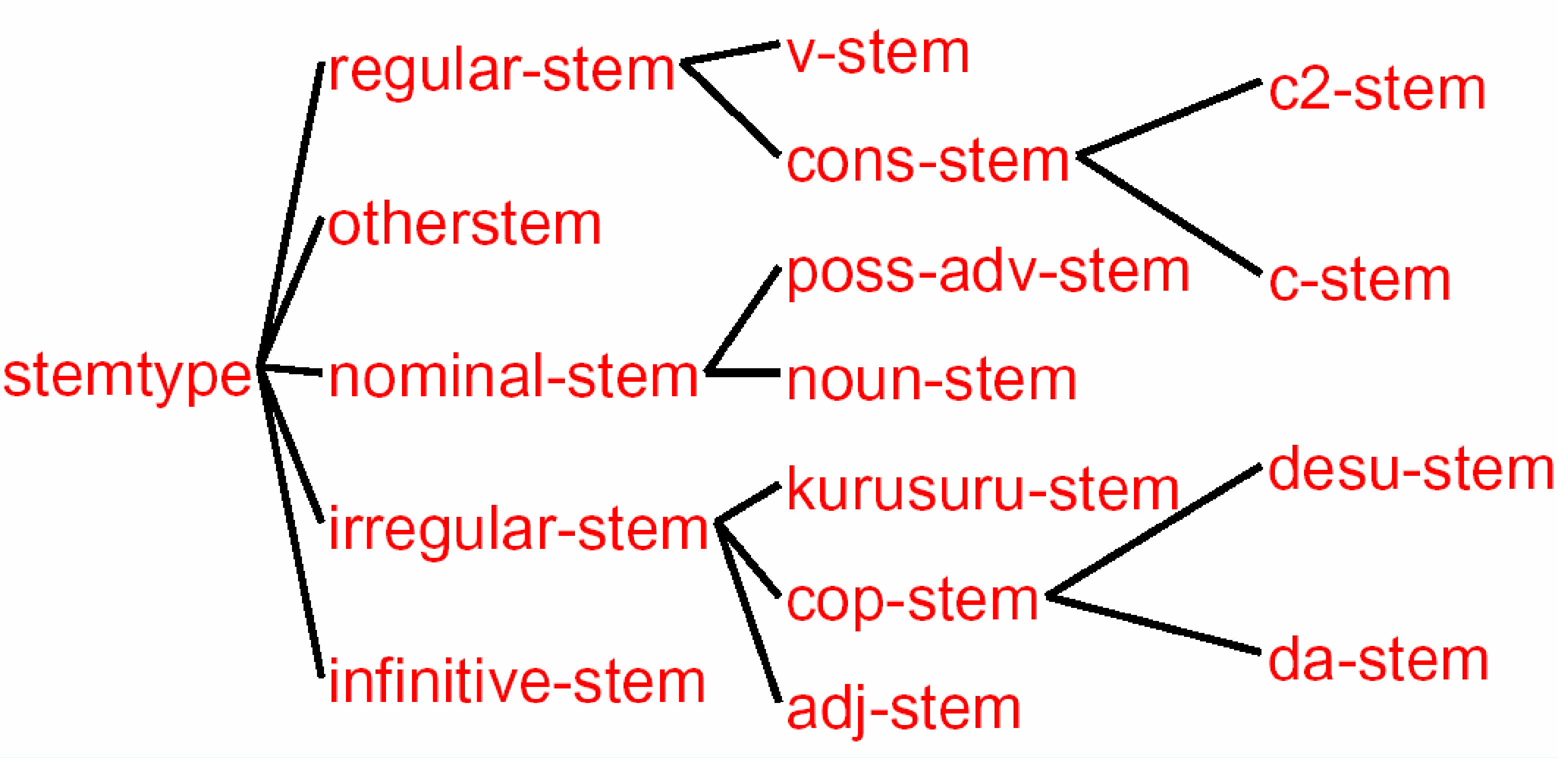
Word stems

- Verbs:
  - c-stem: 置く
  - v-stem: 食べる
  - c2-stem: 行く, いらっしゃる, 問う
  - kurusuru-stem: 来る, する
  - cop-stem: です、だ
- Adjectives:
  - adj-stem: 高い
- Nouns:
  - ordinary-nohon-n-lex: 本

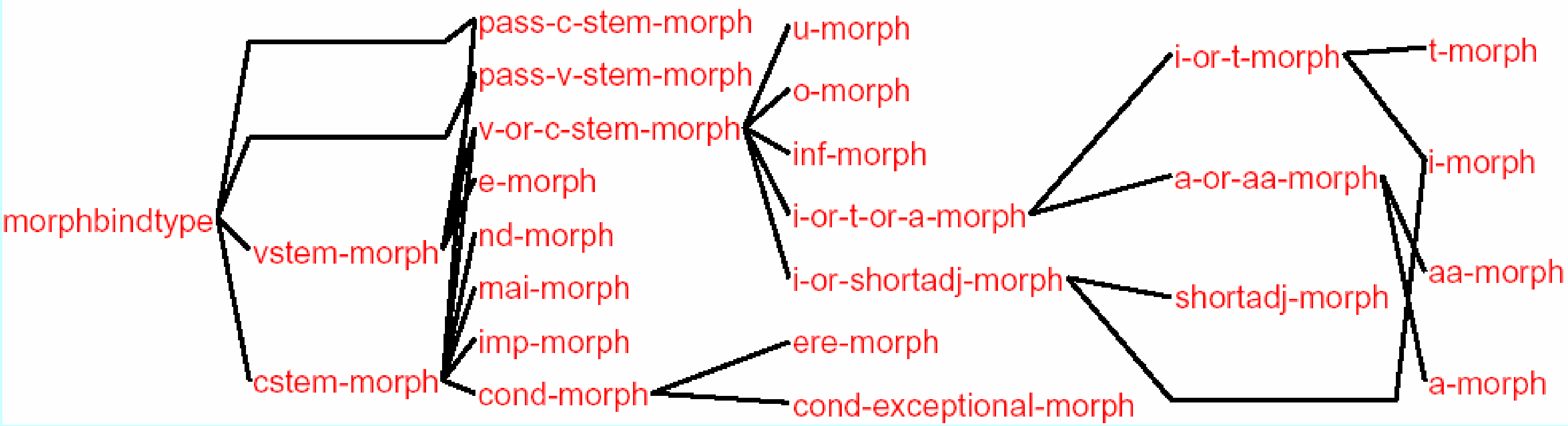
Inflectional rules apply to stems

stem types	**inflection when combined with past tense ending**	**example**
v-stem	no change	食べ → 食べ
c-stem	く→い、る→っ、う→っ、す→し、つ→っ、む→ん、ぐ→い、ぶ→ん、ぬ→ん	聞く→聞い
c2-stem	く→っ、る→っ、う→う	行く→行っ
kurusuru-stem	来る→来、する→し、くる→き	来る→来
cop-stem	だ→だっ、す→し	です→でし
adj-stem	い→かっ	高い→高かっ

The type hierarchy of stem types



Inflectional rules can make morphologic changes and give the result a morphological type



example of the information an inflectional rule adds:

RMORPH-BIND-TYPE	*i-morph*
SYNSEM.LOCAL.CAT.HEAD.MODUS	*indicative*
ARGS.FIRST.STEMTYPE	*c-stem*

JACY – HPSG Processing of Japanese

Rules that pipe stems to words (no ending)

Name of inflection rule	Change of morphological type	Example
ru-lexeme-infl-rule	(no change to the morphology)	食べる
eru-lexeme-infl-rule[1]	(regular-stem -> u-morph)	学べる
infinitive-lexeme-1-infl-rule	(regular-stem -> inf-morph)	食べ
imperative-c2-stem-infl-rule	(c2-stem -> imp-morph)	下さい
desu-lexeme-infl-rule	(cop-stem -> u-morph)	です
de-lexeme-infl-rule	(desu-stem -> u-morph)	で
ra-lexeme-infl-rule	(da-stem -> u-morph)	なら
kuru-lexeme-infl-rule	(kurusuru-stem -> u-morph)	来る
infinitive-lexeme-2-infl-rule	(kurusuru-stem -> inf-morph)	来
adj-i-lexeme-infl-rule	(adj-stem -> u-morph)	ではない

[1] This rule transforms to potential form.

Derivational rules

Rule	Derivation	Inflection	Example
adj2adv-lexeme-infl-rule	adj-stem -> adverb	い → く	高い → 高く
adj2v-infl-rule	adj-stem -> verb	い → 過ぎる	高い → 高過ぎる
v2vn-infl-rule	regular-stem -> vn	く → き、 す → し (and others)	食べる → 食べ
v2n-infl-rule	regular-stem -> n	む → み方、 く → き方 (and others)	食べる → 食べ方
n2adj-lrule	ordinary_noun_head -> adj	add 的	本 → 本的
n2predadj-lrule	ordinary_noun_head -> adj	add 的	本 → 本的
n2i-adj-lrule	ordinary_noun_head -> adj	add らしい	本 → 本らしい

Inflectional rules that apply to verbs, which then need verbal endings (examples)

Name of inflectional rule	Change of morphological type	Attachement to ending	Example
i-lexeme-c-stem-infl-rule	(c-stem -> i-morph)	ます、ました、まして、ません、ませんでした、はじめる、ましたら、ましたらば、ながら、ましょう、よう、たい、たがる、たく、られる、なさい	読みます
i-lexeme-c2-stem-infl-rule	(c2-stem -> i-morph)	ます、ました、まして、ません、ませんでした、はじめる、ましたら、ましたらば、ながら、ましょう、よう、たい、たがる、たく、られる、なさい	行きません
i-lexeme-v-stem-infl-rule	(v-stem -> vstem-morph)	ます、ました、まして、ません、ませんでした、はじめる、ましたら、ましたらば、ながら、ましょう、よう、たい、たがる、たく、られる、なさい， た、たり、て、たら、たらば、てる、ちゃう， ありませんでした、ない、ありません、なさ、ぬ、ないで、ずに、なる、ざるをえません，なさ， う， させる、さす、られる	食べます
a-lexeme-negative-cons-stem-infl-rule	(cons-stem -> a-or-aa-morph)	ず、ありませんでした、ない、ありません、なさ、ぬ、ないで、ずに、なる、ざるをえません	読まない

Stem and inflected stem

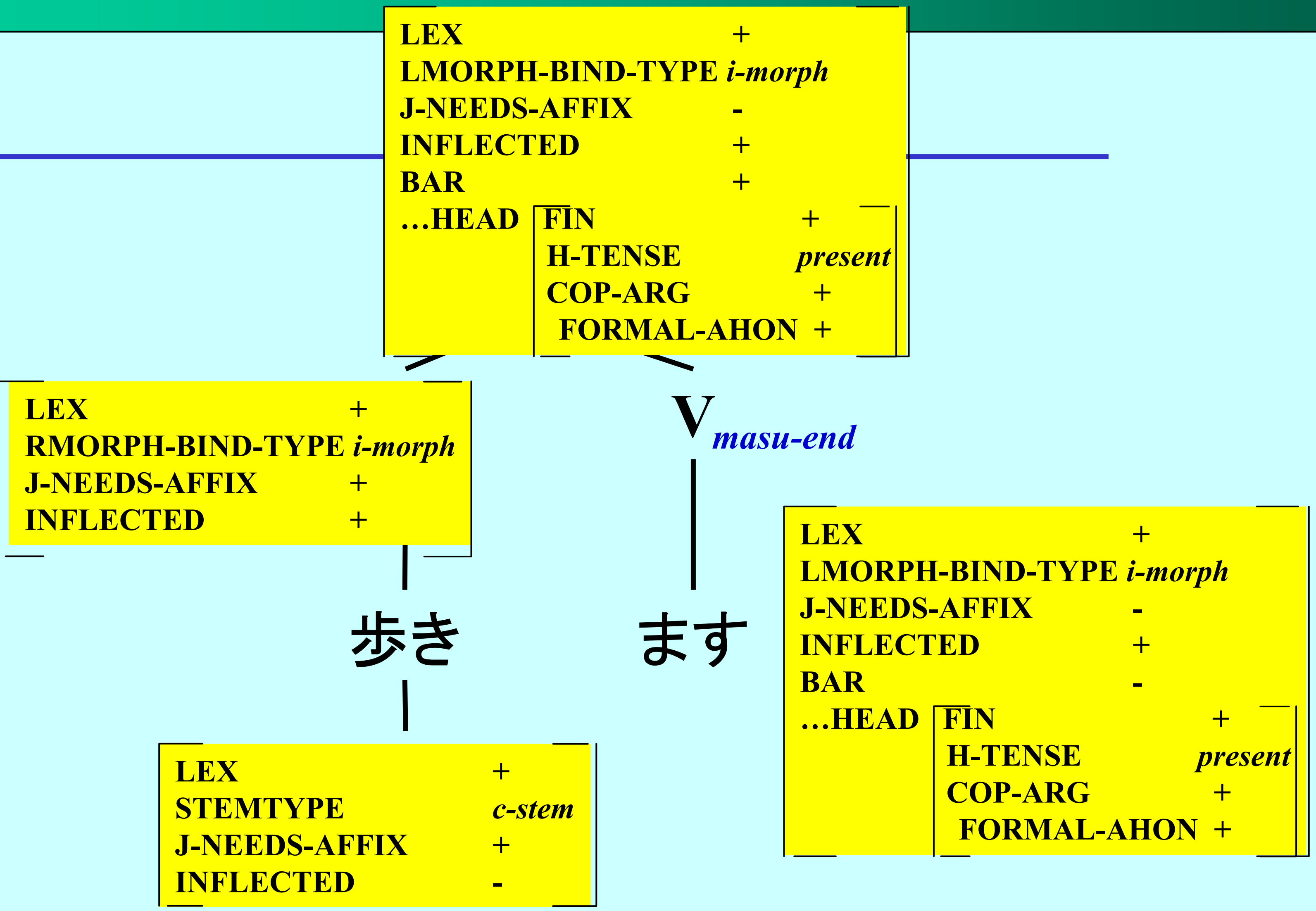
- Information on the stem:
  - LEX +
  - STEMTYPE
  - J-NEEDS-AFFIX +
  - INFLECTED –
- Information on the inflected stem:
  - LEX +
  - RMORPH-BIND-TYPE
  - J-NEEDS-AFFIX +
  - INFLECTED +

Endings

- Verbal endings are separated in ChaSen and therefore attached with a binary rule (vstem-vend, an instance of head-specifier).
- They add various information about (addressee) honorification, tense, mood, etc.
- The argument structure of the stem-ending complex comes from the stem. The ending subcategorizes for the stem (SPR).

Ending and stem-ending

- Information on the ending:
  - BAR –
  - LEX +
  - LMORPH-BIND-TYPE
  - J-NEEDS-AFFIX –
  - Head information, such as honorification, tense, fin, cop-arg, modus
- Information on the stem-ending-complex:
  - BAR +
  - LEX +
  - J-NEEDS-AFFIX –



LEX +
LMORPH-BIND-TYPE *i-morph*
J-NEEDS-AFFIX -
INFLECTED +
BAR +
…HEAD [FIN +
H-TENSE *present*
COP-ARG +
FORMAL-AHON +]

LEX +
RMORPH-BIND-TYPE *i-morph*
J-NEEDS-AFFIX +
INFLECTED +

V *masu-end*

歩き

ます

LEX +
LMORPH-BIND-TYPE *i-morph*
J-NEEDS-AFFIX -
INFLECTED +
BAR -
…HEAD [FIN +
H-TENSE *present*
COP-ARG +
FORMAL-AHON +]

LEX +
STEMTYPE *c-stem*
J-NEEDS-AFFIX +
INFLECTED -

Auxiliaries

- Japanese auxiliaries combine with verbs.
  - Provide either aspectual or perspective information or information about honorification.
- In a verb-auxiliary construction, the information about subcategorization is a combination of the SUBCAT information of verb and auxiliary, depending on the type of auxiliary.
- The rule responsible for the information combination is the *head-specifier-rule*.

Auxiliary types

- Aspect auxiliaries.
  - Treated as raising verbs
  - E.g., いる、ある
- Perspective auxiliaries.
  - Add a *ni* (dative) marked argument to the argument structure of the whole predicate.
  - E.g., くれる
  - Treated as subject control verbs.
- Obj-id auxiliaries.
  - E.g., もらう
  - Establishes a control relation between the *ni*-marked argument and the embedded subject.

Aspectual types

Example	**Aspect**
おる (oru), いる (iru), いらつしやる (irassharu)	progressive
おく(oku)	prospective
いく(iku)	inceptive
しまう (shimau)	terminative
ある (aru), ござる (gozaru)	perfective
くる (kuru)	perfect_progressive
みる (miru)	modal

Aspect auxiliaries

- Pure aspect:

ケ-キ	を	食べて	いる
keeki	*wo*	*tabete*	*iru*
cake	*ACC*	*eat*	*AUX (progressive)*

  - Add only the aspect information to the MRS semantics of the sentence.

- Aspect:

ケ-キ	が	食べて	ある
keeki	*ga*	*tabete*	*aru*
cake	*NOM*	*eat*	*AUX (perfective))*

  - Make changes to the valence of the verbal complex.
  - Attach to transitive verbs.

Aspect auxiliaries

- Complex aspect auxiliaries:

花子	が	ケ-キ	を	食べて	見る
Hanako	*ga*	*keeki*	*wo*	*tabete*	*miru*
Hanako	*NOM*	*cake*	*ACC*	*eat*	*AUX (modal: try to)*

  - Add a relation to the MRS.
  - Their ARG1 is identical to the ARG1 of the verb and their ARG2 is the handle of a proposition that outscopes the verbal relation.

Perspective auxiliaries

先生	が	私	に	本	を	買って	くれた
sensei	*ga*	*watashi*	*ni*	*hon*	*wo*	*katte*	*kureta*
teacher	*NOM*	*I*	*DAT*	*book*	*ACC*	*buy*	*AUX (subj-control)*

- **Make their ARG1 identical to the ARG1 of the verb.**
- **Add a ni-OBJ as ARG2.**
- **Link the handle of the proposition on top of the verb to their ARG3.**

Auxiliaries can add to CONTEXT

- Honorificational information:
  - *kudasaru* and *yaru* add subject honorification with positive polarity.
  - *sashiageru* adds subject honorification with negative polarity.
- Empathy information:
  - The empathy is set to ARG1 in the cases of *ageru*, *sashiageru* and *yaru* and to ARG2 in the cases of *kureru* and *kudasaru*.

Pragmatic Information is stored in CONTEXT

e.g. 食べております

$$\begin{bmatrix} \textit{ctxt} & \\ \text{C-INDICES} & \begin{bmatrix} \textit{c-indices} & \\ \text{SPEAKER} & \boxed{1} \\ \text{ADDRESSEE} & \boxed{2} \end{bmatrix} \\ \text{BACKGROUND} & \textit{*diff-list*} \\ & \left[\left\{ \begin{bmatrix} \textit{addr-honor-rel} & \\ \text{HONORER} & \boxed{1} \\ \text{HONORED} & \boxed{2} \\ \text{POLARITY} & \end{bmatrix} , \begin{bmatrix} \textit{subj-honor-rel} & \\ \text{HONORER} & \boxed{1} \\ \text{HONORED} & \boxed{3} \\ \text{POLARITY} & \textit{minus} \end{bmatrix} \right\} \right] \\ \textit{plus} & \end{bmatrix}$$

JACY MRS Semantics

- Grammar development in a multilingual context means a high premium placed on parallel and consistent semantic representations.
- Most parts of the semantic representation in the ERG were straightforwardly applicable to Japanese.
- Special treatment was, e.g., needed for
  - Nominalization and verbal nouns
  - Numeral classifiers
  - Relative clauses and adjectives

Integration of Preprocessing and a Morphological Analyzer

- Needs: Word segmentation, POS tagging, lexical coverage, shallow pre-processing
- Integration of ChaSen (Asahara&Matsumoto 2000):
  - Word segmentation as pre-processing.
  - Default lexicon entries:
    - Assign a type to words unknown to the HPSG lexicon.
    - Contains features typical to its part-of-speech.
    - Often used for names, but also nouns, adverbs, interjections, verbal nouns, verbs and adjectives.
  - Extension of the ChaSen lexicon with domain-specific entries (e.g., names in the domain of banking)
- Integration of a pre-processing tool for numbers, date expressions, email addresses, addresses, URLs, telephone numbers, currency expressions. (More modular solution in HoG)

Conclusions and Future Work

- Japanese HPSG with the following basics:
  - Precise syntactic, semantic and pragmatic information in a feature structure.
  - Combination of hand-coded lexical information with default lexicon entries.
  - Multilingual context with parallel and consistent semantic applications.
  - Efficient and robust processing.
- Future Work:
  - Application to other domains (with including more phenomena)
  - Embed in new NLP applications
  - Include stochastic disambiguation methods
  - Further enhance the discussion about Japanese HPSG with the dissemination under the open-source license.