

5/9/87  
Thurmair

METAL ADJECTIVE ANALYSIS  
=====

The following documentation starts with an overview of the old (METAL 2.2) AST- and ADJ-analysis and its problems (part 1); then, some reasons for changing this analysis are given (part 2); then, the rules involved are compared (part 3); and finally, the new scheme is described (part 4).

1. ADJECTIVE ANALYSIS IN METAL V2.2  
=====

The adjective part of the METAL V2.2 analysis system consists of rules expanding the categories

- AST-LCL (participles used as ADJs)
- AST
- ADJ-LCL
- ADJ

(The rule numbers refer to an old Munich scheme which differs from the "official" V2.2 grammar as discussed at Austin in November 86)

1.1 AST-LCL ANALYSIS

AST-LCL expands to

|  |       |                 |              |
|--|-------|-----------------|--------------|
| ZUPRED   | D     | (abzuarbeitend) | AST-LCL-1110 |
| VB   |       | (abgearbeitet)  | AST-LCL-1120 |
| VB   | D     | (abarbeitetend) | AST-LCL-1130 |
| and the recursive rule <i>→ per que' a vie vogle recursive</i> |       |                 |              |
| AST-LCL  | A-DEG |                 | AST-LCL-1140 |

AST-LCL-1140 1. does not work and 2. it shouldn't be recursive as only 1 degree element can be added.

The features available on the AST-LCL nonrecursive level are:

CL CAN PX ARGS PF TT WI WF;  
the features TA AX DG ADJCONST are set inconsistently.  
AST-LCL (recursive) in addition has OR COMDG.

(COMDG is used in ADJ-1450 only; maybe it's superfluous)  
Of these, ADJCONST PX TA WF are superfluous as they do not pass the ADJ-LCL border (see below); the others are used.

1.2. AST ANALYSIS *RECURSIVE RULES*

AST expands to

|     |       |  |          |
|-----|-------|--|----------|
| AST |       | (comparatives)                         | AST-1110 |
| AST |       | (superlatives)                         | AST-1120 |
| NUM | AST   | (fuenffach)                            | AST-1130 |
| NUM | PNCT  | AST (fuenf-fach)                       | AST-1140 |
| AST | A-DEG | (schoener) <i>per que' a recursive</i> | AST-1150 |

all these rules are recursive. AST-1110 and AST-1120 should be combined in order to be more effective in parsing. The NUMs are a special problem; they are treated as category as well as feature; a consistent solution should be found. AST-1150 is recursive, but attachment of more than 1 A-DEG has to

be blocked.

Features attached to the resulting AST category:

- from lexical entry: AST-features:  
ABB ALO CA CL DG FC GD IN MC NOADV NU NUM ORD PLC PRF SNS TA TAG
- resulting ASTs have these minus ALO plus WI WF (inconsistently) OR PO (inconsistently)
- The AST built from NUMs in addition have the features:  
PO WI WF (PO is overwritten in AST; WI WF ?)
- add OR to rule NUM-1140 (it's used in AST-1130/1140 but not set)
- feature COMDG is set in AST-1150
- feature PO should be set consistently (always to ATR and PRD)

### 1.3 ADJ-LCL ANALYSIS

ADJ-LCL expands to

|         |                         |              |
|---------|-------------------------|--------------|
| AST-LCL |                         | ADJ-LCL-1130 |
| AST-LCL | A-FLEX                  | ADJ-LCL-1160 |
|         | and the recursive rules |              |
| PP      | ADJ-LCL                 | ADJ-LCL-1170 |
| DEG     | ADJ-LCL                 | ADJ-LCL-1110 |
| NP      | ADJ-LCL                 | ADJ-LCL-1140 |
| ADV     | ADJ-LCL                 | ADJ-LCL-1120 |
| ADJ     | ADJ-LCL                 | ADJ-LCL-1150 |

In the non-recursive part, the features

ADJCONST ARGS CAN CL COMDG OR PF PX TA TT WI WF are present. The ADJ-LCLs have these minus CL ALO PO (where PO and ALO are not set), and add PO WI. In addition, ADJ-LCL-1160 adds NU GD CA IN (but ADJ-LCL-1130 does not! This can cause errors later on).

The resulting node ADJ-LCL has the feature set:

CAN NU CA GD IN TT ARGS PF WI PO PREDADJ VBARG (if consumed). PREDADJ is not represented in the dictionary and has an unclear status. VBARG is set if an NP is consumed (valency problem)

The features OR and COMDG should be carried over as they are needed in later rules.

Outside the ADJ-rules, the following ADJ-LCL features are tested:

CA NU IN GD  
PO ADJLCL PF

### 1.4 ADJ-ANALYSIS

ADJ expands to the following trees:

|         |  |          |
|---------|--|----------|
|         | non-recursive rules                      |          |
| NUM     |  | ADJ-1240 |
| NUM     |  | ADJ-1400 |
| AST     |  | ADJ-1440 |
| AST     | (compar.)                                | ADJ-1450 |
| AST     | NUM (AST ">,<,"=)                        | ADJ-1310 |
| AST     | A-FLEX                                   | ADJ-1390 |
| ADJ-LCL | (PAPL, pred.)                            | ADJ-1460 |
| ADJ-LCL | PP                                       | ADJ-1190 |
|         | recursive rules (position not specified) |          |
|         | (can applied to any value of DG and PO)  |          |
| DEG     | ADJ (bisher etc.)                        | ADJ-1350 |

|                |                |      |                       |                     |
|----------------|----------------|------|-----------------------|---------------------|
| AM             | ADJ            |      | (superl. only)        | ADJ-1370            |
| <del>NP</del>  | <del>ADJ</del> |      |                       | <del>ADJ-1370</del> |
| <del>ADJ</del> | <del>ADJ</del> |      | beide/andere:ATR only | <del>ADJ-1370</del> |
| ADJ            | PNCT           |      | (abbr.)               | ADJ-1230            |
| ADJ            | DEG            |      |                       | ADJ-1150            |
| ADJ            | NP             |      |                       | ADJ-1320            |
| ADJ            | PP             |      |                       | ADJ-1420            |
| ADJ            | PAR            |      |                       | ADJ-1170            |
| PNCT           | ADJ            | PNCT | ("rote")              | ADJ-1380            |

recursive rules for predicative ADJs only  
no specification for DG

|     |         |         |                |          |
|-----|---------|---------|----------------|----------|
| ADJ | ZUCL    |         | (predic.)      | ADJ-1220 |
| ADJ | PNCT    | ZUCL    | (predic., ",") | ADJ-1410 |
| ADJ | CLS-SUB |         | (predic.)      | ADJ-1340 |
| ADJ | CLS-SUB |         |                | ADJ-1260 |
| ADJ | PNCT    | CLS-SUB | (predic.)      | ADJ-1140 |
| ADJ | PNCT    | CLS-SUB | (predic.)      | ADJ-1280 |

rules for ADJ conjunctions  
neither PO nor DG specified (usable for both)

|      |      |      |                     |          |
|------|------|------|---------------------|----------|
| ADJ  | CONJ | NP   |                     | ADJ-1200 |
| ADJ  | PNCT | ADJ  | (",")               | ADJ-1300 |
| ADJ  | CONJ | ADJ  |                     | ADJ-1110 |
| ADJ  | PNCT | CONJ | NP ("und")          | ADJ-1180 |
| ADJ  | PNCT | CONJ | ADJ ("und")         | ADJ-1250 |
| CONJ | ADJ  | CONJ | ADJ (entweder oder) | ADJ-1430 |

recursive rules for treatment of comparatives only  
no spec for PO (except 1120)

|      |      |      |                   |          |
|------|------|------|-------------------|----------|
| CONJ | ADJ  |      | ("desto", "umso") | ADJ-1290 |
| ADJ  | CONJ | NP   | (predic.!, "als") | ADJ-1120 |
| ADJ  | CONJ | LCL  |                   | ADJ-1360 |
| ADJ  | PNCT | CONJ | LCL ("als")       | ADJ-1130 |

ASTs have the features mentioned above. The nonrecursive ADJ nodes build the following feature set:

- ADJs built from NUMs:  
GD CA IN PO DG NU WI WF ORD NUM TY (partly)  
((missing: FC MC PF AX ABB OR!)) ((not used TY WF))
- ADJs built from ASTs:  
GD CA IN PO DG NU WI WF FC MC NOADV NUM ORD PLC PRF SNS TA TAG OR (partly)  
((missing: AX ABB OR!)) ((not used: NOADV PLC PRF SNS TA TAG))
- ADJs built from AST + NUM:  
GD CA IN PO NU WI OR NUM  
((missing: DG FC MC PF AX ORD ABB))
- ADJs built from AST + AFLEX  
GD CA IN PO DG NU WI FC MC NOADV NUM ORD PLC PRF SNS TA TAG OR  
((missing: PF ABB AX)) ((not used: NOADV PLC PRF SNS TA TAG))
- ADJs built from ADJ-LCLs:  
GD CA IN PO NU WI TT ARGS PF PREDADJ VBARG COMDG OR  
((missing: DG MC FC AX NUM ORD ABB))  
((not used: TT ARGS PREDADJ VBARG COMDG))

These features have to be standardised!

In tests of other rules, the following features are used:

- Within ADJ-rules, the following features are used in tests:  
CA NU IN GD

PO DG  
WI ABB NUM  
FC MC PF COMP  
ADJCNJ ADJLCL PP-ADVCOMP NPCOMP AJTCAN

(feature BOTH is accessed without having been set)

The following features are set additionally:

ADJCNJ AJTCAN FORMULA COMCON COMP PP-ADVCOMP PNM ADJLCL MADJ CONJCU NPCOMP WI  
(CONJCU is never used; PF ZU is never set; AJTCAN is inconsistent; PNM and  
ADJLCL are used as NO-features but never carried over)

- Outside ADJ-rules, the following features are tested:

CA DG NU PO PF (CLS-1240, RCL-1290) NUM IN (DET-1120) ORD (NO-1280, PP-1210)  
MADJ (NO-1280) ADJ-CONST (NP-1650) FORMULA (NP-1580) OR (NP-1610, NP-1210)  
ADJCNJ (NP-1210) AX (RCL-1290, S-1280) AF (RCL-1290) ADJLCL ()  
(AF is never set).

- As a result, the following features have to be provided with ADJ-nodes:

(1) CA NU IN GD

PO DG

FC MC PF AX (AF)

WI OR

NUM ORD ABB

(2) COMP MADJ ADJLCL ADJCNJ ADJ-CONST PP-ADVCOMP NPCOMP AJTCAN FORMULA

The non-recursive ADJ-rules have to make sure that the feature set (1) is  
usable (features 2 are set by the rules themselves).

## 2. BASIC PRINCIPLES OF CHANGES

=====

### 2.1 ELIMINATION OF SPECIAL CATEGORIES FOR PARTICIPLES

In METAL V2.2, as described, were two different categories for adjective constituents: ADJ (for pure adjectives) and ADJ-LCL (for adjectives derived from verbs; inflected participles), and AST and AST-LCL, respectively.

They had different feature sets, especially in the treatment of complements:

- ASTs had features like FC and MC
- AST-LCLs had features like ARGS and did framing and CLSXFR-operations (which never worked properly, esp. ADJ-LCLXFR).

Accordingly, all the rules handling adjectival constituents were written for ADJs as well as for ADJ-LCLs - a source of many inconsistencies and unclear coverage.

From the point of view of morphology, participles are verb derivatives; but in their syntactic function and distributional behaviour, they can be treated just like adjectives, as shown below.

The differences within the participle categories are more important: present ("laufend"), present passive ("abzugehend") and past ("getestet") participles behave differently with respect to complement behaviour: PRPLs can have all the complements they have as verbs ("der ihm das Buch gebende Mann"), PAPLs can have a NP-complement only if they are ditransitive ("das ihm von ihr gegebene Buch"), PRES-PASS-PARTs cannot have NP-complements. Beyond that, the information of the ARGS feature is not used in METAL 2.3 because of framing problems.

#### 2.1.1 Distributional Behaviour

- ADJs as well as ADJ-LCLs can have inflections and degree information  
der bekannteste Spieler ist Mueller
- both can form an "absolute superlative"  
am schoensten / aufgeregtesten
- both can be used in attributive, predicative and adverbial function (except PRES-PASS-PARTs)  
das Programm wird getestet/schnell ausgeliefert  
das schnelle/getestete Programm  
der Himmel bleibt grau/bedeckt
- both can be used as NP heads  
das Schoene ist das Vergessene
- outside NPs, ADJs and ADJ-LCLs have the same distribution as well; this could be seen from the duplications of rules, like LCL -> ADJ LCL vs. LCL -> ADJ-LCL LCL, RCL -> RCL ADJ vs. RCL -> RCL ADJ-LCL, etc.

#### 2.1.2 Valency Behaviour

- normal ADJs can have valencies just as participles:  
dem Kind fremd - seiner Sache sicher - auf etwas stolz etc.
- they also can have more than one complement (which the present FC/MC-mechanism does not allow):  
ihm dafuer dankbar - ihm etwas schuldig (HFM)
- they can have PPs and ADVs in rather free attachment, e.g.  
das unter diesen Umstaenden schnell verarbeitete / feste Puddingpulver

#### 2.1.3 practical problems

- there were gaps in the coverage of the rules: ADJs could not

take complements that ADL-LCLs could. If these inconsistencies had been removed, the PSG set of rules would have been just identical

- all kinds of conjunctions between ADJs and ADJ-LCLs are possible and have to be covered by rules:

der freche/unausgesprochene und beleidigte/inkompetente Mensch

#### 2.1.4 Consequences

The proposal for METAL 2.3 is as follows:

- Only one category AST and ADJ; AST-LCL-rules and ADJ-LCL-rules have to be removed, AST and ADJ rules have to be adapted
- a new rule (AST -> VB) takes care of participles; participles are known by the (PK) feature; it can have the values PRPL, PAPL, PRES-PASS-PART.
- Well defined feature sets attached to these two categories
- common control of possible ARGS for ADJs (this was not done in V2.3 as the global decision about framing was missing). In 2.3, MC/FC features in the Transfer part cannot be handled (as it is not known to which German nodes they refer; currently, just the first nominal node is taken)
- review of the rule package / removal of errors etc.
- complements of ADJs must be allowed beyond the ARGS or just MC/FC information (free PPs, but also some NPs: "das 4 Meter lange Schiff", "der den ganzen Tag lustige Mann"). This leads to problems with competitive rules applications.

NEW RULE  
NEW FEATURE?

## 2.2 STANDARD STRUCTURE OF ADJ-TREES

Adjectives can have the following complements:

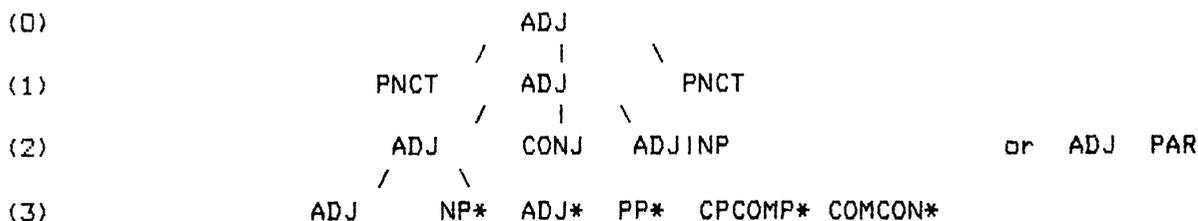
- NPs: der (ihm)treue / vertrauende Soldat
  - PPs: der auf ihn stolze / wartende Mann
  - ADVs: das gestern abgeschlossene / zufriedene X
  - MODs: das sehr gute / geloeste X
  - ADJs (predicative), but only with PKs (participles): das schoen bleibende Wetter
  - CLS-constituents: ZUCLs (sicher zu tun), SUB-CLSs (fraglich ob X), but only in predicative use
  - comparative complements: schoener als X, also only in predicative use
- Then there are conjunctions of ADJs with ADJs (schnell und gut) and of ADJs with NPs (schlau und ein guter Rechner).

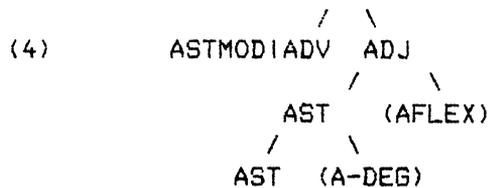
As all these complements are attached recursively, there are two consequences:

1. We have to control the order of rule application to exclude incorrect analyses like "\*fraglich ob X unter diesen Umstaenden": PP attachment must precede CLS attachment. To do this, features like PPCOMP, NPCOMP etc. are introduced.
2. We have to set up a flat structure for framing once the ADJ constituent has been built. This means to apply flattening transformations and has some nasty consequences.

As framing is not done in V2.3, we must know where to put which constituent by ordering the complements of the ADJ.

The result is the following standard structure for ADJ constituents (numbers are excluded here):





(its a little more than two-bar ADJ; but thats real life).

Explanation:

- level 0: This level is used outside (in NO-rules, in other rules taking ADJ-constituents).
- level 1: This level considers PNCTs like quotes etc.; also parens, hyphens etc (not yet in V2.3). It means that PNCTs include maximal projections of ADJ nodes
- level 2: Here, conjunctions are treated (with ADJ or NP head but in ADJ function). PARs (like "das rote (gruene) Auto") are treated as conjunctions in V2.3. CONJ treatment see below.
- level 3: Here, all ADJ complements are collected. Within this level, COMCONs and CPCOMPs (= all CLS-complements) are attached AFTER the other complements, and only for ADJs with PO PRD. Within the PP complements, a possible passive subject PP ("das von ihm gekaufte Buch") is put as the first PP node.  
In English, if this level is present, the whole constituent will be flipped behind the head (see NP-Documentation).
- level 4: Here, the modifications are treated. ADJs take ASTMODs here ("das sehr gute Buch" ~ "die ausserordentlich gute Qualitaet": here, an ADV -> ASTMOD) whereas participles take ADVs ("das sorgfaeltig getestete Programm" - "das sehr getestete Programm": here, ASTMOD -> ADV). Not all ADVs can take this place: no complex ones (see ADV rules), not all ADJ-derivations.  
The difference between ASTMOD / ADV complements and NP / PP etc. is that they do not cause the whole ADJ constituent to be flipped. (?? This is a target-specific analysis!)  
As the word order in German is rather free ("der sehr von ihm abhaengige X" vs. "der von ihm sehr abhaengige X"), the ADVs have to be moved to their place by transformations.
- levels below: this is morphology (FLEX-attachment to AST, DEGREE-attachment to stems). Recursive application of AST -> AST A-DEG is blocked.

The whole structure is controlled by the -COMP features. The derivation starts out with the NOCOMP feature (no complement) = no level 3. Every derivation of level 3 must not have a ADJCNJ / ADJPARG feature (assigned to level 2), etc. If an NPCOMP, PPCOMP etc. ist present, the NOCOMP feature is deleted (= level 3 exists). Transformations move the new constituent to their place.

### 2.3 TREATMENT OF CONJOINED ADJECTIVES

Conjunctions of ADJs must deal with maximal projections ( not "ihm (vertrauende und gute)" but "(ihm vertrauende und gute)"); therefore, framing of the ADJs involed has to be carried out (the right constituent is framed only if it is not ADJCNJ itself) (in V2.3, only PKs are framed, and the whole procedure is very nasty!). They can have different heads, but only if the ADJ is PRED ("er ist schnell und ein guter Schwimmer"). Real ADJs as well as PKs can be involved. The analysis structure for the conjunction is shown above. In METAL 2.2, there was still another solution for CONJs: not just tree (2) below but also tree (3). This solution increases the number of interpretations and conjoines different things, because what is conjoined is



| |  
fehlerfr. Programm

This means that the whole CONJ constituent is destroyed and grouped according to the criterion of complexity.

There are two major drawbacks of this solution:

1. The scope relationships are completely destroyed (which might be relevant if restrictive and nonrestrictive readings of ADJs are mixed; if there are just restrictive readings then there is no change)

2. The CONJ element itself is sometimes eliminated, like in  
das schoene und von ihm getestete Programm ->  
the fast program tested by him

This leads to problems if the CONJ is semantically relevant ("das schoene aber von ihm getestete Programm") or if the CONJ has two elements ("das weder schoene noch von ihm getestete Programm").

The only solution I can see is to form a relative clause from these complex ADJs using a copula. The problem is not to determine the number of the copula but its time ("the program which is?/was? neither fast not tested by him"). This is not done in METAL V2.3 but could be easily added. Currently, these constructions can result in errors in the analysis.

Another problem is that all this is done in the CONSTR part of the rules although it belongs to the ENGLISH part (?).

### 2.3 ADJECTIVES IN NP-APPPOSITIONS AND IN PREDICATIVE POSITION

consider the following sentences:

Mueller, frei von Sorgen, schlief

Mueller, von Sorgen frei, schlief

Mueller, gestochen von Muecken, erwachte

Mueller, von Muecken gestochen, erwachte

(1) and (2) have to be treated via ADJ rules, (4) is handled by NFCL rules, (3) was not handled at all in METAL 2.2 and is now also handled by ADJ rules. Intuitively, these appositions are kinds of non-finite constituents and should be treated the same way.

There are two ways to do this:

1. Relaxation of the NFCL rules so that they

- can take ADJ heads
- can expand to the right

2. Treatment of all these cases as ADJs.

Proposal (1) gives a nice generalisation, and during the Austin workshop October 86, a proposal was made to treat the ZUCLs, NFCLs and ADJ-LCLs all the same way (Roberts idea).

The reason to do so was:

- they all have basically the same constituent structure
- they are sentential complements somehow
- they can use the same framing mechanism (with the possibility of missing constituents)

This interferes with the treatment of ADJ-LCLs as ADJs as shown above: ADJ-LCLs behave like ADJs with respect to

- inflectional behaviour
- taking (obligatory and optional) complements
- attributive and predicative position
- adverb formation ("er laeuft schnell" - "er arbeitet gezielt")

etc.

The next step would be to collapse the two sets of problems and build just one kind of structures: NFCLs. This can be done by leading the ADJ-LCLs into ADJs (via AST -> VB / VB D rules above), then ADJs into NFPREDS, then to NFCLs, and after that into NOs (with: NO -> NFCL NO, where in case of a participle a frame test is performed, in case of an ADJ some equivalent)

The idea of building just one type of rules for ADJs, ADJ-LCLs, NFCLs and ZUCLs, however, does not seem to work well:

- the rules (eg. context macro in NFCL -> NFPRED) would be extremely weakened, as
  - NFCLs could attach components also on their right (if predicate position or apposition is considered: "X ist bekannt fuer Y", "X, fuer Y geplant, ..."): no context can be specified as the NFPRED can even be surrounded by complements ("X, gestern geplant fuer Y, ..."). This could be covered by rules like NFCL -> NFCL PP (attachment on the right), but the fact is that
  - different types of word ordering have to be considered simultaneously by the same rules.

A study of the behaviour of elements has to be postponed for time reasons. (If there are extrapositions in NFCLs, we need a rule like NFCL -> NFCL PP anyway: "Er hat einen Unfall gehabt gestern nacht"). That's a big topic.

- different information has to be carried over depending on the different functions of the NFCLs:
  - inflectional information, reordering information, position, degree etc. are needed for NO rule forming NFCLs
  - verbal information has to be carried for "real" sentential purposes of a NFCL
- different parts of rules have to be applied depending on the use of a NFCL: NFCL -> ZUCL NFCL does not hold for prenominal modifications, whereas modification with a DEG is impossible for "real" NFCLs. This information has to be carried along.

Therefore, the proposal is to separate the NFCL / ZUCL on the one hand, and the ADJ / ADJ-LCL on the other hand.

The disadvantage is, of course, that the interrelations between NFPREDS and ADJ-LCLs cannot be expressed any more:

|                  |   |                              |                              |
|------------------|---|------------------------------|------------------------------|
| "A schlaeft"     | - | -                            | (intransitive nonperfective) |
| "A verblueht"    | - | "das vebluehte A"            | (intransitive perfective)    |
| "A schlaegt B"   | - | "der von A geschlagene B"    | (transitive)                 |
| "A gibt dem B C" | - | "das von A dem B gegebene C" |                              |

etc.

The solution would be to have NFCLs (merging NFCLs and ZUCLs) on the sentence level and ADJs (merging ADJ-LCLs and ZUCLs) on the nominal level. The ADJ-rules would be:

ADJ -> NP | PP | ADV (!) | ADJ | PRT | CON + ADJ

ADJ -> ADJ + PP | PRT | ADV

and they would form structures with the ADJ (or participle) as leftmost constituent, followed by a flat structure. For participles, a frame test should be carried out.

### 3. RULES FOR ADJECTIVES IN METAL V2.3

=====

#### 3.1 RULES FOR AST CONSTITUENTS

##### 3.1.1 Phrase structure part

ASTs are built:

- from the dictionary
  - from ASTs themselves (AST-100; Maybe superfluous)
  - from verbs (then they have a PK feature; within the VB analysis, all PRPL and PRES-PASS-PART morphology has been done)
  - from NUMerals (numbers or ASTs like "vierundzwanzig")
- recursive AST rules:
- just AST can take an DEGREE suffix

##### 3.1.2 Features

Features are copied from ASTs and VBs (via CPX). All these features are documented in the coding manual (except PK).

a) Obligatory features:

WI WF OR

CAT CAN CL

b) Additionally, there are different feature sets depending on the kind of ASTs:

- Features derived from lexical ASTs:

TA MC FC

ABB ORD NUM NOADV

CA NU GD IN PO DG (if present in the lexicon)

- Features derived from verbal ASTs: They must have a PF feature!

PF TT PX AX ARGS

ADJ-CONST is added and sent down for verb transfer

c) features tested inside AST rules:

CL WI CMPD PK NUM ORD TA OR

d) features added at AST level:

DG CL PO PK NU ORD CMPD ASTCNJ

e) others categories features tested:

- VB: PK SPX PX

- A-DEG: DG

- NUM: WI WF

- CONJ: CAN WI WF CORR

f) AST-features tested outside: (see ADJ)

- in ADVs: WI WF PLC NUM CAN NOADV DEG PK ORD

- in NSTs: OR NUM CAN WF

#### 3.2 RULES FOR ADJ CONSTITUENTS

##### 3.2.1 Phrase structure part

a) non-recursive rules

ADJs can be formed from:

- AST

- AST A-FLEX

- NUMs

b) recursive rules

ADJs can expand differently in PRD or ATR case

- rules for ATR and PRD:

ADJ -> NP | PP | ADJ | ADV | ASTMOD ADJ  
- rules for PRD only:  
ADJ -> ADJ NP | PP | ADV | ZUCL | CLS-SUB | COMCON  
- conjunction rules:  
ADJ -> (CONJ) ADJ CONJ ADJ | NP

### 3.2.2 Features of the Adjective nodes

obligatory are: NU CA GD IN PO

a) ADJ-node features: ADJ get features from:

A-FLEXs: NU CA GD IN PO WF

ASTs: everything; tested are:

CL NU CA GD IN PO DG PK CAN WI OR NUM WF

PK PV(132) ORD(340)

NUMs: NUMSUF PERCENT PURE-NUM MTH DATE

ASTMODs: INT

b) ADJ-node features: new on ADJ-nodes (= set within ADJ-rules)

NOCOMP ABSSUP ADJMOD NPCOMP PPCOMP ADJCOMP COMPLADJ SHORTTU (125)  
PASSUBJ(132) CPCOMP ADVCOMP NUMODCAN(194) COMCON NUMOD  
PNM MIXED (300) ADJCNJ PREPCAN(340) QADJ(520) FORMAT(521) UNK(530)  
BOTH

These features mean:

Control of complements of ADJs: no complement (NOCOMP), ADJ/ADV/PP/NP-  
complement (ADVCOMP, ADJCOMP, NPCOMP, PPCOMP), sentential complement  
(CPCOMP), comparative construction (COMCON), modification via  
ASTMOD (ADJMOD), subject of passive participle construction (PASSUBJ)  
Control of complexity / conjunction: COMPLADJ (ADJ with NPCOMP or PPCOMP  
or ADJCOMP), PNM (simple ADJ: prenominal modification), ADJCNJ (ADJ  
conjunction), MIXED (conjunction formed from PNM and COMPLADJ)  
Control of Numerals: NUMOD (modified NUMERAL, via NUMOD), NUMODCAN (its  
CAN-form), PREPCAN (for NUMs with PREPs inside: their CAN)  
Others: ABSSUP absolute superlative ("am besten")  
SHORTTU defective clause ("Daten eingegeben")  
QADJ quoted ADJ  
FORMAT format description (the ALO of the format)  
UNK unknown ADJ  
BOTH construction with "beide" (a DET!!)  
ANOTR construction with "ander-"

c) ADJ-features tested: inside ADJs

DG CA NU IN GD FC MC OR PK NUM PO TT TA(125) NPCOMP  
ADJCNJ ADJPAR CPCOMP COMCON NOCOMP PPCOMP CAN

ADJ-features tested: outside ADJs

DG CA NU TA (ADV-103) PK PO CAN (VB-177) OR (NP-390) ORD (PP-120)  
AX (RCL-130) AF (LCL-130) PF (CLS-130)

d) other CAT's features tested in ADJs:

ASTs: WI WF ABB NM CL DG ALO

PNCT: WF CAN ALO(520)

NUM: WI ORD

ASTMOD: PO CAN

NP: TY

PP: CAN (prep for MC)

PREP: CAN(340)

PAR: KD  
 CLS-SUB:MK  
 ADV: ADV-FROM-PP  
 ZUCL: NFCLCNJ MK  
 NUMOD: CAN  
 COMCON: COMTYPE  
 CONJ: CU CAN

other CATs features set in ADJs:

NP: MEANP (125) AGENP (125) (measure NP, age NP)  
 PP: PASSUBJ (passive Subject: "von X gelesene y")

e) macros used in ADJ:

test-hyphen ALO-number? date? convert-date decimal-convert  
 conj-check(220) guess-cat

f) Features sent down in the English part and their origins

|           |   |
|-----------|---|
| ADJ-CONST | informs the generation that a participle is b |
| ISTO      | (?)   |
| TY        | sem-type; from case framing                   |
| DG        | degree  |
| NU        | (inconsistently; from NP-level)               |
| PO        | (from NP-level (?))                           |

features copied in the English part:

MC ON PO

### 3.2.3 ADJs outside NOs

PP-120 PP -> PREP ADJ hielt X fu

The following rules are eliminated:

NP-104, NP-105 : used for numbers, formulae etc.: now -> NP  
 NP-180, NP-190 : can have ADJs: now NO->ADJ = NO-180  
 NP-185, NP-187, NP-195 : special rule (NP-125) for NADJs (NP -> DET NO)  
 NP-240 : is a clause rule with NFPRED  
 NP-390 : is a gapping rule; solved via NP -> NP NP  
 NP-537 : for units/measures: NUM-rule

PP-150 : part of PP-120 (PP-120 may be superfluous)

### 4.2 Other nonsentential categories with ADJs (have not been checked yet)

|      |    |      |     |               |              |
|------|----|------|-----|---------------|--------------|
| ADV  | -> | AM   | ADJ |               |              |
| ADV  | -> | ADJ  |     | replaces ADV- | (-> AST      |
|      |    |      |     | and ADV-      | (-> ADJ-LCL) |
| COMP | -> | COMP | ADJ | (?)           |              |
| DET  | -> | PREP | ADJ | (??)          |              |
| PAR  | -> | ADJ  |     |               |              |

### 4.3 Sentential categories with ADJ

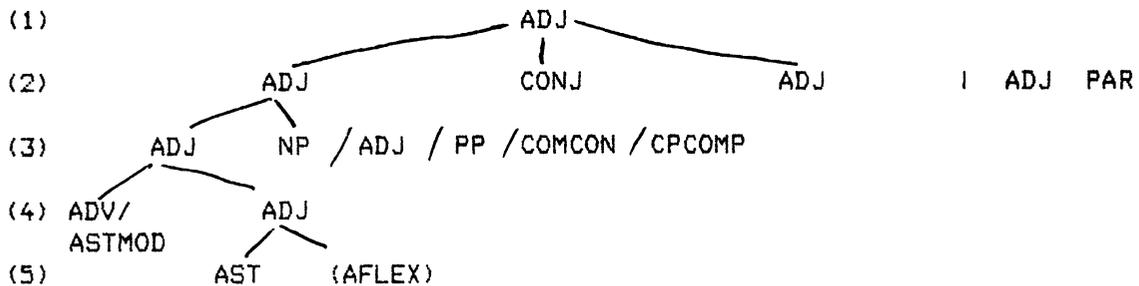
(these rules have not been checked yet: but all ADJ-LCLs and all ZUCLs are eliminated)

|          |      |    |      |      |      |     |     |  |
|----------|------|----|------|------|------|-----|-----|--|
| CLS-125  | CLS  | -> | ADJ  | RCL  |      |     |     |  |
| CLS-420  | CLS  | -> | CONJ | PNCT | ADJ  | RCL |     |  |
| CLS-425  | CLS  | -> | CONJ | ADJ  | PNCT | CON | RCL |  |
| LCL-130  | LCL  | -> | ADJ  | LCL  |      |     |     |  |
| NFCL-130 | NFCL | -> | ADJ  | NFCL |      |     |     |  |
| RCL-130  | RCL  | -> | RCL  | ADJ  |      |     |     |  |
| S-160/1  | S    | -> | ADJ  |      |      |     |     |  |
| S-530    | S    | -> | CLS  | PNCT | ADJ  |     |     |  |
| S-912/4  | S    | -> | CLS  | PNCT | CONJ | ADJ |     |  |
| S-951    | S    | -> | NP   | ADJ  |      |     |     |  |
| ZUCL-130 | ZUCL | -> | ADJ  | ZUCL |      |     |     |  |

FINAL DOCUMENTATION

GERMAN ADJECTIVE ANALYSIS

1. CANONICAL STRUCTURE OF AN ADJ-PHRASE

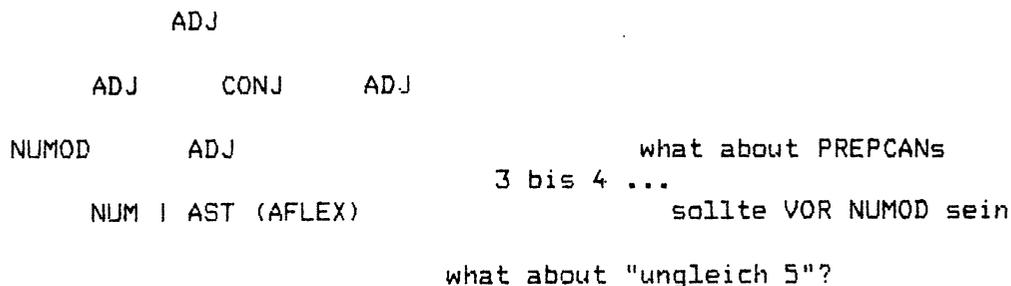


Control via the -COMP and -CNJ/-PAR - features. COMCON and CPCOMP are valid only for PRED ADJs and are attached after the NP/PP/ADJ-COMPs.

This structure does not hold for the following rules:

- ADJ-103 (ABSSUP) should be extra
- NUMs cannot have COMPs, but NUMODs (194) and CONJs (340/345)
- ADJ-220 (umso) must be like an ASTMOD at the moment
- ADJ-390/395 (CONJ with different heads) not properly set up yet
- ADJ-540 is not an ADJ at all
- ADJ-550 is a DET-problem

Structures for NUMs:



3. MAIN PROBLEMS:

- framing for ADJs (PKs as well as FCs)
- MC-treatment
- CONJ-treatment
- umso schneller, je ....
- tuning: es gab (zu dieser Zeit lange) Gesichter

echte FCs vs freie FPs als MCs (boost)

- 4.3 Features not used any more are:
- |            |   |
|------------|---|
| PREDADJ    | should be a lexical feature but isn't                                       |
| VBARG      |   |
| COMDG      | replaced by the respective DG value   |
| CONJCU     | was never used  |
| AJTCAN     | is now more general CONJCAN   |
| ADJLCL     | was used confusingly  |
| FORMULA    | is part of the NUM-rules  |
| AF         | was always unclear  |
| COMP       | are (incorrectly) used for ADJ-complements:                                 |
| NPCOMP     | now in NFCL-analysis  |
| PP-ADVCOMP |   |
| PNM        | prenominal modifier: now to control word order of<br>complex NO-complements |