



Dictionary Entry Parsing

Lothar Lemnitzer, Claudia Kunze

`lothar@sfs.uni-tuebingen.de`, `kunze@sfs.uni-tuebingen.de`

Computational Lexicography at ESSLLI 2005



- What is Dictionary Entry Parsing ?
- The Structure of Dictionary Entries
- Problems of Parsing Entries
- Architecture of the *LexParse* Parser
- Resources



What is Dictionary Entry Parsing

Dictionary Entry Parsing

- takes entries of print dictionaries as input
- segments the entries
- classifies the segments according to their function
- converts the entry into a tree-like or an sgml presentation



Why parsing dictionary entries?

- Printed dictionaries are considered to be a store of valuable linguistic information
- They have been re-used as a source for lexical databases in NLP
- Some print dictionaries have been converted into electronic dictionaries (electronic publishing)



The structure of dictionary entries

black [blæk] 1 *adj* (+*er*) (a) schwarz. ~ man/woman Schwarze(r) *mf*; ~ and blue grün und blau; ~ and white photography/film Schwarzweißfotografie *f*/film *m*; to swear that ~ is white schwören, daß zwei mal zwei fünf ist; the situation isn't so ~ and white as that die Situation ist nicht so eindeutig schwarz-weiß; a western makes things ~ and white ein Western stellt alles in Schwarzweißmalerei dar.

(b) (*dirty*) schwarz.

(c) (*wicked*) *thought, plan, deed* schwarz. he's not so ~ as he's painted (*prov*) er ist nicht so schlecht wie sein Ruf.

(d) *future, prospects, mood* düster, finster. he painted their conduct in the ~est colours er malte ihr Betragen in den schwärzesten Farben; things are looking ~ for our project es sieht für unser Vorhaben ziemlich schwarz or düster aus; maybe things aren't as ~ as they seem vielleicht ist alles gar nicht so schlimm, wie es aussieht; in ~ despair in tiefster Verzweiflung; this was a ~ day for ... das war ein schwarzer Tag für ...

(e) (*fig: angry*) looks böse. he looked as ~ as thunder er machte ein bitterböses Gesicht; his face went ~ er wurde rot vor Zorn.

(f) (*during strike*) to declare a cargo etc ~ eine Ladung etc für bestreikt erklären; ~ goods bestreikte Waren.



The structure of dictionary entries

Most dictionary entries bear the following structural characteristics:

- They consist of information items (e.g. part of speech, equivalent(s))
- Information items serve the function to provide information about the headword
- Some of the information items are optional
- Some of the information items are grouped



The structure of dictionary entries

Structural relations between information items in dictionary entries

- Linear precedence: some information items precede / follow others
- (Immediate) dominance: some higher nodes dominate groups of information items (grammar → part of speech, inflection)



Additional difficulties

- Implicit information must be made explicit (e.g. gender → part of speech)
- Abbreviations must be resolved (e.g. the tilde symbol representing the headword)



Structure indicators

- Structure indicators are essential to the format of dictionary entries
- They mark the beginning and end of information items ('fields')
- Punctuation and other symbols are used as *nontypographic structure indicators*
- Fonts and typefaces are used as *typographic structure indicators*



Elements of a dictionary entry grammar

- The dictionary entry grammar guides the analysis of the entries
- It defines the set of well-formed entries
- A dictionary entry grammar is a quadruple (CEI, CNI, R, WA)
- CEI=terminal alphabet; CNI=non-terminal symbols, R=set of rules; WA=initial symbol



Tasks of the *LexParse* parser

- Split any (standard) entry of any dictionary into segments
- Reconstruct the hierarchical structure of the entry
- Resolve abbreviations and make all information explicit
- Report on malformed entries
- Represent the data in a well-defined format (e.g. sgml, database records)

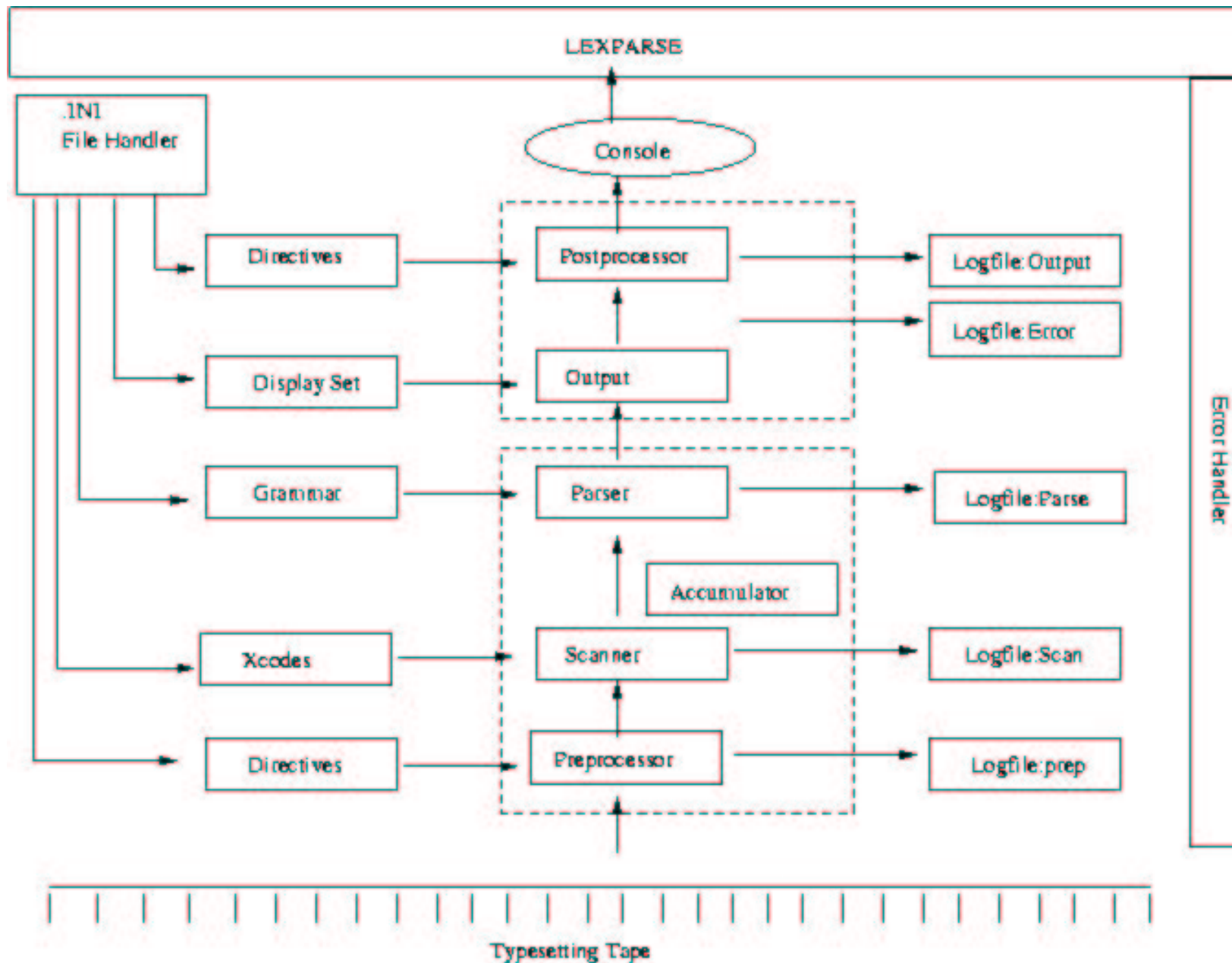


Parser configuration and input

- General: executable programme, configuration file
- Specific: dictionary entry grammar, dictionary data



Architecture of LexParse





Directives for preprocessing

- Directives prepare the input file or typesetting tape for the parser
- Delete superfluous lines and patterns
- Convert patterns into XCodes



- Xcodes reflect structure indicators
- Typeface information, brackets and some special characters should be converted into XCodes
- e.g.: Cat → XBRPO, *, XBRPC (a category is expanded to a string enclosed by brackets)
- ambiguous cases are resolved by treating the majority of cases correctly or by defining sub-patterns (e.g. for the semicolon)



- The grammar is a set of rewrite rules
- non-terminal symbols are expanded to sets of terminal and non-terminal symbols
- e.g.: $WA \rightarrow FK, SK$
- LexParse style: $WA \rightarrow XFLBE, FK, SK, XFLEN$



Display options

- Specifies the format of the output
- Options are: SGML, Tree, Map



Directives for postprocessing

- Directives clean the output
- Delete superfluous lines and patterns
- Convert patterns (e.g. German Umlaute)



Conclusion

- *LexParse* is a general parser for (standard) dictionary entries
- *LexParse* deviates in some respects from a general language parser, since the language of dictionary entries is special
- *LexParse* prepares the data for subsequent formal processing (e.g. in a lexical database)
- *LexParse* provides error reports and is therefore useful for consistency checking