Descriptive morphological analysis in Montage:

Developing a tool for the morphological analysis of underdescribed languages

Jeff Good, MPI EVA
(good@eva.mpg.de)

Emily M. Bender, Univ. of Washington
(ebender@u.washington.edu)

Presentation to be posted at:
http://depts.washington.edu/uwcl/Montage/

Outline of talk

- Give background Montage project
- Discuss the Montage model for morphological annotation and search
- Relate that model to existing proposals and tools in grammar engineering and computer-assisted linguistics

Background: Montage

- Suite of tools to assist in the documentation of underdescribed languages (Bender et al. 2004)
- Focus on grammar (especially morphology and syntax)
- Integrate with other initiatives building tools for transcribed texts and lexicons (e.g., ELAN, FIELD, AGTK)

Background: Montage

- Overarching goal: Allow the "ordinary working linguist" (or OWL) to make use of sophisticated grammar engineering tools without being grammar engineers themselves
- This talk: Morphological annotation, searching, and collating (morphology broadly construed)

Background: Montage

- Some research within Montage
  - Good (2004) on developing a model for descriptive grammars in XML
  - Bender and Good (to appear) on a morphological model for Montage
  - Drellishak and Bender (2005) on coordination within the Grammar Matrix

Background: Montage

- Other research
  - Search parameters (with Laurie Poulson)
  - Work on an implemented Slave grammar (with Anya Dormer)
  - Using Wiki-based environments for grammar writing as a prototype for a general annotation system

Annotate and analysis

- Rest of this talk
  - Brief discussion of “ideal” search
  - Annotation
  - Ontologies
  - Search and parsing in Montage
- We’re actively seeking feedback on the design as well as people interested in serving as advisors

Ideal search

- Give me all examples collected yesterday
- Give me all example sentences containing annotation for past tense
- Give me all examples that *could* contain a past tense based on what I've said I know about past tense

Ideal search

- Give me all examples of sentences annotated as being conditionals not matching my defined conditional template
- Give me all examples of sentences containing transitive verbs with only one overt argument

Towards the ideal

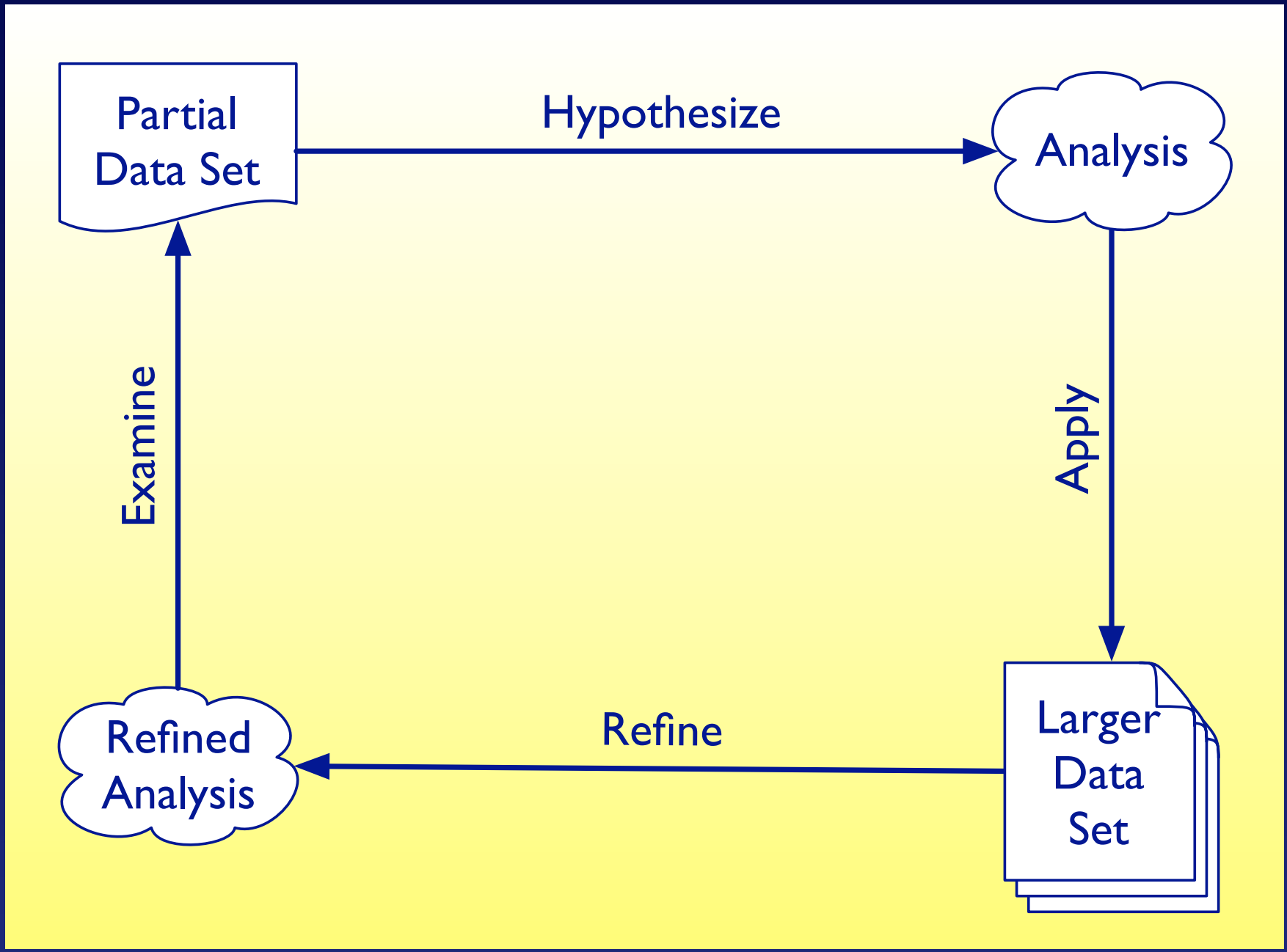
- Three steps
  - User-driven annotation
  - Ontological linking
  - Standard and parser-based search

Annotation

Workflow

- (Partial) Workflow model in language description
  - Develop preliminary analysis of a set sentences
  - Use those analyses to help analyze further sentences, noting discrepancies
  - Refine earlier analyses based on this larger data set

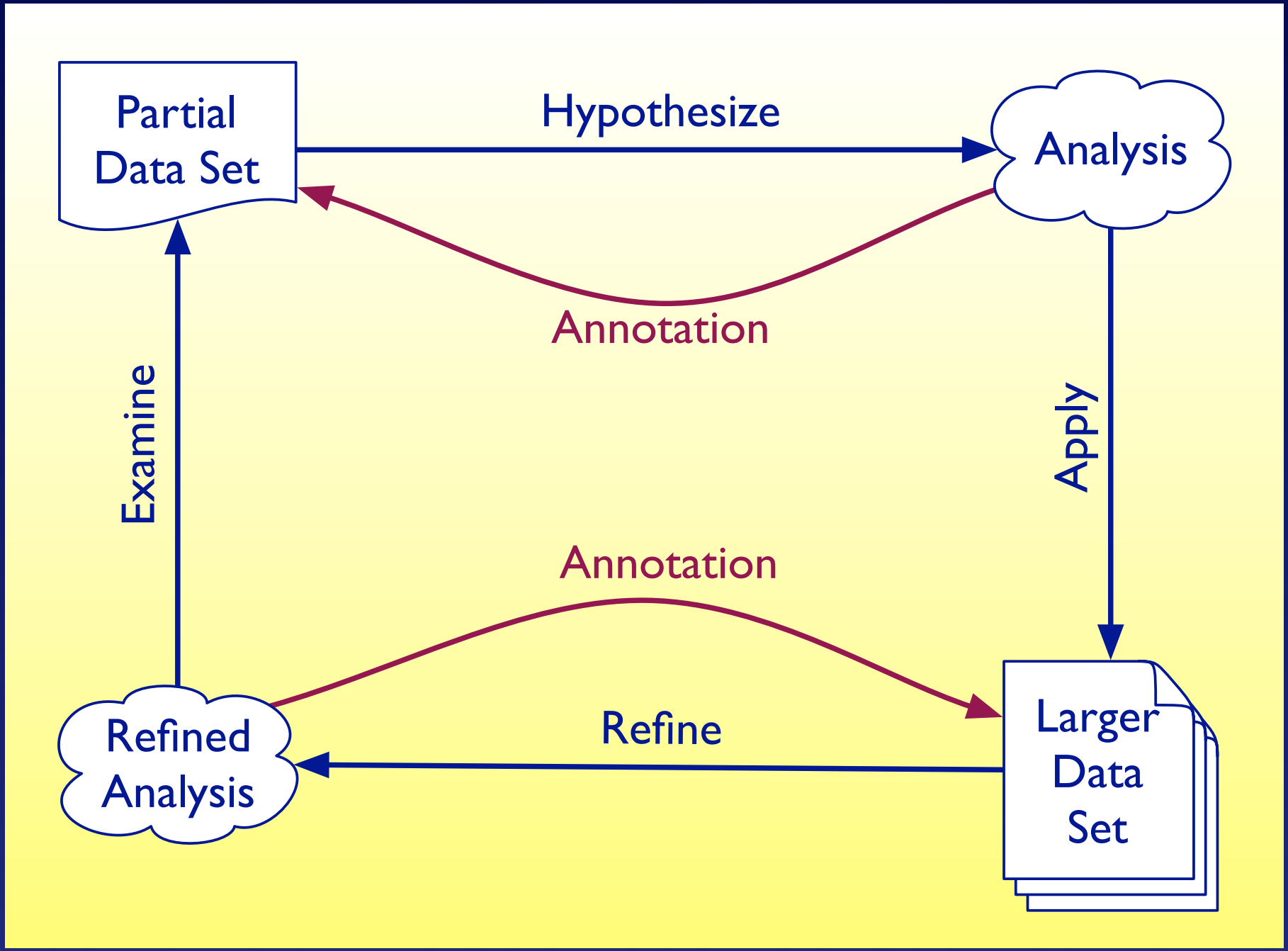
Analysis development



Analysis development

- At this stage, Montage is focusing on text data sets
- During analysis texts are *annotated* for grammatical information
  - Traditionally by hand
  - Electronically in various ways like word processors (unstructured annotation) or special annotation tools like Elan or Shoebox

Analysis and annotation



Annotation types

- What is annotation?
  - Transcription of recorded speech
  - Interlinearization
  - Even grammatical description, depending on how it's conceptualized
- Montage is focusing, at present, on morphological annotation

Interlinear text

Cicko,	*[ch'aara ´a*	*goj,]*		*'i*	*bu'u.*
cat.ERG	fish	&	see.CVPAN	3S.ABS	B.eat.PRS

'The cat sees a fish and eats it.'

(Example from Chechen)

Annotation

- Desiderata for an annotation system
  - Based on an extensible, open standard
  - Annotated resources in best-practice XML
  - Intuitive user interface

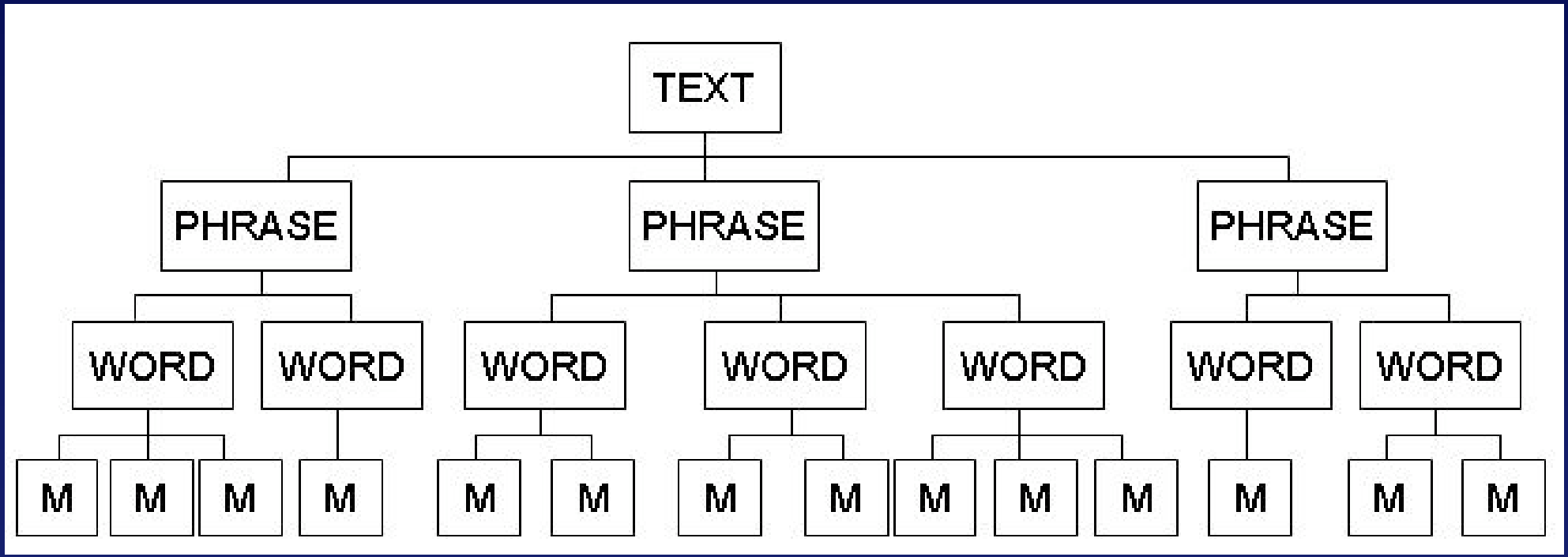
Annotation

- Some key aspects of producing good linguistic annotation have been largely “solved”
- In particular, the Annotation Graph model (Bird and Liberman 1999) is well-suited for linguistic data
- Elan annotation format offers another model which could be used

Annotation

- Similarly, there has been work on modeling the annotation structure of particular linguistic data types
  - Lexicons (TEI, FIELD)
  - Interlinear text (Bow et al. 2003)
  - Linguistic paradigms (Penton et al. 2004)
  - Descriptive grammars (Good 2004)

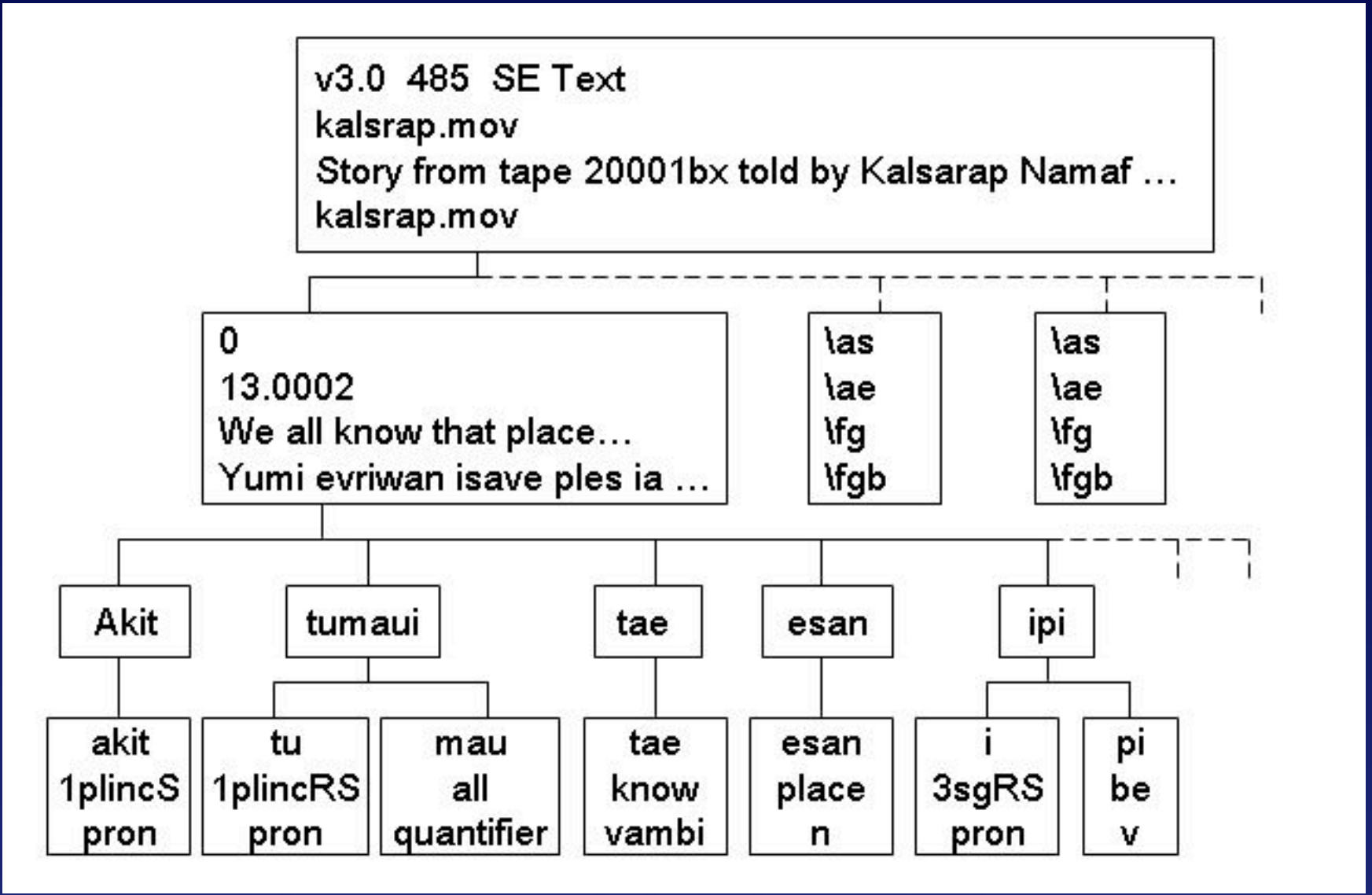
Interlinear text



The Melbourne model for interlinear text

(Source: http://www.cs.mu.oz.au/research/lt/projects/interlinear/)

Interlinear text



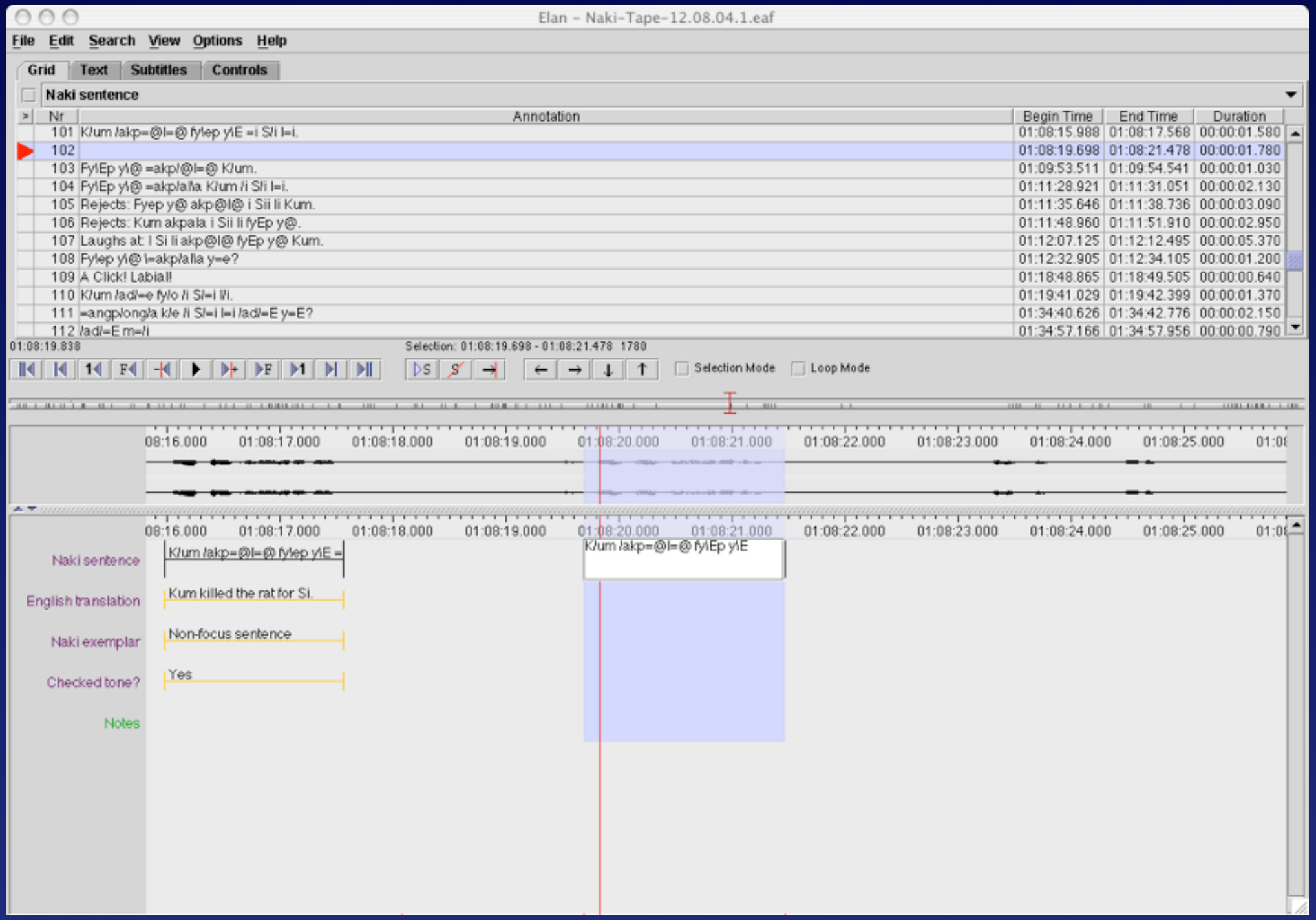
Example from South Efate (Austronesian)

(Source: http://www.cs.mu.oz.au/research/lt/projects/interlinear/sefate-tree.jpg)

Annotation

- The basic workflow for annotation seems well-understood
  - Select
  - Choose annotation type
  - Annotate

Select, choose, annotate in Elan



Ontologies

Ontologies

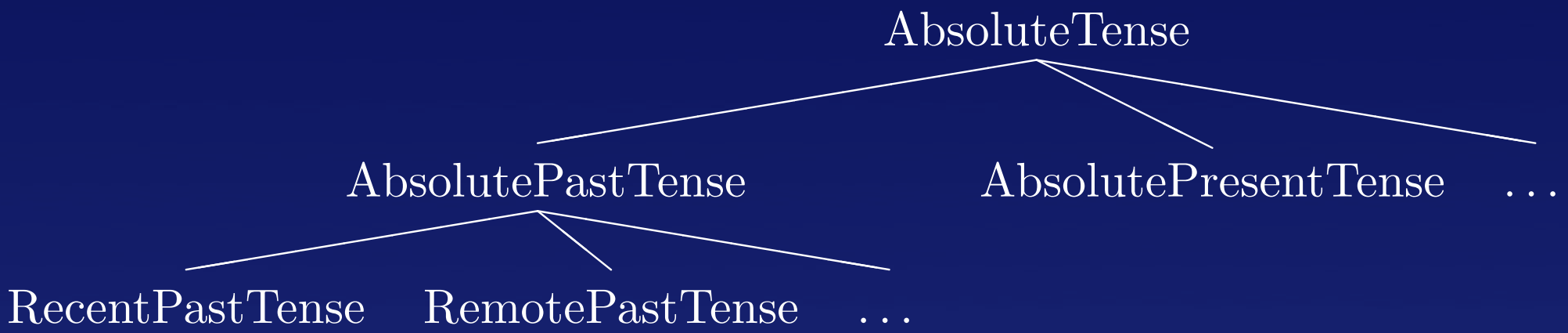
- Annotation structure may be well-understood from some perspectives
- However, an important, largely unsolved, question involves how to manage the categories used in annotation

Ontologies

- Ontology: A structured system of categories specified for the relationships which hold among those categories
- The most comprehensive linguistic ontology effort to date is the GOLD ontology (Farrar and Langendoen 2003)

A piece of an ontology

- A "sample" from the first version of the GOLD ontology



Ontologies

- For the purposes of descriptive linguists, ontologies serve as a sort of "interface" between the computer and categories used in description
- They also will play a role in "unifying" contextually-determined differences in category naming conventions within a resource

Ontologies

- Different ways of annotating categories
  - “word” vs. “ abc ”
  - “sentence” vs. “S” vs. “A b c d.”
  - “past tense” vs. “PST”
  - “ergative case” vs. “ergative” vs. “ERG”

Ontologies

- Two distinct problems
  - Keeping track of one's own terminology
  - Relating language-specific terminology to general terminology

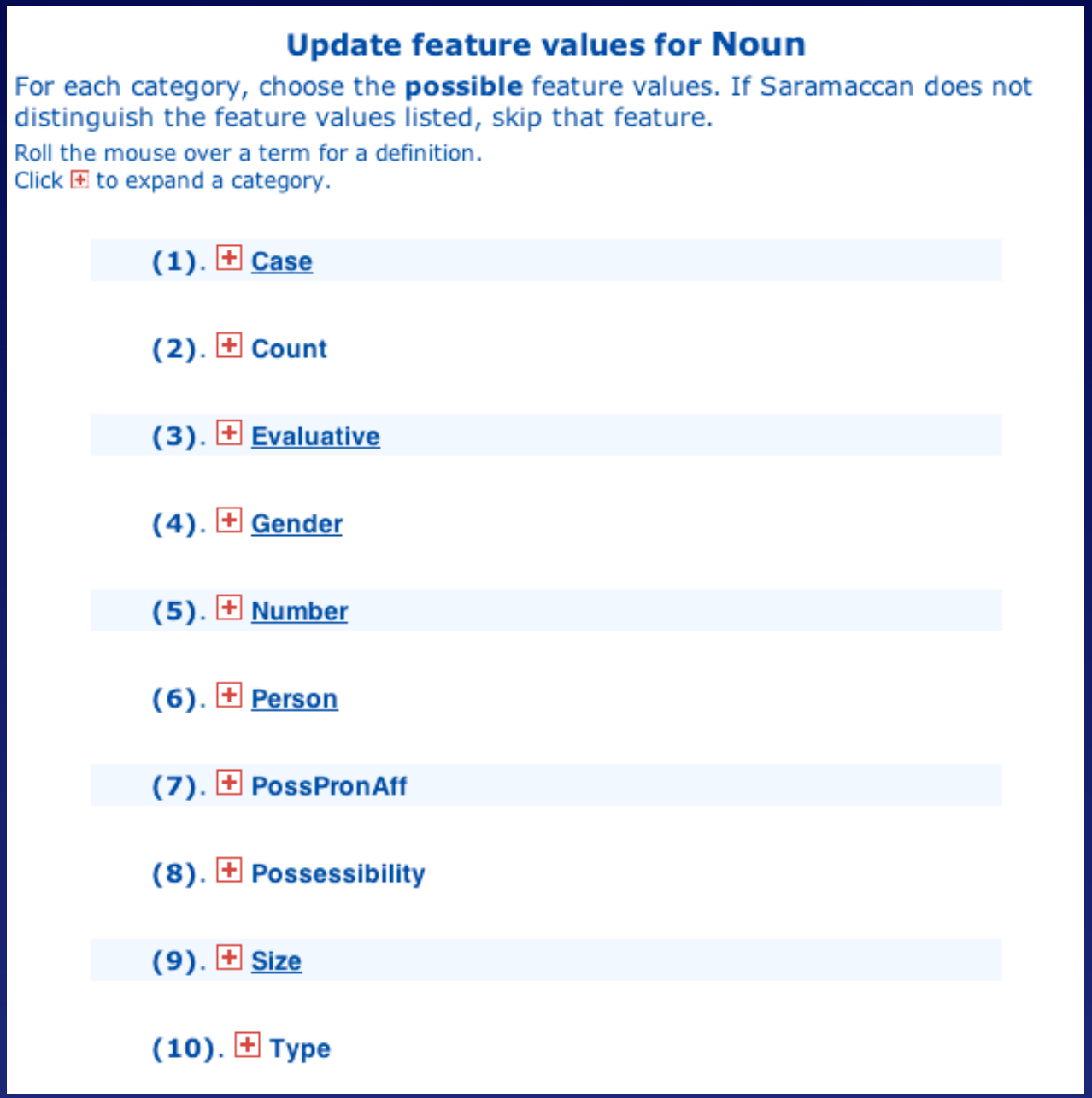
Ontologies

- The value of keeping track of one's own terminology is clear: Computers need consistency
- Linking specific terms to general ones
  - Will allow linguists to more easily discover resources they are interested in
  - Allow exploitation of typological knowledge

Ontologies and typology

- The FIELD tool, for example, “knows” basic verbal and nominal typology
- You can call “nouns” whatever you like, in principle
- But, if you tell FIELD that “MyNoun” is something like a “noun”, it will present useful customization options

FIELD, ontologies, and typological knowledge



A screenshot of one part of the FIELD interface

Grammar Matrix

- An important aspect of Montage: The Grammar Matrix (Bender et al. 2002)
- The Grammar Matrix: A framework for the development of implemented grammars for diverse languages
- For the purposes of the OWL: It serves as a bridge between work done on “big” languages and work to be done on “small” languages

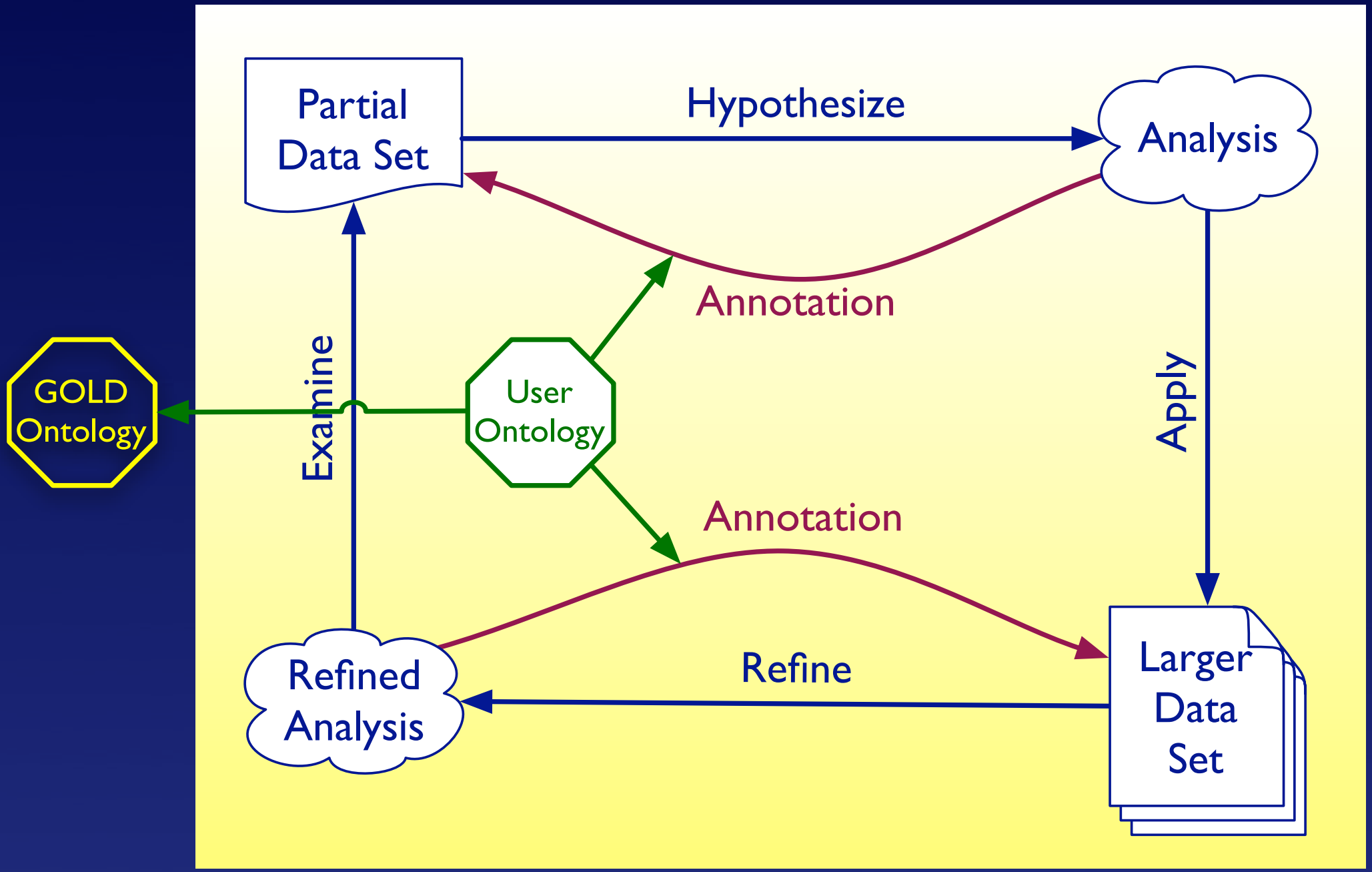
Grammar Matrix

- Like FIELD, the Grammar Matrix is designed to exploit typological knowledge
- For example, that nouns are often associated with case but not typically associated with mood
- The Matrix does not rule out unusual grammars—rather, it predefines common patterns

Ontologies and search

- The use of general ontologies, we hope, will allow for sophisticated searches with less annotation than would have been required in the past
- For example, a default definition of "oblique case" could be used even if the linguist has not yet explicitly defined that term for their language

Analysis, annotation, and ontologies



Parsing and search

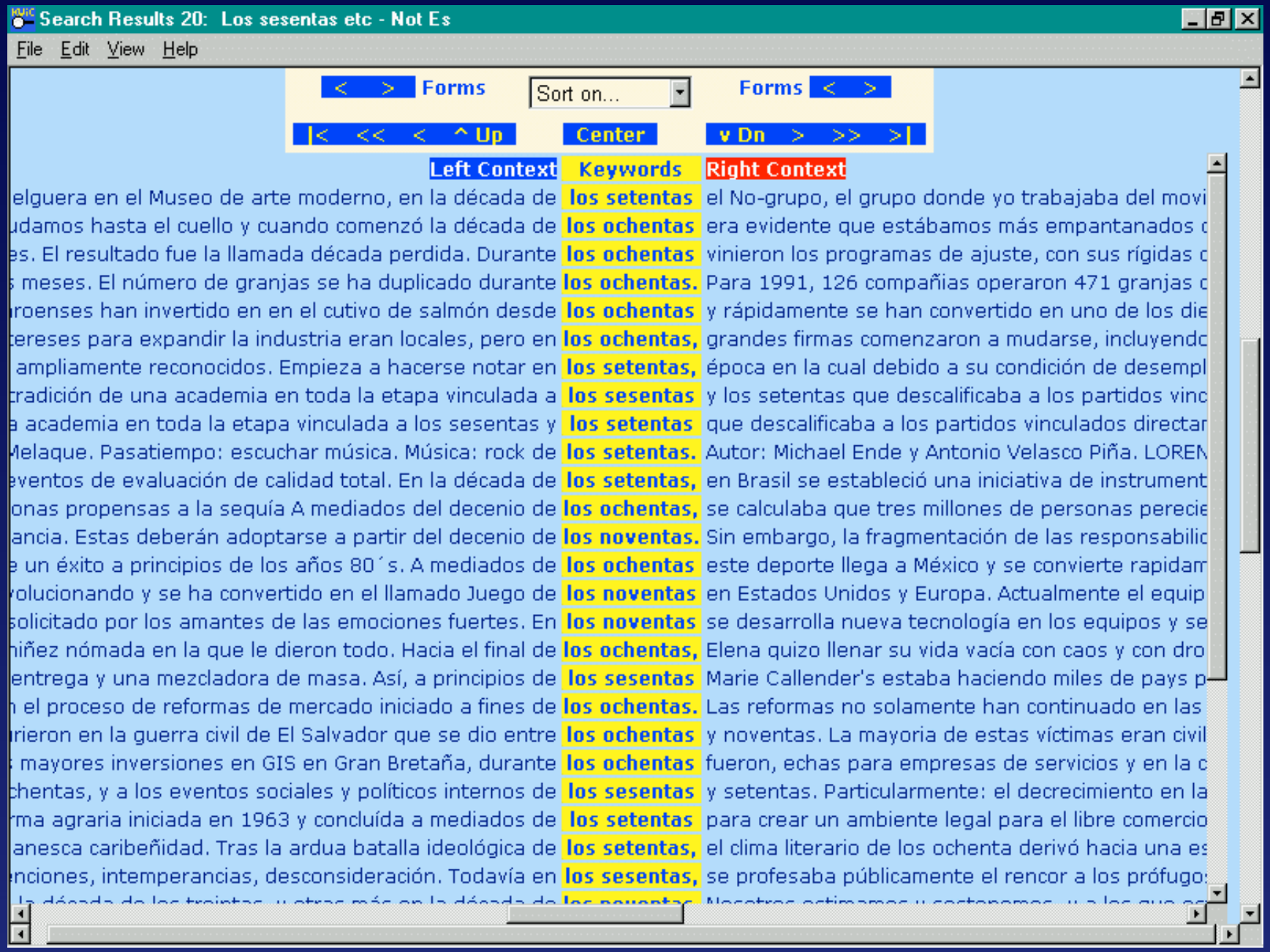
Parsing and search

- Annotation gets your data marked up for categories
- Ontologies define relationships among those categories and how they relate to categories in other resources
- The final piece of search involves parsing the data and the annotations to obtain the relevant “hits”

Search types

- Transcription searches
  - Substring: Like MSWord "find"
  - KWIC: Keyword in context (example following)
- From a technical perspective, these are relatively simple to implement (and are not specifically linguistic)

KWIC search



(Source: http://miniappolis.com/KWiCFinder/sesentasKWiC.gif)

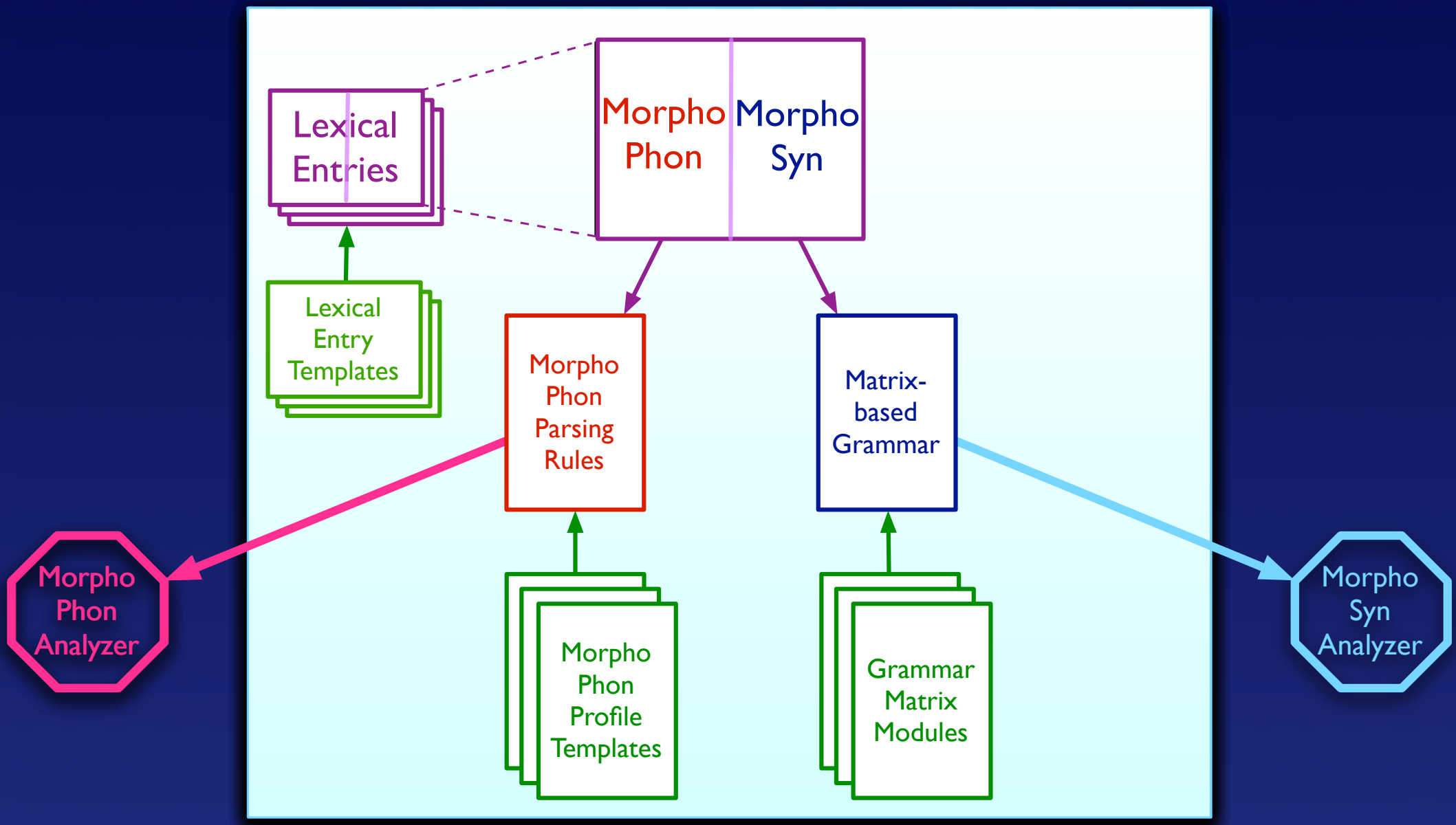
Search types

- Harder searches
  - Highlight (e.g. interlinear gloss) and find
  - Category strings: Like VO or OV
  - "Find more like these" (hard)
- These all require a system that knows at least a little linguistics

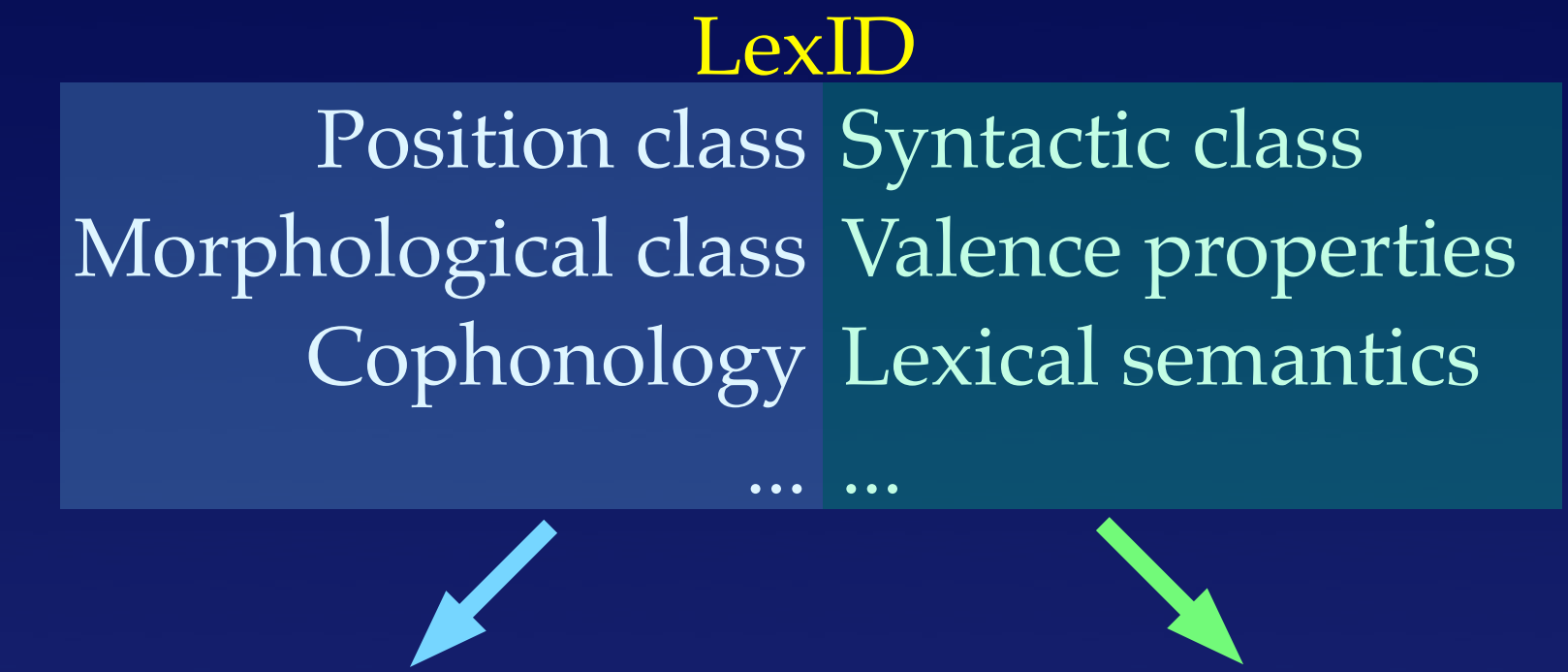
Parsing

- Linguistically smart searches will require a parsing system
- A key aspect of the Montage project is to exploit existing parsing and generating technologies to build tools for descriptive linguistics

Morphological parsing



Lexical Database



Reusing tools

- Comments on the implementation
- XFST finite state transducer system (Beesley and Karttunen 2003) for "morphophonology"
- LKB syntactic analysis system (Copestake 2002) for "morphosyntax"

Reusing tools

- Benefits of these tools
  - Bidirectional: Parse and generate forms
  - Use and produce resources in formats readily adaptable for other computational linguistic purposes
  - Resources more easily repurposed for CALL technology, word-processing applications, etc.

Reusing tools

- Drawbacks of these tools
  - User interface too complex for OWL
  - Resources produced and used not in a format easily usable or creatable by OWL
  - Tools missing core functionality for language description, like searching and collating

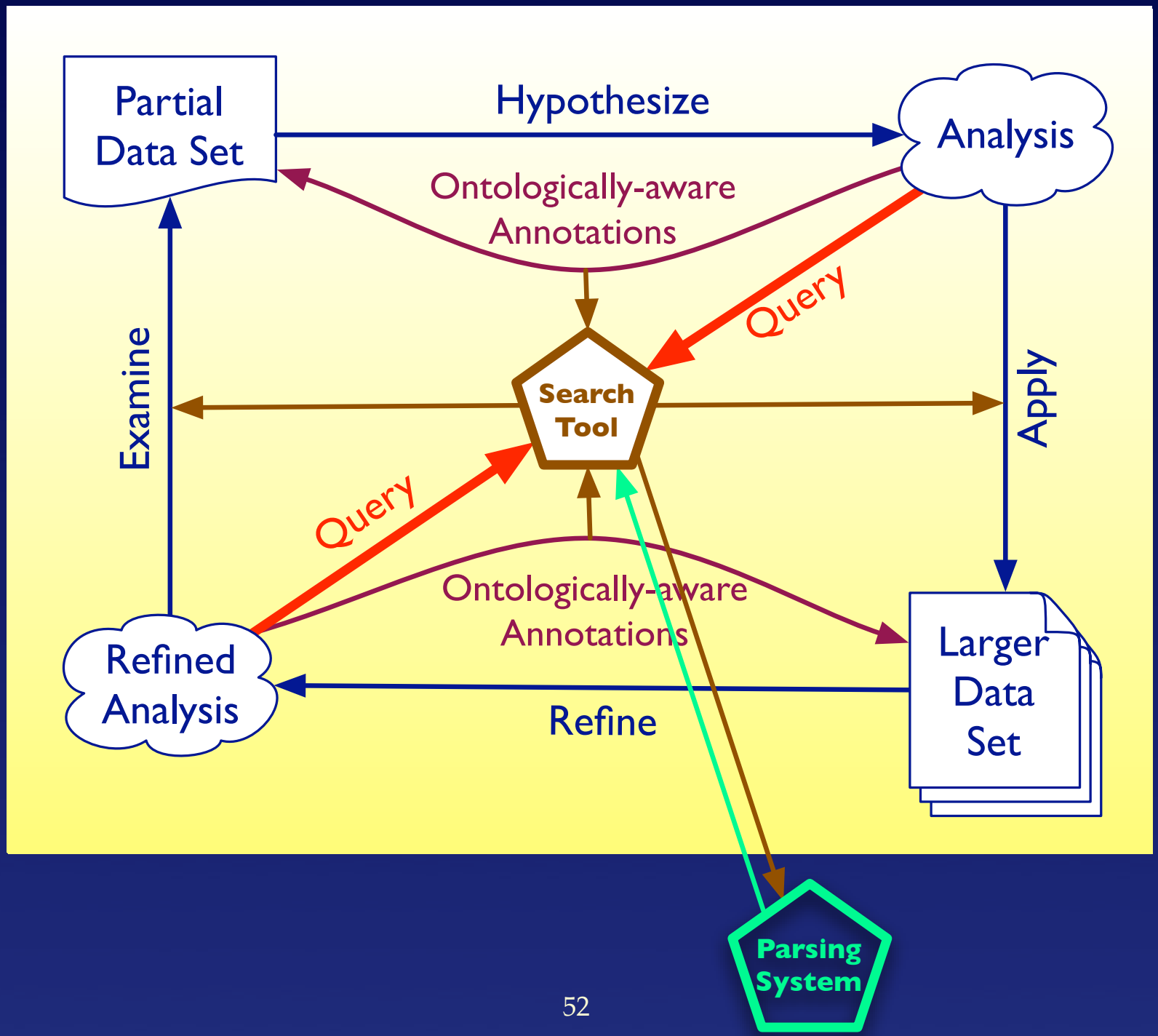
Reusing tools

- We are planning one major extension to these tools at present
- "Robust parsing"—allowing a parser to provide useful results even when the morphosyntax is not richly defined

Parsing

- What if I don't want to work on the resources needed by the parser?
  - Your search capabilities will be limited, but many important aspects of Montage will still work fine
  - Also, our design plans include the possibility that a computational linguist and field linguist will collaborate

Annotate and search



Conclusion

- While some aspects of Montage design are relatively fixed at this point, most are quite open
- We'd be curious to get feedback in any area from potential users of the toolkit

References

Beesley, Kenneth R. and Lauri Karttunen. 2003. *Finite State Morphology*. Stanford: CSLI.

Bender, Emily M., Dan Flickinger, Jeff Good and Ivan A. Sag. 2004. Montage: Leveraging advances in grammar engineering, linguistic ontologies, and mark-up for the documentation of underdescribed languages. *Proceedings of the Workshop on First Steps for the Documentation of Minority Languages: Computational Linguistic Tools for Morphology, Lexicon and Corpus Compilation, LREC 2004*. Lisbon, Portugal.

Bender, Emily M., Dan Flickinger and Stephan Oepen. 2002. The Grammar Matrix: An Open-source starter-Kit for the rapid development of cross-linguistically consistent broad-coverage precision grammars. *Proceedings of the Workshop on Grammar Engineering and Evaluation at the 19th International Conference on Computational Linguistics*. Taipei, Taiwan. pp. 8-14.

Bender, Emily M., and Jeff Good. To appear. Implementation for discovery: A bipartite lexicon to support morphological and syntactic analysis. *Proceedings of Chicago Linguistic Society 41: Panel on computational linguistics*.

Bird, Steven and Mark Liberman. A formal framework for linguistic annotation. Tech Report MS-CIS-99-01, Dept of Computer and Information Science, University of Pennsylvania.

Bow, Cathy, Baden Hughes, and Steven Bird. 2003. Towards a general model for interlinear text. In Proceedings of E-MELD Workshop 2003: Digitizing and annotating texts and field recordings, East Lansing, Michigan, July 11–13. http://emeld.org/workshop/2003/bowbadenbird-paper.html.

References

Copestake, Ann. 2002. *Implementing Typed Feature Structure Grammars*. Stanford: CSLI.

Drellishak, Scott and Emily M. Bender. 2005. Coordination Modules for a Crosslinguistic Grammar Resource. To be presented at HPSG 2005, Lisbon, August 23–24.

Farrar, Scott, and Terry Langendoen. 2003. A linguistic ontology for the semantic web. *Glot International* 7:97–100.

Good, Jeff. The descriptive grammar as a (meta)database. Proceedings of the E-MELD Workshop 2004: Linguistic Databases and Best Practice, July 15–18 2004, Detroit, Michigan. (http://emeld.org/workshop/2004/jcgood-paper.html)

Kaplan, Ronald M., John T. Maxwell III, Tracy Holloway King, and Richard Crouch. 2004. Integrating finite-state technology with deep LFG grammars. In Proceedings of the Workshop on Combining Shallow and Deep Processing for NLP (ESSLLI 2004), Nancy, France. Penton, David, Catherine Bow, Steven Bird, and Baden Hughes. 2004. Towards a general model for linguistic paradigms. In *Proceedings of the Proceedings of E-MELD Workshop 2004: Linguistic databases and best practice, July 15–18 2004, Detroit, Michigan*. http://www.emeld.org/workshop/2004/bird-paper.html.

Siegel, Melanie and Emily M. Bender. 2002. Efficient deep processing of Japanese. In *Proceedings of the 3rd workshop on Asian language resources and international standardization at the 19th International Conference on Computational Linguistics.* Taipei, Taiwan.

Abbreviations

AGTK	Annotation Graph Toolkit.
	http://www.ldc.upenn.edu/Projects/AG/
ELAN	EUDICO Linguistic Annotator.
	http://www.mpi.nl/tools/elan.html
FIELD	Field Input Environment for Linguistic Data.
	http://emeld.org/tools/fieldinput.cfm
Grammar Matrix	Precision Grammar Starter Kit.
	http://www.delph-in.net/matrix/
LKB	LKB Grammar Development Environment.
	http://www.delph-in.net/lkb/
Montage	Grammatical analysis toolkit (under development)
	http://depts.washington.edu/uwcl/Montage/
XFST	Xerox Finite State Transducer.
	http://www.fsmbook.com/

Presentation to be posted at:
http://depts.washington.edu/uwcl/Montage/