

# Downloading and running XNL-Soar

Deryle Lonsdale and Nathan Glenn  
lonz@byu.edu / nathan.g.glenn@gmail.com

# Modeling language in Soar (UTC)

- “Language should be approached with caution and circumspection. A unified theory of cognition must deal with it , but I will take it as something to be approached later rather than sooner.”
- “I find it formidable in terms of the degree of controversy that seems to attend competing views in linguistics. There seems to be no way to enter the fray with a little theory, even a highly approximate one. To do so is to invite the wrath of the linguistic gods. Full blown from the brow of Zeus or nothing!”

# XNL-Soar is...

- Incremental language processor built on Soar
- Designed to help in language modeling
  - Language acquisition/attrition
  - Language/task integrations
  - Interfaces, ambiguity, parsing, lexicon, etc.
- English, Japanese
- Syntax, morphology, semantics
- Parsing strategies, epmem, smem, analogical modeling, etc.
- More detail in prior years

# The system

- Built on Soar 9.3.1
- Extra functionality via Java
  - Interface to WordNet 3.1
  - GraphViz for graphing (parse trees, semantic graphs, etc.)
  - Callouts for similarity metrics
- SVN repository
- Eclipse IDE

# Out of the lab

- Goal: public release this summer
  - Long overdue, lots of unanswered requests
- Tension: development vs. support
  - Our own definition of “public”
    - Can't afford to support lots of people
    - First priority to Soar community?
  - Our own definition of “release”
    - Not sure we could coordinate/track submissions from wide community

# Downloading and installing

- For now: tarball upon request
  - Future: SVN or GitHub or SourceForge?
- System requirements
  - Soar, Perl, (WordNet), GraphViz
  - XP, Win7, Linux (haven't tried OS-X or Win8 yet)
- Have put a lot of effort into wiki (history, references, instructions, etc.)  
[nlsoar.byu.edu](http://nlsoar.byu.edu)

# Running

(1)

- Sentence(s) read in from file
- Constants file to choose runtime options
  - `PERL_APP=perl`
  - `GRAPHING_MEDIUM=gv, tex`
  - `CREATE_GRAPHS, DISPLAY_GRAPHS`
  - `GRAPH_SYNTAX, GRAPH_SEMANTICS, GRAPH_ARSET`
  - `NODE_INDEX=wmeid | nodeid | none`
- Via run/debug configurations in Eclipse
- At command line via Ant

# Running

(2)

- Interactive mode
  - Invokes Soar Debugger
  - Separate window for graph(s)
- Batch mode
  - Runs sentences from file(s) in folder
- Regression testing
  - Generates JUnit output, tracks success/fail
- Structure testing
  - Checks actual output structures against gold standard



# Caveats and FAQ

- The system is still evolving. There are many types of linguistic structures that it does not yet support. We are working to make the system more complete and more robust. Work within the limits of its current capabilities, and suggest improvements!
- Sampling of regression test sentence types.
- Ambiguity is still a huge issue we're working on. Try using unambiguous words wherever possible---you'll get better results. ...
- Semantic processing is not as advanced as syntax yet.
- The assigners/receivers set isn't guaranteed to work correctly yet.
- The XNL-Soar productions need considerable cleanup. Peruse it at your own risk!
- Not included: specialized components that we have developed for specific publications such as those that use semantic memory, episodic memory, or analogical modeling.
- How to find grapher output files, control graphing, etc.
- We'll add whatever is needed...

# Open issues

- License: free, but:
  - Which type? standalone? derivative?
- Which version?
  - 9.3.1 vs. 9.3.2 vs. 9.3.3 vs. 9.4
- Checking out/(in?) the code
- English vs. Japanese: separate or merged?
- Support for other IDE's (NetBeans?)

# Underway or contemplated

- More morpho-syntax-synsem knowledge
- Interface with Sphinx (spoken language)
- Multilingual agent (language control)
- Language acquisition modeling
- Careful reading (e.g. of web content)
- Chunking
- Generation: sem  $\rightarrow$  syn  $\rightarrow$  sentence
- Mapping: sem<sub>L1</sub>  $\rightarrow$  sem<sub>L2</sub>
- Discourse/dialogue processing
- Buffered input, decay (c.f. SimTime)
- French
- Massively ambiguous parses
- Finish semantics
- Semsnips: semantic snips

# Conclusions

## NUGGETS

- System is maturing
- Supports exploration of various strategies, different linguistic accounts
- Pretty easy, flexible installation and running
- Documentation much improved
- “Public” “release” imminent

## COAL

- Complex to understand system operation
- Available support might limit distribution
- Don't yet have exact date, distribution method
- Caution and circumspection are still advisable when dealing with language