Indexing of Chart Edges

- Generation chart is indexed by *semantic indices* (aka logical variables);
- one-dimensional chart: one cell per distinct variable in input semantics;
- *passive* edges are organized according to the index that they ‘provide’;
- *active* edges are organized according to the index that they ‘look for’;
- compatible pairs of edges from the *same* cell are tried for combination.

Dealing with Underspecified Indices

- some (often semantically empty) lexemes may not have an index value;
- some (typically active) edges may lack information about their index;
- chart *pen* for underspecified indices: conceptually ‘adjacent’ to all cells.
Relating Logical Variables to Indices

Thoralf Skolem (Norwegian Logician; 1887 – 1963)

- Lexemes of the grammar have feature structures of type index (INDEX);
- functors (e.g. chase) and grammar rules conspire to unify INDEX values;
- generation: ensure that indices for same logical variable are identical;
- assume that distinct logical variables introduce incompatible indices;

→ skolemization – specialize \[ SKOLEM *string* \] on indices by variable name.

(defun skolemize (dag ep)
  (let* ((dag (copy dag))
         (relation (ep-to-dag ep)))
    (unify dag relation '(SEM RELS LIST FIRST))))
Underspecified Indices

- Chart indices in generation are strings (i.e. the SKOLEM value on signs);
- for *underspecified* indices (no INDEX or \[SKOLEM \*string*\] value) use nil;
- storage: chart-adjoin() of an edge with a nil index adds to the pen;
- retrieval: passive-edges-at() et al. return all edges for nil position;
- adjacency: non-nil index in retrieval returns cell plus pen elements.

```
(defun dag-index (dag &optional path)
  (let* ((dag (select dag path))
         (skolem (and dag (select dag '(SEM INDEX SKOLEM))))
         (type (and skolem (dag-type skolem))))
    (when (stringp type) type)))
```
\{ x1: \texttt{the}\textunderscore q\textunderscore rel[ARG0 x1], \texttt{dog}\textunderscore n\textunderscore rel[ARG0 x1] \}
Generation Efficiency

The Problem
- Underspecified indices result in chart degeneration: ‘one cell’;
- everything tries to combine with everything → ‘shake and bake’;
- exponential combinatorics, no indexing benefits through chart.

Head-Driven Generation
- The first argument position of the ‘Specifier – Head’ construction will never determine its index value;
  → bi-directional rule instantiation: look for (semantic) head first.

Specifier – Head Rule

Lexemes (the and dog)
Reminder: Three Dimensions to the Parsing Problem

Vertical
- *top-down* successively rewrite $S$ until input is matched (goal-oriented);
- *bottom-up* combine constituents until $S$ is derived (data-oriented).

Depth
- *exhaustive* find all derivations (for each parsing goal) in ‘parallel’;
- *best-first* find one (or set of n-best) derivations as soon as possible.

Horizontal
- *uni-directional* instantiate rule RHSs left-to-right (or right-to-left);
- *bi-directional* instantiate rule RHSs in variable order, e.g. head-driven.
Benefits of (Bottom-Up) Chart-Based Generation

- **Lexicalisation**: virtually all semantics originates in lexemes: data-driven;
- **input semantics** minimally (i.e. non-) recursive: straightforward look-up;
- **no** (semantic) subsumption test at run-time; \texttt{independentp()} instead;
- **very efficient implementation** of \texttt{independentp()}: bit-vector conjunction.

Further Complexity in LKB Generator

- **lexical decomposition**  
  lexemes introduce more than one predicate;
- **construction semantics**  
  some rules introduce additional semantics;

→ simple adaptations of our algorithm (e.g. in look-up and skolemization).
Suggested Background Activities

- Retrieve the model solution for the last exercise from the course site;
- compare our solution to your submission; how is ours better (or not)?
- read [Kay, 1996], reflect on his (chart) parsing – generation parallelism;
- have a look at [Carroll, Copestake, Flickinger, and Poznanski, 1999], be sure to understand the efficiency concerns in modifier attachment;
- review all course materials; identify those aspects that remain unclear;
- email us your questions on (i) course content and (ii) NLP background.