Outline for January 21, 1999

1. Greetings and felicitations!
   a. Please get your project proposals in as soon as possible, so you can get started
   b. Hand out new homework; point out #6 and review rules of attacking *lassen* over the network
2. Take-Grant
   a. Show bridges (as a combination of terminal and initial spans)
   b. Show islands (maximal subject-only tg-connected subgraphs)
   c. can•share(r, **x**, **y**, G_0) iff there is an edge from **x** to **y** labelled r in G_0, or all of the following hold: (1) there is a vertex **y''** with an edge from **y'** to **y** labelled r; (2) there is a subject **y'** which terminally spans to **y''**, or **y'** = **y''**; (3) there is a subject **x'** which initially spans to **x**, or **x'** = **x**; and (4) there is a sequence of islands I_1, ..., I_n connected by bridges for which **x'** is in I_1 and **y'** is in I_n.
   d. Describe can•steal; don't state theorem
3. Lattice models
   a. poset, ≤ the relation
   b. highest and lowest
   c. Set of classes SC is a partially ordered set under relation ≤ with GLB (greatest lower bound), LUB (least upper bound) operators
   d. Note: is reflexive, transitive, antisymmetric
   e. Examples: (A, C) ≤ (A', C') iff A ≤ A' and C is a subset of C';
   LUB((A, C), (A', C')) = (max(A, A'), ∪(C, C')), GLB((A, C), (A', C')) = (min(A, A'), ∩(C, C'))
4. Decidability vs. Undecidability
   a. Notion of type; subject, object types
   b. SPM: types fixed at creation
   c. tickets: **X**/r (apply right r to entity **X**), **X**/r:c (c copy flag, so can give right away)
   d. link predicate: **X** possesses tickets in set dom(**X**). Then $link$(**X**, **Y**) = rights over **X** and **Y** ∈ dom(**X**) and dom(**Y**), and true; basically, everything connecting **X** and **Y**
   e. copying **X**/r:c from dom(**X**) to dom(**Z**): needs **X**/rc ∈ dom(**X**), link(**X**, **Z**), and τ(**X**)/r:c ∈ f(τ(**X**), τ(**Z**)), f filter function
   f. Do TG example pp. 82-83
   g. create: can-create (cc) relation says that subject of type a can create entities of type b iff cc(a, b) holds
   h. create rule: $cr_p(a, b)$ = set of tickets added to dom(A), $cr_c(a, b)$ = set of tickets added to dom(B)
   i. attenuation: $cr(a, b) = cr_p(a,b)|cr_c(a, b)$ is attenuating if: (1) $cr_c(a, b) \subseteq cr_p(a, b)$ and $a/r{:}c \in cr_p(a, b) \Rightarrow$ $self/r{:}c \in cr_p(a, b)$
   j. cyclic: $cc(a, b) \Rightarrow cc(b, a)$
   k. if attenuating acyclic, it's decidable; so that is sufficient. Open question: is it necessary?