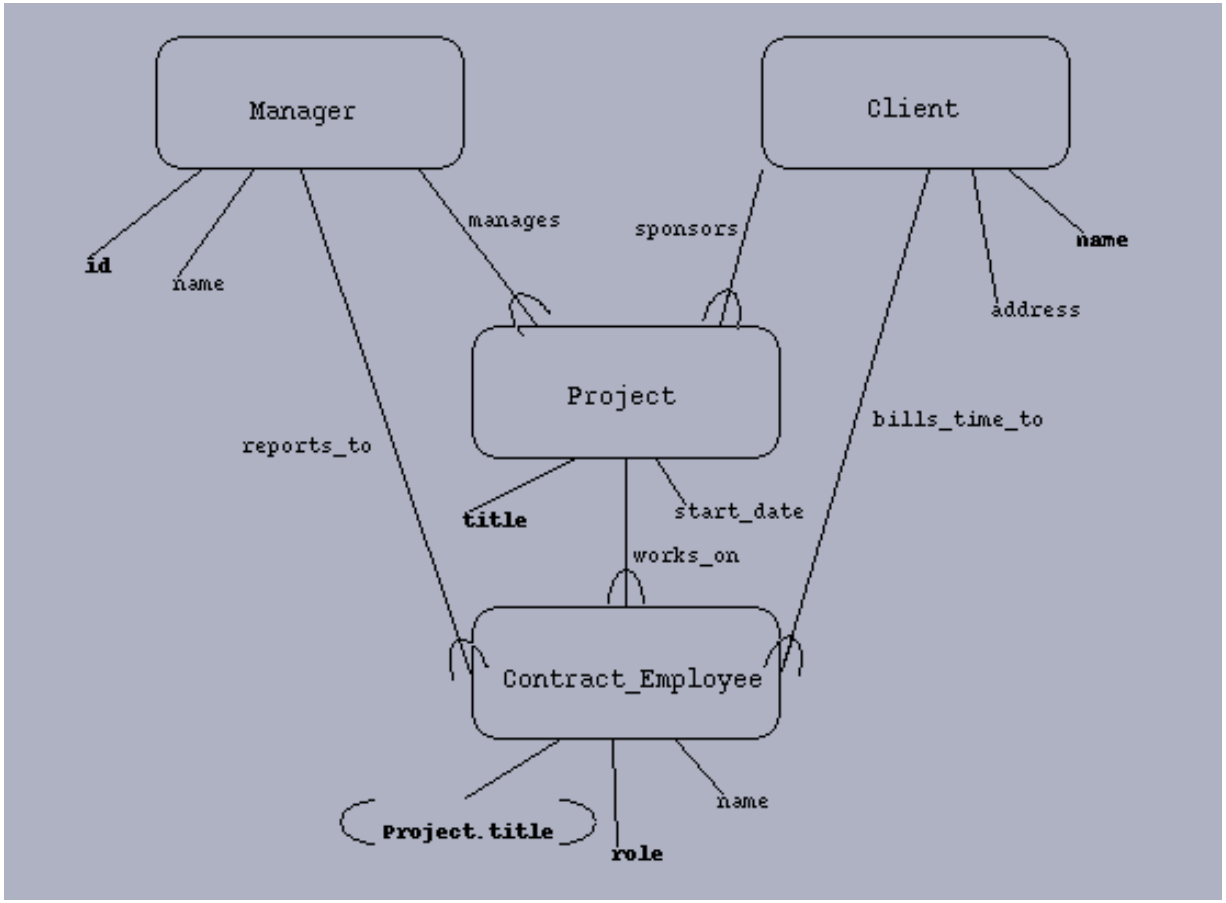
Comprehension of Semantic Models

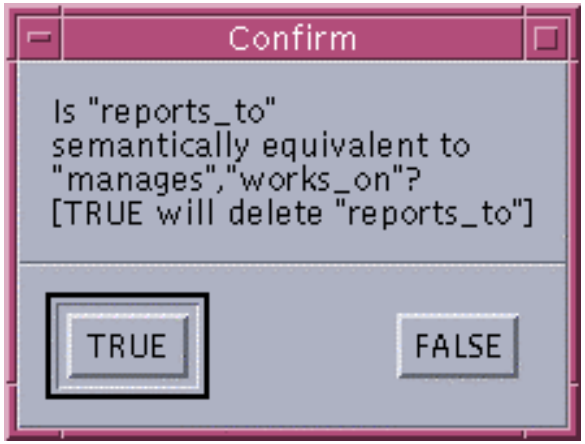
- **User-oriented notations**
  - **usually graphical**
    - ***so that you can draw silly things!***
- **(Some) rigorous foundations**
  - **which need to be kept …. as foundations!**
- **Transformation into well-engineered code**
  - **"lower CASE"**
  - **pervading assumption that ANY resulting design "needs" "improvement"…**
- **Detection of errors / omissions / surprises**
  - **Diagrams intended for communication-**
    - ***but even* analysts *may not understand them***
  - **Interaction with client(s) tends to be confined to "early" stages**
- **CASE support**
  - **making the "waterfall" flow *both* ways!**
  - **more than just "clever diagrammers"**

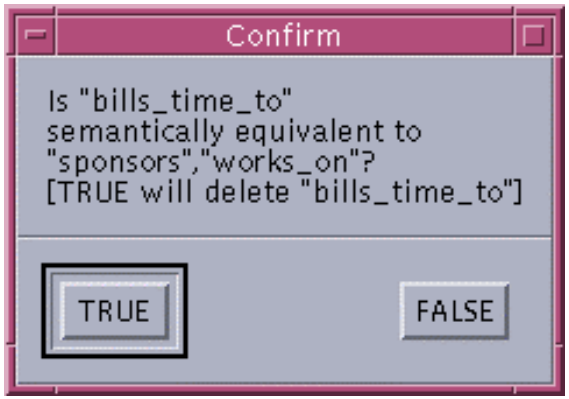
Primary Areas of Focus

- **Entity Relationship Modelling**
  - Relational Synthesis
  - Detection of Redundancies
  - Construction by analysis of textual descriptions
  - Generalisation to other notations with similar morphology
- **Behaviour Modelling**
  - State-based behaviour
  - Parallel composition
    - *"Aggregation"*
  - Emergent Behaviour
- **Meta-CASE**
  - Support for specific (partial) notations
  - Reasoning about models
  - Medium, rather than message

Detecting Redundancy in an ER model







Parallel Composition of State-Based Behaviour - The Problem

- **Parallel composition is exponentially explosive**
  - **Each component can, potentially, coexist in any state with every other component also in any state**
- **Need composition for encapsulation**
  - **An aggregate object behaves as a single object**
- **Would like state reduction**
  - **Identification of unreachable / specifically excluded states/transitions**
- **Constraints likely to be semantic**
  - **Require specification / verification by client**
- **Require medium comprehensible to clients**
  - **Graphical rather than mathematical**