Task, Backend Info

- **Task: Medical diagnosis**
  - Used supplied diagnosis tree

- **Backend Info:**
  - Supplied text documents
What needed to be changed

- **Domain module (from travel to diseases)**
  - Task model (linear task to decision tree)
  - Object model (slots needed from user)

- **Lexicon module (from travel diseases)**
  - Text input/output part of lexicon module

- **Dialogue model unchanged**
  - Questions Under Discussion
  - Started from GoDiS sample TrindiKit 2.0
• All execution is asynchronous, except IS updates
• System updates IS, selects new move whenever input moves change
• Initiative passes to user when system stops selecting new moves
• Lexicon does “heavy lifting” for parsing and generation
• Domain associates plans and questions, maintains plan instances
Domain Model

- **Tasks**: plans for soliciting information
- **Questions**: slots required to complete task
- **Restricted view of objects, relations**
  - no iteration or cardinality > 1
  - all tasks apply to single flat set of questions

---

**Task sample: start of diagnosis task**

```plaintext
task diagnose_disease { 
  findout headache(X1);
  if headache(yes) then { 
    findout fever(X2);
    if fever(yes) then { 
      findout oconus(X3);
      if oconus(yes) then { 
        findout malarial(X5);
        if malarial(yes) then { 
          inform disease(malaria);
        } else { 
        } 
      } else { 
      } 
    } else { 
    } 
  } else { 
  } 
} 
```

**Question sample: slots for diagnosis task**

```plaintext
question headache expects yesno;
question fever expects yesno;
question oconus expects yesno;
question malarial expects yesno;
question swimfresh expects yesno;
question hematouria expects yesno;
question dust expects yesno;
```
Highlights/Lowlights of DM for this Exercise

- **Low**
  - Doesn’t handle multiple symptoms very well, so easier to do diagnosis tree than open-ended questions
  - Performance not up to targets

- **High**
  - Meets goals for our DM
    - write in abstract rule-based language, translates out to java
    - flexible, open representation
  - Fast to change domains
  - Something about object modeling here?