CS646
Lab & Mini-project Assignment

Mechanics

- Everything is to be submitted through Boddington
  - ***Everything must be identified with your library/University ID number!***
    - **Otherwise you are throwing it away!**
  - **Include your library card number in all *filenames* AND in *contents* of each zip file as a read me or with the write up**
- **If you do not have a Boddington account, email samantha creighton – screighton@cs.man.ac.uk**

Anonymous marking

- Please do not put your name on your work
- DO put your library card number on all work and in all file names
- DO be sure Janet has your library card number and email so she can forward any mail to you if there are problems

Summary - Labs

- The labs during the week are primarily for experimentation and learning. However, we expect evidence of work done. All labs and miniproject must be submitted by end of term. Extensions to 1st day of exam period possible but not advised.
  - 1 zip file each containing the final ontology(ies) and up to one page of A4 describing your work and problems for each of
    - The Normalised Ontology of People (Tuesday). 5 marks
    - Ontology problems – domains and range constraints and debugging (Wednesday AM) + Ontology patterns (Thursday) 10 marks
    - The DL reasoning problems from Ian Horrocks (Wednesday PM) 10 marks
    - An extension to the high level ontology described on the end of this sheet (Friday – may be combined with mini project) 10marks
    - Questionnaire on Protégé Owl (5 marks)
    - Miniproject 60 marks
- Course work total: 60% of assessment (half on labs; 60% on miniproject)

Summary - All labs: Commenting, documentation and testing

- All ontologies should be fully commented
  - There should be a paraphrase of the intended meaning for each class and additional comments on any special issues or usages of that class. Any classes that are intended as ‘axioms’ or ‘probes’ should be clearly commented.
- Ontologies are software
  - *Software is not complete unless documented and provided with test data and a test mechanism!*

Monday Lab

- Work through the Pizza example to define and get correct classification for vegetarian pizza and protein lover’s pizza
- Use the Manchester Pizza Finder to see the results of implementing your ontology
  - Instructions on separate sheet
- No lab to turn in.

Tuesday- Lab

- Take cards for University ontology to produce an ontology for the university including the personnel department’s equal opportunities officer
- Group the cards and form initial hierarchies
  - Separate likely primitives, modifiers, roles, defined concepts and properties, classes and individuals
  - Ladder up to provide abstractions as needed
    - And fill in siblings
  - Propose a normalised ontology
    - Classify it to see that it works correctly
      - Provide probe classes to check both classification and unsatisfiability
        - » One file to turn in
  - Download the tangled ontology proposed by the personnel department
    - Untangle it
      - A second file to turn in

Wed AM Lab

- Work through the lab handout on Existential and Universal restrictions and domain and range constraints
- Use the domain-checking to determine if there are problems
- Create an ontology that demonstrates the difference in behaviour of domain and range constraints
- Be able to answer in your write up why range constraints cannot ‘coerce’ classes to new positions in the hierarchy
- Create a definition for an all girl-child family that can be trivially satisfied.
  - Modify it so that it can only be non-trivially satisfied
  - Show why the incorrect version is wrong

Thursday Lab

- Download the simple university ontology from the web
- Create a new value partition for the difficulty of a module
- Re-represent the difficulty of a module as a class to give both a reason and an extent of difficulty
- Represent the assessment for a module and its parts – lab, lecture, mini-project
- Represent the rules that an assessment is failed if any of its parts are failed and passed if all of its parts are passed
- Indicate in your write up how you would do this using concrete domains and values for a pass mark of 50
  - It cannot yet be done in the tools well
- Create a set of probe classes to test your ontology

Friday Lab

- Explore the top ontology in ~rector/CS646/Labs/Friday
  - Add (if you have not already done so on Thursday
    - A new kind of building
    - Lecture theatres
    - Clubs – and perhaps student clubs
    - Staff of rank “Senior lecturer” and “Reader”
    - Features (“qualities”) for difficulty of modules, comfort of chairs, and quality of lectures.
    - The parts of the building – rooms, doors, lab-spaces, the front of lab spaces, etc.
    - Parts of modules
    - Express the axiom that a module is failed if any of its parts are failed
- You may not complete all of this. Finish what is practical.
- Write a one-page critique of the upper ontology comparing it with Dolce/Ontoclean

Protégé Problems

- The last release of Protégé before the module fixed some bugs but added others. One bug causes ontologies with nested properties to run VERY slowly at times. The current upper ontology seems to avoid this, but if it happens.
  - a) Rebuild from the OWL
  - b) If that doesn’t work, use the earlier version of Protégé in www.cs.man.ac.uk/~drummond/cs646
    - This is a simple zip file that should work if unzipped anywhere
  - c) We expect a new release today or Monday that will fix the problem. A message will be posted in in the lab space and a zip version placed in the same directory as the above.

Protégé Problems (2)

- Both current versions of Protégé occasionally make transitive properties functional or inverse functional for no apparent reason and refuse to maintain corrections
  - Fortunately the interface with the reasoner filters these errors out, but it does give warnings. Ignore them!
    - Hopefully all will be fixed in the new version.

Mini Project

- The basis of the miniproject is to use what you have learned to create an interesting ontology. It is assumed that you will start from the top ontology in …/~alr/CS646 but you may use another
- Our suggested topic is to extend the ontology to cover other modules and aspects of the University
  - Accommodation – buildings, staff, etc
  - Eating facilities –
  - Study facilities – the library, what's in it, how it is used, who uses it, who works there
  - Greater detail on teaching activities – different kinds of labs, lecturers, etc.
  - You may combine the mini-project and Friday's assignment, but you must put a note in the Friday assignment pigeon hole in Boddington to say that you have done so.
- The assignment is to turn in a single zip file containing:

Miniproject submission

- 1 zip file for your mini project containing
  - **A report of up to five pages documenting your work.**
    - What you have done
    - What you would like to do but couldn't because it was outside the OWL paradigm
    - What you would like to have done but didn't know how to
    - Bugs and problems with the starting ontology
      - Check regularly for updates. We will try to fix bugs as they are found.
    - A sample of how the ontology might be used to mark up some known resources - informally
  - The ontology itself
- **This will be 60% of the course work mark**

Other Mini-project Alternatives

- Extended pizza ontology to baking and selling pizzas
- Parts and wholes ontology for any engineering or similar activity
- Ontology of software design tasks and methodologies – or similar CS topic
- Enriching/untangling some part of the ontology from Wikipedia or Open Directory
- Extending, enriching some other ontology from the web in OWL DL
- If you have a topic you would particularly like to work on, talk to us. Please get permission before you pick an alternative topic.

Marking the Mini Project

- The write up is primary, I expect to know
  - The scope of the ontology – what it is for.
  - The decisions made and the rationale for them
  - The testing procedures and whether it passed them
    - I would rather see an interesting ontology with sensible test procedures that did not quite succeed than a trivial one with no testing procedures that did
  - A description of how it would be used to annotate web pages or other resources, preferably with specific examples and how concepts or statements using concepts from the ontology would be used to describe them.
    - Preferably in OWL or RDF, but since this has been covered only briefly, an informal version will do.

Marking the Miniproject (2)

- The ontology
  - Should use a principled upper ontology – the one supplied or another
    - feel free to simplify, but state in your write up what you did.
  - Should have a normalised domain ontology
    - i.e. domain primitive entities should form disjoint trees. All multiple classification should be the result of inference
  - Should demonstrate interesting use of the reasoner
  - Ideally, should be satisfiable
    - If not, identify problems and efforts made to trace them to source.
      - You may even find a bug, so don't panic, just identify it!

Please remember to put your ***library card number*** on all work

Please be sure that Janet has your email address and library card number so we can reach you if there are problems with any assessments.

Include your library card number in all filenames

Appendix: to Download and Install Protégé OWL and related

–Full instructions

- http://www.co-ode.org/resources/tutorials/iswc2004

–Summary – you need

- Protégé 3 Beta – complete installation
- Racer – plus a shortcut to start it easily
- GraphViz – please install in default location
- CO-ODE extras from www.co-ode.org
  - class description plugin
  - probe class framework plugin
  - alt syntax widget (optional)
  - debugging plugin (optional)www.co-ode.org
- Example ontologies
- Optional: Long version of Pizza tutorial
  “Pizza finder” application