# Software Engineering I & 2

Bijan Parsia
bijan.parsia@manchester.ac.uk

September, 2017

# Software Engineering

- Software Engineering is a discipline
  - concerned with all aspects of software production
- Goal:
  - produce successful software systems
  - by means of successful software development projects.
- This isn't easy!
  - Software is different from many other engineered things
    - Software is a complex system
  - Software is increasingly *part* of those other things!

# Aspects of Software Engineering

- Requirements and design
  - What are the desired properties?
  - What's the blueprint for the system?
- Construction and maintenance
  - How do we *build* the system?
  - How do we *evolve* it to meet new circumstances?
- Testing and quality assurance
  - How do we know we have the desired properties?
- Management and methodology
  - How do we *run* the project?

# Software Engineering 1

*This theme aims to provide students with core concepts as well as an experiential grip on central aspects of software engineering.*

*Research informed*

- We examine
  - All aspects of software engineering
    - From management to construction
  - How to build successful *products* with successful *projects*
- You get
  - Practical skills and experience
  - Theoretical and conceptual understanding

# Course Units in SET

**COMP61511**: Software Engineering Concepts in Practice

- Period 1
- From theoretical to practical and back again:
  - software engineering as systems engineering
  - translating concepts and research into practice
  - aims at a *systematic grasp*
  - focus on *you*

**COMP62521**: Agile and Test-Driven Development

- Period 2
- Hands-on Agile:
  - cultivate an “agile mindset”
  - understand *methodology*

# Key bit!

- COMP61511 and COMP62521 align
  - Mostly!
  - Both use Python as the core language
    - Brush up or run through a tutorial if you don't know it!

- there will be
  - new content
  - new exercises & coursework
  - new labs
  - new software

# COMP61511: SE Concepts in Practice

- Assessment: 50% Coursework; 50% Exam
- Weekly coursework:
  - quiz
  - short essay
  - programming tasks (in labs and at home)
  - written and interactive feedback
- We expect you to read.

# COMP62321: Agile and TDD

- Assessment: 25% Coursework; 75% Exam
  - Exam is based on coursework
- Weekly coursework
  - Working in teams, with pair programming
    - Focus on Scrum
  - Weekly miniquiz
  - Reflection and interactive feedback

# Software Engineering 2

*This theme aims to provide students with an understanding of two major approaches to software development: components and patterns.*
*Research driven.*

- We examine
  - The CBD process and various component models
  - The notion of patterns and patterns for software and e-business design
- You get
  - Practical skills
  - Theoretical and conceptual understanding

# Course Units in SE2

- **COMP62532**: Component-based Software Development
  - Beyond object-oriented programming:
    - aim to make software engineering more like manufacturing
    - programming as assembling ready-made components
    - how to specify composable components
    - research led teaching! CBD is not a solved problem!
- **COMP62542**: Pattern-based Software Development
  - Language of design:
    - pattern = "a solution to a recurring problem in a given context"
    - patterns started in design but exist for all aspects of SE
    - how to describe patterns
    - how to recognise problems in context for applying a pattern

# Software Development

- Assessment: 50% Coursework; 50% Exam
- Feedback in lectures is given
  - interactively both verbally and
  - via Classroom Presenter - a software system for interactive lectures.
- Feedback in labs is given
  - both interactively (verbally) and in written form.
- Feedback on group presentations is given
  - interactively (verbally).

# Software Development

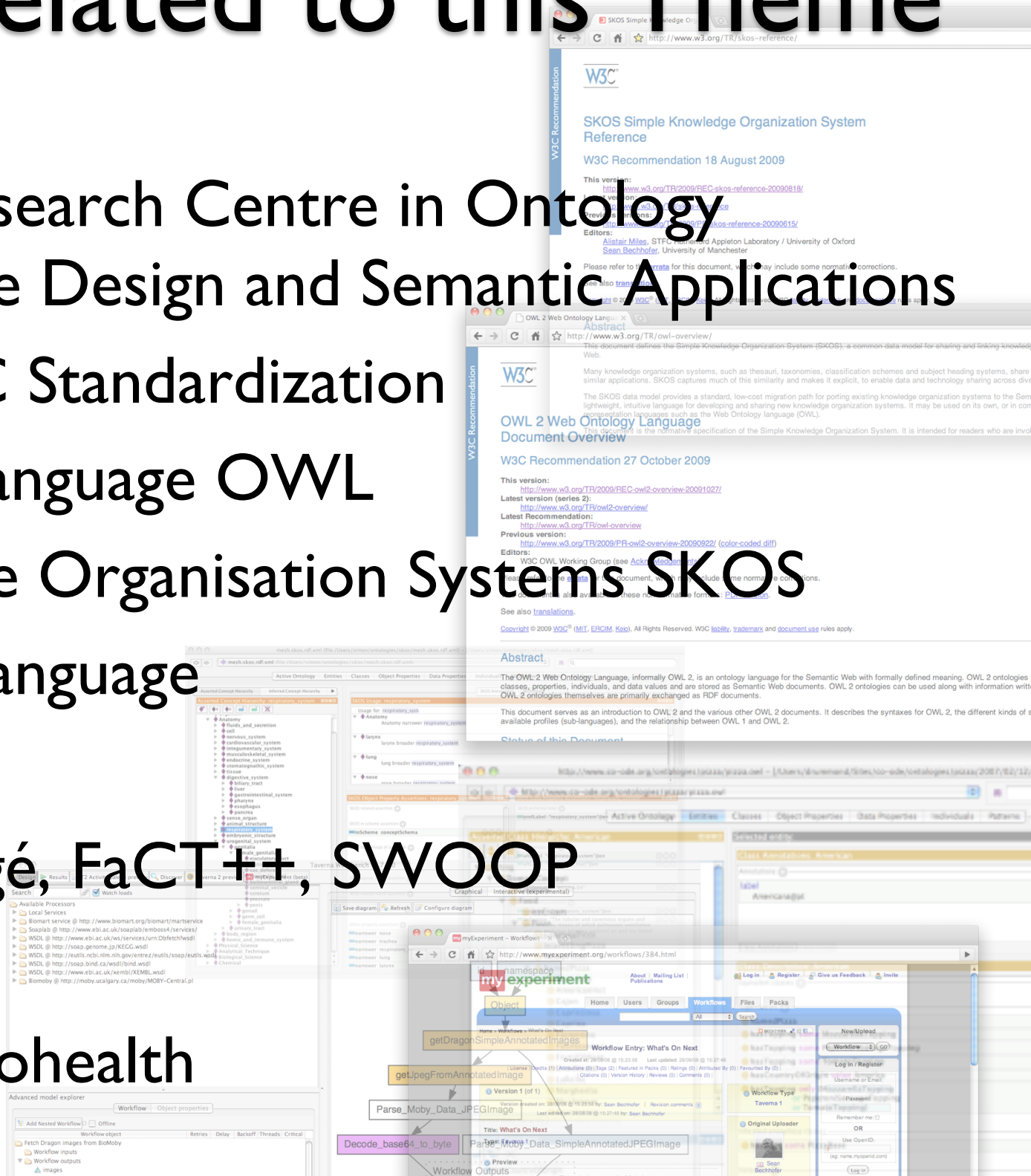
- Assessment: 50% Coursework; 50% Exam
- Coursework consists primarily of case studies
  - Applying patterns to given problems
    - In software design
      - Gang of Fours style
    - For e-Business/Business process modeling
      - IBM's 'patterns for e-business'
  - Feedback is written

# Pre-Requisites

- for both: a background in Databases
  - a good UG module "Fundamentals of Databases"
  - remember tables, SQL queries, Joins,…
- 60411: confident Java programming
- both: being happy to
  - think things through
  - analyse pros & cons
  - understand technically challenging concepts

# Research Related to this Theme

- UoM is a Leading Research Centre in Ontology Engineering, Language Design and Semantic Applications
- Instrumental in W3C Standardization
  - Web Ontology Language OWL
  - Simple Knowledge Organisation Systems SKOS
  - SPARQL Query language
- Tool Development
  - OWL API, Protégé, FaCT++, SWOOP
- Applications
  - Life Sciences & Biohealth
  - eScience



# Is it for me?

http://www.flickr.com/photos/-ba

These themes **are not** for those …

- who want a programming refresher
- who don't like to "get their hands dirty"
- who don't like to read around the subject

These theme **can be** for those…

- looking to understand what software engineering is all abo
- seeking professional development as a software engineer
- interested in software engineering research

# QUESTIONS?

(feel free to come chat with me later…my office is 2.88a)