CS405 Introduction to Empirical Modelling - 2010-11

Empirical Modelling ("EM") is a body of principles and tools concerned with computing activity that is based on observation and experiment (hence 'empirical') ...

... the principal theme of CS405 is alternative ways to think about "computing-in-the-wild" - with special attention to computer *programming* and computer *science*

Introductory notes for Lab 1

Orientation

Initial focus will be on "programming" in a non-standard style

More appropriate to regard this as "model-building" rather than traditional programming

Compare the aspirations in:

- object-oriented programming - seeing programming as modelling
- agile development approaches: emphasis on *evolving* specifications and programs

CS405 draws attention to the semantic / philosophical issues that are raised by making these moves
- e.g. what is the status of the traditional story about specification, and program/PL semantics?

Tools

Will be making use of two environments for EM in the course of the module:

- the EDEN interpreter, a well-established tool that has been used for over 20 years

- CADENCE, an environment comprising a suite of modules based on the DOSTE engine, a research prototype being developed by Nick Pope

Some familiarity with EDEN will be essential, as many of the illustrative models to be discussed in the module make use of it. In the assessed work, students can make use of either CADENCE or EDEN. Since one of the modules that is being developed for CADENCE is based on a variant of EDEN, it is also possible to make use of both in the same model-building exercise. The DOSTE engine does not itself have interfaces for input and graphical output, but these can be supplied. When using CADENCE without EDEN, textual input can be given by the DASM notation, and visualisation can exploit the Warwick Games Design (WGD) module.

How tools are used in the module

Characteristics of tools to be introduced in the module ... they are concerned with modelling in which we:

- observe meaningful things
- adopt a *constructivist* stance
- exploit an *empirical* approach

that we wish to reconcile / can be reconciled with the more abstract, rationalist, theoretical framework that characterises classical computer science
Key notions

In Empirical Modelling, we build *construals* rather than programs. A construal is an artefact you can interact with which helps to make sense of something else. In this lab, the computer model we study is a construal of the Stargate - interacting with it helps to understand what a Stargate is, whether or not you already have some idea of this.

In building construals rather than programs, we need different concepts. The key concepts we introduce are:
    observable - dependency - agent ("ODA")

- an **observable**: - something perceived as having an identity and being subject to change (e.g. a chevron in the Stargate)
- a **dependency**: - a relationship that is perceived to connect a change to one observable to a change to another (e.g. when the orientation of the Stargate changes, so does the orientation of the "puddle" within it)
- an **agent**: - something that can be deemed to initiate change (e.g. the person who presses the "dial" button in the Stargate)

A key issue is that these notions are *subjective* - they are relative to an observer and a mode of observing and interacting.

About the assessed work - variety of different ways in which can relate to the module

- can choose your own topic for the assessment - which comprises a paper + a model
- can assign a weighting to the written and practical work in the assessment in ratio from 70:30 to 30:70
- can contribute through
    - model development (using a variety of tools available)
    - extension / commentary / analysis of existing models
    - extension of the tools

The module is wide-ranging in scope, covering many different themes (to include non-logicist foundations, educational technology, concurrent systems modelling and software development), and spanning issues both technical, methodological and philosophical. You will not be expected to understand everything in depth, or to master all the techniques.