

CS405 An Introduction to Empirical Modelling

Module Lecturers:
Meurig Beynon and Steve Russ

1

Empirical Modelling

What does Empirical Modelling (EM) entail?

principles and tools for constructing artefacts to
embody what is directly experienced

primarily interested in computer-based artefacts

2

Orientation

The approach, methods, emphasis and goals of Empirical Modelling (EM) are very different from conventional computer science, or programming.

The difference is to do with perspective, breadth, openness and the role of interpretation and experience. EM is compatible with, and important for, many branches of computer science and computing.

3

Principles & Tools

Principles:

analysis, and model construction, based on the
central concepts of observation, dependency, agency

Tools:

EDEN: an evaluator for definitive notations

the Abstract Definitive Machine (ADM)

JaM2: a Java API for dependency maintenance

4

Embodiment of experience

'artefacts that embody what is directly experienced':

- have observables that serve as counterparts of observables in the referent *in an experiential sense*
- exhibit similar indivisible links between changes to observables
- admit agency similar to that associated with the referent

5

Applications of EM

... potential, but with proof-of-concept

- engineering design
- education
- business
- humanities

- a new 'foundation' for computing practice
- observation-oriented software development

6

Plan for the module

Part 1: a practical introduction to EM based on EDEN

Principal focus: exposition and review of EM
Based on Concurrent Systems Modelling module

Part 2: a more detailed study of principles, concepts
and application areas (+ a little philosophy).

Will include more reflection on ideas of EM in
relation to conventional areas of CS

7

Support for the module

Major support for the work of the module this year is coming
from four PhD students who are acting as tutors:

Russell Boyatt,
Eric Chan,

Charles Care,
Antony Harfield

Sometimes helping in lectures, usually helping on practical
sessions on Fridays - make use of their help!

8

A reflective exercise

Is Computer Science really a science?

Goal is to decide 'yes', 'no', or 'don't know' + reasons.

Consider individually (or pairs) [5 mins]

Pool resources in groups with tutor [5mins]

Plenary session to assess any common themes or common
problems [5mins]

9

Notes: Is CS a science?

What makes a science? Experiments, theories, predictions,
connections with other sciences?

Consider physics, astronomy, geology, biology.

Natural philosophy, natural science, experimental science. Science
as body of knowledge (18C).

CS as science using computers as tools or instruments, or science
of the behaviour of computers themselves?

Does the universe, or nature, compute? Can CS be a natural
science?

Are any modules in CS degree like science? Are maths/logic
sciences? Why? Or Why not? Is there a science of
programming?

Is chip design and manufacture a science?

If CS is a science, what is it a science of? if not, what is it?

10

Resources 1

The EM webpage:

- extensive discussion about the nature of EM
- complete lists of publications for EM
- details of postgraduate theses from the EM group
- archive / screenshots of EM models
- information and documentation for tools
- instructions for downloading versions of tkeden

11

Resources 2

The CS405 on-line materials page

<http://www2.warwick.ac.uk/fac/sci/dcs/research/em/teaching/cs405/>

- will contain handouts from lectures and labs
- also various other useful materials and links

The EM archive, linked from the EM webpage:

See directory [/dcs/emp/empub/public/projects](#)

- a web repository largely consisting of EM models
- most of the models are recent, and can be downloaded

12

Resources 3

EM publications – index accessible via EM webpage
See directory /dcs/emp/empublic/publications

- an electronic library of EM publications: papers and talks

MSc92-9 – accessible via Teaching link from EM webpage

- a web-based compendium of the resource material for "Empirical Modelling for Concurrent Systems", a one week taught module delivered as part of a one year MSc programme run by the department from 1992-9.

13

EM research contributors

Postgraduate research personnel 1984-2005

Meurig Beynon, Steve Russ
Russell Boyatt, Antony Harfield, Charles Care, Eric Chan, Karl King
Ashley Ward, Chris Roe

Edward Yung, Mike Slade, John Buckle, Samia Meziani, Simon Yung
Alan Cartwright, Paul Ness, Dom Gehring, Patrick Sun, Richard Cartwright
Y-C Chen, Suwanna Rasmeequan, Soha Maad, Jaratsri Rungrattanaubol
Tim Heron, Ruyuan Wang, Allan Wong, J-P Dupont, K-C Tan

14

Summary of main ideas

EM is primarily concerned with specific and concrete situations
(cf. the emphasis on abstraction in classical modelling)

EM aims at creating artefacts that capture state-as-experienced

The complexity and subtlety of state-as-experienced is awesome,
and cannot be circumscribed

The flexibility of human interaction in the world contrasts with the
rigidity of the relationship between a typical computer system
and its environment

15

Things to do

1. Look at the main `jugs.e` file in the jugs models, see how it works, and experiment with those models.
2. Consult the `/dcs/emp/empublic/publications` directory for characterisations and illustrative examples of observables, dependencies and agents.
3. Reflect on how the concepts of agency, observation and dependency relate to some part of life with which you have a lot of experience.

16