Principles, Tools, Examples and Perspective

What is 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

Short 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 areas of computer science and computing.

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

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

Applications of EM

… potential, but with proof-of-concept

- engineering design
- education
- business
- humanities

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

Plan for the module

- Part 1: a practical introduction to EM based on EDEN
- Principal focus: exposition and review of EM based on Modelling with Definitive Scripts
- Part 2: modelling with many agents: principles and tools for concurrent systems
- Part 3: application areas with reflections on foundations of modelling and computing

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

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.

The role of tkeden in EM

- *So far illustrated …*
- HOW we can model using with tkeden
- power of dependency
- rich and flexible models (if a trifle odd)

WHAT is the character of models?

- WHY build models this way?
- are there principles to support their construction?
- why don't we have better models and tools?

Reflection on the Room model 1

- Features of the interaction
- state-based, timeless, unconstrained
- cartoon-like quality of the artefact
- Key idea:
- Create an artefact with which interaction can be contrived to resemble interaction with an external referent

Reflection on the Room model 2

- *Key idea:*
- *Create an artefact with which interaction can be contrived to resemble interaction with an external referent …*
- interaction and design guided by reference to an independent experience
-
- 'one experience knows another'
- William James, c.1902

Reflection on the Room model 3

- *Possible extensions of observation …*
- recording dimensions of the room (what area? what cost?)
- supplying the model to a disabled person as an interface
- elaborating object-based design for the room furniture

Reflection on the VCCS model 1

- … model development proceeds *from* state *to* behaviour
- contrast concept of snapshot (*from* behaviour *to* state: capturing at an instant what is perceived to change)
- with perception of identity and change emerging from dwelling in the experience of a particular state
- possible function of the artefact is to stimulate the modeller's imagination about purposes and scenarios for interaction
- [cf. "this is how the model is to be interpreted and used"]

Reflection on the VCCS model 2

- … ambiguity of viewpoints
- Who's doing the observing?
-
  - an external observer studying the VCCS?
  - the driver of the vehicle?
  - an engineer interested in design decisions
  - an EM modeller, studying the artefact-building?

Reflection on the VCCS model 3

- *… clock 'isn't switched on' …*
- ... can design speedo, car shape, profile of the hill
- ... change curSpeed (but what is speed without time?)
- to test the speedo, as an engineer might;
- to shape the speedo as a dashboard designer might;
- to check to see whether there's a zero stop or not

Reflection on the VCCS model 4

- *… clock is switched on*
-
- ... touch brake triggers the cruise control to switch off vs cruise control on is dependent on brake off
- ... granularity of observation in computing dynamics
- ... take account of load on vehicle as passengers get on or off etc
- ... driving in conditions where gravity is different
- ... maintaining cruise speed by dynamically changing the road
- … making the length of the vehicle depend on its speed

Possible views of modelling …

- 1) you're only building small models / programs
- 2) it won't scale up
- 3) object-abstractions would make tkeden much more powerful
- 4) global observables are too dangerous for real applications
- 5) it's essential to be able to assert and impose constraints
- 6) you need better unified notations, and a formal semantics
- 7) your notations are far too primitive and low-level

& why it has such flexibility …

- 1) tkeden is a brilliantly engineered piece of software
- 2) it's basically parametric design
- 3) it's basically constraint-based programming
- 4) it's a constructive form of algebraic specification
- 5) dependency is a very powerful programming construct
- 6) you're only building small specific models

An EM perspective on tkeden? 1

- tkeden is only a first prototype for supporting EM
- many deficiencies, but lack of OO probably isn't the primary one
- we have a lot to understand about the design of definitive notations, and this raises issues (such as mode of definition and mode of observation) for which there little precedent in traditional programming language design

An EM perspective on tkeden? 2

- the expressive power of EM stems primarily from its conceptual roots, not from the technical support afforded by current tools
- flexibility and richness in model-building using tkeden is intimately associated with the quality of our *construal* of experience

An EM perspective on tkeden? 3

- … **construal** …
- what agencies do we believe to be at work?
- what observables do we presume to mediate their interaction?
- what dependencies do we suppose govern their observations and action?
- Good construal ↔ reliable expectations re interaction

An EM perspective on tkeden? 4

- EM is primarily about *constructing artefacts that embody our construals* …
- Construals are
  - personal
  - provisional
  - particular to circumstances and needs

An EM perspective on tkeden? 5

- personal: Simon Gardner appreciates what a Formula 1 driver observes much better than I do (cf [racingGardner1999])
- provisional: I have an idea that eating carrots helps you to see in the dark, but I may be wrong
- particular to circumstances etc: I don't / don’t need to understand in detail how a piano action works, even though I play the piano

Principles, Tools, Examples and Perspective

What is 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

Short 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 areas of computer science and computing.

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

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

Applications of EM

… potential, but with proof-of-concept

- engineering design
- education
- business
- humanities

- a new ‘foundation’ for computing practice
- observation-oriented software development

Plan for the module

- Part 1: a practical introduction to EM based on EDEN
- Principal focus: exposition and review of EM based on Modelling with Definitive Scripts
- Part 2: modelling with many agents: principles and tools for concurrent systems
- Part 3: application areas with reflections on foundations of modelling and computing

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

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.

The role of tkeden in EM

- *So far illustrated …*
- HOW we can model using with tkeden
- power of dependency
- rich and flexible models (if a trifle odd)

WHAT is the character of models?

- WHY build models this way?
- are there principles to support their construction?
- why don’t we have better models and tools?

Reflection on the Room model 1

- Features of the interaction
- state-based, timeless, unconstrained
- cartoon-like quality of the artefact

- Key idea:
- Create an artefact with which interaction can be contrived to resemble interaction with an external referent

Reflection on the Room model 2

- *Key idea:*
- *Create an artefact with which interaction can be contrived to resemble interaction with an external referent …*

- interaction and design guided by reference to an independent experience
-
- 'one experience knows another‘
  - William James, c.1902

Reflection on the Room model 3

- *Possible extensions of observation …*

- recording dimensions of the room (what area? what cost?)
- supplying the model to a disabled person as an interface
- elaborating object-based design for the room furniture

Reflection on the VCCS model 1

- … model development proceeds *from* state *to* behaviour

- contrast concept of snapshot (*from* behaviour *to* state: capturing at an instant what is perceived to change)
- with perception of identity and change emerging from dwelling in the experience of a particular state

- possible function of the artefact is to stimulate the modeller's imagination about purposes and scenarios for interaction
- [cf. "this is how the model is to be interpreted and used"]

Reflection on the VCCS model 2

- … ambiguity of viewpoints
- Who's doing the observing?
-
  - an external observer studying the VCCS?
  - the driver of the vehicle?
  - an engineer interested in design decisions
  - an EM modeller, studying the artefact-building?

Reflection on the VCCS model 3

- *… clock ‘isn’t switched on’ …*
- ... can design speedo, car shape, profile of the hill
- ... change curSpeed (but what is speed without time?)
- to test the speedo, as an engineer might;
- to shape the speedo as a dashboard designer might;
- to check to see whether there's a zero stop or not

Reflection on the VCCS model 4

- *… clock is switched on*
-
- ... touch brake triggers the cruise control to switch off vs cruise control on is dependent on brake off
- ... granularity of observation in computing dynamics
- ... take account of load on vehicle as passengers get on or off etc
- ... driving in conditions where gravity is different
- ... maintaining cruise speed by dynamically changing the road
- … making the length of the vehicle depend on its speed

Possible views of modelling …

- 1) you're only building small models / programs
- 2) it won't scale up
- 3) object-abstractions would make tkeden much more powerful
- 4) global observables are too dangerous for real applications
- 5) it's essential to be able to assert and impose constraints
- 6) you need better unified notations, and a formal semantics
- 7) your notations are far too primitive and low-level

& why it has such flexibility …

- 1) tkeden is a brilliantly engineered piece of software
- 2) it's basically parametric design
- 3) it's basically constraint-based programming
- 4) it's a constructive form of algebraic specification
- 5) dependency is a very powerful programming construct
- 6) you're only building small specific models

An EM perspective on tkeden? 1

- tkeden is only a first prototype for supporting EM
- many deficiencies, but lack of OO probably isn't the primary one
- we have a lot to understand about the design of definitive notations, and this raises issues (such as mode of definition and mode of observation) for which there little precedent in traditional programming language design

An EM perspective on tkeden? 2

- the expressive power of EM stems primarily from its conceptual roots, not from the technical support afforded by current tools

- flexibility and richness in model-building using tkeden is intimately associated with the quality of our *construal* of experience

An EM perspective on tkeden? 3

- … **construal** …

- what agencies do we believe to be at work?
- what observables do we presume to mediate their interaction?
- what dependencies do we suppose govern their observations and action?

- Good construal ↔ reliable expectations re interaction

An EM perspective on tkeden? 4

- EM is primarily about *constructing artefacts that embody our construals* …

- Construals are
  - personal
  - provisional
  - particular to circumstances and needs

An EM perspective on tkeden? 5

- personal: Simon Gardner appreciates what a Formula 1 driver observes much better than I do (cf [racingGardner1999])

- provisional: I have an idea that eating carrots helps you to see in the dark, but I may be wrong

- particular to circumstances etc: I don't / don’t need to understand in detail how a piano action works, even though I play the piano