WV. 015 178 8 Belvedere Rd, Earlsdon

1. General principles

Conventionally, we associate supplying foundations with explaining general phenomena in terms of an unambiguous universally agreed formal framework. There are obviously limitations on what can be achieved in this way. Such foundations can't account for experience that is private in nature, sensory dependent, subjective and culturally dependent. Conventional foundations can't account for experience

Well-recognised that it's difficult to bring much of the agenda of modern computing within the scope of conventional foundational framework ... must accept either that a quite radical new framework is required, or that there is no adequate framework within which to pursue a major part of this agenda New framework or no framework

Twofold concern

1. what kind of programme can deliver what is required of foundations for modern computing?
2. in what sense can traditional foundations be sensibly extended?

What do you expect of formalisation?

- formalisation is a means of *communication*. We describe an experience in terms of concepts that others can interpret unambigiously.
- one role of formalisation is to give a *precise description* of a system e.g. so that it can be reconstructed.
- formalisation as a way of *reasoning* about a system.

Communication

Merit of formal frameworks is presumed independence of observer: universal nature

There are examples of experiences that are private: how would sighted person convey contents of a sketch to a blind person?

In general, relation between formal and experiential is similar to that between a computer program and its interface: the relationship between mechanism and its interpretation needs to be established informally

Our approach

Constructs models that imitate the phenomena they represent directly in experiential terms: e.g. sketch resembles a building. Such models are often useful in communication, but rely heavily on contextual factors – sketch is only helpful to a sighted person when it isn't dark etc. The models that experimental scientists construct of phenomena ("construals"?) are of this kind. The private nature of such models is highlighted by the issues concerning communication raised by Gooding's work. The use of private models can also require particular skill that can only be shared to a certain degree and it takes a long time to attain public acceptance. Consider e.g. devices such as the microscope. Eyetest is an example of this kind of activity.

Precision

Formal systems are often used to give unambiguous descriptions of systems or phenomena. The interface to the user is an issue here: there is nothing to connect the abstract model with experiential aspects. In general there's a very high degree of ambiguity in how things are experienced – cf functional program interface. The most appropriate way to express how something is to be experienced is to emulate it: an animated sketch of the screen layout, or a display of a colour. The limitations of formal systems are typically overcome (and disguised) by developing and invoking conventions for experiential interpretation, and building reliable devices (such as a colour screen display) to convert a formal specification into a sensation. Another aspect of the use of formalism is that involves considerable simplification associated with abstraction and idealisation. Only those qualities that are communicable and are subject of universal theory are typically captured.

Our approach

Shift emphasis onto creating experiences that imitate what is being described. Typically involves observing a phenomenon from several different perspectives

Manner in which the model of a particular client's sight is developed is significant: not derived solely from theory but from an empirical process that essentially involves the client.

Can make use of personal experiences and languages in establishing a precise model. Cf. subjective activities, such as writing a poem, or giving a programme to a piece of music. What we find similar to the experience of listening to a piece of music is very personal and particular to our experience. Also reflects our culture.

Reasoning

Formal descriptions are a means to encapsulate a complete body of knowledge about a phenomena – all things of interest that are true of the phenomenon are consequences of what has been asserted. This is a root to giving order to (rationalising) information about a phenomenon. Development of such involves a process of circumscription: this is all that is to be said about this system in some respect. Typically not all users have such a view of system as to permit effective reasoning: may have too imperfect a view on it – things will appear non-deterministic to one observer that are explicable to another. (The problem of "ignorance representation".)

Our approach

Regarding reasoning, we don't have the same commitment as the logicians' description of a system. We are only concerned with what we believe to be true of our experience so far, not a universal / absolute belief. This admits the possibility of capturing subjective experience and ideas.

incrementability – we can incrementally build up a model by introducing new modes of observing a phenomenon. Our way of constructing a model is from bottom up, changing understanding of the observations is easy to handle.

Issues and examples for illustration

[Aesthetic judgement]
[tree with special leaves]
→ Can modify the detail: don't have to define top-down / bottom-up exclusively.
[colour is a private experience dependent on our sensory device]
[different colour mixing models]
[left eye has different sensory to the right eye for the same colour]
[following instruction to interpret/reconstruct a system. How about if the instruction is in French?]
[drawing a sketch for a blind]

Summary

Our approach is more general in some respects:

- What we are doing is not necessarily to do with communication nor reasoning.

• Regarding communication, we are free to use language that is private to our own interest, e.g. ways of sorting pears or jigsaw pieces.

Dependence on observer in our methods is reflected in fact that our current tools are not suitable for blind people; they can't see the graphics.

• Regarding reasoning, we don't have the same commitment as the logicians' description of a system. We are only concerned with what we believe to be true of our experience so far, not a universal belief.

Relevant issue for understanding relation between our methods and classical approach

Is logic independent of sensory experience?

Logic is rather like a *reliable* computational device or a scientific instrument where a lot of its activities do not have sensory significance. The sensory experience is analogus to the user of a program whereas logic / the activities inside a machine is viewing from the angle of the programmer.

Reliability fulfils expectation, so long as the viewpoint for which reliability is measured is the same viewpoint as the expectation is raised. For example, we may have difficulty in talking about reliability and expectation in the case of the invention of the microscope.

We want to bring about a shift of emphasis from specification of machine to understanding relationship between what the machine does and its interpretation.

We associate formality with reliable devices (scientific instruments) that imitate independent phenomena.

What kind of foundations can we supply?

2. Fundamental concepts

observables

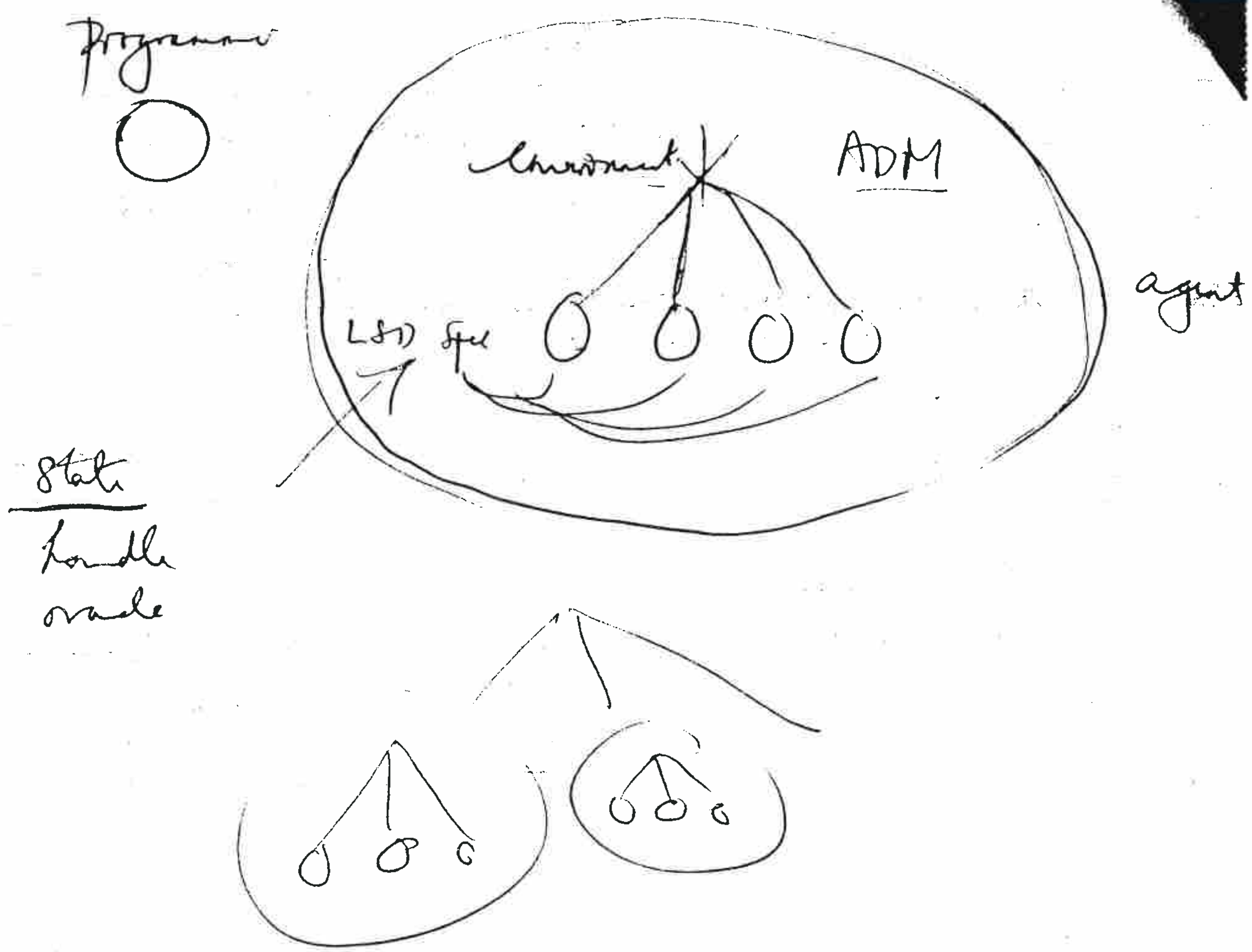
Observables are features of phenomena that have a distinguishable identity but can attain many values. Value is typically observed by an experimental procedure, though can also be directly perceived. (Observation by an experimental procedure is process that reduces what can't be directly perceived – doesn't appeal to human senses, or is too elusive etc – to things that can be perceived.)

Observables are typically associated with a context. Association of observables may reflect indivisibility, or association with an entity. (Criterion for association with common entity – existing whilst particular object / agent is present.) The same identity can give rise to many observables, depending on how many observers are involved. Observables can also be associated with groups of identities e.g. "has the meeting finished?"

indivisible relationships

Certain groups of observables are associated in as much as a change to one will necessarily cause changes to occur in others within the group. . Examples include: mechanical linkages, content relations (match won = ball crosses boundary) ... indivisible relationships are associated with independent threads of state-change

agency


Programme
ADM
agent
LSD
Spec
State
formula
oracle

Changes to the values of observables are attributed to agents. Threads of state-change enable us to distinguish between actions of one agent and another. Combinations of agents interacting in an environment supplied by indivisible relationships and constraints (as established by a context for observation) can serve as another agent

programmability

The elements of programmability are: one agent (the programmer) is in a position to prescribe the activities of other agents (the components of the computing system). These components have to exhibit reliable stimulus-response patterns (the machine-code) that can be set according to the set of initial instructions of the programmer (the program). To describe an appropriate generalisation, we also need to consider the role of the environment in which the component agents interact - this is typically preconceived in classical computer programming, but has to be specified when considering the specification of an engineering system. The programmer & system-of-computing-agents pattern occurs in many guises: e.g. manager of a group of design agents, choreographer of ballet, composer for orchestra, experimentor involved in engineering design. In general, this framework involves putting different degrees of emphasis on the various ingredients.

← can't prescribe exactly what actor / dancer / musician does or takes like

Abstraction is agent in the role of programmer of system of component agents (hence a hierarchical agent relation). The programmer agent specifies an environment (in form of an ADM environment, with definitions and constraints to specify relations between observables) within which the component agents interact, possibly introducing additional agents, and possibly retaining measure of privilege to participate in the interaction. In some circumstances, there may be many such environments, each corresponding to different scenario and mode of observation.

Factors in the interpretation of the programming activity are:

> expectation - does the programmer know what degree of reliability to presume of the components of the system?
>
> commitment - to what extent does the programmer agent retain privileges to modify and intervene in interaction.

Illustrative exanples

Reactive system

Programmer = engineer / system designer, component agents are human, electronic and electomechanical components in the system. In this context, might use many different environments to gain empirical insight into how/when/whether system will function reliably. Different degrees of presumption of reliability re action of component agents.

(Hopefully) particular interpretations within the above framework

comprehension of states: a system of agents in an environment is associated with a mode of observation - equivalent to a comprehension of component agents ... this comprehension of observables effectively creates scope for another level of observation. For instance, can consider observation of the entire OXO board as comprehension of observation of individual squares. Can also consider observation of progress of meeting in conjunction with iindividual perceptions: e.g. acting on basis of agenda status as well as reactions of other participants in meeting. important that what is defined by comprehension can be observed by the participants in some cases ... components can usurp the role of the observer

↳ member of meeting may react to procedural aspects of its activity decomposition of agents: can regard a programmed system of agents as a composite agent - exhibiting stimulus-response patterns through corporate effect ... here have refinement picture in place

structure : the context for interaction established by components and environment can be more or less dynamic according to the status of the agents involved. Expect through specialisation and extension of the hierarchy to be able to specify a variety of different structures / object like abstractions / functions for underlying algebra (?)

HOOPs

Other issues relate to behaviour of the system over time

circumscription of behaviour

it may be that the programmer agent has an overall view of the behaviour of the system, expressed in terms of universal statements about possible states and state-transitions ... this can be expressed either in terms of how the activities of the individual agents combine, or in terms of absolute knowledge about overall properties of the system (such as might be expressed in temporal logic). A prerequisite of circumscribing the behaviour is that the programmer agent should be either without privilege to perform experiments (defn of experiment can be expressed in terms of expectation and privilege?) or abdicate privileges to intervene. *Being without privilege to perform actions* is an empirical concept.

Circumscription can give rise to observables that can be perceived by the agents within the system: e.g. velocity etc - this is the basis of hybrid modelling (discrete event + continuous modelling)

expectation

Knowledge of system behaviour can also be expressed in terms of sequences of events (event is modelled as *instantaneous change of state*). This can be a feature of the observables that are accessible to the programmer agent and in principle to all agents in the system also. Examples: protocols between agents such as the guard / stationmaster etc, procedural stages in the agenda at a meeting, preconceptions of the programmer about what kind of interaction and interpretation are involved in the computation.

reliable devices

in connection with expectation, significant concept is that particular agents may exhibit predictable stimulus-response patterns ... reliability is the key to programmability

commitment

commitment relates to the interventions of the programmer agent: can see the process of shaping the behaviour of a programmed system of agents as involving experimental activity, actions to prescribe the environment for interaction and the protocols for the interacting agents which typically leads to framing a system that has a progressively higher degree of autonomy. In the system design process, commitment involves making design decisions that fix aspects of the environment and agent protocols, and also involves shaping of expectations. Commitments are associated with activities such as implementation that entail substituting particular agents to fulfil abstract roles. Commitments influence what can be specified: e.g. can introduce object structures, can make presumptions about what patterns of state-transition will arise, can circumscribe features of the environment - e.g. to justify presumption that certain observables are present, or are appropriately related

underlying algebra / user-defined operators

3. Relationship to the 4 areas

Programming paradigms

Analysis of interfaces and stimulus-response interactions between reliable state-changing agents

Programmability of agents

Software/Systems Engineering

Agent as designer and engineer of agents (reliable devices and interaction patterns)

Interacting systems of agents as in concurrent engineering

Commitments and expectations

Foundations

Formality as physical realisibility

Respectable state and authentic variable and function

Artificial Intelligence

Empirical roots of intelligent behaviour: extent to which reflects learning from example

Origins of language, models of non-verbal communication

References

Gooding
History of Science book (microscope)

Programming paradigms

Programming paradigms for graphics

Questions

Is it appropriate to expect a programming paradigm *for graphics*?

Considerations:

"horses for courses" view of paradigms is suspect
difficult to express how paradigms interface and interact

in what sense is *graphics*
sensibly characterised as a domain for application of programming?
why wouldn't a paradigm for graphics be good for other things ?
what is particularly distinctive about graphics?

plausible that
graphics has many different aspects, favouring multi-paradigm
programming for graphics must interact with other programming tasks

Our position

There is a distinctive aspect of programming closely bound up with graphics viz. *metaphorical representation of state using computers*

[graphics associated with visual metaphors:
v. important because dominant sensory importance of sight to people]

existing paradigms don't handle this well, since not state-orientated

importance of such use of graphics extends to design [cf Harel]

believe that in principles associated with empirical modelling
have powerful ways to address metaphorical representation of state
with implications for specifying interaction between agents in general

In particular, can use these principles
to construct interfaces between program components
to deal with interface between user and computer
to assist the designer with perceptualisation
in the development of complex systems

Systems Engineering

Problems of systems engineering expressed by Harel and Brooks

"Semantics" for agent-oriented models

1) is there an agent present? (unidentified)

? does ∃ an observable that changes in a fashion uncorrelated with my actions.

ANSWER in any case { empirical / not Y/N sharply / viewpoint context dep.

2) Am I the sole agent in this environment?

? ~~does~~ is change to any observables ~~change~~ correlated with changes that I ~~choose~~ initiate.

HANDLE.

3) Does a particular ~~an~~ agent exercise conditional control over the value of an ~~agent~~ observable?

4) Is ~~there~~ a particular agent present?

? is there a correlation between presence of observable char. of an identity and

? is there a correlation between presence/absence of observables and disposition to change in the model

5) is an observable constant or subject to variation?

? does an observable undergo change or not over period of observation.

ORACLE

6) is an observable an oracle for an agent?

? is ~~there a correlation~~ pattern of behaviour of an agent intention attributed to an agent

correlated in its nature with value of observable

7) How many agents are present?

Many experimental approaches

a. particular changes to observables characteristic of particular agents (testify to presence)

b. simultaneous change to several observables not correlated individually —

c. pattern of system behaviour characteristic of particular combinations of agents.

b. Tap on window at same time as light switched on at same time as bumps in middle of room.

a) door is unlocked (who has key?)

c) 2-body vs 3-body system / discovery of Pluto

Relevant issues Has the super agent prior perception (?)?

What is understood re context?
identity of observable ⟷ integrity of context

(cf variables paper)