Is the PIE past its sell-by date?

Alan Dix
Lancaster University, UK
www.hcibook.com/alan/talks/mdh-fest-2004

20 years ago
the PIE and all that

interaction models

• generic models of classes of system
• mainly to aid understanding of general issues

the PIE model (Dix & Runciman, 1985)

• ‘minimal’ model of interactive system
• focused on external observable aspects of interaction

properties - WYSIWYG

\[ \exists \text{predict} \in (D \rightarrow R) \text{ s.t. predict } \circ \text{display} = \text{result} \]

• but really not quite the full meaning

proving things - undo

\[ \forall c : c \text{ undo } \sim \text{ null } \]

only for \( c \neq \text{undo} \)

\[ S_0 \xrightarrow{a} S_a \xrightarrow{\text{undo}} S_0 = S_a \]

\[ S_0 \xrightarrow{b} S_b \xrightarrow{\text{undo}} S_0 = S_b \]
the cube (Mancini, 1997; Dix & Mancini, 1997)

- generic framework for layered systems
- undo, back and history

still going strong

- first book - citeseer.nj.nec.com citation count

... and more

- chapter in Carroll theory book
- Upside down Ys and algorithms ...
- in a leading textbook!!
- ... and the Italians love it!

λ

further back

a formal methods success story

problem

- context
  - mid 80s
  - local authority DP dept
- transaction processing
  - vast numbers of users
  - order processing, pos systems etc.
  - COBOL!
- existing programs ... didn’t work

what happens

1. user edits form
2. message goes to TP engine
3. passed to application module
4. which processes the message
5. and prepares new screen
6. which is sent to the user
7. ....
**structure of programs**

- `if ..`
- `if ..`
- `if ..`
- `if ..`
- `if ..`
- `if ..`
- `if ..`

**why?**

- program is trying to work out what is happening!
  - standard algorithm
    - program counter implicit
  - TP, web, event-based GUI
    - need explicit dialogue state

**many users - one application**

- user
- terminal
- corporate database
- central server

**mixed up state**

- user A starts multi-screen search list
  - application remembers ‘next_record’
- user B starts multi-screen search list
  - application overwrites ‘next_record’
- user A selects ‘next screen’ ...
  - application uses remembered ‘next_record’
- user A sees next screen of B’s search!

**solution?**

- flowchart!
- not of program ... but of dialogue
- a formal dialogue specification!
- hand transformed to boiler-plate code

**details ...**

- miniature screen sketch
details ...

- minimal internal details

C2

C3

other

C2 answer?

C3 delete record

lessons

- useful
  - addresses a real problem
- communication
  - mini-pictures and clear flow easy to talk through with client
- complementary
  - different paradigm than implementation
- fast pay back
  - quicker to produce application (at least 1000%)
- responsive
  - rapid turnaround of changes
- reliability
  - clear boiler plate code less error-prone
- quality
  - easy to establish test cycle
- maintenance
  - easy to relate bug/enhancement reports to specification and code

changing nature of the interface

- ubiquitous computing
  - computers everywhere!
- many simple systems
  + complex interactions
- sounds like a job for .... formalism

formal futures

ubiquity and physicality

study the old to design the new

- work with Masitah Ghazali
- look at ordinary consumer devices
  - washing machine, light switch, personal stereo
- why?
  - we are used to using them ourselves
  - they have been ‘tested’ by the marketplace
  - they embody the experience of designers

physical-logical connections

- device
  - physical input
  - physical movement
  - physical sound
  - physical light
- logical system
  - physical-logical mappings
  - system
  - system
  - system

λ

λ

λ

λ

study the old to design the new

- work with Masitah Ghazali
- look at ordinary consumer devices
  - washing machine, light switch, personal stereo
- why?
  - we are used to using them ourselves
  - they have been ‘tested’ by the marketplace
  - they embody the experience of designers

physical-logical connections

- device
  - physical input
  - physical movement
  - physical sound
  - physical light
- logical system
  - physical-logical mappings
  - system
  - system
  - system

λ

λ

λ

λ

formal futures

ubiquity and physicality

study the old to design the new

- work with Masitah Ghazali
- look at ordinary consumer devices
  - washing machine, light switch, personal stereo
- why?
  - we are used to using them ourselves
  - they have been ‘tested’ by the marketplace
  - they embody the experience of designers

physical-logical connections

- device
  - physical input
  - physical movement
  - physical sound
  - physical light
- logical system
  - physical-logical mappings
  - system
  - system
  - system
fluidity

- 'naturalness' of device-logical mapping

device & logical states

- switch
- light

user pushes switch up and down

DOWN

ON

OFF

UP

compliant interaction

- device & logical states
- user pushes switch up and down
- light OFF, ON

incidental interaction

- car courtesy lights
- turn on
  - when doors unlocked/open
- turned off
  - after time period
  - when engine turned on

driver's purpose is to get into the car

incidentally the lights come on

issues and process

- no accepted methods but ... general pattern
- uncertainty
  - traditional system due to errors
  - sensor-based so intrinsic to design
  - uncertain readings, uncertain inference
  - usually control non-critical aspects of environment
- process ... identify
  - input – what is going to be sensed
  - output – what is going to be controlled
  - scenarios = desired output and available input

designing a car courtesy light

- available input
  - open door, car engine
- desired output
  - light
- identify scenario
- label steps
  0. don't care
  1. deactivate alarm
  2. walk up to car
  3. key in door
  4. open door & take key
  5. get in
  6. close door
  7. adjust seat
  8. find road map
  9. look up route
  10. find right key
  11. key in ignition
  12. start car
  13. seat belt light flashes
  14. fasten seat belt
  15. drive off

illegal to drive with interior light on
in summary ...

after 20 years

the PIE is still fresh