Lecture 8: Designing complex systems as interaction spaces
Case studies on applying theory to hard HCI problems
Overview of the course

• Theory driven approaches to HCI
• Design of visual displays
• Goal-oriented interaction
• Designing efficient systems
• Designing smart systems
• Designing meaningful systems (guest lecturer)
• Evaluating interactive system designs
• **Designing complex systems**

Rescheduled!
Next Monday (20th Feb)
LT1, 3:00 pm
What are some things that make designs complex?

- How complex is the domain?
- How many different tasks might a user perform?
- How well defined are the outcomes? (Wicked problems, L3)
- How easy is it to understand each part?
- When the parts are put together how easy is to guess the behaviour?
- Does the system do things when the user isn’t there? (Attention Investment from L3)
Designing tasks vs interaction spaces

Consider a (slightly silly) API for sending a message:

(1) `sendTheRightMessage()`
(2) `sendMessage(Enum message)`
(3) `sendMessage(String message, Urgency status)`

- Naive design would produce (1). Complex systems tend to be built out of reusable components that the users configure (2,3)
- Building this kind of system involves discussing tradeoffs as well as detailed design decisions
- This is the kind of interaction space that most of you will build: Programming languages, APIs, AI systems
Broad brush techniques

- Descriptions of specific actions result in a ‘death by detail’
- Don’t describe specific actions with an interface
  - Describe interaction with a level of *analytical distance* from the interface
  - Use an *analytical frame* which is a way of structuring a description of an interaction
  - The description can then be compared to an ideal for a domain to become a critical perspective (see Lecture 1)
- The right level of detail resembles an object-oriented design pattern, but for *human behaviour* rather than software.
Cognitive Dimensions of Notations (CDNs): Analytical Frame

A user performs an activity Interface containing notations, described along a number of dimensions
Cognitive Dimensions of Notations (CDNs): Analytical Frame

A user performs an activity on an interface containing notations, described along a number of dimensions.
CDNs: A simple example

Me  Exploratory Design  Interface (Google Slides)

“Can I make my slides less ugly?”
CDNs: A simple example (Demo)

- One described change “Make the font of the headings Comic Sans”
  - Select the first slide, change the font
  - Select the second slide, change the font
  - Yawn.
- This is repetition *Viscosity*, many operations to perform one change
- Design maneuver: Introduce an *Abstraction* (master slide), decreases *Viscosity*, but increases *Premature Commitment*
- NB: CDNs analysis is meaningless independent of an interface.
CDNs: Activities

**EXPLORATION**
Manipulating both information and structure. Exploration involves manipulating and changing both the content and the structure of the information.

**MODIFICATION**
Changing structure only. Modification involves changing the structure of the information, but keeping the content the same. Attention is on structuring.

**TRANSSCRIPTION**
From one notation to another. Transcription involves copying information from one notational form to another, often between different media as well.

**INCREMENTATION**
Adding new items. Incrementation involves adding new content, but leaving the information structure alone.

**READING**
Seeking information or gist. Reading often involves changing the interaction structure, but seeking information or gist is often for reference or understanding.
CDNs: Dimensions

- **Role Expressiveness**: How much elements suggest their purpose
- **Consistency**: Similar meanings, similar syntax
- **Premature Commitment**: Constraints are the order of decisions
- **Simplicity**: To change the interface, one option
- **Add age to knowledge giving weightage**
  \[
  \text{age} + \text{age}
  \]
- **Diffuence**: The amount of no routes of information
- **Useful Awkwardness**: Thinking-hard is sometimes useful
- **Abstraction**: Solutions for general
- **Synopses**: Overview of the whole
- **Hidden Dependencies**: Unexpected relationships
- **Hard Mental Operations**: Some things are just hard
- **Provisonality**: Eights of commitment to Sheila
- **Legibility**: Readability of the notation
- **Closeness of Mapping**: Connections in the domain being represented
- **Progressive Evaluation**: Evaluation through the mapping
- **Secondary Notation**: Escape from notation
- **Juxtaposition**: Difference through comparison
CDNs: Profile

- Activity
  - Low
  - Medium
  - High

- Dimension
  - Ideal value of the dimension for the profile
CDNs: Profile
Case Study: Dynamo’s type system
Code Block

Point.ByCoordinates(0,0,20);
Code Block
```
Point.ByCoordinates(0,0,0);
```

Code Block
```
Line.ByStartPointEndPoint(x,y);
```

Code Block
```
Point.ByCoordinates(10,0,0);
```
Dynamo

• Language for exploring building designs
• Includes a constructor `Point(x, y, z)` and array literal syntax `[1,2]

Design question for discussion:

“What should `Point([0, 1, 2], 10, 10)` do?”
- What activities are important?
Intelligent systems as interaction spaces

“Any sufficiently advanced technology will be indistinguishable from magic”
- Arthur C. Clarke, 1962
Individual interactions with AI systems often seem quite trivial to perform, even if we don’t know how we go about doing them.

Because we don’t know how they work, if something goes wrong there’s no obvious way of intervening.
AI can be magic at a distance

Having developed the capability to perform the task once, without human attention, we can now achieve results that would otherwise be impossible.

E.g. ‘find all the cats on the internet’
AI can feel like magic at a distance

Applying the same technique to animal behaviour analysis
How do you trust magic?
How do you control it?

If this was the result you saw - would you believe it?
A simple example of programming with ML

- Research in 2011 by Sumit Gulwani at Microsoft Research
- “Synthesises a program from input-output examples”
  - How do you choose the examples? (Premature commitment?)
  - How do you know what will happen? (Progressive evaluation?)
- Now Excel FlashFill
  - Paste a list of semi-structured text data into the left column
  - Type an example transform result in top cell to the right, then <Enter>
  - Press <Ctrl+E>
Conversational agents

- Do they build a user model, goal model or task model?
- Will this be more or less complex than FlashFill?
- How can you see that model?
  - i.e. what is the notation?
- How could you modify the model?
  - … in response to errors (yours, or the system’s)
  - … if you change your goals?
- Does having a ‘body’ help?
  - (remember metaphor)
Human issues in machine learning

• Ethics and accountability
  • automating and/or justifying bias and prejudice
• Digital humanities
  • treating text and images as meaningful and sophisticated
  • (rather than just statistical fodder)
• Reward
  • who does the intellectual ‘work’ of providing training corpus content, data labelling, how are they paid, and where do the profits go?
Some [other] current research problems
VR and AR still use visual correspondence
Programming, or direct manipulation?

- Many Internet of Things (IoT) devices have physical switches etc
  - But how do you define configuration, policy, future action?
  - Now we need a notation - or a programming language
- Remember behavioural economics and attention investment
  - Even around your house, bounded rationality happens
Global challenges

• Is knowledge infrastructure built to ...
  • … prioritise low income populations
  • … advance Sustainable Development Goals (human rights, education etc)?
Further interest...

- Part II: Project
- Part II/Part III Computer Music (not in 2023)
- Part II/Part III Advanced Graphics
- Part III: Interaction with Machine Learning
- Research Skills: Working with artists and designers; How to interpret experimental results; Introduction to qualitative research methods; How to design surveys; Assessing the quality of experience
- Alan’s new book: “Moral Codes: Designing software without surrender to AI”
  https://moralcodes.pubpub.org