Tasks = Data + Action + Context
automated task assistance through
data-oriented analysis

Alan Dix
Lancaster University


today I am not talking about ...

• intelligent internet interfaces
  fuzzy personal ontologies and
  structure from folksonomies

• situated displays, eCampus,
  small device – large display interactions

• fun and games, artistic performance,
  slow time

• physicality and design, creativity and bad ideas
  + modelling dreams and regret!!
... or even lots of lights

http://www.hcibook.com/alan/projects/firefly/

one morning at breakfast

Fir0002: http://commons.wikimedia.org/wiki/Image: Cornflakes_with_milk_pouring_in.jpg
**general message**

real tasks are complex:
  habitual, reactive, considered, situated
some say:
  too subtle and too nuanced to model
I have said:
  in humility try  
  \[(\text{Tamodia 2002})\]
but if it is hard for a human analyst ...
... automated analysis for task assistance ... !!!!!

**people**

Athens: Akrivi, Costas, Giorgos, Yannis, +++
Lancaster: Azrina, Devina, Nazihah, Stavros, +++
Madrid: Estefania, Miguel
Rome: Antonella, Tiziana, +++

plus the old aQtive team
plans ...

0. to make mug of tea

1. boil kettle
2. put in tea bag
3. pour in hot water
4. add milk

4.1. fetch from fridge
4.2. pour into mug

Plan 4.
if milk not out do 4.1
then do 4.2
pre-planned sequence

Plan 4.
if milk not out do 4.1
when milk in hand do 4.2
reactive / environment-driven

kinds of sequenced activity ...

<table>
<thead>
<tr>
<th></th>
<th>pre-planned</th>
<th>environment-driven</th>
</tr>
</thead>
<tbody>
<tr>
<td>explicit</td>
<td>known-plan</td>
<td>means-end</td>
</tr>
<tr>
<td>implicit</td>
<td>proceduralised / routine / habit</td>
<td>stimulus-response reactive</td>
</tr>
</tbody>
</table>
### learning ... but ...

<table>
<thead>
<tr>
<th>explicit</th>
<th>pre-planned</th>
<th>environment-driven</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>known-plan</td>
<td>means-end</td>
</tr>
<tr>
<td>implicit</td>
<td>proceduralised / routine / habit</td>
<td>stimulus-response reactive</td>
</tr>
</tbody>
</table>

### principle of parsimony

embodied mind theorists:
- humans fitted for perception–action cycle

Andy Clark:

“In general evolved creatures will neither store nor process information in costly ways when they can use the structure of the environment and their operations on it as a convenient stand-in for the information-processing operations concerned.”

only do inside your head what you can’t outside of it
representation everywhere

environment
  milk in the hand

plans
  after pouring tea ...

context
  in the middle of preparing grapefruit

Environment

data driven interaction
onCue in action

user selects text and copies it to clipboard
slowly icons fade in

the dancing histograms are very useful at getting out some of the textile sites you want to see. You can go to x’s page at http://www.hiraeth.com

kinds of data

- short text — search engines
- single word — thesaurus, spell check
- names — directory services
- post codes — maps, local info
- numbers — SumIt! (add them up)
- custom — order #, cust ref ...
- tables — ...
class of systems ‘data detectors’

- late 1990s
  - Intel selection recognition agent
  - Apple Data Detectors (Bonnie Nardi)
  - CyberDesk (Andy Wood led to onCue)
- recently
  - Microsoft SmartTags
  - Google extensions
  - Citrine – clipboard converter
  - CREO system (Faaberg, 2006)
- way back
  - Microcosm (Hypertext external linkage)

\{ syntactic / regexp \}
\{ 'semantic' / lookup \}
Snip!t origins

- MSc project 2002 (Jason Marshall)
- studying bookmarking
  - focus was organisation
- exploratory study
  - found users wanted to bookmark sections
- so one evening Alan has a quick hack
  ... and about once or twice a year since
- now being used for other projects
- live system ... try it out

Snip!t

1. users selects in web page and presses “Snip!t” bookmarklet
2. Snip!t pops up page with suggested things to do with the snip (and saves it for later, like bookmark)
Snip!t

recognises various things e.g. dates

play yourself at www.snipit.org

bringing it together

Kinds of recogniser:

syntactic – regular expression / patterns / structure
e.g. post code, email address

lexicon – large look up tables
e.g. countries, internet TLD

hybrid – popular first names => full name

+context – telephone number ... what country??
architecture

- server-side ‘intelligence’
- recognisers + services again
- different kinds of recogniser chaining:
  - from semantics to wider representation
    e.g. postcode suggests look for address
  - from semantic to semantic
    e.g. domain name in URL
  - from semantic to inner representation
    e.g. from Amazon author URL to author name

representation vs. semantics very important

personal ontologies

- all use ‘general’ categories:
  - post code, name, place
- linking to personal ontology
  - users own entities and categories
- how to build?
  - by hand (during useful interactions)
  - automatically (mining files, emails, etc.)
    - e.g. Gnowsis and other semantic desktop projects

me

Azrina

Geoff

married

Devina

member

Project: TIM

member

distributes

supervises
Context

what to do and what to do it to

spreading activation over ontology

long-term modification of schema relation weights

spread activation through relation instances

weaker spread through 1-m links than m-1

initial activation through use

instances
context in forms

but what is the relationship?
maybe semantic markup on form
   – good SemWeb style ... but not very personal
... or more inference ...

context in forms - inference

match terms in form to ontology
look for ‘least cost paths’
   • number of relationships traversed, fan-out
context in forms - inference

match terms in form to ontology
look for ‘least cost paths’
  • number of relationships traversed, fan-out
later suggest based on rules

Sequence
from traces to plans
• trace as ubiquitous semantics

• HTA as grammar over traces

• inferring structure over traces

**HTA as grammar**

• can parse sentence into letters, nouns, noun phrase, etc.

The cat sat on the mat.
parse scenario using HTA

0. in order to clean the house
1. get the vacuum cleaner out
2. get the appropriate attachment
3. clean the rooms
   3.1. clean the hall
   3.2. clean the living rooms
   3.3. clean the bedrooms
4. empty the dust bag
5. put vacuum cleaner and attachments away

task inference

• long history (lots of work early 1990s)
• limited success
  – interleaved tasks
  – generalisation
• ontology helps :-)
  – input/output links like ‘string of pearls’
  – ontology type allows single step learning
how to get links?

• user interaction:
  – drill-down from previous values

• system inference:
  – same form field linking as before

so what?
lessons

• rich interplay: tasks & data
  => (relatively) easy automated support
• time for intelligence
  • within careful interaction context
  • involve the user
• human task analysis?
  • artefact focus – for analysis and support
  • represent physical artefacts and data
  => design for the embodied user