Understanding Interaction in Digital Libraries: a case study on applying Interaction Framework

Nick Bryan-Kinns & Ann Blandford

July 2000

We are interested in understanding interaction – what it is and how it can be used in designing and evaluating interactive systems. The rest of this document sets out to explore what interaction is and how we can understand it. First the basics of our Interaction Framework (IF) are outlined. After this the framework's notions are exemplified through examples from use of various digital libraries. The examples in this working paper are based on video analysis of a protocol taken with a single experienced user, working with various digital libraries.

Interaction Framework

Our framework for understanding interaction is composed of several forms of description of the interaction which draw upon each other culminating in descriptions of general interaction properties which we believe are important to understand for design and evaluation (illustrated in Table 1). It is developed from earlier work by Blandford, Harrison and Barnard (1995). This section briefly works from the most basic description of interaction to interactional properties. In contrast the next sections consider interactional properties and reasons for trouble in interaction, and relates them to other forms of the framework.

Form of description	**Concerned with**		
Properties	Interactional properties *e.g.* familiarisation, serendipity, lucidity.		
Causes	What causes interactional trouble *e.g.* discriminability of events, mutuality of agents.	↔	Hooks to other modelling techniques or theories
Symptoms	Whether trajectory is canonical, and if not, what interactional trouble exists *e.g.* interactional detours.		
Traces	Interaction trajectories - partially ordered events to attempt to achieve objectives. Whether objectives are achieved or not.		
Fundamentals	Agents, agent grouping, communication channels between agents, events between agents, system state, objectives described in terms of system state.		

Table 1: Interaction Framework levels of description

Fundamental Interaction

Fundamentally, we consider interaction occurring between *agents* in a *system via* communication *channels* – interaction is in the form of *events* communicated between agents along these channels. An important aspect of IF is its ability to consider interactions at different levels. We might consider an event at the micro-level where it is essentially one bit of information. If we considered all interactions in terms of such events we would quickly become swamped with a plethora of events and would not be able to gain a more holistic view of the situation. Considering macro-level events allows us to move beyond such constraints whilst still being able to delve to the micro-level if necessary. For example, when entering a search term for an online library we might consider the whole piece of text entry as a single macro-event for the purposes of our analysis. However, if there were significant delays between a user pressing a key and the system displaying the keystroke (*e.g.* due to network delays on a telnet session) we might wish to consider this micro-level of analysis and how this affects the interaction.

Interaction occurs for a reason; therefore, at least one of the agents involved in the interaction has some set of explicit and implicit *objectives* they wish to attempt to satisfy. An explicit objective is usually one which an agent actively tries to meet *e.g.* finding a book relevant to their current topic of work. An implicit objective, on the other hand, is one which is not usually actively being pursued, but which can become an explicit objective due to some event *e.g.* when searching for books on model railways (their explicit objective), the agent may come across books on voltage transformers which are relevant to an implicit objective they had. They may then take up the search for information on voltage transformers as an additional objective.

Furthermore, each agent has *state* as does the whole system. System state is made up not only of agents' states, but also the communication channel configurations and the constraints on these channels. Systems move from one state to another due to the initiation of some event – a *transition* event. For each state there is a *state potential* for each agent involved in the interaction. This is the set of events which the agent could issue. This does not necessarily mean that the agent is aware of all of these possibilities. Related to this state potential is the *event potential* of that state. This is the set of events that an agent could issue to proceed towards an objective and is therefore a subset of the state potential. At this fundamental level objectives are described in terms of pre and post conditions on the state of the system.

As with events, we may also impose different groupings of agents to provide leverage on our understandings of different aspects of the interaction. For example, we may consider the different user interfaces to a digital library as different agents, or may wish to differentiate between storage and user interface agents, or indeed may just consider the whole computer system as one agent.

Interactive system

- Φ: set of all Agents
- ξ: set of all Events

System state

- A: set of agents involved (Φ)
- C: configuration of the agents A

  Set of pairs of agents which can communicate with each other

{source agent, destination agent}

- P: state potential

  For each pair of communicating agents (<α,β> C) the set of possible events that α can issue to β

  {source agent, destination agent, {possible events}}

- E: event potential

  The subset of P whose events lead α towards their objective

  {source agent, destination agent, {possible events}}

Clearly there may be conflicting sets of event potentials – each agent has its own set of objectives and so at each stage of the interaction they may wish to follow different trajectories. Resolving these conflicts, or at least considering how they can be arbitrated, is the role of the interaction designer.

Agent

- O: set of objectives

  Set of pairs of pre and post conditions <p_0, p_1>

  Note that through an interaction objectives might change in terms of their post-conditions

- S: state of agent

- B: belief that objectives can be met

  Set of pairs of objectives and probability that objective can be met <objective, probability>

Conditions

Set of constraints on the system state

Traces of Interactions

As mentioned previously, forms of description in the framework build on other forms (except for the fundamental form). As such the *trace form* builds on the fundamentals to describe *interaction trajectories* which are partially ordered sets of events constituting an episode of communication between agents attempting to meet an objective. For example, in searching for a text in a collection (an objective) there is a sequence of events for a user attempting to find the text. We are interested in trajectories that do not meet objectives efficiently, and the reasons behind this.

Trajectory

Partially ordered set of state and event pairs <s, e> - e is the transition event that leads to s_1.

Symptoms of Interactions

Building on the trace level, the *symptoms level* categorizes trajectories in terms of whether the trajectory was *canonical* (most efficient for achieving the objective), and if not, what the symptoms of interactional *trouble* were. Troubled trajectories do not meet objectives, or do not meet the objectives in the least convoluted manner. For example, mistakes may be made by the user in issuing an event which leads to the interaction involving a detour to return to a point from which the objective can be met. Reasons for non-canonical trajectories include *blind alleys* (where an agent proceeds for some time before they realise that they are not progressing towards their objective and so must *restart* the interaction), *interactional traps* (where an agent believes that their objective is unachievable even though it is achievable), and *interactional detours* (where an agent performs interaction which does not directly move them to their objective). The point at which the trajectory moves away from the most efficient trajectory is called the *point of deviation*.

Once symptoms of interactional trouble have been identified the *causes level* is used to suggest causes for the trouble. At this point IF can use hooks to other theories and modeling techniques to provide explanations for trouble occurring. For example, models of human cognition such as ICS (Barnard & May, 1999) could be used to suggest the relationship between users' goals (related to their interactional objectives) and the structure of the information conveyed by the computer system. IF also provides some suggestions for causes of interactional trouble *e.g.* the poor *discriminability* of events (how easy it is to discriminate between events) can be an reason for interactional trouble. Another possible cause suggested by IF is poor *mutuality* between the agents (how much mutual understanding there is between the agents, based on common ground (Clark and Brennan, 1991)).

Finally IF considers the interactional *properties level*. At this level IF describes properties of the states and trajectories such as serendipity - happening upon relevant information. Such properties can be related to lower levels of the framework in order to inform both analysis of current systems and design of new systems.

So, IF is concerned with agents interacting together to achieve some non-empty set of objectives. Moreover, by grouping agents we can understand interaction at different granularities. The key is that we are interested in the trajectories – how agents get from one state to another. This can be a rich and varied journey, an interesting analogy can be seen in Roquentin’s description of his life:

> …I for my part have had some real adventures. I can’t remember a single detail, but I can see the rigorous succession of circumstances. I have crossed seas, I have left cities behind me, and I have followed the course of rivers towards their source or else plunged into forests, always making for other cities. I have had women, I have fought with men; and I could never turn back, any more than a record can spin in reverse. And all that was leading me *where*? To this very movement, to this bench, in this bubble of light humming with music.
>
> (Sartre, 1963).

One interpretation of this description of his life can be used to throw light on IF as follows. In his description he saw his life as a journey – a trajectory – not knowing the details – the specific events – he was nonetheless aware of states that he had been through – seas, forests, *etc.* – and was aware of his driving objective – to make for other cities. Similarly, we can use IF with various amounts and kinds of information to help us understand where agents have been in interactions, where they are trying to get to, and why. Moreover, this description highlights the irreversibility of events – he could never turn back, and the notion of the current state to which he has arrived at – sitting on a bench in a café.

The following table summarises the properties developed to date in the Interaction Framework. Moreover, it illustrates how properties relate to either a state potential – what an agent could do from that point – or a trajectory – how an agent's interaction unfolds over time. This is an important distinction and highlights the power of IF to consider not only single events, but also sequences of events over time.

Property of	**Property**	**Description**
State potential – set of events an agent could instigate	Serendipity	Set of events that lead to an implicit objective from the current state. There is a set for each agent relating to the agents it interacts with.
	Discriminability	How distinct possible events are.
	Event potential	Set of events that lead to an explicit objective from the current state There is a set for each agent relating to the agents it interacts with.
	Event awareness	Set of events that an agent is aware they could issue from the current state. There is a set for each agent.
Trajectory	Lucidity	How sane and rational a trajectory is – whether events result in expected changes in state over time
	Familiarity	How easy it is to grasp some aspect of another agent - how easy it is to become familiar with the structure, content, or coverage of another agent.

Table 2: Interaction properties

The Observation

This section illustrates the use of the Interaction Framework to understand properties on interaction. In this section we examine the interaction involved in using digital libraries to locate articles required for a user's information needs.

In our observation an experienced academic computer user (a member of the Digital Libraries research team) volunteered to be video taped for eighty minutes as she used digital libraries in supporting her work – tasks were therefore user defined and not imposed for the purposes of the study. The user was asked to attempt to give verbal descriptions of her actions and thought processes so that these verbalisations could be used later to help interpret the interaction. From this video recording a thirty minute compilation of 'interesting' interaction was derived – by interesting we mean interaction which we can use to illuminate the description of IF, the remaining recording was mostly uneventful or repetitious. These clips were then transcribed (including speech and some description of interaction between user and computer system) and are used in the following sections as source materials for examples. Short debriefing interviews were then conducted with the user to ascertain her motives and understandings of certain aspects of the interaction. Table 3 illustrates the notation used in the transcript.

Notation	**Use**
`Blah`	Vocalisation by subject
`[> blah]`	Action: input to computer 'blah'
`[< blah]`	Action: computer output 'blah'
`[- blah]`	Current computer state 'blah'
`[blah]`	Note 'blah'
`…`	Continuation of vocalisation across action

Table 3: Transcript notation

User's Objectives

The user had one main objective: to find information from digital libraries relevant to her current work interest. This can be decomposed into four more specific objectives: to find information from digital libraries on:

- usability of digital libraries
- European cognitive modelling and its relationship to HCI
- usability of calendars and diaries
- the relationship between AI and HCI

Agents in the Observation

In order to understand the various aspects of the digital libraries we consider three main agents when a user interacts with a digital library:

- the user
- the user interface to the digital library – several windows may be open showing different versions of user interfaces to the library, these are all considered to be separate agents
- the collection of the digital library – if there are multiple collections then these are regarded as individual agents of the same class

This division allows us to separate discussion of the user interface to the library from discussions of the structure of the collection (which is in some way presented to the user *via* the user interface).

Summary of the Interaction

The user in this study had four specific objectives as discussed previously. In meeting these objectives she used four information resources: the ACM digital library, the Google search engine, the IDEAL digital library, and the New Zealand Digital Library (NZDL). These were used in a relatively orderly sequence of ACM, followed by brief use of Google, then use of IDEAL, and finally the NZDL. One interesting aspect of this sequence is that the search objectives are repeated to some extent with each information source as opposed to meeting each objective in sequence. Maybe this tells us something about the effort of switching between information sources, or possibly the support for the notion of serendipity which allows her to stumble upon entries related to other objectives.

In detail, she spent most of her time using the ACM DL. Initially this involved trouble in just logging on (finding the page and entering user information). First the user tried to locate articles on calendars and diaries by searching for such terms. This then involved plenty of 'sifting through articles' to find relevant entries. Following on from this the user encountered problems with the ACM DL notion of 'bookshelves' and 'binders' – what on earth is a bookshelf, and how do binders relate to this concept? More to the point, she questioned whether she had a bookshelf and binder. Once this situation had been dismissed as irrelevant, she started to view individual articles to judge their relevance. Any articles that seemed relevant were then printed for further review (as happened in other information sources). The whole observation is summarised in the following table which in itself illustrates the relative time spent using each information resource - activities in the following table is referred to in the following sections.

Information source	**Apparent focus**	**Activity**
ACM DL		Logging in
	Digital libraries	Search for digital librar
		No results found – due to manual stemming of search terms
	Calendars and diaries	Search for calendars and diaries
		Sifts through results
		What is a bookshelf?
		Printing out papers that seem relevant (after viewing the abstract and article on-line)
	European cognitive modelling	Search for Rasmussen
		Sifts through results – none seem relevant
		Search for Hollnagel
		No results
	AI and HCI	Search for Artificial Intelligence Human

		No search results (problem is that Hollnagel is still in author search box)
		A lot of effort trying to find articles on artificial intelligence
		Attempt to find Artificial Intelligence journals/ conferences
		Many iterations of viewing conference descriptions
		Start 2nd navigator window
		Browses list of conferences – mostly looking at AI publications
		Gets to DIS and discovers doesn't have most recent conference
		Browses through Computing Surveys
		Problems with lack of support for browsing to previous years
IDEAL		Move to IDEAL
ACM		Switch back to first window to see if an ACM article has finished downloading
Google	European cognitive modelling	Search for Hollnagel
IDEAL		Switch to second window – IDEAL
		Problems with current issue of IJHCS – seems to be empty
	Calendars and diaries	Search for diaries
		None found
NZDL		Go to NZDL page
		HCI bibliography
	AI	Search for artificial intelligence
	Cognitive modelling	Search for cognitive
		Reformulate query – same suspects found
	AI	Browse AI bibliography

Understanding Interactional Properties in the Observation

This section outlines interaction properties as we understand them with respect to excerpts from the observation.

Familiarisation

Familiarisation refers to how easy it is for an agent to develop an understanding of some aspect of another agent such as their behaviour or the structure of their state.

Familiarisation

Given two agents A and B, a preliminary definition of A's familiarisation with the aspect z of B is:

- Every event initiated by B has a post-condition that is a significant increment on A's understanding of aspect z in the pre-condition.
- There is a point early in the interaction where a basic understanding of aspect z is developed. Therefore the length of the interaction to achieve familiarisation is short.

Familiarisation can apply to a class of agents in which case the following are also applicable:

- The aspect z is reliably similar across members of the class of agents B belongs to. Therefore A's knowledge of aspect z can be usefully employed in understanding aspect z of other agents which appear to belong to the same class of agents as B.
- Agents of the same class are obviously similar in terms of aspect z.

A key issue with familiarisation is which aspects are agents concerned with. Two main aspects have come to light to date which are discussed in the following sections: the state of the agent (in digital libraries: the collection(s) of the library), and their interaction possibilities (in digital libraries: the user interface to the collections).

Familiarisation with Collections

As mentioned previously, we consider the user interface and the collection(s) of digital libraries as separate agents. In this section we discuss familiarisation with the collection(s) of a digital library – the next discusses familiarisation with the user interface. Three aspects of collection agents have been identified as objects of familiarisation:

- Content – how familiar a user agent is with the content of a specific series in the collection *e.g.* the likely content of a journal.
- Coverage – how familiar a user agent is with the content of a collection, or whole library *e.g.* how relevant the journals in a library might be to an objective. This form of familiarisation is effectively familiarisation with the class of collections.
- Structure – how familiar a user agent is with the organisation of a collection or class of collections.

Familiarisation with Collection Content

A user may be familiar with the content of a collection – what particular artefacts are in the collection. For instance, upon selecting the table of contents of the DIS conference series in the ACM digital library our user said:

```
[- ACM DL details of Symposium on Designing Interactive Systems]
Hmm, I can’t find DIS ’99, I’m sure it happened.
```

Implying that she thought that she knew the contents of the DIS conference collection, and it should have contained DIS 99. This shows a situation in which outside knowledge of what should be in a collection contradicts what is actually in the collection. In terms of design, we need to be considering how to develop collections with which users can easily become familiar with the content – to know what items are in the collection.

A more striking example comes when the user visits the IDEAL library and wants to look at the current issue of the IJHCS journal. The extract below shows part of the confusion the user suffers when she visits the current issue of IJHCS and find that there are no articles displayed – confounding her expectation from her content familiarisation that the current issue’s articles should be available (illustrated in the following screenshot). This confusion leads to plenty of aggravated interaction and eventually giving up on the issue – a question here is what that does to her understanding of the content of the collection. In the table below user initiated events are indicated by →, library issued events are indicated by ←, and attempts by the user to instigate events which are not actually part of the state potential are indicated by ❖ (these illustrate how confused the user has become about the contents of the current issue based on their content familiarity).

Screenshot of IJHCS current issue page

Transcript	**Events**
[- IDEAL library IJHCS page shown (current issue) with pop-up navigation menu (“SELECT: All Issues“)]	
Let’s just go to journal home page…	
[> selects journal home page from pop up menu]	→ IJHCS index
[< IJHCS journal homepage appears (in IDEAL lib)]	← IJHCS index
… don’t know how I got there.	
[> clicks on go (next to pop up menu saying current issue)]	→ IJHCS current issue
[< current issue page replaces journal homepage (note - no contents)]	← IJHCS current issue

Shall I go to the current issue and have a look what's there.	
[pause]	
Hmmmm.	
[> clicks journal logo]	❖
[> clicks journal logo]	❖
[> clicks go button (next to pop up menu saying list of issue)]	→IJHCS index
Where am I going?	
[< list of journal issue replaces current issue page]	←IJHCS index
OK.	
Errm.	
[> clicks on most recent issue link]	→IJHCS current issue
[< current issue page replaces journal homepage (note again – no contents)]	←IJHCS current issue
Err.	
What am I doing wrong?	
[> clicks journal logo]	❖
[> clicks journal logo]	❖
[> clicks on pop-up navigation menu]	❖
[> clicks on IDEAL logo]	❖

Familiarisation with Collection Coverage

In contrast to familiarisation with collection content, familiarisation with collection coverage refers to understanding what kinds of items should be in the collection as opposed to what specific items are in it. The following transcript extract shows how the user believes that she is familiar with the coverage of the ACM Computing Surveys journal *i.e.* what kinds of articles should be in ACM Computing Surveys, and so cannot understand why a particular article is in the Surveys.

```
[- ACM DL contents of latest computing surveys]
What's this doing in.
Surveys for goodness sake?
Doesn't look like what I'd consider to be a survey.
Why's that in the surveys thing then?
[> clicks on article link]
[< page displaying article replaces contents page]
[pause]
[> scrolls down]
[< bottom of first page comes into view]
Hmm.
Looks very.
Looks completely irrelevant anyway…
[> clicks browser back button]
[< contents of survey issue replace article page]
… but I don't know why it is where it is.
```

A more implicit indication of the user's familiarity with the ACM Digital Library is that she chose to visit it first, before other libraries. This is due to that fact that previously she found it to have the best coverage. It may also be due to her familiarity with the user interface which is discussed later.

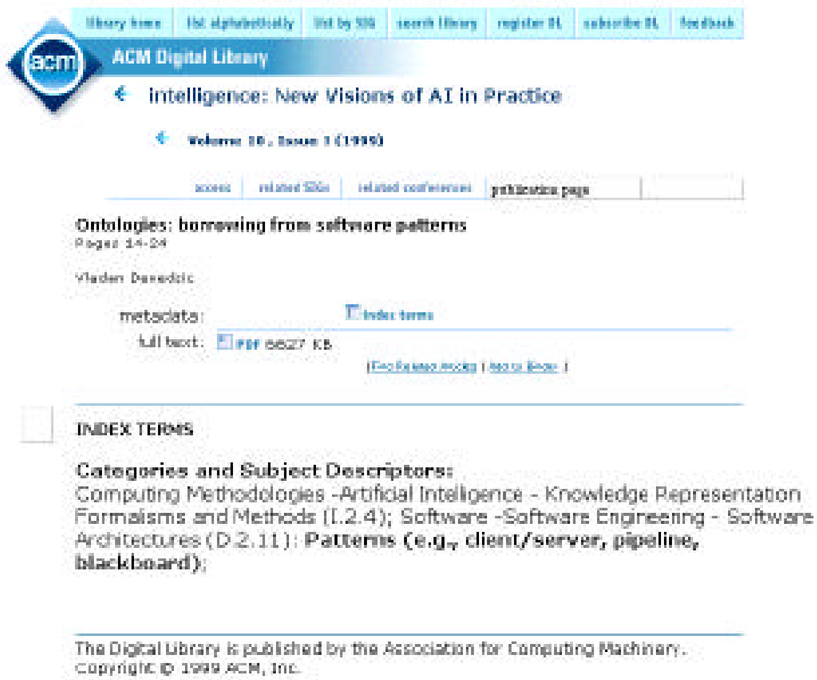
Familiarity with collection coverage is also evident in the user's use of the New Zealand Digital Library (NZDL). The first page of the NZDL shows several different collections within the library as partly illustrated in the following screenshot. The user is able to identify three collections which are relevant to her objectives as illustrated in the following transcript. This may be from the minimal information provided in the interface (titles of collections), but more probably from previous familiarisation with the coverage of these collections as some of the titles are rather nondescript.

NZDL main page

```
[- NZDL home page]
[> scrolls down page]
[< NZDL home page scrolls to show more collections]
Ah.
[pause]
OK. So there's three different collections.
There's the HCI bibliography.
```

Familiarisation with Collection Structure

In contrast to familiarity with content or coverage of a collection, familiarity with the structure allows us to build up a model of the organisation of the collection – what kinds of things we might expect to find in different parts of the collection. An interesting example of a user's familiarity with the structure of a collection comes when in our observations the user is engaged with the ACM DL. In this case she appears to be familiar with the idea that each article in the library has a separate abstract which can be viewed independently of the article itself (presumably to provide a more succinct description, and shorter download time). In the following extract we first see the user identifying an article that might be of interest (ontologies). She then issues an event to view the meta-data, but when the meta-data comes up it is particularly terse (in fact it only shows the keywords and a link to the full text as illustrated in the following screenshot). This appears to not be what the user expected to see at that point given her familiarity with the structure of the collection *i.e.* she expected an abstract.



Screenshot of meta-data for ontologies paper

```
[- contents page for volume 10 of Intelligence journal]
Ontologies is borrowing from euggh.
Shall we have just a little shufty?
[> Clicks on abstract link]
[< abstract of article displayed replacing list]
[> scrolls down]
[< bottom of abstract shown]
Doesn't seem to want to tell us anything apart from wanting to give us
full text which is going to be a…
[> clicks on full text link]
[< full text replaces abstract page]
… nightmare.
[> starts scrolling down]
Oh.
```

Familiarisation with User Interface

As with other forms of familiarisation, we are concerned not only with familiarisation with a particular user interface, but also classes of user interfaces. In the following transcript it is evident that the user's familiarity with search formulation in general led her to manually truncate her search terms from 'digital libraries' to 'digital librar' (this eventually resulted in no search results being returned). However, the ACM digital library allows user to specify that the user interface should truncate terms for them. Therefore this is an example of familiarity causing the user extra effort. As an interaction designer, we might ask ourselves how this incorrect familiarity could he highlighted for the user, or addressed by the user interface in some way.

```
[- ACM DL search page]
Terms such as digital libraries…
[>types ‘digital libraries’ into search box’]
[< search box contains ‘digital libraries’]
… seems kind of bizarre to look just for titles, so
if I put libraries I’m going to miss those with library so I’ll put…
[> changes ‘libraries’ to ‘librar’]
[< search box contains ‘digital librar’]
```

Lucidity

Interaction is lucid when it proceeds in a sane, rational, and easy to understand way with respect to agents’ objectives and constraints, and their understanding of the interactional possibilities of the system. So, for instance, whilst typing this document I expect that pressing the ‘k’ key on my keyboard will result in a letter ‘k’ being produced, I do no expect any other letter, or even combination of letters to appear – that would not be sane and rational.

Lucidity

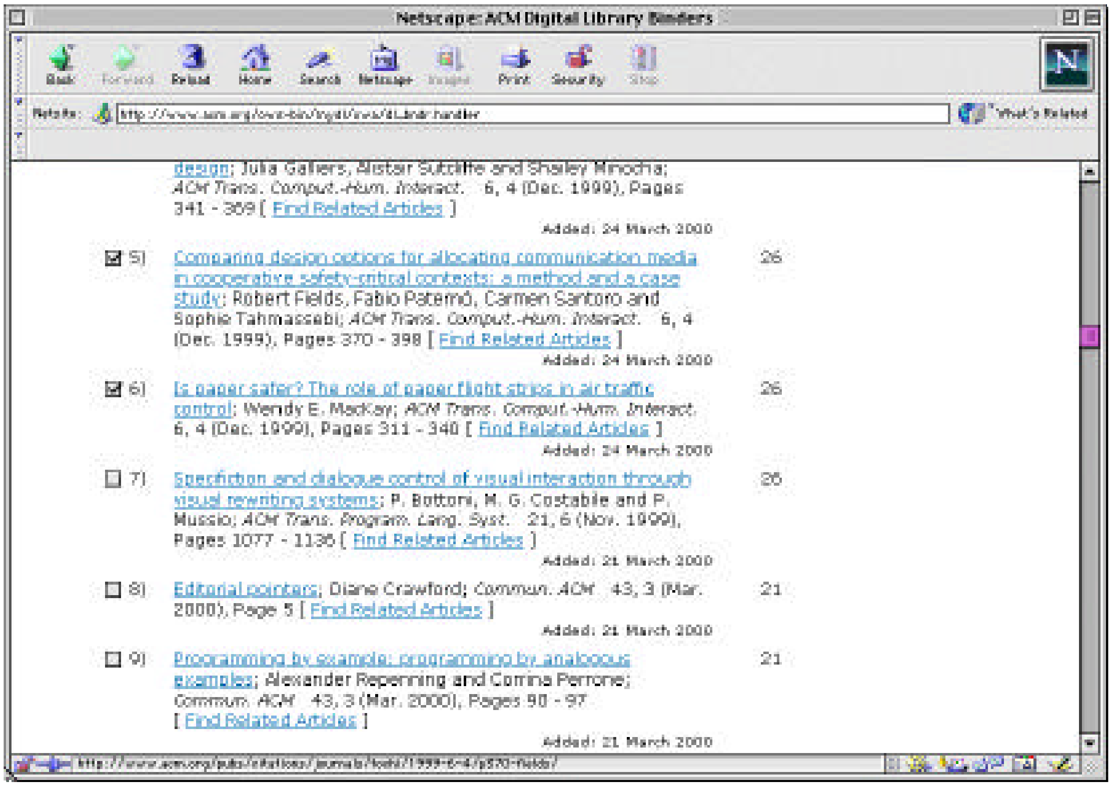
Given two agents A and B, a preliminary definition of A’s lucid interaction with B is:

- Every event initiated by B has a post-condition that A understands why B initiated the event given the pre-condition, and moreover understand what the event means. Note that pre-conditions are constraints on the system-state and so may include notions of the interaction history.

At this point we need to draw the distinction between explainable outcomes of events and expected outcomes of events. A lucid interaction may involve unexpected events, but the agent needs to be able to understand and explain them. Indeed, becoming familiar with and agent may involve some interaction which involves unexpected events. An interaction designer’s function may be to develop systems which can support familiarisation in a lucid manner *i.e.* unexpected events may occur, but they are explainable and, moreover, understandable to the user.

The following transcript extract illustrates two periods of non-lucid interaction in the ACM digital library. The first, smaller, period occurs just after a set of search results have been presented on a web page. Each search result has a tick box to the left of it (illustrated in the following screenshot); ticked articles are kept for the next stage of interaction. Unfortunately the user does not have any idea what these tick boxes do as illustrated in the following extract:

```
[- ACM DL search results page]
I don’t know what happens when I tick them.
```



Screenshot of ticked boxes with unknown purpose

Luckily she ticks the relevant articles anyway. Once all the relevant articles have been ticked a longer non-lucid interaction occurs as the following illustrates:

```
[- ACM Dl search results page with several articles ticked]
I’ve clicked these things now and I haven’t got a clue what I’m meant to
do with them. Hmm. I don’t know what a binder is…
[> clicks and holds on binder link]
[< browser pops up menu of possible actions – open page etc.]
… help…
[> releases mouse]
[< page is replaced with page containing list of articles selected]
… please choose a binder from your bookshelf.
I don’t have a binder, and I don’t have a bookshelf as far as I’m aware.
Errm OK I’d better just have a quick shuft through…
[> clicks on first article in list]
[< page is replaced with abstract of first article]
… I guess.
```

In this extract the user doesn’t know what to do with the articles now, and again luckily issues the correct event. Even then the system does not appear to be ‘speaking the user’s language’ as she remarks that she doesn’t have a binder or bookshelf which the interface has been referring to – the user interface has issued events which the user does not know the rationale behind, nor understand their meaning. All in all this leads to a disconcerting and possibly off putting interaction.

The non-lucid interaction surrounding the binders and bookshelf is eventually resolved by the user herself. The following extract can be seen both as an attempt to resolve the non-lucid interaction, and also to familiarise herself with the user interface:

```
[- ACM DL list of selected articles]
Where would I find, I’m I’m intrigued about this bookshelf, where would I
find my bookshelf?
[> scrolls down]
[< list of selected articles scrolls down – my bookshelf details appear]
Oh, my bookshelf.
[> scrolls down]
[< page scrolls to very bottom]
[> scrolls up]
[< page scrolls up to show some selected articles and some of my
bookshelf]
Please select a binder from your bookshelf.
Well I can’t be bothered right now so lets go back.
[> clicks browser back button]
[< list of previous search results shown]
Don’t think I need that. OK.
```

Errors are a common cause of non-lucid interactions; especially if other agents are not aware of the reason for or solution to the error. In these cases agents do not understand why events are issued, nor necessarily what they mean. For example, in the following extract the user agent has decided to return to the search page and reformulate her query (in effect a reset after a failed search). However, on the second press of the back button the interface agent starts to behave in a non-lucid manner – displaying an error and then returning to the search results list when reload is clicked – not what the user agent expected.

[- ACM DL search results page – note no matches] No matches. Ha ha ha. [> clicks browser back button] [< search page replaces search results page] Right OK lets. [> clicks browser back button]	User reset
[< error page saying document not found in cache replaces search page] Documents results expired in cache. Lets try a bit of reloading.	User interface error
[> clicks browser reload button] OK, I don’t know what that question meant. What ever it was will	User attempt to recover from error
[< page of search results is displayed replacing error page] OK, I didn’t want search results. It wasn’t. I just wanted to go home again.	Non-lucid interaction from user interface agent

Finally, the common situation of being ‘lost’ in an interface also relates to our notion of lucidity. In these situations user agents are confused about their state with relation to other agents’ states *e.g.* their position within a hypertext. Issuing events to user interface agents often results in response events which are confusing to the user agent – not what they expected. In the following extract we see the user first stating that she is lost, and then scrolling up and down the main ACM DL page to try and resolve her 'lostness' or lack of lucid interaction with the user interface agent. Eventually the user agent appears to have resolved her non-lucid interaction – so she understands the events the user interface agent is issuing and why – and selects the Computing Surveys to view.

```
[- ACM DL list of publications]
OK, so I don't know where we are.
[> scrolls up]
[< top of page comes into view]
Errm. ACM magazines and journals table of contents.
[> scrolls down]
[< more of list comes into view]
Standard view.
[> scrolls down]
[< more of list comes into view]
Journals.
OK, so those were magazines. I just spent ages printing out a magazine.
Errm, theoretical computing science.
OK, that's something else.
Errm.
Computing Surveys.
[> clicks on computing surveys link]
[< computing surveys table of contents replaces list of publications]
```

Interactional properties do not just occur in isolation. In the following transcript we see that the user's interaction becomes non-lucid due to her familiarisation with the structure of the collection. In this case the user believes that each abstract should lead to an article, but in this case no article is linked to. We can see an indication of the lack of lucidity in the number of spurious events the user issues in moving about the abstract page – scrolling up and down, and clicking on the abstract link. Eventually she assumes that the article is not relevant anyway – this may be an example of an *interactional trap*. A trap is a situation in which the agent believes that the objective is not achievable even though it is – maybe there is some reason why the article is not linked to, or maybe there is some other way of reaching it, but the user is not aware of either of these and so assumes that it does not exist, and moreover assumes that it is not relevant to her objective.

```
[- ACM DL list of selected articles]
[> clicks on link to abstract of an article]
[< abstract of article displayed (page replaces list of selected
articles)]
… which one's relevant.
[> scrolls down]
[< bottom of abstract shown]
Oh. Looks like…
[> clicks on abstract link]
[< page scrolls to bottom]
… design, human factors, and theory…
[> clicks browser back button]
[< abstract page returns to top]
… it doesn't tell us very much about it at all and I can only assume it
hasn't been errm digitised for the library which seems strange really.
```

```
[> clicks browser back button]
[< list of selected articles replaces abstract page]
OK.
Well, I'm prepared to assume that's not relevant.
```

Causes of Interactional Trouble in the Observation

At the *causes* level IF posits possible reasons for interactional trouble (in addition to drawing on external theories). This is a crucial level for IF as it highlights problems in the interaction, provides some possible reasons, and leads to discussion of interactional properties. This section discusses two causes of interactional trouble that have been developed for IF.

Mutuality

In interaction, agents communicate events along communication channels. These events attempt to communicate some information about the state of the agent. But issuing an event does not necessarily mean that the recipient will receive the event let alone understand and act on it appropriately. We draw on Clark's notion of *common ground* (Clark and Brennan, 1991) and in particular the interpretation of common ground developed by Healey and Bryan-Kinns (2000) to develop the notion of event mutuality, or mutual understanding of events.

Mutuality

For an agent A issuing an event e intended for agent B we define five states of event mutuality which agent B could be in with respect to the event e:

- **State 0**: B is unaware that event e exists.
- **State 1**: B is aware that e exists.
- **State 2**: B recognises e as being of a particular type.
- **State 3**: B understands the content of e.
- **State 4**: B understands what actions are associated with e.

Before considering these states of mutuality it is worth remembering that in some cases agents just do not issue the appropriate events. For example, in the following transcript we see that the user has issued an event to start the download of a document (pdf). The Netscape agent issues an event in response which basically clears the Netscape page, but does not issue events indicating how long the download will take. In fact it takes about 30 secs to download the document – clearly some indication of this would have been useful for the user.

```
[- ACM DL abstract of an article previously selected]
… about, so where's the pdf?
[> clicks on pdf link]
[< page clears, adobe acrobat starts up]
[- blank browser page]
[20 seconds]
Oh joy. I can't remember how big it was now so don't know how I'll know
when we're getting there.
[10 seconds]
```

```
[- acrobat viewer within netscape showing article with thumbnails to the
left hand side]
```

Looking carefully at the interaction we can see that the digital library agent actually indicates the size of the file to be downloaded next to the download link. This means that in the example above the user has missed this information *i.e.* is at state 0 of mutuality with respect to the size of the file. The following extract illustrates a situation which contrasts the above in that the user has noticed the size of the file, and moreover, the digital library agent is providing events which detail the progress of the download. In this case the user is much happier with the interaction than before.

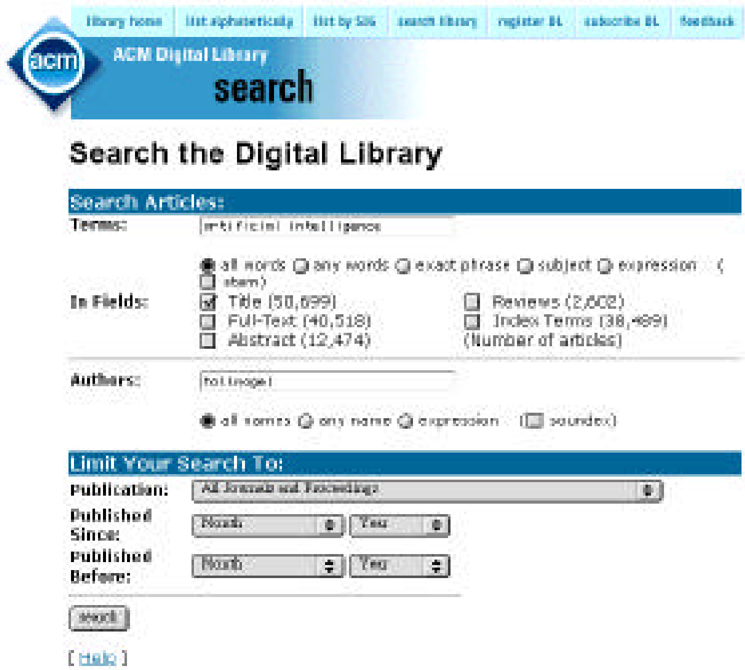
```
[- ACM DL abstract of paper]
Hmm, so that's another 1000k.
[> clicks on pdf link]
[< acrobat page appears with article only (no thumbnails)]
Oh.
But at least this one came up quickly, that's much nicer.
Presumably because it didn't have to hang around for the errm cute errm
little finger prints things, thumbnails, or whatever they're called.
[< percentage done appears and is updated]
Percentage has come up, that's much nicer.
I much prefer that.
[> scrolls down]
[< scrolls to end of article]
OK.
That's a vast improvement.
```

We see a similar lack of event issuing in other parts of the library. In the following extract it is clear that the user agent is interested in citation information for an article. The collection agent knows this information – some of it would have been presented in the index list. However, when the article is displayed the only citation information is implicit as part of the article which the user has to search for rather than being explicit presented by the user interface agent. This means that the user has to actively issue scroll events to move the page down and bring into view the citation information. From interviews it became clear that the user supposed that citation would be at the bottom of the first page from prior familiarisation with the structure of articles.

```
[- acrobat viewer showing article]
Can't now remember when it was, which conference, or which year.
[> scrolls article down]
[< article scrolls down to bottom of first page]
1994. Ah, so its actually very old.
```

State 0

If agents do issue an event, the worst case is that other agents do not even notice that the other agent has issued an event for them. Clearly this may lead to interaction trajectories which have low lucidity. The observation gave a striking example of such a situation as exemplified in the following transcript. The user is searching for articles related to artificial intelligence and HCI (the user's fourth objective discussed previously). She therefore enters 'artificial intelligence human' into the subject search box. However, she had previously been searching for authors with surname 'Hollnagel' – Hollnagel is still in the author search box. Therefore the user interface believes that the user wishes to search for articles containing the words 'artificial intelligence human' and written by 'Hollnagel' as illustrated in the follow screenshot. Needless to say no articles are found.



Screenshot illustrating searching for both terms and authors

```
[- ACM DL search page]
OK. Then so the other thing I wanted was…
[> types artificial intelligence into subject search box, correcting a
typing error part-way through typing 'intelligence']
[< artificial intelligence appears in subject box – note that Hollnagel is
still in author box]
… artificial intelligence, oops, intelligence, and human interaction, well
it could be human computer interaction or human factors…
[> adds human to end of search terms]
[< subject search is now 'artificial intelligence human']
… so lets.
Which one worked last time?
I can't even remember which one I did last time.
Was it full text or abstract?
Lets go for full text.
[> clicks on full text tick box]
[< full text selected]
[> clicks search]
Errm, I don't know what reviews or index terms would be.
[< search results page replaces search formulation page (note – no
matches)]
No matches. Great.
```

The user is nonplussed by this lack of search results – she is in mutuality state 0 with respect to the author search field *i.e.* she has no idea that it is being used in the search. This supposition on our part is backed up by the user's attempt to reformulate the search. As illustrated in the follow excerpt, she reformulates the subject search box, but does not change the author search box:

```
[- ACM DL search page]
So even on full text.
Of course, its possible, no.
What happens if I turn off the human and try again?
[> removes human from subject search terms]
[< subject search is now ‘artificial intelligence’ (note – author search
is still ‘Hollnagel’)]
I would expect to get a fair amount.
[> clicks search]
[< search results replace search formulation page – note no matches]
No matches. Ha ha ha.
```

We can characterise this as a non-lucid interaction – the agents that the user interacted with are just not giving the responses she expected, but this is because she is unaware of some of the information offered by the agents. In terms of interaction design, we need to try to ensure that agents within the system aim for as high a state of mutuality as possible in order to reduce such problems. Moreover, this example forms an interactional trap – the user now believes to some extent that the library does not contain articles indexed by the terms ‘artificial intelligence’, even though it does.

State 1

Moving on to mutuality state 1 we encounter interactions where an agent is aware of another’s event, but does not know what kind of event it is, nor what it means. In our study the user encounters such a problem with the ACM DL search page. Whilst selecting various search options (illustrated in the following extract) the user wonders what the numbers next to these options indicate. In this case she is aware of the information but does not know its meaning. In fact the numbers indicate how many articles are covered by particular criteria such as ‘full-text search’ as explained textually just below the search options as illustrated in the screenshot below:

Screenshot extract showing search fields with numbers

```
[- ACM DL search page]
Errm, actually probably looking through full text is probably a waste of
time…
[> clicks on full text option]
[< full text option deselected]
… but looking through the abstracts might help.
[> clicks on abstract option]
[< abstract option selected]
Errm, I don’t know what those numbers after the things are.
Anyway, let’s try searching and…
[> clicks on search]
… see what we come up with.
```

The following transcript extract illustrates a situation in which the user has started to download a document and then switched to work in a different window whilst the document downloads. The user is aware that the download window (agent) issues events to indicate the current download state (*i.e.* percentage done so far). However, as she is working in a different window which obscures the display of these events she only reaches level 1 – she is aware of their existence. Therefore she has to switch to the background window to check whether the paper has finished downloading. In the following extract the user switches to the background window to check whether the article has downloaded. Note that she is still unsure whether it has fully downloaded and so has to scroll to the very end to be sure. In the end she has ascertained that the document has fully downloaded and that she feels that she should print it out (state 4).

```
[- IDEAL page of article]
What do I have running in the background?
[> clicks on 2nd window]
[< 2nd window comes to front to obscure 1st]
[< ACM DL article from DIS – guiding people]
That paper and I haven’t printed it out yet.
[> scrolls down]
[< second page comes into view]
Let’s just check it’s got to the end now.
[> drags scroll bar to end]
[< end of paper comes into view]
Yep.
So let’s just print that one.
```

State 2

One of the main problems found with the digital libraries visited was the difficulty of ascertaining whether articles are relevant from a short description or abstract, *i.e.* without viewing the document itself. In terms of mutuality, this indicates that the user reaches level 2 with respect to the events pertaining to the article – she knows that it exists, and what kind of document it is – but she does not fully understand the content. In our studies we found that this leads to many blind alleys as the user had to proceed for some time in the trajectory then *reset* and start again.

In IF we define a blind alley as an interaction in which an agent proceeds for some time before realising that they are not progressing towards their objective. They must then return to a previous point in the interaction by resetting (backtracking). This notion of resetting to a previous state is unrealistic. For a start, a user agent would never be able to return to a previous state as we cannot (presently at least) return our brains to a previous state. Even with computer based agents resetting may not produce an identical state to before *e.g.* navigating using a web browser and then resetting to a previous page using the back button tends to bring up the previous page, but with links of pages that we have since visited represented in different ways to before *e.g.* using different colours. A more useful definition of resetting is:

For an agent A, resetting to system state z actually involves progressing to system state y which is sufficiently similar to system state z for the resetting agent.

The ACM Digital Library is typically structured hierarchically as below. In the figure ⌣⌣ indicates a one to many hierarchical relationship between the item above and the ones below, whereas | indicates a one to one relationship. There is a main index page from which different sections are available *e.g.* conferences, publications. Each of these sections then allows access to particular series *e.g.* a particular conference such as ICE. From particular series pages there is then usually access to individual publications such as the proceedings for a specific year or a particular journal issue. And then finally access to particular articles of that publication (possibly *via* an intermediary level of the article's abstract).

Main index page	
⌢	
List of conferences	
⌢	
Conference series	*e.g.* ACM ICE
⌢	
Specific conferences (usually identified by year)	*e.g.* ACM ICE '99
⌢	
Article abstracts	*e.g.* abstract of an article in ACM ICE '99
|	
Articles	*e.g.* an article in ACM ICE '99

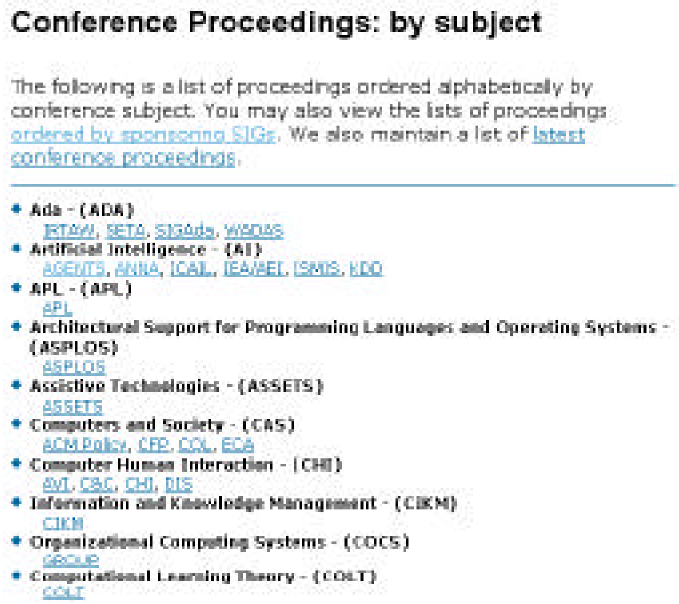
Importantly, such hierarchical structures are found repeatedly throughout digital libraries, not just in the structuring of browsing access to the library, but also in structuring the search results as below. In this case a list of search results is presented; each entry in the list gives access to the abstract and the article – the intention of the access to the abstract appears to be to provide a summarised form of the article.

Search results	*e.g.* list of articles matching 'digital library'
⌢	
Article abstracts	*e.g.* abstract of an article in ACM DL '99
|	
Articles	*e.g.* an article in ACM DL '99

Not only do such structural patterns repeat themselves throughout the library, but also interaction trajectories constrained by these patterns are evident. As we mentioned at the start of this section, users tend to be involved in a lot of blind alley interactions due to the description of entries being insufficient.

The worst cases seem to involve the user agent traversing two levels of the library structure – from the list of conferences, to a conference series, and then to a particular conference – as illustrated in the following table, screenshots, and transcript. After ascertaining that the conference is not relevant they then have to reset and look at the next conference.

List of conferences	
|	
Conference series	ICAA
|	
Specific conference	Agents 99

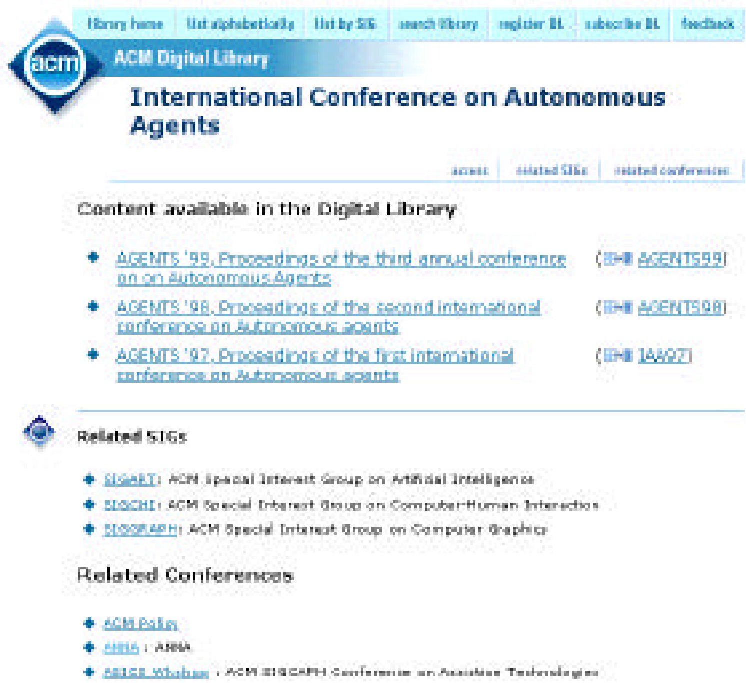


Conference Proceedings: by subject

The following is a list of proceedings ordered alphabetically by conference subject. You may also view the lists of proceedings ordered by sponsoring SIGs. We also maintain a list of latest conference proceedings.

- Ada - (ADA)
  IRTAW, SETA, SIGAda, WADAS
- Artificial Intelligence - (AI)
  AGENTS, ANNA, ICAIL, IEA/AEI, ISMIS, KDD
- APL - (APL)
  APL
- Architectural Support for Programming Languages and Operating Systems - (ASPLOS)
  ASPLOS
- Assistive Technologies - (ASSETS)
  ASSETS
- Computers and Society - (CAS)
  ACM Policy, CFP, CQL, ECA
- Computer Human Interaction - (CHI)
  AVI, C&C, CHI, DIS
- Information and Knowledge Management - (CIKM)
  CIKM
- Organizational Computing Systems - (COCS)
  GROUP
- Computational Learning Theory - (COLT)
  COLT

Screenshot of part of conference list



library home | list alphabetically | list by SIG | search library | register DL | subscribe DL | feedback

ACM Digital Library

International Conference on Autonomous Agents

access | related SIGs | related conferences

Content available in the Digital Library

- AGENTS '99, Proceedings of the third annual conference on on Autonomous Agents (AGENTS99)
- AGENTS '98, Proceedings of the second international conference on Autonomous agents (AGENTS98)
- AGENTS '97, Proceedings of the first international conference on Autonomous agents (IAA97)

Related SIGs

- SIGART: ACM Special Interest Group on Artificial Intelligence
- SIGCHI: ACM Special Interest Group on Computer-Human Interaction
- SIGGRAPH: ACM Special Interest Group on Computer Graphics

Related Conferences

- ACM Policy
- ANNA : ANNA
- ASSETS Workshop : ACM SIGCAPH Conference on Assistive Technologies

Screenshot of part of agents conference series

Screenshot of part of Agents '99 conference

```
[- ACM DL journals and conference proceedings page]
I don't know what any of these things are.
[> clicks International Conference on Autonomous Agents publication link]
[< International Conference on Autonomous Agents page replaces
publications list]
Well, I know what an agent is.
Agents 99.
Related SIGs.
SIGCHI fine.
[> clicks on most recent conference link]
[< Table of Contents page starts to appear (some delay)]
Hermatic, believable emotional agents [inaudible]
Eugh.
Pets with evolving emotional intelligence.
Gosh.
It would be so much easier to just sit there with a paper copy and flick
through and actually see.
I don't think any of it's particularly useful.
[> clicks browser back button]
[< details of conference page replaces content of conference page]
Errm.
[> clicks browser back button]
```

A more frequent pattern is the need to traverse down one level of the structure to properly decide whether the entry is relevant. This may seem a trivial amount, but the effort involved in these extra events and reset mount up, especially when the user is presented with long lists of possibly relevant information *i.e.* lists with poor discriminability. In the following table we see repeated trajectories which involve the user agent in viewing a conference description and then resetting. This involves not only effort on the part of the user agent, but also on the part of the library agent which may have network constraints *etc.* to deal with. Note that in the whole of this trajectory, no relevant conferences were found by this method – each time the user is simply trying to ascertain whether the conference is relevant, but has to navigate to the conference description to find out. The transcript of the observation is to the left of the table, to the right the transcript is annotated with the interaction - → indicates a user agent event, ← indicates a library agent event, and indicates a user agent instigated reset.

Transcript	**Interaction**
[- ACM DL list of conferences page]	
Errm. I don’t know what…	
[> clicks on ANNA publication link]	→ANNA
[< details of ANNA publication page replaces list of publication]	←ANNA
… ANNA stands for.	
Analysis of neural networks applications.	
ANNA 91.	
[> clicks back button]	
[< list of publications replaces details of ANNA]	←conference list
I don’t think that’s going to be any use at all.	
[> clicks on IJAL]	→IJAL
[< IJAL details replaces list of publication]	←IJAL
AI and law.	
Errm.	
[> clicks back button]	
[< list of publications replaces details of IJAL]	←conference list
I don’t think that’s going to be any use either.	
IEA.	
[> clicks on IEA link]	→IFA
[< IEA details replace list of publication]	←IFA
Industrial and Engineering Applications of AI and Expert systems.	
Seems to be another…	
[> clicks back button]	
[< list of publication replaces details of IEA]	←conference list
… dead conference.	
[> clicks on ICMS link]	→ICMS
[< ICMS details replaces list of publications]	←ICMS
ICMS.	
Methodologies for intelligence systems.	
ICMS ’86.	
Wow.	
[> clicks browser back button]	
[< list of publications replaces details of ICMS]	←conference list
This is a reaaaaly…	
[> clicks on Knowledge discovery conference link]	→KDD
[< Conference of Knowledge Discovery in Data details replaces list of publications]	←KDD
… unhelpful way of listing things.	
[> clicks browser back button]	
[< list of publication replaces CKDD details page]	←conference list
OK.	
So that was a conference on knowledge discovery.	
HCI.	
What’s AVI?	

```
I don't know.
[> clicks on AVI link]                                          →AVI
[< details of AVI conference replaces list of publications]     ←AVI
AVI '94.
Advanced Visual Interfaces.
Hmm. That's not relevant.
```

We can see similar trajectory patterns when trying to decide whether articles are relevant. Again the user agent traverses one or two levels of the collection structure in order to determine relevance. In the case of articles this is from the list of articles (either a search results list, or a list of articles in some publication) to the abstract, and maybe even the article itself before its relevance is ascertained. The design implications for these situations seem to be that designers should try to raise the mutuality state as soon as possible, so that the user agent does not need to be involved in extended trajectories which may be fruitless. Of course, users need to be able to ascertain whether articles, conferences, *etc.* are relevant, but in the examples given here they are becoming involved in inefficient interactions to discover whether they are relevant.

In fact the above examples show not only the problems caused by low levels of mutuality, but also non-lucid interactions. The following extract highlights a particular instance of repeated lack of lucidity for shorter trajectories than those discussed in previous sections. In this case (repeated in several other parts of the interaction) the user has navigated a couple of levels down the hierarchy of the digital library's structure – from the list of conferences, to a particular conference (ICE in this case) and then to a particular conference of that series (illustrated in the following figure) – and then wants to return to the list of conferences to browse to other conferences. As we have seen, this pattern is repeated several times within the conferences collection as well as in similar ways in other parts of the digital library. The aspect we focus on here is that when the user *resets* the interaction with the digital library agent (*i.e.* goes back to the list of conferences page) by pressing the back button twice on the browser, the list of conferences page is scrolled to a different position than she left it. For the user this causes annoyance (see transcript) and extra effort as the conference list has to be scrolled down to locate the previous location (assuming the user can remember where that was). In terms of our definition of resetting it means that the user agent has to expend more effort than is desirable in resetting the system state to a situation she regards as suitable.

```
[- ACM DL ICE contents of conference page]
Cost benefit trade-off says this isn't the place to be.
[> clicks browser back button]
[< ICE details page replaces contents of conference page]
OK.
[> clicks browser back button]
[< list of publications replaces ICE details]
So, there's nothing new on digital libraries that I don't already know
about.
Aggh, it keeps on coming back to the beginning
[> scrolls down]
[< two more pages of publication list scroll into view]
```

As illustrated in the table below it is clear that to reset to the user's desired previous state (the conference list before she viewed the contents of a particular conference) requires two mouse clicks and then some more (scrolling) events to locate the appropriate part of the page. Some more efficient form of resetting would be useful in this case, and the many other similar cases seen in the observation. In terms of redesign this affects not only the design of the digital library (how the user gets back to the list of conferences page), but also the browser itself (how the back button works in terms of scrolling the page).

In terms of the ACM DL the reset problem is due to the browser and the manner in which the pages of the digital library are served. In this case the pages are served dynamically which means that the browser does not keep track of the page when other pages have been loaded resulting in the fact that when the page is reloaded it is reloaded from the server and the previous scroll position is not used as it may no longer be appropriate. Why the browser does not keep a cached copy of the page and then reload it when the page is moved back to is anybody's guess, but it does provide us with food for thought when implementing the interaction.

Event	User Agent Event	Netscape Agent Event	Perceived purpose
e1	Click on conferences link		Navigate to ICE conference
e2		Displays list of conferences page with ANNA towards top	
e3	Scroll down twice		
e4		View of conferences page scrolled to show ICE conference at top	
e5	Click on ICE conference link		
e6		Display details of ICE conference with links to particular conferences	
e7	Click on ICE 99 link		Navigate to ICE 99
e8		Details of ICE 99 displayed	
e9	Click on back button		Reset
e10		Display details of ICE conference with links to particular conferences	
e11	Click on back button		
e12		Displays list of conferences page with ANNA towards top	
e13	Scroll down twice		
e14		View of conferences page scrolled to show ICE conference at top	

So, the interaction trajectory in this case is e1, e2, e3, e4, e5, e6, e7, e8, e9, e10, e11, e12, e13, e14

Meeting the sub-objective of finding the ICE conference is e1, e2, e3, e4, e5, e6

Then meeting the sub-objective of investigating the ICE conference through a particular instance is e7, e8

Reset (*i.e.* returning to the list of conferences) consists of e9, e10, e11, e12, e13, e14

These trajectories are illustrated in the following table. The first column details the page of the ACM DL and the use of links by the user (indicated by a |). Next to these are the events used to navigate to a particular instance of the ACM ICE ’99 (events to move to a new page indicated by ↓). Finally the events needed to reset the interaction are shown (events to move back indicated by ↑) – note the number needed.

ACM DL page	**Navigate to ICE 99**	**Reset**
List of conferences	e1, e2, e3, e4	e12, e13, e14
|	e5↓	e11↑
ACM ICE	e6	e10
|	e7↓	e9↑
ACM ICE ‘99	e8	

A similar pattern of trajectories occurs when the user tries to navigate between siblings of page – the user must navigate *via* the parent page and so moves up and down levels in a pattern similar to those discussed before, but for a different reason. Our redesign might include consideration of how to move between articles directly – rather than having to go *via* the list of articles. For example, to move between articles in a journal, or to move between different issues of a journal, the user must navigate *via* the super ordinate list. This makes the trajectory longer than may be necessary and causes the user aggravation. In the following transcript we see the frustration of the user having to issue these extra events, and the extra events themselves, in navigating *via* the list. In this case she is moving from one issue of Computing Surveys to the next and so have to go *via* the Computing Surveys main index page.

```
[- ACM DL content of an issue of computing surveys]
[> scrolls down]
[< bottom of contents comes into view]
Now I can’t just even browse through the previous year…
[> clicks browser back button]
[< list of computing surveys page replaces content of issue page]
[> click on link to a previous computing surveys]
[< page of contents of a previous computing survey replaces list of
surveys]
… without going back there.
```

State 3

So, we may know what kind of thing an event pertains to, but then we need to understand its content. This is what we mean by mutuality state 3 – that the agent understands the content of the event. The following extract illustrates a move from state 2, to state 3, and finally state 4 (discussed later) of mutuality. In this example the user has been looking through a list of search results. She comes across an article in this list entitled ‘2000 AD’ which she thinks might be relevant. At this stage she knows little about the event and we can regard her as being at level 2 –

she knows that it exists and that it is an article. By clicking on the abstract link she gets a better picture of what the article is about (to do with dates) and so she is at level 3 – she believes that she understands the content of the event. Finally she decides what actions are appropriately associated with it (to reject it as not relevant) – state 4 - and so resets to the search list.

Transcript	State
[- ACM DL search results page]	
2000 AD I wonder what that might be…	2
[> clicks on 2000 AD abstract link]	
[< abstract of article displayed replacing search results page]	
… its given me an abstract errm…	
[> scrolls down]	
[< bottom of abstract displayed]	
… oh it’s to do with dates,	3
no that’s not relevant	4
[> clicks browser back button]	

State 4

Of course, interaction would be very difficult if state 4 was never reached, or was difficult to reach as we have discussed previously. The observation showed many times at which state 4 was more easily reached as illustrated in the following extracts.

First off, the following extract comes after the user has entered ‘calendar diaries’ as the search term. It shows that in the list of results returned by this search contains an entry which the user is aware of. That is she has state 4 about the article and so knows that she can skip it as she has previously obtained and read it.

```
[- ACM DL search page]
[> click search]
[< search results page shown]
Ah ok.
Developing calendar visualisers for the information visualiser.
Errm.
The diary study, well, I happen to know what that work is.
```

A more interesting example is highlighted in the following transcript. In this example the user notices an article in the list of search results which is relevant to a colleague (P). She then proceeds to view the pdf and finally save it to print out for P (not illustrated in this extract). Again this shows that she has reached state 4 about the entry – she understands the content from the abstract and knows what to do with it in the situation – save it and send to her colleague.

```
[- ACM DL abstract of article page]
This is probably relevant…
[> scrolls down]
[< bottom of abstract comes into view]
[> scrolls up]
[< top of abstract page comes into view]
… to P.
[> clicks on pdf link]
```

Interestingly this example also illustrates the property of serendipity which is a function of the state potential. Here the digital library agent communicates the state potential (*i.e.* possible events) in such a way that she can identify the events pertaining to this article (clicking on the abstract, article, and pdf links) as leading to an implicit objective which may be to note interesting material for colleagues. An important aspect of designing digital library agents is that they should be able to convey information to meet such implicit objectives without seeming to present completely irrelevant information.

Discriminability

At the simplest level, discriminability is the ability for an agent A to usefully (in terms of their objective) distinguish between potential interactions with agent B. As we are concerned with agents that attempt to meet objectives, this can be more rigorously defined as:

Discriminability

- The ease with which an agent A can identify the event potential of interaction with agent B from the state potential of agent B.
- And additionally, whether A can distinguish between the salient events in the event potential.

A simple observation that falls out of this definition is that the larger the state, or event potential are, the harder it will be for an agent to discriminate between the events and form a useful plan of which event to follow for the next step of their trajectory. In order to understand what makes events more or less salient we could employ other theories. For example, using theories of perception can help us to understand how user agents perceive possible events, and why some presentations of possible events are more discriminable than others.

As discussed previously, we consider a digital library to be at least two agents: the user interface and the collection(s). Therefore in this example we can consider the discriminability of user interface options as well as contents of a collection. Discriminability is closely related to mutuality – the possible events in the state potential set need to be communicated in such a way that other agents can discriminate between them *i.e.* they need to have at least mutuality state 2 whereby they can be distinguished in terms of type. Therefore discriminability is about the mutuality for a set of possible events – are we able to understand enough about the possible events to work out whether they are relevant or not.

In the long example given above in which the user is constantly moving between the lists of articles, for example, and the abstracts, we observe poor discriminability between the potential events. The user is made to do more work because of this poor discriminability – moving between the list and the abstracts. The most obvious form of the lack of discriminability is a situation where there are two possible events which appear identical **-** as opposed to the previous example in which events do appear different, but the user agent can not work out their difference just by the event itself. In the following extract the user is working through a list of search results. As she scrolls down she notices that one of the entries is listed twice. This is poor discriminability for those two entries – maybe they are not the same, maybe they are, the key is that the user is unable to tell the difference from the information communicated to them.

```
[- NZDL HCI bib search results page]
OK. Know about that one.
Cognitive research.
```

```
[> scrolls down]
[< more search results come in to view]
Eugh.
[pause]
It errm.
[> scrolls down]
[< more search results come in to view]
It appears to be here twice.
```

Good discriminability means that agents can determine the differences between potential events, and to criteria they have in order to meet their objectives. In the following extract the user agent performs a search for the author Rasmussen. The library returns a list of authors, none of whom are the particular Rasmussen she was looking for (Jens Rasmussen). She ascertains this without having to view the abstracts, or enter into any other interaction other than scrolling. Therefore we regard this set of events as providing good discriminability when the objective is to find articles by a particular author – maybe the same events would provide poor discriminability for other objectives such as those discussed previously.

```
[- ACM DL search formulation page]
Let’s, let’s go the other way and…
[> enters an author in the author search box]
… just look for one or two authors.
Let’s search for something by Rasmussen or Hollnagel I think.
[> clicks search button]
[< list of search results shown in page replacing search page]
Right, err, um, who the Rasmussen.
Lars, oh there’s lots of Rasmussens.
Errm, I don’t think I want VLSI or randomised load balancing.
[> scrolls down a bit]
[< next search result comes in to view]
A.P. B.A. ha ha ha
Alright.
[> scrolls down]
[< page scrolls down to end]
There are lots of Rasmussens in the world and none of them is the one I
was looking for. Errm…
[> clicks browser back button]
[< search page replaces list of results page]
… do I want, I wanted Jens Rasmussen.
[> starts editing author search box]
```

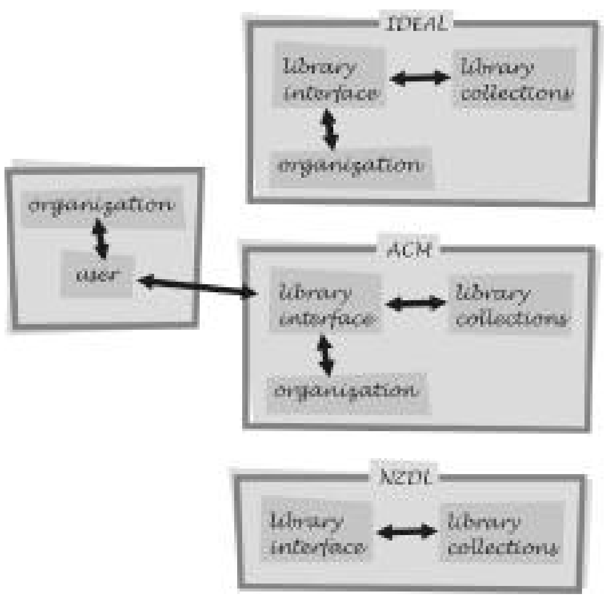
Furthermore, we can use the idea of discriminability to give us some idea about user agent’s discrimination abilities – that is, how large a state potential a user feels happy with, or that they feel they could discriminate between the possible events of. For instance, on receiving 16 results the user felt that it was not too bad, whereas being presented with more than 50 seemed to be a problem – we might postulate that this was because the user felt swamped by the state potential and concerned about whether she would be able to successfully identify the event potential within it.

System Configuration

Problems with interaction not only come from trajectories and events, but also from the basic configuration of agents. As mentioned earlier, for the purposes of this analysis we viewed the interacting system as being composed of three agents illustrated in the following diagram: the user, the library interface, and the library collection.

Interaction system – user, interface, collection

In analysing the interaction we find that there is a need for an 'organisational' agent – one which takes responsibility for organising search results. In the observation we found that the user took responsibility for organising relevant articles simply by printing them and producing a physical collection of relevant articles. However, libraries such as the ACM digital library provide organisational agents (referred to in the case of ACM as bookshelves) which the user in question did not use. There may be many reasons for this including their lack of awareness of the bookshelf (as discussed previously in terms of interactional lucidity). A more fundamental reason for this may be that each library provides an independent organisational agent meaning that relevant articles would be spread across different agents as illustrated in the following diagram

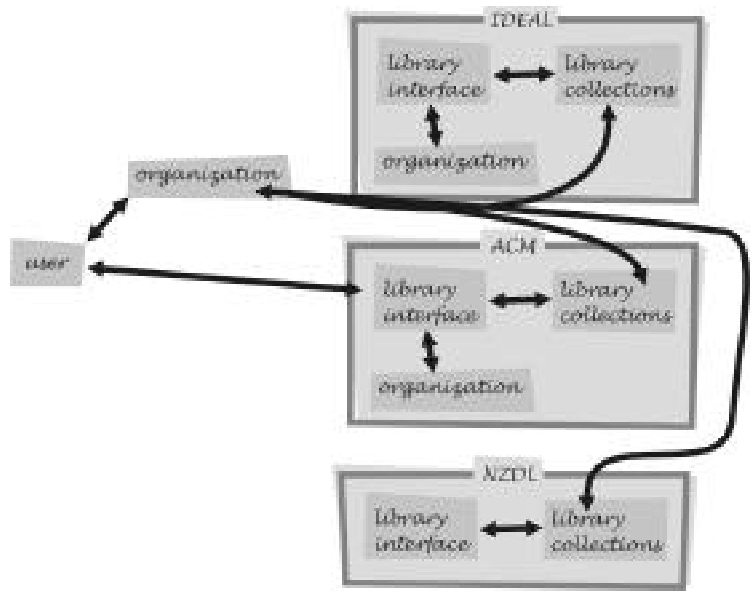


Interaction system – organisation and different collections

Organising articles by printing them out brings its own problems as illustrated in the following transcript – the user no longer knows what articles she has collected, nor which ones will be printed. A related problem is that once printed the user may have difficulty relating the articles back to their originating collections if necessary *e.g.* to find citation information.

```
[- ACM DL window showing ontology paper]
OK.
So I don't know how I will know if I've got everything I've printed out.
Because I'm losing track of it completely.
But I think that's been sent and not needed anymore.
```

One possible design solution would be to develop a separate organising agent as illustrated in the following diagram. This would allow the user to organise articles from various libraries whilst maintaining links to the originating collections. One could argue that a typical filing system could meet this requirement, but such filing systems typically provide poor information about the content of documents and their origins.



Interaction system – design to include organisation agent

Conclusions

In terms of interaction, our study led us to several interaction design considerations. These stemmed from more detailed understandings of what searching and browsing involves: in particular, not only supporting familiarisation, but also support for extended interaction over time. We also suggested that interactions should have high event potential as well as good discriminability between potential events within the interaction. Future work needs to develop our understandings of what searching and browsing are, and how these understandings relate to strategies people employ when using information resources.

Acknowledgements

We are grateful to the user whose transcript is the focus for this report. This work is funded by EPSRC Grant GR/M81748.

References

Barnard, P. & May, J. (1999). Representing cognitive activity in complex tasks. *Human Computer Interaction*, ***14***, 93-158.

Blandford, A. E., Harrison, M. D. & Barnard, P. J. (1995) Using Interaction Framework to guide the design of interactive systems *International Journal of Human-Computer Studies*, 43, 101-130.

Clark, H.H. and Brennan, S.E. (1991). Grounding in Communication. 127-149 In Resnick, L.B., Levine, J and Behrend, S.D. (Eds.) *Perspectives on Socially Shared Cognition.* Washington DC.: American Psychological Association.

Healey, P. and Bryan-Kinns, N. (2000). Analysing Asynchronous Collaboration. *Proceedings of HCI 2000*.

Sartre, J.-P. (1963). *Nausea*. (translated from French version of 1938). Penguin Books.