# A Comparison of Algorithms for Hypertext Notes Network Linearisation

MIKE HA LE

## 2. Hypertext linearisation

There are two main occasions where hypertext linearisation can be of use: in supporting a reader browsing through a hypertext database, and in providing a version of a hypertext in linear form.

### 2.1 Supporting browsing

A reader browsing through a hypertext may not always wish to choose, or be able to make, an informed choice of which path to follow. A hypertext display will typically show the names of the successor nodes, but will not tell the reader whether a link is important, nor whether it leads to a major new part of the network or just to a dead end. A linearisation algorithm could provide a default linear path for the hypertext reader, selected according to criteria such as the reader's interests (for example, a path containing chosen topics), the reader's abilities (for example, a path containing explanations and digressions matched to the reader's knowledge), or a given point of view (for example, material chosen to represent one side of an argument).

### 2.2 Hypertext to linear text

There are many reasons to print out a copy of a hypertext database, such as to check it for completeness, to provide a paper copy in a report or dissertation, or to produce a book version of a hypertext reference work. The printed version may still indicate the links between nodes as conventional text references, but the hypertext must be linearised to fit the format of a printed work, with pages collated into sequential order. A simple algorithm, such as listing the nodes in order of time of creation, or in alphabetic order, may be acceptable for some purposes, but others, such as producing a printed reference book with a preferred linear reading, may require a more sophisticated linearisation.

Linear texts have served as the main means of storing and transmitting knowledge over the past five thousand years and there is no sign that printed books, magazines and journals are about to be superseded. For most purposes, there is no need to present the reader with a hypertext; writers want to be in control of the order in which a text is read, and readers are content to be carried along by a narrative flow.

One use of hypertext is, instead of giving choice to a reader, to assist the author of a conventional text, by providing 'mind maps' (Buzan, [illegible]), or 'notes networks' (Trigg & Irish, [illegible]; Sharples et al., [illegible]) which act as 'intermediate representations' between mental associations and draft text. A productive way for a writer to generate new material is to follow mental trails of association, setting down each new idea as a note, and linking the notes together to form a visual map of the topics to be covered. The use of notes networks as an aid to study and writing has been advocated by Buzan ([illegible]), Trigg and Suchman ([illegible]) and others. Creativity comes, in part, from exploring a 'conceptual space' of related ideas (Boden, [illegible]). An interlinked network of notes representing ideas or topics can make this creativity visible and explicit.

Xerox NoteCards (Trigg & Irish, [illegible]) is a computer based implementation of a notes network. A user can 'brainstorm' ideas, write each one on a simulated file card, and link the cards together graphically on the screen. Two or more people can work with the same set of

during the creation of the network. Information potentially available to the algorithm includes node names, node contents, link names, connectivity, time of creation and spatial layout.

***It should be deterministic***

The order of linearisation should be fully determined by the information contained in the network, not by the processing priorities of the programming language.

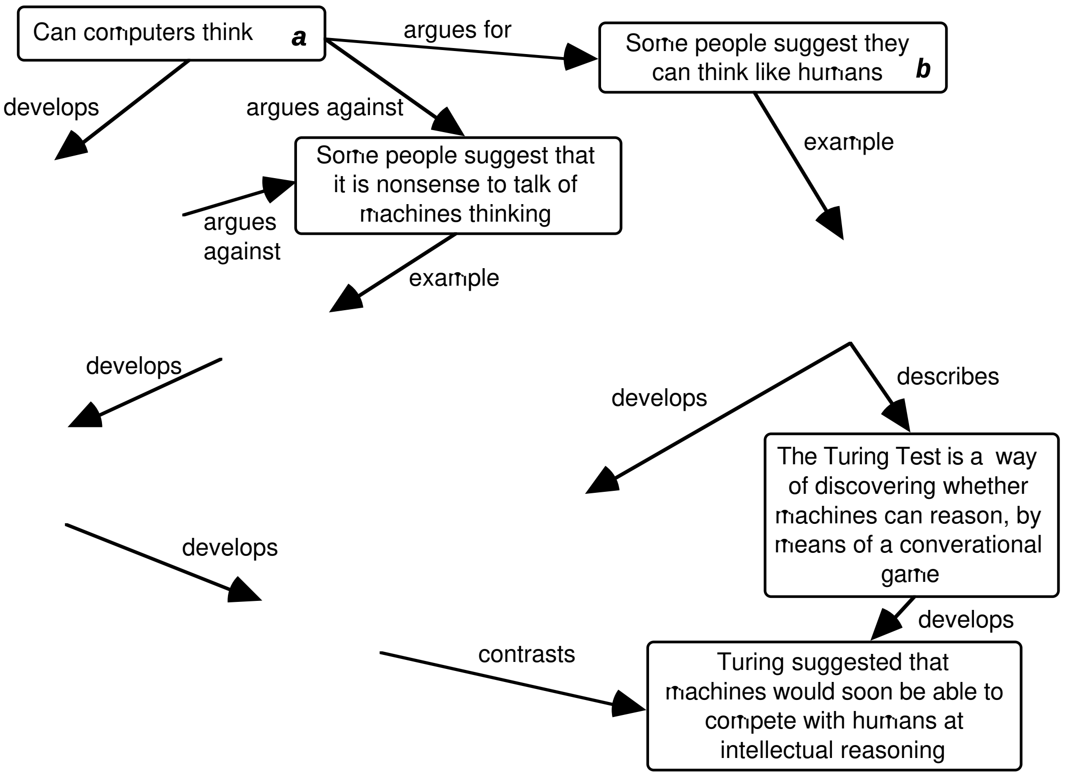
***It should produce a linearisation which is acceptable to a human writer.***

A linearisation algorithm differs from a search algorithm in that the criterion for success is not to reach a specified goal, but to produce an order of nodes which meets the expectations of a reader. A coherent ('easy') narrative leads the reader onwards, by following trains of association that resonate with the reader's experience. Thus, it follows that a linearisation algorithm can only be judged by subjective criteria.

## 3. Descriptions of the algorithms

The experiments described in this paper compare two algorithms for hypertext linearisation. Both algorithms satisfy the requirements above, apart from the 'deterministic' criterion. The best first algorithm has four heuristics to resolve similarity of links, and other information such as the time of node creation can be employed to ensure the algorithm is deterministic. They treat a hypertext as a directed, labelled graph. They ignore the contents of the nodes and so can also be used to linearise a network containing non textual elements, but make use of the 'priority' of the link type. The algorithms have both been implemented in Prolog.

Both algorithms can be implemented to run in time $O(nl)$ where $n$ is the number of nodes in the network and $l$ is the mean number of links from each node. Although $l$ will depend on node are included before large ones. This extension was not implemented for the experiments described below.


Can computers think a
argues for
Some people suggest they can think like humans b
develops
argues against
Some people suggest that it is nonsense to talk of machines thinking
argues against
example
example
develops
develops
describes
The Turing Test is a way of discovering whether machines can reason, by means of a converational game
develops
develops
contrasts
Turing suggested that machines would soon be able to compete with humans at intellectual reasoning

-F nd a untrave ed n s fro eac node n LINEA I ED- e ove eac n fro t e rap -

-Mer e t e n s w t O EN[ owest va ue n to t e front-

a-If t ere are two or ore cand date n s w t t e sa e va ue[ t en put t e on O EN n order of t e s ze of sub rap ead n fro t e n [ ar est sub rap to t e front- e s ze of t e sub rap s ca cu ated on t e pruned rap [ w t nodes a ready on LINEA I ED re oved-

b-If t ere are two or ore cand date n s w t t e sa e va ue[ and sa e s ze of sub rap [ t en put t e on O EN n order of t e va ue of t e n ead n to t e node fro w c t e n s depart owest va ue to t e front -

c-If t ere are two or ore cand date n s[ w t t e sa e va ue[ s ze of sub rap [ and va ue of nco n n [ t en put t e on O EN n order of t e d stance of t e n fro t e start node furt est fro t e start node to t e front -

d-If t ere are two or ore n s at t e sa e d stance fro t e start node[ and one or ore s a ready on O EN[ t en put any new n s n front of t e one s a ready on O EN-

e-If t ere are st two or ore cand date n s[ t en put t e on O EN n so e order deter ned by nfor at on conta ned n t e networ suc as t e t e t e node was created -

-If O EN s e pty and not a nodes ave been re oved fro t e rap [ t en reverse a t e re a n n n s n t e rap -Go to -

-If O EN s e pty t en stop-

- e ove t e n at t e front of O EN-

-Ca t e node fro w c t s n departs t e FOC NODE and t e node to w c t e n po nts t e CCE O NODE-

-If t e CCE O NODE s a ready on LINEA I ED t en o to -

-Add t e CCE O NODE to LINEA I ED n pos t on ed ate y after t e FOC NODE-

-F nd a untrave ed n s fro t e CCE O NODE-

-Go to -

*Figure 4. The Best First algorithm*

e eur st cs are des ned to favour t e c o ce of pr or ty n s w c ead to or fro ar er and t erefore ore e y to be portant sub parts of t e networ -Heur st c a requ res t e s ze of t e sub rap fro a node to be co puted[ but t e co putat on can be bounded w t out s n f cant y affect n t e operat on of t e a or t -L ne a ows for networ s w ere so e nodes cannot be reac ed due to t e d rect on of t e n s- It s needed because occas ona y a sub ect connected a c uster of nodes to t e an networ w t a n n t e reverse d rect on- Add n a reac ab e nodes to t e near st and t en revers n a t e re a n n n s as t e effect of nc ud n t e re a n n nodes n t e near sat on[ but at ow pr or ty-

e best f rst a or t overco es t e part cu ar prob e s of t e c b n a or t [ produc n a near sat on of . a d b c e f for t e networ n F ure [ and . a d b c e f for t e networ n F ure . It a so as t e advanta e of f n t e LINEA I ED st n order of n pr or ty[ so t at[ by v n a cut off va ue for t e n pr or ty[ t can f ter out parts of t e ypertext networ [ reta n n on y t ose nodes on a n pat s

### 3.4 The algorithms in operation

o ve an exa p e of t e a or t s n operat on[ F ure s ows a s a notes networ produced by a wr ter on t e top c of /Can co puters t n = A or t bu ds up t e near sed st n t e order s own n F ure

a
a h
a h g
a h g i
a h g i j
a h g i j k
a h g i j k f
a h g i j k f b
a h g i j k f b c
a h g i j k f b c d
a h g i j k f b c d e

***Figure 5. Order of nodes produced by the hillclimbing algorithm for the network in Figure 4***

e near sed text correspond n to t e f na order of nodes s as fo ows

Can computers think

Computers may be able to think in non human ways.

Some people suggest that it is nonsense to talk of machines thinking.

f filling the.hs r mC.S4 T TL Tw So4444 c . . Tc Tw f fiTL Tc , TL Tc , Algorithm 1 creates the linear seqlist in the order shown in Figure , and the final order of nodes produces the linear text below

Can computers think

Computers may be able to think in non human ways.

Some people suggest computers can think like humans.

Turing suggested an operational definition of thinking.

The Turing Test is a way of discovering whether machines can reason, by means of a conversational game.

This is reminiscent of behaviourist psychology.

Some people suggest that it is nonsense to talk of machines thinking.

Searle argues that machines do not have intentionality.

Machines have syntax but no semantics.

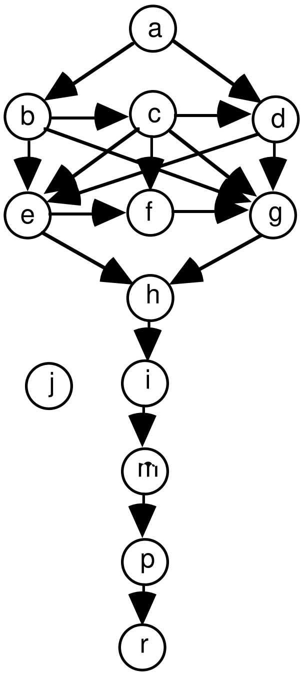
Machine thought is impossible in principle.

Turing suggested that machines would soon be able to compete with humans at intellectual reasoning.

The ordering of Algorithm 1 provides a more plausible framework for a linear text, which the writer could then flesh out with connecting phrases to create a first draft

### Can computers think?

Computers may be able to think in non human ways, ***but*** some people suggest computers can think like humans. Turing suggested an operational definition of thinking. The Turing Test is a way of discovering whether machines can reason, by means of a conversational game. This is reminiscent of behaviourist psychology. ***However,*** some people suggest that it is nonsense to talk of machines thinking. Searle argues that machines do not have intentionality. Machines have syntax but no semantics.


a
b
c
d
e
f
g
h
j
i
m
p
r

There is no generally agreed set of basic link types, and hypertext systems which provide prespecified links range from IBIS, with eight link types intended for developing argumentation (Halasz and Conklin, [illegible]) to Trigg's TEXTNET [illegible], with over eighty different link types. The set of links chosen for the experiment was intended to be small enough to be manageable by the experimental subjects, but large enough to cover the main types of conceptual relation for the text types used in the experiment. The subjects were instructed to say during the experiment if they required any further link types, and at the end each subject was asked to suggest further link types which they might have found useful. No subject indicated that an additional link type would be necessary to complete the task. Three subjects suggested links which they might have found useful: each of the three people suggested two or more links. These were "undermines", "contexts", "codescription", "but", "new episode", "facilitates", "suggests".

### 3.1.4 Design

A repeated measures design was used, with each subject producing a hypertext for each of the four texts. The order of texts was counter balanced.

### 3.1.5 Procedure

Each subject was shown a list of link types and the experimenter explained the meaning of each of the link types. The subject was also shown an example hypertext (Figure [illegible]). The subject was then given the set of cards containing the text chunks for the first example text, as well as the text itself. The text was available for reference during the experiment. The subject was asked to stick the cards onto a whiteboard and to use a board marker to draw in relational links. Each link should have an arrow indicating its direction and a label chosen from the set of available link types. The subject was encouraged to use whichever strategy seemed natural to construct the hypertext. Some subjects placed all the cards on the board and then drew in the links; others added links after placing each card. Subjects were allowed as much time as they wished to carry out the task. The experimenter recorded the layout of the hypertext on paper as the subject created it. When the subject was satisfied that the hypertext was complete the experimenter removed it from the board and gave the subject the next set of cards. The experiment ended when the subject had created four hypertexts.

|  | Hillclimbing Labelled |  |  |  |  | Best First Labelled |  |  |  |  | Hillclimbing Unlabelled |  |  |  |  | Best First Unlabelled |  |  |  |  |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
|  | seq | cmp | cau | des | ***mn*** | seq | cmp | cau | des | ***mn*** | seq | cmp | cau | des | ***mn*** | seq | cmp | cau | des | ***mn*** |
| S |  | [illegible] |  | 4 | . [illegible] | 4 |  |  | [illegible] | . [illegible] |  | [illegible] |  | [illegible] | .[illegible] | 4 | [illegible] | 4 | [illegible] | 4.[illegible] |

Table [illegible] shows the mean scores for the human evaluator and the scores produced by the least cost algorithm. The algorithm produced scores in the range [illegible] to [illegible], these have been normalised to ease comparison in the table. There was a significant correlation between the scores ($r_s$ [illegible], p [illegible]).

All the original texts were rated [illegible] by the evaluator. The highest score for any of the linearisation algorithms is [illegible] for the best first algorithm applied to the labelled Description Hypertext. The randomly ordered texts were all rated [illegible]. The lowest score for any of the algorithms is [illegible] for the hill climbing algorithm applied to the unlabelled Sequence Hypertext.

The mean score for the best first algorithm applied to the hypertexts with labelled links is [illegible], and the corresponding mean score for the hill climbing algorithm is [illegible]. A Wilcoxon test (N [illegible], adjusted for ties) shows the difference between the algorithms to be significant at p [illegible] (one tailed).

The scores for best first algorithm applied to the hypertexts with the labelled links ([illegible]) and to the hypertexts with the link values removed ([illegible]) are almost identical.

### 4.3 Discussion

The good correlation between the ratings of the human evaluator and the least cost restoration scores indicates that computing the restoration distance of a linearised text from the original text is a useful means of measuring the effectiveness of linearisation and could provide a comparative test of new linearisation algorithms. Any new algorithm could be applied to the hypertexts used in this experiment and the linearisations could be reliably compared with those of the two algorithms tested here.

As expected, the best first algorithm was significantly more effective at linearising the hypertexts than the hill climbing algorithm, and the mean score of [illegible] (above the "somewhat organised, somewhat disorganised" level) suggests that it produced texts with sufficient organisation to be useful as first drafts. The low score of [illegible] for the Causation Hypertext suggests that automatic linearisation may be less useful for hypertexts which contain a number of distinct, but related, topics. One approach may be to group the nodes into topics by hand and then apply a version of the algorithm which keeps together text on the same topic.

The unexpected finding was that removing information about link types did not alter the effectiveness of the linearisation. There are a number of possible explanations for this result. The subjects may not have had enough practice in creating hypertexts to be able to put appropriate labels to the links, or the range of link types may not be adequate to indicate the conceptual links in the text, or the algorithm may not make good use of link information due to poor heuristics, or the priorities assigned to the link types may be inappropriate. It would be possible to investigate the latter explanation by using the results produced by the computer evaluation of the linearised texts as input to a learning algorithm which determines the optimal priority for each link type.

In this experiment the subjects created hypertexts from published linear texts by applying their skills of reading comprehension to make explicit the referential links embedded in the text. The experiment measured how effective the linearisation algorithms are in selecting links and traversing them in an appropriate order. But creating a hypertext as part of writing is not quite the same activity. A writer, in producing a notes network, is following trails of mental association, with no textual cues for guidance. A hypertext produced during the writing process may have links indicating deep conceptual relations, rather than surface textual ones, and may thus be more difficult for an algorithm to linearise. The second experiment tests this possibility.

## 4.4 Experiment 2

### 4.4.1 *Rationale*

The aim of the experiment is to test the algorithms on hypertext notes networks generated as part of a writing activity. It differs from experiment one in that the subjects are generating their own hypertexts on a given topic. The assumption is that the hypertext acts as a means of 'externalising cognition', allowing the writer to represent a pattern of mental associations between topic items. The two linearisation algorithms were compared against human linearisation, carried out by the authors of the hypertexts, and a random ordering of nodes. The linearisations were scored by two human assessors on a five point scale for textual organisation.

***Hypothesis 1:*** The hand linearised texts will have higher ratings than the best method of automatic linearisation.

***Hypothesis 2:*** All the automatically linearised texts will have higher ratings than the random ordering.

***Hypothesis 3:*** The best first algorithm with labelled links will produce higher ratings than the combination algorithm with labelled links.

***Hypothesis 4:*** The best first algorithm with labelled links will produce higher ratings than the best first algorithm with no link information.

### 4.4.2 *Subjects*

The subjects were the same as for experiment one.

### 4.4.3 *Materials*

The subjects were shown the same card of link types as for experiment one. Each subject was given a stack of blank cm x cm file cards on which to write the text. The materials for creating the hypertexts were as for experiment one.

### 4.4.4 *Design*

Each subject produced one hypertext.

### 4.4.5 *Procedure*

The subjects were given a list of three topics and asked to choose one topic on which they would create a hypertext. The topics were: 'How to choose a summer holiday', 'Should I sell my car and cycle to work', and 'The role of Britain in Europe'. Four of the subjects chose the Holiday topic, seven chose the Bicycle topic, and one chose the Europe topic.

Each subject was given a stack of twenty blank file cards and was asked to generate short sentences on the topic, writing each sentence on a separate card. The subject was asked to by t e sub ects= an near sat on[ were rando y ordered and ven to t e two ndependent eva uators— e eva uators ar ed t e texts us n t e sa e f ve po nt sca e as n t e prev ous exper ent—For out of t e texts t e ar s d ffered by two sca e po nts[ and for two texts t e ar ers d ffered by t ree sca e po nts— e eva uators were as ed to re ar t ese texts w t out reference to t e prev ous scores— e re ar n produced s x cases w ere t e eva uators d ffered by ore t an one sca e po nt—

### 4.5 Results of Experiment 2

ab e s ows t e scores after re ar n of t e two eva uators for eac of t e texts produced by t e near s n a or t s[ rando order n [ and and near sat on—

| | Hillclimbing Labelled | | Best First Labelled | | Hillclimbing Unlabelled | | Best First Unlabelled | | Random | | Hand Linearised | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Eval | Eval | Eval | Eval | Eval | Eval | Eval | Eval | Eval | Eval | Eval | Eval |
| S | 4 | | 4 | | [illegible] | 4 | [illegible] | 4 | 4 | | [illegible] | 4 |
| S | 4 | 4 | 4 | | [illegible] | | | | | | 4 | 4 |
| S | | 4 | 4 | | | 4 | | | | | [illegible] | [illegible] |
| S4 | 4 | 4 | | 4 | 4 | 4 | | 4 | | | [illegible] | 4 |
| S | 4 | | | | 4 | | | | | | 4 | |
| S | 4 | | [illegible] | | 4 | | 4 | 4 | | | 4 | 4 |
| S | | | | | | | | | | | 4 | 4 |
| S | | | 4 | [illegible] | | | 4 | [illegible] | | | [illegible] | [illegible] |
| S | | | | | | | | | | | [illegible] | 4 |
| S | 4 | 4 | | 4 | | 4 | | 4 | | | [illegible] | [illegible] |
| S | | 4 | [illegible] | 4 | 4 | 4 | 4 | [illegible] | | | [illegible] | [illegible] |
| S | | | | | | 4 | 4 | 4 | | | 4 | [illegible] |
| Mean | . | . | . | . | .4 | . | . | . | . | . | 4. | 4. |
| Mean | . | | .4 | | . | | .4 | | .4 | | 4.4 | |

*Table 4. Scores of the two evaluators for the linearised texts.*

e corre at on bet t e scores of t e two eva uators s r — [ s n f cant at p — — e ean scores of t e two ar ers for eac of t e text types see F ure s c b n abe ed —[ best f rst abe ed — c b n un abe ed — best f rst un abe ed — rando — and near sed — —A t e auto at ca y near sed texts ave ower rat n s t an t e texts near sed by and[ and a are er t an t e rando order n s— e d fferences bet t e scores are s n f cant n bot cases at p — coxon[ N [ one ta ed — e two scores produced by t e best f rst a or t are er t an t ose for t e c b n one but t e d fferences are not s n f cant—As for exper ent one[ t ere s no s n f cant d fference bet t e scores for t e best f rst a or t w t abe nfor at on and w t out t—

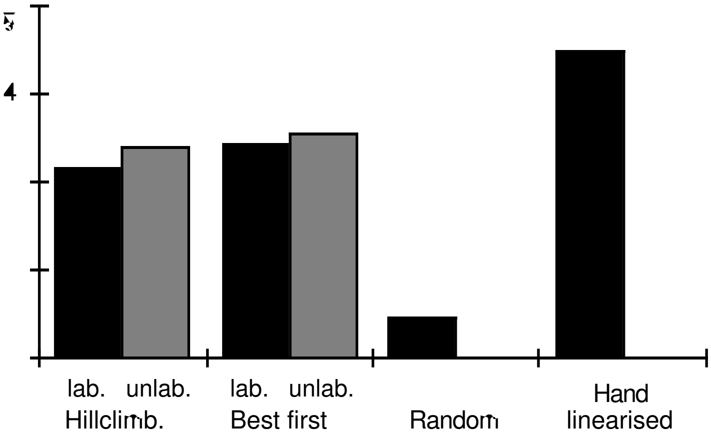


*Figure 8. Mean scores for the linearised texts*

### 4.6 Discussion

Bot a or t s produce near sat ons n t e ran e between /so ew at or an sed so ew at d sor an sed=and/acceptab y or an sed=It nd cates t at auto at c near sat on cou d prov de a usefu br d e between an deas or an ser and a text ed tor as part of a wr t n env ron ent— e near text ay we need to be ed ted[ but t e a or t s[ t

reverse d rect on for exa p e[ t e /causes= n wou d be ven a pr or ty for ts reverse d rect on correspond n to t e/ s caused by=re at on − s was t e favoured et od[ but t was not p e ented due to a ac of ev dence to u de t e c o ce of n we ts−

## 6. Conclusions

e ave descr bed a robust enera a or t for notes networ near sat on w c as been p e ented as part of a wr t n env ron ent w c co b nes an deas or an ser w t a docu ent ed tor− e exper ents su est t at t e best f rst near sat on a or t s acceptab e for creat n a f rst draft of a near text fro a notes networ but t at furt er wor s needed to a e ood use of t e n nfor at on to u de near sat on and to eva uate t e use of t e a or t as part of a wr t n env ron ent−

## References

Boden[ M− *The Creative Mind: Myths and Mechanisms*− London e denfe d and N co son−

Bo ter[ J−D− Joyce[ M−and t [ J−B− *Storyspace: Hypertext Writing Environment.*