Negotiating Access within Wiki

A System to Construct and Maintain a Taxonomy of Access Rules

Andrew Lincoln Burrow
RMIT University
GPO Box 2476V
Melbourne VIC 3001, Australia
andrew.burrow@rmit.edu.au

ABSTRACT

A wiki hypertext is typically accessible and editable by all. While this removes impediments to collaboration, it often deters participants who would rather incubate ideas before bringing them to the group. This is especially the case where creative ideas are at stake. Creating additional wikis with restricted access is a costly solution: it requires participants to distinguish between and navigate between wikis; it requires administrators to construct wikis and their access rules; and it does not account for the movement of content from private to public. In this paper, we describe a system that augments the hypertext in order to solve these problems. This system automatically creates and maintains access rules in response to browsing and editing of the wiki hypertext. In doing so, it improves the targeting of documents in the hypertext, and identifies significant collections of documents and participants.

Categories and Subject Descriptors

G.2.3 [**Discrete Mathematics**]: Applications; H.4.3 [**Information Systems Applications**]: Communications Applications—*Bulletin boards*; H.5.4 [**Information Systems Applications**]: Hypertext/Hypermedia—*Architectures*

General Terms

Design, Algorithms

Keywords

order theory, hyperlink types, collaboration

1. INTRODUCTION

The open and collective authorship of hypertext is the basis of wiki. It was pioneered by Ward Cunningham in the *Portland Pattern Repository*[2] as a means to discuss software engineering strategies. In essence, wiki provides a model for collaboration, because it removes many impediments to shared authorship. However, it does not represent and restrict access to a document, and is thus not in a position to model the movement of a document from a narrower to a wider audience. Where creative ideas are at stake, this is often a barrier to the use of wiki.

We observe two causes for wariness toward early disclosure. The first is the need to incubate ideas: premature comparison or criticism endangers the development of an idea, because every new idea requires a certain suspension of disbelief. The second is a conundrum of collaboration: we must share ideas to realize their value, but in doing so we diminish our own control. For these reasons, certain valuable types of collaboration require flexibility in determining access rules.

The purpose of this paper is to describe extensions to wiki that introduce flexibility in determining access. They are designed to introduce minimal changes to wiki. The central idea is to represent the access rules of each document in order to model its progressive exposure. This also allows key audiences to be identified as sharing key collections of documents. These extensions allow wiki to support collaboration in new settings.

We use the term *lattice structured wiki* to denote our proposed system. This system comprises a hypertext and an additional data structure that augments the hypertext to represent access rules. In use, the system behaves as a wiki, except where access rules are updated, in which case it uses the additional data structure to interpret the participant's interactions with respect to modifications to the access rules. This system is defined in Section 2.

1.1 Application Scenario

We now provide an application scenario to motivate our presentation. In this scenario, access rules change over the lifetime of the hypertext, differ between documents in the hypertext, and record outcomes of the collaborative process. These properties motivate our extensions to wiki.

Consider a funding body that has been newly created to foster cooperative research. It funds research by teams that involve two tertiary institutions and an industry sponsor. Note that this requires participants to form teams and collaborate on proposals, and thus exchange information at various levels of sensitivity. We can form a simplified time line of this activity in terms of a document collection and access to its elements.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
HT'04, August 9–13, 2004, Santa Cruz, California, USA.
Copyright 2004 ACM 1-58113-848-2/04/0008 ...$5.00.

During the initial phase, documents are gathered and distributed to two groups. The initial collection of documents is public. It includes the body's charters, and information for potential applicants. The next collection of documents is collated by the body in response to expressions of interest. It is a register of participants that includes their skills and projected level of involvement. Access to this collection is restricted to the participants. This arrangement of access rules is common to many hypertext systems.

During the team building phase, two new types of document collection arise. In the first instance, participants advertise ideas to one another by releasing outlines. Participants divide their time between evaluating outlines and expanding their own, and occasionally negotiate restricted access to discuss sensitive information. This peer evaluation occurs in waves: overlapping groups form around interests and skills, and then divide again to form teams. The team building phase concludes with proposal writing. Each newly formed team collaborates on a proposal, and then submits this proposal for funding. Throughout the team building phase, varied access rules arise at the discretion of participants, and reflect the composition of groups and teams. In particular, documents accumulate readers according to negotiations between participants.

While it is possible to elaborate this scenario to additional movements in the access rules, it reflects our observations in using wiki for collaboration. Namely, we observe: participants struggling with the circulation between private and collaborative efforts, because an open wiki does not provide cover for tentative steps; requests to sequester space for confidential group work, because an open wiki forgoes timely release; and considerable housekeeping in moving documents from confidential wikis to public wikis, because confidential and public documents are frequently intermingled. Extensions to wiki that responded effectively to these three cases would allow wiki to service the application scenario described above.

1.2 Minimal Definition of Wiki

We now provide a simple definition of wiki. This grounds our proposal, and allows us to consider the existing and related work.

A wiki is a collaborative hypertext environment, with an emphasis on open and easy access and modification[2]. It comprises a markup language for hypertext defined by wiki name and structured text rules, and a collection of tools to view, edit, and interrogate the resulting hypertext. Documents in the hypertext are termed *pages* or *topics*. The markup language elides keywords, so that it is feasible to edit the hypertext without specialized authoring tools. The wiki software amplifies the hypertext, for example, by allowing incident hyperlinks to be listed for a page. Thus, a wiki system is defined by a syntax for hypertext, and a web application implementing the hypertext.

The goal of the hypertext syntax is to allow markup without intrusive syntactical elements, and thus without requiring specialized tools. Therefore, wiki systems attempt to recognize plain text as hypertext. The first step is to reserve a class of words for naming pages. These *wiki names* are recognized by lexical rules based on capitalization, and each utterance is rendered as a hyperlink to the named page. The second step is to accord a style of plain text layout the privilege of syntax. Thus, *structured text* is a markup language that formalizes conventions in typewritten text. Structured texts are rendered using the markup language of the browser, namely, HTML. This system provides a hypertext authoring environment with interesting properties.

Wiki names are pivotal to wiki, while structured text is an enabling technology. The most interesting properties of wiki hypertext derive from wiki names. In comparison, structured text is best understood as a mechanism for simplified formatting. In practice, the intrusiveness of existing structured text systems depends on the expectations of the participant. However, improvements to browser software allow replacement by word processing controls. Therefore, we can expect to provide a simplified formatting system to a variety of audiences. Any formatting system that emphasizes open and easy access and modification, and preserves the role of wiki names will satisfy our definition of structured text.

The core of the wiki web application is a system to browse, search, and edit the hypertext. Each wiki has a *front page* providing an entry point to the hypertext. Participants traverse the wiki by following the hyperlinks corresponding to wiki name utterances. A wiki page is edited in situ by switching to a source view. Search facilities provide additional means to browse the hypertext. In particular, most wikis provide: a *recent changes* page listing the most recently edited pages; page *backlinks* showing the incident hyperlinks for a page; and full text search. In this paper, we consider further extensions to the web application that amplify the hypertext.

We now restate the definitions of wiki hypertext and the act of browsing and editing this hypertext. We do this for an idealized and simple wiki that includes recent changes, back links, and search.

Wiki Hypertext A wiki is a web accessible hypertext that comprises a collection of pages and inter page links, where each link is anchored by the name of the target page.

Wiki Browsing A participant browses a wiki by following links corresponding to the occurrence of wiki names in the hypertext. Browsing is augmented by three forms of search: a listing of the most recently edited pages, a mechanism to discover pages by content, and a listing of pages that link to the current page. Since wiki names may refer to pages that do not yet exist, browsing has one special case: pages are explicitly created on the first visit.

Wiki Editing Upon visiting an existing page, a participant can edit the page in situ. Upon following a wiki name to a non-existent page, a participant can create the page in situ. Once an edit is committed, the wiki renders the page according to the simplified formatting system, and generates a hyperlink for each wiki name occurrence.

Figure 1 depicts the construction of a very small wiki. It demonstrates the outward growth from existing pages, and implicit page creation. Each page is brought into existence in two stages. In the first stage, the wiki name is used on an existing page. In the second stage, a hyperlink is followed to the nonexistent page, and the page is created on the first visit. In this case the visitor supplies the initial content.

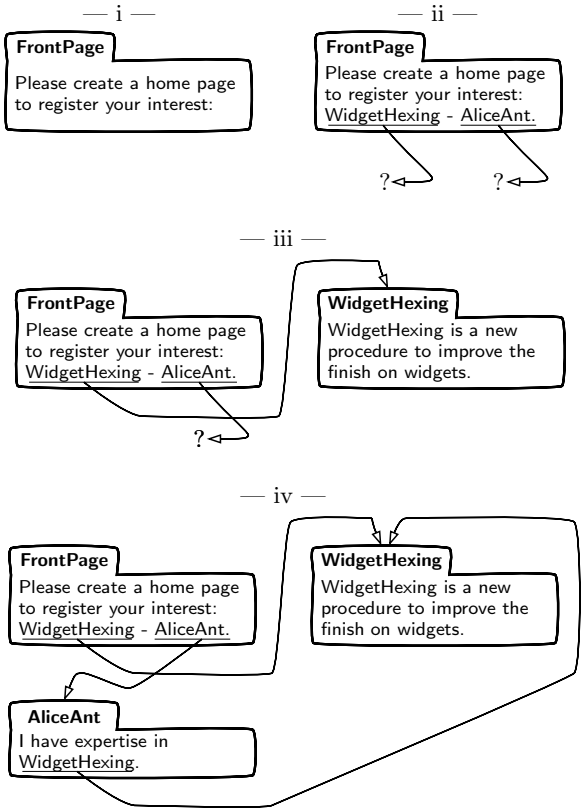

Figure 1: A sequence of wiki edits: (i) a singleton wiki; (ii) Alice edits `FrontPage` to include two wiki names; (iii) Alice creates `WidgetHexing`; and (iv) Alice creates `AliceAnt`.

1.3 Related Work

This paper addresses the role of hypertext in knowledge management where the goal is to foster collaboration in teams. Ricardo terms this use *knowledge management as social nexus*[8]. Following his analysis, the proposed extensions fold the maintenance of access rules into the process of content access and creation. This activity occurs in the setting of existing wiki practices. Therefore, this work relates to techniques that modify the hypertext according to access patterns, and also to existing wiki systems.

De Bra et al [4] describe adaptive hypermedia in terms of content-adaption and link-adaption occurring in response to user access patterns. Lattice structured wiki can be understood in terms of link-adaption, in the sense that access rules decide which links are available to a participant. The key distinction is that wiki participants both read and write the hypertext, hence the system determines link-adaption in consultation with document owners. This is possible because the authors are expected to share the space with the readers. The basis of link-adaption in lattice structured wiki is a form of link typing.

Lattice structured wiki employs a very simple link taxonomy. Link typing does not reflect the meaning of the hypertext as suggested by [10]. The only distinction made is whether a link connects documents with different access rules. Link classification is automatic. Since the link taxonomy is described by a taxonomy of access rules, the system bears a close relationship to the interpretation of hypertext by semantic nets [7]. In the case of lattice structured wiki, the role of inference is to interact with participants in order to classify documents according to their accessibility.

Lattice structured wiki also relates to knowledge representation, in particular formal concept analysis[11]. Thus, it strongly resembles the formal concept analysis of document collections. However, whereas [1] draws attributes from matches against an existing dictionary, lattice structured wiki dynamically constructs the classification under the effects of wiki browsing. Furthermore, the classification explicates public and private space rather than search results.

We conclude by considering wiki solutions to confidentiality. Our intention is that existing facilities for structuring wiki hypertext are preserved. Therefore, we restrict our attention to features that relate to access control. Of particular note is the Twiki [9] system of Twiki webs. These spaces are subdirectories. While each has its own access rules, references to its documents contain the name of the subdirectory. This requires participants to distinguish between and navigate between these subdirectories, and requires administrators to construct subdirectories and their access rules. Furthermore, it does not account for the movement of content from private to public. Lattice structured wiki differs in its approach because it does not demand that documents with different access rules be stored in separate collections, and because the access rules are inferred in the process of link-adaption.

2. LATTICE STRUCTURED WIKI

We now describe a framework that extends wiki to subdivide a single wiki into a collection of privileged views. *Lattice structured wiki* operates via conventional interaction with the hypertext: authoring and browsing. By browsing the hypertext, participants effectively bid for the exposure of nodes. We use this mercantile metaphor to describe the operation of the system.

The subdivision of the wiki is defined in terms of subsets of participants and subsets of documents. Given the system refers to certain subsets of participants and documents, we use the term *participant subset* to denote such a subset of participants, and the term *document subset* to denote such a subset of documents. The collection of participant subsets is ordered by set inclusion, closed under intersection, and includes the set of all participants. That is, the collection of participant subsets is a *topped intersection structure*. Since it is finite, it is also a *complete lattice*. This paper does not attempt to cover order theory and lattices. Instead, we explain the necessary properties in the context of the Hasse Diagram, which is a complete and diagrammatic representation of a finite partial order. For an introduction to the field, we refer readers to [3].

We show that the lattice of participant subsets both models the access rules, and is maintained by simple inferences based on link access. We show that this corresponds to a readily explainable process. Furthermore, the lattice is not a structure apparent in the hypertext, but only in the access rules, so that a casual reader of the hypertext will be unaware of its existence. Where the lattice is apparent, it acts to suggest responses to a request to modify access rules.

The lattice and its sublattices also produce diagrams relating the participant subsets and document subsets. This technique and the connection between participant subsets

and document subsets are formalized by formal concept analysis [11]. Despite using formal concept analysis as justification, we do not consider this machinery further.

In summary, participants in a lattice structured wiki construct a hypertext in which subsets of participants view overlapping subsets of the hypertext. We interpret these actions as dynamically constructing collaboration spaces. The effect is to allow flexible migration of documents between spaces, so that topics can germinate in restricted spaces and migrate to more public spaces on demand. The basis of this system is an elegant characterization of implicit page creation in wiki. The system provides an ideal platform for sharing resources and collaborating on documents, because it does not require collaboration channels to be pre-built, yet can accommodate a wide variety of organizational structures.

2.1 Publishing in a Lattice

The first part of our exposition motivates the lattice structure. To do this we explain how a lattice of participant subsets is able to capture an interesting model of publishing. In this simple model, publishing is described as a process in which a document becomes more widely accessible. The purpose of this model is to describe how lattice structured wiki distributes documents across a patchwork of states of accessibility. We begin by describing the lattice as a diagram.

A *lattice structured wiki* comprises a structure over two collections of objects: documents and participants. It is a special class of partial order. However, we avoid introducing the machinery of order theory, and instead describe the lattice through its diagrammatic representation.

The *Hasse diagram* is composed of nodes and upward edges. In the case of lattice structured wiki, a node represents a *participant subset*. An upward path from a node x to a node y exists if and only if the participant subset at x is a strict subset of the participant subset at y. This can easily be confirmed for the example in Figure 2. In set theoretical language, the upward edges define a binary relation, and the transitive closure of this relation is set inclusion. In fact, the diagram contains the minimum set of edges necessary to recover the transitive closure. These facts are accounted for by order theory. However, the important point is that an upward path describes a sequence of participant subsets, such that each subset is a strict superset of the last.

This structure over participant subsets is a model for incremental publishing. In this model, we interpret the participant subsets as denoting participants sharing access to a collection of documents. The extremum nodes cover the boundary conditions. The top element of the lattice is the subset containing all participants. It obtains those documents that are accessible by all participants. The bottom element of the lattice is the empty set. It obtains those documents that are accessible by no participants. Between these two extremes are participant subsets that exclude some fraction of the total participants.

Stepwise upward movement of documents in the lattice generalizes both single stage publishing and implicit document creation. Upward movement generalizes single stage publishing, because moving a document along an upward edge in the lattice increments the accessibility of the document. Therefore, the lattice generates a collection of paths, where each describes stages in the dissemination of a document. Upward movement also generalizes implicit document

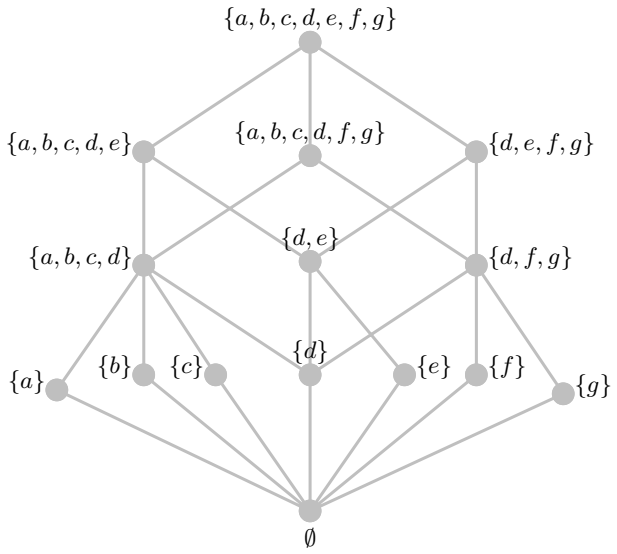

Figure 2: A lattice of participant subsets drawn from a set $\{a, b, c, d, e, f, g\}$ of participants.

creation, because the bottom node in the lattice obtains those documents that are inaccessible to all participants. Therefore, creating a document is equivalent to publishing a previously inaccessible document, because the created document moves upward from the bottom node into a node denoting access by its author. In this model, the bottom node represents the pool of unused wiki names, and the top node represents the wiki names of public documents.

This model brings two immediate benefits. Firstly, the model of publishing is incremental with respect to the growth of an audience. Secondly, the lattice structure allows greater freedom in the sequence of disclosures, because there are multiple upward paths between nodes. This feature distinguishes the lattice structure from the conventional tree structured hierarchy. Given these results, it is useful to test whether further lattice properties enjoy sensible interpretations.

The definitive property of a lattice is the existence of binary join and meet operators. Rather than provide an abstract definition, we describe the effect in terms of lattice structured wiki. Given a lattice L of participant subsets, and participant subsets $x, y \in L$, the *join* of x and y is the smallest participant subset $z \in L$ such that z is a superset of x and of y. Again, the existence of a join for each pair can be easily confirmed in Figure 2.

In lattice structured wiki, the join operator locates candidate participant subsets during publishing. Consider two participants a and e in Figure 2. If a and e wish to collaborate on a document, the smallest participant subset to which they both belong can be determined by the join of $\{a\}$ and $\{e\}$, in this case $\{a, b, c, d, e\}$. If the join is an unsuitable target, this failure indicates that the lattice should be extended to include new participant subsets. This is the subject of the scheme to build the lattice described in Section 2.6.

This dynamic view of movement on the lattice captures the intent of the lattice structured wiki. We now move on to produce a precise static mathematical definition. In turn, this informs the dynamic construction of a sequence of lat-

tice structures as the lattice evolves to accommodate collaboration.

2.2 Goals of Lattice Structured Wiki

So far we have motivated the lattice structure in terms of a publishing model, and sketched a static description of the structure. These views are antagonistic only in that we have yet to provide an account of how they interact. In order to frame this next task, we collect the following desirable properties of the system and its design.

Few Interface Concepts Throughout the design of lattice structured wiki, we should seek parsimony in the explanation of user interaction. While the system is free to exploit sophisticated mathematical structures and inferences, it should interact with users via a minimal number of new concepts.

Preservation Under Publishing As a document is published upward in the lattice, each link traversable by a participant should remain traversable by that participant. Namely, a document becoming more public should not reduce accessibility for any participant.

Maintenance Through Publishing The act of publishing should be the mechanism by which the diversity of participant subsets is maintained. In particular, we should not require a separate mechanism to generate structure. This restates the goal of parsimony.

Reversible Publishing It should be possible to remove documents to narrower participant subsets. This avoids cluttering public space with documents that have served short term communication needs.

These goals channel our efforts to describe the mechanics of lattice structured wiki. Given the goal of parsimony, our approaches to the remaining goals are determined by the minimal definition of wiki from Section 1.2, and the account of publishing in the lattice from Section 2.1. That is, we endeavor to describe a system based on interactions with the existing hypertext that is explainable in terms of the lattice. Therefore, we address the preservation of hypertext during publishing in Section 2.3 using the lattice to define the namespace of wiki names and accessibility of documents, and address the ecology of participant subsets and reversibility of publishing in Section 2.6 using the existing components of the hypertext interface.

2.3 Access Rules in Lattice Structured Wiki

Two key system features are determined by the goal to preserve hypertext under publishing. It determines that we retain the flat namespace of wiki, and it determines that the lattice defines the access rules of documents rather than their storage. A flat namespace with lattice structured access rules is a marked contrast from other wiki systems. Conventional wiki has a flat namespace with global access. This is like a file system directory without subdirectories, in which files are readable and writable by all. Other partitioned wikis use a hierarchy for the namespace with access rules for each subdirectory. This is like a file system with a tree of subdirectories, in which each subdirectory is readable and writable by a single group. The reasons for the exotic combination of flat namespace and lattice structured access rules becomes clear when we consider the effect of moving documents between participant subsets.

Consider Figure 2. Suppose there were two documents p and q sharing a single wiki name, and suppose these documents were accessible to participant subsets $\{a, b, c, d\}$ and $\{d, f, g\}$ respectively. Which document would participant d see? We now show that this name clash cannot be resolved without offense to parsimony. Instead we must accept that wiki names are drawn from a single flat namespace.

The obvious proposal, concerning the above name clash, would partition the wiki into clusters. A participant would have to visit the cluster to view its contents. This proposal is not parsimonious because it requires the participant comprehend and navigate the lattice structure; it does not preserve the hypertext because publishing a document removes it from a cluster; and in any case, it does not prevent name clashes at the time of publishing. In consequence, the meaning of wiki names must be uniform across all participant subsets, and cannot be a function of the participant subset currently hosting the document.

Having determined the nature of the namespace, it is now possible to describe the access rules. Each document is associated with a participant subset, and a participant can access the document if they are a member of this participant subset. The access rules do not regulate wiki name usage, but instead decide whether the corresponding document appears to exist. In other words, the lattice of participant subsets represents the user model for each participant, and link-adaption makes inaccessible documents appear as if they do not exist. For example, consider Figure 2 and documents p and q which are respectively associated with the participant subsets $\{a, b, c, d, e, f, g\}$ and $\{d, f, g\}$. Document p is accessible by all participants, so in any document the wiki name of p would appear as a functioning link when viewed by any participant. However, document q is not accessible by a so the wiki name of q would appear as a broken link when viewed by a. In this way the generalization of implicit page creation is made seamless.

In browsing the hypertext, participants do not move about the lattice. Hence, they do not need to comprehend the lattice structure. Thus, lattice structured wiki is indistinguishable from conventional wiki for the purpose of browsing. Communication is possible because the document subsets overlap, and the lattice identifies these overlaps and arranges this information into a diagrammatic structure. Therefore, an explanation of lattice structured scope that refers to the Hasse Diagram is a powerful tool in reasoning about communications.

2.4 Worked Example

We now demonstrate the collective authorship of a lattice structured wiki. The example is helpful in the formal definition of Section 2.5, and also in the description of procedures in Sections 2.6 and 2.7. Figure 3 depicts a sequence of states in the early life of a lattice structured wiki. We begin by describing how such diagrams represent the system, and then provide a possible interpretation for the sequence of states.

Each diagram in Figure 3 depicts a hypertext with its corresponding lattice. The depiction of hypertext derives from Figure 1, namely, each document is depicted by a labelled oblong, and each link by a curved arrow. However, document contents and anchor text are omitted. The depiction of the lattice derives from Figure 2, namely, the Hasse dia-

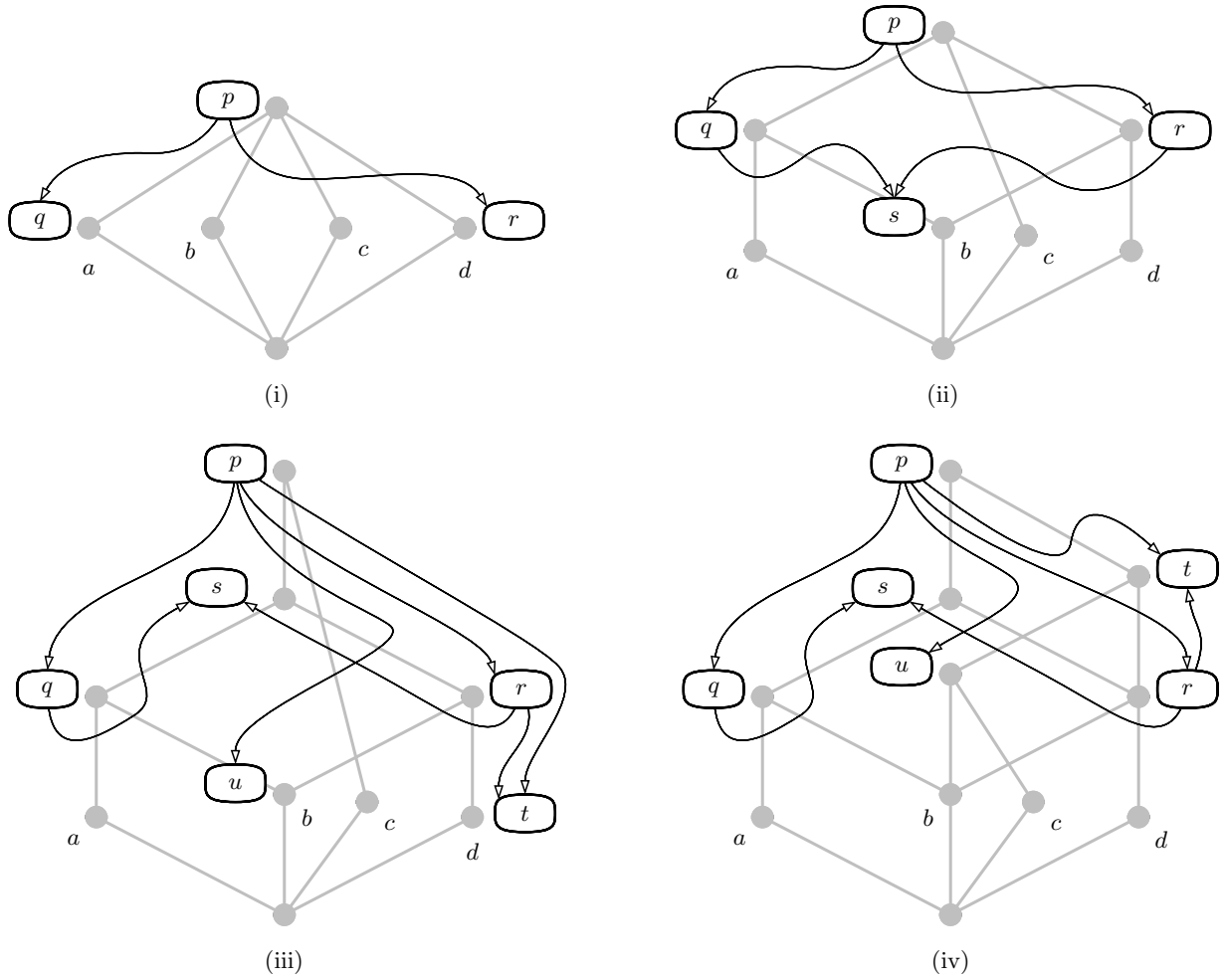

Figure 3: A sequence of states for participants $\{a, b, c, d\}$ and documents $\{p, q, r, s, t, u\}$: (i) links advertise q and r; (ii) q and r each admit b, and new links advertise s; (iii) s admits a and d, and new links advertises t and u; and (iv) t admits a, c, and d, and u admits c.

gram is drawn in grey, and the singletons $\{a\}$, $\{b\}$, $\{c\}$, and $\{d\}$ are labelled. However, the remaining subsets must be inferred. The hypertext is superimposed on the lattice so that a participant subset obtains the adjacent documents.

We determine the participant subset at a node n by downward traversal of the Hasse diagram. The participant subset includes x if and only if there exists a downward path from n to the node labelled x. This test includes the empty path. In diagram (iii): the second to highest node represents the participant subset $\{a, b, d\}$; the node labelled b represents the participant subset $\{b\}$; and the bottom node represents the empty participant subset. This is a corollary of the principle of inclusion described in Section 2.1.

We determine the documents accessible by a participant by upward traversal of the Hasse diagram. A participant x can access a document y if and only if there exists an upward path from the node labelled x to the node adjacent to y. This test includes the empty path. In diagram (iii): a can access the subset $\{p, q, s\}$; b can access the subset $\{p, q, r, s, u\}$; c can access the subset $\{p\}$; and d can access the subset $\{p, r, s, t\}$. Participants in a lattice structured wiki access overlapping subsets of the hypertext.

Three activities construct the hypertext and the lattice: link creation, document creation, and updates to the access rules. Section 1.2 describes link and document creation in the context of wiki, but updates to access rules are a feature of lattice structured wiki. A document is initially accessible only by its creator. In Figure 3(i), participants a and d have created documents q and r respectively. A document subsequently becomes accessible by others according to the system described in Section 2.6. Between (i) and (ii), b has been granted access q and also to r. This change to the access rules of q and r has expanded the participant subset lattice.

In order to better motivate the preceding analysis we provide a story line for the sequence of Figure 3 involving Ann, Bill, Cate and David as participants a, b, c, and d. At (i), Ann and David have written separate proposals for the group in documents q and r. By posting links from p they hope to attract participants. At (ii), Bill has accepted both proposals in the sense that he has requested and been granted access to documents q and r. Bill has also created a new document s, hoping to discuss issues common to each proposal. At (iii), Ann and David have joined Bill in using s

to discuss common issues. Bill and David have created documents u and t to track project administration and posted a link from p to draft Cate. At (iv), Cate has joined Bill at u to discuss Bill's administrative tasks, and Bill and Cate have both joined David at t to discuss administrative tasks related to proposal r.

2.5 Lattice Structured Wiki Defined

We next introduce a mathematical notation to analyze lattice structured wiki. We begin by denoting the set of all participants by Participants. The next step is to describe the hypertext as a graph data structure over the set of all documents.

A directed graph is conventionally rendered as a pair of disjoint sets of vertices and edges, and two total maps describing the composition of each edge. For example, see [5]. In our case, a vertex is a document and an edge is a link from a source document to a destination document. The bottom of the lattice obtains all unseen documents, so documents can safely be identified with wiki names. Thus, we denote the set of all documents by WikiNames and the set of all links by Links, and introduce the following total maps to call out: the source and destination of each link; a distinguished front page; and a distinguished back page for each participant.

$$\mathrm{src} : \mathsf{Links} \mapsto \mathsf{WikiNames} \tag{1}$$

$$\mathrm{dst} : \mathsf{Links} \mapsto \mathsf{WikiNames} \tag{2}$$

$$\mathrm{front} : \ \mapsto \mathsf{WikiNames} \tag{3}$$

$$\mathrm{back} : \mathsf{Participants} \mapsto \mathsf{WikiNames} \tag{4}$$

The maps $\mathrm{src}(l)$ and $\mathrm{dst}(l)$ respectively decompose a link into its source and destination documents. The map front() is a constant indicating the initial public document, while $\mathrm{back}(x)$ maps the participant x to an initial private document. Furthermore, we ensure a minimum of two participants by addition of guest and administrator accounts.

We reduce notational clutter by employing a shorthand for links. We say $\langle s, d\rangle$ is a link, and understand this as a reference to some $l \in \mathsf{Links}$ such that $\mathrm{src}(l) = s$ and $\mathrm{dst}(l) = d$. This is inaccurate because directional graphs, and specifically hypertext, may have multiple instances of the same source and destination pair.

The remainder of the structure is described by mappings between documents and participants. The lattice is induced by the document access rules represented by the following total map.

$$\mathrm{viewers} : \mathsf{WikiNames} \mapsto \mathfrak{P}(\mathsf{Participants}) \tag{5}$$

The map $\mathrm{viewers}(p)$ maps the document p to the subset of participants that can access p. In order to anchor the public and private documents, we supply the following invariants.

$$\mathrm{viewers}(\mathrm{front}()) = \mathsf{Participants} \tag{6}$$

$$\mathrm{viewers}(\mathrm{back}(x)) = \{x\} \tag{7}$$

Equation (6) ensures that all can access the front page, while (7) ensures that the back pages remain private to their respective owners.

Forsaking formal concept analysis, we construct the lattice directly in the terminology of order theory. Thus, we begin with the partial order $\langle P, \subseteq\rangle$ where

$$P = \{\mathrm{viewers}(p) \mid p \in \mathsf{WikiNames}\} \ . \tag{8}$$

This structure is the collection of participant subsets as enumerated by recovering the participants that can access each document. It is ordered by set inclusion. We complete this partial order to the minimal lattice $\mathrm{DM}(P)$ via the Dedekind-MacNeille completion. The requirement that $|\mathsf{Participants}| \geq 2$, and (7) together ensure the bottom element of $\mathrm{DM}(P)$ is the empty set because $\mathrm{DM}(P)$ is closed under intersection. Since the Dedekind-MacNeille completion is not the subject of this paper, we simply note the existence of algorithms to construct this structure [6].

Rather than resort to mathematical notation throughout, we offer the following terminology. We refer to an element of the lattice as a *cluster* because it obtains a cluster of documents. Then, since each cluster is an equivalence class of documents suffering access by the same participants, we describe the partial order in terms of how public a document or cluster is. Given two documents x and y, x is *less public* than y if and only if $\mathrm{viewers}(x) \subset \mathrm{viewers}(y)$, x is *more public* than y if and only if $\mathrm{viewers}(x) \supset \mathrm{viewers}(y)$, and x and y are *incomparable* if and only if $\mathrm{viewers}(x) \not\subseteq \mathrm{viewers}(y)$ and $\mathrm{viewers}(x) \not\supseteq \mathrm{viewers}(y)$. Finally, we say that a document belongs to a cluster with the obvious meaning, and that a link belongs to a cluster if it originates on a document belonging to that cluster.

Returning to Figure 3, we can now describe the action of the access rules in terms of the Hasse Diagram. Each document p is located on the lattice, because $\mathrm{viewers}(p) \in P$ by (8). Furthermore, each individual x is represented by a singleton $\{x\}$ covering the bottom element by (7). The set of documents accessible by x is the union of documents obtained by participant subsets reachable by upward paths from $\{x\}$. This is equivalent to the union of documents obtained by participant subsets containing x. In summary, documents are obtained by participant subsets that match the set of viewers, and these documents flow down the edges of the lattice to the individuals.

2.6 Page Offers and Requests

We now introduce page offers and requests as a generalization of the implicit page creation of wiki. These two concepts refer to the posting and following of links in order to publish documents. They are further classified into a taxonomy of offers and requests that is used to describe the behavior of the system in all cases. Together, page offers and requests describe a process for negotiating increased access to documents.

For a given participant, a link is a *page offer* if the source is accessible but the destination is not. We say the source *offers* the destination, since the participant is able to *request* the destination by following the link. A reasonable response to the request might be to grant the participant access to the destination. This constitutes a step in the incremental publishing of the destination.

Consider an example from Figure 3. Ann privately develops a new proposal on the document q. This document is an element of her personal cluster. In order to advocate the proposal, she selects the document p, because it is accessible by all participants including Bill, and then posts a link from p to q. Thus, she offers the document to Bill and other viewers of p. If Bill follows this new link, the result is a request from Bill for the document q. We will return to this example in discussing the types of offer and request.

This example motivates the mechanism of page offers and

requests, but the lattice allows further classification of inter cluster links. Ann's offer is a link between two clusters where the source is strictly more public than the destination. We refer to this as an *advertisement*. The other two cases are also of interest. When the destination is strictly more public than the source we say the link is an *endorsement*, and when the destination and source are incomparable we say the link is a *recruitment*. Given a link $\langle s, d\rangle$ we have the following.

$$\langle s, d\rangle \text{ is a } \textit{page advertisement} \iff \text{viewers}(s) \supset \text{viewers}(d) \tag{9}$$

$$\langle s, d\rangle \text{ is a } \textit{page recruitment} \iff \text{viewers}(s) \parallel \text{viewers}(d) \tag{10}$$

$$\langle s, d\rangle \text{ is a } \textit{page endorsement} \iff \text{viewers}(s) \subset \text{viewers}(d) \tag{11}$$

Given these classifications, we are able to use the lattice to recognize the situations to which the system must respond, and not just those of our idealized example.

2.6.1 Types of Page Advertisement and Recruitment

The relationship between the source, the destination, and the participant posting an offer classifies the act of posting an inter cluster link. The purpose of this classification is to describe the system responses to link creation. Certain links are likely to indicate name clashes, while others have a clear interpretation. The system can helpfully intercede in several easily recognized cases.

Advertisements are a preferred link type. The source defines a target for the publishing of the destination. Furthermore, they increase connectivity from public to private, which aids navigation. A *rightful advertisement* is made by a participant who can view the destination — one should only advertise what one holds. In the example above, Ann posted a rightful advertisement.

An advertisement that is not rightful is either accidental or creative. It is made by a participant that cannot view the destination. The common case is the *creative advertisement*, in which the document is not currently accessible by any participant. In case the document exists in the view of some other participant, the offer is charitably termed an *accidental advertisement*. Both types of advertisement are important.

Creative advertisements bring documents into existence. In terms of the lattice, they are links to members of the absurd cluster containing those documents inaccessible to all participants. Monitoring the posting of these links allows participants to be alerted to documents that require creation by an initial visit. In other words, these offers constitute deferred content. In the example above, Ann created the document q after posting a creative advertisement on one of the documents in her private cluster.

Accidental advertisements signal name clashes. Since the participant posting the link cannot view the destination, they have accidentally selected the same wiki name as the author of the destination. This affords an opportunity to negotiate wiki names. The possible negotiations reflect the possible relations between source and destination. In the example above, if Bill had decided to start the document q in public before Ann had offered her private content, the problem is resolved by waiting for Ann to offer her content. The lattice makes this inference possible, because Bill's accidental offer is a rightful offer if made by Ann.

Recruitments are a more problematic link type. This is because the source does not define a target for the publishing of the destination, so the target must be inferred by a lattice join. Since page recruitments connect incomparable clusters, they are only traversable by those in the intersection of the two participant subsets. Only *rightful recruitments* and *accidental recruitments* are possible, because all participant subsets are more public than the empty set.

As their name suggests, recruitments can be useful in merging two participant subsets. Consider a recruitment from s to d, where each participant requesting d is added to viewers(d). If all viewers of s traverse the link, the viewers of s and d will merge at d. With an advertisement, each request would have to be vetted, but a recruitment can restrict requests to an exact participant subset. This mechanism is so important that discrete messages indicating recruitment are a useful addition to the system.

2.6.2 Handling Page Requests

There are three important classes of response to a request. Firstly, the owner of the destination of the offer can reject the request outright. In case they accept the request, they can either grant just the request, or they can grant access to the join of the requesting participant and the destination's participant subset. The response determines whether the participant subset lattice is modified.

Granting just the request will modify the lattice of participant subsets if the resulting participant subset is not a member of the lattice. In Figure 3(iii), Bill has granted Ann and David access to s to create the participant subset comprising Ann, Bill, and David. This was the intention of posting the links from q and r to s. In essence, responding to an advertisement by granting access to requests selects a subset of the participant subset at the advertisement source.

Granting access to the join of the requesting participant and the destination's participant subset will never modify the lattice of participant subsets, because a lattice defines a join for each pair of elements. Consider the transition from (ii) to (iii) in Figure 3 and assume first Ann and then David request s. When Ann requests s, Bill could grant access to the join of $\{a\}$ and viewers(s), namely, the participant subset $\{a, b\}$. Then, when David requests s, Bill could grant access to the join of $\{d\}$ and viewers(s), namely, the participant subset $\{a, b, c, d\}$. This does not follow the story line, but in an established lattice of participant subsets it will often be the most suitable response, because join determines the smallest participant subset that contains both operands.

The formation of a lattice of participant subsets is marked by the granting of access to individuals. As the lattice begins to reflect the established groups, more of the requests are granted to the join.

2.7 Page Retraction

We briefly describe a scheme for page retraction. This is the process of reverting publishing. Again the scheme operates in terms of the hypertext. Namely, page retraction is triggered by link removal — when a link is removed a page retraction is computed. The retraction is limited by the existence of other links. In particular, if we can argue that a participant could use a remaining link to access the document, then we should apply the rule of least surprise and ensure they can still access the document.

The retraction is computed by taking the join of all partic-

ipant subsets that can access the document via the remaining links. Recall that the join is the smallest participant subset that contains every element in an operand, namely, the smallest participant subset that is a superset of the union of the operands. Performing this computation ensures that no participant able to access the document via a remaining link loses access.

We now enumerate these participant subsets by link type. For an advertisement, the participant subset at the source can access the page; for an endorsement, the participant subset at the source can access the page; and for a recruitment, the the meet of the participant subsets at the source and destination can access the page. If the join of these participant subsets is more public than the page, then we leave it in place, otherwise we retract it to the join. When there are no remaining links the join of the empty set of participant subsets is the bottom element of the lattice, namely, the page is recycled to the unused pages.

Unfortunately, this page retraction algorithm cannot determine cycles. In essence, it is the analogue of reference counting in garbage collecting in computer programming languages.

3. CONCLUSION

In this paper, we have proposed extensions to wiki that seek to address difficulties with certain collaboration scenarios. These extensions create a hypertext system that demonstrates a limited, but very useful, form of link-adaption. It differs from other adaptive hypermedia in that the link-adaption is a consequence of negotiation between participants. The user model active in this system is represented by a lattice of participant subsets, and a mapping from documents to elements of this lattice. This system describes access rules for the documents in the hypertext.

A strength of this model is that it contains a great number of interesting work-flow schemes as subclasses. For example, if every request is automatically granted to the join, then lattice structured wiki behaves as conventional wiki. In effect, this paper describes a taxonomy of access rules that includes the degenerate case of global access, as well as antichain, chain, tree, and lattice structured access rules. Seeding the lattice to install stereotypical publishing schemes is a useful approach to initiating collaboration.

We can summarize the many desirable properties that lattice structured wiki enjoys as a collaboration tool.

Elegant Theory The system is based on an elegant characterization of the implicit page creation of wiki. It does not clutter wiki with competing concepts, but places wiki in a complementary framework based on well understood mathematical concepts of order theory.

Deferred Overhead Participants are not required to master additional interface concepts until they seek behavior outside of conventional wiki. In particular, casual browsers experience system behavior that is indistinguishable from conventional wiki.

Improved Audience Targeting Each participant subset is incrementally constructed, and the result is independent of the order in which the audience for a document is constructed. Thus, an audience can be refined under new information, avoiding many problems of poor recipient selection in email-based collaboration.

Diagrammatic Representation The lattice of participant subsets is depicted by a diagram describing their relationships. This structure is associated with well developed theories of diagrammatic reasoning on the taxonomy [11].

Expressive Group Structures Since lattices are a broader class of hierarchical structures than trees, the collection of access rules does not suffer the arbitrary encodings found in tree structured taxonomies.

Flexible Notification The system exposes a broad selection of notification triggers based on the posting of page advertisements and page requests.

Computational Efficiency The web application is able to exploit efficient lattice based inferences to compute access rules for documents.

4. FUTURE WORK AND EVALUATION

The model described in this paper is currently being evaluated as part of Australian Research Council funded research into ontology-based collaboration in design. This work includes mechanisms for specifying automated responses to link traversal, and the selection of pages suitable for hosting page advertisements.

5. ACKNOWLEDGEMENTS

This paper benefitted greatly from the rolling review process. In particular, the author wishes to thank reviewers in the first round for their detailed comments. This paper would not be possible without Australian Research Council funding of ontology-based collaboration in design.

6. REFERENCES

[1] Richard J. Cole. *The Management and Visualisation of Document Collections Using Formal Concept Analysis*. PhD thesis, Griffith University, Australia, December 2000.

[2] Ward Cunningham. Wiki design principles. *Portland Pattern Repository*, 27 November 2003.

[3] B. A. Davey and H. A. Priestley. *Introduction to Lattices and Order*. Cambridge University Press, 1990.

[4] Paul De Bra, Geert-Jan Houben, and Hongjing Wu. AHAM: A dexter-based reference model for adaptive hypermedia. In *Proceedings of the tenth ACM conference on hypertext and hypermedia*, pages 147–156, Darmstadt, Germany, 1999. ACM Press.

[5] Reinhard Diestel. *Graph Theory*, volume 173 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, second edition, 2000.

[6] Bernhard Ganter and Sergei O. Kuznetsov. Stepwise construction of the Dedekind-MacNeille completion. In *International Conference on Conceptual Structures*, pages 295–302, 1998.

[7] Roy Rada. Hypertext writing and document reuse: The role of a semantic net. *Electronic Publishing*, 3(3):125–140, August 1990.

[8] Francisco J. Ricardo. Hypertext and knowledge management. In *Proceedings of the twelfth ACM conference on Hypertext and Hypermedia*, pages 217–226, Århus, Denmark, 2001. ACM Press.

[9] Peter Thoeny. Twiki - a web based collaboration platform, 23 November 2003.

[10] Randall Trigg. *A Network-Based Approach to Text Handling for the Online Scientific Community*. PhD thesis, Department of Computer Science, University of Maryland, November 1983.

[11] Rudolf Wille. Concept lattices and conceptual knowledge systems. *Computers Math. Applic.*, 23(6–9):493–515, 1992.