

Enhancing documents with annotations and machine-readable structured information using Notate

Robert Cannon and Fred Howell

Textensor Limited, www.textensor.com

4th March 2007

<http://www.textensor.com/enhancing-documents-2007.html>

Summary

Textensor Limited is developing tools for improving the communication and exploitation of text based information. Our main product, Notate, is a web based system that enables authors and readers to layer structured annotations on top of documents so that the resulting combination can be reliably processed automatically while maintaining the integrity of the original source and the provenance of all annotations.

The system has a wide variety of applications including attaching sticky notes and discussions to web pages, sharing documents and notes within a small group, on-line document review and sophisticated data curation tasks. It aims to bring the authoring of semantically rich structures within the capabilities of normal users, making it dramatically easier to produce well-structured content and opening up possibilities for further automated processes such as creating indexes to the research literature and curating more high-quality information into databases. The initial requirements and example applications are taken from the needs of the biomedical research community, but the core technology is not domain specific and has similar applications in other fields that deal with large volumes of documents containing complex and interlinked information.

This white paper describes the origin of the underlying ideas for Notate in hypertext and web research communities, and places our work in the context of other recent advances in web technologies such as semantic wikis and 'Web 2.0'.

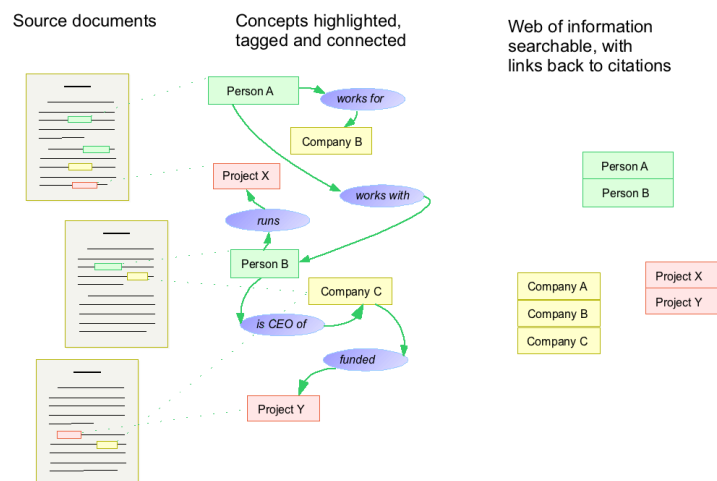


Figure 1. Notate starts with documents. Terms of interest are highlighted and notes and 'tags' can be attached - here different people, projects and companies are annotated. Connections between terms can also be expressed (such as who works for which project), and the resulting web of structured information can be efficiently sorted and viewed.

Further information on the latest version of Notate, as well as a sandbox for trying out the core features, can be found on www.texttensor.com.

Origins

With a growing number of researchers worldwide, and experimental techniques generating ever increasing volumes of data the time has long passed when written texts alone were an adequate means of communicating and storing scientific information. Carefully constructed and meticulously curated databases meet a clear need for certain types of data, but such systems are not suited to the deeply heterogeneous material that emerges from much modern research. This leads to a pressing need for better ways to present, store and disseminate scientific content.

This concern is not, however, only a recent development. In 1945, Vannevar Bush wrote (Bush 1945)¹:

"There is a growing mountain of research. But there is increased evidence that we are being bogged down today as specialization extends. The investigator is staggered by the findings and conclusions of thousands of other workers' conclusions which he cannot find time to grasp, much less to remember, as they appear."

and continued:

"Professionally our methods of transmitting and reviewing the results of research are generations old and by now are totally inadequate for their purpose. The difficulty seems to be, not so much that we publish unduly in view of the extent and variety of present day interests, but rather that publication has been extended far beyond our present ability to make real use of the record."

He goes on to describe the "memex", a mechanical system for storing documents and data that supports associative indexing and notes attached to particular parts of the text, and which can be consulted rapidly and flexibly as a supplement to the user's memory. The theme was taken up by Doug Engelbart in his 1962 paper (Engelbart 1962)² Augmenting Human Intellect: A Conceptual Framework, now backed up by electronic computers and presaging much of his later work on hypermedia and groupware. Although the technology has changed, the problems described in these visionary works are now more pressing than ever. And the solutions they proposed, based around annotation, indexing and structured two-way links are equally pertinent.

The key objectives for a system of presentation and dissemination are that

- the system should contain complete and correct documents as its starting point
- it should be efficient to query and locate material when needed and
- it should allow incremental extension with new metadata (tags, connections, notes etc) that improves its utility.

The emphasis on documents as the starting point, rather than more formal systems or databases arises from the flexibility of text and the skill sets of authors. Text is the only form in which most authors are able to create a comprehensive and accurate record of their work. Such documents provide the basis for whatever information is migrated into more structured formal systems or databases.

The challenge, therefore, is not to remove or replace plain text content, but to develop technology routes whereby the text can be extended or enhanced with machine-readable annotations such that objectives of locatability, comprehensive search and advanced querying can be met more effectively. The original text remains the definitive source and can be consulted for clarification or verification.

The first question for such a technology is who should create the annotation layer? Can it be fully automated, with computer programs crafted to read and understand natural language text and enter rele-

vant summary statements into databases? Or should it be left to teams of highly skilled curators such as the team behind CAS³, the American Chemical Society's extensive database populated from journal abstracts? Or should it be possible for anyone who is able to write or understand the text in question to annotate it with more structured links?

The fully automated case is still a research area (text mining), with some successes in restricted domains such as spotting protein names in text, but the resulting annotations tend to include a large number of errors. Using teams of highly specialised curators to read documents and populate databases is an expensive activity which can be justified for certain types of content for which there is strong commercial benefit, but these methods may not deliver the most cost effective solution overall. This leaves the third option: allowing everyone who can write or understand the original content to push it further and add more structure so that the key content is more accessible to other readers and can be processed reliably and more readily by automated systems.

As has been remarked by Anita de Ward⁴ on the difficulties of text mining "if you want to mine the content, then why did you bury it in the first place?" At present, the reason information is 'buried' in text is not because authors want to hide it, but because they have no other option. Given the time and effort involved in performing the work and presenting it as a paper, if they had had the opportunity of also delivering a more machine readable version that would be readily accessible to automated systems, then perhaps they would have done so. There are various possible motivations, but one of the strongest is likely to be visibility. For example, in the case of corporate websites, whenever Google changes its ranking algorithms, the Search Engine Optimisation (SEO) community sets about modifying the presentation of its content in order to get high Google page rankings. The attention paid to journal impact factors and citation counts suggests there is a similar drive within research, although there is no comparable mechanism, yet, for improving the impact and findability of one's work. Within private organisations there are also many other ways to reward behaviour that enables rapid and efficient access to information.

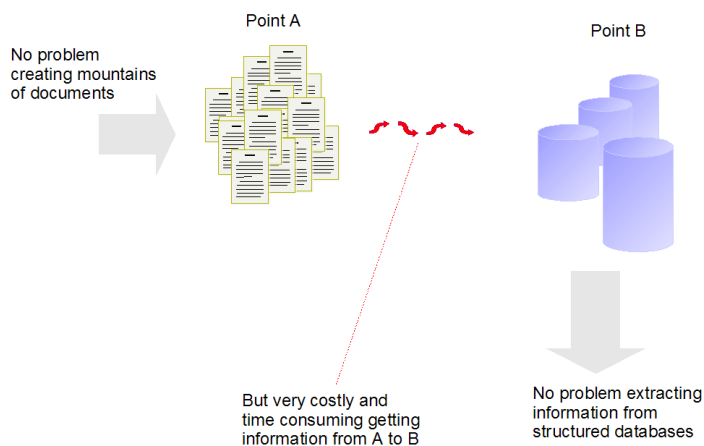


Figure 2. People find it natural to write documents, but information is much easier to search and sift once it is in more structured forms such as databases. Note that lets people add progressively more structure and annotations to their documents so they can get the benefits of databases without the pain.

In summary, the features that Textensor has identified for its technologies are:

- Start with definitive primary content in plain text or standard document formats (MSWord, PDF and HTML). We do not expect people to re-author or modify their documents into more structured forms (such as XML with a particular schemas).

- Operations should be carried out through a user-friendly system that is accessible to individuals with normal computer skills, including any author of the original document.
- It should allow incremental and ongoing enhancement of text with tags, notes and links.
- Annotations that are not immediately picked up by automated processing systems are still useful: users can always read them directly, and mining technology is likely to improve. Authors should not be prevented from entering precisely the content they think is needed simply because the system does not know how to use it yet.

A good starting point for enhancing text with structures and cross references is the techniques already used within traditional publishing. Most text books have indexes and these list the most useful places in the text (typically by page, but sometimes by section or paragraph) where a particular item is discussed. They may also introduce synonyms not mentioned in the text and cross-references to related terms. Unfortunately, indexing technical books is a skilled occupation that can rarely be accomplished by the author alone. Can technology make it sufficiently simple and rewarding for large numbers of authors and readers to take on this role themselves? The growth of social bookmarking sites such as Connotea⁵ and del.icio.us⁶ show the feasibility of community-wide page-level tagging, and how a 'folksonomy' of useful index terms can develop according to what people find useful. Textensor's goal is to maintain the same level of usability while pushing the granularity of tags and annotations down to individual words or phrases in documents, and adding the capability to relate fine-grained markers to one-another so as to generate semantically rich machine readable structures.

Structured Data

Structured data (as contrasted with plain text) is easily processed by computers. Examples include the tables of information in databases, where each entry of a 'Contacts' table might have information slotted into 'name', 'address' and 'telephone number' fields, or a list of 'References' would have author, journal, title, volume, year fields. Structured information systems typically also allow cross referencing to other types of information, such as 'Company' details, or hyperlinks. Automatically extracting, sorting and sifting slices of information once it is available in structured form is very efficient, with no need for manual retyping or searching, but it is hard for programs to make sense of plain text (in document formats such as HTML or PDF) beyond building basic search engine indexes of the words used.

Setting up databases for holding structured data, or designing schemas and creating valid XML documents is currently the preserve of programmers and database designers. Most people are happy to write documents, and perhaps populate databases designed for restricted types of information (e.g. reference management tools, contacts databases). Notate aims to provide a way to connect the complex information embedded in documents to new structured data so it can be stored, sifted and communicated more effectively.

Textensor's core platform is *Notate 2* which comprises a web-browser based rich client (developed using Javascript / DHTML / AJAX technologies) for working on documents via a web browser; and a web server for storing and processing the index of annotations. It combines document management with annotation processing and implements the core objectives outlined above. The following sections describe the main functionality of Notate 2, then explore a number of typical applications, and finally discuss how it relates to a wide range of other Web 2.0 and semantic web technologies.

Notate 2 in detail

Notate operates in a web browser without requiring any software installation. Upon registering, the user is given a 'Snapshot' button to add to their browser. This takes a copy of the web page they are currently viewing, transfers it to their private space on the Notate server, and then shows the annotation notebook for adding notes. The server also allows Word documents and PDF files to be uploaded and annotated in the same way.

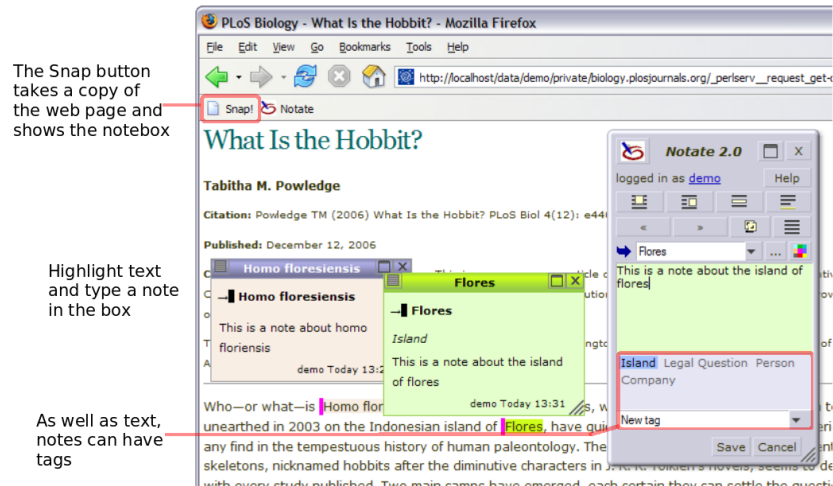


Figure 3. After snapshotting a web page, the user can highlight text and write notes using the notebook. Notes can also have semantic 'tags' attached - in this example 'Flores' is tagged as an 'Island'.

Notes and tags

When the user highlights a word or phrase on an activated web page, the annotation box is displayed. This is the entry point to all the annotation capabilities. The simplest operation is to type a note and save it. This attaches a 'sticky note' to the page that sits just above the highlighted text. There is also an area on the annotation box for adding tags to the note. For example, in a simple business context, appropriate tags could be 'Person', 'Company', 'Product', or 'Competitor'. There is no restriction to which tags are used; if none of the previously used ones is suitable, users can simply type their own. However, it is quicker to choose an existing one from the menu of recently used tags. The benefit of adding such tags is that from the index on the server users can rapidly locate, say, lists of all People, Companies, Products or Competitors mentioned in any document and see the precise context on the page in which they are mentioned.

Adding user created 'tags' or keyword labels is a very flexible way to categorise and index concepts without having to select from a restricted set of pre-defined labels. Shirky's 'Ontology is Overrated: Categories, Links, and Tags'⁷ provides an interesting comparison of rigid taxonomies with user-created 'folksonomies' such as are used to organise photographs in flickr⁸. Social bookmarking sites like del.icio.us⁹, Connotea⁹ and citeulike⁹ allow anyone who posts a bookmark to also add their own tags describing the page. Tagging will be brought to the masses with its incorporation into the Windows Vista operating system (e.g. see Jon Udell's screencast of tags in Windows Vista photo gallery¹⁰ for an online demonstration).

Notate takes the idea of tagging to a new, finer grained level, where tags can be added to individual phrases and words in a document, rather than to the entire document. Aggregation on the server exploits both the user-entered data (tags and notes) and other available information (who the user is, what page is being annotated, the context of the highlighted term, the date) to provide a wide range of searching and sorting functions to help access tagged or annotated content.



Figure 4. Notes are stored in an index page, where they can be searched, sorted and filtered by tags.

Privacy, security and collaboration

A central theme of Notate is collaboration and communication within a trusted community. Each user is given a 'private' space on registration where they can write notes that are only visible to themselves. They can also create new shared group spaces and have email invitations generated that authorize others to join. By default, the notes and pages within a group are only visible and editable by group members, but it is also possible to make a group 'public' so the notes are published on the web.

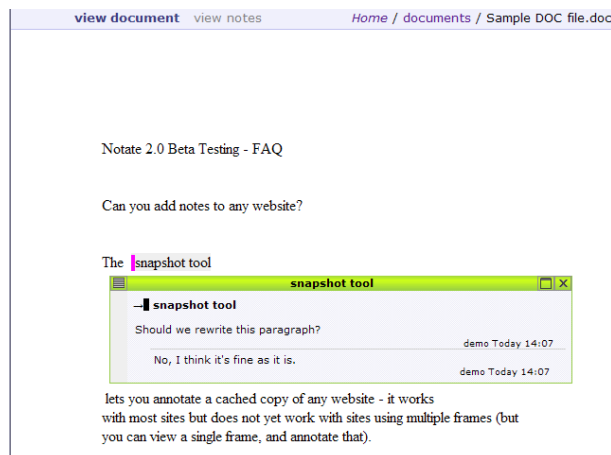


Figure 5. Members of a group can reply to each other's notes, making Notate useful for discussing drafts of documents online. This screenshot shows annotations attached to an uploaded Word document. It is also possible to issue invitations authorizing other individuals to view and annotate just one particular document.

Within a group, replies can be added to a note by any member, making it simple to base a discussion around a specific section of a text. This is particularly useful for document review and collaborative authoring of reports: by pooling the comments on a draft in one place on the web rather than emailing different versions back and forth, the process of integrating versions becomes much simpler, and the commenting process is more interactive.

Linking notes to new pages and ontology entries

By default, when a note is attached to a particular highlighted phrase in a document the selected text becomes the subject of the note. For example a user could highlight the word 'Textensor' and tag it as a Company. The subject of the note can, however, be changed from the default plain text, to a hyperlink to a web page or database entry that describes or defines the subject of the note. For example, the text 'Textensor' could be linked to the website *www.textensor.com*. For biomedical texts, many public databases are available with detailed information about particular proteins or genes, so it can be useful, for example, to link an occurrence of a particular protein name in the text to its unique Swissprot database ID. Notate allows links to database IDs to be entered in a shorthand form such as "SP:12345" and then linked out to the full URL of the target database entry.

In many cases, however, there will be no suitable web page describing the highlighted concept. The 'subject' chooser in the notebbox also supports linking to a new page to be created on the Notate server. Such pages are editable using a web browser with a wiki-style text editor. The user can also attach documents, images, and other data files to the system, making it a convenient shared workspace for groups. These pages on the server also list all the *incoming* links showing all the notes and documents which mention that page, providing a way to pool references to a given term or concept.

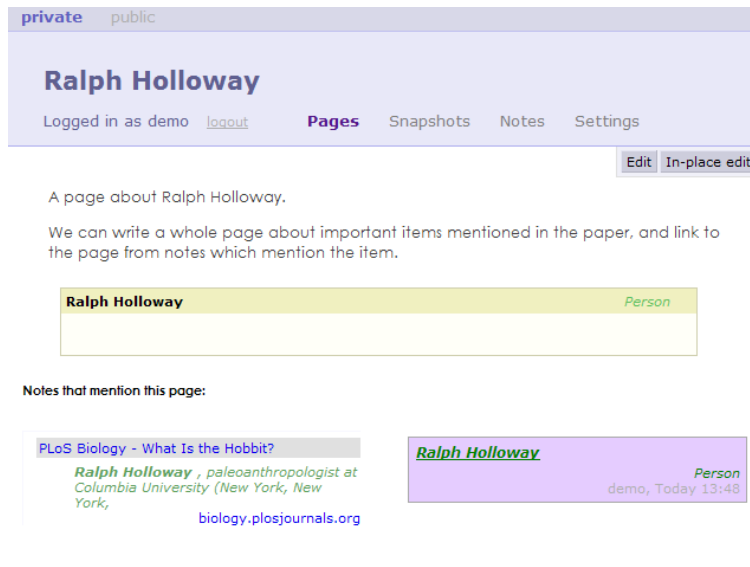


Figure 6. Notes can be linked to URLs or to new pages in the Notate content management system. In this example, a note about a person has been linked to a new page, which can be edited online using a web browser. The page also displays all the notes which link to it, along with their context, making this a useful facility for pooling references from many papers to a particular concept or item.

Linking a note to a web page is similar to adding a hyperlink in a web page or wiki. One significant difference between Notate and normal web pages, however, is that adding the link does not change the original text (it is not embedded in the page) but rather adds a layer on top of the text. This is important because it preserves the integrity of the original document while gaining the benefit of links. Furthermore, the annotations are all signed and dated so the full provenance of any entry in the system is available.

Connections and links between concepts mentioned in a text

In addition to adding notes to pages and connecting them to external resources or new pages, it is also possible to add connections between notes. For example, the user could start by highlighting the names of two people mentioned in text - 'Joe Bloggs' and 'Jane Doe', and then add the additional information that Joe Bloggs *works for* Jane Doe, a connection between the two notes. When viewing the notes about either person in the index, the connected notes will also be displayed.

As with the tags, the user is free to define their own connections. Autocomplete functions encourage the re-use of existing terms to help prevent the same concept being entered multiple times with slightly different spelling. In a business context, useful connections could be 'works for', 'works with', 'has email' or simply 'knows'. Biological relations might include 'reacts with', 'has valency', 'inhibits' or 'causes', but the list is practically endless.

In version 2.1 of Notate, such connections are of the form *subject verb object*, also known as 'triples'. More complex connections (e.g. a reaction where A, B and C combine to make D) can be expressed by creating a note about the reaction itself, and adding multiple statements (e.g. 'Reaction X has reagent A, has reagent B, has product D'). This use of triples is similar to RDF¹¹, but Notate stores additional information with each connection so its provenance can be easily established (who said it, when, and in the context of which document). The subject of the connection also shows the exact place in the supporting text from which it arises.

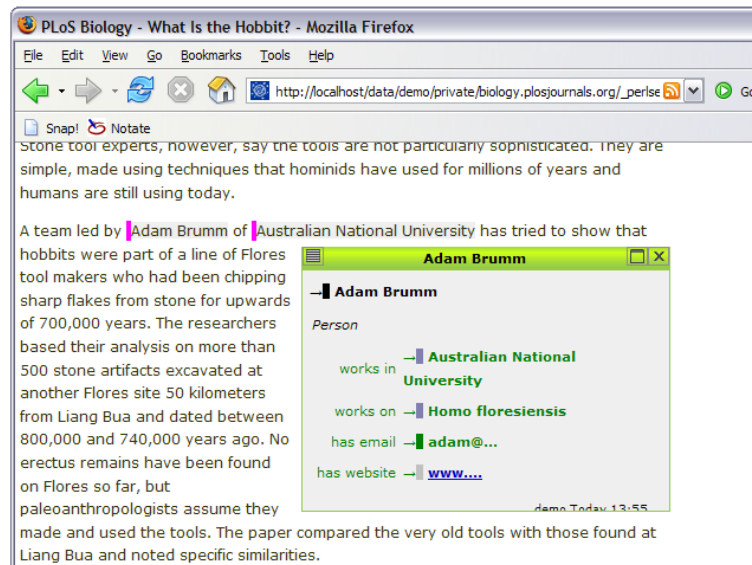


Figure 7. The user interface allows connections between highlighted terms to be expressed; this example relates a person to the project they work on and provides some structured properties. Notate is unusual in allowing *any* verb phrase to be used for the connection - in the same way that the user is free to add any text for the tags.

Applications

The core technology for creating, editing, storing and linking fine-grained annotations on text documents can be used in a wide range of scenarios from individual note-taking to large scale curation efforts. Not all uses require all the features previously mentioned. Here we present five usage scenarios covering a range of scales and degrees of sophistication.

1: An online version of hand written notes on printouts

Underlining important sections of text on a printed document, or making notes in the margin are natural and effective means of recording one's thoughts. But they are typically short term activities: if the annotations are not acted on promptly the context and meaning may be lost. Notate provides a means to create annotations that are usable and meaningful in the longer term, that can be easily shared with colleagues, and that are accessible via other routes in addition to the document itself.

A key difference between electronic and hand-written annotation is that the latter are tied to a particular physical copy of the document. In the Notate version, the document can be viewed with or without annotations and the same annotations can be accessed from multiple places and by multiple users. Annotations can be viewed on the document, in the index, in lists of recent annotations and via any tags that they may include. These added rewards for the effort of making annotations promote the creation of better notes which in turn leads to more meaningful and durable content.

2. On-line review of documents

The 'reply' feature on annotations makes Notate a convenient tool for on-line review and discussion of a manuscript. The author uploads the document onto their space on the Notate server, then invites other users to comment on the page using their web browser. Author and reviewers can reply to each other's comments, creating threaded dialogs about different parts of the document.

The privacy model, where notes are only visible within selected groups, is designed to avoid two recurrent problems with other annotation systems. Open comments frequently suffer from the 'shyness problem' from having a large potential readership known to the author (see the nature peer review trial¹², in which researchers were reluctant to make substantive comments public), or the 'graffiti problem' where unmoderated notes lead to spam and large quantities of nonsense notes (e.g. see the gibeo.net¹³ experiment in public notes on websites). See also PlosOne annotations¹⁴ for an early-stage system for adding public comments to papers. It remains to be seen how the community will adapt to and exploit such capabilities. Notate allows for public comments, but we expect that initially people will choose to restrict their comments to a trusted group.

3. Authoring structured metadata for data files

Setting up databases of experimental data requires significant programming skills, and may take more effort and expense than is available in many cases. However, data files are of little use on their own without the associated metadata describing experimental conditions, materials, protocols etc. Good records are also essential for keeping track of an individual's research activity. Notate provides a straightforward path to build on essential record keeping and construct comprehensive shared repositories of experimental data, without requiring database programming skills.

The process starts with simply attaching data files to a folder in the Notate content management system. This uses a web interface, and as with email attachments, the author can type a text description of the files at the same time. If the author chooses, they can also annotate this text description using the Notate highlighting interface, and link it to other notes and documents in the system, expressing relationships

such as the protocols used, the papers the methods were described in, the analyses and results obtained. These relationships are bidirectional links, so backwards connections will also be added from the pages describing results, analyses, protocols etc. In this way it is possible to build up a shared resource of data files, metadata and connections. The creation of such a comprehensive record helps ensure that data will be usable by the author and others in the long term, and will help reduce the risk of serious data loss with changes of personnel.

4. Curation

Curating source texts for populating high-quality databases requires significant domain expertise and is often undertaken by geographically distributed teams of experts who also work on many other activities. It is therefore important that the tools they use should be easy to learn and convenient to use from a variety of platforms and locations. Operating via a web browser, Notate is an ideal platform for use by such curators. A typical scenario involves tags, local pages and target databases being configured in advance and in accordance with the curation manual. Texts are then allocated to individual curators, possibly in sequence through different processing stages. All annotations are stored on the server and can be exported in an XHTML microformat for further processing with other systems.

5. Issue management and workflow collaboration

The focus in the previous applications has been on attaching notes to existing text or writing new pages. Notate also supports the creation of standalone notes on the server in exactly the same way as notes are attached to pages except that there is no original context for the note. With appropriate tags, these form a convenient, light-weight collaborative issue-tracking and task management system. Combinations of tags such as 'Bug', 'Feature', 'Version 3', 'TODO' etc, can be assigned to notes as they are entered in the system. As items are worked on, replies can be added to record the history of an issue and old tags can be removed or new tags added. The note display shows the entire tag history - when each tag was added or removed - so again, as with all Notate applications, there is a full record of who did what, and when.

Related technologies

Issues around tagging, indexing, annotation and linking have been extensively discussed from the days of Vannevar Bush (1945) and Doug Engelbart (1962) onwards. There are a great many systems on the web implementing various aspects of these ideas ranging from proof of concept demonstrations to full commercial systems. Here we present examples of other technologies for tagging, sticky notes, content editing and richer semantic structures, and show how each case relates to the purpose and capabilities of Notate.

Tagging

Tagging and annotation on the level of a whole document or file is widespread in social bookmarking tools such as Connotea¹⁵, del.icio.us¹⁵, and blinklist¹⁵, and in image sharing systems such as flickr¹⁶ or photobucket¹⁶. Notate has a finer grained structure for its annotations which target a particular word or phrase within the document rather than the document as a whole. This is essential if the resulting annotations are to be used to work with the actual content of the document. The popularity of bookmarking sites demonstrates the genuine appeal and approachability of user-defined tagging (in contrast to the use of strict ontologies discussed below, which are for more specialist use).

Notes on documents and websites

Karger et al. (2003)¹⁷ note that 'Passing annotated documents between colleagues is a highly effective way to exchange ideas and engage in collaboration' and present an experimental implementation of computer based annotation in the Haystack¹⁸ project. However, no system for web based sticky notes has gained widespread use yet. The Annotea project¹⁹ has demonstrated a browser plug-in interface to link to complex RDF content, but it is not as simple to use as the popular link bookmarking and tagging sites.

Many document editing and viewing applications include annotation features. For example, notes can be embedded in Microsoft Word documents or Adobe PDF files. Recently these have been extended to improve features for document review, typically by emailing documents with embedded notes back and forth, and merging comments, or using an electronic white board (Acrobat Connect). In these cases, notes are used primarily for document review. Notes added using Notate can also be used for document review via a web browser, but notes in Notate are also the entry point to attaching tags and links from markers in the text out to a knowledge base and index.

Recent interest in Web 2.0 and interactive websites has led to new tools such as mystickies²⁰ and posticky²¹ which provide bookmarklets²² or browser plug-ins for attaching post-it style notes to third party websites. However, the notes are attached to a fixed X,Y location on the page rather than to a section of text, and they lack the facility for linking text to new content. Other online 'sticky notes' sites that do not allow notes to be attached to any web site include stikkit²³ and Thinkature²⁴ which includes notes as part of an electronic whiteboard for collaboration.

Layered annotations and provenance

The idea of separating links from text (as Notate does, and HTML does not) has been around since the earliest days of hypertext research and development. Early examples include Ted Nelson's original coining of 'hypertext' in the 1960s with hyperland²⁵, and Project Xanadu (www.xanadu.com)²⁶ which complains that the web does not live up to his original vision:-

The World Wide Web (another imitation of paper) trivializes our original hypertext model with one-way ever-breaking links.

Likewise Wendy Hall's MICROCOSM history²⁷ notes several advantages if links can be layered on top of documents rather than embedded within them:

The separation of data and link structure permits the reuse of existing data without affecting it in anyway. This creates the potential for any user of the system to be a hypermedia author.

Vannevar Bush also proposed that notes and links should be layered on existing (read-only) text in his hypothetical 'memex' machine for reading and annotating papers: (Bush 1945)²⁸:

'He [the user] can add marginal notes and comments [...] just as though he had the physical page before him. [...] any item may be caused at will to select immediately and automatically another. This is the essential feature of the memex. The process of tying two items together is the important thing'.

Keeping track of the provenance of statements is a major challenge for scientific databases (e.g. Buneman et al 06²⁸ which discusses how to keep track of the history, origin and trustability of each database entry, particularly if it has been imported from another database or transcribed from somewhere within a journal article). The issue of citing particular phrases in web documents has also been raised previously, with paragraph-level identifiers added by the purplesurple²⁹ tool, and the xpath³⁰ notation developed for addressing particular fragments of an XML document. However, the fine-grained addressing facility

provided by Notate is still unusual for citations, which mostly refer to the entire article rather than a particular phrase in the document.

Wikis

Wikis and blogs provide convenient interfaces for authoring new content, including adding links between documents. Wikis specifically make it very easy to create and link to a new document thereby allowing the rapid development of interconnected networks. The most widely used wiki, wikipedia³¹, allows editing by anyone, and all information is public. In a scientific context, the openwetware³² project allows for public uploading and sharing of data and protocols. A great many organizations also use Wikis internally as one component of their record keeping and collaboration systems.

Notate also allows easy on-line editing of pages, but unlike wikis, these pages are organised in a hierarchy of folders. This is closer to the approach adopted by content management systems and some 'hierarchical wikis' or 'structured wikis'. The aim is to support familiar structures for organizing pages and to avoid the 'spaghetti' of pages that wikis can turn into.

However, the most important distinction between Notate and wikis is that Notate allows the user to add links from *particular phrases in external source pages*, whereas wikis only allow links to be embedded within pages within the wiki. The user does not need to be able to edit a document in order to attach links from its text. This is a great advantage, as it allows the entire world wide web to be used as source material.

Experimental *semantic wikis*, which add a degree of structured content to plain-text wikis, are now starting to appear. The semantic mediawiki³³ extension of the wikipedia server allows embedding of typed links and properties within a wiki page. Ikewiki³⁴ allows layering connections from the current wiki page to terms on the same page, with connection types restricted to a pre-defined set of RDF terms. Knewco³⁵ combines a semantic wiki with automated text mining to try and guess what the document is about. These systems share a number of features with Notate, although the focus in most cases is on use by IT experts or skilled curators, rather than by authors or readers in general.

Text mining

Text mining has seen substantial investment both from the research community and from commercial interests that develop curated databases. Although there are promising leads, the problem of automated content extraction from written text is notoriously difficult and such systems are generally used as an aid for curators rather than as a direct source of trusted database content. Notate can be used in conjunction with a text mining algorithm in a variety of ways. Manual annotation and tagging can be used for marking up source material prior to text mining or for generating training data that the mining algorithms ought to be able to reproduce. After mining, auto-generated annotations produced by the mining algorithm can be manually reviewed and confirmed, edited or deleted as appropriate using Notate's user friendly web based interface.

Ontologies and the Semantic Web

Ontologies and controlled vocabularies provide ways to resolve terms in text to unambiguous definitions. They are an integral part of many semantically rich systems. At the entry level, however, for authors writing text, a requirement to use ontologies can seem daunting and unpalatable. Notate provides a gentler way to introduce ontologies as a second tier of structure on top of the original text. For example an annotation can be used to link a term to an external definition. And, significantly, if the author is uncertain or the fit is not perfect, this information can be included in the text of the note, thereby maintaining the integrity of the system. Notate also enables users to construct their own ontologies using the "link to

new page about..." function which lets the user accumulate references to a particular entity or idea, and relations between concepts can also be expressed using the links/connections features. When viewing references to a particular ontology entry, the system lists all mentions of that term, along with the context, giving a sophisticated index.

The original vision for a 'semantic web' of RDF data objects and statements (Berners-Lee et al. 2001)³⁶ has not yet been realised; e.g. Shadbolt, Hall, Berners-Lee (2006)³⁷ state that the 'Semantic Web isn't yet with us on any scale', and perhaps give a clue as to the reason why with the figure legend describing a section of RDF

[... RDF is] 'actually quite clumsy syntactically, and its lack of transparency and readability might have been a factor inhibiting rapid adoption of RDF'

The difficulty arises from the effort needed to start using machine readable formats and the lack of any immediate reward for doing so. The areas where significant amounts of information are now available in structured forms include simple applications such as the FOAF project³⁸ (Friend Of A Friend) network, and 'RSS' news feeds. Most other semantic web projects have been demonstrations built by and for computer and information scientists familiar with ontologies, RDF, description logics, and numerous other fields. The Haystack project³⁹ is a good example, where most of the information has been extracted from existing sources by 'screen scraping'. Existing databases are another source of RDF and are technically straightforward to export in this form since all the work in structuring the data has already been done. But for the same reason, the results are perhaps of limited use, their main benefit being the possibility of searching across several datasets in the same way. Notate aims to make it substantially easier and more rewarding for users to create and exploit their own new structured content. All the semantic information they enter can be exported as XML which is straightforward to convert into other structured data formats.

Ontologies such as the Gene Ontology⁴⁰ have proved very useful for expert curators trying to impose order on biological databases. But part of the use is in maintaining strict control on how different terms and verbs are used (such as 'is part of', 'contains', 'reacts with'), and creating and using ontologies is generally restricted to such bioinformatics experts. The most well-established tool for editing ontologies, Protege⁴¹, is targeted at developers and ontology or linguistics researchers rather than at mainstream users. A Protege plugin, Knowtator⁴², adds facilities for highlighting text in articles and linking to existing RDF ontologies in a similar way to Notate, but it is aimed mainly at natural language processing researchers with a special interest in ontologies.

An interesting recent project to address the effort / reward balance for authors of websites familiar with HTML and Javascript coding is Exhibit⁴³, a Javascript library for providing a more dynamic web page displaying lists of items (e.g. publications). Data is entered in JSON⁴⁴ (JavaScript Object Notation), a programming language definition of arrays of data which is simpler to parse and type than RDF or XML. The javascript on the site then provides a range of views of the data and lets the user interact with it to change the display style and sort or search on particular attributes. This style of reward is similar to the Notate server which gives a flexible way of mining annotations once they have been made, and also uses JSON techniques to present a fast and responsive user interface.

In contrast with all these projects, the focus with Notate is not on completing the transition from text to structured data in one step (usually such a complicated step that only programmers can perform the conversion), but on lowering the barriers to entry so that the process is accessible to a much wider range of users. It provides immediate rewards to doing so, in the form of well ordered indexes and the ability to collaborate with documents and annotations in new ways. All annotations produced using Notate are also available in structured data file formats for further processing, enabling Notate to be used as an initial information extraction phase prior to curating databases.

Relational databases

Relational database systems are the standard solution for storing highly structured data, but they require programming skills to set up and customize. Thus, although many users can add data to a database set up by a programmer, only a minority are able to modify the schema. To ease this bottleneck, several systems, including Filemaker⁴⁵ and Access⁴⁶, are intended for use by non-programmers. But the complexity involved in designing relational schemas, normalising tables, and specifying foreign keys will still put off potential users. Various more user-friendly web based systems exist, including DabbleDB⁴⁷ (which lets users copy and paste data from spreadsheets) and Google Base⁴⁸ (which lets users set up and populate a web form with fields). But unfortunately the ease of use is gained by cutting out much of the relational functionality.

XML, Object Oriented, hierarchical and semi-structured databases

For less uniformly structured data, a variety of forms-based tools and XML databases have been developed (including LiveCycle Designer⁴⁹, Neurosys⁵⁰, Catalyzer⁵¹, InfoPath⁵² and various XML Schema and UML tools), but again these are mainly targetted at software developers to integrate into back end database systems, and are not end user tools. These tools are usually based on specifying an object oriented *schema* for the data to be stored (the classes and fields which are allowed), and designing schemas is a hard task requiring software skills. Once the schema has been defined, it is simple for users to add new data records by filling in forms.

The problem with such schema-based approaches is that users who are not programmers are typically unable to modify or extend the schema when their data or concepts do not fit. This results in the familiar feeling of frustration when filling out forms: what do you do when none of the options is appropriate, or when the questions make assumptions which do not apply, or when there is relevant information for which there are no entries on the form? The form seems to be saying "we only want to know about how your data fits with our expectations" and yet you know that if you go along with it, the end result will be misleading or even wrong.

Aware of the usability limitations of databases built around schemas, Notate takes the more flexible approach of allowing users to create their own tags (similar in function to classes or tables) and also their own connection types (used for associations and properties of objects, or fields). This allows the user to express exactly the information they would like to store in the system, and by providing an indexing system designed around this less restrictive data model, Notate can give most of the benefits of traditional schema-based databases in sorting and filtering structured information, while remaining accessible to mainstream users.

Another possible approach to creating structured documents is to embed the structured content within the document. This is the approach used by XML editors (e.g. XML Spy⁵³), which allow authors to generate documents conforming to a strict schema such as the DocBook⁵⁴ standard for publications. Using such tools to generate correct XML is significantly harder for users than using standard word processors to create normal (unstructured) documents. Journals typically accept submissions in plain document formats and use highly trained specialists to mark up the text into a given XML format. Getting authors to create XML to a fixed schema would not in any event achieve the objectives of Notate, since any given schema (by definition) only covers a restricted range of concepts. Because of this, the starting point for Notate is any document, in plain text, viewed in a web browser.

Conclusion

There is a pressing need in scientific communication and information storage to move beyond text documents as the only record created by researchers. Recent developments in a range of web-related technologies are delivering tools that are simultaneously easy enough to use that they can be adopted by the majority of authors and powerful enough that they can make a significant step towards meeting the growing need for structured content. The requirement is not to jump in a single bound from creating documents to creating highly structured databases, but rather to enhance text incrementally to the point where the results can be picked up and further processed by automated tools.

Among these emerging technologies, Textensor Notate is focused on addressing the practical issues currently limiting the uptake of structured content technologies. The first requirement is usability by subject experts who are not IT experts. The second is to ensure that the rewards for using the tools are immediately apparent and compensate the user for putting the effort in. It comprises a package of features operating around fine-grained annotation of documents taken from the web or uploaded from local files, and slots all annotations into a sophisticated tag-based index and content management system to give the user immediate benefits as more material is added.

Search engines are necessary, but not sufficient, for organising scientific content. Automated text mining and natural language processing is an interesting research topic, but machines cannot yet be trusted to read, comprehend and summarize the important concepts from papers. Authors and researchers, on the other hand, are extremely good at reading, comprehending, and summarizing the articles they read. Notate aims at enabling authors and readers to use their knowledge to highlight, tag and link concepts in documents, and in the process create sophisticated indexes and cross-references of research material. It focusses on meeting the immediate needs of researchers in organising the documents they read, enabling them to use the system as a more reliable memory and (in Vannevar Bush's words):-

... 'reacquire the priviledge of forgetting the manifold things he does not need to have immediately at hand, with some assurance that he can find them again if they prove important'

A focus on meeting these immediate needs for a broad base of users is the best way to drive the technology forward and develop usable and effective tools for generating and using structured scientific information.

Links and References

- [1] Vannevar Bush: As we may think, The Atlantic Monthly, July 1945: www.theatlantic.com/doc/194507/bush
Seminal early paper on annotations and links
- [2] D C Engelbart, Augmenting Human Intellect: a conceptual framework (1962), SRI proect 278): <http://www.bootstrap.org/augdocs/friedewald030402/augmentinghumanintellect/AHI62.pdf>
- [3] CAS, www.cas.org *American Chemical Society database curated from chemistry abstracts*
- [4] Semantic Structures for Scientific Writing, Anita de Waard: www.cs.uu.nl/people/anita/papers/SWDaysDeWaard1209.pdf
- [5] Connotea, www.connotea.org *Social bookmarking tool by Nature Publishing Group*
- [6] del.icio.us, www.del.icio.us *Centralized social bookmarking*
- [7] 'Ontology is Overrated: Categories, Links, and Tags', www.shirky.com/writings/ontology_ouerrated.html
- [8] flickr, www.flickr.com *Social tagging of photographs*
- [9] citeulike, www.citeulike.org
- [10] Jon Udell's screencast of tags in Windows Vista photo gallery, blog.jonudell.net/2007/02/21/tagging-and-folding-in-photo-gallery
- [11] RDF, www.w3.org/RDF/ *Resource Description Framework*
- [12] Nature (2006) | doi:10.1038/nature05535: www.nature.com/nature/peerreview/debate/nature05535.html
Overview, Nature's peer review trial
- [13] gibeo.net, www.gibeo.net

- [14] PlosOne annotations, www.plosone.org
- [15] blinklist, www.blinklist.com
- [16] photobucket, www.photobucket.com
- [17] D Karger, B Katz, J Lin, D Quan, Proc 8th Intl Conf on intelligent user interfaces: doi.acm.org/10.1145/604045.604091
- [18] Haystack, haystack.lcs.mit.edu/
- [19] Annotea project, www.annotea.org
- [20] mystickies, www.mystickies.com *stickies that can be attached to any web page by x,y location*
- [21] posticky, www.posticky.com *leaving sticky notes on 3rd party sites with a bookmarklet*
- [22] bookmarklets, en.wikipedia.org/wiki/Bookmarklet *Descriptions of bookmarklets*
- [23] stikkit, www.stikkit.com *Text mining to guess people's names*
- [24] Thinkature, thinkature.com *Electronic whiteboard on a website*
- [25] hyperland, hyperland.com
- [26] (www.xanadu.com), www.xanadu.com
- [27] MICROCOSM history, www.mmrg.ecs.soton.ac.uk/projects/microcosm.html
- [28] Peter Buneman, Adriane Chapman, and James Cheney. Provenance management in curated databases. In Proceedings of ACM SIGMOD International Conference on Management of Data, pages 539-550, 2006.: <http://homepages.inf.ed.ac.uk/opb/papers/sigmod2006.pdf>
- [29] purpleslurple, purpleslurple.net *adding identifiers to paragraphs of web pages*
- [30] xpath, www.w3.org/TR/xpath *addressing particular nodes of XML documents*
- [31] wikipedia, www.wikipedia.org
- [32] openwetware, openwetware.org *a shared project for uploading and publishing experimental data and protocols*
- [33] semantic mediawiki, en.wikipedia.org/wiki/Semantic_MediaWiki
- [34] Ikewiki, ikewiki.salzburgresearch.at *a semantic wiki which layers connections from the current wiki page to terms on the same wiki page*
- [35] Knewco, www.knewco.com
- [36] T. Berners-Lee, J. Hendler, O. Lassila, The Semantic Web, Scientific Am, May 2001, pp 34-43: www.sciam.com/article.cfm?articleID=00048144-10D2-1C70-84A9809EC588EF21
- [37] N. Shadbolt, W. Hall, T. Berners-Lee, The Semantic Web Revisited, IEEE Intelligent Systems, May 2006, pp 96-101: eprints.ecs.soton.ac.uk/12614/
- [38] FOAF project, www.foaf-project.org
- [39] Haystack project, haystack.lcs.mit.edu
- [40] Gene Ontology, www.geneontology.org
- [41] Protege, protege.stanford.edu
- [42] Knowtator, bionlp.sourceforge.net *Knowtator, a protege plugin for natural language processing researchers*
- [43] David Huynh, David Karger, Robert Miller, Exhibit: Lightweight Structured Data Publishing, to appear May 2007, in proc. WWW 2007: people.csail.mit.edu/dfhuynh/research/papers/www2007-exhibit.pdf
- [44] JSON, www.json.org *Javascript object notation - a simple way to exchange structured data*
- [45] Filemaker, www.filemaker.com
- [46] Access, office.microsoft.com/access
- [47] DabbleDB, dabbledb.com
- [48] Google Base, base.google.com
- [49] LiveCycle Designer, www.adobe.com/products/server/adobedesigner/
- [50] Neurosys, neurosys.cns.montana.edu
- [51] Catalyzer, www.axiope.com
- [52] InfoPath, office.microsoft.com/infopath

[53] XML Spy, www.altova.com

[54] DocBook, www.docbook.org