Semantic Annotation and Transcoding: Making Text and Multimedia Contents More Usable on the Web

Katashi Nagao
IBM Research, Tokyo Research Laboratory
1623–14 Shimotsuruma, Yamato, Kanagawa 242–8502, Japan
knagao@jp.ibm.com

Abstract

This paper proposes an easy and simple method for constructing a super-structure on the Web which provides current Web contents with new value and new means of use. The super-structure is based on external annotations to Web documents. We have developed a system for any user to annotate any element of any Web document with additional information. We have also developed a proxy that transcodes requested contents by considering annotations assigned to them. In this paper, we classify annotations into three categories. One is linguistic annotation which helps the transcoder understand the semantic structure of textual elements. The second is commentary annotation which helps the transcoder manipulate nontextual elements such as images and sounds. The third is multimedia annotation, which is a combination of the above two types. All types of annotation are described using XML, and correspondence between annotations and document elements is defined using URLs and XPaths. We call the entire process "semantic transcoding" because we deal with the deep semantic content of documents with annotations. The current semantic transcoding process mainly handles text and video summarization, language translation, and speech synthesis of documents including images. Another use of annotation is for knowledge discovery from contents. Using this idea, we have also developed a system which discovers knowledge from Web documents, and generates a document which includes the discovered knowledge and summaries of multiple documents related to the same topic.

1 Introduction

The conventional Web structure can be considered as a graph on a plane. In this paper, we propose a method for extending such planar graph to a three-dimensional structure that consisting of multiple planar layers. Such metalevel structure is based on external annotations on documents on the Web. (The original concept of external annotation was given in [5]. We use this term in a more general sense.)

Figure 1 represents the concept of our approach.

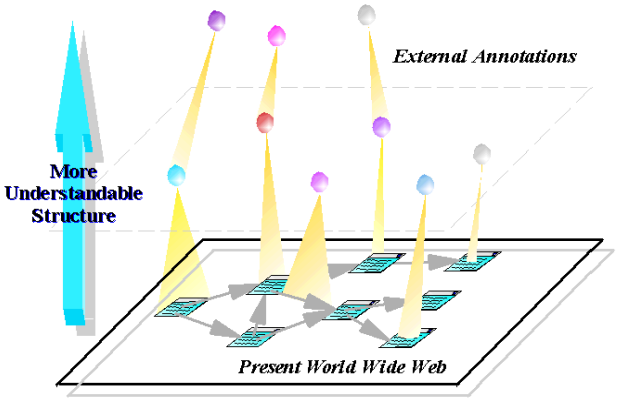

Figure 1: Super-structure on the Web

A super-structure on the Web consists of layers of content and metacontent. The first layer corrensponds to the set of metacontents of base documents. The second layer corresponds to the set of metacontents of the first layer, and so on. We generally consider such metacontent an external annotation. A famous example of external annotations is external links that can be defined outside of the set of link-connected

documents. These external links have been discussed in the XML (Extensible Markup Language) community but they have not yet been implemented in the current Web architecture [17].

Another popular example of external annotation is comments or notes on Web documents created by people other than the author. This kind of annotations is helpful for readers evaluating the documents. For example, images without alternative descriptions are not understandable for visually-challenged people. If there are comments on these images, these people will understand the image contents by listening to them via speech transcoding. This example is explained later in more detail.

We can easily imagine that an open platform for creating and sharing annotaions would greatly extend the expressive power and value of the Web.

1.1 Content Adaptation

Annotations do not just increase the expressive power of the Web but also play an important role in content reuse. An example of content reuse is, for example, the transformation of content depending on user preferences.

Content adaptation is a type of transcoding which considers a users' environment such as devices, network bandwidth, profiles, and so on. Such adaptation sometimes also involves a deep understanding of the original document contents. If the transcoder fails to analyse the semantic structure of a document, then the results may cause user misunderstanding.

Our technology assumes that external annotations help machines to understand document contents so that transcoding can have higher quality. We call such transcoding based on annotation "semantic transcoding."

The overall configuration of semantic transcoding can be viewed in Figure 2.

There are three main new parts in this system: an annotation editor, an annotation server, and a transcoding proxy server. The remaining parts of the system are a conventional Web server and a browser.

There are previous work on device dependent adaptation of Web documents [7]. The developed system can dynamically filter, convert or reformat data for content sharing across disparate systems, users, and emerging pervasive computing devices.

They claim that the transcoding benefits include:

1. eliminating the expense of re-authoring or porting data and content-generating applications for multiple systems and devices
2. improving mobile employee communications and effectiveness, and
3. creating easier access for customers who are using a variety of devices to purchase products and services.

The technology enables the modification of HTML (HyperText Markup Language) documents, such as converting images to links to retrieve images, converting simple tables to bulleted lists, removing features not supported by a device such as JavaScript or Java applets, removing references to image types not supported by a device, and removing comments. It can also transform XML documents by selecting and applying the right stylesheet for the current request based on information in the relevant profiles. These profiles for preferred transcoding services are defined for an initial set of devices.

Our transcoding involves deeper levels of document understanding. Therefore, human intervention into machine understanding of documents is required. External annotations of additional information is a guide for machines to understand. Of cource, some profiles for user contexts will work as a guide to transcode, but it is clear that such profiles are insufficient for transcoders to recognize deep document characteristics.

1.2 Knowledge Discovery

Another use of annotations is in knowledge discovery, where huge amounts of Web contents are automatically mined for some essential points. Unlike conventional search engines that retrieve Web pages using user specified keywords, knowledge miners create a single document that satisfies a user's request. For example, the knowledge miner may generate a summary document on a

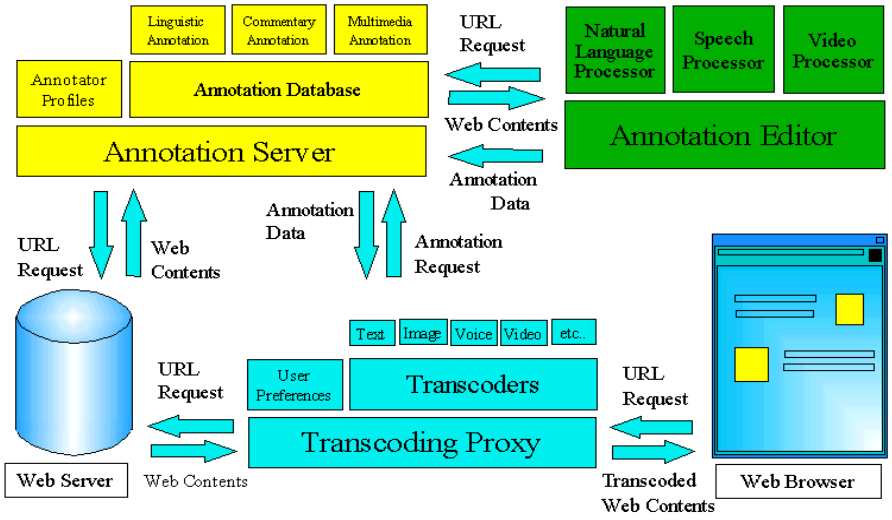

Figure 2: Configuration of semantic transcoding

certain company's product strategy for the year from many kinds of information resources of its products on the Web.

Currently, we are developing an information collector that gathers documents related to a topic and generates a document containing a summary of each document.

There are many unresolved issues before we can realize true knowledge discovery, but we can say that annotations facilitate this activity.

2 External Annotation

We have developed a simple method to associate external annotations with any element of any HTML document. We use URLs (Uniform Resource Locators), XPaths (location identifiers in the document) [18], and document hash codes (digest values) to identify HTML elements in documents. We have also developed an annotation server that maintains the relationship between contents and annotations and transfers requested annotations to a transcoder.

Our annotations are represented as XML formatted data and divided into three categories: linguistic, commentary, and multimedia annotation. Multimedia (especially video) annotation is a combination of the other two types of annotation.

2.1 Annotation Environment

Our annotation environment consists of a client side editor for the creation of annotations and a server for the management of annotations.

The annotation environment is shown in Figure 3.

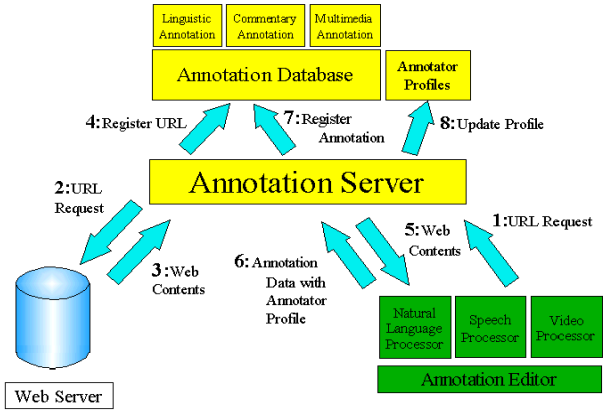

Figure 3: Annotation environment

The process flows as follows (in this example case, an HTML file is processed):

1. The user runs the annotation editor and requests an URL as a target of annotation.
2. The annotation server accepts the request and sends it to the Web server.
3. The annotation server receives the Web document.

4. The server calculates the document hash code (digest value) and registers the URL with the code to its database.

5. The server returns the Web document to the editor.

6. The user annotates the requested document and sends the result to the server with some personal data (name, professional areas, etc.).

7. The server receives the annotation data and relates it with its URL in the database.

8. The server also updates the annotator profiles.

Below we explain the editor and the server in more detail.

2.2 Annotation Editor

Our annotation editor, implemented as a Java application, can communicate with the annotation server explained below.

The annotation editor has the following functions:

1. To register targets of annotation to the annotation server by sending URLs

2. To specify any element in the document using the Web browser

3. To generate and send annotation data to the annotation server

4. To reuse previously-created annotations when the target contents are updated

An example screen of our annotation editor is shown in Figure 4.

The left window of the editor shows the document object structure of the HTML document. The right window shows some text that was selected on the Web browser (shown on the right hand). The selected area is automatically assigned an XPath.

Using the editor, the user annotates text with linguistic structure (grammatical and semantic structure, described later) and adds a comment to an element in the document. The editor is

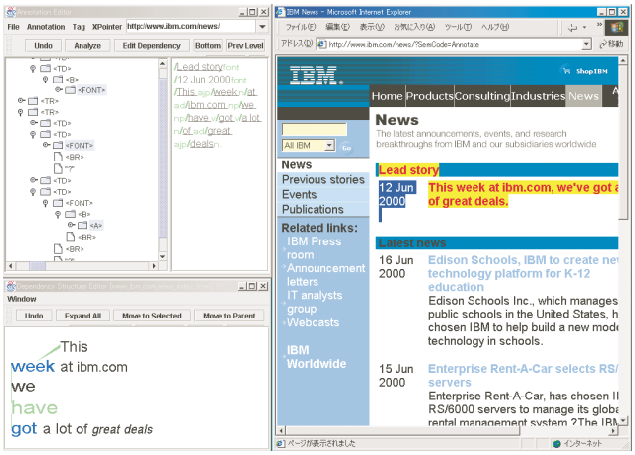

Figure 4: Annotation editor with Web browser

capable of natural language processing and interactive disambiguation. The user will modify the result of the automatically-analyzed sentence structure as shown in Figure 5.

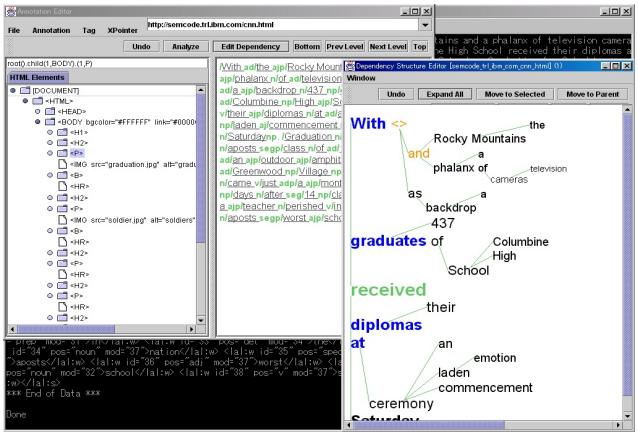

Figure 5: Annotation editor with linguistic structure editor

2.3 Annotation Server

Our annotation server receives annotation data from any annotator and classifies it according to the annotator. The server retrieves documents from URLs in annotation data and registers the document hash codes with their URLs in its annotation database. The hash codes are used to find differences between annotated documents and updated documents identified by the same URL. A hash code of document internal structure or DOM (Document Object Model) enables the server to discover modified elements in the annotated document [8].

The annotation server makes a table of annotator names, URLs, XPaths, and document hash codes. When the server accepts a URL as a request from a transcoding proxy (described below), the server returns a list of XPaths with associated annotation files, their types (linguistic or commentary), and a hash code. If the server receives an annotator's name as a request, it responds with the set of annotations created by the specified annotator.

We are currently developing a mechanism for access control between annotation servers and normal Web servers. If authors of original documents do not want to allow anyone to annotate their documents, they can add a statement about it in the documents, and annotation servers will not retrieve such contents for the annotation editors.

2.4 Linguistic Annotation

The purpose of linguistic annotation is to make WWW texts machine-understandable (on the basis of a new tag set), and to develop content-based presentation, retrieval, question-answering, summarization, and translation systems with much higher quality than is currently available. The new tag set was proposed by the GDA (Global Document Annotation) project [4]. It is based on XML, and designed to be as compatible as possible with HTML, TEI [15], CES [2], EAGLES [3], and LAL [16]. It specifies modifier-modifiee relations, anaphor-referent relations, word senses, etc.

An example of a GDA-tagged sentence is as follows:

```
<su><np rel="agt" sense="time0">Time
</np><v sense="fly1">flies</v>
<adp rel="eg"><ad sense="like0">like
</ad><np>an <n sense="arrow0">arrow
</n></np></adp>.</su>
```

`<su>` means sentential unit. `<n>`, `<np>`, `<v>`, `<ad>` and `<adp>` mean noun, noun phrase, verb, adnoun or adverb (including preposition and postposition), and adnominal or adverbial phrase, respectively[1] .

[1] A more detailed description of the GDA tag set can be found at `http://www.etl.go.jp/etl/nl/GDA/tagset.html`.

The `rel` attribute encodes a relationship in which the current element stands with respect to the element that it semantically depends on. Its value is called a relational term. A relational term denotes a binary relation, which may be a thematic role such as agent, patient, recipient, etc., or a rhetorical relation such as cause, concession, etc. For instance, in the above sentence, `<np rel="agt" sense="time0">Time</np>` depends on the second element `<v sense="fly1">flies</v>`. `rel="agt"` means that `Time` has the agent role with respect to the event denoted by `flies`.

The `sense` attribute encodes a word sense.

Linguistic annotation is generated by automatic morphological analysis, interactive sentence parsing, and word sense disambiguation by selecting the most appropriate paraphrase.

Some research issues on linguistic annotation are related to how the annotation cost can be reduced within some feasible levels. We have been developing some machine-guided annotation interfaces that conceal the complexity of annotation. Machine learning mechanisms also contribute to reducing the cost because they can gradually increase the accuracy of automatic annotation.

In principle, the tag set does not depend on language, but as a first step we implemented a semi-automatic tagging system for English and Japanese.

2.5 Commentary Annotation

Commentary annotation is mainly used to annotate non-textual elements like images and sounds with some additional information. Each comment can include not only tagged texts but also other images and links. Currently, this type of annotation appears in a subwindow that is overlayed on the original document window when a user locates a mouse pointer at the area of a comment-added element as shown in Figure 6.

Users can also annotate text elements with information such as paraphrases, correctly-spelled words, and underlines. This type of annotation is used for text transcoding that combines such comments on texts and original texts.

Commentary annotaion on hyperlinks is also

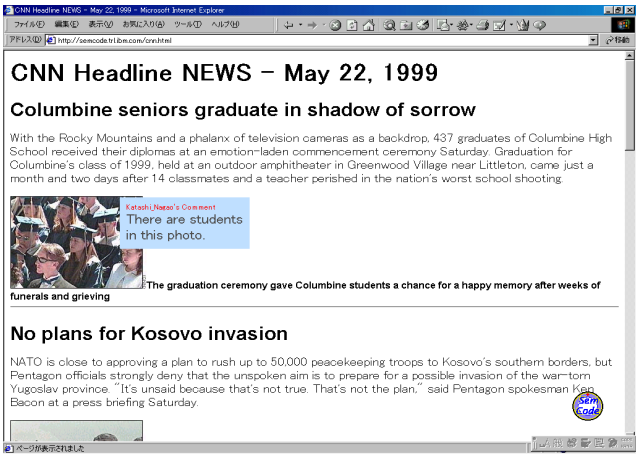

Figure 6: Comment overlay on the document

available. This contributes to quick introduction of target documents before clicking the links. If there are linguistic annotations on the target documents, the transcoders can generate summaries of these documents and relate them with hyperlinks in the source document.

There are some previous work on sharing comments on the Web. ComMentor is a general meta-information architecture for annotating documents on the Web [11]. This architecture includes a basic client-server protocol, meta-information description language, a server system, and a remodeled NCSA Mosaic browser with interface augmentations to provide access to its extended functionality. ComMentor provides a general mechanism for shared annotations, which enables people to annotate arbitrary documents at any position in-place, share comments/pointers with other people (either publicly or privately), and create shared "landmark" reference points in the information space. There are several annotation systems with a similar direction, such as CoNote and the Group Annotation Transducer [13].

These systems are often limited to particular documents or documents shared only among a few people. Our annotation and transcoding system can also handle multiple comments on any element of any document on the Web. Also, a community wide access control mechanism can be added to our transcoding proxy. If a user is not a member of a particular group, then the user cannot access the transcoding proxy that is for group use only. In the future, transcoding proxies and annotation servers will communicate with some secured protocol that prevents some other server or proxy from accessing the annotation data.

Our main focus is adaptation of WWW contents to users, and sharing comments in a community is one of our additional features. We apply both commentary and linguistic annotations to semantic transcoding.

2.6 Multimedia Annotation

Our annotation technique can also be applied to multimedia data such as digital video. Digital video is becoming a necessary information source. Since the size of these collections is growing to huge numbers of hours, summarization is required to effectively browse video segments in a short time without losing the significant content. We have developed techniques for semi-automatic video annotation using a text describing the content of the video. Our techniques also use some video analysis methods such as automatic cut detection, characterization of frames in a cut, and scene recognition using similarity between several cuts.

There is another approach to video annotation. MPEG-7 is an effort within the Moving Picture Experts Group (MPEG) of ISO/IEC that is dealing with multimedia content description [9].

Using content descriptions, video coded in MPEG-7 is concerned with transcoding and delivery of multimedia content to different devices. MPEG-7 will potentially allow greater input from the content publishers in guiding how multimedia content is transcoded in different situations and for different client devices. Also, MPEG-7 provides object-level description of multimedia content which allows a higher granularity of transcoding in which individual regions, segments, objects and events in image, audio and video data can be differentially transcoded depending on publisher and user preferences, network bandwidth and client capabilities.

Our method will be integrated into tools for authoring MPEG-7 data. However, we do not currently know when the MPEG-7 technology will be widely available.

Our video annotation includes automatic segmentation of video, semi-automatic linking of video segments with corresponding text segments, and interactive naming of people and objects in video frames.

Video annotation is performed through the following three steps.

First, for each video clip, the annotation system creates the text corresponding to its content. We employed speech recognition for the automatic generation of a video transcript. The speech recognition module also records correspondences between the video frames and the words. The transcript is not required to describe the whole video content. The resolution of the description effects the final quality of the transcoding (e.g., summarization).

Second, some video analysis techniques are applied to characterize scenes, segments (cuts and shots), and individual frames in video. For example, by detecting significant changes in the color histogram of successive frames, frame sequences can be separated into cuts and shots.

Also, by searching and matching prepared templates to individual regions in the frame, the annotation system identifies objects. The user can specify significant objects in some scene in order to reduce the time to identify target objects and to obtain a higher recognition success ratio. The user can name objects in a frame simply by selecting words in the corresponding text.

Third, the user relates video segments to text segments such as paragraphs, sentences, and phrases, based on scene structures and object-name correspondences. The system helps the user to select appropriate segments by prioritizing based on the number of objects detected, camera movement, and by showing a representative frame of each segment.

We developed a video annotation editor capable of scene change detection, speech recognition, and correlation of scenes and words. An example screen of our video annotation editor is shown in Figure 7.

On the editor screen, the user can specify a particular object in a frame by dragging a rectangle. Using automatic object tracking techniques, the annotation editor can generate descriptions of an object in a video frame. The description is represented as XML data, and contains object coordinates in start and end frames, time codes of the start and end frames, and motion trails (series of coordinates for interpolation of object movement).

Figure 7: Video annotation editor

The object descriptions are connected with liguistic annotation by adding appropriate XPaths to the tags of corresponding names and expressions in the video transcript.

As mentioned later, the annotation of video objects can be used for creation of hyper-video, in which annotated objects are hyperlinked with external information, and objects are retrieved with keywords.

3 Semantic Transcoding

Semantic transcoding is a transcoding technique based on external annotations, used for content adaptation according to user preferences. The transcoders here are implemented as an extension to an HTTP (HyperText Transfer Protocol) proxy server. Such an HTTP proxy is called a transcoding proxy.

Figure 8 shows the environment of semantic transcoding.

The information flow in transcoding is as follows:

1. The transcoding proxy receives a request URL with a client ID.

2. The proxy sends the request of the URL to the Web server.

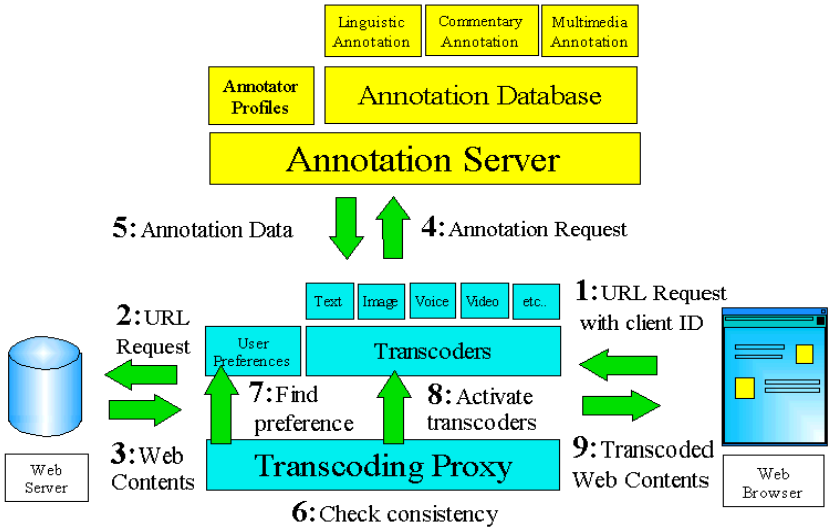

Figure 8: Transcoding environment

3. The proxy receives the document and calculates its hash code.

4. The proxy also asks the annotation server for annotation data related to the URL.

5. If the server finds the annotation data of the URL in its database, it returns the data to the proxy.

6. The proxy accepts the data and compares the document hash code with that of the already retrieved document.

7. The proxy also searches for the user preference with the client ID. If there is no preference data, the proxy uses a default setting until the user gives the preference.

8. If the hash codes match, the proxy attempts to transcode the document based on the annotation data by activating the appropriate transcoders.

9. The proxy returns the transcoded document to the client Web browser.

We explain in more detail the transcoding proxy and various kinds of transcoding.

3.1 Transcoding Proxy

We employed IBM's WBI (Web Intermediaries) as a development platform to implement the semantic transcoding system [6]. WBI is a customizable and extendable HTTP proxy server. WBI provides APIs (Application Programming Interfaces) for user level access control and easy manipulation of input/output data of the proxy.

The transcoding proxy based on WBI has the following functionality:

1. Maintenance of personal preferences

2. Gathering and management of annotation data

3. Activation and integration of transcoders

3.1.1 User preference management

For the maintenance of personal preferences, we use the web browser's cookie to identify the user. The cookie holds a user ID assigned by the transcoding proxy on the first access and the ID is used to identify the user and to select user preferences defined at the last time. The ID stored as a cookie value allows the user, for example, to change an access point using DHCP (Dynamic Host Configuration Protocol) with the same preference setting. There is one technical problem. Generally, cookies can be accessed only by the HTTP servers that have set their values and ordinary proxies do not use cookies for user identification. Instead, conventional proxies identify the client by the hostname and IP address. Thus, when the user ac-

cesses our proxy and sets/updates the preferences, the proxy server acts as an HTTP server to access the browser's cookie data and associates the user ID (cookie value) and the hostname/IP address. When the transcoding proxy works as a coventional proxy, it receives the client's hostname and IP address, retrieves the user ID, and then obtains the preference data. If the user changes access point and hostname/IP address, our proxy performs as a server again and reassociates the user ID and such client IDs.

3.1.2 Collecting and indexing annotation data

The transcoding proxy communicates with annotation servers that hold the annotation database. The second step of semantic transcoding is to collect annotations distributed among several servers.

The transcoding proxy creates a multi-server annotation catalog by crawling distributed annotation servers and gathering their annotation indeces. The annotation catalog consists of server name (e.g., URL) and its annotation index (set of annotator names and identifiers of the original document and its annotation data). The proxy uses the catalog to decide which annotation server should be accessed to get annotation data when it receives a user's request.

3.1.3 Integrating the results of multiple transcoders

The final stage of semantic transcoding is to transcode requested contents depending on user preferences and then to return them to the user's browser. This stage involves activation of appropriate transcoders and integration of their results.

As mentioned previously, there are several types of transcoding. In this paper we describe four types: text, image, voice, and video transcodings.

3.2 Text Transcoding

Text transcoding is the transformation of text contents based on linguistic annotations. As a first step, we implemented text summarization.

Our text summarization method employs a spreading activation technique to calculate the importance values of elements in the text [10]. Since the method does not employ any heuristics dependent on the domain and style of documents, it is applicable to any linguistically-annotated document. The method can also trim sentences in the summary because importance scores are assigned to elements smaller than sentences.

A linguistically-annotated document naturally defines an intra-document network in which nodes correspond to elements and links represent the semantic relations. This network consists of sentence trees (syntactic head-daughter hierarchies of subsentential elements such as words or phrases), coreference/anaphora links, document/subdivision/paragraph nodes, and rhetorical relation links.

Figure 9 shows a graphical representation of the intra-document network.

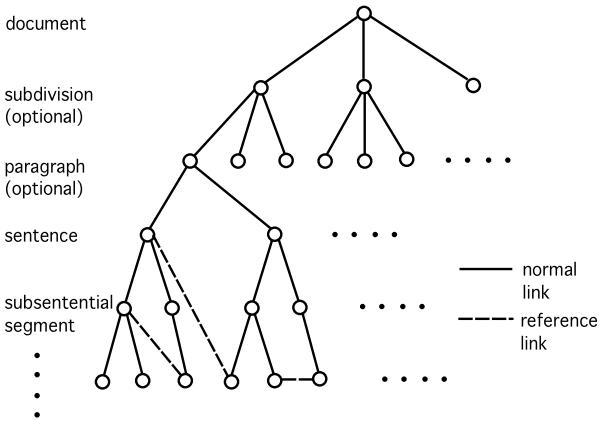

Figure 9: Intra-document network

The summarization algorithm works as follows:

1. Spreading activation is performed in such a way that two elements have the same activation value if they are coreferent or one of them is the syntactic head of the other.

2. The unmarked element with the highest activation value is marked for inclusion in the summary.

3. When an element is marked, the following elements are recursively marked as well, until no more elements are found:

- the marker's head
- the marker's antecedent
- the marker's compulsory or *a priori* important daughters, the values of whose relational attributes are `agt` (agent), `pat` (patient), `rec` (recipient), `sbj` (syntactic subject), `obj` (syntactic object), `pos` (possessor), `cnt` (content), `cau` (cause), `cnd` (condition), `sbm` (subject matter), etc.
- the antecedent of a zero anaphor in the marker with some of the above values for the relational attribute

4. All marked elements in the intra-document network are generated preserving the order of their positions in the original document.

5. If a size of the summary reaches the user-specified value, then terminate; otherwise go back to Step 2.

The size of the summary can be changed by simple user interaction. Thus the user can see the summary in a preferred size by using an ordinary Web browser without any additional software. The user can also input any words of interest. The corresponding words in the document are assigned numeric values that reflect degrees of interest. These values are used during spreading activation for calculating importance scores.

Figure 10 shows the summarization result on the normal Web browser. The top document is the original and the bottom one is the summarized version.

Another kind of text transcoding is language translation. We can predict that translation based on linguistic annotations will produce a much better result than many existing systems. This is because the major difficulties of present machine translation come from syntactic and word sense ambiguities in natural languages, which can be easily clarified in annotation. An example of the result of English-to-Japanese translation is shown in Figure 11.

Furthermore, we are developing a dictionary-based text paraphrasing as another repertoire of text transcoding. Using word sense attributes in linguistic annotation and dictionary definitions,

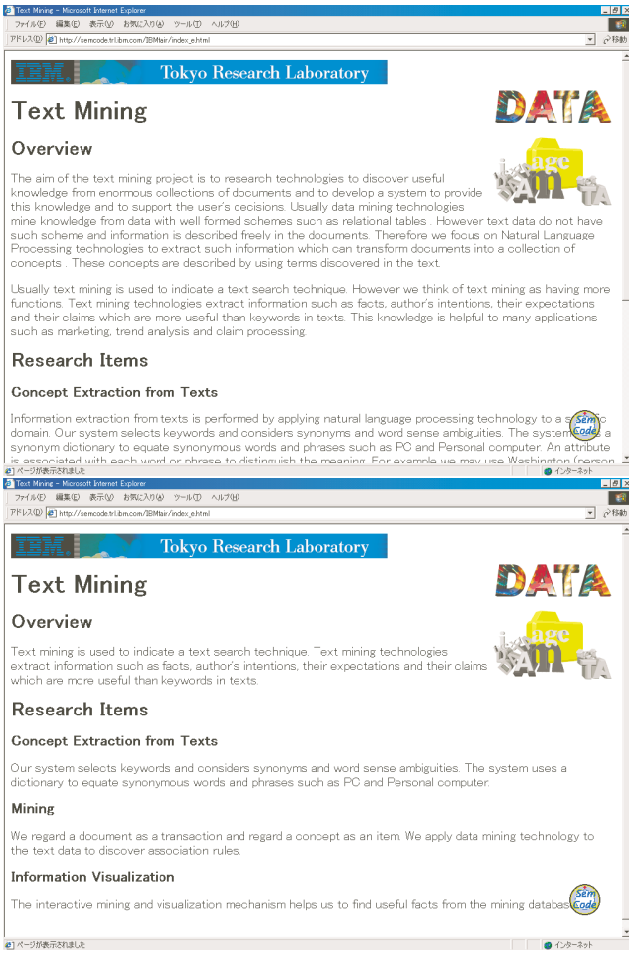

Figure 10: Original and summarized documents

Figure 11: Translated document

difficult words are replaced with more readable expressions. These expressions are generated by modifying the dictionary definitions for word senses according to the local contexts of the target words.

3.3 Image Transcoding

Image transcoding is to convert images into these of different size, color (full color or grayscale), and resolution (e.g., compression ratio) depending on user's device and communication capability. Links to these converted images are made from the original images. Therefore, users will notice that the images they are looking at are not original if there are links to similar images.

Figure 12 shows the document that is summarized in one-third size of the original and whose images are reduced to half. In this figure, the preference setting subwindow is shown on the right hand. The window appears when the user double-clicks the icon on the lower right corner (the transcoding proxy automatically inserts the icon). Using this window, the user can easily modify the parameters for transcoding.

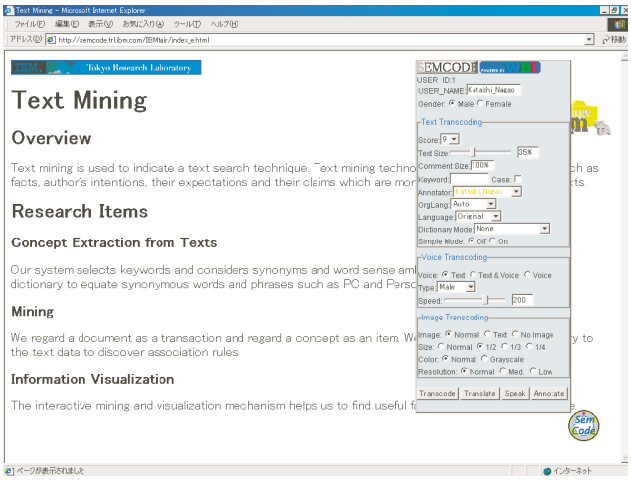

Figure 12: Image transcoding (and preference setting window)

By combining image and text transcodings, the system can, for example, convert contents to just fit the client screen size.

3.4 Voice Transcoding

Voice synthesis also works better if the content has linguistic annotation. For example, a speech synthesis markup language is being discussed in [12]. A typical example is processing proper nouns and technical terms. Word level annotations on proper nouns allow the transcoders to recognize not only their meanings but also their readings.

Voice transcoding generates spoken language version of documents. There are two types of voice transcoding. One is when the transcoder synthesizes sound data in audio formats such as MP3 (MPEG-1 Audio Layer 3). This case is useful for devices without voice synthesis capability such as cellular phones and PDAs (Personal Digital Assistants). The other is when the transcoder converts documents into more appropriate style for voice synthesis. This case requires that a voice synthesis program is installed on the client side. Of cource, the synthesizer uses the output of the voice synthesizer. Therefore, the mechanism of document conversion is a common part of both types of voice transcoding.

Documents annotated for voice include some text in commentary annotation for non-textual elements and some word information in linguistic annotation for the reading of proper nouns and unknown words in the dictionary. The document also contains phrase and sentence boundary information so that pauses appear in appropriate positions.

Figure 13 shows an example of the voice-transcoded document in which icons that represent the speaker are inserted. When the user clicks the speaker icon, the MP3 player software is invoked and starts playing the synthesized voice data.

3.5 Video Transcoding

Video transcoding employs video annotation that consists of linguistically-marked-up transcripts such as closed captions, time stamps of scene changes, representative images (key frames) of each scene, and additional information such as program names, etc. Our video transcoding has several variations, including video summarization, video to document transformation, video translation, etc.

Figure 13: Voice transcoding

Video summarization is performed as a by-product of text summarization. Since a summarized video transcript contains important information, corresponding video sequences will produce a collection of significant scenes in the video. Summarized video is played by a player we developed. An example screen of our video player is shown in Figure 14.

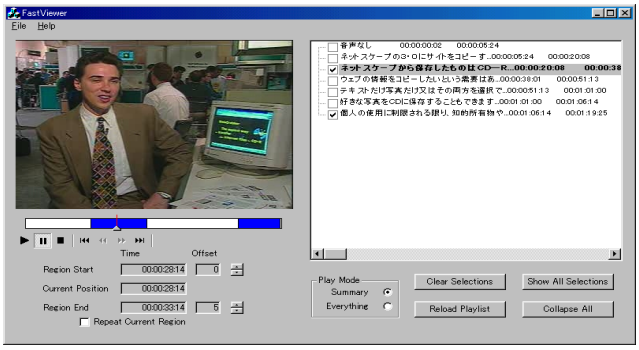

Figure 14: Video player with summarization function

There are some previous work on video summarization such as Infomedia [14] and CueVideo [1]. They create a video summary based on automatically extracted features in video such as scene changes, speech, text and human faces in frames, and closed captions. They can transcode video data without annotations. However, currently, an accuracy of their summarization is not practical because of the failure of automatic video analysis. Our approach to video summarization has sufficient quality for use if the data has enough semantic annotation. As mentioned earlier, we have developed a tool to help annotators to create semantic annotation data for multimedia data. Since our annotation data is task-independent and versatile, annotations on video are worth creating if the video will be used in different applications such as automatic editing and information extraction from video.

Video to document transformation is another type of video transcoding. If the client device does not have video playing capability, the user cannot access video contents. In this case, the video transcoder creates a document including important images of scenes and texts related to each scene. Also, the resulting document can be summarized by the text transcoder.

Our system implements two types of video translation. One is a translation of automatically generated subtitle text. The subtitle text is generated from the transcript with time codes. The format of the text is as follows:

```
<subtitle duration="00:01:19">
<time begin="00:00:00"/><clear/>
No speech
<time begin="00:00:05"/>....
<time begin="00:00:07"/>....
<time begin="00:00:12"/><clear/>
....
</subtitle>
```

The text transcoder can translate the subtitle text into different languages as the user wants, and the video player shows the results synchronized with the video.

An example screen of the video player with subtitle window is shown in Figure 15.

Figure 15: Video player with subtitle window

The other type of video translation is performed in terms of a combination of text and

voice transcodings. First, a video transcript with linguistic annotation is translated by the text transcoder. Then, the result of translation is converted into voice-suitable text by the voice transcoder. Synchronization of video playing and voice synthesis makes another language version of the original video clip. The duration of each voice data is adjusted according to the length of its corresponding video segment by changing speed of synthesized voice.

Using object-level annotations, the video transcoder can create an interactive hyper-video in which video objects are hyperlinked with information such as names, times, locations, related Websites, etc. The annotations are used for object retrieval from multiple video clips and generation of object-featured video summaries.

The above described text, image, voice, and video transcodings are automatically combined according to user demand, so the transcoding proxy has a planning machanism to determine the order of activation of each transcoder necessary for the requested content and user preferences (including client device constraints).

4 Future Plans

We are planning to apply our technology to knowledge discovery from huge online resources. Annotations will be very useful to extract some essential points in documents. For example, an annotator adds comments to several documents, and he or she seems to be a specialist of some particular field. Then, the machine automatically collects documents annotated by this annotator and generates a single document including summaries of the annotated documents.

Also, content-based retrieval of Web documents including multimedia data is being pursued. Such retrieval enables users to ask questions in natural language (either spoken or written).

While our current prototype system is running locally, we are also planning to evaluate our system with open experiments jointly with Keio University in Japan. In addition, we will distribute our annotation editor, with natural language processing capabilities, for free.

5 Concluding Remarks

We have discussed a full architecture for creating and utilizing external annotations. Using the annotations, we realized semantic transcoding that automatically customizes Web contents depending on user preferences.

This technology also contributes to commentary information sharing like ComMentor and device dependent transformation for any device. One of our future goals is to make contents of the WWW intelligent enough to answer our questions asked using natural language. We imagine that in the near future we will not use search engines but will instead use knowledge discovery engines that give us a personalized summary of multiple documents instead of hyperlinks. The work in this paper is one step toward a better solution of dealing with the coming information deluge.

Acknowledgments

The author would like to acknowledge the contributors to the project described in this paper. Koiti Hasida gave the author some helpful advices to apply the GDA tag set to our linguistic and multimedia annotations. Hideo Watanabe developed the prototype of English-to-Japanese language translator. Shinichi Torihara developed the prototype of voice synthesizer. Shingo Hosoya, Yoshinari Shirai, Ryuichiro Higashinaka, Mitsuhiro Yoneoka, and Kevin Squire contributed to implementation of the prototype semantic transcoding system.

References

[1] A. Amir, S. Srinivasan, D. Ponceleon, and D. Petkovic. CueVideo: Automated indexing of video for searching and browsing. In *Proceedings of SIGIR'99*. 1999.

[2] Corpus Encoding Standard (CES). Corpus Encoding Standard. http://www.cs.vassar.edu/CES/.

[3] Expert Advisory Group on Language Engineering Standards (EAGLES). EAGLES online. http://www.ilc.pi.cnr.it/EAGLES/home.html.

[4] Koiti Hasida. Global Document Annotation. http://www.etl.go.jp/etl/nl/gda/.

[5] Masahiro Hori et al. Annotation-based Web Content Transcoding. In *Proceedings of the Ninth International WWW Conference*. 2000.

[6] IBM Almaden Research Center. Web Intermediaries (WBI). http://www.almaden.ibm.com/cs/wbi/.

[7] IBM Corporation. IBM WebSphere Transcoding Publisher. http://www-4.ibm.com/software/webservers/transcoding/.

[8] Hiroshi Maruyama, Kent Tamura, and Naohiko Uramoto. XML and Java: Developing Web applications. Addison-Wesley, 1999.

[9] Moving Picture Experts Group (MPEG). MPEG-7 Context and Objectives. http://drogo.cselt.stet.it/mpeg/standards/mpeg-7/mpeg-7.htm.

[10] Katashi Nagao and Koiti Hasida. Automatic text summarization based on the Global Document Annotation. In *Proceedings of COLING-ACL'98*. 1998.

[11] Martin Roscheisen, Christian Mogensen, and Terry Winograd. Shared Web annotations as a platform for third-party value-added information providers: Architecture, protocols, and usage examples. *Technical Report CSDTR/DLTR*. Computer Science Department, Stanford University, 1995.

[12] The SABLE Consortium. A Speech Synthesis Markup Language. http://www.cstr.ed.ac.uk/projects/ssml.html.

[13] Matthew A. Schickler, Murray S. Mazer, and Charles Brooks. Pan-browser support for annotations and other meta-information on the World Wide Web. *Computer Networks and ISDN Systems*. Vol. 28, 1996.

[14] Michael A. Smith and Takeo Kanade. Video skimming for quick browsing based on audio and image characterization. *Technical Report CMU-CS-95-186*. School of Computer Science, Carnegie Mellon University, 1995.

[15] The Text Encoding Initiative (TEI). Text Encoding Initiative. http://www.uic.edu:80/orgs/tei/.

[16] Hideo Watanabe. Linguistic Annotation Language: The markup langauge for assisting NLP programs. *TRL Research Report RT0334*. IBM Tokyo Research Laboratory, 1999.

[17] World Wide Web Consortium. Extensible Markup Language (XML). http://www.w3.org/XML/.

[18] World Wide Web Consortium. XML Path Language (XPath) Version 1.0. http://www.w3.org/TR/xpath.html.