Agent-based Content Management System

Hidekazu Kubota
Graduate School of Informatics, Kyoto University
Yoshida-Honmachi, Sakyo-ku, Kyoto 606-8501 Japan
kubota@ii.ist.i.kyoto-u.ac.jp

Jaewon Hur
Graduate School of Information Science and Technology, The University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan
jwhur@kc.t.u-tokyo.ac.jp

Toyoaki Nishida
Graduate School of Informatics, Kyoto University
Yoshida-Honmachi, Sakyo-ku, Kyoto 606-8501 Japan
nishida@i.kyoto-u.ac.jp

Abstract

This paper describes the management of huge heterogeneous content using the agent-based content management system. The knowledge card that can wrap up heterogeneous content is proposed. The knowledge channel that is a model of content management system is also proposed and implemented by using conversational agent and dynamic program table. An empirical experiment conducted in three communities implies that the knowledge channel allows the management of huge heterogeneous content.

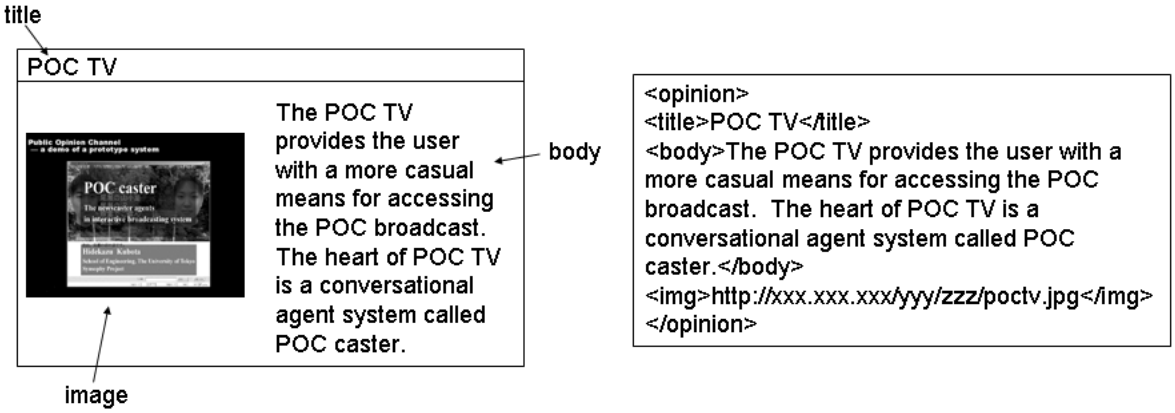
Keywords: Personal content management system, Conversational agent

1 Introduction

The purpose of this paper is computational management for personal content creation. The personal content means here is a set of essays and papers that are created casually. The management of personal content is essential work for human intellectual life at present, for instance many people publish their personal journals on web pages. However, it is troublesome to manage huge personal content because the content can include heterogeneous parts such as text, photos, music and movieclips, its constituent parts are different sizes, and moreover boundaries of its topics are not clear.

This paper is intended as a computational management of the unstructured heterogeneous content mentioned above. The essential idea is a knowledge card that is a well done piece of heterogeneous content. A knowledge card wraps up a few sentences of text and an image (or a movieclip) that represent one topic, so intra-card structure is too heterogeneous for a computer to understand, while inter-cards boundaries are very clear. Such distinction between cards is important clues for a computer to understand the whole structure of the content. In addition to that, these cards are easy to rearrange because their grain size is generally regular.

To manage knowledge card formed content, a knowledge channel model has been proposed. This is a model of an agent-based content management system that integrates conversational process and editorial process for content management. These processes complementarily grow content. The conversational process manages time series of content through conversation, while the editorial process manages spatial relations among content. To support the conversational process, we have developed conversational agent that can talk with people about content, and



(a) The component of the knowledge card

(b) XML form of the knowledge card

Figure 1: The knowledge card

gather their opinions about it. On the other hand, we have developed knowledge landscape that enables people to grasp huge content spatially by using knowledge visualization.

2 Knowledge card approach

The knowledge card can include heterogeneous information, as a title sentence, a few body sentences and an image or a movieclip, nevertheless the card can be roughly retrieved by using text retrieving techniques because natural language sentences in the title and the body are good clues to understand the whole card. At this point, we don't use any formal language or mark up language since describing such language is a hard task for nonprofessional people.

Fig.1(a) shows the components of the knowledge card that consists of "title" and "body" sentences and an "image".

a. title: The title represents a summary of the knowledge card. This is very useful for both humans and computers to understand the whole card. The title in Fig.1(a) shows this card describes about the system called "POC TV".

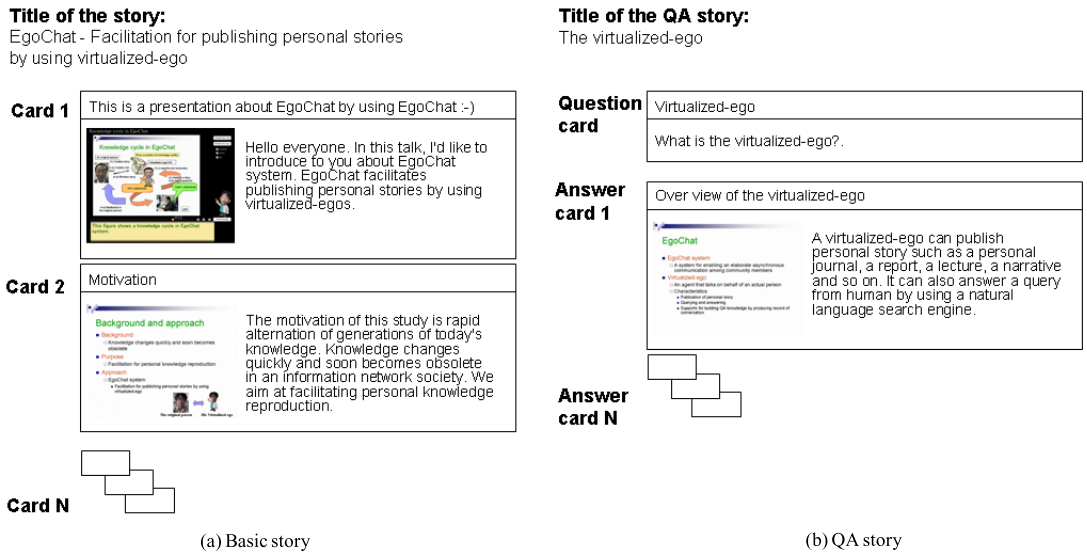
b. body: The body represents detail of the knowledge card. This can be free text about the title of the card.

c. image: The image represents a related figure, picture or movieclip. The image supplements the body text.

The knowledge card is represented by using XML form (Fig.1(b)). The $< opinion >$ element shows the whole card. The text in the $< title >$ element shows the title, the text in $< body >$ element shows the body and the URI in $< img >$ element shows the location of image.

People can create their content by writing and arranging the knowledge cards. "Basic story" with "QA story" is the form of the content. By using the basic story form, people can give context to knowledge cards, each of which is context independent. The basic story consists of one title that summarize the whole story and a sequence of N knowledge cards that are arranged according by order of topics. Fig.2(a) shows an example of the basic story that is about the system called "EgoChat". The first card (Card 1) introduces the story, the second card describes about the motivation of EgoChat, and the following cards explain details.

The QA story represents a conceivable conversation about the basic stories. By using QA story, people can create conversational content. The QA story consists of a question card that



(a) Basic story (b) QA story

Figure 2: Examples of the story

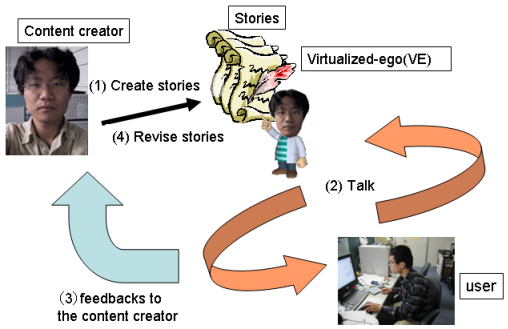


Figure 3: The content circulation mediated by the virtualized-ego

is conceivable question and a sequence of answer cards that is a basic story answering the corresponding question. Fig.2(b) shows an example of the QA story. In Fig.2(b), the conceivable question is "What is the virtualized-ego?" and the following N answer cards answer it.

To manage knowledge cards mentioned above, we have developed EgoChatIII system (Kubota, 2002). The aim of EgoChatIII is to extend opportunity for people to present their content and get comments about them by using conversational agents. It is important for people to gather opinions about their content because the sustainable growth of the content needs a lot of comments. In EgoChatIII system, people delegate their own agent called virtualized-ego to talk about his content.

The virtualized-ego can circulate content between a content creator and a user as follows (Fig.3):

Process 1: The creator accumulates his content into his virtualized-ego by creating basic stories by hand.

Process 2: The virtualized-ego can make a presentation to a user by reading out the text and displaying the images in its basic stories. The virtualized-ego can also talk with the user by using full text search (Kiyota, 2002) from QA stories.

Process 3: The virtualized-ego brings user's opinions about the content back to its creator by email.

Process 4: The creator revises his stories referring to the user's opinion.

An example of the conversation using stories in Fig.2 is as follows:

Virtualized-ego: Hello everyone. In this talk, I'd like to introduce to you about EgoChat system. EgoChat facilitates publishing personal stories by using virtualized-egos.
User: What is the definition of the virtualized-ego?
Virtualized-ego: A virtualized-ego can publish personal story such as a personal journal, a report, a lecture, a narrative and so on...

The virtualized ego searches user's question from his question cards, then replies the answer cards corresponding to the question card. The full text search (Kiyota, 2002) allows a gap between user's question and a question card by resolving modifier-head relation gap [1] .

There are three advantages to using knowledge cards : First, the creator needs not to care about formalizing content except boundaries of cards. Free text can be included in the card. Second, the knowledge card improves reusability of the content. It is easy for the creator to make variations of content because a constituent part of the content is represented by an independent card that can be briefly attached and detached with the content. Third, the knowledge card is easy to retrieve because recent technologies of NLP are good at text processing in a paragraph like the body text of a knowledge card.

3 Knowledge channel

In the content management, the conversational agent like the virtualized-ego seems to be fit for supporting rough sketching process because conversation is the style for informal communication. However, editorial space like a white board is necessary in the process of putting rough sketches into shape because conversation is not accumulative but temporary. To integrate conversational process and editorial process, this paper proposes a knowledge channel model that is a model of an agent-based content management system.

The knowledge channel model consists of a conversational agent and landscape. Scattered stories are assembled into streamed content called "knowledge channel" by a channel policy. The channel policy is a policy document that enables people to declare the intention of managing content. The channel policy controls miscellaneous basic stories under one common policy written by a user who manages the channel. The channel policy is a XML document that consists of four main elements: (a) "flowStrategy" element describes the arrangement of stories. (b) "outlineStrategy" element describes the policy for viewing stories on landscape. (c) "delegationStrategy" element describes the policy for making a presentation by using a conversational agent. (d) "accessStrategy" describes access permission for stories.

The landscape enables a user to understand a set of content spatially. This is not just visualization of content, but externalization of user's understanding about content. The user has leadership in the process of mapping (e.g. categorizing, scaling or locating) content on the landscape while the computer has leadership in the visualization system. Specially, the characteristic of the landscape is growth of the map according to progress of user's understanding. For example, the landscape supports a novice user by not only visualizing a primary map, but hinting the step to following growth of the map. The landscape system should not be a push system but a proposal system. The system must not relocate content without user's permission because the user may be confused if his map of understanding is changed by others.

A user can speak and listen to stories by using a conversational agent while he can view and construct stories by using landscape. The conversational agent is suitable for activating and stirring stories while landscape is suitable for fixing and arranging. People can brush up their stories by repeating such two styles of interaction like annealing steal.

[1] This is indeed Japanese full text search, the English example above is just an approximation.

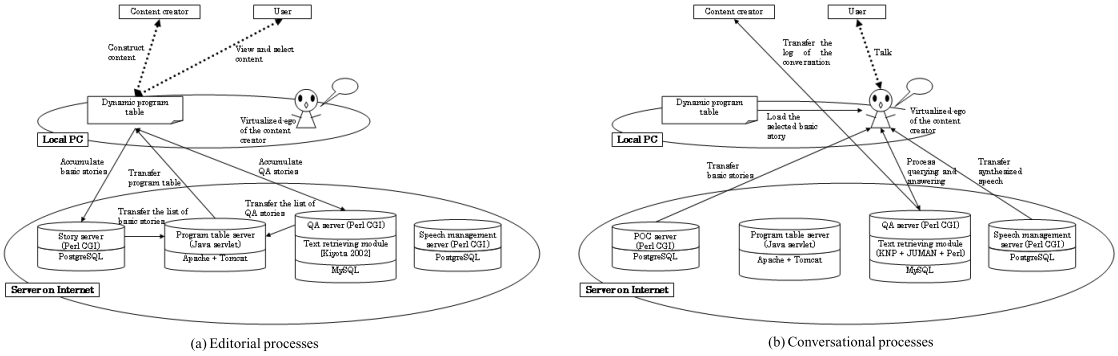


Figure 4: The knowledge channel system

4 Implementation

The knowledge channel model is materialized on a web-based system where the knowledge landscape is partially implemented as a dynamic program table system that visualizes stories as a TV program table according to a channel policy. Fig.4 shows overview of the knowledge channel system. The knowledge channel consists of four servers (Story server, Program table server, QA server and Speech management server) and two web interfaces (Dynamic program table and virtualized-ego).

Story server: Story server accumulates basic stories with their creator, modified date and permission by using PostgreSQL. (This server is originally developed by Fukuhara(Fukuhara, 2001).)

Program table server: Program table server manages the arrangement of stories according to the channel policy.

QA server: QA server accumulates and retrieves QA stories by using text retrieving module (Kiyota, 2002). QA server also transfers the log of the conversation.

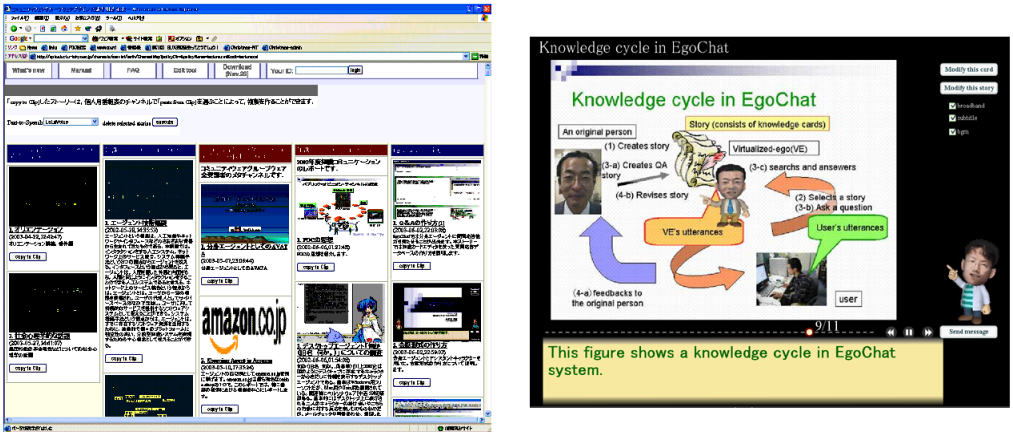
Speech management server: Speech management server manages real-time speech synthesis. This server provides CGI interface to speech synthesis systems and caches speeches that are once synthesized. In this implementation, we use Galatea talk [2] for Japanese speech synthesis and Festival [3] for English speech synthesis.

Dynamic program table: The dynamic program table is the web-based landscape that provides the overview of the stories in HTML form (Fig.5(a)). It also provides knowledge card editor by using Dynamic HTML.

Virtualized-ego: The virtualized-ego is the web-based conversational agent that can talk about the content on behalf of the content creator. Fig.5(b) shows the screen of the virtualized-ego. The agent character with the face of the actual person is on the right end of the screen. It can read out the body text by using synthesized speech, show the image on the center of the screen and accept the user's comment from the keyboard. It is implemented by Macromedia Flash.

Fig.4(a) shows the editorial processes where a content creator constructs his content and a user views and selects interested content on the dynamic program table. The dotted arrows represent human-computer interaction and the solid arrows represent the outline of the content flow. the dynamic program table shows the overview of stories on demand according to the list of basic

[2] http://hil.t.u-tokyo.ac.jp/ galatea/index.html
[3] http://www.cstr.ed.ac.uk/projects/festival/



(a) The screen of the dynamic program table

(b) The screen of the EgoChat

Figure 5: The screen shot of the knowledge channel system

stories and QA stories organized by the channel policy. The content creator can create and edit his stories by using the knowledge card editor on the dynamic program table, and then the content is accumulated into Story server and QA server. On the other hand, the user can view the creator's content and select one of it on the dynamic program table, then the conversational processes start.

Fig.4(b) shows the conversational processes where a virtualized-ego of the content creator talks with a user about the content. The user can ask and listen to the basic story that is read out by the virtualized-ego. The conversation between the user and the virtualized-ego are recorded and sent to the creator of the content.

5 Experiment

The knowledge channel has been experimentally used in 3 communities; "Lecture community", "RISTEX community" and "Conversational informatics community". We have provided 83 program tables that visualize 290 channels including 475 stories. "Lecture community" consists of a professor and class members. This community was formed together with the lecture of Prof. Nishida in Tokyo University (the first term: 2002/5/7 - 2002/8/9, the second term: 2003/4/8 - 2003/9/16). Two channels called "Knowledge communication (2002)" and "Knowledge communication - Paper" were managed in the first term. "Knowledge communication (2002)" is a channel about knowledge communication, which includes 27 stories (274 knowledge cards) created by the professor. "Knowledge communication - Paper" is a channel for papers about agent technologies and work-flow models, which includes 140 stories (1031 knowledge cards) submitted by 58 class members. In the second term, "Community-ware and groupware design (2003)" and "Community-ware and groupware design - Paper" were added to this community. "Community-ware and groupware design (2003)" includes an orientation story for the class members and a story about psychological issues of group behavior. "Community-ware and groupware design - Paper" is the channel for papers about agent technologies, awareness and community support systems, which includes 94 stories (988 knowledge cards) submitted by 43 class members.

Each class member in 2003 has a personal program table that integrates and visualizes three channels; "Personal channel" including personal secret content, "Public channel" including public open content and "Community-ware groupware channel" including his papers. The professor has a personal program table and a program table for grading. In the grading table, "Community-ware groupware channel" of the whole class members are arranged. The professor and the class members shared one program table called "'Community-ware and groupware' program table" that integrated and visualized four channels; "Knowledge communication (2002)", "Knowledge communication -

Paper", "Community-ware and groupware design (2003)" and "Community-ware and groupware design - Paper". "Community-ware and groupware design - Paper" is a meta-channel that enables people to view plural channels as one channel by integrating 43 "Community-ware groupware channel"s.

"RISTEX community" consists of researchers in RISTEX Mission-Oriented Research Program I [4] that aims at creating new technological system in order to solve social problems related to safety and security. This community has been formed to open the research results to citizens from Dec. 2003. 14 research groups have their own channels and publish them on the public program tables. Program tables are distinguished by "accessStrategy" (public or members only) and language (Japanese or English). The public tables are opened to everyone for communicating widely while the private tables are closed for professional discussion. Most of stories are Japanese, and public stories are less than private stories because the content that is opened to unspecified people forces more attention to content creators than private content.

"Conversational informatics community" consists of researchers in the conversational informatics project (Apr. 2001 - Mar. 2006) aided by Japan Society for the Promotion of Science. This community was formed to discuss research issues and publish research results from Sep. 2003. This community has three symposium channels that virtualize real symposiums and workshops of the conversational informatics project. In "Conversational informatics channel", 2 days symposium on conversational informatics is roughly summarized in 15 stories (total play time is 4 hour) created by 15 presenters. And other channels include keynote lectures and an interim report of this project.

6 Discussion

These experiments suggest that the knowledge channel can manage huge content. The half-year lectures can be summarized into about 260 stories, and 2 days symposium can be summarized into about 15 stories. Students' papers are shared in the class. Moreover, all the content is accumulated in reusable form by using knowledge cards.

The knowledge card allows heterogeneous content. The lecture content is written in a formal style on the contrary the papers of class members include a casual style like a talk show. Fig.6(a) shows an example of the story that starts and ends with greeting. Fig.6(b) shows an example of the demonstrative text that complements the picture about workflow for a bank system. Each constituent part (title, body and image) is ambiguous, however the whole card makes a sense. Fig.6(c) shows an example of the stage effects. The image represents sunset that is not related to the body text directory, however it represents a change of topic by having a break.

The above results implies that the knowledge channel allows the management of huge heterogeneous content.

There are some related works in the personal content management area. Weblog systems (Rebecca, 2002) like Movable Type [5] and TypePad [6] support for a user to publish his personal journal by using a card like form called an "entry". They also support communication between users by the trackback function that enables users to link their entries. However, each entry is too independent to make a large content. On other hand, our knowledge cards can be organized into a long story. The knowledge channel differs in that the weblog supports text-based communication while the knowledge channel supports human-agent conversation.

7 Conclusion

This paper described the agent-based content management system that allows huge heterogeneous content. The knowledge card that can wrap up heterogeneous content has been proposed. The knowledge channel that is a model of agent-based content management system has been proposed and implemented by using virtualized-ego and dynamic program table. The empirical experiment

[4] http://www.ohriki.t.u-tokyo.ac.jp/S-Tech/M1/eng/index.html
[5] http://www.movabletype.org/
[6] http://www.typepad.com/

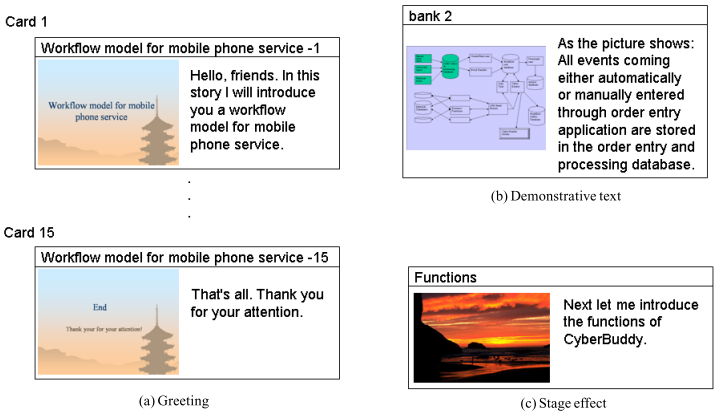


Figure 6: Examples of the heterogeneous knowledge card

conducted in three communities has implied that the knowledge channel allows the management of huge heterogeneous content.

References

Kubota, H., Kurohashi, S., Nishida, T.(2002). Virtualized-egos using Knowledge Cards. In Proceedings of Seventh Pacific Rim International Conference on Artificial Intelligence(PRICAI-02) WS-5 International Workshop on Intelligent Media Technology for Communicative Reality (IMTCR2002), pages 51-54.

Kiyota, Y., Kurohashi, S., Kido, F.(2002). "DialogNavigator": A Questions Answering System based on Large Text Knowledge Base, Proceedings of The 19th International Conference on Computational Linguistics (COLING 2002), pp.460-466.

Sadao Kurohashi and Makoto Nagao. KN Parser : Japanese Dependency/Case Structure Analyzer, In Proceedings of The International Workshop on Sharable Natural Language Resources, pp.48-55 (1994.8).

Sadao Kurohashi, Toshihisa Nakamura, Yuji Matsumoto and Makoto Nagao. Improvements of Japanese Morphological Analyzer JUMAN, In Proceedings of The International Workshop on Sharable Natural Language Resources, pp.22-28 (1994.8).

Tomohiro Fukuhara, Shintaro Azechi, Nobuhiko Fujihara, Ken'ichi Matsumura, Hidekazu Kubota and Toyoaki Nishida: Public Opinion Channel: Facilitating Community Knowledge Circulation, Workshop on Community Knowledge, ECSCW2001, Bonn Germany, September 2001.

Blood Rebecca. "THE WEBLOG HANDBOOK", PerseusPublishing, 2002.