Visualization for Sharing Knowledge in Creation Processes

Masaki Ishihara, Kazuo Misue and **Jiro Tanaka**
Department of Computer Science, University of Tsukuba
{ishihara, misue, jiro}@iplab.cs.tsukuba.ac.jp

Abstract

In creation processes, relationships between knowledge are also important as relational knowledge. We tried to represent such the relational knowledge of working processes visually by using Ripple Presentation, which is a technique of visualization that enables to express both the time-series and the categories as attributes associated with information. We propose extraction of the relational knowledge from the file-access history, and then we show how to visualize it by using the Ripple Presentation.

1 Introduction

It is important to understand the relationships between knowledge in creation activities. For example, there is various knowledge used as material in process of putting together product of a project. Most information treated in the business fields has been accumulated in the computers, so knowledge is managed as files in the computers in many cases. Then the only person who create products in a project understand the relationship (when, who, how use it) between the files associated to this product.

In this case, it is important to preserve this kind of knowledge like file relationships, in order to be able to share this knowledge with a new people who could join the work group. Although it is relatively easy to clone a *result file* (product such as report, thesis and so on.) and *association files* with the *result file* (materials such as memos, references and so on.), it is difficult to share the relationships of these files with others, because most of these relationships are not formalized.

The purpose of our research is to support sharing knowledge of the work process with co-workers. Here, "knowledge of the work processes" suggests the relationships between the *result file* and the *association files* when creative activities are accomplished. In order to support sharing knowledge of the work process, we developed a technique for expressing the relationships between these files by "Ripple Presentation".

2 Ripple Presentation and Layout Technique

2.1 Expressing both Time-series and Category

We focus two viewpoints which are time-series and categories in information visualization. In general, when information is classified and arranged, it is arranged by the degrees of freshness and classified by the category. The examples of this arrangement are file management systems and the topic list of news articles.

When this time series and the categories are expressed at the same time, a hierarchical expression of a tree structure (Figure 1) is often used. However, in the case of searching for information that belongs to certain time periods each category, this searching task takes a lot of trouble if volume of information increases. Especially, when information is added frequently, a new expression method suitable for history information is requested.

In this research, we invented an expression technique "Ripple Presentation" (Masaki Ishihara, Kazuo Misue and Jiro Tanaka, 2006), which was able to express history information by the time series and the categories simultaneously and the link relationships of information. The concept of the Ripples Presentation is "The old information is positioned far away and the new information is positioned near from the user's viewpoint." The name Ripple Presentation comes from the feature that it looks like the ripples extend. In addition, it expressed the history information as a graph in order to visualize for the relationships of information.

In the Ripple Presentation, the hierarchy of time-series is represented by concentric circles (ripples) centering on positions of the parent nodes of each subtree as shown in Figure 2. The cycle of ripples is constant. Assuming that the attribute time of the node was the same as the creation time of the ripple, the node would be positioned on the spreading ripple, because the ripples are synchronized with the nodes.

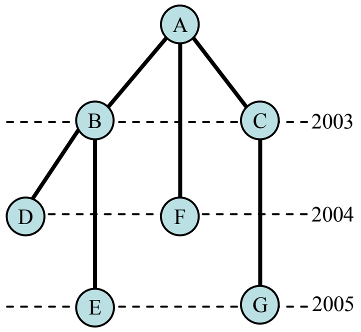


Figure 1. Time-series hierarchy expression of tree structures

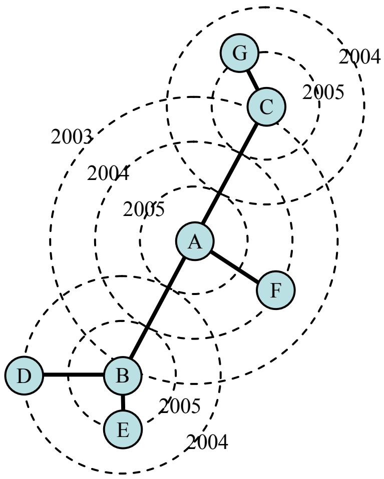


Figure 2. Ripple Presentation for tree structures

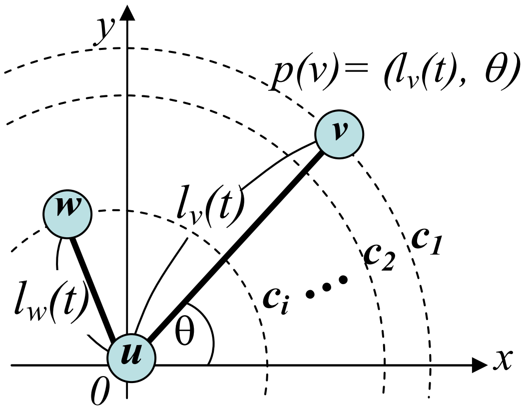


Figure 3. Polar representation of child nodes

2.2 Layout Technique of Ripple Presentation

Positions of nodes: In Ripple Presentation, the position of the node is expressed by polar coordinates as shown in Figure 3. That is, the position of each node is decided depending on the length and the angle of the edge as the parameter. For example, the position $p(v)$ of child node v at time t is defined by polar representation $(l_v(t), \theta)$, when the length of edge is $l_v(t)$ and the angle is θ. That is, positions of all nodes are relatively defined when position of root node is defined.

Ripples: Subtree rooted for node having only leaf node as child node is described by G_{sub}. In addition, center location, where ripples are generated at regular time-interval, is the position of the parent node of a subtree G_{sub}. Moreover, the positions where the nodes and the ripples are drawn are synchronized mutually. Therefore, ripples start generating at the time point when child nodes are created. That is, assuming that the oldest node (the most outside node) in all nodes except for root node that composes a subtree is v. The creation time of node v is described with $d(v)$, and then ripples start generating at the time $t_0 = d(v)$.

The ripples not only are generated synchronizing with the child nodes but also are generated at a constant frequency. Therefore, the viewers can aware the elapsed time to accommodate the length of the edges by seeing the ripples.

Suppose that $\{c_1, c_2, \cdots, c_n\}$ is set of ripples centered on for a parent node of subtree G_{sub}. Then, number n of ripples (where it is drawn at time t) should fill the expression (1).

$$n = \left\lceil \frac{r(t)}{\lambda} \right\rceil \quad (1)$$

In the expression (3.1), the wavelength of ripples is λ (common parameter in all ripples).

At the time t, the radius $r_1(t)$ of the ripple c_1 (the most outside ripple) which generated at the time t_0 is defined in the expression (2). Frequency of the ripples is f.

$$r_1(t) = \lambda \cdot f \times (t - t_0) \quad (2)$$

Therefore, at the time t, the radius $r_i(t)$ of the ripple c_i (inside ripples from c_1) which generated after the time t_0 is defined in the expression (3).

$$r_i(t) = r_1(t) - \lambda(i-1) \quad (i = 1, 2, \cdots, n) \quad (3)$$

Length of edges: At the time t, the length $l_w(t)$ of edge from parent node $u \in V$ to child node $w \in V$ is defined in the expression (4).

$$l_w(t) = r_1(t) - \lambda(j-1) \quad (j = \lceil f \times (d(w) - t_0) \rceil) \tag{4}$$

An important point to emphasize is the expression (4) shows that lengths of each edge synchronize with the radius of ripples depending on frequency f of ripples.

Angles of edges: The angles of the edges is defined the relationship between the parent nodes and each child nodes. Parameter used at this time for drawing is category attributes of the nodes.

For instance, assuming that the content of each node is a news article, its categories such as "Society," "Politics," and "Economy" could be obtained by the content of the article. The angle of each edge which has child nodes with the same category is set within a certain constant range of the angle.

3 Application to Sharing Knowledge

In this paper, sharing knowledge is to share information of the work process (knowledge) with other co-workers; what file is made and in what order worker used. In order to support sharing knowledge, we visualize an access history of the files. In particular, we apply the Ripple Presentation as a technique of visualization for when, what and how files associated with the result file are used.

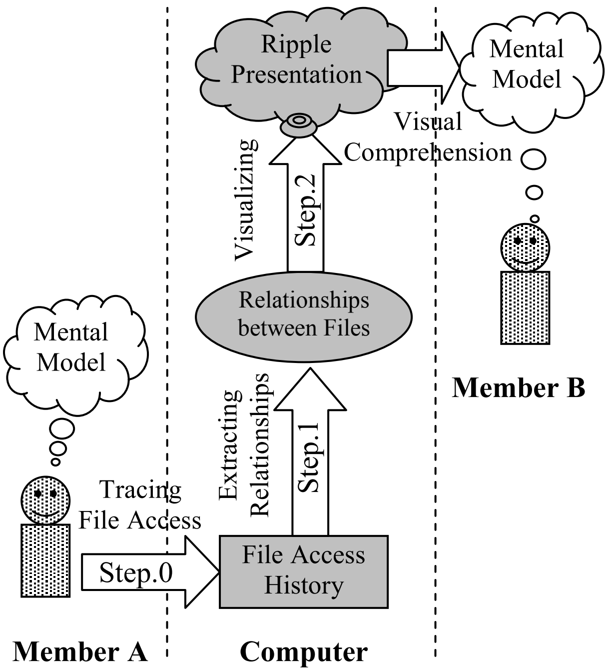


Figure 4. Sharing knowledge in creation process by visualizing relationships between files

3.1 Sharing Knowledge in the Creation Process

For example, it supposes that the files made by Member A in a project are shared with a new Member B. At this time, these files are able to copy onto PC of Member B easily, however the relationships of these files are not able to copy. These relationships are maintained only in the head of the person (Member A) who has worked by using those files.

In order to share knowledge of relationships between these files with members, two steps shown in Figure 4 should be supported by the system side. Each step is described in detail as follows.

Step.0: Tracing files access

Member A uses various files in his creation process. Therefore, the process of file creating as the product of work consists of the file access history. The network (mental model) associated with materials (*association files*) and a product (*result file*) is constructed in Member A's head.

In the working process, this mental model is not positively externalized in documents by the file creator, and after the product is completed, Member A usually write out (externalize) his simplified the mental model as a reference.

In fact, an actual work process is a trial and error process, there are a lot of key files as pieces of ideas, but unnecessary to record. So this process based on interaction of the *association files* is indispensable knowledge (tacit knowledge) when the worker creates the product.

Then, in order to externalization of this knowledge, a framework which traces and records the file access history as a log is necessary. Here, the file access history indicates history information that starts each suitable application to use or to view the file. Therefore, the name, the type (the extension), the location of each file and a time-stamp such as opened and closed time of the application associated with the file are stored in the file access history. The below section are described based on the assumption that the file access history be able to extract from the computer.

Step.1: Extracting relationships between files

It is natural that unrelated files with the product on each work process are actually included in the file access history recorded in the computer. For instance, a worker is often doing multitask in parallel on a computer, for this reason it is a necessity for clearly dividing the dependence of the file access history in each work process. Then, it pay attention on the product as *result file* in the work process of Member A, and the *association files* with this product is extracted from the file access history.

Concretely, these *association files* with the product indicate the files which have the relationships of simultaneous utilization with the *result file*. The link relationships between focused the *result file* and the *association files* are extracted from the file access history on the basis of a follow rule. If the files extracted as the *association files* are focused as the *result file*, the *association files* would be additionally extracted in the same way. That is, the relationships are extracted recursively, and then the link relationships finally become the tree structure having the *result file* as a root.

Step.2: Visualizing for relationships between files

Even if the relationships between the files are extracted, Member B must being presented an easily comprehensible expression of the relationships for understanding the work process of Member A. The easily comprehensible expression for Member B is an expression that can visualize a use situation of the *result file* and the *association files* in the work process, such as the name, the date and the period of files used by Member A.

We apply the Ripple Presentation to visualize the relationships between these files. The Ripple Presentation is a technique of visualization that enables to organize the link relationships from the aspects of categories and time-stamps of a target data.

3.2 Ripple Presentation for File Access History

It is assumed that an access history of six files such as File A, B, C, D, E, and F was traced in step.0 foregoing section, and describes below how to process Step.1 and 2 concretely in detail.

First of all, the file access history from April 24 to May 3 was shown by the time lines in Figure 6. Assumptions that File A is the *result file* in the work process, and then it should pay attention a period of time used File A (the shaded area in Figure 6). The files used at the same time with File A are File B, C, E, and F. These files are assumed the *association files* with File A. In addition, the files that does not overlap with the *association files* of File A and used at the same time with the *association files* of File A are File F and D.

Then, suppose that File A as the *result file* is a root and the *association files* are nodes, and then the relationships between these files is expressible as shown in Figure 7. The relationships between the *association files* with File A make up a tree structure. However, information presented by Figure 7 is only the link relationships of files. There is no essential information about the time series; how much period, when and in what order the files used.

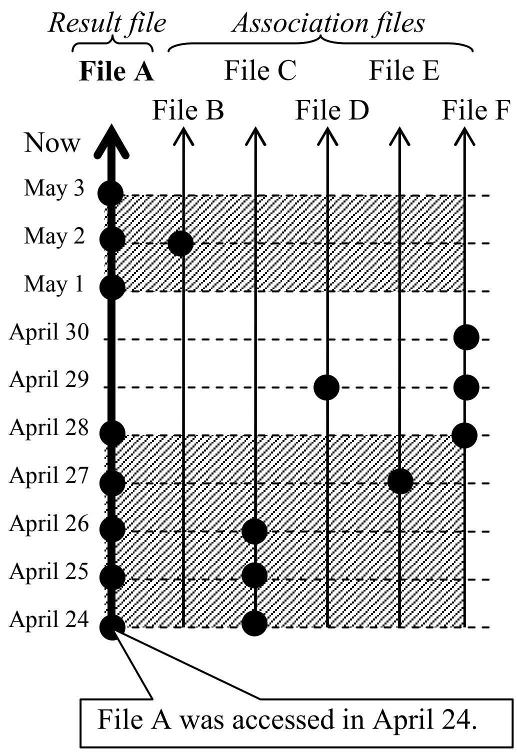


Figure 6. Time line of file access history

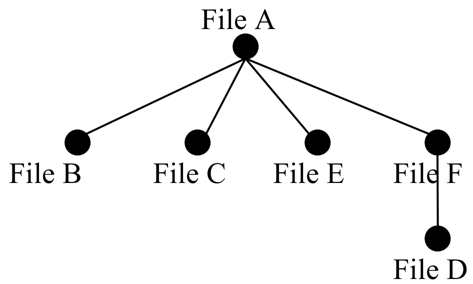


Figure 7. The relationships of *association files* with File A

For a solution of these problems, we apply the Ripple Presentation (Figure 8) which expresses both the time series of information like Figure 6 and the relationships of information like Figure 7.

In this, a rank at the using period of the *association files* with the *result file* is calculated and an angle is allocated based on this rank. In this case, each file is arranged at intervals of 60 degrees because the *association files* with File A appear six times at all periods (total of using period of File A). In addition, the rank are calculated based on the length of the access time, that is the high rank files are used much time. And then, the angle is allocated so that the file with the highest rank is arranged upside, the file with the lowest rank is arranged downside oppositely.

In total work processes (period) of Member A, the change along the time line, access time and relationships of each file become visually comprehensible by the Ripple Presentation.

3.3 Scenario

It is assumed that Member A created File A as an interim report in a project. We think about a scene that succeeds work to Member B as a new project member. Member A copied Member B File B, C, D, E, and F as the project materials associated with the interim report (File A). However, Member B does not understand "how" and "when" these files were used. Therefore, Member B wants to know how Member A used these files in the project. Unfortunately there was no concrete description (e.g. reference) about work process in the content of File A.

Then, it is thought that Member B is able to understand the work of Member A if it is possible to see in order using the *association files* with File A as the *result file* centered in the work process on the project. Therefore, Member B decided to visualize the file access history centered on File A by the Ripple Presentation.

First of all, Member B drags File A from file management system (e.g. Windows Explore) to Ripple Presentation viewer and drop it. The history information of File A is visualized as Rip the Ripple Presentation centering around (rooted) the drag-and-drop file (Figure 8).

How *association files* were used? : On Ripple Presentation viewer, the user be able to modify time interval of ripples. Suppose, for instance, the time interval of ripples is 1 week, the view is Figure 9(a). The *association files* which the access time is comparatively long are arranged within the upper area (the shadow area in Figure 9(a)) in a view of File A. It follows that File C and F (the files enclosed with the solid line in Figure 9(a)) are mainly used with File A by Member A. In other words, File C and F mean that degree of association with File A are strong. Therefore, there is a high possibility that the content of File A is reflected from that of File C and F. So, Member B find out that File C and F are essential for the effective understanding of the creation process of File A. In addition, Member B also find out that File D is indirectly associated with File A, because File D (the file enclosed with the dotted line in Figure 9(a)) is the *association file* with File F.

In the Ripple Presentation, the visualization for degree of association between File A and the *association files* enables Member B to understand how the *association files* were used by Member A.

When *association files* were used? : In order to check the *association files* in more detail in the week until April 24 – April 30, suppose that the time interval of ripples is modified 1 day. And then, the view becomes Figure 9(b). That is, the shadow area in Figure 9(b) means the ripples where the most outside ripple in Figure 9(a) was expanded.

In the Ripple Presentation, the length of the edge corresponds to the elapsed time from creation-time of edge to time of the view (present). The files far from the position of FileA mean ones used in the initial creation process of File A, and the files near the position of File A mean ones used in the final creation process of File A. Therefore, by the distance from File A, Member B is able to understand visually when the *association files* were used.

The point is the visualization of the file access history by the Ripple Presentation helps the user to understand on the two viewpoints; the link relationship (degree of association and structures) and the order relation (time-series) between the *result file* and the *association files*. As a result, Member B is able to share smoothly the knowledge in the creation process of the *result file* of the project with Member A.

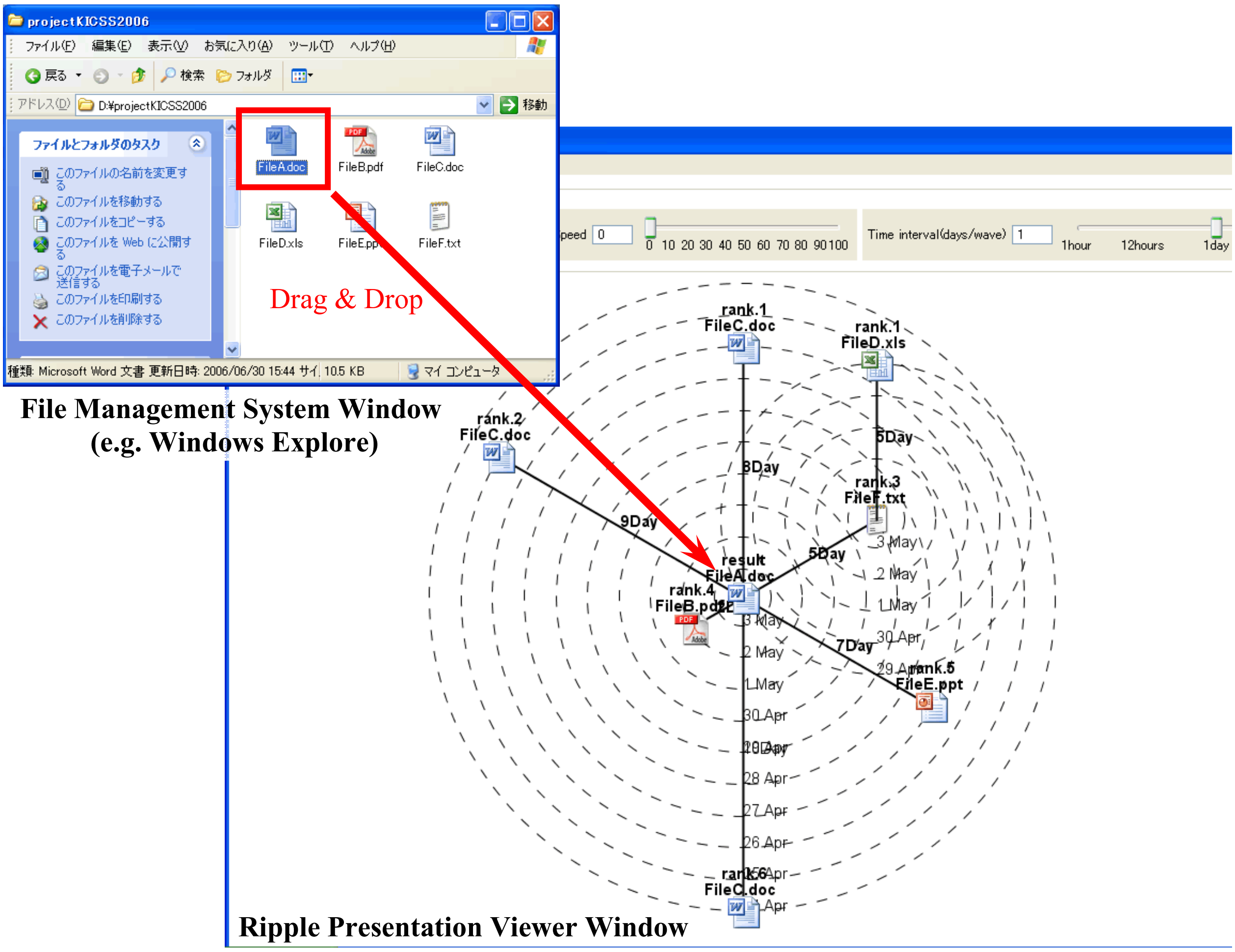
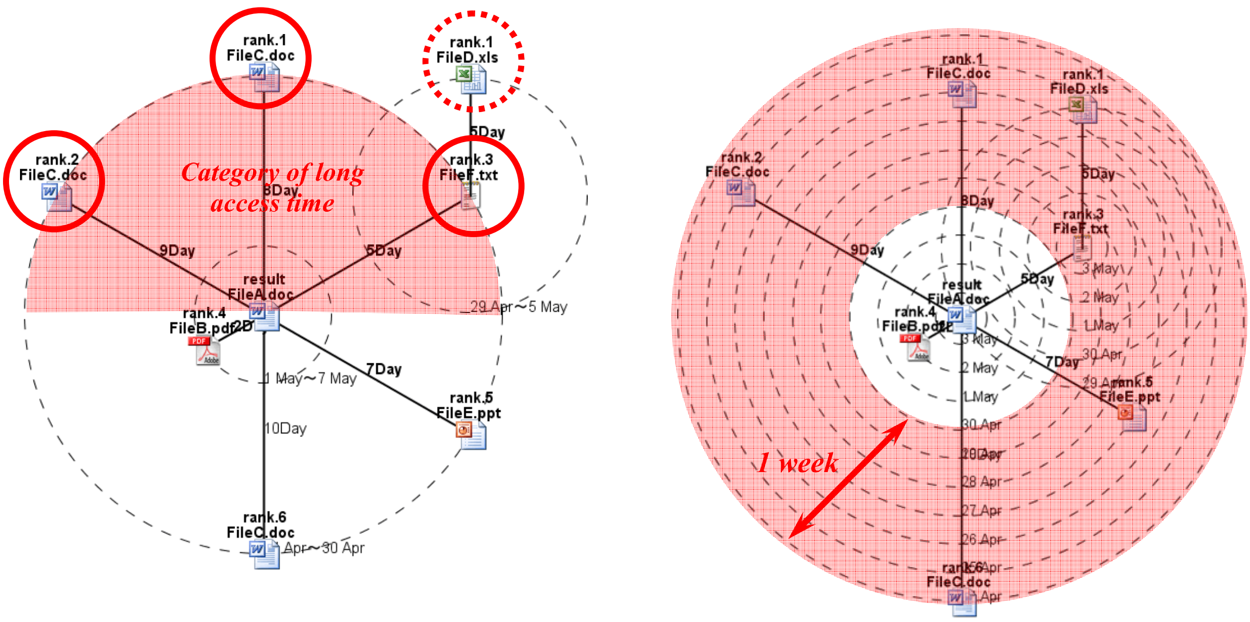


Figure 8. Ripple Presentation of file access history



(a)Time interval of ripples: 1 week. **(b)Time interval of ripples: 1 day.**

Figure 9. Difference of view by setting at time interval of ripples

4 Discussions

In this paper, it was described to be able to share knowledge in the creation process among multi-user by visualization of relationships between the files by the Ripple Presentation. There are two points that should be discussed.

First point is extracting the relationships between the files from the file access history in the viewpoint of the user's simultaneous utilization. Strictly, there is a possibility that not only the file in the computer but also information from printed documents and daily life affect the products. Such offline information not recorded by the computer is not able to be presented by the Ripple Presentation in the technique of this paper. It will be necessary to innovate the framework that can be recorded as the history information of the creation process in the computer (online) for the reference information in off-line as our future work.

Secondarily, it is necessary to evaluate whether the Ripple Presentation is able to help constructing a user's mental model. We insist that immediate visual comprehension of overall relationships between the files by the Ripple Presentation is useful for organizing the mental model. We think that there are the following three important common perspectives on human recognition of information.

1. Relationships of information

2. Time series of information

3. Categories of information

For instance, when the user recalls the file, he think about this three point comprehensively; with what file used it (relationships), when used it (time series) and what positioning was it (categories).

The feature of the Ripple Presentation is expressible of such three perspectives by one figure. Concretely speaking, when the user wants to grasp the relationships of information, all he has to do is paying attention to the basic information, as a result associated information become comprehensible by positional relationships from the viewpoint of user. That is, the effectiveness of the Ripple Presentation is that the categories and relationships of history information are understood in the sight according to the time series (Figure 10). Therefore, it is thought that the sharing knowledge can be smoothly done by the Ripple Presentation that is the immediate visually comprehensible expression by the user.

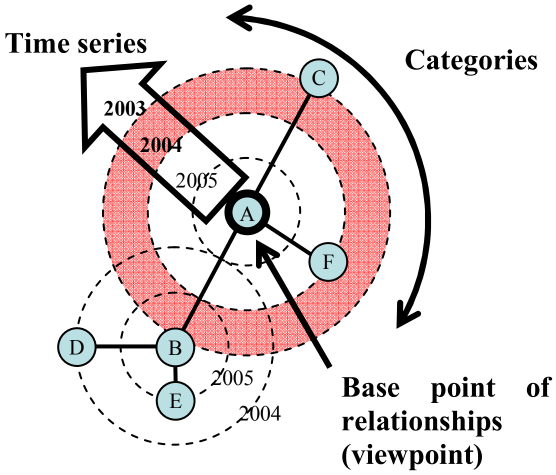


Figure 10. Perspective that can be understood in sight from Ripple Presentation

5 Related Works

The relation researches on the expression technique and on the knowledge management support by information visualization are described as follows.

An expression for history information with the tree structure was generally a hierarchical expression and a list expression. To express relationships of information, information visualization devising graph layout has been also actively researched (John Lamping and Ramana Rao, 1996; Brian Johnson and Ben Shneiderman, 1991; Jun Rekimoto and Mark Green, 1993). In case of expression for some attributes of information according to time series, the expression are often applied the layout technique approaches such as a circular disposition and a spiral arrangement (Ka-Ping Yee, Danyel Fisher, Rachna Dhamija and Marti Hearst, 2001; John V. Carlis and Joseph A. Konstan, 1998). Though the Ripple Presentation looks like a radial layout (Ka-Ping Yee, Danyel Fisher, Rachna Dhamija and Marti Hearst, 2001) that arranges information in the circle according to hierarchy of tree, the Ripple Presentation is different in the point to be able to express a time change of link relationships by the recursive concentric circle arrangement.

On the other hand, in the scene of the knowledge management, a graphic representation is often used to make the relationships between information easy to understand, such as a work schedule of a project, a concept chart, and a class diagram of the program etc. In particular, the relationships of information are often expressed by a two dimension map. There are the Mind Map (Tony Buzan, 1991), the Knowledge Map (John L. Gordon, 2000), the Concept Map (John W. Coffey, Robert R. Hoffman, Alberto J. Cañas and Kenneth M. Ford, 2002), and the Keygraph (Yukio Ohsawa, Nels E. Benson and Masahiko Yachida, 1998) as researches of the knowledge management by such the two dimension map. These researches visualize the relationships between information as a network diagram, and support data mining that the user is able to discover the useful information.

Moreover, Takanori Ugai, Terunobu Kume and Kazuo Misue (2005) focused the files used in the creation process in the computer. This research visualizes the relationships between a final product and various intermediate products in a project of software development as the two dimension map. We also focus the product as the base file and the *association files* with the base file. The point of our approach is visualization of the file access history according to the time-stamps by the Ripple Presentation, in order to share knowledge of the creation process between multi-users.

6 Summary

We described the Ripple Presentation that enables to express both the time-series and the categories of information is able to apply for the sharing knowledge. Visualization for the file access history by the Ripple Presentation externalizes the use time, the category, and the relationships between the files in the work process of users.

References

Brian Johnson and Ben Shneiderman. 1991. Tree-Maps: A Space Filling Approach to the Visualization of Hierarchical Information Space. Proceedings of the IEEE Visualization '91: 275-282.

John Lamping and Ramana Rao. 1996. The Hyperbolic Browser: A Focus+context Technique for Visualizing Large Hierarchies. *Journal of Visual Languages and Computing*, 7(1): 33-55.

John L. Gordon. 2000. Creating knowledge maps by exploiting dependent relationships. Knowledge-Based Systems, 13: 71-79.

John W. Coffey, Robert R. Hoffman, Alberto J. Cañas and Kenneth M. Ford. 2002. A Concept Map-Based Knowledge Modeling Approach to Expert Knowledge Sharing. Proceedings of IASTED International Conference on Information and Knowledge Sharing (IKS'02): 212-217.

John V. Carlis and Joseph A. Konstan. 1998. Interactive Visualization of Serial Periodic Data. Proceedings of Symposium on User Interface Software and Technology 1998 (UIST'98): 29-38.

Jun Rekimoto and Mark Green. 1993. The Information Cube: Using Transparency in 3D Information Visualization. Proceedings of the Third Annual Workshop on Information Technologies & Systems (WITS'93): 125-132.

Ka-Ping Yee, Danyel Fisher, Rachna Dhamija and Marti Hearst. 2001. Animated Exploration of Dynamic Graphs with Radial Layout. Proceedings of the IEEE Symposium on Information Visualization 2001 (INFOVIS'01): 43.

Masaki Ishihara, Kazuo Misue and Jiro Tanaka. 2006. Ripple Presentation for Tree Structures with Historical Information. Proceedings of Asia Pacific Symposium on Information Visualization 2006 (APVIS2006): 153-160.

Takanori Ugai, Terunobu Kume and Kazuo Misue. 2005. An Interface of Multi-viewpoint Maps for Re-using Software Products. Proceedings of HCI International 2005.

Tony Buzan 1991. The Mind Map Book. New York, Penguin.

Yukio Ohsawa, Nels E. Benson and Masahiko Yachida. 1998. KeyGraph: Automatic Indexing by Co-occurrence Graph based on Building Construction Metaphor. Proceedings of Advanced Digital Library Conference (IEEE ADL'98): 12-18.