Metadata-Driven Personal Knowledge Publishing

Ikki Ohmukai[1], Hideaki Takeda[2], Masahiro Hamasaki[1], Kosuke Numa[1], and Shin Adachi[3]

[1] Graduate University for Advanced Studies,
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo, Japan,
i2k@grad.nii.ac.jp,
http://www-kasm.nii.ac.jp/~i2k/
[2] National Institute of Informatics,
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo, Japan
[3] Waseda University,
3-4-1 Okubo, Shinjuku-ku, Tokyo, Japan

Abstract. We propose a personal knowledge publishing system called *Semblog* is realized with integration of Semantic Web techniques and We-blog tools. Semblog suite provides an integrated environment for gath-ering, authoring, publishing, and making human relationship seamlessly to enable people to exchange information and knowledge with easy and casual fashion. We use a lightweight metadata format like RSS to ac-tivate the information flow and its activities. We define three level of interest of information gathering and publishing i.e., "check", "clip" and "post" and provide suitable ways to distribute information depending on the interest level. Our system called Semblog platform consists of two types of extended content aggregator and information retrieval / rec-ommendation applications. We also design a new metadata module to define personal ontology that realizes semantic relations among people and Weblog sites.

1 Introduction

We propose a personal knowledge publishing system called *Semblog* is realized with integration of Semantic Web techniques and Weblog tools. Semblog suite provides an integrated environment for gathering, authoring, publishing, and making human relationship seamlessly to enable people to exchange information and knowledge with easy and casual fashion.

We are aiming to build systems that support not only human information but also human communication activities. We show the scheme of human infor-mation and communication activities in Fig.1. It is an extension of "Activities and Relationships Table" proposed by Shneiderman[1]. The first layer has three elements that concern information handling, i.e., collect, create and donate in-formation. Three activities form a cycle, i.e., they are invoked repeatedly. The second layer has also three elements that concerns communication handling, i.e.,
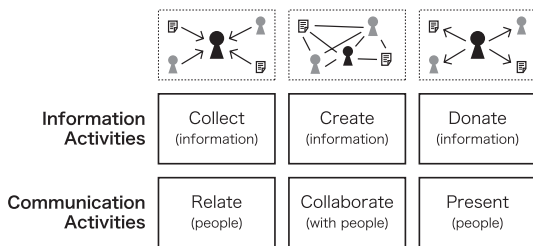
Fig. 1. Information and Communication Activities

relate, collaborate and present people. They also form a cycle. The two layers are not isolated rather closely related. Organization of information in the first layer is performed by the communication process of multiple people in the second layer. It implies that support for information activities requires support for communication activities, and vice versa.

We set Weblog as a platform for systems to support human information and communalization activities, because Weblog supports more these activities than Web. Web only supports publishing of information, and other information activities are left to other services like HTML editors and search engines. Weblog directly supports authoring and publishing in an integrated way, and furthermore it commits communication activities. Weblog users tend to refer to each other and form so-called "Weblog communities", and Weblog tools support such communication like TrackBack and ping. But the support is partially, indirectly, and not integrated.

Semblog suites extend Weblogs by adding flexible but uniform operations for Weblog sites and entries like aggregation and clipping, and facilities for searching and contacting to other Weblog sites. It means that Semblog suite supports communication activities as well as information activities.

The organization of the paper is as follows; in the following section, we discuss relationship between Weblog and Semantic Web. Then we identify architecture of Semblog in Section 4. We describe details of Semblog suites, first platforms of Semblog and then applications of Semblog in Section 5 and 6 respectively. We furthermore introduce ontology in Semblog platform in Section 6. Finally we conclude the paper in Section 7.

2   Semantic Web and Weblog

In this section, we overview the current situation of information distribution on Web, Semantic Web and Weblog. Web lacks functionality of information distribution and Semantic Web aims to fulfill the functionality by metadata. We support Semantic Web approach but it has difficulty of metadata annotation. So we focus on Weblog that also use metadata for information distribution.

2.1   Information Processing with Semantic Web Techniques

We propose a content distribution support system for individuals with Semantic Web techniques. Information distribution process does not mean just publishing but an integrated process containing information gathering and authoring. In the current web, however, there is no framework to support the whole process of information distribution despite the fact that Berners-Lee specified the first world wide web to support both authoring and publishing process equally[2]. Protrusion of information publishing just accelerates so-called information overload.

There is a great hope that the Semantic Web technologies will resolve information overload. According to the manifesto[3], Semantic Web is an environment, which consists of the contents with machine-readable (semantic) tags and the software agents, to realize autonomous information distribution and syndication. Resource Description Framework (RDF)[4] and other ontology definition languages[5] are recommended by W3C as elemental technologies of the Semantic Web and these are now in practical use.

However it is difficult to produce contents with semantic tags because of their complicated syntax and vocabulary. Ordinary people hardly find a merit of semantic annotation although it is a time-consuming task. It is also impossible to annotate the semantic tags to existing enormous information on the Internet. There are some researches about automatic annotation with AI techniques and natural language processing[6] however their effects are still unclear.

In our approach, we use lightweight metadata formats, i.e., RSS 1.0 (RDF Site Summary[7]), to activate the information flow and its activities. RSS is one of the metadata to describe a summary of a web site. It contains general attributes of the site i.e. title and publisher's name of the web site, and excerpt and updated date of its contents. A number of web sites already publish RSS metadata, and several applications and services called RSS aggregator are provided based on this trend.

An aggregator collects these RSS from various web sites and reform them to organize this large amount of contents to show at a glance. There are two types of aggregator; one is a standalone application that is executed on client PCs. The other is an aggregation service that runs on the internet server and the user access via her/his web browser. The former applications provide rapid browsing of RSS by their flexible user interface and the latter enables the user to access their information wherever she/he is.

Use of RSS and the aggregator decreases information gathering, but it is just a part of information distribution process as we mentioned. Gathering should be related to information authoring and publishing otherwise it will be another search engines without any selection or extraction.

2.2   Information Creation with Weblog

Weblog is one of the most advanced systems that use metadata in gathering, creating and publishing.

Recently Weblog has come into the spotlight in the Web[8]. There is no strict definition about Weblog but it is recognized as a web site that consists of miscellaneous notes updated daily[9]. In such sites the authors do not make efforts to knit up these contents because Weblog tools align them in chronological order with a well-designed HTML format. We call these frequently posted contents as small contents in this paper. Small contents include various subjects including journal, expertise and critique. One of most popular topics is the introductions and comments of the web sites ranging from news sites to the other small contents. Some Weblog sites attract the attention with their own editorial policy. The authors of Weblog sites reedit the existing web contents by quoting them. Moreover there are new types of Weblogs that criticize the other Weblogs so that these Weblogs are regarded to organize the "Weblog community". Now there are millions of weblog sites in the World. It is a surprising number because these people are now active information creators and distributors as well as information receivers thanks to weblog.

Most of Weblog sites use so-called Weblog tools that are a kind of content management systems (CMS). Weblog tools enable the author to describe and edit the small contents via a web browser and transform the contents form text format to HTML files. These tools are implemented based on MVC (Model / View / Controller) model which is the fundamental concept of web applications. The author defines a view template once then do not have to decorate the contents with various HTML tags. This model decreases the cost of publication remarkably in comparison with traditional style that requires local text editor and FTP. This feature contributes abundant production of the small contents.

Weblog tools usually generate RSS automatically. General attributes such as publisher's name are set as a profile by the user. Excerpt and updated date of each content are generated by the tools. Most of distributed RSS are generated by these Weblog tools. Main purpose of RSS aggregator is also to browse the Weblog site. Currently the number of news site feeding RSS is expanding.

3   Semblog Architecture

We propose a personal publishing system using Semantic Web techniques and Weblog tools. The system supports the whole process of information distribution which includes gathering, authoring and publishing. Furthermore it connects information distribution process of different people seamlessly, i.e., it also supports finding collaborating and advertising people for information distribution.

We divide system architecture into the following four levels, i.e., metadata format, metadata management, metadata aggregator, and metadata application. In Weblog, they correspond to RSS, Weblog tools, RSS aggregator, and various applications on Weblog respectively. In Semblog, Metadata on person and interpersonal relations is added in order to include activities on the communication levels. We adopt FOAF (Friend-Of-A-Friend) as person metadata. Content (RSS) and person metadata should be processed seamlessly in every level.
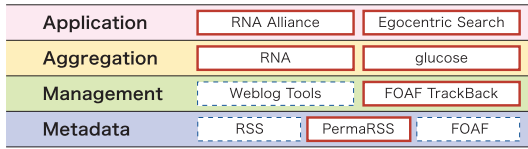
Application	RNA Alliance	Egocentric Search
Aggregation	RNA	glucose
Management	Weblog Tools	FOAF TrackBack
Metadata	RSS PermaRSS	FOAF

Fig. 2. Semblog Architecture

We build our systems by integrating new technologies and existing technologies shown in Fig.2. A box with solid line indicates our proposal and a box with dashed line an existing one.

In the following two sections, we explain platforms and application of Semblog. The former corresponds the first three levels and the latter the last level.

4   Semblog Platform

Fig.3 shows the system architecture of Semblog platform. Semblog system consists of the two types of extended RSS aggregator, semblog applications and conventional Weblog tools. Each module exchanges data in RSS format and communicates with XML-RPC protocol[10] for dynamic invocation. We use Movable Type system as Weblog tool[11], which supports RSS and XML-RPC call. In this section, we explain extended RSS aggregators that support authoring and publishing with different levels of interest.

4.1   Different Levels of Interest

We define the level of interest of information gathering and publishing i.e., "check", "clip" and "post". Our system provides different ways to distribute information depend on the interest level.

"Check" means that the user routinely browses particular web sites and information sources. The user does not know "what" is described at those sites exactly but she/he knows "what kind" of contents are included in these sites. We assume that these "what kind" knowledge are important for information distribution so that our system supports the user to present her/his interest by publishing the list of URIs she/he always accesses.

"Clip" makes shortcut to a content to which the user has strong interest among various contents of "Checked" sites. Our system automatically publishes the list of "clipped" contents. We suppose that "clip" link presents stronger interest of the user than "check" link because it points individual content directly. In addition, contents of "checked" links are changing momentary but those of "clipped" links are persistent (called "Permalink").

Online bookmark systems were developed for the same objective. These systems make a backup of "bookmark" or "favorite" of the user's web browser and she/he can obtain the list from anywhere. The user can also make the list to be public or not so the list will be a web content. However that bookmark is only
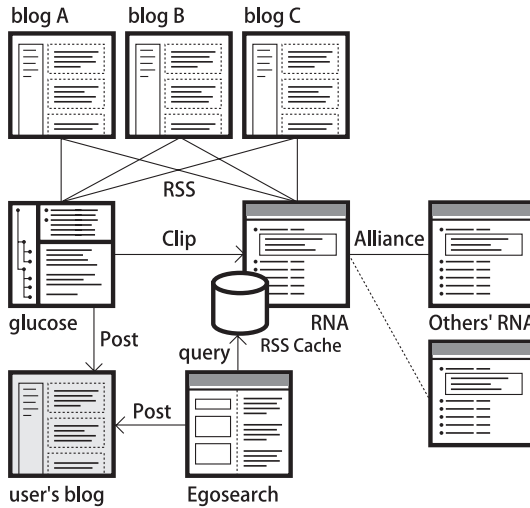
Fig. 3. System Architecture

a static list of URIs and is lacking functions for change in time. Therefore these online bookmark systems are not useful for everyone except the user.

Conventional bookmark systems do not distinguish the URIs between to a site and content. It is necessary to switch the way to distribute these links because they have different level of interest as described above.

"Post" means that the user quotes content, adds some comment, and publishes it as new information. In that case the user presents not only strong interest but also her/his personal opinion. In our system, "post" type of information publication is made by the Weblog tools.

Providing different levels for publishing reduces the psychological burden of publishing. In standard BBSs, it is said that lurkers show feeling of belonging to communities in sprite of their reluctant behavior[12]. This is important to involve ordinary people into the cycle of information gathering and publishing.

The all process is realized by metadata handling therefore it is open to other systems and applications. It enables seamless connection of information distribution of different people.

4.2   RNA: RSS Aggregation Service

RNA is an extended RSS aggregator described with Perl CGI. Fig.4(left) shows a snapshot of RNA. The user puts this script to her/his own web server and operates it. Basic function and interface of RNA are shown below.

– RSS registration and loading
  The user should register RSS in a configuration interface. Once she/he enters the URIs of RSS, RNA obtains these RSS files via HTTP. The user is able to
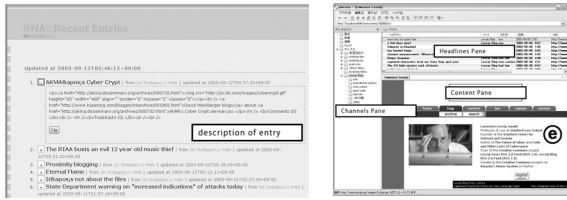
Fig. 4. Snapshot of RNA (left) and Glucose (right)

classify each RSS using categories. The list of registered sites is transformed into RSS and it can be used by the other applications. RNA can also import or export an OPML file which is standard format for the bookmarks.

– Building RSS tree

RNA parses multiple RSSs from various sites and builds a "global" RSS tree from each RSS tree. The global tree stores all information obtained by RNA. Next RNA recomposes the global tree into several sub trees according to rule templates. In default setting, RNA reproduces three types of sub trees i.e., list of contents sorted by chronological order and updated contents in each site. The user can describe own template script to generate a new sub tree.

– RSS redistribution

Generated sub trees are distributed as new RSS and are visualized by server / client side XSLT engine using XSL stylesheet. RNA can also transform these RSS into HTML with its own template system. With these templates, the user is easily able to customize an output style rather than XSLT since the syntax of template is similar to HTML.

– Content clipping

The user can register content to a clip list in one click. Clipped contents are stored in "clipped" RSS tree and it is published like the other RSS. Each element of RSS tree is changing momentary but those of "clipped" tree are persistent.

– Updating

It is necessary to get RSS and build trees occasionally since its contents are changeable with update of its distributor. RNA can update periodically by a Cron interface of the server. Update interface can be called both manually and remotely by XML-RPC message the Weblog tools automatically generated. The latter feature enables the author of a Weblog to notify her/his new content to RNA.

– TrackBack tracing

RNA also queries to registered Weblog tools whether its content has Track-Back links (reverse link provided by Weblog tools) or not. If exist, RNA extracts these links and adds them to RSS tree.

– Sanitizing and caching RSS

RNA checks syntax of acquired RSS and corrects them if they are not valid. Currently three versions of RSS are proposed i.e., 0.91, 1.0 and 2.0. RNA converts all versions into 1.0, which is based on RDF model. Every RSS is cached and decomposed to attach to each content as a fine-grained metadata.

"Antenna" services are already provided to check update of registered sites. Conventional antenna obtains HTTP LastModified header of each URI and sorts the site list by chronological order. Using RNA the user can get richer information than the antenna since RSS contains not only updated time but its excerpt and so on. RNA also enables the user to edit information in the contents level such as clip and TrackBack tracing.

4.3   Glucose: Standalone RSS Aggregator

Glucose is also an extended RSS aggregator but a standalone program for Windows. Fig.4(right) shows a snapshot of Glucose. Unlike orthodox aggregators, Glucose is developed to support information distribution process by coordination with RNA. Main functions and interfaces of Glucose are shown below.

- RSS registration
  Like in RNA, the user registers URIs of RSS or OPML site list. Glucose can access several news sites without RSS by "sensor" script which extracts articles and converts them to RSS.
- Three-pane interface
  Glucose has three panes interface. The left pane shows "RSS Channels" which is subscribed by the user. The upper right pane indicates the headline list of contents including title, updated time, source and category. The lower right pane shows original contents.
- TrackBack tracing
  Glucose can extract TrackBack links from each content. Obtained links are shown below the corresponding entry in headline pane with "Re:" just like a mailer.
- Posting to Weblog
  With a Weblog editor interface in Glucose, the user can post an entry to her/his Weblog if she/he has strong interest for content. This interface uses XML-RPC protocol.
- RNA clipping
  The user can post content to the clip list of own RNA using XML-RPC.
- P2P content recommendation
  Glucose runs as a P2P servant and the users automatically construct P2P network. In this network, several contents read by someone are distributed randomly and the other user may receive these contents as a recommendation. The user can clip or post it to her/his own system and can also register URI of its RSS to the site list.

We distribute RNA and Glucose in our web site (http://www.semblog.org/wiki/?en). About 2000 users downloaded RNA and over 20000 users downloaded Glucose from September 2003.

5   Application on Semblog Platform

Using functions on Semblog platform, we develop a new type of recommendation and retrieval systems.

5.1 RNA Alliance

Each RNA has XML-RPC interface that can send and receive its data dynamically. RNA alliance is a content recommendation system based on cooperation of multiple RNA.

We use FOAF metadata to identify each RNA. FOAF is RDF-based metadata format for describing human relationship. Besides the basic elements such as name, email and URI of the user, FOAF provides a statement that user X knows user Y.

The current version of RNA can generate FOAF data. RNA also has an interface for FOAF management to extend social network easily. We call this method as "FOAF TrackBack".

First the user X enters an RNA URI of the user Y in her/his own FOAF manager. The manager X asks the manager Y to acquire the FOAF data of Y, and writes "X knows Y" link in its FOAF. The manager Y records "Y isKnown X" link in its FOAF and notifies to the user Y. If the user Y agrees, her/his manager registers "Y knows X" link. Repeating this process, a personal network of the user is constructed. Following recommendation methods are performed in the network.

Collaborative Recommendation. Collaborative recommendation is based on difference of registered sites or clips among multiple RNAs. At first it calculates similarities: S_i between the user's RNA: R_0 and a RNA on the personal network: R_i $(1 < i < n)$. Each RNA has a list of URIs: $R_i = \{u_0, \dots, u_k\}$.

$$S_i = \frac{|R_0 \cap R_i|}{|R_0| + |R_i|}$$

The system gives recommendation score: $V(u)$ to each URI by the following formula:

$$V_i(u) = \begin{cases} S_i & \text{if } u \in R_i \\ 0 & \text{if } u \notin R_i \ (i = 1, \dots, n) \end{cases}$$

$$V(u) = \frac{\sum_{i=i}^{n} V_i(u)}{n}$$

This score is used for recommendation to R_0's user if URI u is not included in R_0.

The system shows the list of recommended URIs sorted by the score. The user can add these URI to her/his own "check" list.

Categorical Similarity and Recommendation. The user of RNA can categorize the sites and clips she/he registered. Using this category, relationship of *interest* among users can be identified and recommended. We apply a recommendation method based on a categorical similarity to this objective[13].
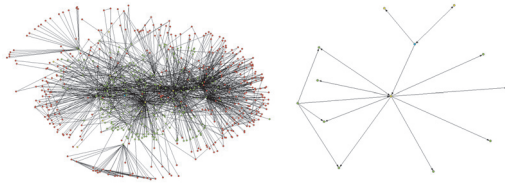
Fig. 5. Document-based (left) and Person-based (right) Network

Relational Filtering. Relational filtering method realizes access control of information using the categorized social network. By merging the personal network of every user, a large human network like a small world is constructed. This method extracts multiple communities from this network and enables information sharing in single community[14]. The user can manage her/his contents which have various level of disclosure through an unified interface.

5.2 Egocentric Search

Egocentric search provides subjective search which collects and evaluates information for each user[15]. We think that it is suitable for handling of the small contents like Weblog entries because connection is usually more subjective than other Web pages.

The users daily write and post contents to their Weblog sites with the editor interface. Egocentric search interface scans a content that the user is posting. If this content contains a hyperlink, the editor acquires the whole content and RSS of the link, and constructs an entry network around the user's content (Fig.5(left)). This network indicates not only relations of contents but also human relationships because all entries on Weblog are owned by some authors. Our method organizes a document network into person-based network (Fig.5(right)). Each path of the personal network is weighted relatively to the frequency of citation. Once the user cites some site as a topic in the new content (entry), the editor interface performs egocentric search and shows the result.

- Relative Chain Search
  Relative chain search returns the contents which is directly linked with the entry cited by the authoring content. This model is based on a simple model but consequently it seems most trustful. (Fig.6(a))
- Co-citation Search
  Co-citation search discovers the entries that link the same contents as the authoring entry links to. Co-citation entries are retrieved from the RSS cache and the search result contains the weight of authors. (Fig.6(b))
- Keyword Search
  Keyword search method picks up the entries by keyword matching from the RSS cache. Unlike the conventional search engines, our method targets only related sites around the user's Weblog. (Fig.6(c))
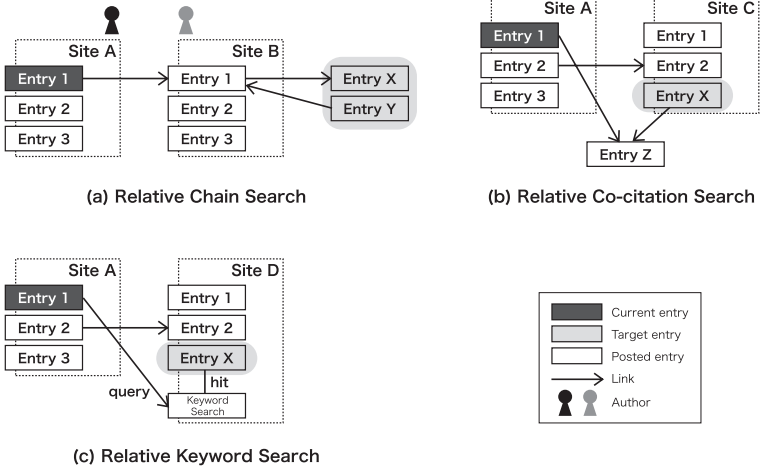
Fig. 6. Egocentric Search Methods

The user can read search results by these methods with the right-hand side window while authoring, and can append the link of some helpful contents to the currently authoring content. This process expects to enrich the user's content and change the search result of the system in turn.

6   Personal Ontology

We propose a bottom-up personal ontology framework using RSS and FOAF metadata. To process small contents in various forms, we have to annotate a semantic markup with an ontology language to those contents. It is difficult to organize practical ontology hierarchy with top-down approach because building and maintaining such well-organized large ontology takes a lot of efforts. We aim to develop loose and bottom-up ontology system by combining personal classification, because we consider that personal knowledge will be represented with a routine work such as categorization and arrangement of information. Fig.9 indicates a conceptual architecture of the personal ontology system.

At first we define a personal ontology as a hierarchical system of categories. Everyone has those categories, and they routinely classify described and collected contents to the category. A label of a category can be named arbitrarily by user.

Unlike the conventional ontology, the personal ontology has to be related to the person who produces it. Therefore we apply FOAF metadata to link between the ontology and the person.

Personal ontology metadata consists of FOAF, RDFS Ontology and Contents RSS. The FOAF describes personal information, and the RDFS ontology shows a structure of the categories, and the contents RSS shows written and collected contents by the user.
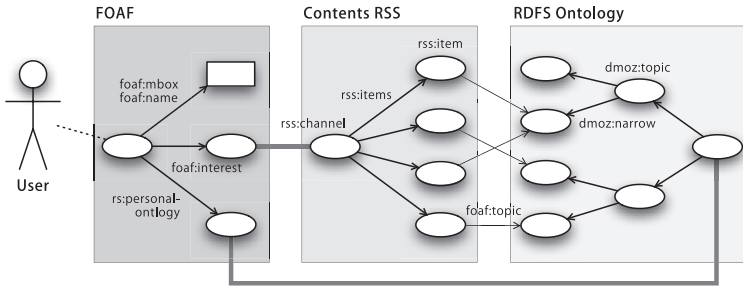
Fig. 7. Personal Ontology Framework

```
<rdf:RDF
 xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
 xmlns:foaf="http://xmlns.com/foaf/0.1/"
 xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
 xmlns:rs="http://www.roughsemantics.org/rs/0.1/"
>

<foaf:Person>
<foaf:name>Ikki Ohmukai</foaf:name>
<foaf:nick>i2k</foaf:nick>
<foaf:mbox rdf:resource="mailto:i2k@grad.nii.ac.jp" />
<foaf:weblog rdf:resource="http://www.semblog.org/i2k/" />
<rdfs:seeAlso rdf:resource="http://www-kasm.nii.ac.jp/~i2k/foaf.rdf" />
<foaf:interest rdf:resource="http://www-kasm.nii.ac.jp/~i2k/index.rdf" />
<rs:personalontology
           rdf:resource="http://www-kasm.nii.ac.jp/~i2k/ontology.rdf" />
....

</foaf:Person>
</rdf:RDF>
```

(a) Extended FOAF

```
<RDF xmlns:rdf="http://www.w3.org/TR/RDF/"
     xmlns:dc="http://purl.org/dc/elements/1.0/"
     xmlns="http://directory.mozilla.org/rdf">

<Topic rdf:id="Top">
  <tag catid="1"/>
  <dc:Title>Top</dc:Title>
  <narrow rdf:resource="Top/Arts"/>
  <narrow rdf:resource="Top/Business"/>
  <narrow rdf:resource="Top/Economy"/>
  <narrow rdf:resource="Top/Tech"/>
  ....
</Topic>

<Topic rdf:id="Top/Arts">
  <tag catid="2"/>
  <dc:Title>Top/Arts</dc:Title>
  <narrow rdf:resource="Top/Arts/Fine"/>
  ....
</Topic>
```

(b) RDFS Ontology

```
<item rdf:about="http://www.semblog.org/i2k/archives/000304.html">
<title>Blog Hacks</title>
<link>http://www.semblog.org/i2k/archives/000304.html</link>
<description>
Monday's child is fair of face, Tuesday's child is full of grace,
Wednesday's child is full of woe, Thursday's child has far to go,
Friday's child is loving and giving, Saturday's child works hard for his living,
And the child that is born on the Sabbath day is bonny and blithe, and good and gay. ...
</description>
<dc:subject>trivia</dc:subject>
<foaf:topic rdf:resource="http://www-kasm.nii.ac.jp/~i2k/ontology.rdf#Top/Arts">
<dc:creator>i2k</dc:creator>
<dc:date>2004-04-09T01:24:16+09:00</dc:date>
</item>
```

(c) Contents RSS

Fig. 8. Personal Ontology Metadata

We add two elements to basic FOAF model shown in Fig.8 (a). One is `<foaf:interest>` which is to point the contents RSS, and the other is `<rs:personalontology>` that is originally defined by our Rough Semantics project (`http://www.roughsemantics.org/`) to indicate the RDFS ontology.

The RDFS ontology is described with the form of Open Directory RDFS format shown in Fig.8 (b). Each node has a fragment ID.

The contents RSS is similar to a conventional RSS. Our RSS uses `<foaf:topic>` to point a category on the RDFS ontology, while the conven-
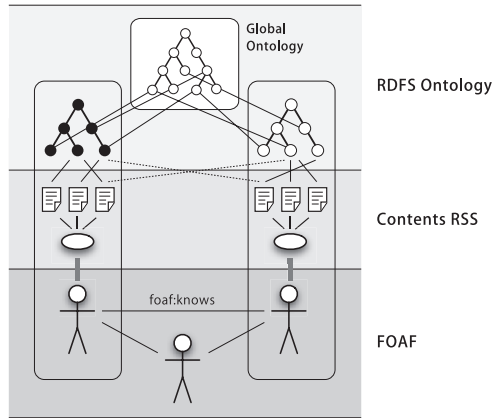
Fig. 9. Bottom-up Ontology

tional model applies `<dc:subject>` to express a thesis of a content. This makes our RSS to have backward compatibility. Example of this RSS is shown in Fig.8 (c). It should be noted that topics pointed by this tag are not restricted to those in their own ontology, rather any topics in others' and some global ontology. Separating ontology and instances enables such flexible management.

FOAF, RDFS ontology, and RSS are described in separate files so that we can keep compatibility with existing applications on these formats. This is a great benefit that our system can cope with such existing applications via these files.

Our framework enables applications and services to produce new types of search or recommendation. For example, mapping methods between two directories or bookmarks are applicable to the personal ontology. Egocentric search is also able to realize easily by building a social network with `<foaf:knows>` in the users' FOAF.

Unlike these peer-to-peer model, we can calculate a similarity among a personal ontology and the global ontologies such like WordNet and ODP in advance. Multiple personal ontology can be matched each other via the global ontology and this method needs less computation cost as shown in Fig.9. In addition, it is not necessary to modify that algorithm in P2P model and personal-global model because both ontology has the same structure.

7   Conclusion

In this paper we propose a personal publishing system with Semantic Web techniques and Weblog tools. We use a lightweight metadata format like RSS to activate the information flow and its activities. We define three level of interest of information gathering and publishing i.e., "check", "clip" and "post" and provide suitable ways to distribute information depending on the interest level.

Our system called Semblog platform consists of two types of extended content aggregator and information retrieval / recommendation applications.

Through these techniques and systems, we have shown that metadata can be used to realize more efferent and more personalized information distribution. Metadata design should be careful because it should be acceptable by many people and systems. Our approach, i.e., extending and integrating RSS and FOAF, is successful in this aspect since existing tools like Weblog tools are ready to use them. We hope that our approach will be a bridge between emerging Semantic Web technologies and other growing technologies on the Internet.

References

1. Shneiderman, B.: Leonardo's Laptop: Human Needs and the New Computing Technologies. MIT Press (2002)
2. Berners-Lee, T.: Weaving the Web. HarperCollins (1999)
3. Berners-Lee, T.: A roadmap to the Semantic Web. `http://www.w3.org/DesignIssues/Semantic.html` (1998)
4. World Wide Web Consortium (W3C): Resource Description Framework (RDF) Model and Syntax Specification. `http://www.w3.org/TR/REC-rdf-syntax` (1999)
5. World Wide Web Consortium (W3C): OWL Web Ontology Language Overview. `http://www.w3.org/TR/owl-features/` (2003)
6. Dill, S., Eiron, N., Gibson, D., et al.: SemTag and Seeker: Bootstrapping the Semantic Web via Automated Semantic Annotation. Proceedings of the Twelfth International World Wide Web Conference (WWW2003) (2003)
7. RDF Site Summary 1.0 Specification Working Group: RDF Site Summary (RSS) 1.0. `http://web.resource.org/rss/1.0/spec` (2001)
8. Blood, R.: We've Got Blog: How Weblogs are Changing Our Culture. Perseus Publishing (2002)
9. Aimeur, E., Brassard, G., Paquet, S.: Using Personal Knowledge Publishing to Facilitate Sharing Across Communities. Workshop on (Virtual) Community Informatics, Held in conjunction with the Twelfth International World Wide Web Conference (WWW2003) (2003)
10. UserLand Software: XML-RPC Specification. `http://www.xmlrpc.com/spec` (1999)
11. Six Apart: Movable Type. `http://www.movabletype.org/` (2003)
12. B.Nonnecke, J.Preece: Shedding light on Lurkers in Online Communities. Ethnographic Studies in Real and Virtual Environments: Inhabited Information Spaces and Connected Communities (1999) 123–128
13. M.Hamasaki, H.Takeda: Find better friends? – re-configuration of personal networks by the neighborhood matchmaker method –. In: The International Workshop on Semantic Web Foundations and Application Technologies (SWFAT). (2003) 73–76
14. I.Ohmukai, H.Takeda: Social Scheduler: A Proposal of Collaborative Personal Task Management. Proceedings of Web Intelligence (WI2003) (2003)
15. I.Ohmukai, K.Numa, H.Takeda: Egocentric Search Method for Authoring Support in Semantic Weblog. Workshop on Knowledge Markup and Semantic Annotation (Semannot2003), the Second International Conference on Knowledge Capture (K-CAP2003) (2003)