# Agents in Delivering Personalized Content Based on Semantic Metadata

**Teppo Kurki** **Sami Jokela** **Reijo Sulonen**

Helsinki University of Technology
Otakaari 1, Espoo, Finland
Tel: +358-9-4511
teppo.kurki@hut.fi, sami.jokela@hut.fi, reijo.sulonen@hut.fi

**Marko Turpeinen**

Alma Media Corp.
Eteläesplanadi 24, Helsinki, Finland
Tel: +358-9-50771
marko.turpeinen@almamedia.fi

### Abstract

In the SmartPush project professional editors add semantic metadata to information flow when the content is created. This metadata is used to filter the information flow to provide the end users with a personalized news service. Personalization and delivery process is modeled as software agents, to whom the user delegates the task of sifting through incoming information. The key components of the SmartPush architecture have been implemented, and the focus in the project is shifting towards a pilot implementation and testing the ideas in practice.

## Introduction

Internet and online distribution are changing the rules of professional publishing. Physical constraints of media products are not any more the key factors in the publishing process. In newspaper publishing, the bottleneck was what could fit in the paper. Today publishers are capable of creating more content, but they need new methods for personalized distribution. Incremental price for publishing on the Internet is negligible and the limiting factors are in producing the right content and delivering it to the right user.

Locating relevant information is time consuming and expensive. Customers need personalized information flow from trusted information sources. This service should adapt to customer's changing needs and interests.

The main goal of the SmartPush -project is to bring together and satisfy the needs of both the professional publishers and their customers. This is achieved by targeting and personalizing the information flow.

Content providers see metadata as a promising way to make their information better searchable and usable on the web. Currently news organizations deliver targeted information based on fixed and very limited metadata schemes. These specifications are often proprietary for one content provider. Sometimes providers agree upon a shared metadata scheme, like ANPA format (ANPA 1984), which is widely used in newswire content header description. In our approach, authors augment the information flow with rich metadata descriptions that are used in the delivery process.

Information filtering is usually based either on textual information flow or on some manual classification scheme embedded in the flow. Our system uses semantic metadata for personalization.

The customer should be able to delegate the filtering task to a software agent. The agent should possess enough machine intelligence to match changing customer needs with continuous information flow from the content providers. Agent autonomy promises flexibility in adding new functionality and changing the way the system works on the fly (Figure 1).


*Figure 1. Agents in content delivery.*

## Metadata-based Information Filtering

Textual information flow is normally analyzed using some variation of the term vector model (Salton and McGill 1983). Manual classification has typically been based on keywords or categories, which are used to make simple decisions with Boolean logic.

These approaches are fundamentally different. Classification system relies on human assistance, where an expert makes the decision about what categories the information covers. Term vector model based approaches treat information as is. With term vector models the decision process is more abstract and lends itself better to fully automatic personalization and delivery.

Our vision is that one can combine these two by using metadata for information filtering. This means that as the material is published, it is enriched with a semantic metadata representation of the content. The associated high quality metadata has enough expressive power to function as a surrogate for the entire content in information filtering.

Compared to simple keywords or categorization, the metadata should be more expressive and have wider coverage. Expressiveness facilitates more accurate personalization. By using the metadata for filtering we eliminate the need to transfer the full-length content, thus diminishing the required bandwidth.

The fundamental question in information filtering is whether or not a given user should be provided with a particular piece of information. This question can be answered in different stages of the content delivery process (Figure 2). Simple categorization done by the domain expert implicitly targets the content very early in the process. On the other hand, an agent armed with a user's knowledge base has to first fetch the information and then analyze it to decide if it should be given to the user.

If we think of an information creation and delivery process, we can use software agents in various stages of the process. The selection or matching that results in a personalized feed is one suitable task for an agent. The user delegates the task of sifting through the available information to the agent. The intelligence required for this task depends on the nature of information that the material has.

If the material is "dumb" ASCII text, the agent's task is hard - in principle the agent has to mimic exactly the user's thoughts in reading textual information. This task becomes an order of magnitude harder, if the information is not textual, but sound or video.

Structured documents contain more relevant information, since they carry information about the different components and their relations within the document. This way the agent has more information to work on. For example it can tell from the structure what parts of the documents contain heading information, which describes the content in a compact form.

To develop this idea of augmenting the information content to the extreme, the information flow could be tagged for each potential user already by the content provider. The agent's job would be just to route relevant information to the end users.

Somewhere between these extremes is the optimal distribution of intelligence between the agent and the information flow. If we can describe the content in a way that can express the most important aspects of the end user's interest profile, the job of the agent becomes considerably less demanding. It only has to compare the descriptions of new pieces of content against the user's interest profile, and make delivery decisions based on how well these two match each other.

This approach has naturally its drawbacks. Developing a model that can capture even the most important aspects covering the interests of a group of users is not an easy task. The expressiveness of any model is seriously limited when compared to what can be expressed in natural language, not to mention subtle semantics introduced by layout structures or temporal structures in sound and video.

However, the metadata-based approach has definite advantages. Instead of each and every agent in the system making their own analysis of the information, this work can be done once and made available to all interested parties. This way we can achieve economics of scale that

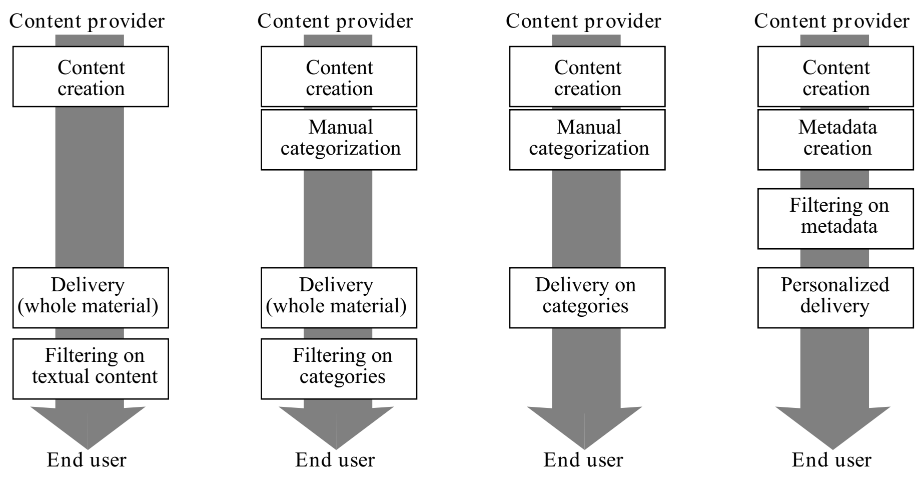


*Figure 2. Different alternatives for personalized content delivery.*

make it possible to use considerable human and machine resources for the metadata creation process. If the work is done only once, it can be done properly instead of haphazardly doing it over and over again.

| | Pros | Cons |
|---|---|---|
| **Early** | • Bandwidth savings<br>• Multiuse advantages<br>• Mobility<br>• Uniformity | • Loss of richness in information, if only metadata is delivered |
| **Late** | • Flexibility<br>• Full material available<br>• Own judgment | • Requires the full-scale information flow<br>• Nonuniformity |

*Table 1. Position of content analysis in the filtering process.*

## Generating Metadata during Content Development

SmartPush architecture relies on compact structural representation of the content, the metadata (Figure 3). Metadata means machine understandable information about information. An introduction to metadata standards can be found e.g. in (W3Ca 1998, IFLA 1998).

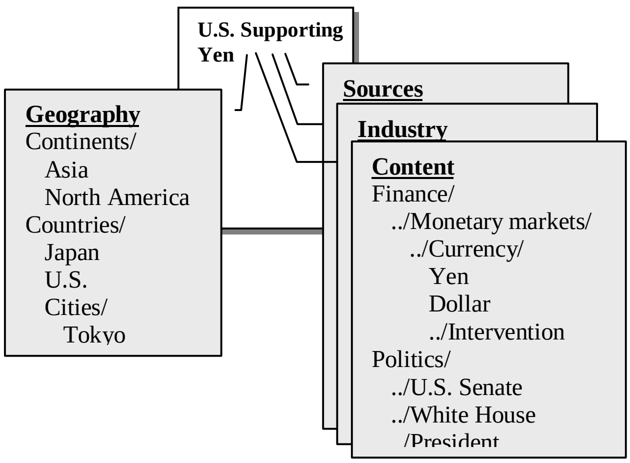


*Figure 3. A news article with its semantic metadata representation.*

Our approach requires that we can provide the delivery process with both the actual content and structural descriptions of the content semantics. This in turn leads to the need to generate this metadata during the content development (Figure 4).

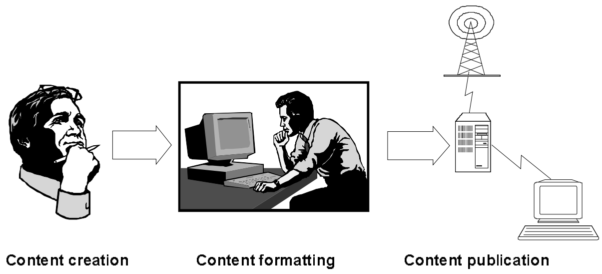


*Figure 4. Steps in the content development process.*

The amount and type of required metadata depends on the media and the end products created from the raw content.

Metadata can be generated during content creation or separately for each product during the content formatting and publication phases. Some of the issues related to generating semantic information within content development process are discussed below:

- *Automated generation*. Fully automated analysis of information semantics is difficult, because the interpretation of the content relies on human expertise. Metadata tools must therefore assist the generation as much as possible while leaving the judgment in classification to the human experts.
- *Different interpretations of the same content.* Everyone has his or her own interpretation and side of a story. However, if professionals using somewhat objective and systematic methods describe the content, we can rely on one conceptual model that can then be extended. These experts are available mostly during content creation and content formatting phases.
- *Author expertise*. Creator or editor together with assisting tools is the best source for ideas and thoughts behind the content. In a later phase of the process all aspects of the content can not be known.
- *Traditional publishing rules inapplicable.* Ensured quality, brands and structures similar to traditional media world do not exist, because online producers are not bound to expensive production facilities or real estate. It is thus easy to imitate media conglomerates and add own possibly false interpretations of the content. If semantic information is accepted only from entrusted content providers or their subcontractors, we can rely on those descriptions and thus improve perceived quality.
- *Need for coordination*. If semantic information does not exist in the information flow, every instance requiring that information must develop methods for generating and storing it. This in turn leads to inefficient and repetitive analysis of content and varying storage methods. In addition, if the content descriptions are later changed, distribution might lead to consistency problems.

In the SmartPush system, semantic analysis and metadata generation are tightly integrated with the authoring. This means that during the content creation phase authors generate the required semantic information with the help of a custom-made tool (Figure 5). This tool generates a suggestion of suitable semantic information for the information flow, after which authors make necessary modifications and publish this information together with the actual content.

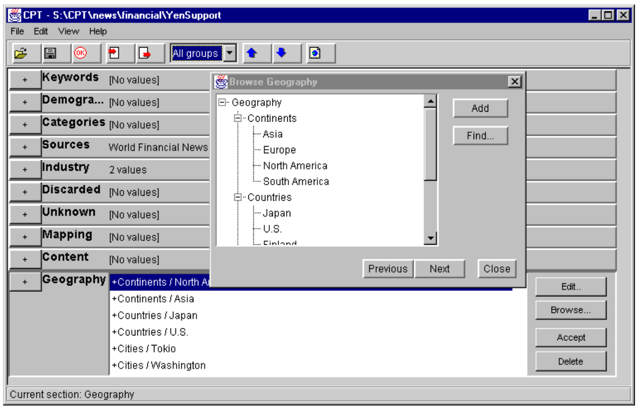


*Figure 5. Content provider's tool for metadata analysis and creation.*

## Metadata for Information Filtering

The following preconditions have to be met to successfully use metadata representations in information filtering.

- *Expressiveness and coverage*. Metadata system must be able to cover users' interests sufficiently to make personalization decisions possible. One must understand what qualities customers value in the information flow and how these are expressed in the metadata structure. Defining such structure is a difficult task, because customers have different interpretations and prioritizations regarding the content.
- *Machine understandability*. Once metadata is extracted, agents must be able to process it without human assistance and without original content.
- *Structural compatibility*. Agents require uniform or compatible metadata format to process content originating from different sources. This dictates that the format is flexible and expressive to accommodate different needs, for example to describe different media formats like video and audio as well as material in different languages.
- *Semantic compatibility*. In addition to a common format, agents need shared understanding of the metadata semantics. This is achieved with a common vocabulary with which to describe different types of content. If we want to use the same metadata format with multiple organizations, the descriptions must be agreed upon between these instances. Agents are of little help in resolving compatibility issues, as this task requires deep understanding of the content.

Several standardization efforts for creating uniform metadata structures are under way. The Resource Description Framework, RDF (Lassila and Swick 1998), is a data model for describing attributes and relations between documents. RDF is implemented with Extensible Markup Language, XML (W3Cb 1998). Another effort is Dublin Core (Weibel and Miller 1997), which defines a set of attributes for storing bibliographical information. Information Content & Exchange, ICE, concerns logistic information for republishing and syndicating electronic content.

None of these metadata efforts define usable semantics for describing content. We have chosen RDF as the metadata interchange format for our system, but have developed our own method for describing topics covered by an article. Our method consists of several independent dimensions. Each dimension describes one aspect of the content such as source, geographical location, story type or subject area.

Each dimension has its own data model. For describing the subject area of the material we have extended a simple hierarchic category system. The document's relation to different concepts is expressed as weights in the leaf nodes of a hierarchical concept model (Figure 6). By expressing the relative importance of different concepts as weights we gain a lot more expressive power compared to a simple binary belongs-to relation. This model provides also a natural way for summarizing related lower level topics to higher level concepts. In the current model the content provider determines the concept model, but it could also be defined and refined by feedback from the users.

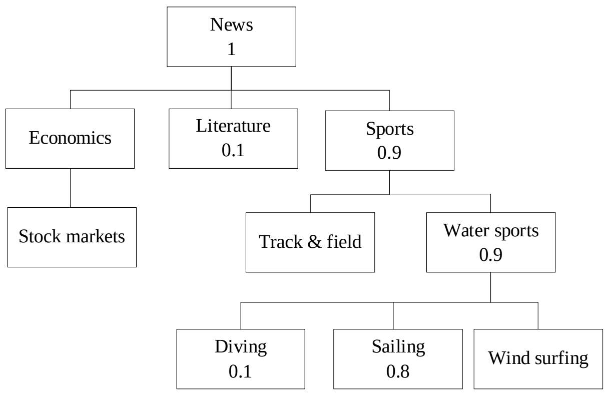


*Figure 6. Simplified example of a hierarchical model.*

## SmartPush Architecture

One part of the SmartPush project is to implement a working prototype for an information targeting environment. The underlying goal is to develop a framework where a content producer can provide a multitude of different, customized information feeds and where the end user can filter different incoming feeds to get a personalized information service.

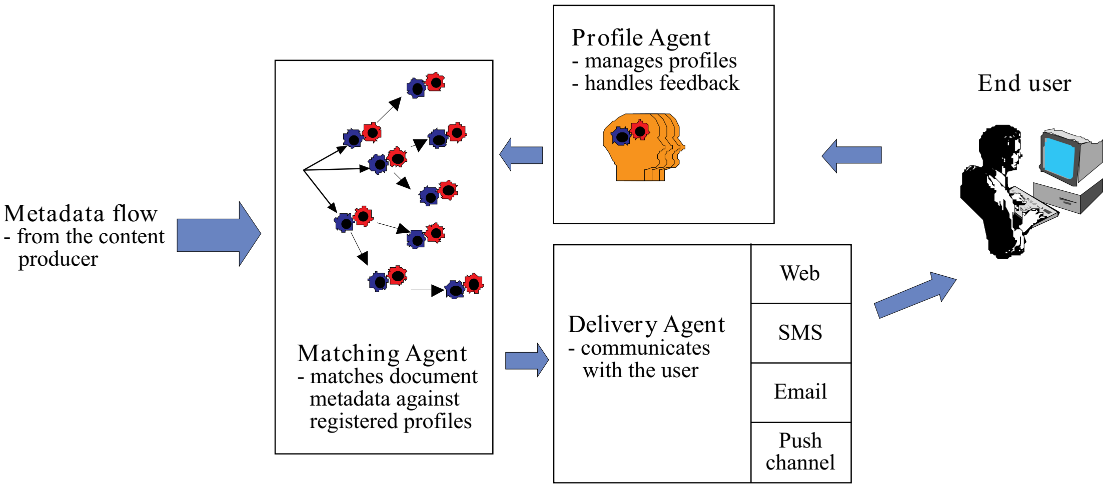


*Figure 7. Main agent types.*

Our architecture rests on two assumptions about semantic metadata. Firstly, we assume that it will be possible to create a domain specific ontology capable of expressing the different facets on which the end user decides his or her interest for a given information item. Secondly, we assume that it will be not only possible but also commercially viable to enhance the content with metadata descriptions as it is created.

SmartPush relies on high quality metadata representation of the content. We can think of the content producers as sources for metadata flows that describe the introduction of new content in the producers' multimedia databases. This flow can be distributed to any and all interested parties, since it acts as an advertisement for the actual content. The metadata flow is used to tell which content references should be delivered to each user.

This type of a system can be implemented as a monolithic information system based on a central database or as a system of relatively independent software entities. We believe that in today's networked information marketplace a rigid centralized system can not keep up with the changes in the environment. Content providers can no longer control the whole distribution chain. Internet Service Providers, portal managers and others interfacing with the end user need professional, high quality content. Thus there is a growing need for a distributed system that channels high quality content to the right users.

In the SmartPush system there exist three different major agent types. The tasks involved in managing user profiles, matching incoming metadata records and delivering information to the user are divided among three different processes (Figure 7). For facilitating dynamic collaboration among the agents we need also a coordination agent and ontology agents. This division of tasks among several different agents allows flexibility in distributing processing load, changing the flow of information on the fly and distributing tasks across organizational boundaries.

## Profile Agent

The profile agent's tasks involve managing profile and user information (Figure 8). When a new user enters the system he or she needs a fresh profile, which can be created in many ways. For example the user can rank sample documents and the system derives the profile information from them. The user can also give explicit keywords describing the profile or pick one or more prototype profiles as a baseline profile that will start to adapt. Once the interest profile is created it needs to be stored permanently.

The user needs to specify also options for delivering the results of the information filtering process. The profile contains information about how the system should act when sufficiently interesting material appears. If the system judges that the user's interest for an incoming document exceeds a certain threshold it can take direct action such as sending an email message to deliver the suggestion. An example of a more passive delivery option is keeping a list of ten most interesting suggestions for the user to retrieve at his discretion.

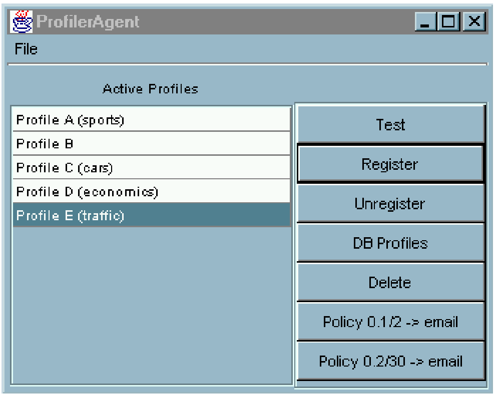


*Figure 8. Profile agent administration view.*

Adaptation is an important part of the overall system and requires either implicit or explicit feedback from the user. As the user gives feedback for the system's document suggestions the profile agent's job is to modify the contents of the interest profile to be better in line with the feedback received over time. This means that the profile agent may need to maintain explicit feedback history as well as history of the documents that have been suggested to the user and the user's reactions to them. Depending on the implementation of the adaptation algorithm history information may be stored in a summary form or as a complete log.

The location of the profile agent in the content provider - mediator - consumer chain has bearing on what functions are possible. The profile can be stored and managed on the end user's machine. This discloses as little information as possible about the user, but makes all aspects of inter-agent communication and social filtering harder to implement. Furthermore the user has to carry a device containing the profile with him for using the system on a mobile platform or at different locations. Distributed profile management makes also software upgrades more troublesome.

We have made the decision to have the profile agent in the network hosted by an information broker. One or more profile agents provide services to several users, manage their profiles and modify the profiles as the users interact with the system. Thus the user can contact his or her agent when and where he or she wants and make changes independent of location, available bandwidth, or device.

Separating the different functions in the delivery chain necessitates trusting only one party, the profile agent, with the user's interest information. The user's identity can be hidden from the matching agent and content provider if needed.

A database covering several users' profiles provides also a foundation for exploring social information filtering (Shardanand and Maes 1995). Similarities between profiles can be used to detect common areas of interest uncovered by each individual's profiles. The users benefit from this analysis of the profile database by getting suggestions they might otherwise have missed.

## Matching Agent

The actual matching process is carried out in a matching agent (Figure 9). The interaction between the profile agent and the matching agent functions on subscribe-unsubscribe basis, where the profile agent expresses the user's interest areas by sending over a compact interest representation. The matching agent sees the user as an abstract identification code, an interest profile and possible delivery options such as a minimum interest value defined for incoming suggestions.

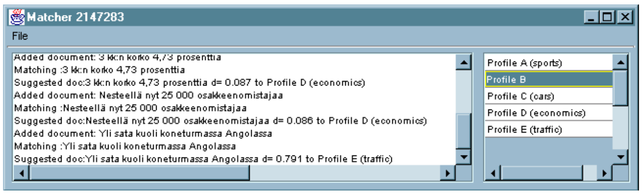


*Figure 9. Matching agent status view.*

The task of the matching agent is to compare incoming metadata records against the base of registered profiles. The agent listens for incoming metadata, decodes and parses it as it arrives and compares it against each profile. The comparisons are based on a distance measure that measures how far the document is from each profile (Figure 10). We have developed an asymmetric distance measure that considers how well the document fits some part of the profile (Savia, Kurki, and Jokela 1998). The best matching document does not have to cover all the areas that the user has interest for. Many other systems use a symmetric distance measure, such as the cosine measure used with the term vector model.

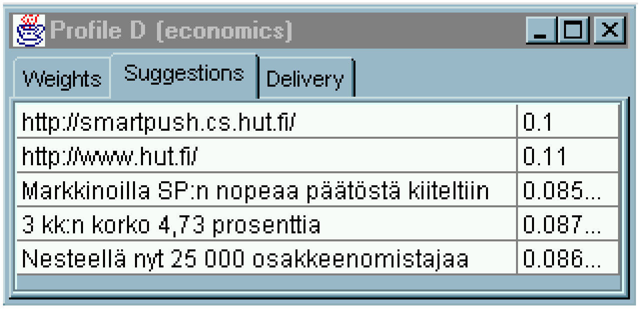


*Figure 10. Example suggestions with distance measures.*

Optimized computations are required for a scalable matching process. Our model for content description is hierarchic and summarizes the user's interest to several top level categories. This offers a natural way to limit the number of profiles that have to be considered as possible matches. The agent can take into account only those profiles that have something in common with the document on the top level of the hierarchy.

## Delivery Agent

Once the matching agent has decided to suggest a certain reference to the user it hands it over to the delivery agent. The delivery agent's role is to deliver the suggestion to the user according to the user's delivery options.

The delivery agent has different modules as its actors for communicating the suggestions to the user. The delivery process can involve two-way communication through a relatively restricted channel. For example the user can ask for current suggestions by sending a short message from a mobile phone to the delivery agent and have it return a suggestion list.

The basic idea in using a separate delivery agent is to separate the end user communication details from the relatively high level interagent communications. This way new communication and interaction mechanisms can be added without changes in the basic setup.

### Ontology Agent

The incoming metadata flow and the profiles managed by the profile agent have to have common representations for the matching agent to be able to make the comparisons in a meaningful way. Even if the matching agent understands the digital representations, they have to be compatible also on a conceptual level.

There is no doubt that creating a universal conceptual model and wide scale adoption of a single model is impossible in the real world. Thus we need to take into account the need for ontology agents that are able to translate between different representations available from different content producers. If there is no ontology agent that is able to offer at least a partial bridge between the semantic islands the system remains specific to a single source of information.

### Coordination Agent

If there are individual agents that need to discover each other, we need a coordination agent. In our case for example dynamic introduction of new ontologies and delivery services necessitates a system where new services announce their availability and contact information to a central coordination agent that acts as a clearinghouse for the current capabilities of the whole system.

A linear distribution chain consisting of a single triple of one matching agent, one profile agent and one delivery agent does not really need automatic coordination techniques. The agents' interconnections are hard coded or set up once with no dynamic reconfiguration.

## Agenthood in SmartPush

When applying *"Agenthood as an ascriptor"* (Bradshaw 1997) model to the SmartPush agents, we can easily admit that the components are not *really* agents. As creators of the system we possess knowledge about its inner workings and regard it as a deterministic system. However, if we observe the system from the user's point of view, the system may seem to be a rational entity with its capability to learn from users and to recommend interesting information.

As a common definition for agenthood does not exist, we analyze some common agent qualities and compare these to the existing and near future SmartPush architecture.

### Mediation

Agents have been characterized as intelligent middleware modules that mediate between resources and customers (Wiederhold 1992). Our agents' tasks can be mostly considered as mediation services. They access data from multiple resources, and represent the information in an integrated and homogenized package to the users.

Resource models and customer models in mediation services are conceptually based on hierarchies (Wiederhold and Genesereth 1997). This allows inheritance, selection, and aggregation in the model. Hierarchical approach is also a key to SmartPush functionality.

The matching agent is the matchmaker between the metadata used as the resource model, and the customer model that represents the end user. This agent acts as a logical mediator that connects the user with right information objects, as they become available. The functionality of the matching agent is similar to content-based routing of *matchmakers* in (Kuokka 1995).

Profile agents encapsulate user's interests and represent the user to other agents in the system. They provide a mediation service functioning as gatekeepers by not revealing private information unless necessary.

### Adaptivity

The profile agent acts as an *interface agent* (Maes 1994) that monitors user's behavior, learns from these interactions, and adapts to user's needs. This means changing user's interest profile according to monitored behavior.

SmartPush users provide feedback on suggestions and the system is able to adapt to changing interests. Thus the system acts as an adaptive agent by learning and improving its performance with experience.

### Cooperation and Communication

Shoham has defined agenthood as "agent is a software entity which functions continuously and autonomously in a particular environment, often inhabited by other agents and processes" (Shoham 1997).

*Social ability*, meaning an agent possessing explicit model of other agents, does not exist in SmartPush. The only moment in SmartPush where an agent has knowledge about interiors of another agent is when they communicate directly with the same data structures.

SmartPush agents are not currently *cooperative*, but collaborative filtering functionality is to be incorporated to the future versions of the system. When those facilities are ready, SmartPush will be able to recommend documents based also on other people's interests. Whether collaborative filtering qualifies for multi-agent behavior, is whole another question.

Without common semantics and understanding of the content there is little to communicate between different parts of the system. SmartPush does not currently use any generic agent communication language (ACL) such as KQML, but instead relies on Java Remote Method Invocation (RMI) mechanism for communication between the agents.

Without any coordination or ontology translation services in the current implementation there is little need for a universal communication mechanism. All agents in the system use shared, proprietary concepts. The main advantages for using separate agents for different tasks relate to load balancing, scalability issues and privacy.

In the future multiple content providers using different conceptual models require the use of an ACL. With accessible ontology agents SmartPush would be able to consolidate information from different sources so providing the user with more coherent service.

### Autonomy

Whether the SmartPush system acts *proactively or reactively* depends on the point of view. From customer's point of view the system is proactive with its ability to monitor its environment and recommend new material by using its information push facilities.

If we observe the system from the content provider's viewpoint, it seems to act reactively. From that angle the system is activated by the distribution of a new information object, which triggers the delivery of new material to the relevant customers. All in all, the system acts autonomously after the initialization.

### Mobility

The current model of dividing tasks among different models does not require agent mobility. The tasks are clearly divided and there is no need to migrate functionality from one agent to another. Bandwidth savings that motivate mobile agent architectures are not significant in our approach (Harrison, Chess, and Kershenbaum 1995).

## Related Work

In most agent-based information filtering systems agents analyze the information in complete and original form and create an index for document matching (Moukas and Maes 1998, Mostafa et al. 1997). Our matching agent compares metadata descriptions of incoming material against user profiles as new material is published. The computational requirements for the matching process are therefore greatly reduced.

Information Lens (Malone et al. 1987) is a rule-based system for information filtering for co-operative work environments inside organizations. Users create their own agents that filter the information based on the metadata that the material contains.

Fishwrap is a personalized newspaper that receives material from multiple information sources. The system uses a knowledge-base of topic definitions and matches incoming material against the specified topics (Chesnais 1995). This pre-categorizing significantly speeds up the personalization process for each Fishwrap reader.

InfoFinder is an agent that learns to find and categorize information using sample documents selected by the user. By using these samples, it extracts semantically significant phrases from documents and tries to learn optimal search queries for each category (Krulwich 1997).

EdInfo project at Swedish Institute of Computer Science uses *edited adaptive hypermedia* model, which combines human expertise with machine intelligence in the filtering process (Rudström, Waern, Höök 1997). A human expert acts as the information broker between information sources and readers. EdInfo users are able to collaboratively annotate and categorize the information. This approach has been tested in the ConCall system in filtering conference call messages (Waern et al. 1998). In SmartPush, we assume that the human expertise for metadata creation can be located at the information source instead of intermediary human brokers. However, the possibility of flexible user defined classification schemas is worth exploring.

Push technology has been touted as the solution to information overload problem. The information is distributed using the channel metaphor where users switch between channels of different types of information content. Most commercial push-oriented services, such as PointCast Network, are based on static user models derived from data entered directly by the user. Our work is more focused on personalization and adaptivity, leaving push technology as one option for delivery.

In our earlier work, we have specified a generic agent-based architecture for news filtering (Turpeinen et al. 1996), and introduced the idea of using edited semantic metadata in the filtering process (Saarela et al. 1997).

## Conclusions and Future Work

We have presented in this paper an information delivery architecture based on semantic metadata and agents. Once professional authors have generated both the content and its semantic metadata, agents add flexibility to the delivery by performing user profiling, document matching and delivery tasks.

The idea in our approach is to use the author's expertise and perform the semantic analysis of the content as early as possible, within the creation of the actual content. By using well-defined and expressive semantic models, we can generate personalized information flows with relatively straightforward computations. Incorporated semantic information allows agents to concentrate on the delivery without a need to understand the content. These ideas together lead to a more accurately targeted and higher quality information flow to the customers.

This paper describes work in progress. Our work has had a strong emphasis on methodologies for content and profile representation as well as for matching algorithms. The present implementation of the system can be seen as a proof of concept working on artificial data and profiles. Our current focus is shifting towards a pilot implementation and testing the ideas with a continuous real-world data flow. The future work in the project includes also a field trial within a commercial online news service.

The ideas presented in this paper have backing from media industry professionals and the described scenario seems viable for applications in today's online media. However, as the current results are still purely academic, the ideas can be fully validated only after the completion of the pilot phase.

## Acknowledgements

This research is part of the SmartPush project sponsored by the Finnish Technology Development Centre TEKES and Alma Media, WSOY, Sonera, ICL, Nokia Research Centre and TeamWARE Group.

## References

ANPA Format: Wireless Service Transmission Guidelines 1984. Report Number 84-2. *American Newspaper Publishers Association*. Washington, DC.

Bradshaw, J. 1997. An Introduction to Software Agents. In *Software Agents*, Bradshaw, J.(ed.), AAAI Press / The MIT Press. Cambridge, USA.

Chesnais, P.; Mucklo, M.; and Sheena J. A. 1995. The Fishwrap Personalized News System, *IEEE Second International Workshop on Community Networking Integrating Multimedia Services to the Home*. June 1995, Princeton, New Jersey, USA

Harrison, C. G.; Chess, D. M.; and Kershenbaum A. 1995. Mobile agents: Are they a good idea? *IBM Research Division*, T.J. Watson Research Center. Yorktown Hights, NY, U.S.A.

IFLA 1998. Metadata Resources
http://www.nlc-bnc.ca/ifla/II/metadata.htm

Krulwich, B. and Burkey, C. 1997. The InfoFinder Agent: Learning User Interests through Heuristic Phrase Extraction. *IEEE Expert*, September/October 1997.

Kuokka, D. and Harada 1995. Supporting Information Retrieval via Matchmaking. *Working Notes 1995 AAAI Spring Symposium on Information Gathering in Heterogeneous, Distributed Environments*. Technical Report SS-95-08, AAAI Press, 1995.

Lassila, O. and Swick, R. 1998. Resource Description Framework (RDF) Model and Syntax Specification. W3C Working Draft 08 October 1998, http://www.w3.org/TR/WD-rdf-syntax/ (work-in-progress).

Maes, P. (1994). Agents that Reduce Work and Information Overload. *Communications of the ACM*, Vol. 37, No. 7.

Malone, T. W.; Grant K. R.; Turbak F. A.; Brobst S. A.; and Cohen M. D. 1987. Intelligent information-sharing systems. *Communications of the ACM*, Vol. 30, No. 5, pp. 390-402.

Mostafa, J.; Mukhopadhyay, S.; Lam W.; and Palakal M. 1997. A Multilevel Approach to Intelligent Information Filtering: Model, System, and Evaluation. *ACM Transactions on Information Systems*, Vol. 15, No. 4, October 1997, pp. 368-399.

Moukas, A. and Maes, P. 1998. Amalthaea: An Evolving Multi-Agent Information Filtering and Discovery System for the WWW. *Autonomous Agents and Multi-Agent Systems*, Vol.1, No.1, pp. 59-88

Rudström, Å.; Waern, A.; Höök, K. 1997. Interactive Adaptation of Intranet Newsletters. *Sixth International Conference on User Modeling*, Sardinia, 2-5 June 1997

Saarela, J.; Turpeinen, M.; Puskala, T.; Korkea-aho, M.; and Sulonen, R. 1997. Logical Structure of a Hypermedia Newspaper. *Information Processing & Management*, Vol. 33 No. 5, pp. 599-614.

Salton, G. and McGill, M.J. 1983. *Introduction to Modern Information Retrieval*. McGraw-Hill, New York

Savia, E.; Kurki, T., Jokela, S. (1998). Metadata based Matching of Documents and User Profiles. In *Human and Artificial Information Processing, Proceedings of the 8th Finnish Artificial Intelligence Conference*, pp. 61-69: Finnish Artificial Intelligence Society.

Shardanand, U. and Maes, P. 1995. Social Information Filtering: Algorithms for Automating "Word of Mouth". *CHI'95 Proceedings*, Denver, Colorado, USA.

Shoham, Y. 1997. An Overview of Agent-Oriented Programming. In *Software Agents*, Bradshaw, J.(ed.), AAAI Press / The MIT Press. Cambridge, USA.

Turpeinen, M.; Saarela, J.; Korkea-aho, M.; Puskala, T.; Sulonen, R. 1996. Architecture for Agent-mediated Personalised News Services. *PAAM'96: The First International Conference on Practical Application of Intelligent Agents and Multi-Agent Technology*, 22-24 April 1996, London, UK.

W3Ca 1998. Metadata Activity. *W3C Technology and Society domain*,
http://www.w3.org/Metadata

W3Cb 1998. Extensible Markup Language (XML™). *W3C Architecture Domain*,
http://www.w3.org/XML/

W3Cc 1998. The Information and Content Exchange (ICE) Protocol. *Submission to the World Wide Web Consortium*, http://www.w3.org/TR/1998/NOTE-ice-19981026

Weibel, S. and Miller, E. 1997. Dublin Core Metadata Element Set WWW homepage.
http://purl.org/metadata/dublin_core

Waern, A.; Tierney, M.; Rudström, Å.; and Laaksolahti, J. 1998. ConCall: An information service for researchers based on EdInfo. *SICS Technical Report T98-04*, October 1998.
http://www.sics.se/~mark/papers/t9804/t98-04.htm

Wiederhold, G. 1992. Mediators in the Architecture of Future Information Systems. *Computer,* Vol. 25, No.3, March 1992, pp. 38-49.

Wiederhold, G. and Genesereth, M. 1997. The Conceptual Basis for Mediation Services. *IEEE Expert*, September/October 1997, pp. 38-47.