# FeedWinnower: Layering Structures Over Collections of Information Streams

**Lichan Hong, Gregorio Convertino, Bongwon Suh, Ed H. Chi, and Sanjay Kairam**

Palo Alto Research Center (PARC)

Palo Alto, CA 94304, USA

{hong, convertino, suh, echi, kairam}@parc.com

## ABSTRACT

Information overload is a growing threat to the productivity of today's knowledge workers, who need to keep track of multiple streams of information from various sources. RSS feed readers are a popular choice for syndicating information streams, but current tools tend to contribute to the overload problem instead of solving it. We introduce FeedWinnower, an enhanced feed aggregator that helps readers to filter feed items by four facets (topic, people, source, and time), thus facilitating feed triage. The combination of the four facets provides a powerful way for users to slice and dice their personal feeds. In addition, we present a formative evaluation of the prototype conducted with 15 knowledge workers in two different organizations.

## Author Keywords

RSS feed, feed reader, RSS overload, faceted browsing, information stream, information overload.

## ACM Classification Keywords

H5.2 [Information Interfaces and Presentation]: User Interfaces - Graphical User Interfaces.

## General Terms

Design, Human Factors.

## INTRODUCTION

Today's knowledge workers face the challenge of keeping track of numerous information streams from different sources, e.g., email messages sent by colleagues and friends, news stories related to topics of interest, new tweets posted to Twitter, and status updates in Facebook and LinkedIn. To avoid the headache of visiting different web sites and tools, many people have adopted RSS feed readers such as FriendFeed (friendfeed.com) and Google Reader (reader.google.com). RSS, aka Really Simple Syndication, is a popular web-based format for publishing frequently updated content and is supported by most Web 2.0 tools.

For many people, particularly those subscribing to a large

number of feeds, the problem of information overload arises quickly. As is the case with email triage [7], combing through lots of unread feed items and deciding what to read can be daunting, especially to busy knowledge workers. In most feed readers, feed items are displayed in a linear, reverse-chronological list, which provides limited support for feed triage at the user interface level. For example, Google Reader offers the functionality to filter items by people or source (but not both), while FriendFeed allows users either to filter by people or to use a form-based search tool (friendfeed.com/search/advanced). To deal with the overload problem, people have tried various strategies such as organizing feeds into tiers of different priorities or adopting filtering tools. Most filtering tools, however, assume that users have a prior idea of what they want to find. This assumption violates the idea of serendipity that people are hoping for when subscribing to feeds.

In this paper, we describe *FeedWinnower,* a web-based faceted browser designed to support feed triage (Figure 1). Given a set of feed items collected through FriendFeed, our prototype generates four facets: *topic, people, source,* and *time*. As shown in Figure 1, users interact with the facets on the left to construct a query to filter the feed items down to a subset of items, which are displayed reverse-chronologically on the right. For each feed item, we show its creator, source, content, and time of creation.

The topic facet serves a dual role, not only providing an overview of the content of the items, but also enabling users to choose a topic of interest. Selecting a topic results in



**Figure 1. The user interface of FeedWinnower: facet controls on the left and feed items on the right.**

filtering down to the subset of items related to the chosen topic. When combined, these facets allow users to "slice and dice" the items, helping them to browse items by features such as *who*, *where*, *what,* and *when*.

The main contributions of this paper are: (1) applying the combination of topic, people, source, and time facets to filter feed items and (2) proposing an *interactive* topic facet whose topics are automatically computed from the content of feed items.

**RELATED WORK**

Designers of interfaces for managing RSS feeds can benefit to a great extent from the prior research on email overload, e.g., [7,8]. Feed items share many characteristics with email messages; in fact, a feed item includes information such as *who* created it, *where* it was created, *when* it was created, and *what* it is about. With email, Whittaker et al. [8] noted that filing messages in folders is time-consuming and can be problematic if users' focus changes frequently, suggesting the need for flexible interfaces to allow on-the-fly browsing of content. Studying email triage, Neustaedter et al. [7] found that sender, receiver, and time were main attributes that people used to judge the importance of email. They also observed that people organized folders according to time, project, person, or interests. These findings are well aligned with the design of FeedWinnower, which includes topic, people, source, and time as facets for filtering.

There is extensive prior research on faceted browsing [5]. A facet is a category used to characterize information items in a collection [5]. Hearst suggested that social tags provide an excellent basis for the formation of topic structures for faceted browsing [5], but stressed that acquisition of facet metadata is a problem remaining to be addressed. Among the four facets utilized in FeedWinnower, we extract information about people, sources, and time directly from the feed items. However, many feeds provide no topic tagging. To compute a set of browsing topics, we developed a novel technique that mines the content of the items.

Related research also includes the design of blog search and browsing interfaces [1,6]. Hearst et al. [6] suggested design choices such as "the temporal/timelines aspect of blogging" and "automatic creation of a feed reader on the subtopics of interest". Baumer and Fisher [1] proposed an interface for organizing blogs around a list of extracted topics.

Probably the most closely related work is the tool by Dork et al. [4], which organizes RSS feeds along three dimensions: *time*, *location*, and *tags*. It also supports a faceted browsing interface. They assumed that feed items have titles and descriptions, time of creations, locations, and tags. A key difference is that we make no assumptions about the presence of tags or manually added metadata. Instead, we construct the topic facet from the content of the items. In addition, learning from email triage [7], we recognize that people and sources are salient indicators of feed importance, and embrace them as part of our design.

**THE FOUR FACETS**

In this section, we describe how we construct the four facets, followed by an illustration of their usage. Our design was informed by the findings of a prior field study [3]. The four facets described in this section were selected based on the multiple rounds of surveys and interviews with two groups of busy professionals in a large enterprise.

**Topic Facet**

A common practice for users of existing feed readers is to start by skimming through the list of unread items to form a mental model of what topics are being discussed. The goal of our topic facet is to provide at a glance this high-level summary. Some of the extracted topics may be long-standing ones that appear frequently while others may emerge serendipitously due to an outburst of temporal events.

Generally, a feed item includes a short text snippet, the specific content of which is dictated by the system creating the item. For example, a Twitter item contains the text content of a tweet, and a del.icio.us item includes the title of a bookmarked URL. We extract topics from these text snippets. We have tried a variety of methods and settled on the following noun identification technique. Inspired by past work using noun phrases to represent concepts of text snippets (e.g., [9]), we first identify nouns appearing in the content of each item using the Stanford part-of-speech tagger (nlp.stanford.edu/software/tagger.shtml). Then, for each noun identified, we count its occurrences across all items. Subsequently, the top $n$ nouns most frequently appearing in the items are chosen and displayed as a tag cloud (see Figure 2), where $n$ is pre-determined.
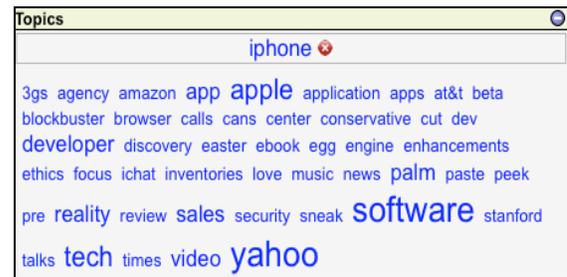


**Figure 2. A topic facet showing the key topics covered by the feed items. The topic "iphone" was selected and is displayed above the cloud of topics.**

The user can click any topic in the facet (e.g., "iphone") to filter the feed items down to the subset of items containing that topic, reducing the number of items being displayed on the right side of Figure 1. Consequently, the facet is updated to display a new tag cloud corresponding to the remaining items. The topics of this new tag cloud are constructed from the content of the remaining items by executing the same algorithm described above. Selecting additional topics will further reduce the remaining items to those containing all the selected topics, thus functioning as an AND query. The user can repeat this process to select

any number of topics in order to further filter the items or unselect a chosen topic to expand the set of items in focus.

By no means do we suggest that the noun identification technique is the best choice for constructing the topic facet. Other techniques that we have tried include TF-IDF and a term expansion algorithm [2]. When we ran these alternatives on our testing feeds, however, the resulting topics were not as meaningful. In addition, highlighting the noun phrases in the display of the remaining items helps to explain to the user why the items are included in the result.

**People, Source, and Time Facets**
*People*. Each feed item has information about *who* created it. Aggregating the list of creators, we construct a people facet as shown in Figure 3. An image icon represents a creator (i.e., a person or an organization), and a number indicates how many items that creator has contributed. By clicking, the user can select any subset of creators. For example, in Figure 3, the user has chosen HP and SUN, filtering the items down to those produced by HP or SUN.
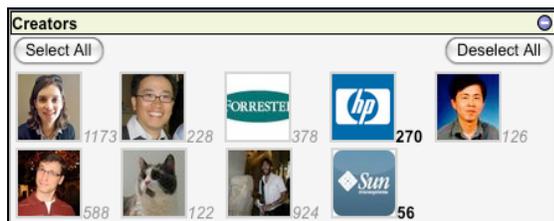


Figure 3. A people facet showing the list of creators and the number of items each creator has produced.

*Source*. Each item has information about *where* it was created or delivered from, e.g., Facebook or del.icio.us. The source facet shown in Figure 4 contains a list of these source services. A text label, along with a number indicating how many items were created, represents each source. The user can click to select any subset of sources.



Figure 4. A source facet showing the list of sources and the number of items coming from each source.

*Time*. Each item has information about *when* it was created. Aggregating the time information, we construct a time facet as shown in Figure 5. Each bar represents the number of items created in that time period. We offer three time granularities, enabling the user to group items by day, month, or year. By clicking a bar, the user can zoom into a more detailed view (e.g., going from a monthly view to a daily view of a chosen month). At any point, the user can also click on a button at the top of the facet to return to a more general view (e.g., going back to the yearly view).
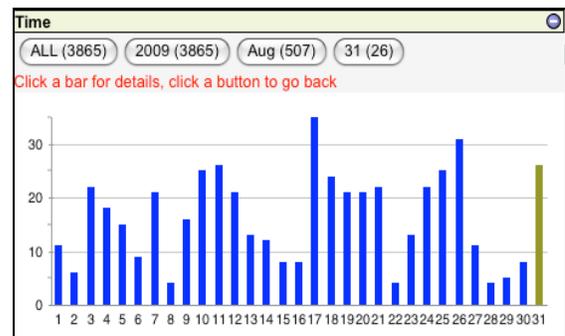


Figure 5. A time facet showing the distribution of items over the month of Aug 2009.

**Usage Scenarios**
By combining the four facets, users can examine and navigate their feeds, deciding what items to skip and what to read. Here we give two illustrative real-world scenarios.

*Scenario 1:* At the end of a workday, Mary opens FeedWinnower to get a sense of what has been happening around her. Using the time facet, she finds out that 507 items came into her account earlier in the day. Glancing at the topic facet, she sees "iphone" and a few other topics being talked about. As she clicks on "iphone", the right screen shows only 7 items after filtering out other items. In the people facet, she identifies that these 7 items came from 4 of her friends and decides to read those items in detail.

*Scenario 2:* John wants to find out what his friends have been chatting about on Twitter lately. He selects "Twitter" in the source facet and chooses "yesterday" in the time facet. This yields 425 items. In the people facet, he then excludes those creators that he wants to ignore, filtering down to 324 items. Looking at the topic facet, he sees "betacup" and wonders what it is about. After clicking on "betacup" and reading the remaining 7 items, he now has a fair understanding about the term "betacup".

In these two scenarios, we see how the four facets enable users to construct simple queries to accomplish their needs. We also see how the topic facet is essential in obtaining an overview of the topical trends in the feeds and helping users to decide what is worth reading in depth.

**Implementation**
Our prototype takes advantage of the aggregation functionality of FriendFeed, which we believe is an example of functionalities becoming increasingly available on the web. Once a user creates a FriendFeed account, we collect his feed items through FriendFeed's API. From the items we construct the four facets as described above. For performance purposes, we store a copy of these items in our server to dynamically update the facets and interactively display the items. In addition, we periodically update the stored items as new items arrive in the user's feed.

**FORMATIVE EVALUATION**

FeedWinnower is a research prototype and its functions are still evolving. As part of a larger iterative design project [3], we ran a formative evaluation to assess the usability and utility of the current prototype and elicit new requirements.

In total, the evaluation involved 15 knowledge workers: 8 senior professionals from a large IT enterprise and 7 managers from a mid-sized research organization. All shared in their work a need to manage multiple information streams while coordinating people and projects. For each participant, we collected data through an 1-hour interview, run in the participant's own work setting, and followed with a usability questionnaire. In the first part of the interview (25~30 minutes), each participant reported about her/his information streams and practices. In the second part, s/he was given an overview of the tool functions on her/his own computer. The tool had been pre-populated with a rich set of streams, which was kept constant across the participants.

Participants were first given two example tasks (which were recurrent and central to their jobs [3]) as task context for the evaluation: (1) foraging useful bits of content for a new project and (2) monitoring status updates of colleagues in various projects. They were also encouraged to consider other personally relevant tasks. Then, they were invited to try out each facet (and combinations), assess its utility, and propose improvements. This period lasted 25~30 minutes. Finally, a few days after the interview, they were asked to use the tool independently and give more feedback via a questionnaire mirroring the second part of the interview.

Through the interview and questionnaire, the evaluation yielded two main results about the prototype. First, all participants saw the utility of the faceted browser for managing information fragments and status updates across their information streams, including the added benefits of being able to dynamically combine the facets. They explored the flexibility of the tool by quickly formulating various types of queries during attempts to find relevant items. Two participants commented: "*Combining them is really interesting*"; "*I think it would be really useful during the information gathering stage, when you bring in loads of information together from different sources*". The usability ratings from the questionnaire were consistent: about 1 point above the neutral value or mid-point (4.0) of the 7-point scales. Average facet ratings: "Useful" (4.9, SD=1.5), "Easy to Use" (4.8, SD=1.5). Overall ratings: "Satisfaction" (5.1, SD=1.5), "Easy to Use" (5.1, SD=1.1).

The second result was feedback on preferred facets and combinations of facets. The preferred combinations of facets varied widely, which was expected given differences in their daily tasks. However, the interview and questionnaire data suggested that the most commonly preferred facets were topic (9/15 chose as most relevant) and source (4/15). A participant said: "*I would definitely put the topic first…the one about the source I would probably put at the second.*" People (2/15) and time (2/15) were chosen by fewer subjects. The ratings of usefulness by facet were consistently higher for topic (5.1, SD=1.6) and source (5.1, SD=1.4) than for people (4.8, SD=1.1) and time (4.6, SD=1.7), although this difference was not significant. The panel displaying the feed items was rated as easy to use (5.3, SD=1.3).

The data collected also pointed to new requirements for future design, including extensions of the current functions, such as search and content preview, as well as new functionality such as allowing organization of content and task-related items (foldering, to-dos) or supporting social functions (shared folders, item rating).

**CONCLUSION**

We presented FeedWinnower, a feed aggregator where the topic, people, source, and time facets can be combined to support feed triage. We also introduced a technique for constructing the topic facet that does not require manual creation of tagging data. In our evaluation, 15 enterprise professionals saw the value of being able to dynamically combine the four facets, and rapidly adopted different combinations of facets to fit their diverse tasks. The topic and source facets emerged as particularly valuable. Finally, we collected promising directions for future design.

**REFERENCES**

1. Baumer, E. and Fisher, D. Smarter Blogroll: An Exploration of Social Topic Extraction for Manageable Blogrolls. *Proc. HICSS'08.*

2. Bernstein, M., Suh, B., Hong, L., Chen, J., Kairam, S., and Chi, E. Eddi: Interactive Topic-based Browsing of Social Status Streams. *In review.*

3. Convertino, G., Kairam, S., Hong, L., Suh, B., and Chi, E. Designing A Cross-channel Information Management Tool for Workers in Enterprise Task Forces. *In review.*

4. Dork, M., Carpendale, S., Collins, C., and Williamson, C. VisGets: Coordinated Visualizations for Web-based Information Exploration and Discovery. *IEEE Trans on Vis. and Computer Graphics*, 14(6), 1205-1212.

5. Hearst, M. UIs for Faceted Navigation: Recent Advances and Remaining Open Problems. *Proc. 2008 Workshop on Human-Computer Interaction and Information Retrieval*.

6. Hearst, M, Hurst, M., and Dumais, S. What Should Blog Search Look Like? *Proc. 2008 ACM Workshop on Search in Social Media*, 95-98.

7. Neustaedter, C., Brush, A., and Smith, M. Beyond "From" and "Received": Exploring the Dynamic of Email Triage. *Proc. CHI'05*, 1977-1980.

8. Whittaker, S. and Sidner, C. Email Overload: Exploring Personal Information Management of Email. *Proc. CHI'96*, 276-283.

9. Xu, J. and Croft, W. B. Query Expansion Using Local and Global Document Analysis. *Proc. SIGIR'96*, 4-11.