

# Syndication Aggregation

## Web Architecture and Information Management [./] Spring 2009 — INFO 190-02 (CCN 42509)

Erik Wilde, UC Berkeley School of Information

2009-04-08



<http://creativecommons.org/licenses/by/3.0/>

This work is licensed under a [CC Attribution 3.0 Unported License](http://creativecommons.org/licenses/by/3.0/) [http://creativecommons.org/licenses/by/3.0/]

## Contents

• Abstract	2
• Feeds as Data Sources	3
• 1 Syndication Aggregation	
◦ End-User Aggregation	5
◦ Aggregation Intermediaries	6
◦ 1.1 FeedBurner	
▪ Fixing Feeds	8
▪ Load Balancing	9
▪ Statistics/Analytics	10
▪ Query Capabilities	11
▪ Supporting Queryable Feeds	12
• 2 Aggregating Delicious Tags	
◦ Using Feeds for Data Management	14
◦ Delicious Tags as Feeds	15
◦ Google Reader Feeds	16
◦ Google Reader Widgets	17
◦ Explaining Delays	18
• Feeds as Data Containers	19

## Abstract (2)

Feeds are useful as information sources adhering to a standardized format, but they usually have a single source. In a larger picture of information flows on the Web, the question is how feeds can be handled more efficiently and flexibly. This includes questions such as *load balancing* and *user statistics*, but also more complex scenarios such as *aggregating* (and maybe even *filtering*) feeds. Since feeds use a standardized data format, such repurposed information again can be published as a feed, creating a flexible architecture of feed-based information dissemination, allowing an arbitrary number of feed aggregation steps.

## Feeds as Data Sources (3)

- Feeds are published by an increasing number of sites
- Reading feeds in *feed reader* is convenient
- Handling feeds in a more sophisticated way is interesting
- Feeds will evolve into the main container format on the Web

# Syndication Aggregation

## End-User Aggregation

(5)

- Users often have a small number of preferred Web sites
  - news can be tracked by subscribing to their feeds
  - this requires an feed-aware client (Firefox is not a good feed reader)
- AJAX-based feeds could be implemented as a client-side application
  - retrieve the feed, transform it to HTML, and render the result
  - but the security restriction of XMLHttpRequest gets into the way
  - [Google Reader](http://www.google.com/reader/view/) [http://www.google.com/reader/view/] accesses Google's own copy of the feed ([FeedBurner](#) [FeedBurner (1)])
- How can users find a feed?
  - feeds have URIs (they are Web resources) and can be referenced from within HTML



```
<link rel="alternate" type="application/rdf+xml" title="..." href="..." />
```

```
<link rel="alternate" type="application/rss+xml" title="..." href="..." />
```

```
<link rel="alternate" type="application/atom+xml" title="..." href="..." />
```

## Aggregation Intermediaries

(6)

- RSS is a frequently used format between content providers
  - news agencies want to distribute news items as easy as possible
  - by using a single format, producers and consumers can more easily interoperate
- Multiple RSS versions caused publishers to look for something better
  - [RSS History](#) [Content Syndication; RSS History (1)] had a head start and still is widely used
  - [Atom History](#) [Content Syndication; Atom History (1)] is a much better format and will eventually replace RSS
- User-configured portals are the merger between both scenarios
  1. users select a number of feeds as their preferred information sources
  2. the feeds are presented on a [single personalized Web page](http://www.google.com/reader/view/) [http://www.google.com/reader/view/]
  3. by [sharing](http://www.google.com/reader/shared/16601496766743088901) [http://www.google.com/reader/shared/16601496766743088901] and [re-publishing](http://www.google.com/reader/public/atom/user/16601496766743088901/state/com.google/broadcast) [http://www.google.com/reader/public/atom/user/16601496766743088901/state/com.google/broadcast] items, users are becoming aggregation intermediaries

### Shared items

Document Design Matters  
from dretblog

HTTP over SOAP over HTTP  
from dretblog

"URI Scheme for GSM Short Message Service", Erik Wilde,....  
from Recent Internet-drafts

Grouping in XQuery  
from O'Reilly News: XML

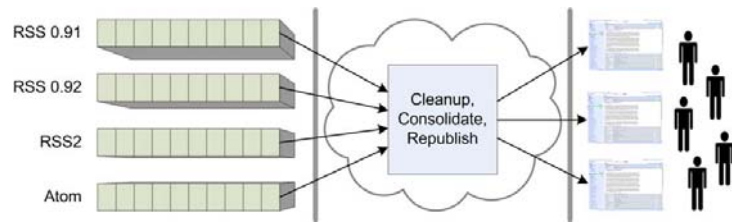
XInclude Processing in XSLT  
from O'Reilly News: XML

[Read more...](#)

# FeedBurner

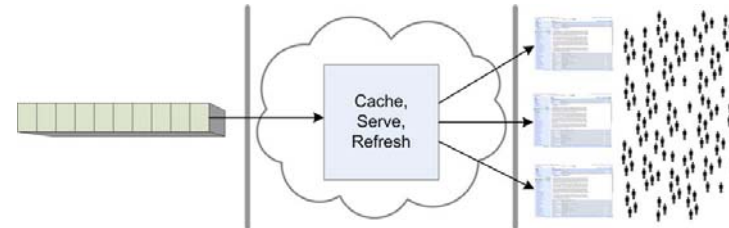
## Fixing Feeds

(8)



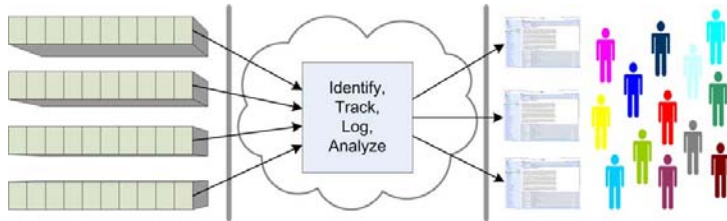
## Load Balancing

(9)



## Statistics/Analytics

(10)



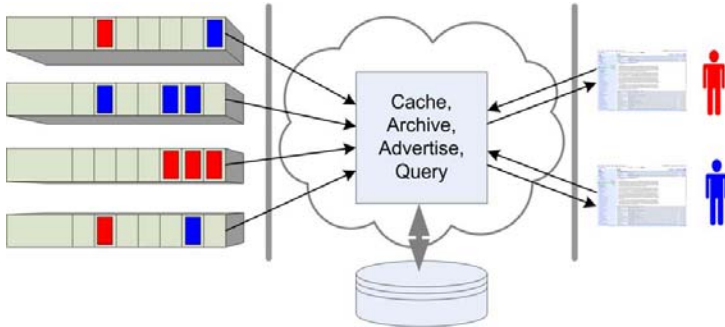
## Query Capabilities

(11)

- Feed technology is still evolving
- Feeds are mostly viewed as ordered by time
  - allows optimization for accesses and caching
  - makes it hard to use feeds for non-timed information
- Feeds could be ordered by any sort key
  - makes server-side feed processing much more expensive
  - enables customized feeds that are processed on the server-side

## Supporting Queryable Feeds

(12)



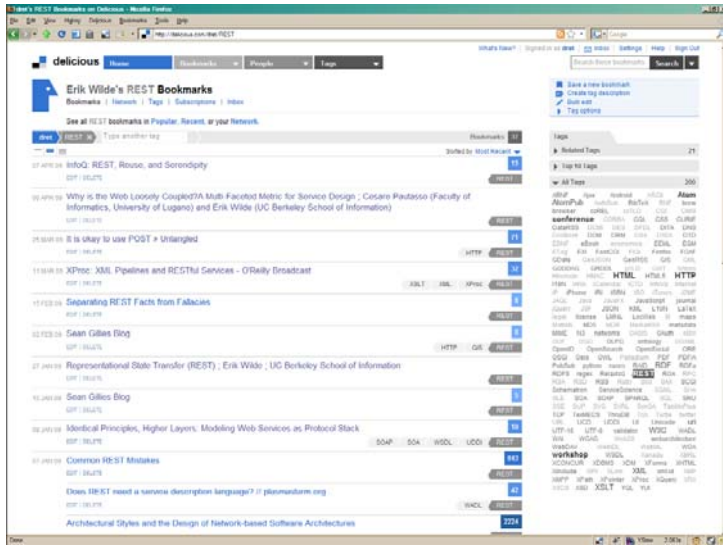
## Aggregating Delicious Tags

### Using Feeds for Data Management

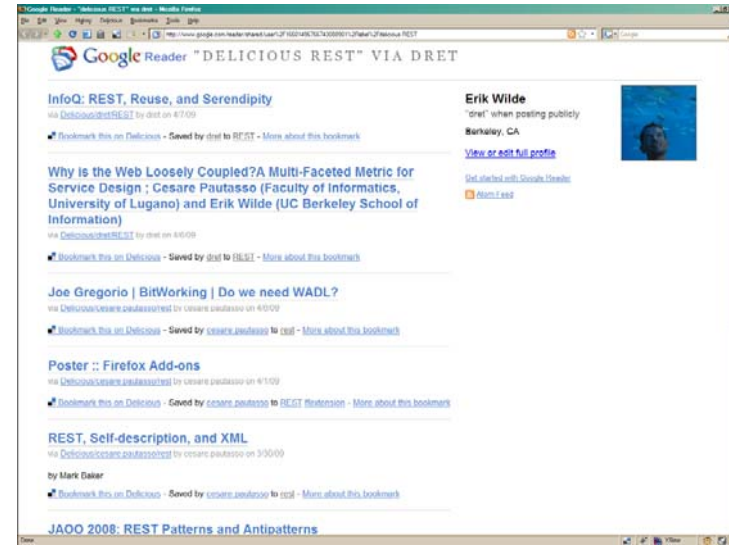
(14)

- Delicious tags are available as feeds
- Creating a "joint tag feed" is as simple as aggregating feeds
  1. identify all tag feeds
  2. add all feeds to a feed aggregation platform
  3. republish the aggregate information as *one* feed
  4. use that feed for the aggregate information for all tags

# Delicious Tags as Feeds (15)



# Google Reader Feeds (16)



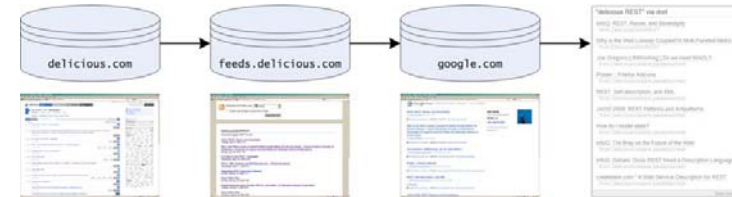
## Google Reader Widgets (17)

```

<table>
  <thead>
    <tr>
      <th>Feed Roll</th>
      <th>Feed Items</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td valign="top">
        <script type="text/javascript"
src="http://www.google.com/reader/ui/publisher-en.js"></script>
        <script type="text/javascript" src="http://www.google.com/reader/public
/javascript-sub/user/16601496766743088901/label/delicious
REST?callback=GRC_p(%7Bc%3A%22gray%22%2Ct%3A%22My%20%5C%22delicious%20REST
%5C%22%20Feed%20Roll%22%2Cb%3A%22true%22%7D)%3Bnew%20GRC"></script>
      </td>
      <td valign="top">
        <script type="text/javascript"
src="http://www.google.com/reader/ui/publisher-en.js"></script>
        <script type="text/javascript"
src="http://www.google.com/reader/public/javascript/user/16601496766743088901
/label/delicious REST?n=10&callback=GRC_p(%7Bc%3A%22gray%22%2Ct%3A%22
%5C%22delicious%20REST%5C%22%20via%20dret%22%2Cs%3A%22true%22%2Cb%3A%22false
%22%7D)%3Bnew%20GRC"></script>
      </td>
    </tr>
  </tbody>
</table>

```

## Explaining Delays (18)



## Feeds as Data Containers (19)

---

- [Containers](http://en.wikipedia.org/wiki/Containerization) [http://en.wikipedia.org/wiki/Containerization] revolutionized the trade of physical goods
- Feeds are the containers of the Web
- Managing feeds will become much more important
- Feeds being produced by all kinds of networked "things"
- Better feed management and querying is still under development