

# mle: Enhancing the Exploration of Mailing List Archives Through Making Semantics Explicit

Michael Hausenblas, Herwig Rehatschek

Institute of Information Systems & Information Management,  
JOANNEUM RESEARCH Forschungsges. mbH,  
Steyrergasse 17, 8010 Graz, Austria  
`firstname.lastname@joanneum.at`

**Abstract.** Following and understanding discussions on mailing lists is a prevalent task for executives and policy makers in order to get an impression of one’s company image. However, existing solutions providing a Web-based archive require substantial manual effort to search for or filter certain information. With `mle` we propose a new way to automatically process mailing list archives. The tool is realised based on two Semantic Web technologies: Firstly, SIOC is utilised as the primary vocabulary for describing posts, people, and topics; secondly the RDF metadata is deployed by means of embedding it in the Web page encoded in RDFa.

## 1 Motivation

Though instant messaging, blogs, feeds, etc. become more and more important means of communication and discussion, mails remain one of the cornerstones of the business world on the Internet. The consumption and navigation in mailing discussion threads—actually the process of understanding discussions—is still a prevalent task for executives and policy makers in order to get an impression of one’s company image.

Existing solutions that provide a Web-based access of a mailing list archive, such as `hypermail`<sup>1</sup>, or `marc`<sup>2</sup>, offer limited support w.r.t. search or filter information, and for further processing the content. In order to carry out, eg., a company image analysis—based on postings with extreme positive or negative statements regarding the company—the processing has to be automated. Another reason to further processing mailing lists is to gather public knowledge on certain companies or products to efficiently support market researchers, as for example done in the “Understanding Advertising” (UAd) project<sup>3</sup>.

To enable mailing list archives to successfully enter the Semantic Web, a sensible RDFizing of the implicit metadata in a top-performing and scaleable way is desirable.

---

<sup>1</sup> <http://www.hypermail-project.org/>

<sup>2</sup> <http://marc.info/>

<sup>3</sup> <http://www.sembase.at/index.php/UAd>

Attempting to meet the above listed requirements, we propose a new way of interacting with, and processing of mailing list archives: `mle`, the mailing list explorer. We briefly discuss the utilised RDF-based metadata in section 2; we then describe the system architecture (cf. section 3), including features and usability issues. Finally we reflect our experiences with `mle` in section 4.

## 2 RDF-based Metadata in `mle`

Typically two main issues arise when developing a Semantic Web application: (i) the selection or definition of the vocabularies used, and (ii) the actual deployment of the metadata. Whereas for the vocabulary the most important issue might be the reuse of existing bits, one has to note that regarding the deployment no standardised way—be it referenced or embedded—was available. However, in the last decade a number of proposals can be recorded<sup>4</sup>.

Due to the W3C activities addressing the RDF-in-HTML issue, this situation has changed. With the advent of RDFa [1, 2] it seems we now have a standardised, technical sound and widely supported solution for embedding RDF in (X)HTML pages.

### 2.1 Vocabularies

When the domain of discourse is defined rather sharply, it is quite straightforward to pick appropriate vocabularies to cover it as a whole. In our case there was no need to extend existing vocabularies; available vocabularies targeting at the social domain were adopted.

The **Friend of a Friend (FOAF)**<sup>5</sup> project is creating a Web of machine-readable pages describing people, the links between them and the things they create and do. The FOAF [3] vocabulary makes it easier to share and use information about people and their activities, e.g., photos, calendars, blogs, to transfer information between Web sites, and to automatically extend, merge and re-use it online.

**Semantically-Interlinked Online Communities (SIOC)**<sup>6</sup> is a vocabulary [4] to describe interconnected discussions in various so-called containers, as blogs, forums and mailing lists etc. It partially builds upon, and extends FOAF. Recently, SIOC was submitted for W3C standardisation<sup>7</sup>, hence a wide-spread and uniform adoption is very likely. A related approach to ours was reported in [5]; for a comprehensive list of SIOC applications and implementations, the reader is referred to [6].

As a more generic vocabulary, **Dublin Core** [7] is utilised in `mle` to capture simple or generic properties as `title`, `date`, etc.

<sup>4</sup> The interested reader is referred to <http://infomesh.net/2002/rdfinhtml/> for a rather complete overview.

<sup>5</sup> <http://www.foaf-project.org/>

<sup>6</sup> <http://sioc-project.org/>

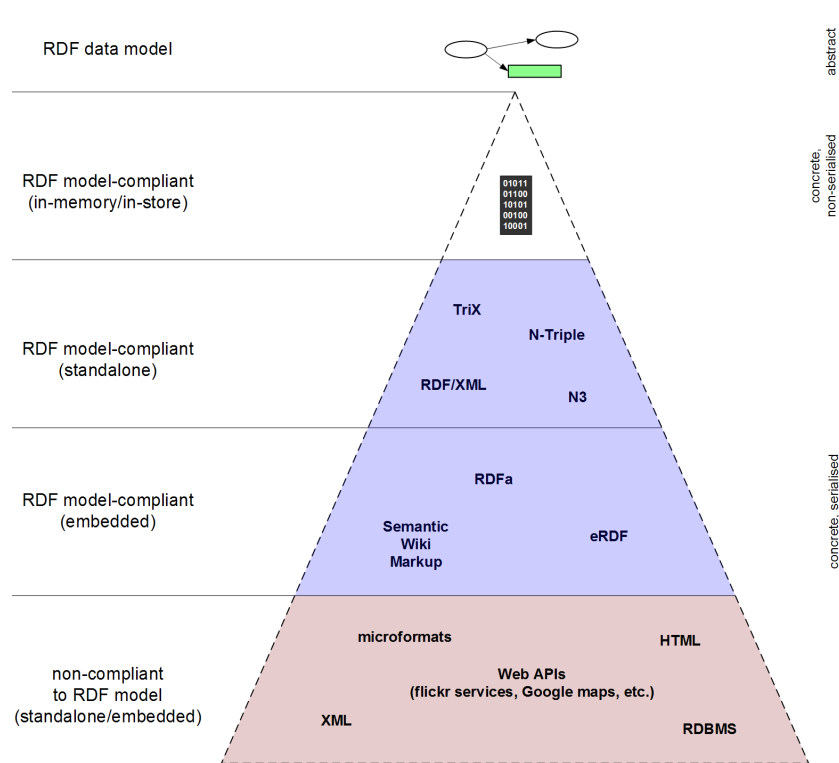
<sup>7</sup> <http://www.w3.org/Submission/2007/02/>

## 2.2 Deployment Issues

Due to the widespread objections against the 'official' concrete RDF syntax—RDF/XML [8]—and prosperous activities arising from the microformats community, the deployment of RDF-based metadata demands a critical review.

RDFa [1, 2] (*RDF* in *attributes*) is a W3C draft, currently in the process of being finalised. Roughly speaking RDFa allows to embed an RDF Graph into an (X)HTML document using attributes (as `@about`, `@href`, `@rel`, etc.).

In [9] we elaborated on RDF representations and respective performance and scalability issues, as summarised in Fig. 1. There, we concluded that RDFa is a concrete serialisation syntax of the RDF model, embedded in HTML, and throughout usable for exchanging metadata on the Web.



**Fig. 1.** The RDF representation pyramid, as presented in [9].

For example, the XHTML+RDFa snippet depicted in Fig. 2 would yield the triple shown in Fig. 3. The resulting RDF triples form the basic information asset, an RDF-aware agent operates on.

```

<h3>Tuesday, 10 July</h3>
<div xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
      xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
      xmlns:sioc="http://rdfs.org/sioc/ns#"
      xmlns:dc="http://purl.org/dc/elements/1.1/"
      xmlns:dcterms="http://purl.org/dc/terms/">
  <div href="sioc:Post" rel="rdf:type"
        about="http://lists.w3.org/.../2007Jul/0077.html">
    <a href="http://lists.w3.org/.../2007Jul/0077.html">
      <span property="dc:title" datatype="xsd:string">
        Re: [RDFa] ISSUE-28: following your nose
          to the RDFa specification
      </span>
    </a>
    <span property="dc_terms:created"
          content="2007-07-10" datatype="xsd:date"/>
    <span rel="sioc:has_container"
          href="http://lists.w3.org/.../public-rdf-in-xhtml-tf"/>
    <span property="dc:creator"
          content="Ben Adida" datatype="xsd:string"/>
    (Ben Adida)
  </div>
</div>

```

**Fig. 2.** A sample RDFa+XHTML snippet about a mailing list post.

```

@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix dc: <http://purl.org/dc/elements/1.1/> .
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix sioc: <http://rdfs.org/sioc/ns#> .

<http://lists.w3.org/.../2007Jul/0077.html> rdf:type sioc:Post ;
dc:creator "Ben Adida"^^xsd:string;
dc:title "Re: [RDFa] ISSUE-28: following your nose
         to the RDFa specification"^^xsd:string;
dcterms:created "2007-07-10"^^xsd:date;
sioc:has_container <http://lists.w3.org/.../public-rdf-in-xhtml-tf> .

```

**Fig. 3.** Resulting triples of the sample XHTML+RDFa snippet.

### 3 System Description

As discussed in [9], one has to carefully select metadata sources regarding their nature. Mailing list archives have two distinguishing properties, one can exploit when writing a Semantic Web application:

1. Depending on the granularity (typically on the month-level), the content of a container is understood to be dynamic;
2. Past time units (e.g. months) are considered to be static w.r.t. the content and the metadata, as no new entries can be added.

Hence, we have a semi-dynamic source that allows for well-performing RDFizing for all 'closed' issues, and a dynamic representation of 'current' ones<sup>8</sup>.

Fig. 4 shows the principle architecture of `mle`, which was designed based on the above mentioned observations:

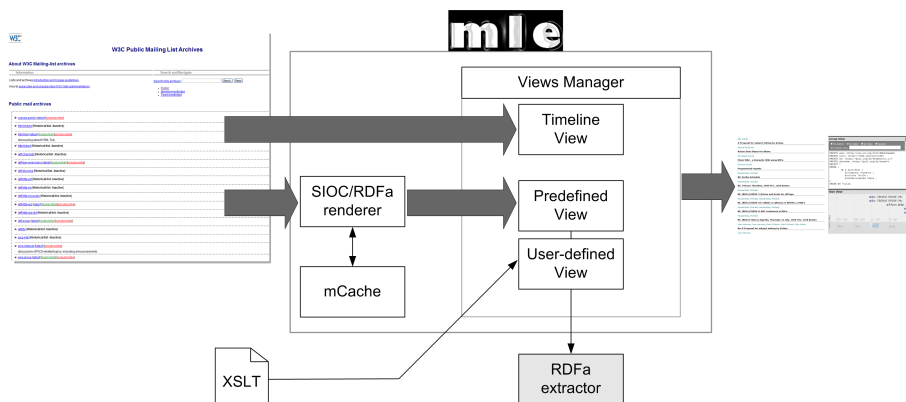


Fig. 4. `mle` System Architecture.

The **SIOC/RDFa** renderer is the core of `mle`, basically generating RDFa in XHTML by applying XSLT onto the input mailing list archive available in XHTML. To speed up processing, the so called **mCache** enables to store the output of the the SIOC/RDFa renderer, and recall it in case the according item is closed (hence the content does not change anymore).

A **view** in our understanding is a query—represented in SPARQL [10]—defining the filter criteria, along with a style sheet (again in XSLT) that provides for the visual layout. The actual result formatting can only be done after executing the SPARQL query. For the SPARQL query, the embedded RDF metadata needs to be extracted; currently this is done by invoking an external service<sup>9</sup>.

<sup>8</sup> Note that it is possible to further improve the performance in terms of limiting the current items down to the day level, etc.

<sup>9</sup> <http://torrez.us/services/rdfa/>

### 3.1 Features & Usability

In the following, the main features of `mle` are listed:

- RDFize mailing list archives, resulting in a self-contained and self-explanatory XHTML document using RDFa;
- Allow applying (user defined) views on query results;
- Provide for alternative views on the mailing list, implemented via a timeline.

As a matter of fact, `mle` targets not only to support machine processing of the enhanced mailing list archive, but also enables human users to easily use and adapt it; Fig. 5 depicts the tool in action.

The screenshot shows the `mle` web application interface. At the top, there is a navigation bar with 'home', 'doc', and 'about' links. Below this is a search bar for 'W3C Mailing List URL' with a search button. The main content area displays a list of mailing list posts, grouped by date (Friday, 13 July; Thursday, 12 July; Wednesday, 11 July; Tuesday, 10 July). On the right side, there is a 'Group View' panel with a search bar and a 'Get Custom view' button. Below this, there is a 'Time View' panel showing a calendar grid for the month of July, with a legend for different categories of posts. The interface is designed to allow users to interact with and filter mailing list data.

Fig. 5. `mle` in action: Applying a user defined view.

## 4 Conclusions & Outlook

In this research we have proposed a new way of interacting with, and processing of mailing list archives. The automatic processing of mailing list archives (and related 'social Web sources' as blogs, etc.) turns out to be a vital feature for executives, policy makers, and market researchers. Making the semantics explicit will significantly pave the currently quite rocky way to gather information such as a company image, or public knowledge related to companies or products. Further research such as competitor analysis is made possible.

To allow Web-based mailing list archives to successfully enter the Semantic Web, we propose to enhance them with SIOC-based metadata, and embed the metadata using RDFa.

Our main finding was that in principle `mle` is very handy for supporting routine work. However, results only may be delivered quickly, when a certain search depth is not exceeded. The so called 'Dig Deep' feature that would basically use the information present in the mail header, and try to look up user-related information based on the mail address was deactivated in the first phase exactly due to this reason. Further experiments are pending.

Future extensions of `mle` are in discussion. These may include the integration of topic-sensitive annotations, adding more information about the author of a post, and making the viewing sub-system more flexible. Finally it is planned that the Time View is coupled more tightly with the other views to allow graphical querying of the mailing list archive.

## 5 Acknowledgements

Parts of the research presented herein was carried out in the "Understanding Advertising" (UAd) project<sup>10</sup>, funded by the Austrian FIT-IT Programme, and has been partially supported by the European Commission under the IST research network of excellence K-Space of the 6<sup>th</sup> Framework programme.

Special thanks go out to Uldis Bojars (DERI Galway) for his support regarding SIOC.

## References

1. B. Adida and M. Birbeck. RDFa Primer 1.0. W3C Working Draft, W3C Semantic Web Deployment Working Group, 2007.
2. B. Adida and M. Hausenblas. RDFa Use Cases: Scenarios for Embedding RDF in HTML. W3C Working Draft, W3C Semantic Web Deployment Working Group, 2007.
3. D. Brickley and L. Miller. FOAF Vocabulary Specification. <http://xmlns.com/foaf/0.1/>, 2004.
4. U. Bojars and J. G. Breslin. SIOC Core Ontology Specification. <http://rdfs.org/sioc/spec/>, 2007.
5. S. Fernandez, D. Berrueta, and J.E. Labra. Mailing Lists Meet The Semantic Web. In *Proc. of the BIS 2007 Workshop on Social Aspects of the Web*, Poznan, Poland, 2007.
6. U. Bojars, J.G. Breslin, and A. Passant. SIOC Ontology: Applications and Implementation Status. W3C Member Submission 12 June 2007, W3C Member Submission, 2007.
7. Dublin Core Metadata Initiative. <http://purl.org/dc/>, 1999.
8. D. Beckett and B. McBride. RDF/XML Syntax Specification (Revised). W3c recommendation, World Wide Web Consortium, 2004.

<sup>10</sup> <http://www.sembase.at/index.php/UAd>

9. M. Hausenblas, W. Slany, and D. Ayers. A Performance and Scalability Metric for Virtual RDF Graphs. In *3<sup>rd</sup> Workshop on Scripting for the Semantic Web (SFSW07)*, Innsbruck, Austria, 2007.
10. E. Prud'hommeaux and A. Seaborne. SPARQL Query Language for RDF. W3C Candidate Recommendation, RDF Data Access Working Group, 2007.