A semantic framework for modelling quotes in email conversations[*]

Alexandre Passant, Antoine Zimmermann, Jodi Schneider
Digital Enterprise Research Institute
National University of Ireland, Galway
firstname.lastname@deri.org

John G. Breslin
School of Engineering and Informatics
National University of Ireland, Galway
john.breslin@nuigalway.ie

ABSTRACT

Quoting is a common practice in online conversations, notably in email discussions and bulletin boards. Despite many applications that offer special functionalities to handle them, there is no agreement on how to semantically represent these quotes, so that they could be queried and analysed uniformly whatever the original application is. While various Semantic Web models, notably SIOC, have become popular for interlinking, exporting and exchanging information about online communities and their conversations, they do not provide means to model such quotes. In this paper, we present (i) an OWL2 extension of SIOC for representing quotes and (ii) a framework for automated quotes extraction from email archives. In addition, we demonstrate the utility of such formal representation for querying email archives with finer-grained queries and for mining social networks based on the argumentative structure of email conversations.

1. INTRODUCTION

While many conversations have moved from traditional applications to Web 2.0 services such as blogs, microblogging, *etc.*, mailing lists and bulletin boards are still widely used. Especially, the formers are actively used for technically-related discussions, from software development to standardisation work — for example, the W3C hosts 325 active mailing lists[1]. In that context, some threads may quickly generate many responses, especially when it comes to issues that have not yet reached an agreement, *e.g.*, when discussing features that should go to a recommendation process. So far, most mailing list archive services allow only, from one post, to browse all the related answers. However, people tend to quote and reply to particular sub-parts of the conversations, especially in technical discussions. Hence, if participants are interested in a particular aspect of the discussion (*e.g.*, an ISSUE raised in a Working Group), they have to browse all replies to find which ones are related.

To make such process easier, there is a need to (1) extract quotes from these email discussions; and (2) provide a common modelling scheme for such, so that they can be more efficiently identified and queried. Since it comes to data modelling and knowledge representation, the Semantic Web and the use of ontologies are particularly well suited for such purposes. In this paper, we provide a new approach to model quotes (and their replies) that appear in email discussions. To achieve this goal, we define a framework that consists in (1) a lightweight model to represent such quotes, defined as a module of the SIOC Ontology[2], hence re-using its main classes and properties and based on the newly standardised OWL2-RL; and (2) an application to extract quotes from existing mailing lists archives and model them using the aforementioned ontology.

The rest of this paper is organised as follows. In Section 2, we present an analysis of two W3C mailing lists over a 3-month period, identifying the different reply patterns used in such conversations and how quotes are used in this context. In Section 3, we describe our model for capturing the structure of these quotes in a machine-readable way, based on a lightweight SIOC extension. We also present how we modelled particular axioms of the ontology by taking advantage of OWL2-RL, so that some properties of the quotes can be entailed from the discussions themselves. In Section 4, we discuss related applications, *i.e.*, the process of extracting quotes from email archives and how these semantically-enhanced quotes can be used for retrieving information and identifying social networks in email conversations. We finally present related work in Section 5, before concluding the paper.

2. QUOTING: A COMMON REPLY PATTERN IN MAILING LISTS

2.1 Quotes in email conversations

Using quotes in online conversations is a common practice, especially in bulletin boards and email conversations. While many of these conversations have recently moved from these services to, *e.g.*, blogs or microblogs, they are still widely used in some particular communities. For instance, clipping or quoting is a hugely popular trend in the Korean blogosphere [17]. In addition, mailing lists — which we will focus on in this paper — are still considered as the primary means

[*] This work has been funded by Science Foundation Ireland under Grant No. SFI/08/CE/I1380 (Líon 2).

[1] http://lists.w3.org/

[2] http://rdfs.org/sioc/spec/

of discussing some standardisation work and related issues, notably in W3C.

Generally, when people reply to each other in that context, they include their answer inside the email, by directly quoting each part they are replying to, as represented in Figure 1[3]. Such practice, also known as *inline reply*, is common in computer science communities due to the related netiquette.

```
> One of the good thing with this approach is that
     the "patterns" of
> information held in the target document can be
     arbitrarily complex -

As far as I can remember, all the examples that
     people have given
could be addressed with a simple property-based
     approach.
```

Figure 1: An *inline reply* in an email conversation

However, in other contexts, different behaviours may happen, such as people copying and pasting various emails in a new one to answer to different emails at the same time[4]. Yet, we have not observed such practice in the emails we analysed, as we will now discuss.

2.2 Analysis of existing mailing lists

In order to figure out how inline replies are used in email conversations, we manually analysed email discussions from January 2010 to March 2010 from 2 public mailing lists hosted by W3C. We focused on the Linking Open Data (LOD) mailing list[5] and the Media Fragments (MF) Working Group mailing list[6]. A total of more than 800 emails from the two mailing lists were analysed, from which we studied the structure of replies and quotes according to the following classification:

- **Non-reply messages**, *i.e.*, first message in a thread (sometimes not leading to further discussions, *e.g.*, conference CfP);
- **Replies w/o message**, *i.e.*, replies that do not contain any quote (authors removing the original message when replying);
- **Replies on top**, *i.e.*, replies that quote the entire original message at the end;
- **Inline replies**, *i.e.*, replies in which the user quotes the original message, as in the previous listing.

While relatively different in terms of the purpose they serve — the former being a public discussion list, the latter being a Working Group archive (with ISSUES, resolutions, *etc.*) — and in spite of having only one poster in common (during the given period) we observed similar behaviours on these two mailing lists. The related statistics can be found in Table 1, and we can then observe that more than 72% of messages (81% on `public-lod`) are **replies**. Among these replies, more than 63% (72% on `media-fragments`) are **inline replies**, while only a small percentage (less than 0.5%) are **replies without message**, the rest being **replies on top**. In both cases, we therefore observed that more than 50% of the whole messages are **inline replies** and contained the aforementioned quoting patterns.

Table 1: Analysis of two W3C mailing lists over a 3-months period

	LOD	MF
Total messages	554	275
Total users	103	16
Non-replies	105 (18.95%)	75 (27.27%)
Replies	449 (81.05%)	200 (72.73%)
Replies on top	162 (36.08%)	55 (27.50%)
Inline replies	287 (63.92%)	145 (72.50%)
Replies w/o message	15 (00.33%)	7 (00.35%)

While we focus on two particular lists in this study, it has been observed that similar patterns appear in other communities [1]. This could be explained by two main reasons. First, replies are normally motivated by a specific part of the content of the replied email, thus the quotations highlight what brought the respondent to reply [9]. Second, email client software is usually configured to automatically include a quotation of the replied email whenever the reply button is used. As a result, even though some people are not keen on quoting others, their reply will contain a quotation anyway.

We also identified some common patterns in terms of distribution of users and posts within these two mailing lists. The most active user on the LOD mailing list contributed to 125 posts on a total of 554, *i.e.*, 22.5% while on the Media Fragments list, the most active participant created 71 posts out of 275, hence 27%. In addition, we observed that the top 24 users of the `public-lod` mailing list (23.30%) contributed to 395 posts (71.30%) while on the `media-fragments` one, the top 4 (25%) contributed to 184 posts (66.90%). In both cases, this lead to a Pareto distribution, a phenomena that can be observed in various social networking applications and in other mailing lists [21].

3. A MODEL TO REPRESENT QUOTES IN ONLINE CONVERSATIONS

During the past few years, there has been a lot of work around the Social Semantic Web [4]. Among others, part of this research field focused on defining RDF(S)/OWL modelling capabilities to enable semantic representation of social data, capturing the various aspects of online communities, their interactions and the content they share. For instance, FOAF — Friend Of A Friend [5] — aims at representing online identity and social networking aspects, while SIOC — Semantically-Interlinked Online Communities [3] — focuses on representing the interactions happening in online communities and the content generated within. However, while SIOC can be used to represent email discussions, notably

[3] From `http://lists.w3.org/Archives/Public/public-lod/2009Jan/0002.html`.

[4] `http://groups.google.com/group/code4libcon/msg/82989c67a49fb065?dmode=source` shows an example of such.

[5] `http://lists.w3.org/Archives/Public/public-lod/`

[6] `http://lists.w3.org/Archives/Public/public-media-fragment/`

threads (using the `sioc:Thread` class) and previous/next replies (using the `sioc:previous_by_date`/`sioc:next_by_date` properties), it does not provide finer-grained representation of the quoting patterns that we previously introduced. Indeed, SIOC focuses on modelling conversations at the thread or item level, but considers items, *i.e.*, emails in our case, as simple plain blocks. Therefore, there is a need to provide a specific model to enable such finer-grained modelling of the quoting patterns.

Thus, and rather than designing our own model to represent these quotes, we designed a lightweight SIOC module named "SIOC Quotes", available at `http://rdfs.org/sioc/quotes`[7] and validated through Vapour[8] to ensure it complies with the best practices for publishing RDF vocabularies. We decided to base our model on SIOC since (1) it is already widely deployed on the Web, and (2) it provides basic classes and properties that can be used in this extension, such as `sioc:Item` or `sioc:has_creator`, as we shall see later. That way, elements such as the quoted message, or the author of a reply can be represented with SIOC.

3.1 The SIOC Quotes vocabulary

Our model is represented by Figure 2 and consists of the following classes and properties:

- `quotes:Block` (`owl:Class`, subclass of `sioc:Item`) represents a block, *i.e.*, the combination of a quote from the answered item and the reply to this quote. The previous listing (Figure 1) represents such a block;
- `quotes:Quote` (`owl:Class`, subclass of `sioc:Item`) represents the quote part in such blocks;
- `quotes:Response` (`owl:Class`, subclass of `sioc:Item`) represents the response part in such blocks.
- `quotes:has_block` (`owl:ObjectProperty`) links an instance of `sioc:Item` to an instance of `quotes:Block`; inverse of `quotes:block_of`;
- `quotes:has_quote` (`owl:ObjectProperty`) links an instance of `quotes:Block` to an instance of `quotes:Quote`; inverse of `quotes:quote_of`;
- `quotes:has_response` (`owl:ObjectProperty`) links an instance of `quotes:Block` to an instance of `quotes:Response`; inverse of `quotes:response_of`;

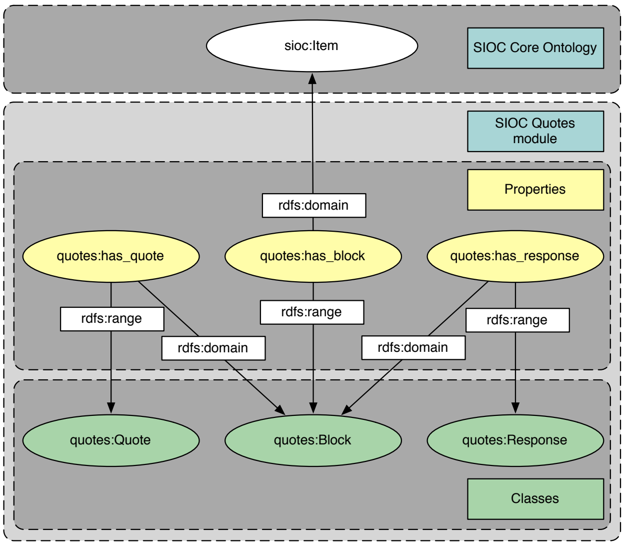


Figure 2: The SIOC Quotes module

```
@prefix sioc: <http://rdfs.org/sioc/ns#> .
@prefix quotes: <http://rdfs.org/sioc/quotes#> .
@prefix : <http://example.org/> .

<http://example.org/public-lod/2009Jan/0002>
  a sioc:Item ;
  sioc:has_block :block .

:block a quotes:Block ;
  quotes:has_quote :quote ;
  quotes:has_response :response .

:quote a quotes:Quote ;
  sioc:content "One of the good thing with this
approach is that the \"patterns\" of information
held in the target document can be arbitrarily
complex" .

:response a quotes:Response ;
  sioc:content "As far as I can remember, all the
examples that people have given could be addressed
with a simple property-based approach." .
```

Figure 3: Modelling quotes in RDF.

This model can thus be used to represent the statements of our first example, as represented in Figure 3. In addition, while the three classes are independent, we did not introduce any formal disjointness (*e.g.*, via `owl:disjointWith`) between them in our model. Indeed, in a thread, a `Block` can later become a `Quote` in the reply. This ensures a better flexibility of the model. Moreover, we intentionally do not add cardinality restrictions both for efficiency and flexibility. An application could further restrict the use of the properties, for instance by enforcing that a `Block` have exactly one `Quote` and one `Response`. However, some people may quote several parts of an email before writing a single overall response. This can be formalised coherently with our model.

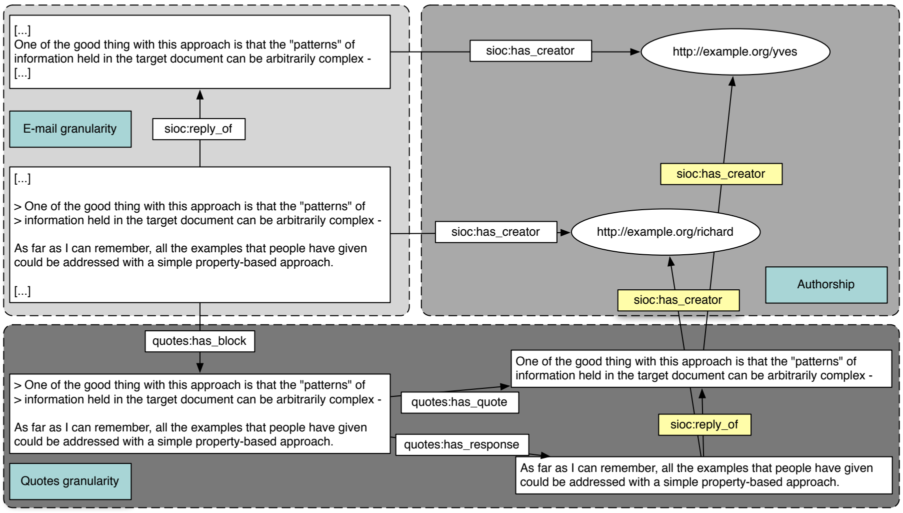
Finally, although our main motivation and case study are based on mailing lists and their related discussions, the model can be used for quotes in bulletin boards, where the practice is also quite common — especially as some bulletin board applications have a dedicated feature in their engine — as well as any application involving quoting patterns.

3.2 Enriching the model using OWL2 axioms

The recent adoption of OWL2 [14], in October 2009, as the new W3C recommendation for the Web Ontology Language opens new areas in ontology engineering. OWL2 extends the previous version [22] with new semantic features, syntactic variants, and sub-languages that are meant to cover practical needs better than the former languages OWL-Lite, OWL DL and OWL Full.

[7] In the rest of the paper, we are using the prefix `quotes` to refer to the namespace `http://rdfs.org/sioc/quotes#`. Similarly, we are using common abbreviated prefixes such as `rdf`, `rdfs`, `owl` and `sioc`

[8] `http://validator.linkeddata.org/`


[...]
One of the good thing with this approach is that the "patterns" of information held in the target document can be arbitrarily complex -
[...]

[...]
> One of the good thing with this approach is that the "patterns" of
> information held in the target document can be arbitrarily complex -

As far as I can remember, all the examples that people have given could be addressed with a simple property-based approach.
[...]

> One of the good thing with this approach is that the "patterns" of
> information held in the target document can be arbitrarily complex -

As far as I can remember, all the examples that people have given could be addressed with a simple property-based approach.

One of the good thing with this approach is that the "patterns" of information held in the target document can be arbitrarily complex -

As far as I can remember, all the examples that people have given could be addressed with a simple property-based approach.

Figure 4: Inferring new statements when representing quotes

Three profiles are defined in the specifications which correspond to sub-languages with interesting computational properties [23]. We are especially interested in the OWL2 RL profile, which has two advantages: (1) it can be efficiently implemented as a rule-based system and (2) all standard reasoning tasks are in the polynomial complexity class. In principle, this means that, if an ontology stays within the boundaries of OWL2 RL, a dedicated RL reasoner should guarantee reasonable response time for the common reasoning tasks. We will next show how to take advantage of a new OWL2 construct, namely property chain axioms, to improve the expressiveness of our ontology module, while preserving its lightweight nature by staying within the RL profile.

3.2.1 OWL 2 property chains

Property chain axioms can interestingly enrich a concise knowledge base with new facts based on a sequence of predicates/objects $s \xrightarrow{p} x \xrightarrow{q} o$ that links instances s and o. Notably, thanks to such an expressive construct, a coarse-grained email/reply structure can be automatically inferred from a fine-grained quote/response structure. Similarly, it can be used to infer statements about the authors of the `Quote` and `Response` instances, while relying only on the authors of the related `sioc:Item` instances, as can be seen in the Figure 4 — relationships with a yellow background being the ones that can be entailed from existing statements. To make the description of the property chains concise, we write them in Turtle[9] using the RDF serialisation of OWL [24], as in the following generic example (Figure 5).

```
:prop owl:propertyChainAxiom (
      :firstProp
      :secondProp
   ) .
```

Figure 5: Generic representation of property chains in Turtle.

More than two properties can occur in a chain, as we will see later in our ontology. Modelling the chains in the standard Web Ontology Language allows one to use a general purpose reasoner such as Pellet[10] or HermiT[11]. Additionally, since both the SIOC ontology and our `Quotes` extension only use axiom constructs that are in OWL2 RL, specifically optimised RL reasoners[12] such as the Oracle Database 11g OWL Reasoner[13] can be applied to the data.

3.2.2 Property chain axioms used in our vocabulary

Within a block, as represented earlier, a response part is a reply to the quote part in that same block. Yet, instead of modelling this explicitly, it can be inferred from the structure of the block itself: as soon as a block contains a reply (using `has_reply`) and a quote (using `has_quote`), the reply is a reply of the quote, and this entailment can be represented using a property chain axiom. However, here the `has_quote` and `has_response` do not form a proper path *per se*, since it combines two properties applied to the same subject. To get a property chain, we need to use an inverse property. This is the reason we introduced the inverse properties `quote_of`, `response_of` and `block_of` that we use in Figure 6 (this being the first property chain on the listing).

We also wanted to model the relationship between block parts (*i.e.*, instances of `Quote` and `Response`) and their authors using the authorship definitions represented at the email level. That way, authorship and provenance information, particularly relevant when it comes to conversations, can be inferred from the original emails: the creator of a response in a block is the same person as the author of the reply message itself. Once again, we relied on the inverse property of `sioc:has_creator`, *i.e.*, `sioc:creator_of` (defined in the SIOC Core Ontology), to get a proper chain. Finally, the author of a `Quote` is the same as the author of

[9] `http://www.w3.org/TeamSubmission/turtle/`

[10] `http://clarkparsia.com/pellet`

[11] `http://hermit-reasoner.com/`

[12] `http://www.w3.org/2007/OWL/wiki/Implementations`

[13] `http://www.oracle.com/technology/tech/semantic_technologies/index.html`

```
sioc:has_reply owl:propertyChainAxiom (
      quotes:quote_of
      quotes:has_response
   ) .
sioc:creator_of owl:propertyChainAxiom (
      sioc:creator_of
      quotes:has_block
      quotes:has_response
   ) .
sioc:creator_of owl:propertyChainAxiom (
      sioc:creator_of
      sioc:has_reply
      quotes:has_block
      quotes:has_quote
   ) .
```

Figure 6: Property chains in the `Quotes` ontology.

the item that is replied to, that we also modelled through a property chain.

Thanks to these various axioms, one can limit the knowledge bases of extracted quotes to a minimal number of facts, some additional statements being inferred using the property chains axioms combined with the initial knowledge base of email conversations. We believe that this is an interesting way to design and build Social Web ontologies and applications using lightweight models and knowledge bases, while making use of powerful inference capabilities without heavy complexity (since we stay in the RL profile), bringing a new reasoning perspective into the Social Web. Additionally, inferred statements can be directly materialised during the extraction process using a rule-based engine, which would save time and resources at query time.

4. APPLICATIONS

4.1 Extracting quotes from mailing list archives and representing them in RDF

In order to demonstrate the interest of the approach, we designed a system extracting quotes from mailing list archives and representing them in RDF using the aforementioned extension, combined with the SIOC Core Ontology. The system consists in an extractor written in Python, that takes as an input a directory containing mailing list archives formatted using the mbox format [15]. The process is relatively straightforward: it parses all files in the directory, and for each email, it extracts its title, author, content and date as well as the quotes, based on regular expression patterns. The system focuses only on extracting *inline queries* only, so it does not concentrate on the basic structure of the conversation, since such processing can already be achieved using for instance SWAML [10].

We applied the system to the complete archives of the OWL Working-Group mailing list[14], from Sep. 2007 to Apr. 2010. Archives have been downloaded from the W3C mailing list server, and have been processed through the previous script. The process took around 40 seconds on a MacBook Pro Intel Core 2 Duo 2,53GHz with 4Go Ram, and 7,074 messages have been parsed, including 5,139 replies (*i.e.*, 72%). Among these replies, 2,168 messages have been identified as containing inline replies (*i.e.*, 30% of the total messages, comparatively less than in the other lists we analysed, yet almost a third of the messages), leading to 10,146 blocks being created. We then describe two use-cases on how we took advantage of these extracted triples for (1) enhancing information querying and (2) mining social networks.

4.2 Identifying relevant elements in email discussions

One main advantage of modelling quotes — and not only emails — is that queries can identify particular quoted patterns, instead of searching for information in the complete email. For instance, one can easily find replies to quotes mentioning relevant ISSUEs, while standard mailing list browsers and clients provide only plain-search at the email level. Furthermore, such patterns can be extended to find blocks that contain a particular text string both in the quoted pattern and in the reply, as finding all people discussing a particular ISSUE, using a "+1" pattern — which can be particularly useful when trying to see current opinion of members of a Working Group about a particular topic. Finally, another important aspect, that comes as a side effect of the use of RDF(S)/OWL technologies to model these quotes is that the queries are modelled using SPARQL. Hence, the following listing (Figure 7) represents a SPARQL query corresponding to the previous use-case.

```
PREFIX quotes: <http://rdfs.org/sioc/quotes#>
PREFIX sioc: <http://rdfs.org/sioc/ns#>

SELECT ?quote_content ?reply_content
WHERE {
  ?quote a quotes:Quote ;
     sioc:content ?quote_content ;
     sioc:has_reply ?reply .
  ?reply sioc:content ?reply_content .
  FILTER REGEX(?quote_content, "ISSUE")
  FILTER REGEX(?reply_content, "\\+1")
}
```

Figure 7: Identifying "+1" replies to ISSUEs in the OWL-WG mailing list (query).

By applying this query to the OWL2 mailing list dataset that we extracted from the mbox files, we were able to identify 2 relevant blocks, including the one from Figure 8.

4.3 Community detection and dynamics

A second case-study that we implemented concerns the use of the model for community detection. While SIOC can be used to directly analyse replies between messages at an email level (thanks to the `sioc:has_reply` links), using the quotes model allows finer-grained mining of social networks and communities. Our model gives a stronger impression of how a community interacts since instead of having a single link for each message/reply pattern, we can represent the various quotes and responses, thus capturing more closely the argumentative discussions between individuals, as showed by [2] when analysing emails related to open-source projects: "The thematic coherence of the discussion [...] is better represented by the quotation-based links [...] than by the reply-to links".

Figure 9 represents a graph that have been extracted from

[14] `http://lists.w3.org/Archives/Public/public-owl-wg/`

```
{
  "quote_content" : {
    "type": "literal",
    "value": "<![CDATA[\n>OK, then let's at least
         try to move forward on ISSUE-106:\n>\n>
         PROPOSED:  Resolve ISSUE-106 by making the
          owl2 namespace the same as\n>\t   the owl
          namespace.\n>\n>pe
ter\n\n]]>"
  },
  "reply_content" : {
    "type": "literal",
    "value": "<![CDATA[\n+1.\n\nOtherwise, in the
         future we would collect different
         namespaces for OWL, OWL\n2, OWL 2.01 (scnr
          ;-)), and so on. This wouldn't scale well
         , and would lead\nto use
r confusion, I suppose.\n\nCheers,\nMichael\n\n]]>"
  }
},
```

Figure 8: Identifying "+1" replies to ISSUEs in the OWL-WG mailing list (partial answer).

a particular thread (in that case, from the LOD list), where we gave more weight to the edges that appear often (*i.e.*, representing lots of quote/response patterns between the authors). This shows where the argumentation happens, and between which users — the edges having more weight make the nodes be closer in the graph — such patterns could have also been identified by applying an HITS algorithm to the graph [19]. In addition, we could go further and show how such patterns evolve in time, for instance to see in a community which people are involved in the discussion and how discussion moves from one sub-network to another, which might be useful when consensus must be reached in communities.

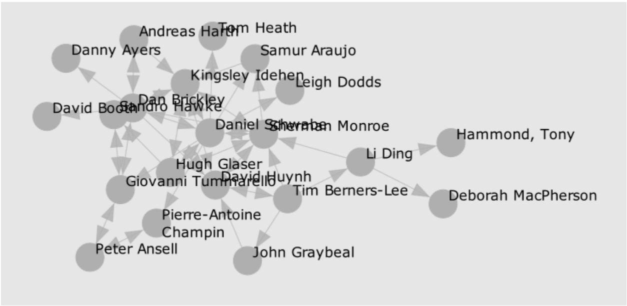


Figure 9: Identifying the network of people in a particular thread

5. RELATED WORK

Various studies and tools have been provided in the email realm [8], mainly focusing on solving the email overload issue [29, 7] and how people argue in email conversations. Barcellini [1] studies quotations in two discussions in the Python open source community, and provide quotation-based visualisations of messages, discussing the type and depth of quotations. Message threads have also been analysed in detail in other formats such as BBS systems [25] and Computer-Mediated Communication (CMC) in general [12]. Venolia and Neustaedter [28] deployed an application capable of handling quotes in CMC but did not consider ways to semantically enhance it.

Regarding semantic representation of quotes, we have seen that quotations typically occur in argumentative discussions when discussants are trying to come to agreement. Thus, argumentation models such as IBIS [20] and the Zeno Argumentation Framework [13] and implementations such as gIBIS [6] and Compendium [27] may provide inspiration for future models of collaborative systems. In some cases (*e.g.*, Zest [30]) argumentation has already been explicitly brought into email, while not leading to any RDF model that we could have used straightforward. Particularly relevant to our context is also the work by Fraser *et al.* [11], defining a proposal for an argumentation ontology for mailing lists, in particular in the W3C context. However, no model has been provided.

From a modelling perspective, more closely to the Social Semantic Web realm and in addition to SIOC that serves as a base for our work, several projects can be mentioned. SWAML [10] focuses on the representations of mailing lists in RDF, and the various classes and properties it has introduced are integrated in SIOC — while the SWAML application itself (which we have used in our implementation) is still active. mle [16] also focuses on representing semantically-enhanced mailing lists in RDFa. However, they do not focus on the quoting patterns; rather they stay at a general email and conversation level. Hence, our approach bridges a gap between common needs for argumentation representation, and existing models for social data.

Finally, in the realm of mailing lists and the Semantic Web, we shall also mention the work done regarding Semanta[15] [26], a semantic email client capable of analysing text patterns to automatically detect and create events to ease the replying process, as well as [18] which focuses on automated semantic email addressing. We believe that our model can efficiently be combined with that first tool, notably to enhance the semantic description of messages and how they are exchanged between applications.

6. CONCLUSION

In this paper, we detailed a semantic framework for representing quotes in email conversations. We first described a lightweight SIOC module using OWL2-RL and aimed at modelling quotes in online conversations in RDF, making them machine processable using standard Semantic Web toolkits and languages, as SPARQL. Then, we showed how quotes can be extracted from existing email archives and represented using this module. We also provided use-cases showing how information querying and social network identification in mailing lists can be enhanced thanks to this semantic framework.

Moreover, though we focus on a use-case centred around email discussions, notice that this model can be applied to other applications involving such quoting patterns, such as bulletin boards, instant messaging and hybrid applications like Google Wave. Such applications will be taken into account in our future work regarding modelling quotes, enabling integration of various components from the Social Web and emphasising their argumentation and discussion capabilities at a finer-grain that what is currently available.

[15] http://smile.deri.ie/projects/semanta/

7. REFERENCES

[1] Flore Barcellini, Françoise Détienne, Jean-Marie Burkhardt, and Warren Sack. A socio-cognitive analysis of online design discussions in an Open Source Software community. *Interacting with Computers*, 20(1):141–165, 2008.

[2] Flore Barcellini, FranÃğoise Détienne, Jean-Marie Burkhardt, and Warren Sack. A study of online discussion in an Open-Source community: reconstructing thematic coherence and argumentation from quotation practices. In *Communities and Technologies 2005: Proceedings of the Second Communities and Technologies Conference*, pages 301–320. Springer, 2005.

[3] John G. Breslin, Andreas Harth, Uldis Bojārs, and Stefan Decker. Towards Semantically-Interlinked Online Communities. In *Proceedings of the 2nd European Semantic Web Conference (ESWC2005)*, volume 3532 of *Lecture Notes in Computer Science*, pages 500–514. Springer, 2005.

[4] John G. Breslin, Alexandre Passant, and Stefan Decker. *The Social Semantic Web*. Springer, 2009.

[5] Dan Brickley and Libby Miller. FOAF Vocabulary Specification. Namespace Document 2 Sept 2004, FOAF Project, 2004. http://xmlns.com/foaf/0.1/.

[6] Jeff Conklin and Michael L. Begeman. gibis: a hypertext tool for exploratory policy discussion. In *CSCW '88: Proceedings of the 1988 ACM conference on Computer-supported cooperative work*, pages 140–152, New York, NY, USA, 1988. ACM.

[7] Laura A. Dabbish and Robert E. Kraut. Email Overload at Work: An Analysis of Factors Associated with Email Strain. In *CSCW '06: Proceedings of the 2006 20th anniversary conference on Computer supported cooperative work*, pages 431–440, New York, NY, USA, 2006. ACM.

[8] Nicolas Ducheneaut and Leon A. Watts. In Search of Coherence: A Review of E-Mail Research. *Human-Computer Interaction*, 20(1):11–48, 2005.

[9] Kerstin Severinson Eklundh and Clare MacDonald. The Use of Quoting to Preserve Context in Electronic Mail Dialogues. *IEEE Transactions on Professional Communication*, 37(4):197–202, 1994.

[10] Sergio Fernández, Diego Berrueta, and Jose E. Labra. Mailing Lists Meet The Semantic Web. In *Proceedings of the BIS 2007 Workshop on Social Aspects of the Web (SAW2007)*, volume 245 of *CEUR Workshop Proceedings*, pages 45–52. CEUR-WS.org, 2007.

[11] Colin Fraser, Harry Halpin, and Kavita Elisheba Thomas. Developing an argumentation ontology for mailing lists. In *Artificial Intelligence: Methodology, Systems, and Applications, 12th International Conference, AIMSA 2006*, volume 4183 of *Lecture Notes in Computer Science*, pages 150–161. Springer, 2006.

[12] Tianjun Fu, Ahmed Abbasi, and Hsinchun Chen. A hybrid approach to web forum interactional coherence analysis. *Journal of the American Society for Information Science and Technology*, 59(8):1195–1209, 2008.

[13] Thomas F. Gordon and Nikos Karacapilidis. The Zeno argumentation framework. In *Proceedings of the 6th international conference on Artificial intelligence and law*. ACM, 1997.

[14] W3C OWL Working Group. OWL 2 Web Ontology Language: Document Overview. W3C Recommendation 27 October 2009, W3C, 2009.

[15] Eric A. Hall. The application/mbox Media Type. Request for comments: 4155, Internet Engineering Task Force, 2005. `http://www.ietf.org/rfc/rfc4155.txt`.

[16] Michael Hausenblas and Herwig Rehatschek. mle: Enhancing the Exploration of Mailing List Archives Through Making Semantics Explicit. In *Semantic Web Challenge*, volume 295 of *CEUR Workshop Proceedings*, 2007.

[17] Jaz Hee jeong Choi. Living in Cyworld: Contextualizing Cy-Ties in South Korea. In Axel Bruns and Joanne Jacobs, editors, *Uses of Blogs*, pages 173–186. Peter Lang Publishing, 2006.

[18] Michael Kassoff, Charles Petrie, Lee-Ming Zen, and Michael Genesereth. Semantic email addressing. *IEEE Internet Computing*, 13(Jan./Feb.):48–55, 2009.

[19] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.

[20] Werner Kunz and Horst W.J. Rittel. Issues as Elements of Information Systems. Working Paper 131, Center for Planning and Development Research, UC Berkeley, 1970.

[21] Victor Kuperman. Productivity in the internet mailing lists: A bibliometric analysis. *Journal of the American Society for Information Science and Technology*, 57(1):51–59, 2006.

[22] Deborah L. McGuinness and Frank van Harmelen. OWL Web Ontology Language: Overview. W3C Recommendation 10 February 2004, W3C, 2004.

[23] Boris Motik, Bernardo Cuenca Grau, Ian Horrocks, Zhe Wu, Achille Fokoue, and Carsten Lutz. OWL 2 Web Ontology Language: Profiles. W3C Recommendation 27 October 2009, W3C, 2009.

[24] Peter F. Patel-Schneider and Boris Motik. OWL 2 Web Ontology Language: Mapping to RDF Graphs. W3C Recommendation 27 October 2009, W3C, 2009.

[25] Judith B. Pena-Shaff and Craig Nicholls. Analyzing student interactions and meaning construction in computer bulletin board discussions. *Computers & Education*, 42(3):243–265, 2004.

[26] Simon Scerri, Brian Davis, Siegfried Handschuh, and Manfred Hauswirth. Semanta — semantic email made easy. In *ESWC 2009 Heraklion: Proceedings of the 6th European Semantic Web Conference on The Semantic Web*, pages 36–50, Berlin, Heidelberg, 2009. Springer-Verlag.

[27] Albert Selvin, Simon Buckingham Shum, Maarten Sierhuis, Jeff Conklin, Beatrix Zimmermann, Charles Palus, Wilfred Drath, David Horth, John Domingue, Enrico Motta, and Gangmin Li. Compendium: Making Meetings into Knowledge Events. In *Proceedings Knowledge Technologies 2001*, 2001.

[28] Gina Danielle Venolia and Carman Neustaedter. Understanding Sequence and Reply Relationships within Email Conversations: A Mixed-Model Visualization. In *Proceedings of the ACM Conference*

on Human Factors in Computing Systems (CHI 2003), ACM. ACM, 2003.

[29] Steve Whittaker and Candace Sidner. Email overload: exploring personal information management of email. In *CHI '96: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 276–283. ACM, 1996.

[30] Ka-Ping Yee. Zest: Discussion Mapping for Mailing Lists. In *Proceedings of the ACM 2002 Conference on Computer Supported Cooperative Work*, 2002.