효율적인 P2P 파일 검색을 위한 RDF 파일 온톨로지 구조

한종욱°, 이승은, 한동윤, 김경석†
부산대학교 컴퓨터공학과°, 부산대학교 정보컴퓨터공학부†
{jwhan°, selee, dyhan, gimgs0}@asadal.cs.pusan.ac.kr

RDF File Ontology Structure for Efficient Peer-to-Peer File Search

Jong-wook Han°, Seung-eun Lee, Dong-yun Han, Kyong-sok Kim†
Dept. of Computer Engineering. Pusan National University°
Division of Computer Science and Engineering. Pusan National University†

요 약

현재 인터넷 등의 네트워크와 개인 컴퓨터의 발전으로 인하여 피어(peer)들 간의 파일을 공유하는 시스템에 대한 필요가 증가하고 있다. 이 피어들 간에는 이질적 시스템과 환경을 가지고 있다. 이러한 피어들 간의 자료교환은 서로의 의미가 달라지며, 사용자가 원하는 자료를 쉽게 찾을 수 없다. 이 문제점의 해결을 위해 현재 웹에서 사용자가 원하는 자료에 보다 정확하게 응답을 하기 위해 사용되는 온톨로지 개념을 P2P 시스템에 적용하여 각 시스템 또는 사용자간의 시맨틱 갭(Semantic Gap)을 없애고, 이를 위한 파일 온톨로지를 본 논문은 제안한다. 본 논문에서는 RDF를 사용하여 파일 온톨로지 개념을 현재의 P2P 파일 공유 시스템 중 하나인 Tapestry[1]에 RDF를 적용하여 자료 검색 시 사용자가 원하는 자료를 보다 정확하게 응답할 수 있는 효율적인 파일 검색 시스템을 제안한다.

1. 서 론

현재 P2P 시스템에서의 자료 검색은 찾고자하는 자료의 검색어와 시스템에 존재하는 자료의 파일명을 가지고 단순 텍스트를 비교하여 그 결과를 보여준다. 그 때문에 찾고자하는 파일을 정확하게 찾기보다는 수많은 응답들 중에 사용자가 몇 개의 파일들을 선택하게 되지만 이것들이 사용자가 원하던 파일이라는 것을 보장할 수가 없다. 이 문제는 P2P 환경뿐만 아니라 웹에서도 가지고 있는 문제이다. 현재 웹에서는 이러한 검색의 문제를 해결하기 위하여 온톨로지[2] 개념을 웹에 적용하기 시작하였고, 이러한 온톨로지 개념을 도입한 웹을 시맨틱 웹[3]이라한다. 시맨틱 웹에서 온톨로지 개념을 적용하기 위해 RDF[4]나 OWL[5]등을 사용한다. 우리는 온톨로지 개념을 P2P에 적용함으로서 사용자가 원하는 자료를 보다 정확하게 검색할 수 있는 방안을 제안한다.

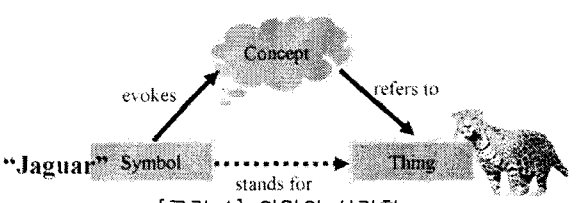
본 논문은 P2P시스템 중 하나인 Tapestry 상에서 온톨로지 개념을 RDF로 구성하고, 이 RDF를 이용하여 원하는 자료를 검색하기 위해 자료검색에 SPARQL[6]을 사용한다. 온톨로지 구성을 위해 사용되는 RDF의 구문들은 기계들이 이해가능하고 처리가능하다. RDF는 URI를 이용하여 자원들을 구분할 수 있는 유연한 구문을 만드는 메타데이터이다. RDF는 트리플(Triple)로 데이터가 구성되고, 트리플은 온톨로지를 참고함으로서 그 의미를 알 수 있게 된다. 우리는 파일 자료에 대한 온톨로지를 제안하고, 이 공통된 온톨로지 구성으로 Tapestry 상에서의 자료 검색 시 사용자가 원하는 자료를 한 번에 찾을 수 있게 한다. 논문에서 제안하는 온톨로지는 RDF로 서술하게 되고, 서술된 RDF는 SPARQL 질의어로 질의되어 원하는 자료를 찾을 수 있다.

2. 관련 연구

이번 장에서는 온톨로지 개념과 RDF(Resource Description Framework), 그리고 이 RDF에서 원하는 자료를 검색하기 위한 질의어인 SPARQL에 대해서 살펴보고 P2P 시스템 중의 하나인 Tapestry에 관하여 알아본다.

2.1 온톨로지

온톨로지는 의미에 대한 형식적인 명세서로서, 의사소통에서 서로의 의미를 통하게 하는 단어 간의 관계를 뜻한다. 예를 들어 A라는 사람이 B라는 사람에게 "Jaguar"라고 말을 했다고 하자, 이때 화자 A가 말한 "Jaguar"의 의미는 동물을 의미했지만, 청자인 B는 이를 영국의 자동차인 "Jaguar"로 생각할 수 있다. 이렇게 서로의 사용된 "Jaguar"라는 단어는 같지만 이를 받아들이는 뜻은 다를 수 있다. 이런 것을 시맨틱 갭이라 한다.



[그림 1] 의미의 삼각형

[그림1]은 의미의 삼각형[7]으로 시맨틱이 어떻게 이루어지는지를 보여준다. "Jaguar"라는 심볼이 있으면 이는 하나의 컨셉에 의해 '동물 재규어'를 가리키게 되고 이 "Jaguar"는 '동물 재규어'를 의미하게 된다. 의미의 삼각형에서 컨셉(Concept)이 온톨로지에 해당이 되며,

시맨틱 갭이 발생하는 원인은 이 온톨로지가 공유되지 않고, 서로 다른 온톨로지를 구성하고 있기 때문에 일어난다.

결국 시맨틱 겝을 해결하기 위해서는 공통으로 의미가 전달되고 공유 가능한 온톨로지를 구성해야 한다.

2.2 RDF (Resource Description Framework)

RDF는 World Wide Web에서 자원에 관한 정보를 표현하기 위한 언어로, 웹 자원에 관한 메타데이터를 표현하기 위한 것이다. 이 메타데이터는 웹 페이지의 표제와 저자, 갱신일자 등을 포함하여 웹 문서의 저작권과 계약 정보 등을 나타내며, 이를 사람들에게만 제시하는 것이 아니라 어플리케이션에서 처리하기 위해 고안 되었다.

RDF는 그래프 모델을 기본으로 하고, 이 모델로부터 주어(Subject), 술어(Predicate), 목적어(Object) 형식으로 이루어진 트리플을 만들게 된다. 트리플에서 주어와 술어, 그리고 목적어는 유사하게 보이는 식별기호와 혼동되지 않으면서 기계처리가 가능한 URI(URIs)[8]식별기호체계를 사용한다. 이는 사물을 식별하고, 간단한 성질과 그 값으로 자원을 기술한다. RDF는 그래프 모델을 기반으로 트리플을 작성한다.

[표 1] 트리플 (Triple)

<http://www.example.org/index.html>
<http://purl.org/dc/elements/1.1/creator>
<http://www.example.org/staffid/85740> .

RDF에서는 트리플 형식의 정보를 교환하기 위하여 RDF/XML이라고 하는 XML 구문을 제공한다.

[표 2] RDF/XML

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf=
          "http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:dc="http://purl.org/dc/elements/1.1/">
  <rdf:Description
        rdf:about="http://www.example.org/index.html">
    <dc:creator rdf:resource=
              "http://www.example.org/staffid/85740"/>
  </rdf:Description>
</rdf:RDF>
```

2.4 SPARQL

SPARQL(SPARQL Protocol And RDF Query Language)는 웹상에서 방대하게 분산된 RDF 데이터들을 질의하고 찾아오게 하는 Data Access 프로토콜이자 질의 언어이다. SPARQL은 "select"와 "where" 절을 사용하여 바인딩(Binding)한다. 여기에서 바인딩이란 질의와 RDF 데이터를 매칭(Matching)하는 것을 의미한다.

[표 3] SPARQL 질의

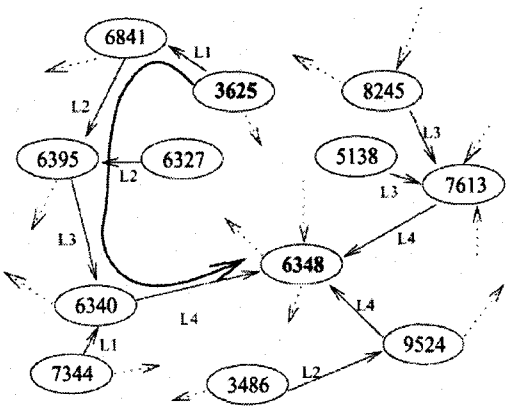
```
SELECT ?title

WHERE ( <http://example.org/book/book1>
          < http://purl.org/dc/elements/1.1/title> ?title )
```

2.5 Tapestry

Tapestry는 hashed-suffix mesh[9]와 유사한 메커니즘을 사용한다. Tapestry와 같은 DHT 기반 P2P 시스템의 자원 탐색시간은 O(logN)으로 알려져 있다.[10][11]

Tapestry는 네트워크 지역성(locality)을 고려하여 IP를 통해 탐색하는 시간으로 측정된 IP 탐색 기반구조에서의 가장 가까운 노드들을 이웃(neighbor)이라고 하며, 각 노드는 자신의 이웃들의 정보를 유지하여 다중 레벨들로 구성된 하나의 이웃 맵(neighbor map)을 가진다. 각 레벨은 목적지 ID인 GUID와 공유된 접두사(prefix)의 단계를 나타내며, 이 탐색 맵을 사용하여 탐색한다. 이것은 요청 NodeID에서 GUID로 ID의 자릿수를 증가시키면서 메시지를 전송하는 방법으로, N번째 홉은 GUID와 적어도 N 길이의 접두사를 공유한다. GUID와 가장 많은 접두사를 공유하며 숫자상 가장 가까운 NodeID는 그 데이터의 루트 노드(root node)가 된다. 즉, Tapestry는 네트워크 내의 모든 노드가 루트가 된 트리들의 하나의 큰 집합으로 볼 수 있다.

데이터 D에 대한 질의 메시지는 GUID를 가지고 D의 루트 쪽으로 순차적으로 레벨들을 탐색한다. 각 단계에서 메시지가 D의 위치 정보를 포함하고 있는 노드를 만난다면, 그 메시지는 즉시 데이터를 포함하고 있는 서버 쪽으로 방향을 전환한다. 만약 만나지 못한다면, 메시지는 루트에 도달하게 되며 D에 대한 위치 정보 탐색을 보장한다. 데이터 D를 가진 서버 S는 D의 루트 노드 쪽으로 메시지를 전달하여 데이터가 네트워크에 참여하여 존재하게 되었음을 알린다. 루트 노드는 서버 S의 위치를 가리키는 간단한 포인터를 저장한다. 이것은 단순한 포인터일 뿐 데이터의 복사본이 아니다.



[그림 2] GUID 6348을 찾는 노드3625의 Tapestry 탐색

3. 시스템 구조

3.1 온톨로지

RDF 형식의 온톨로지는 이미 널리 사용되며, 그 대표적인 예로는 문헌을 기술하기 위해 정의된 더블린코어 메타데이터[12]가 있다. 본 논문에선 파일을 기술하기 위한 최소한의 기술요소(성질)들에 대한 온톨로지를 제안한다.

[표4] 파일기술을 위한 온톨로지

name	: 파일 이름
title	: 자료의 제목
type	: 파일의 종류로 문서, 영상, 이미지 등
category	: 자료의 분류. 영상의 강좌, 영화 등
size	: 파일의 크기
format	: 파일의 형태
creator	: 파일을 만든 사람
modifier	: 마지막으로 파일을 변경한 사람
createdDate	: 파일을 생성된 날짜
modifiedDate	: 마지막으로 변경된 날짜
description	: 파일에 관한 설명
source	: 현재 파일이 파생된 자원에 대한 참조
relation	: 관련 파일들에 대한 참조
keyword	: 검색 시 원하는 주제어
rights	: 파일에 대한 보유된 권리 정보

온톨로지는 특정 도메인에서 통용되는 의미 사전이다. 본문에서 제시하는 온톨로지는 특정 도메인인 "http://asadal.cs.pusan.ac.kr/~jwhan/off"에서 통용되는 온톨로지이다.

[표4]는 본 논문에서 제시하는 파일기술을 위한 온톨로지(off : Ontology For File-description)이다.

[표4]에서 Name은 파일의 이름이고, Title은 문서일 경우 제목을 의미한다. 파일명은 Tapestry에서 이미 고려된 상태이지만, 해쉬한 값으로 찾아오기 때문에 루트 노드가 관리하는 자료가 모두 동일한 검색어라고는 볼 수 없다. 그렇기 때문에 여기에선 파일이름이 명시되어야 한다. 그리고 Type은 자료의 형식이 무엇인지 나타내며, Category는 자료의 분류나 장르를 의미한다. 예를 들어 Title이 "전쟁과 평화"라는 자료는 Type이 "audio/mpeg"고 Category는 "멜로"이다.

3.2 Tapestry 구조

Tapestry는 실질적 자료를 가지고 있는 노드와 그 노드에 대한 Backward-pointer를 가지고 있는 루트 노드로 이루어진다.

3.2.1 Node 구조

Tapestry의 각 노드들은 자신이 가지고 있는 자료에 대한 RDF를 가지고 있으며, 이 RDF는 [표4]에서 제시한 파일기술을 위한 온톨로지로 구성되지만 이 RDF 온톨로지 요소들 모두가 필수사항은 아니다.

[표5] 파일에 관한 RDF

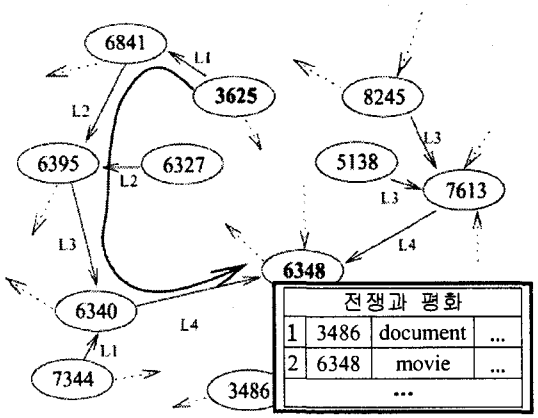
```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:off="http://asadal.cs.pusan.ac.kr/~jwhan/off#"
  xmlns:foaf="http://xmlns.com/foaf/0.1/">
   ....
   <rdf:Description rdf:about="tapestry.html">
      <off:name>tapestry.html</off:name>
      <off:title>
         RDF Ontology on the Tapestry
      </off:title>
      <off:type>Document</off:type>
      <off:format>text/html</off:format>
      <off:description>this document ...</off:description>
      <off:creator>
         <foaf:Person>
            <foaf:name>Jong-wook Han</foaf:name>
            <foaf:homepage rdf:resource=
                  "http://asadal.cs.pusan.ac.kr/~jwhan/"/>
         </foaf:Person>
      </off:creator>
      <off:createdDate>2005-03-05</off:createdDate>
      <off:keyword>
         <rdf:Bag>
            <rdf:li>tapestry</rdf:li>
            <rdf:li>rdf</rdf:li>
            <rdf:li>p2p</rdf:li>
         </rdf:Bag>
      </off:keyword>
   </rdf:Description>
   ....
</rdf:RDF>
```

[표5]는 각 노드들이 가지고 있는 RDF파일의 예제이다. 네임스페이스에 RDF(rdf)와 파일기술에 대한 정의(off)를 선언하고, 사람에 관해 정의를 위해 foaf[13]를 선언하였다. 본 예제는 "tapestry.html"에 관하여 기술을 하는데, name은 자료의 제목을 기술하고, type은 자료의 종류 중 문서(Document)를 기술하였다. description으로 본 자료에 관하여 서술하였고, creator는 foaf를 이용하여 사람을 정의하였다. keyword로는 3가지의 정보를 가지고 있어 검색을 용의하게 하였다. 그 외에 format등의 정보를 기술하였다.

3.2.2 Root Node 구조

Tapestry에서 루트 노드는 자신의 GUID와 동일한 해쉬값의 자료를 가진 노드들에 대한 Backward-pointer를 관리한다. 우리가 제시하는 Tapestry의 루트 노드는 자신에게 조인(join)된 노드들에 대한 Backward-pointer와 그 노드가 가지고 있는 자료에 관한 RDF 정보를 가지게 된

다. 단, 이 RDF정보는 GUID와 해쉬값이 같은 데이터에 관한 RDF정보이며, 노드가 가지고 있는 RDF 정보 전체는 아니다.



[그림 3] GUID 6348을 찾는 3625노드의 Tapestry 탐색과 6348노드의 Backward-pointer와 RDF정보

[그림3]에선 GUID가 6348인 노드가 관리하는 Backward-pointer와 이와 연관된 노드들이 가지고 있는 자료에 관한 정보를 간략하게 나타내고 있다. 여기에는 단지 Type 정보만을 명시하였다. 이외에 많은 정보들을 루트 노드가 가질 수 있다.

3.3 노드 탐색과 질의

Tapestry 탐색은 관련연구 2.5에서 살펴보았다. 노드 탐색 과정은 기존의 Tapestry와 동일하다. [그림3]에선 NodeID가 3625인 노드가 "전쟁과 평화"라는 이름으로 검색을 하였을 때 "전쟁과 평화"의 해쉬 값이 6348이 나오게 되고 Tapestry 탐색 경로를 따라 6348를 찾아가게 된다. 이 때 6348노드는 수많은 Backward-pointer와 정보를 가지고 있을 수 있다. 노드 탐색이 끝나면 찾은 루트 노드에게 질의를 하게 되고, 그 결과를 3625 노드는 받아 오게 된다. 여기에 사용되는 질의는 RDF 정보를 검색하기 위해서 SPARQL 질의를 사용하게 된다.

[표6] SPARQL을 이용한 질의

```
SELECT ?name ?type ?category
PREFIX off:<http://asadal.cs.pusan.ac.kr/~jwhan/off#>
WHERE  ( ?x off:name ?name )
                        AND ?name="전쟁과 평화"
       ( ?x off:type ?type )
      [ ( ?x off:category ?category ) ]
```

[표6]은 아주 간단한 질의를 나타낸다. 이 질의는 type 정보가 있는 것만 찾아 나타내고, 이중에서 category정보를 가지고 있으면 나타내고 없으면 category정보는 생략한다. 이 질의만 사용한다면 [그림3]에서 6348 노드는 3486과 6327등 많은 노드가 선택될 것이다. 하지만 where절에서 "(?x off:type ?type)" 대신 "(?x off:type ?type) AND ?type=document"라는 조건을 사용 한다면 3486만 결과로 나타나게 된다.

3.4 시스템 구현

시스템 구현은 한 노드가 Tapestry에 조인하는 것과 다른 노드가 자료를 검색하는 부분으로 나눈다.

3.4.1 노드의 조인(Join)

"전쟁과 평화"라는 문서를 가지고 있는 한 노드가 Tapestry상에 조인하게 되면, NodeID(3625)를 부여받게 되고, 그 노드의 자료를 해쉬하게 된다.

그 자료에 대하여 해쉬한 값이 6348이 되면 GUID가 6348인 노드를 탐색하고, 6348 노드에 자신의 자료인 "전쟁과 평화"에 관한 RDF 정보를 알려주게 된다. 그럼 루트 노드인 6348은 3486에 대한 Backward-pointer와 함께 3486으로 부터 받은 RDF 정보를 저장하게 된다.

이후 같은 해쉬 값을 가진 자료를 가진 노드들이 6348에게 조인하게 되고, 6348은 이 정보들을 모두 저장하게 된다.

3.4.2 자료의 검색

NodeID가 3625인 노드가 "전쟁과 평화"에 관하여 자료를 찾게 되면 "전쟁과 평화"를 해쉬한 값인 6348과 같은 GUID를 탐색하고, 그 노드에게 SPARQL을 이용하여 원하는 자료에 관한 질의를 하게 된다. 이때 6348 노드는 그 질의의 결과와 Backward-pointer의 정보를 3625 노드에게 주게 된다.

3.4.3 사용자의 자료선택

사용자는 결과로 RDF 정보를 받게 된다. 이때 결과가 하나 이상일 경우, 사용자는 검색 시 입력하지 않은 정보도 검색한 결과와 함께 제공 받게 된다. 사용자는 이 정보를 이용하여 원하던 자료를 선택할 수 있게 된다.

더 나아가서 사용자가 자신이 받은 자료와 연관된 자료들을 찾거나, 자료의 출처가 되는 자료를 찾기를 원할 때는 검색 결과로 받은 RDF 정보 중 Source와 Relation을 이용해 찾을 수 있다. Source정보는 그 자료의 출처가 되는 자료의 정보를 나타내며, Relation은 그 자료와 연관된 자료들을 나타낸다. 이 두 정보는 사용자가 직접 재검색 가능하지만, 시스템구현에 따라 Depth를 두어 자동으로 Depth 만큼 관련된 자료를 찾을 수 있다.

4. 결론 및 향후 연구과제

우리는 Tapestry 기반의 P2P 시스템에서 RDF를 이용한 의미론적 파일 검색 방법과 이것을 위한 온톨로지를 제안하였다. 우리가 파일기술에 대한 온톨로지로써 RDF를 이용한 검색을 제안하였다. 이것은 사용자가 원하는 자료를 쉽게 찾을 수 있고, 기계가 이해하고 처리 가능하여 연관된 자료도 찾을 수 있다. 만약 위의 예제와 같이 "전쟁과 평화"를 사용자가 찾는다면 문서로 된 소설자료와 영상으로 된 영화자료 등 많은 자료들이 검색 될 것이다. 현재의 검색으로 한다면 사용자는 이 많은 검색

결과에서 어떤 자료가 자신이 원하는 자료일지 알 수 없다. 하지만 본 논문이 제시하는 검색을 사용하면 자신이 원하던 문서자료를 바로 찾을 수 있다.

본 시스템은 RDF를 작성해야하고 이를 유지·갱신해야 하는 부담이 있고 우리가 제안한 파일기술을 위한 온톨로지가 완벽하다고 말하지 못 한다. 이 파일기술을 위한 온톨로지는 앞으로 추가 또는 수정, 삭제함으로 더욱 많은 것을 표현할 수 있는 온톨로지를 구축해야 한다.

우리는 본 논문에서 P2P 시스템에서의 보다 효율적인 검색을 제안하였다. 앞으론 구축된 RDF를 이용하여 Keyword 검색이나 Range 검색도 가능하게 확장 할 것이고, 본 논문에서 제안한 파일기술을 위한 온톨로지를 P2P 시스템만 아니라 컴퓨터 안의 모든 자료에 대하여 작성 되게 할 것이다. 이는 결국 Social Semantic Information Space를 형성하기 위한 기초가 될 것이다.

참고문헌

[1] B. Y. Zhao, J. Kubiatowicz, and A. D. Joseph, "Tapestry: An infrastructure for faulttolerant wide-area location and routing", Tech. Rep. UCB/CSD-01-1141, Computer Science Division, University of California, Berkeley, Apr 2001

[2] http://www.w3.org/2001/sw/WebOnt. Web-Ontology (WebOnt) Working Group.

[3] http://www.w3.org/DesignIssues/Semantic.html. Semantic Web Road map

[4] http://www.w3.org/RDF. World-Wide Web Consortium: Resource Description Framework

[5] http://www.w3.org/TR/owl-ref/. OWL Web Ontology Language Reference

[6] http://www.w3.org/TR/rdf-sparql-query. SPARQL Query Language for RDF.

[7] The meaning of meaning. Ogden,C.K. & Richards,I.A(1923)

[8] http://www.w3.org/Addressing. Naming and Addressing.

[9] PLAXTON, C. G., RAJARAMAN, R.,AND RICHA, A. W. "Accessing nearby copies of replicated objects in a distributed environment", In Proceedings of the Ninth Annual ACM Symposium on Parallel Algorithms and Architectures(SPAA) (June 1977)

[10] I. Stoica, R. Morris, D. Karger, M.F Kaashoek, and H. Balakrishman, "Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications", ACM SIGCOMM'01, Aug. 2001

[11] S. Ratnasamy, P. Francis, M. Handley, R. Karp, "A Scalable Content-Addressable Network", ACM SIGCOMM'01, Aug. 2001

[12] http://www.dublincore.org/about/overview/. Dublin Core Metadata Initiative

[13] http://www.foaf-project.org/. The Friend of a Friend (FOAF) project.