Chapter 5

OntoSIR: a prototypical implementation of OntOAIr

The OntoSIR system (*Ontology-based Information Retrieval System*) is a demonstration prototype system of the method to construct ontologies of records (Chapter 3). It implements the information retrieval models presented in Chapter 4.

The main goal of implementing OntoSIR is to provide users with an environment to construct ontologies of records, and an interface to explore and retrieve information from distributed collections. Figure 5.1 shows a general case of use of OntoSIR.

In this scenario, a user selects data providers and introduces a query through a web-based application. Then, OntoSIR harvests metadata records from these data providers by using OAI-PMH *request verbs* and automatically analyzes records. Harvested records, represented as feature vectors and the FIHC algorithm are used to construct an ontology of records for each data provider. OntoSIR provides a query driven recall mechanism for the constructed ontologies, which are centrally maintained.

There are two interaction modes between users and the keyword-based model: syn-

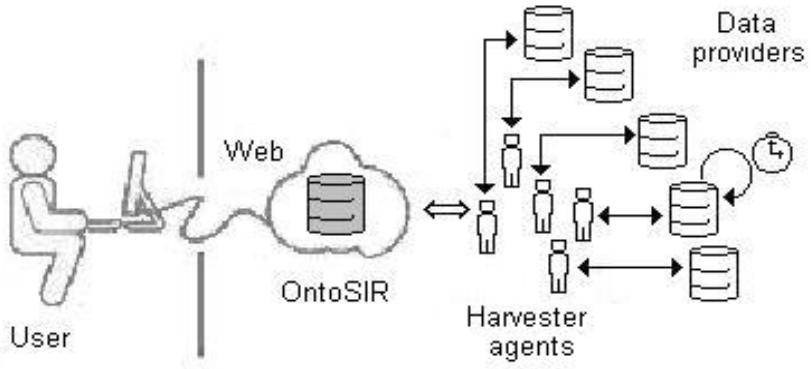

Figure 5.1: A general case of use of OntoSIR

chronous and asynchronous. The former is used when the required ontologies have been constructed. Then, the retrieval tasks are performed on line. In contrast, the latter uses email to send the response, which is an XML file with the potential collection P.

The main components of OntoSIR system are shown in Figure 5.2. The tasks they implement are the following:

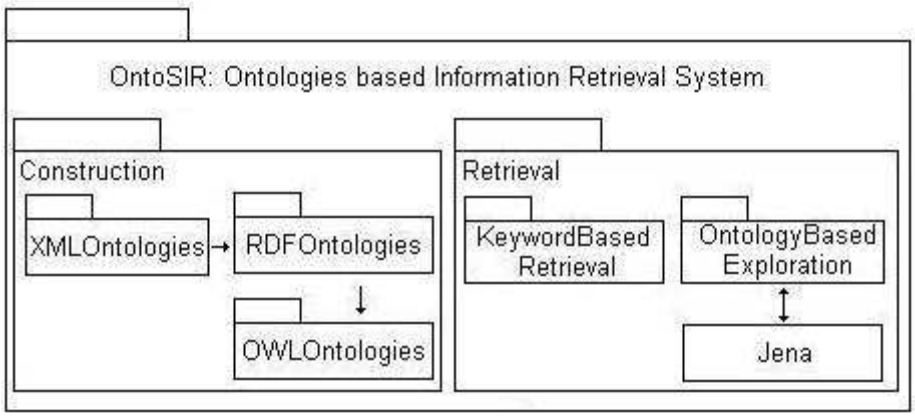

Figure 5.2: Architecture of the OntoSIR system

- *Construction Subsystem*: This subsystem implements the ontologies construction method. It comprises three modules:
  - *XMLOntologies module*: This implements an ontology of records in the XML language. It preserves the structure of the DTD of Table 3.5 but includes the datatypes of Table 3.6 (an XML Schema is used for this purpose)
  - *RDFOntologies module*: This makes uses of the RDF Schema of Table 3.7 and 3.8 to transform an XML ontology into an RDF ontology
  - *OWLOntologies module*: This uses the OWL model of Appendix C to convert an RDF ontology into a valid ontology implemented in OWL DL
- *Retrieval Subsystem*: This subsystem implements the information retrieval models, and comprises two main modules:
  - *KeywordBasedRetrieval module*: It implements the keyword-based retrieval model which is summarized in Algorithm 4.1
  - *OntologyBasedRetrieval module*: It allows users to enter queries that exploit the semantic feature of ontologies. The selection of a specific implementation of ontologies is required. This module is built on top of Jena 2.2. Thus, queries are processed by OntoSIR and then sent to Jena for corresponding actions.

The graphical user interface of OntoSIR allows users to choose a data provider to construct an ontology. OntoSIR can work with any arbitrary ontology of records essentially without restrictions, except that the interface limits the number of data providers in retrieval tasks in order to preserve the performance of the retrieval subsystem. The details of records can be viewed by following hyper-links.

Since OntoSIR deals with data providers (distributed and semi-structured information repositories), its design is based on agents. This approach has been successfully used in digital library applications [62]. Figure 5.3 shows the agents architecture of OntoSIR. The roles of harvester agents are detailed in Table 3.1 and 3.2. The role of searcher and ontological agents are shown in Table 4.2 and 4.4, respectively.

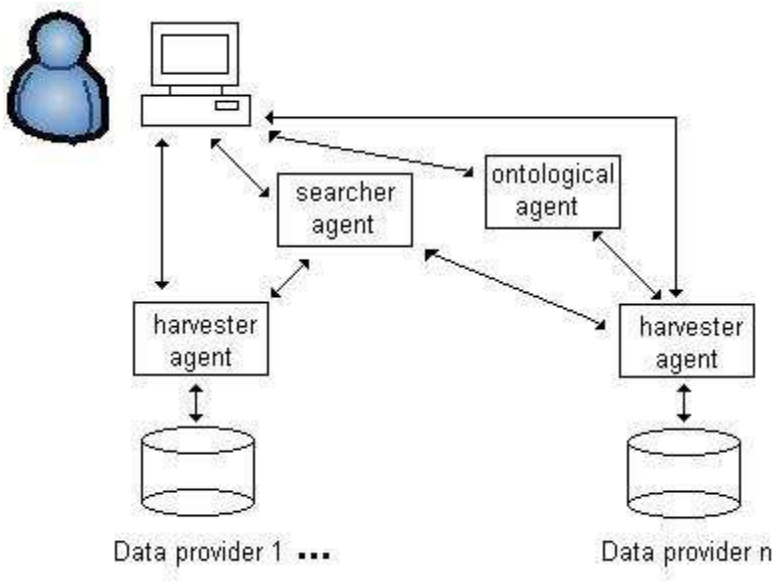

Figure 5.3: Agent architecture of OntoSIR system

In Figure 5.3, a user enters a query on a local machine. A harvester agent accesses to at least one and potentially many data providers. The agent harvests the information from these sources. For retrieval or exploration tasks, a searcher agent contacts harvesters agents, or an ontological agent contacts a harvester agent, according to the type of information required by the user.

The graphical user interface of OntoSIR allows users to carry out the following main tasks: (1) searching relevant groups of records from multiple data providers, (2) clus-

tering the records from a data provider, and (3) exploring a data provider. The results of these tasks are returned to users by e-mail as XML files in such a way that they can be used by parsers and other XML processing software.

Figure 5.4 shows the interface of OntoSIR when the *search* option is selected from the menu (left panel). The right panel contains a selection box and text fields through which data providers and the query can be entered. This interface limits the number of data providers that can be selected in order to maintain the performance of OntoSIR (at most three data providers per query).

The result of a search task is a XML file that contains a list of groups of potentially relevant records with respect to the keywords of the query. This file implements the structure proposed by the DTD of Table 4.1.

Figure 5.5 shows the interface of OntoSIR when the *cluster* option is selected from the menu (left panel). Users can select the name of the data provider to be clustered from the selection box or they can introduce its base URL in the text field. At the time of this writing, there are 830 registered data providers at OAI [1] .

The result of a cluster task is a XML file that contains the ontology of records of the selected data provider. This file implements the structure proposed by the DTD of Table 3.5.

Figure 5.6 shows the interface of OntoSIR when the explore option is selected from

[1] Registered data providers. Open Archives Initiative. October 14th 2007. $< http://www.openarchives.org/Register/BrowseSites >$.

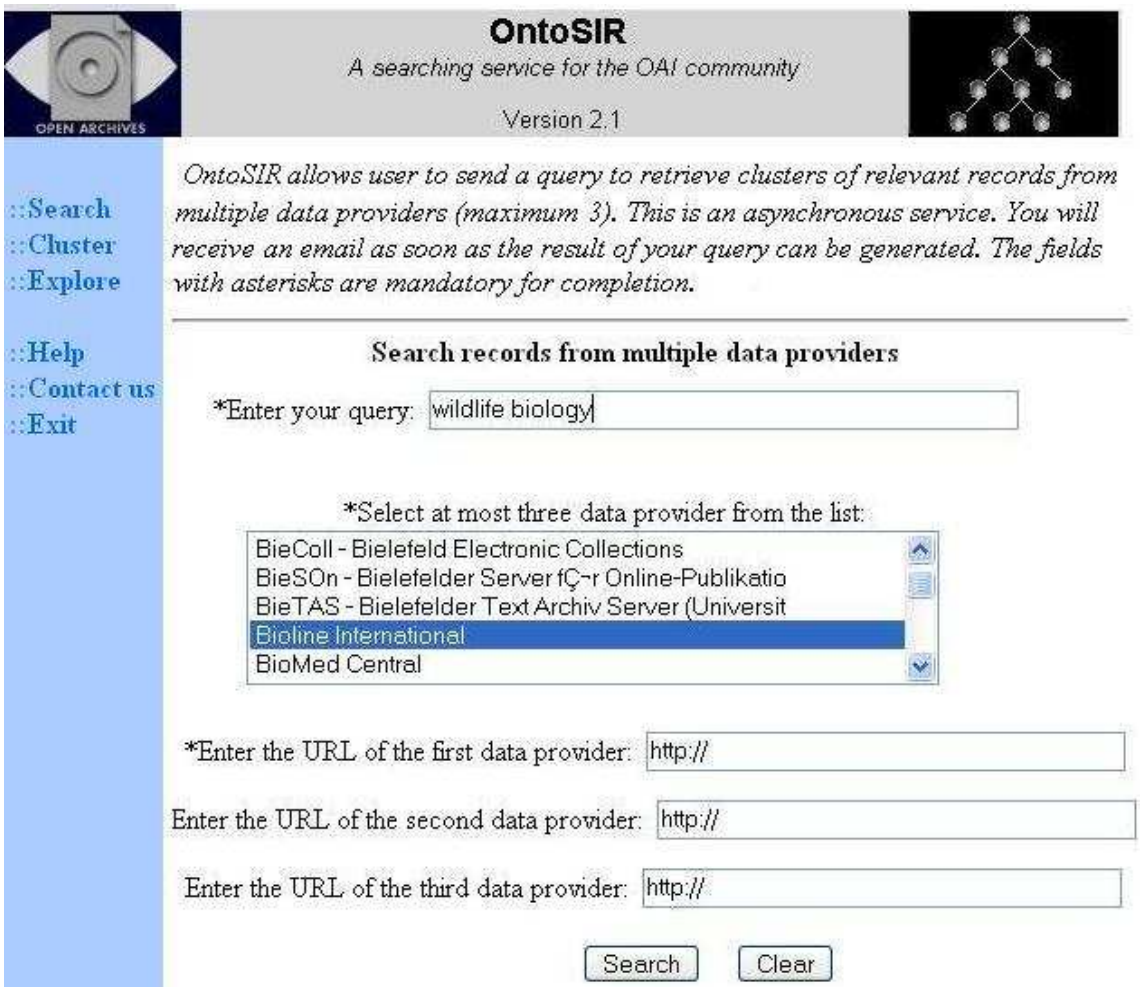

Figure 5.4: Graphical user interface of OntoSIR to search records from multiple data providers

the menu (left panel). Users can select the name of the data provider to be explored from the selection box or they can introduce its base URL in the text field. Then, a text field is used to introduce a structured query and a selection box to choose the mapping of the ontology (XML, RDF or OWL).

The result of the explored task is a XML file that contains a set of instances of tupples according to the structure of the DTD of Table 4.3.

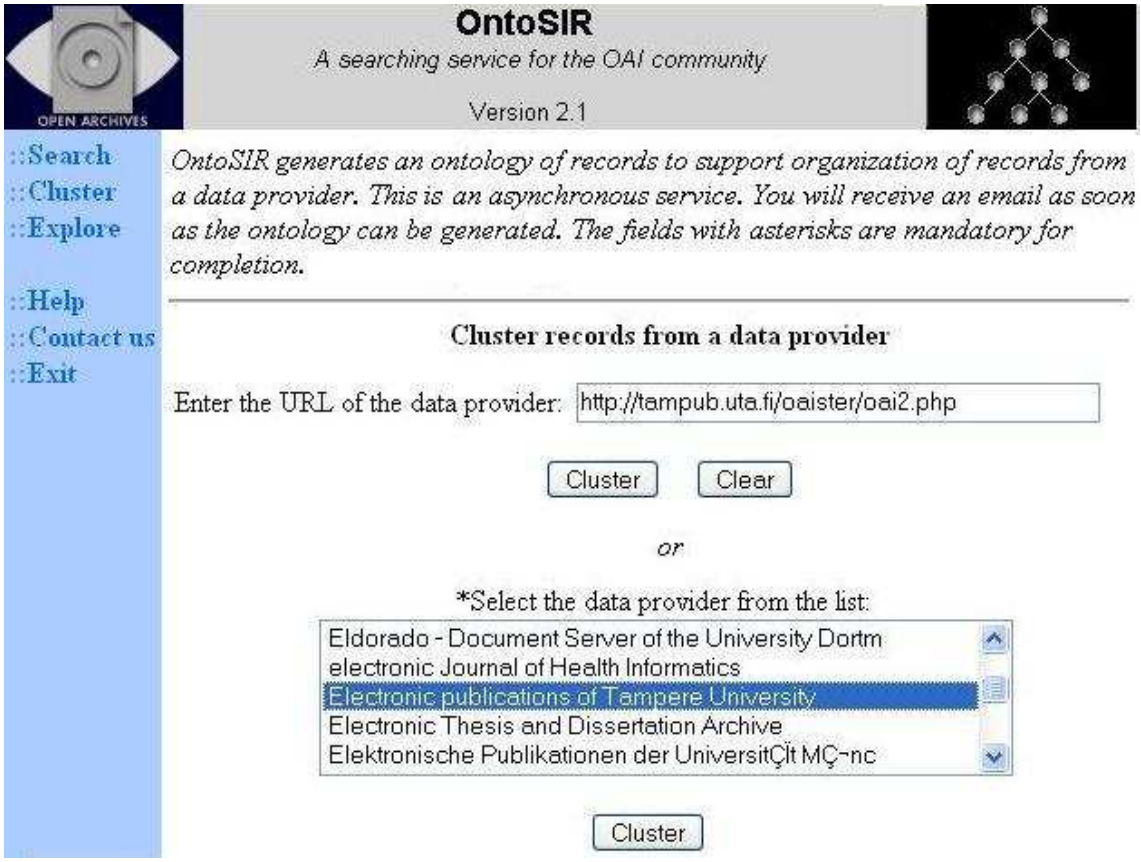

Figure 5.5: Graphical user interface of OntoSIR to cluster records from a data provider

OntoSIR is a web-accessible Java system that requires a servlet container. Apache Tomcat 5.0.29 is used in this capacity. OntoSIR employs MySQL 5.0.4 as its database management system.

The next chapter discusses experimental results of the OntOAIr method. Test collections often referred in document clustering research and information retrieval research are used as a test bed.

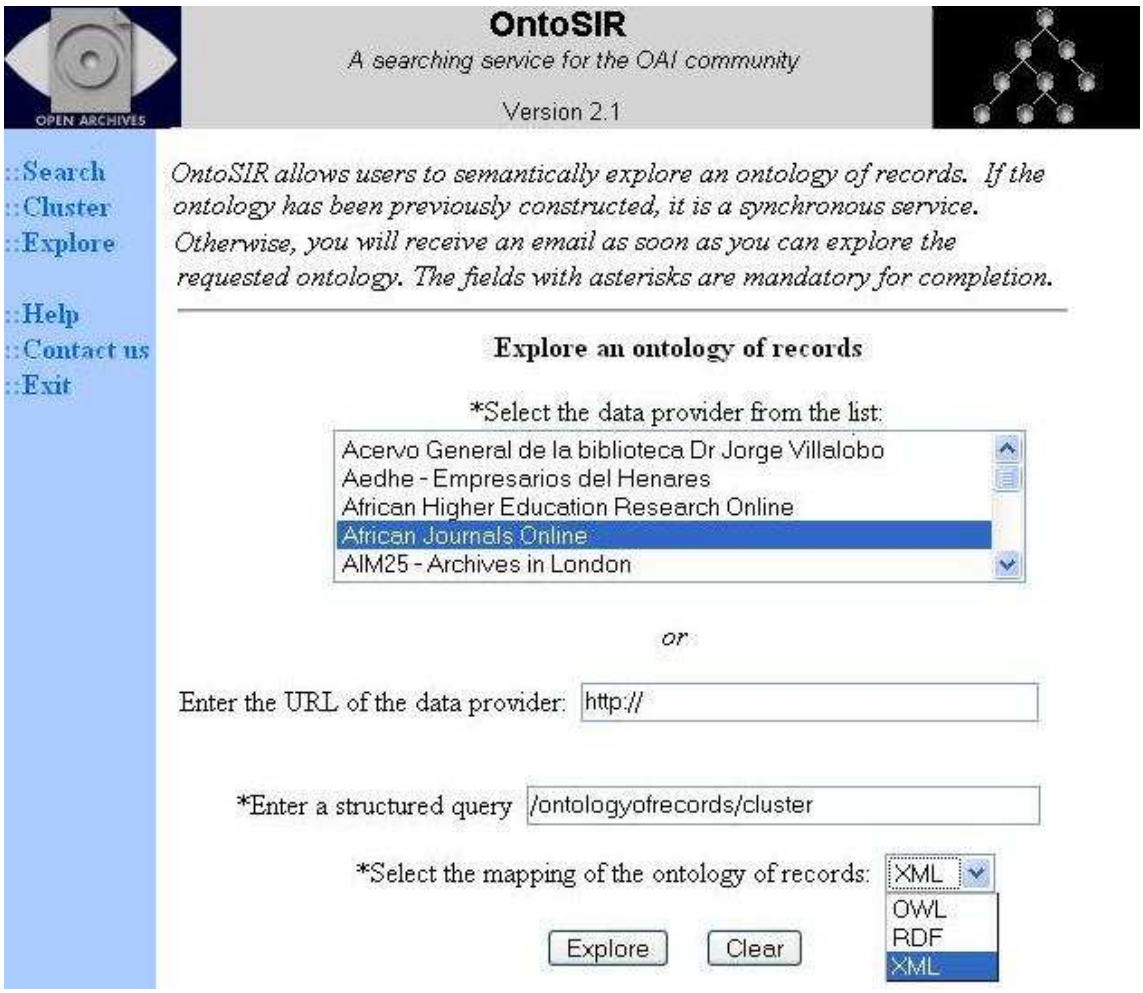

Figure 5.6: Graphical user interface of OntoSIR to explore a data provider