

Ontology-driven Integration of Scientific Repositories¹

Vassilis Christophides^{1,2}, Catherine Houstis^{1,2}, Spyros Lalis^{1,2}, and Hariklia Tsalapata²

¹Department of Computer Science, University of Crete,
P.O. Box 1470, GR-71110 Heraklion, Greece

²Institute of Computer Science, Foundation for Research and Technology – Hellas,
P.O. Box 1385 Heraklion, Greece
{christop, houstis, lalis, htsalapa}@ics.forth.gr

Abstract. There is an increasing need to provide scientists and researchers as well as policy makers and the general public with value-added services integrating information spread over distributed heterogeneous repositories. In order to incorporate available data sets and scientific programs into a powerful information and computational system it is mandatory to identify and exploit their semantic relationship. For this purpose, we advocate an ontological framework that captures these relations and allows the inference of valid combinations of scientific resources for the production of new data. We show how a knowledge base that commits to an ontology can be used to generate workflows on demand for the multiplicity of resources known to the system. To validate our ideas, we are currently developing a prototype for the area of Coastal Zone Management.

1 Introduction

The rapid development of distributed computing infrastructures and the growth of the Internet and the WWW have revolutionized the management, processing, and dissemination of scientific information. Repositories that have traditionally evolved in isolation are now connected to global networks making their content available worldwide. In addition, with common data exchange formats [19], [25], [35], standard database access interfaces [27], [30], and emerging information mediation and brokering technologies (see middleware support in the context of I3 [26] and DLI [22]), data repositories can be accessed without knowledge of their internal syntax and storage structure. Furthermore, search engines [5], [18], [23] are enabling users to locate distributed resources by indexing appropriate metadata descriptions [14], [20], [24], [33]. Open communication architectures [4], [32], [34] provide support for language-independent remote invocation of legacy code thereby paving the way towards a globally distributed library of scientific programs. Finally, workflow management systems exist for coordinating and monitoring the execution of scientific

¹ This work was supported in part by the THETIS Project, EU Research on Telematics Programme, nr. F0069, URL: <http://www.ics.forth.gr/pleiades/THETIS/thetis.html>.

computations. The standardization and interoperability of workflow systems is pursued by the WMC [36].

This technology can be successfully used to address system, syntactic, and structural interoperability of distributed heterogeneous scientific repositories. However, interoperability at the semantic level remains a challenging issue [13]. Users still face the problem of identifying scientific resources that can be combined to produce new data. This is of key importance for providing widely diversified user groups with advanced, value-added information services.

In this paper we address the problem of producing meaningful combinations of available data sets and scientific programs to generate data needed for a particular task (e.g. environmental planning, forecasting). We advocate an ontological framework that captures domain and task specific knowledge in terms of concepts abstracting data sets, and relations between them abstracting implementations of physical models. We show how a Knowledge Base System (KBS) that commits to this ontology can infer valid combinations of scientific resources, alleviating the need to manually construct workflows. This process is dynamic, depending on the availability of data and programs. To validate our approach, we are currently developing a prototype for Coastal Zone Management [9].

The rest of the paper is organized as follows. Section 2 presents an example scenario regarding the generation of waste transport data for coastal areas. Section 3 introduces a corresponding application-specific ontology, provides its formal representation in a KBS, and illustrates the goal-driven generation of data production paths. In Section 4, a system architecture that employs this approach to integrate scientific repositories is sketched. Section 5 gives an overview of related work. Finally, Section 6 concludes and sets future research directions.

2 A Scenario for Coastal Zone Management

Environmental scientists and public institutions working on Coastal Zone Management (CZM) often need to extract and combine data from different scientific disciplines, such as marine biology, physical and chemical oceanography, geology, and engineering, stored in distributed repositories. Consider, for instance, the transport of waste in a particular coastal area given a pollution source. Local authorities could require this information to determine the best location for installing a waste pipeline.

This data is typically generated through a two-step process, involving the execution of two different programs (as shown in Fig. 1). First, Sea Circulation data is produced for the area of interest via an Ocean Circulation model that takes as input local Bathymetry and global (boundary) Current conditions. Then, Waste data is generated via a Waste Transport model that takes as input local Sea Circulation and Bathymetry data as well as the characteristics of the pollution source.

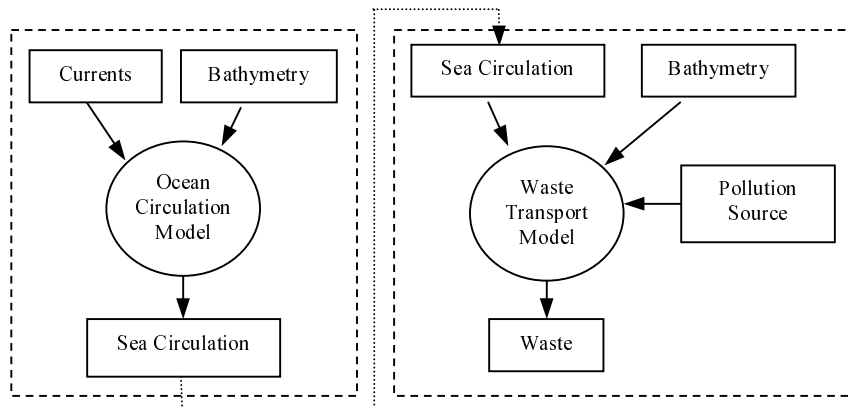


Fig. 1. Combination of data and programs for producing waste transport data. The production of *Sea Circulation* data is sketched on the *left* while the production of *Waste* data, which depends on the existence of *Sea Circulation* data, is given on the *right*.

Provided that the user has no knowledge of this information, the following actions are necessary to discover which productions can be used to obtain *Waste* data for a particular coastal area using the available resources:

1. Locate *Waste* data stored in the distributed repositories. This is typically done through a search engine that takes as input a few keywords and returns a list of relevant resource descriptions (metadata) available in the repositories.
2. Determine usability of search results. Due to the complexity of scientific metadata this requires a detailed inspection, as opposed to casual browsing, of the returned descriptions; the FGDC standard, for example, is several pages long.
3. If no *Waste* data that satisfies the user requirements is available (e.g. there is no data for the particular coastal area), locate programs capable of producing this data. Again, this involves a metadata search and examination of the returned program descriptions.
4. Having identified an appropriate program, i.e. a *Waste Transport* model, determine the required input. Ideally, this information is part of the program's metadata, else it must be acquired by contacting the creators of the program.
5. For each input, locate appropriate sources or determine ways to produce corresponding data sets. For instance, if *Sea Circulation* data is not available for the area of interest the possibility of producing it via an *Ocean Circulation* model must be pursued.

It becomes evident that manual discovery of combinations of data sets and programs that can lead to the production of desired data is tedious. It requires a rigorous investigation and understanding of metadata descriptions. This is difficult for a non-expert operating in a multidisciplinary environment. In addition, the user must keep

track of interrelated searches to discover possible ways of producing data for a particular task. As shown in the above example, this can be hard to achieve even for a relatively simple scenario with merely a couple of intermediate computation steps.

Therefore, support for deriving valid combinations of data and programs in a straightforward way is invaluable, particularly for large information systems. To achieve this, we use a Knowledge Base System (KBS) that captures the expertise of scientists regarding the semantic relationships between scientific resources. This allows on-demand generation of data production paths without significant effort on the user's behalf.

3 Integration of Scientific Resources via a KBS

In the following, we present our approach by introducing an ontology along the lines of the Waste Transport scenario. Then, we show how this ontology can be captured in the form of rules stored in a KBS to support goal-driven generation of valid productions.

3.1 An Ontology for the Waste Transport Scenario

An ontology [7] describes concepts and relationships among them related to a particular domain, task, or application. Ontologies can be used to share knowledge about specific aspects of the real world between individual agents in an unambiguous way. Thus, as stressed in [13], capturing commonly agreed knowledge in terms of ontologies is a major step towards addressing semantic interoperability in information systems.

In Environmental Sciences, concepts such as Air, Water, Wave, and Waste give the intrinsic meaning of data and programs residing in the repositories. Associated with these concepts are several physical notions such as location, grid, or time that are encountered in the various geospatial and environmental metadata standards (e.g. FGDC, UDK). An ontological foundation of this knowledge provides a formal framework for inferring valid combinations of available scientific resources regarding concrete environmental problems, e.g. Waste Management.

Typical real-world applications yield an ontology that includes numerous intricately related concepts². For the purposes of this paper, we consider merely a part of such an all-encompassing ontology where simplified CZM concepts are illustrated for the Waste Transport scenario (Fig. 2). Ontology concepts abstract data sets of environmental information. Relations between the concepts represent the programs employed to produce such data. As an example, Waste Transport models are viewed as functions mapping Sea Circulation and Bathymetry data onto Waste data; for simplicity, the characteristics of the pollution source are omitted. It must be pointed out that several different implementations of scientific models can be instances of the same abstract semantic relationship between concepts. For example, there may be two

² There is ongoing effort for the definition of an ontology in the area of Ocean Engineering at MIT [15].

programs implementing the Waste Transport Model relation, producing results with different precision or for different locations.

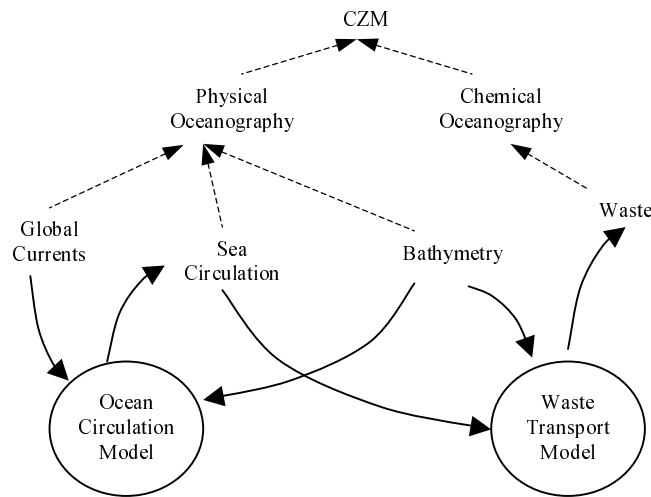


Fig. 2. A task-specific ontology for the Waste Transport application, introducing *Global Currents*, *Bathymetry*, *Sea Circulation*, and *Waste* concepts (data abstractions) that belong in wider thematic areas of CZM. These concepts are linked through *Ocean Circulation Model* and *Waste Transport Model* relations (program abstractions).

The semantics of this ontology are given by the interpretation of the various concepts and relations abstracting the underlying scientific resources. For this purpose, we follow an intentional definition of ontology notions using attributes such as location, grid resolution, etc, which describe the properties of scientific resources and are used to further specify the relations between them. From a logical viewpoint, the descriptions form a set of axiomatic rules, which are considered to be true independently of the actual content of the underlying repositories.

3.2 The Knowledge Base

In order to allow reasoning on the combination alternatives between data and programs we advocate a definition of the ontology notions in a KBS using Horn Clauses³. An ontology notion N is defined as a clause $N(A_1, A_2, \dots, A_n)$, where A_1, A_2, \dots, A_n are it's attributes. Relations between concepts are expressed as rules of the form

³ In this context, other knowledge representation and reasoning formalisms, like Description Logic [21], are less suited for handling arbitrary arity reasoning.

$N(A_1, A_2, \dots, A_n) :-$
 $N_1(A_1, \dots, A_n), \dots, N_n(A_1, \dots, A_n), Expr(A_1, \dots, A_n)$

where “:-” denotes implication and “,” conjunction. The rule body includes program and data concepts N_i as well as constraints $Expr$, e.g. parameter restrictions, for deducing the notion appearing as a consequent in the rule head. Exactly one literal in the body describes the corresponding program notion. The rest of the literals stand for the description of input data required by that program.

According to this notation, the following clauses define the notions introduced in the above ontology (with considerably simplified attribute sets):

$Bathymetry(Location, GridRes)$
 $ExtCurrents(Location, GridRes)$
 $SeaCirc(Location, GridRes)$
 $Waste(Location, GridRes)$
 $OceanCircModel(Location, GridRes)$
 $WasteTranspModel(Location, GridRes)$

In addition, the ontology relations shown in Fig. 2 are formalized using two rules:

R1: $SeaCirc(Location, GridRes) :-$
 $OceanCircModel(Location, GridRes),$
 $ExtCurrents(Location, GridRes'),$
 $Bathymetry(Location, GridRes''),$
 $GridRes \leq GridRes',$
 $GridRes \leq GridRes''.$

R2: $Waste(Location, GridRes) :-$
 $WasteTranspModel(Location, GridRes),$
 $SeaCirc(Location, GridRes'),$
 $Bathymetry(Location, GridRes''),$
 $GridRes \leq GridRes',$
 $GridRes \leq GridRes''.$

Rule R1 states that Sea Circulation data for a specific location and grid resolution can be derived from local Bathymetry and external Current data using an Ocean Circulation program. Similarly, rule R2 states that Waste data for a specific location and grid resolution can be produced by combining Sea Circulation with local Bathymetry data via a Waste Transport program.

Remark. In the above rules, “ \leq ” denotes higher or equal grid resolution; for example $10m^3 \leq 1m^2$ and $10m^2 \leq 1m^3$.

Clauses without a body, called facts, are instances of abstract notions. Without loss of generality, in the following we use strings to denote attribute values. Locations are specified through symbolic names in place of geographical co-ordinates to enhance readability. For example, $SeaCirc(HER, 10m3)$ stands for 3-D Sea Circulation for the area of Heraklion with a grid resolution of ten cubic meters. Similarly, $WasteTranspModel(HER, 1m2)$ stands for a Waste Transport program that

computes 2-D Waste data for the area of Heraklion with a grid resolution of one square meter.

Facts are either extensional, indicating available data sets or programs, or intentional, denoting data sets that can be generated through programs. Notably, there is no need to explicitly store facts in the KBS. Intentional facts are dynamically deduced through rules. Extensional facts can be constructed “on-the-fly” via a metadata search engine that locates the corresponding resources.

3.3 On-demand Generation of Data Production Paths

Given this formal representation of the ontology, requests for data productions translate into queries to the knowledge base. A query is a description of the desired resource in terms of an ontology concept. It must be satisfied through extensional or intentional facts, the latter being sub-queries requiring further expansion. This iterative matching process takes into account all possible combinations of rules and extensional facts. The result is a set of trees, whose nodes are intentional facts and leaves are extensional facts, embodying all valid production paths through which data for the queried concept can be generated.

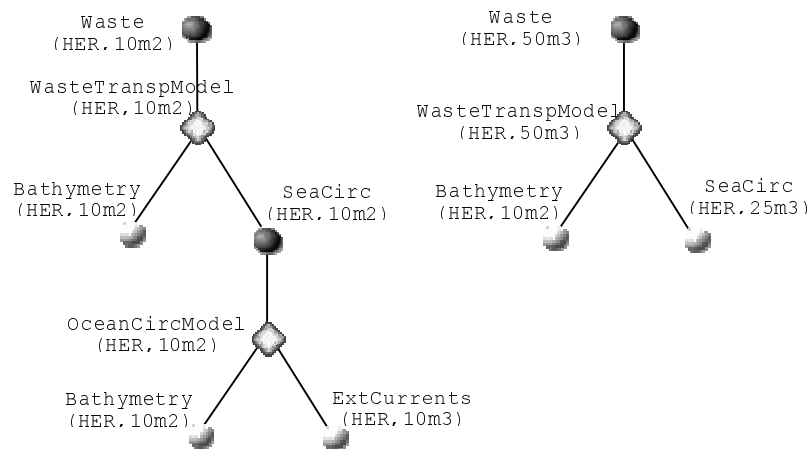


Fig. 3. Productions for *Waste* data, as presented to the user by a graphical user interface. Extensional facts are depicted in *light gray* while intentional facts, derived through rules, are depicted in *dark gray*. Program and data nodes appear as *squares* and *bullets* respectively.

Through a graphical user interface implemented on top of the KBS the user may explore valid data productions in a straightforward way. The GUI is responsible for forwarding user queries to the KBS, receiving and interpreting the resulting trees, and displaying this information.

To illustrate the on-demand generation of data production paths, let us assume that the following resources are available in the system repositories, expressed as extensional facts:

```
Bathymetry(HER, 10m2)
ExtCurrents(HER, 10m3)
OceanCircModel(HER, 10m2)
SeaCirc(HER, 25m3)
WasteTranspModel(HER, 10m2)
WasteTranspModel(HER, 50m3)
```

The user can inquire on the concept of Waste without restricting any attributes by posing the query `Waste(X, Y)`. The result will be a graphical presentation of all productions for obtaining this data (as shown in Fig. 3). Two different production paths are possible for obtaining Waste data, for different sets of attributes: one for the area of Heraklion in 2-D with a 10m^2 resolution and one for the area of Heraklion in 3-D with a 50m^3 resolution. In the first production, the Sea Circulation data needed by the Waste Transport model must be computed via an Ocean Circulation model. In the second production, the required Sea Circulation data is already available.

The production trees depend on the attribute constraints posed by the user and on resource availability. For example, the addition of a repository containing an Ocean Circulation model for the computation of 3-D Sea Circulation data could result in further possible production trees, which would be displayed in subsequent queries. On the other hand, the user could request Waste data for a grid resolution of 50m^3 , in which case only the production on the right of Fig. 3 would be generated. Similarly, the resolution attribute could be restricted to 10m^2 thereby resulting in the production on the left of Fig. 3.

4 Towards an Architecture for Integrated Scientific Repositories

We strongly believe that a KBS along the lines of this paper will be part of next-generation scientific information and experiment management systems. Specifically, we envision an advanced middleware architecture that seamlessly integrates Digital Library, Intelligent Information Integration, and Workflow technology. It is comprised of three main modules, the Metadata Search Engine, the Knowledge Base System, and the Workflow Runtime, which co-operate to provide the user with the desired functionality. The architecture is shown in Fig. 4. The functionality of each component is briefly described in the following.

The Metadata Search Engine is responsible for locating external resources, either data sets or programs. It may also retrieve complementary information stored in the repositories, e.g. user documentation on the available resources. The Search Engine accepts metadata queries on the properties of resources and returns a list of metadata descriptions and references. References point to repository wrappers, which provide an access and invocation interface to the underlying legacy systems where the data and programs reside.

The Knowledge Base System accepts queries regarding the availability of ontology concepts. It generates and returns the corresponding data productions based on the available resources and the constraints imposed by the ontology rules. These productions provide all the information that is needed to construct workflow specifications. The KBS regularly communicates with the Metadata Search Engine to update its extensional database.

The Workflow Runtime System monitors and coordinates the execution of workflows. It executes each intermediate step of a workflow specification, accessing data and invoking programs through the repository wrappers. Checkpoint and recovery techniques can be employed to enhance fault tolerance.

This architecture ensures the scalability and extensibility required in large, federated systems. It allows operationally autonomous and geographically dispersed organizations to selectively “export” their resources. Registering a new resource with the system requires merely supplying appropriate wrappers and metadata descriptions.

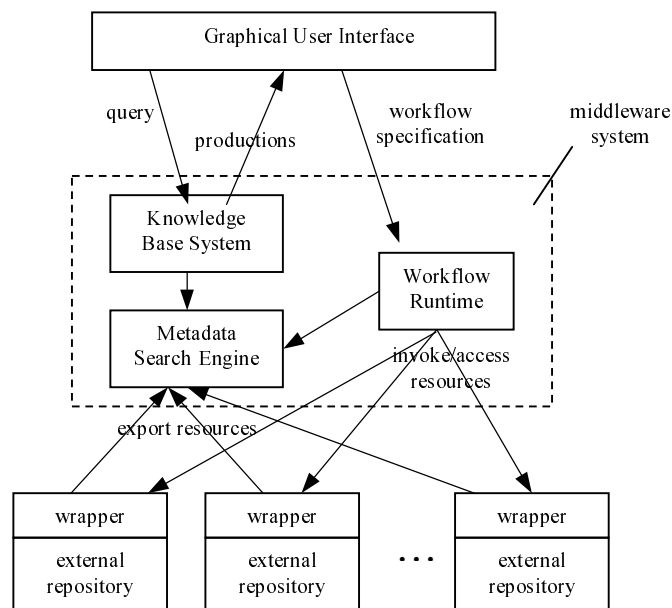


Fig. 4. A middleware architecture for distributed scientific repositories. The system consists of interoperable *Knowledge Base*, *Metadata Search*, and *Workflow Runtime* components.

To enhance performance and fault tolerance the Metadata Search Engine can be distributed across several machines. Also, several knowledge units adhering to different domains can be plugged into the Knowledge Base System to support a wide variety of applications and user groups.

We are currently developing a prototype system based on this architecture for the area of Coastal Zone Management [8]. Shortly, we hope to have a first version that will demonstrate the effectiveness of this design.

6. Related Work

Research on access and integration of heterogeneous information located over a number of distributed sources has attracted considerable attention during the past [22], [26], [28]. In this context, proposals have been presented for building global environmental systems [11]. Integration of scientific information has also benefited from workflow technology; while earlier focus was on business processes [1], workflow systems have recently been suggested as a powerful tool for scientific data management.

The Environmental Decision Support System (EDSS) [6] is a framework for the support of air quality modeling and management. Major physical and chemical phenomena are implemented as independent modules, which the user interconnects via a graphical interface. The selected modules are then obtained, compiled, linked, and executed automatically. Workflow and geospatial process management is explored in Geo-Opera [2], an extension of the Opera system [3] tailored for geographic and spatial modeling. Geo-Opera takes advantage of workflow technology to provide a fully distributed heterogeneous computing platform for managing complex geographic models. Another system that emphasizes management of scientific experiments is Zoo [10]. It uses Moose [16], an object oriented data model allowing computations to be described as graphs, and database functionality to control their execution. WASA [17] is a scientific application management environment, which uses IBM®'s FlowMark® [12], a business workflow management tool, to implement scientific processes. WASA supports dynamic replacement of a sub-workflow at run-time by another completely specified sub-workflow performing the same task.

In the above systems, workflows are explicitly specified. It is the user's responsibility to discover the availability of scientific resources and combine them into workflows. Once a workflow is defined it must be stored so that it can be retrieved at a later point in time. Also, for complex data productions it may be necessary to manually combine separate workflows. The efficiency of this approach greatly depends on the individual expertise of users.

Our approach can be viewed as a workflow generator. Rather than specifying workflows explicitly, a knowledge base committing to an environmental ontology is used to infer possible combinations between data and models automatically. Thus, it is possible even for non-experts to construct arbitrarily complex workflows. The effectiveness of the system is a direct function of the metadata design quality.

7. Conclusions

We have addressed the problem of determining combinations of available data and programs in the context of large-scale distributed scientific systems. We have proposed a Knowledge Base System that captures the semantic relations between data and programs, adhering to an application-specific ontology. Based on this information, the KBS dynamically deduces valid data production paths without user intervention. This support is particularly valuable in multidisciplinary applications.

It is important to further elaborate on the working paradigms of scientists to produce efficient combination rules that accurately capture the scope and restrictions of programs. Another challenging issue in this respect is to capture the knowledge of experts in metadata that could be used to create such production rules automatically.

An interesting extension, which can be easily implemented given the unification mechanism, is adding global constraints to the derivation of data production paths. This would also require appropriate metadata support. The user would then be able to pose restrictions regarding the total processing time or cost of data generation.

Finally, a distributed KBS consisting of a federation of logic modules embodying expertise in different scientific applications could be implemented. The KBS would distribute queries to the modules responsible for the specific concept domain. Interoperability could be achieved via standard protocols for exchanging queries and results [29], [31].

Acknowledgements

We would like to extend our thanks to the team from the Institute of Marine Biology of Crete consisting of George Triantafylou, Kaity Siakavara, George Petihakis, and Anika Polani for their discussions and sharing their knowledge on scientific models and data as they have appeared in our examples. We also thank Nickolas Patrikalakis for his valuable comments on earlier versions of this paper.

References

1. G. Alonso, D. Agrawal, A.El Abbadi, and C. Mohan, Functionality and Limitations of Current Workflow Management Systems. IEEE Expert Special Issue on Cooperative Information Systems, volume 1, number 3, 1997.
2. G. Alonso, H. Hagen, Geo-Opera: Workflow Concepts for Spatial Processes. Proceedings 5th Intl. Symposium on Spatial Databases (SSD '97), Berlin, Germany, June 1997.
3. G. Alonso, C. Hagen, H. Schek, M. Tresch, Towards a Platform for Distributed Application Development. 1997 NATO Advance Studies Institute (ASI). A. Dogac, L. Kalinichenko, T. Ozsu, A. Sheth (editors). August 12 -21, 1997, Istanbul, Turkey.
4. Object Management Group, The Common Object Request Broker: Architecture and Specification. Framingham, Mass., 1994. Revision 2.0.
5. C. Lagoze, E. Shaw, J.R. Davis, and D.B.Krafft, Dienst: Implementation Reference Manual. Cornell Computer Science Technical Report TR95-1514.

6. S. Fine and J. Ambrosiano, The Environmental Decision Support System: Overview and air quality application. Preprints, Symposium on Environmental Applications, January 28-February 2, Atlanta, GA, American Meteorological Society, 152-157, 1996. http://www.iceis.mcnc.org/pub_files/fine1996a.pdf
7. N. Guarino, Formal Ontology and Information Systems. In N. Guarino (ed.), Formal Ontology in Information Systems. Proceedings of the 1st International Conference, Trento, Italy, 6-8 June 1998. IOS Press.
8. C. Houstis, C. Nikolaou, S. Lalis, S. Kapidakis, V. Chrisophides, Open Scientific Data Repositories Architecture. Invited position paper, Invitational Workshop on Distributed Information, Computation and Process Management for Scientific and Engineering Environments proceedings, (edited by N. Patrikalakis), NSF sponsored, Washington DC, May 15-16, 1998.
9. C. Houstis, C. Nikolaou, M. Marazakis, N. Patrikalakis, J. Sairamesh, A. Thomsic, THETIS: Design of a Data Repositories Collection and Data Visualization System for Coastal Zone management of the Mediterranean Sea. Invited publication, D-Lib magazine, The Magazine for Digital Library Research, November 1997. <http://www.dlib.org/dlib/november97/thetis/11thetis.html>
10. Y. Ioannidis, M. Linvy, S. Gupta, N. Ponnekanti, ZOO: A Desktop Experiment Management Environment. Proceedings of 22nd VLDB Conference Mumbai (Bombay), India, 1996.
11. A. Koschel, R. Kramer, R. Nikolai, W. Hagg, and J. Wiesel, A Federation Architecture for an Environmental Information System incorporating GIS, the World Wide Web, and CORBA. Proceedings Third International Conference/Workshop on Integrating GIS and Environmental Modeling, National Center for Geographic Information and Analysis (NCGIA), Santa Fe, New Mexico, USA, January 1996.
12. F. Leymann, D. Roller, Business Process Management with FlowMark. Proceedings 39th IEEE Computer Society International Conference (CompCon), February 1994, pp230-233.
13. A. Seth, Changing Focus on Interoperability in Information Systems: From system, Syntax, Structure to Semantics. Interoperating Geographic Information Systems, M.F. GoodChild, M.J. Egenhofer, R. Fegeas, and C.A. Kottman (eds) Kluwer 1999.
14. O. Gunther, H. Lessing, and W. Swoboda, UDK: A European environmental data catalogue. Proceedings 3rd Int. Conf./Workshop on Integrating GIS and Environmental Modeling, National Center for Geographic Information and Analysis (NCGIA), Santa Fe, New Mexico, January 1996.
15. P.C. Wariyapola, S.L. Abrams, and N.M. Patrikalakis, Ontologies and Metadata for a Coastal Zone Management System. Technical Report (in preparation), MIT Ocean Engineering Design Laboratory, July 1998.
16. J. Wiener and Y. Ioannidis, A Moose and A Fox Can Aid Scientists with Data Management Problems, Proceedings 4th International Workshop on Database Programming Languages, New York, NY, August 1993, pp376-398.
17. Mathias Weske, Gottfried Vossen, Claudia Bauzer Medeiros, Scientific Workflow Management: WASA Architecture and Applications. Fachbericht Angewandte Mathematik und Informatik 03/96-I, Universität Münster, January 1996.
18. AltaVista. <http://www.altavista.com>
19. The CDF standard. <http://nssdc.gsfc.nasa.gov/cdf>
20. The Dublin Core. <http://purl.org/metadata/dublin-core>
21. Description Logics. <http://dl.kr.org>
22. The Digital Library Initiative. http://www.cise.nsf.gov/iis/dli_home.html
Berkeley <http://elib.cs.berkeley.edu>
Carnegie-Mellon <http://informedia.cs.cmu.edu>

- Illinois <http://dli.grainger.uiuc.edu>
Michigan <http://www.si.umich.edu/UMDL>
Stanford <http://www-diglib.stanford.edu>
Santa Barbara <http://alexandria.sdc.ucsb.edu>
23. EnviroTech on-line. <http://www.envirotech.org>
 24. Federal Geographic Data Committee, Content Standards for Digital Geospatial Metadata. March 1994. <ftp://fgdc.er.usgs.gov>
 25. The HDF standard. <http://hdf.ncsa.uiuc.edu>
 26. The Intelligent Integration of Information Initiative. <http://mole.dc.isx.com/13>
GARLIC <http://www.almaden.ibm.com/cs/showtell/garlic>
TSIMMIS <http://www-db.stanford.edu/tsimmis>
Information Manifold <http://www.research.att.com/~levy/imhome.html>
SIMS <http://www.isi.edu/sims>
InfoSleuth <http://www.mcc.com:80/projects/infosleuth>
WebSemantics <http://www.cs.toronto/~georgem/w5>
 27. The JDBC Data Access API. <http://www.javasoft.com/products/idbc/index.html>
 28. The Knowledge Sharing Effort. <http://www-ksl.stanford.edu/knowledge-sharing>
Ontolingua <http://www-ksl.stanford.edu/knowledge-sharing/ontolingua>
Mikrokosmos Ontology <http://crl.nmsu.edu/users/mahesh/onto-intro-page.html>
Ontosaurus <http://www.isi.edu/isd/ontosaurus.html>
Description Logic <http://www.cs.rmit.edu.au/dl>
Knowledge Interchange Format <http://logic.stanford.edu/kif/kif.html>
 29. The Knowledge Query Manipulation Language. <http://www.csee.umbc.edu/kqml>
 30. Microsoft, Microsoft Odbc 3.0 Software Development Kit and Programmer's Reference. Microsoft Press, February 1997.
 31. The Open Knowledge Base Connectivity. <http://www.ai.sri.com/~okbc>
 32. A. Denning, OLE Controls Inside Out. Microsoft Press, 1995.
 33. The Resource Description Framework. <http://www.w3.org/Metadata/RDF>
 34. D. Curtis, Java, RMI and CORBA. <http://www.omg.org/library/wpjava.htm>
 35. The Extensible Markup Language (XML) standard. <http://www.w3.org/XML>
 36. The Workflow Management Coalition. <http://www.aiai.ed.ac.uk/project/wfmc>