# GEON: Ontology-Enabled Map Integration

Kai Lin, Bertram Ludäscher
San Diego Supercomputer Center
University of California at San Diego
9500 Gilman Drive, La Jolla, CA92093-0505
{klin, ludaesch}@sdsc.edu

**Abstract**

In the GEON project we are developing an interoperability system on top of ArcIMS for registering spatial data sets to ontologies and subsequently querying registered data sets through the ontologies for map rendering. The system consists an ontology repository, a data set registration procedure, and a query rewriting system. User-defined ontologies are imported as OWL files and saved in the ontology repository. Structural and semantic heterogeneities of data sources are resolved using information from the data set registration procedure and ontologies. Those are used when rewriting user queries (e.g., a geologic age or rock type will expand to their corresponding "sub-concepts" in the rewritten query). Multiple ontologies are supported in the system by allowing users to define an articulation between two ontologies which equates some concepts in the source ontology to some concepts in the target ontology. Users are able to switch between ontologies for which an articulation exists.

## 1 Introduction

Currently there are many data sets in earth science that are isolated and are not used to their utmost potential. Even though a GIS system can display several data sets within a single map based on their spatial references, integrating and using them in a research environment is not practical. This is primarily due to the fact that each data provider uses his/her own standards and formats. In general, data heterogeneity can be divided into three categories [9]: syntactic heterogeneity, structural heterogeneity and semantic heterogeneity. Syntactic approaches can be used to handle the first two kinds heterogeneity effectively. But they can't be used to solve the semantic heterogeneity. The use of ontologies is considered a possible solution of semantic heterogeneity problem [12].

The difficulty of data integration can be illustrated by the problem of integrating geologic maps. We collected nine state level geology maps of USA in the shapefile format from state geological surveys. These shapefiles contain geologic age or rock type information in the tables with different schemas and vacabularies. Although these shapefiles can be simultaneously displayed on one map, it is not easy to query these data sets through a unified interface and show query results on a map, for instance, the following steps should be done if we would like to use Geology Time Scale from US Geological Survey and show all areas in these 9 states where rock has the geologic age `Mesozoic`:

1. Discovering Sub-concepts:

   We need to know that the geologic age `Mesozoic` has three subages: `Cretaceous`, `Jurassic` and `Triassic`, and each of them consists of several smaller time periods. The areas with these subages should be contained in the query result.

2. Resolving Structural Mismatch:

   For each shapefile, we need to know which column(s) should be used to find geology ages. In more general cases where some data sets are not shapefiles, we also need to know the formats of the data sets and how to access them.

3. Resolving Semantic Mismatch:

   In many cases, the vacabulary used in a data set may not completely match the vacabulary of the unified query interface. Even worse, the same term in the both vacabularies may have different meaning. For example, the geology age `Algonkian` is used in some geology maps, but it is not in the USGS Geology Time Scale. We need to know that `Algonkian` is a part of `Precambrian` so that if users query areas with the age `Precambrian`, the areas with geologic age `Algonkian` will be also included in the query result.

4. Rewriting Query:

   For each shapefile, one or several SQL queries should be created based on the infomation collected above. Furthermore, a query plan that decides a query order is often needed if data sets are dependent or performance is important.

5. Assembling Query Results:

   If the query result from each shapefile still uses their local vacabulary, then users have to understand the difference between these vacabularies. The best practice is to hide local vacabularies so that the query results from multiple data sets look like from a single data set.

It is inefficient and error-prone if all these steps have to be done manually. To alleviate such problems, in the GEON project [4] we are developing an interoperability framework and system that allows a data provider to register a data set with one or more ”mediation ontologies” – i.e. standards for data structure and content – and subsequently query the different data sets in a uniform fashion.

Within the system a user registers a geologic map using interactive tools against previously defined geologic age and rock type classification ontologies. The means to query the mediated data sets is significantly improved when all available data sets are registered in this way: Heterogeneous source vocabularies are made compatible via the ontologies, and multiple conceptual dimensions become queryable simultaneously (e.g., ”for all selected data sets, find regions with igneous rocks from geologic period P having composition C, fabric F, and texture T”). In order to answer such semantic queries against the mediated data sets, our system ”reasons” with the ontologies and builds the distributed query plans accordingly. The system is embedded in the GEON grid environment, i.e., it both uses GEON grid services (e.g., for remote data access), as well as provides new semantic mediation services for the GEON-Grid (e.g., ontology-enhanced query planning). The detail of our system is described in the next section. We also report some initial experiences of using the system to build an Ontology-enabled Map Integrator (OMI) in this paper.

# 2 Design of the Ontology-Enabled Map Integrator

In this section, we discuss the prototype system developed in the GEON project for exploring and integrating maps. The system consists of three components: an ontology repository, the map registration procedure, and a mediator.

## 2.1 Ontologies and Ontology Repository

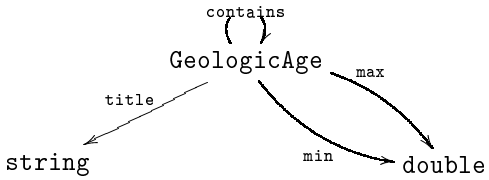
According to [5], an ontology is a specification of conceptualization, i.e., an ontology defines concepts and relationships among them. In the simplest form ontologies provide controlled vocabularies with more or less

formal descriptions of the pertinent concepts. In more sophisticated forms ontologies include formalizations (often through logic formulas) of properties of concepts and "inter-dependencies" of concepts. A prominent emerging standard for ontologies is the Ontology Web Language [13], which comes in three increasingly expressive variants: OWL Lite, OWL DL, and OWL Full. OWL is also an interesting example of how several interoperability levels and standards may be intertwined: for example, OWL DL builds upon the RDF model and syntax which in turn is usually denoted in XML syntax.

In OWL concepts are defined as classes, and relationships among concepts are defined as properties and constraints, and instances of a class are declared as individuals. The following is a fragment of the geology time scale ontology in OWL:

```
............

<owl:Class rdf:ID="GeologicAge"/>

<owl:TransitiveProperty rdf:ID="contains">
  <rdfs:domain rdf:resource="#GeologicAge"/>
  <rdfs:range rdf:resource="#GeologicAge"/>
</owl:TransitiveProperty>

<owl:DatatypeProperty rdf:ID="title">
  <rdf:type
   rdf:resource="&owl#FunctionalProperty"/>
  <rdfs:domain rdf:resource="#GeologicAge"/>
  <rdfs:range rdf:resource="&xsd#string"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="min">
  <rdf:type
   rdf:resource="&owl#FunctionalProperty"/>
  <rdfs:domain rdf:resource="#GeologicAge"/>
  <rdfs:range rdf:resource="&xsd#double"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="max">
  <rdf:type
   rdf:resource="&owl#FunctionalProperty"/>
  <rdfs:domain rdf:resource="#GeologicAge"/>
  <rdfs:range rdf:resource="&xsd#double"/>
</owl:DatatypeProperty>

............

<GeologicAge rdf:ID="Mesozoic" >
  <title rdf:datatype="&xsd#string">era</title>
  <min rdf:datatype="&xsd#double">65.5</min>
  <max rdf:datatype="&xsd#double">251</max>
  <contains rdf:resource="#Cretaceous" />
  <contains rdf:resource="#Jurassic" />
  <contains rdf:resource="#Triassic" />
</GeologicAge>

............
```

In this ontology, a new concept `GeologicAge` with four properties are defined. For a given geologic age, the transitive property `contains` gives some other geology ages it contains, and the property `title` returns a string, for example `eon, era, period` or `age`, and the properties `min` and `max` give its begining and end time, respectively. Note that a geologic age has a unique title and begining and end time, so `title`, `min` and `max` are declared as functional properties. An instance `Mesozoic` of `GeologicAge` is also defined in the fragment above, and it contains three geologic ages `Cretaceous`, `Jurassic` and `Trisassic`. The following diagram illustrates these definitions.




In OWL, an `isa` relation is defined by using the `SubClassOf` tag. For example, if we have the following definition is an ontology `genesis`:

```
.............
<owl:Class rdf:ID="Metamorphic" />
.............
```

then a subclass of `Metamorphic` can be defined as follows:

```
.............
<owl:Class rdf:ID="Marble">
  <rdfs:subClassOf rdf:resource="#Calcium"/>
  <rdfs:subClassOf
   rdf:resource="&genesis#Metamorphic"/>
</owl:Class>
.............
```

The system accepts and saves user-defined ontologies in OWL DL. Any ontologies available from internet can be imported to user-defined ontologies. After parsing OWL files, user-defined ontologies are saved into a database in a format that the system can use them directly in order to avoid unnecessary parsing. The system provides a basic navigation and visualization tool to browse the ontologies in the system repository.

## 2.2 Ontology Articulations

Frequently a domain can be modelled in many different ways. For instance, there are two rock classifications [3, 11] defined seperately by British Geological Survey and a canadian working group. Although these rock classifications have totally different class hierarchies, it is possible to define a so-called articulation translating some classes and properties in one classification into the another. Articulations between ontologies provide the possibility of switching ontologies which are very useful in parctice. More research on articulation can be found in [8, 1, 10].

An articulation between two ontologies defines how these two ontologies are related. Possible relations between two classes from the two ontologies are $\{\perp, \sqsubseteq, \sqsupseteq, =\}$ where $\perp$ indicates that two concepts are disjoint, and $\sqsubseteq$ and $\sqsupseteq$ give a subsume relation between two concepts, and $=$ means that two concepts are equivalent. Our system accepts ontology mappings. Ontology articulations are also provided as OWL files mainly containing `equivalentClass`, `subClassOf` and other tags. The following is a small fragment of an articulation between BGS rock classification and Canadian rock classification:

```
.............
<owl:Class
rdf:about="&br#FoidBearingMonzonite">
  <owl:equivalentClass
  rdf:resource="&ca-composition#FoidMonzonite"/>
</owl:Class>
 .............
```

Articulations can be used in query processing. If a data set is registered to an ontology $O_1$, and there is an articulation between ontologies $O_2$ and $O_1$, then users can choose ontology $O_1$ or $O_2$ to query the data set. In the example above, a user can use BGS rock classification or Canadian rock classification to send their queries to the system. If an another data set is registered to the ontology $O_1$ or $O_2$, then the system that integrates two data sets still can use either ontology for accepting queries.

## 2.3 Data Set Registration

Before an data set is accessible from the system, it must be registered to an ontology. Currently the system only accepts shapefiles [7] for registration. Our registration procedure takes at most 3 interactive steps to register something in a shapefile to an entity in an ontology:

1. **Ontology Entity Selection:** The user is asked to select a class, or a property, or an isa relation in a selected ontology. Choosing a class name indicates that individuals of that class can be generated for some rows in the shapefile. There are three options after selecting a class:
   (a) generating new references in this class;
   (b) generating new references in the subclasses;
   (c) mapping to the individuals in the ontology.
   If the option of generating new references in this class is selected, the system will generate new individuals of the selected class based on the value(s) in each row; otherwise the user is asked to map the data to some subclasses or individuals of the selected class.

2. **Mapping Data to Ontologies:** To register to a selected class $\mathcal{C}$, the user is asked to choose one or several columns with a boolean expression. We call these columns and this boolean expression a *$\mathcal{C}$-registered item*. The tuple of values from these columns in a row is called an *instance of this registered item* if the boolean condition is satisfied.

   If the user selects to generate new references in this class, then an instance of the registered item will be used as a reference of an individual of the selected class; otherwise the system will ask the user to choose one of the following methods to map instances of the registered item to the subclasses or individuals of the class:

   - *All Matches:* If (b) is selected and only one column is in the registered item, the name of a subclass $\mathcal{D}$ of the selected class is matched an instance of the registered item, then this instance is assumed to be an reference of an individual in the subclass $\mathcal{D}$.

     If (c) is selected and only one column is in the registered item and the name of an individual of the selected class defined in the ontology is matched an instance of the registered item, then this instance is assumed to be an reference of this individual in this row.

     This method can be used for a column containing multiple valid options, for example, "`Jurassic /Triassic`".

- *First Match:* If there are multiple matches by using the *All Matches* method, then take the first match to get a subclass or an individual in the ontology. This method can be used to select the begin point of a time interval, for example, "`Jurassic-Triassic`".
- *Last Match:* Similar to the *First Match*, this method takes the last match. This method fit best if the column contains a sequence of narrowing down descriptions, for instance, "`Mesozoic, Jurassic, Malm`".
- *Manual Setting:* The User decides subclasses or individuals for each instance of the registered item.

If multiple columns are selected, then the *Manual Match* method will be used to map the instances of the registered item to subclasses or individuals.

To register to an object property $p$ with the domain $\mathcal{C}$ and the range $\mathcal{D}$, the user is asked to select a $\mathcal{C}$-registered item $d$ and a $\mathcal{D}$-registered item $r$. If an instance of $d$ and an instance of $r$ are in the same row, then the individuals generated from the both instances has the relation $p$.

To register to a functional property $f$ with the domain $\mathcal{C}$ and the datatype $\mathcal{D}$, the user is asked to select a $\mathcal{C}$-registered item $d$ and a column $c$ whose type is $\mathcal{D}$. For any instance of $d$, suppose $o$ is its generated individual, then $f(d)$ is the value of the column $c$ in the same row.

The procedure of registering to an isa relation is almost the same as that of registering to an object property except that we require that any instance of $r$ is also an instance of $d$.

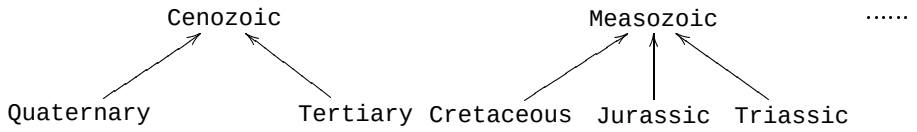
3. **Mismatch Resolution and Manual Setting:** The system prompts this step when one of following two cases occurs: 1) The manual setting is selected in the second step; 2) No match was found for some instances of a registered item by the selected method other than the manual setting method in the second step. The user is then asked to select subclasses or individual for these instances or ignore them.

The registration of a data set may be inconsistent or incomplete, where an incomplete registration defines a partial model of the ontology and no partial model of the ontology can be defined based on an inconsistent registration [2]. An inconsistent registration is an error and will be rejected by the system.

## 2.4 Case Study: Geology Map Integration

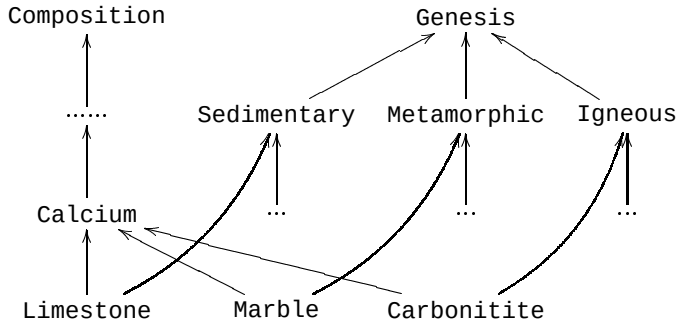
As a study case, a prototype that integrates geology maps from different geological surveys is created by using the system. The objective is to integrate available geologic data sets to provide a web-based interactive geology map for finding the location where rock has a specified geologic age or composition or fabric or texture or genesis property or any combination of these properties. Standard GIS functions are required.

Five ontologies are uploaded to the system: `GeologicAge`, `Genesis`, `Composition Texture` and `Fabric` [11, 6]. All these ontologies represent some controlled vacabularies. The individuals in the geologic age ontology can be simply described as the tree below:




The other ontologies demonstrate the similar tree structures. Furthermore, each subclass of composition or texture or fabric class may be a subclass of three genesis classes `Igneous`, `Sedimentary` and `Metamorphic`.

The diagram below shows a part of composition classification, which says that calcium and limestone rock is also sedimentary rock, whereas marble rock is also metamorphic rock.




Nine state level geology maps are registered to the ontologies above; they contain the rock age information in the states Arizona, Colorado, Idaho, Montana East, Montana West, Nevada, New Mexico, Utah and Wyoming. Additionally Idaho and Montana West data sets also provide rock type information. The following is the table schema of Arizona data set:

```
Arizona(AREA, PERIMETER, AZ_1000_, AZ_1000_ID, GEO,
        PERIOD, ABBREV, DESCR, D_SYMBOL, P_SYMBOL)
```

where the column `PERIOD` provides geologic age information. No rock classification information is provided. To register Arizona data set, we select `GeologicAge` class and the option of mapping to the indiviudal in the ontology at the first step, then select the column `PERIOD` and the *All Matches* method at the second step. The system scans the data in the column `PERIOD` and found two unmatched terms: `Water` and `Algonkian`, we choose `Ignore` for `Water` and `Precambrian` for `Algonkian`.

Other data sets have different schemas. For example, Idaho data set has the following schema:

```
Idho(AREA, PERIMETER, ID_500_, ID_500_ID, FORMATION, UNIT_NAME,
     ROCK_TYPE, ERA, SYSTEM, SERIES,  LITH1, LITH2, LITH3, LITH4,
     LITH5, LITH6, LITH7, LITH8, LOCATION1, LOCATION2, COMMENTS,
     IDCARB, IDK, IDBASE, IDFAM, IDPHOS, IDSG, IDBATHAB, LITHA,
     LITH_FORM,  PERIOD, D_SYMBOL, P_SYMBOL, LITH_MAJOR, LITH_MINOR,
     LITHOLOGY, AGE, IDLITH)
```

Geologic ages can be found in the column `AGE`, whereas `LITHOLOGY` contains information about rock composition, texture, fabric and genesis. All these data sets are registered in the similar manner as above.

Figure 1 shows the interface of the application for automatic map integration, where the map is the result of quering rock with the age `Mesozoic`. If `Info` is chosen, clicking on the map will return the raw data associated with the clicked polygon. Figure 2 shows the result after zooming in and selecting `Info` and clicking on a polygon.

The ontology `RockAndSediment` derived from BSG rock classification and an articulation from the `RockAndSediment` to the union of `Genesis`, `Texture`, `Fabric` and `Composition` are also sumbitted to the system. Then users are able to choose ontologies to send their queries.

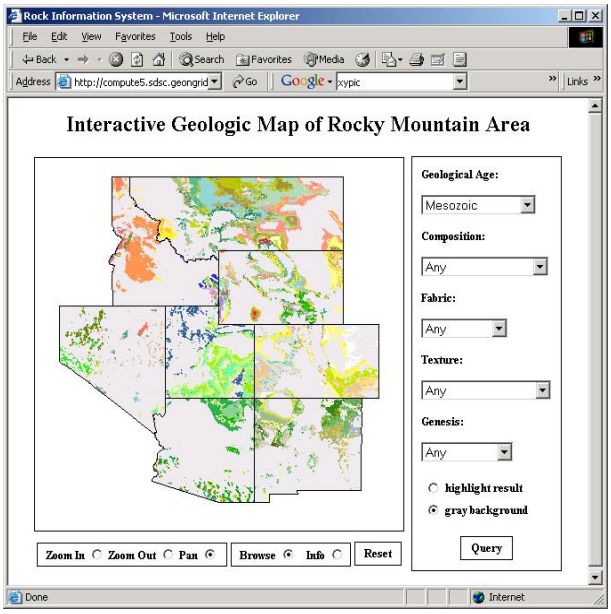



Figure 1: OMI Interface showing the integrated geologic maps (left), and ontology-aware query forms (right).

# 3 Conclusion

In this paper we discussed the algorithm of registring maps to ontologies for semantic map integartion. The experiments we have done show that ontology based map integration approaches are promising for scientific data integration. The novel design of the system makes it extremely easy to plug-in new maps. There are many open problems for future research, for instance, inconsistence detection of a data registration and efficient query processing. Also we paln to extend our system to be able to register databases as well as XML documents in the future.

# References

[1] Trevor J. M. Bench-Capon and Grant Malcolm. Formalising ontologies and their relations. In *Database and Expert Systems Applications*, pages 250–259, 1999.

[2] S. Bowers, K. Lin, and B. Ludaescher. On integrating scientific resources through semantic registration. In *16th International Conference on Scientific and Statistical Database Management (SSDBM'04)*, 2004. Santorini Island, Greece.

[3] British Geological Survey. The BGS rock classification scheme. `http://www.bgs.ac.uk/bgsrcs/home.html`, 1999.

[4] GEON. Geoscience network (geon) project. `http://www.geongrid.org`, 2003.

[5] T. R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5:199–220, 1993.

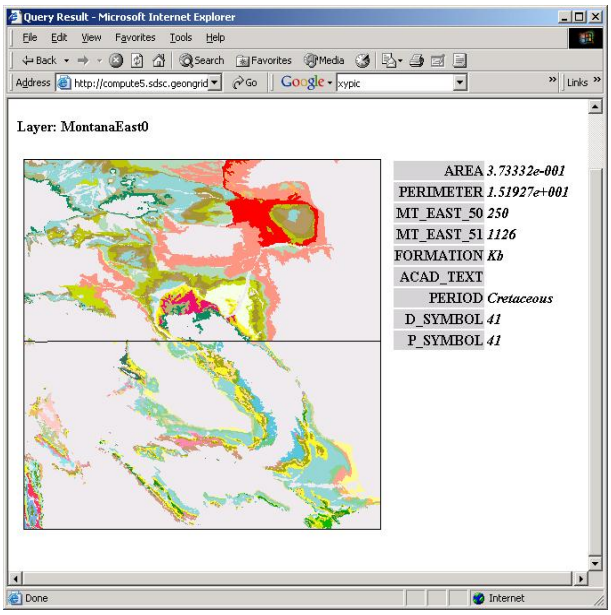



Figure 2: Snapshot of two adjunct geologic map (left), and some raw data (right), obtained by selecting a point of interest.

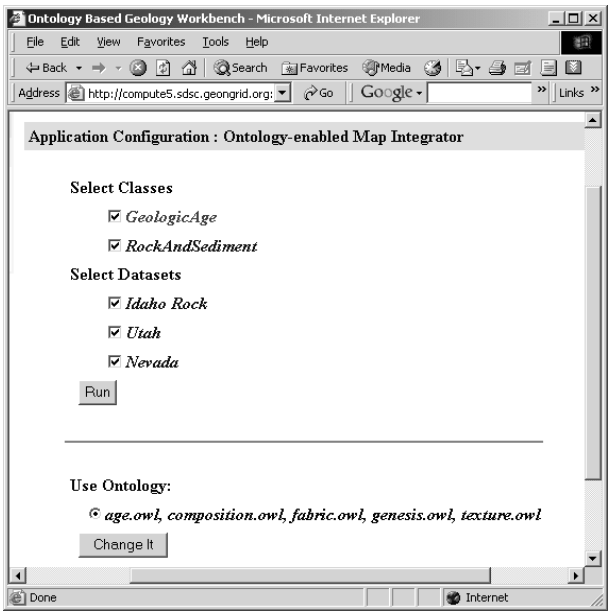



Figure 3: OMI Interface showing the integrated ontology and a web form allowing users to change ontology.

[6] W. Brian Harland, Richard Armstrong, Allan Cox, Craig Lorraine, Alan Smith, and David Smith. *A Geologic Time Scale 1989*. Cambridge University Press, 1989.

[7] Environmental Systems Research Institute. Esri shapefile technical description. `http://www.esri.com/hitepapers/pdfs/shapefile.pdf`, 1998.

[8] Yannis Kalfoglou and Marco Schorlemmer. Information flow based ontology mapping. In *Proceedings of 1st International Conference on Ontologies, Databases and Applications of Semantics*. Springer, 2002. Irvine, CA, USA.

[9] V. Stuckenschmidt Schuster. Ontologies for geographic information processing. `http://citeseer.nj.nec.com/article/visser00ontologies.html`, 2000.

[10] Derek Sleeman, David Robertson, S. Potter, and Marco Schorlemmer. Enabling services for distributed environments: Ontology extraction and knowledge-base characterisation. In *ECAI 2002 Workshop on Knowledge Transformation for the Semantic Web*, 2002.

[11] L.C. Struik, M.B. Quat, P.H. Davenport, and A.V. Qkulitch. A preliminary scheme for multihierarchical rock classification for use with thematic computer-based query systems. `http://www.nrcan.gc.ca/gsc/bookstore/free/cr_2002/d10.pdf`, 2002.

[12] H. Wache, T. Vogele, U. Visser, U. Stuckenschmidt, H. Schuster, G. Neumann, and S. Hubner. Ontology-based integration of information - a survey of existing approaches. In H. Stuckenschmidt, editor, *IJCAI-01 Workshop: Ontologies and Information Sharing*, 2001.

[13] World Wide Web Consortium. OWL Web Ontology Language Reference. `http://www.w3.org/tr/owl-ref/`, 2003.