

Integration of Pre-existing Heterogeneous Information Sources in a Knowledge Management System

Witold Staniszki, Edyta Kalka, Grzegorz Nittner, Eliza Staniszki, and Jakub Strychowski

Rodan Systems S.A., Puławska 465, 02-844 Warszawa, Poland¹
Witold.Staniszki@rodan.pl

Abstract. We present information integration features underlying the ICONS knowledge repository supporting data extraction from a variety of heterogeneous sources, such as structured data (i.e. relational data bases), semi-formatted data (i.e. XML, HTML files), text documents (i.e. plain text files, word processor files, PDF files), binary data (i.e. image files, audio files). The principal challenge is to insert and maintain integrated data objects within the structure of the ICONS repository data model defined in the Knowledge Schema. We discuss the ICONS Knowledge Schema, ICONS Concept Glossary and the advanced graphic interface features supporting storage and manipulation of the integrated heterogeneous information artefacts, access to external data sources with the use of a generalized wrapper architecture, and advanced information categorisation based on machine-learning techniques. The ICONS information integration capabilities are illustrated by a running example.

Introduction

The knowledge management life cycle supported by the ICONS platform [1,2] entails providing support for organization of knowledge artefacts within a formal, yet user-friendly, semantic data model based on a common ontology facilitating storage and manipulation of knowledge artefacts typically comprising information gleaned from pre-existing heterogeneous information sources. Information may be inserted into the ICONS repository content objects by an explicit user action performed on the object representation rendered in the graphic user interface, or implicitly by the appropriate object class method specified either as an application method or inherited from a system object class. Access to the repository is supported by powerful graphic interface features such as knowledge maps, intentional and extensional data model graphic navigation interface, and content object graphic presentation features based on electronic forms.

The content objects may include actual data providing information snapshots pertaining to a particular time frame (a time interval, an instance), or references to external information sources, thus representing virtual data to be materialised and

¹ This work has been supported by the European Commission Project ICONS IST-2001-32429.

presented to users at content object access time. Materialization of structured and semi-structured data is controlled by the wrapper generation rules. The ICONS integration architecture is presented in Fig. 1. Information integration may also be supported by third party data extraction and report generation tools. Such information may be stored in the ICONS repository to be subsequently accessible with the standard ICONS content object categorisation, selection and browsing features. This feature may be used to integrate information from external systems, such as business intelligence systems or transactional systems, to supplement content to be accessible via a knowledge management application.

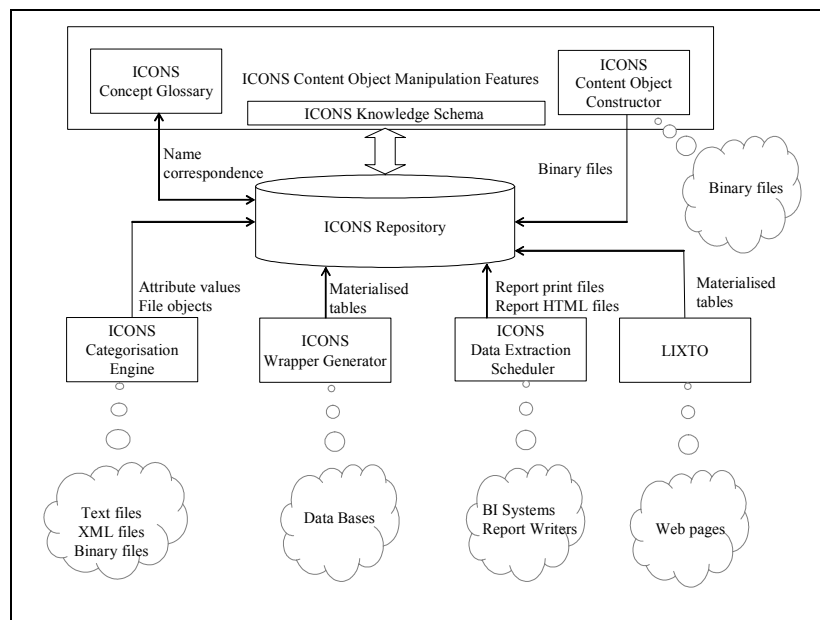


Fig. 1. The ICONS information integration architecture

Our approach to develop an information integration platform integrated with a knowledge management portal featuring an XML-based object repository is similar to integration architectures presented in [12,14]. Special requirements of content integration in the e-business context are exhaustively presented in [18].

In the following sections we present selected aspects of the ICONS Knowledge Schema pertaining to information integration, relevant aspects of the ICONS graphic user interface and content object manipulation, wrapper generation for structured data, integration and semantic reconciliation of structured data based on Disjunctive Datalog inferential content object class methods, as well as categorisation of text documents and binary files. Finally, we conclude highlighting the up to date production application experience and pointing at future research efforts pertaining to information integration.

Selected ICONS Knowledge Schema Features

The multi-paradigm Knowledge Schema [3], representing a conceptual view of the domain specific knowledge to be managed by the ICONS platform application., supports three principal knowledge representation paradigms, namely the structural knowledge, the declarative knowledge and the procedural knowledge paradigms.

The “Structural Knowledge” representations provide meta-information mechanisms for modelling content object class relationships, content object class behaviours, content object class grammars governing the internal object structure, and the object categorisation maps. The “UML Semantic Model” provides facilities to specify the class relationship structure as well as the class behaviour inheritance structure. The internal object class structures are defined with the use of the respective XML schema specifications. The “Content Object Structure” is determined by the corresponding XML Schema providing the grammar for parsing of content objects belonging to a given class as well as for generating default content object electronic form representations and XML editor renderings. The content objects are XML text files representing arbitrary trees compatible with the XML schema grammar defined for the corresponding object class.

The “Knowledge Map Model” provides facilities to represent object categorisations and to manage categorisation trees collectively constructing a knowledge map defined within an ICONS KM application. The object categorisation modes includes the “Value-based Categorisation” providing the principal mechanism for dynamic materialization of categorisation tree control structures, the “Manual Categorisation” supporting information-bearing relationships between a categorisation tree and the corresponding set of content objects, and the “Automatic Text Categorisation” using ontology-based machine learning techniques for text analysis and classification. The automatic text categorisation algorithm inserts the appropriate ontology term(s) into a predefined content object field(s) to provide required values for the consecutive value-based categorisation. The manual categorisation to be performed by the ICONS user is the principal mechanism for constructing the manually maintained categorisations trees stored in the Content Repository as static control structures, or for creation of user defined collections of content objects and relationships called the “Collection Objects”.

The ICONS Concept Glossary, based on the Topic Maps ISO standard [3,16,17], provides a mechanism to enforce and maintain ontological consistency of integrated data objects. Consistency is maintained through the name correspondence constraint meaning that all schema element names must be defined in the Concept Glossary. The explanation feature is supported by meta-information comprised in the Concept Glossary data structure, namely the concept definitions and concept relationships.

Relevant features of the Structural Funds Project Knowledge Portal presented in [4,15] are used as our running example illustrating the ICONS information integration features. The system comprises knowledge management features typical for the e-Government application domain. The SFPKP Content repository comprises content object classes comprising information on statistics data. *Statistics* are the important statistical indicators (economical, social, etc.) that are monitored in the structural funds implementation system. *Statistics values* are statistical indicator data, that are

extracted and integrated from external sources. Objects are presented and manipulated in a uniform way in a graphic interface as electronic forms (see Fig. 2). Initially eForms is generated automatically for a given taxonomy and then it can be modified manually. Taxonomy is derived from object grammar defined in XML Schema.

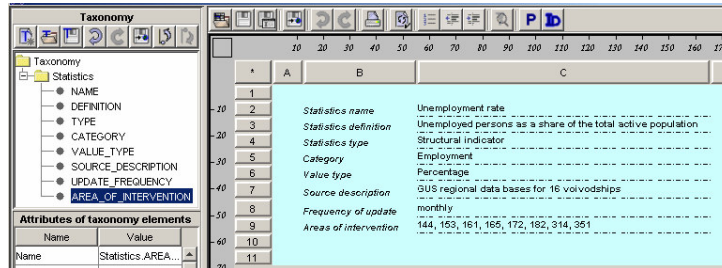


Fig. 2. Presentation of a content object in the ICONS GUI

Structured Data Wrapper Generation

Wrappers providing access to pre-existing relational databases are generated in the form of SQL queries on base tables and views defined in the corresponding database schema. The SQL query results are stored as materialised table attributes of the content objects invoking the wrapper process. The wrapper specification defined within the ICONS Knowledge Schema is based on a parametric Query by Example facility, where parameters are either a reference to the corresponding content object attribute values, or provide slots for values to be provided at the wrapper invocation time by the object method. The wrapper process (SQL query) is either invoked at the content object creation moment to store the materialised table snapshot data within the corresponding content object, to be possibly refreshed within the specified time intervals, or is invoked by an operation accessing the virtual materialised table attribute performed directly by the user on the content object form or by an appropriate content object class method. An arbitrary number of “materialised table” attributes may be defined within the object class XML schema.

In SFPKP statistical data are extracted from 16 regional data bases. Sample regional databases containing the unemployment data are shown in Table 1.

Table 1. External data sources tables:

1.Region: MAZOWIECKIE

Month	Year	Unemployment [%]	Unemployment Rate - Women	Unemployment Rate - Men
01	2003	12	15	10
02	2003	14,1	13,3	11,8

2.Region: MAŁOPOLSKIE

Date	Total Unemployment Rate
January 2003	0,135
February 2003	0,145

These regional databases are regarded as the external information sources. Sources are associated with adequate wrapper instances. Each wrapper takes responsibility for retrieval and transformation of data residing at the associated source. The range of wrapper types supported by given system depends on its needs, this example considers only one wrapper type – Data Extractor Wrapper (see Fig. 3).

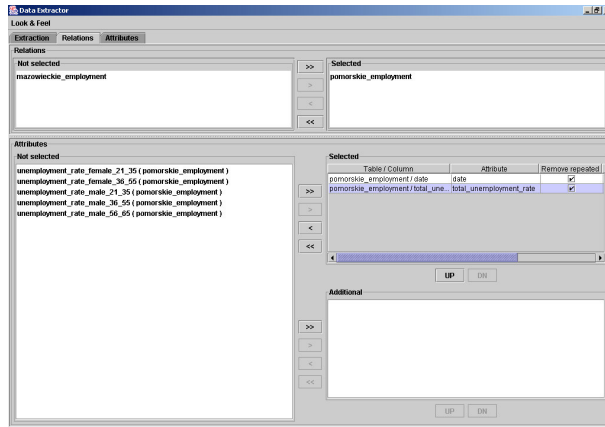


Fig. 3. The relational data wrapper generation screen

The external source content, that is to be retrieved by given wrapper, is described by the external source schema. The external source schema aim is to describe only the data relevant for the system. Thus the wrapper associated with given external source retrieves and transforms data to the form compliant with mentioned schema. Wrapper execution process includes the following actions: data retrieval, data cleaning, data storage (inside the Internal Data Store).

Sample external source schema looks as follows (notice that DATE attribute content is created during the cleaning data process on the basis of external source MONTH and YEAR attributes content.):

```
<xsd:schema
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<xsd:complexType name="MAZOWIECKIE_UNEMPLOYMENT_RATE">
<xsd:all>      <xsd:element name="DATE"
type="xsd:string"/>      <xsd:element
name="UNEMPLOYMENT" type="xsd:string"/>      </xsd:all>
</xsd:complexType></xsd:schema>
```

Additionally, each schema element is mapped to the corresponding element name registered within the ontology of the SFPKP. The ontology elements are stored and managed as concepts in Concept Glossary Manager. Each concept can have multiple names and multiple definitions. Concept names and definitions can be distinguished by their scope (e.g. language, visibility to SFKP end user, information source etc.).

Generation of semi-structured wrappers used for the web page information integration has been based on the LIXTO system exhaustively described in [5,13].

Integration and Semantic Reconciliation of Structured Data

We have taken an application-oriented approach allowing the knowledge application developer to specify inferential methods (i.e. Disjunctive Datalog programs) of the corresponding content object class to provide a facility to manipulate materialized table data in order to produce data integrated over several distinct pre-existing databases and, in particular, for reconciling existing semantic discrepancies.

The first issue in this context is the interpretation and merging of the data extracted from different sources. Interpreting data can be regarded as the task of casting them into a common representation. Moreover, the data returned by various sources need to be converted/reconciled/combined to provide the data integration system with the requested information. The complexity of this reconciliation step is due to several problems, such as possible mismatches between data referring to the same real world object, possible errors in the data stored in the sources, or possible inconsistencies between values representing the properties of the real world objects in different sources [6,7,8].

The ICONS materialized table integration entails two distinct integration steps; (1) data extraction step resulting in creation and storage in the ICONS repository of materialized tables, (2) integration of materialized tables extracted from different database views and/or web pages. The second integration step creating a resulting set of integrated data tables stored in content object attributes as new “materialized table” attributes is to be performed by the respective content class inferential method.

Presence of data cleaning action results from the need of eliminating a number of inessential inconsistencies (e.g. different date formats), that lead to semantically worse integration results. Data cleaning support should be regarded as the extension of wrapper functionality. Further data cleaning and reconciliation is performed by executing a DLV program. DLV rules are generated dynamically and executed on the data stored in corresponding content objects of the ICONS repository, that combined with Concept Glossary ontology data constitute DLV facts. As a result the extracted data are integrated and presented in a uniform SFPKP display format. Additionally the consistency of the data is verified and written is a *status* attribute value.

The corresponding DLV program looks as follows:

```
status(X,consistent) v status(X,inconsistent) :-
statistics_value(display,X,percentage,date).
status(X,inconsistent):-
statistics_value(display,X,_,_),X<0.
status(X,inconsistent):-
statistics_value(display,X,_,_),X>100.
statistics_value(display,X*100,_,_) :-
statistics_value(malopolskie,X,_,_) .
statistics_value(display,X,_,_) :-
statistics_value(mazowieckie,X,_,_)
```

ICONS Integrated Information Categorisation

File elements stored within the ICONS content objects are processed by appropriate categorisation modules matching the respective element type. The categorisation functions produce keyword values (character strings) to be stored in content object attributed to be subsequently used by the knowledge map and object relationship materialisation algorithms. In the case of XML documents the categorisation algorithm is simply copying the appropriate fields, marked by tags predefined as categorisation source data, into the corresponding attributes of the content object. Similarly, values extracted from binary files, specified by address displacement and byte length, are copied to appropriate content object attributes.

Categorisation of text documents [2] is based on an implementation of known text categorisation algorithms exploiting machine-learning techniques, such as the KNN (k Nearest Neighbours), Rocchio algorithm, and SVM (Support Vector Machine) selected dynamically by the text categorisation module depending on the text characteristics and the categorisation goal. An exhaustive discussion of the above text categorisation algorithms may be found in [9,10,11]

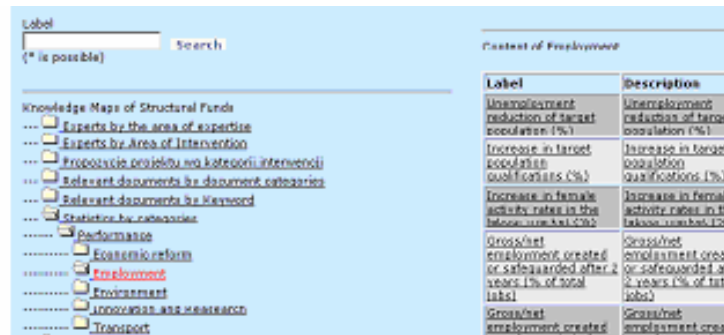


Fig. 4. Knowledge maps of the integrated statistical information

Statistics objects are categorized and presented to the user as a partially closed Knowledge Map (see Fig. 4), where the first node, the type comes from dictionary, inserted manually, whilst the second node, the category is based on the taxonomy of structural indicators used by Eurostat. The category attribute is determined by the automatic text categorisation process. When the value of the definition attribute is written in the content repository, it becomes an input document for the Text Categorisation Engine module. After the module performed categorisation, the result categories are stored in the category attribute. The accuracy of the categorisation is guaranteed by the high-quality classification algorithms and document preprocessing (native format to pure text, tokenization, stemming, removal of stop words, development of the words information gain).

Conclusions

The current research work is concentrated on the ICONS platform usability assessment through development of pilot knowledge management applications. Information integration features have already been ported to the OfficeObjects® content management platform.

References

1. The IST-2001-32429 ICONS Consortium, Intelligent Content Management System. Project Presentation, www.icons.rodan.pl, April 2002
2. The IST-2001-32429 ICONS Consortium, Specification of the ICONS architecture, www.icons.rodan.pl, February 2003
3. The IST-2001-32429 ICONS Consortium, The Multi-paradigm Integrated Knowledge Schema, www.icons.rodan.pl, February 2003.
4. The IST-2001-32429 ICONS Consortium, The Structural Fund Project Knowledge Portal, www.icons.rodan.pl, February 2003
5. Baumgartner, R., Flesca, S., Gottlob, G., Declarative Information Extraction, Web Crawling, and Recursive Wrapping with Lixto, Proc. of VLDB 2001
6. Bouzeghoub, M., and Lenzerini, M., Special issue on data extraction, cleaning, and reconciliation. Information Systems, 2001.
7. Calvanese, D., De Giacomo, G., Lenzerini, M., Nardi, D., and Rosati, R., Data integration in data warehousing. Int. J. of Cooperative Information Systems, 10(3), 237–271, 2001.
8. Galhardas, H., Florescu, D., Shasha, D., and Simon, E.. An extensible framework for data cleaning. Technical Report 3742, INRIA, Rocquencourt, 1999.
9. Joachims, T. 1997. A probabilistic analysis of the Rocchio algorithm with TFIDF for text categorisation. In Proceedings of ICML-97, 14th International Conference on Machine Learning (Nashville, US, 1997), pp. 143–151.
10. Joachims, T. 1998. Text categorisation with support vector machines: learning with many relevant features. In Proceedings of ECML-98, (Chemnitz, DE, 1998), pp. 137–142.
11. Joachims, T. 1999. Transductive inference for text classification using support vector machines. In Proceedings of ICML-99, 16th International Conference on Machine Learning (Bled, SL, 1999).
12. Lee, J., Siau, K., Hong, S., Enterprise Integration with ERP and EAI, Communications of the ACM, Vol. 46, No. 2, February 2003.
13. Lixto Visual Wrapper User Manual, Technical University of Vienna, 2002.
14. Mecella, M., Pernici, B., Designing wrapper components for e-services in integrating heterogeneous systems, The VLDB Journal 10: 2-15, 2001.
15. Staniszki, W., Staniszki, E., Intelligent Agent-Based Interactions in a Knowledge Management Portal, In: Proc.of EGOV2003, Volume 2739, LNCS, Springer Verlag, 2003.
16. ISO/IEC 13250, Topic Maps, Information Technology Document Description and Processing Languages, December 1999.
17. Pepper, S., The TAO of Topic Maps, STEP Infotek, www.infotek.no, November 2000.
18. Stonebraker, M., Hellerstein, J.M., Content Integration for e-business, Proc. of ACM SIGMOD Int. Conference, May 21-24, 2001, Santa Barbara, CA, USA.