Intelligent Configurable Electronic Shop Platform based on Ontologies and 3D Visualization

N. Karatzoulis[1], I. Tsampoulatidis[1], I. Maglogiannis[2], I. Zormpas[3],
D. Tzovaras[1] and M. G. Strintzis[1,4]
1 Informatics & Telematics Institute
1st Km Thermi-Panorama Road, PO Box 361,
GR-57001 Thermi-Thessaloniki, Greece
{nkaratz, itsam, tzovaras}@iti.gr
2 Dept. of Information and Communication Systems Engineering
University of the Aegean
{imaglo}@aegean.gr
3 Horama S.A.
{izor}@horama.com
4 Electrical and Computer Engineering Department
Aristotle University of Thessaloniki
GR-54006 Thessaloniki, Greece
strintzi@eng.auth.gr

Abstract. Although on-line shops have become very popular lately, there are still some sectors of industry, such as clothing and furniture, where the on-line shop concept has not penetrated the market yet. The main reason for that is the diversity of the specific sectors, the mentality of the market players and the lack of user friendly environments, where the customer could actually "see" the products. This paper presents an Intelligent Configurable Electronic Shop Platform, based on Ontologies and 3D Visualization, that aims at enabling a suitable representation of products with the most realistic possible visualization outcome. The platform, designed for the furniture sector, includes all the practicable electronic commerce variants and its on-line product configuration process is controlled by an ontology that was created using the OWL Web Ontology Language.

Please use the following format when citing this chapter:
Karatzoulis, Nikolaos, Tsampoulatidis, Ioannis, Maglogiannis, Ilias, Zormpas, Ioannis, Tzovaras, Dimitrios, Strintzis, Michael, 2006, in IFIP International Federation for Information Processing, Volume 204, Artificial Intelligence Applications and Innovations, eds. Maglogiannis, I., Karpouzis, K., Bramer, M., (Boston: Springer), pp. 147–155

1 Introduction

This paper presents an intelligent and user-friendly e-commerce solution, by adopting additional technologies such as a configuration utility supported by an intelligent help desk system and 3D visualization in a virtual reality environment. The focus of this paper is on the furniture sector, however the intention is that the system to be developed will be suitable for manufacturers, suppliers, and wholesalers, from other sectors, such as clothing, bicycles etc. The conception of the system is as generalised as possible, in order to facilitate the transfer to other industrial sectors.

The technical contribution of this paper is to present how to: (i) integrate and develop a product Visualization Tool capable of handling different media types (2D and 3D), (ii) introduce a novel configuration module, which is capable of integrating different functionalities, input and output devices and (iii) provide enhanced workflow mechanisms and tools for an easy integration into modern legacy systems (ERPs).

The described platform, called hereafter INCOVIS platform, has been designed for furniture products that can be configured on-line. The main modules of the INCOVIS platform are the **Intelligent Configurator Module** (and its ontology structure) and the **3D Visualization Module**. These two modules combined, comprise an advanced 3D Shop system that offers different products to customers, based on their demand and not on a fixed product line.

The paper is structured as following: in section 2 the intelligent configurator module is presented, with special focus on its architecture and ontology structure. In section 3 the 3D visualization module is analysed and in section 4 the e-shop solution and its integration with existing ERPs, is discussed. Finally, section 5 presents some experimental results and concludes the paper.

2 Intelligent Configurator Module and Ontology

The Intelligent Configurator module is a web based application that allows the user to assembly furniture products based on the available furniture parts that are being stored in the systems repository maintained by the furniture manufacturer. Figure 1 displays the Units of the Configurator module.

The Assembly Unit allows the user to insert individual 3D objects to the scene that can consist of a fully functional furniture product. The user can compose the desired product according to his/her needs by selecting the object's parts. The object and the texture selection processes are being controlled by the restriction mechanisms that are generated from the system Ontology [1, 2, 3]. The main functionalities of the Assembly Unit are the following: (i) Insertion of 3D object parts, (ii) Selection of texture and (iii) Assembly process based on rules (i.e. price limit).

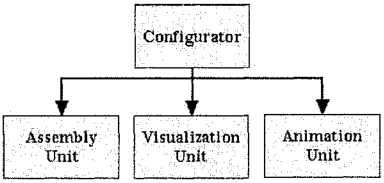
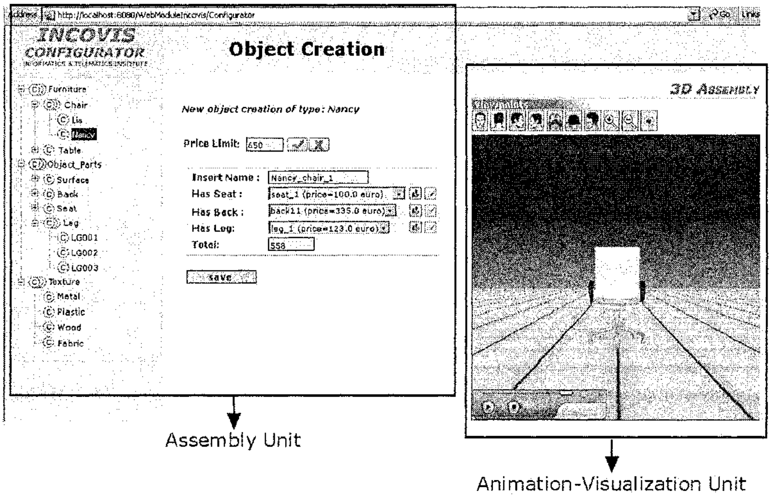


Fig. 1. The Units of the Configurator Module

The purpose of the Visualization Unit is to record and store the 3D object assembly steps in real-time. The assembly consequence is being stored in the 3D animation repository for future reproduction.

The Animation Unit allows the reproduction of the products assembly processes that are being stored in the animation system repository. In the Animation Unit the user can control the viewpoints and the playback of the loaded product assembly process. The web interface of the Configurator Module is depicted in Figure 2.



Object Creation

New object creation of type: Nancy

Price Limit: 650

Insert Name : Nancy_chair_1
Has Seat : seat_1 (price=100.0 euro)
Has Back : back11 (price=335.0 euro)
Has Leg: leg_1 (price=123.0 euro)
Total: 558

Fig. 2. The web interface of the Configurator

2.1 Configurator Architecture

The Configurator is implemented (Figure 3) using Java programming language. The system runs on Apache Jakarta Tomcat [4] as Java Servlet and it is based on the Jena framework [5], which is a Java framework for building Semantic Web applications.

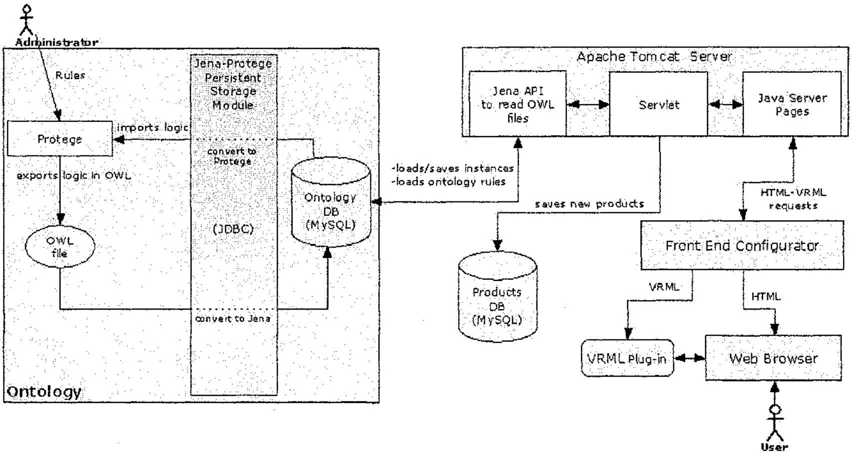


Fig. 3. Configurator Architecture

The ontology is created using the Protégé [6], which is an open source knowledge-base framework. The persistent store of the ontology is achieved using the persistence subsystem of Jena, while the 3D visualization is developed using the VRML [7] standard and External Authoring Interface (EAI) mechanisms.

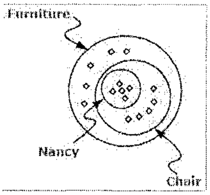
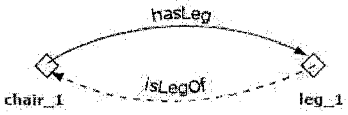
2.2 Ontology Development

The ontology was created using the OWL Web Ontology Language [8], and the Protégé OWL-Plugin [9], which is an extension of Protégé with support for the Web Ontology Language (OWL).

OWL-DL profile which was used in order to create the ontology, is based on Description Logics. Description Logics are a decidable fragment of First Order Logic2 and are therefore amenable to automated reasoning. It is therefore possible to automatically compute the classification hierarchy and check for inconsistencies in an ontology that conforms to OWL-DL [10].

The classes in Ontology are interpreted as sets that contain individuals. They are described using formal descriptions that state precisely the requirements for membership of the class. For example, the class "Chair" contains all the individuals that are chairs in the INCOVIS domain. The taxonomy of the classes is being achieved using the superclass-subclass model hierarchy.

Table 1. Example of the class hierarchy of the class "Chair" and an example of an Object Type Property for the individual "chair_1".

Class Model	Object Type Property
	

There are two types of properties supported by our ontology a) Data Type Properties and b) Object Type Properties. These OWL Properties represent relationships between two individuals.

In OWL, properties are used to create restrictions. In our ontology the restrictions were used to restrict the individuals that belong to a class. We used the universal quantifier ∀ restrictions to constrain the relationships along a given property to individuals that are members of a specific class. For example, the universal restriction ∀ hasBack back_1 describes the individuals all of whose hasBack relationships are members of the class Back.

Cardinality restrictions were used to define the order in which the individual object parts should appear during the 3D assembly process (i.e. real-time animation). The cardinality restrictions provided the way to describe the class of individuals that have at least, at most or exactly a specified number of relationships with other individuals or datatype values.

The hasValue restrictions, denoted by the symbol ∍, were used to describe the set of individuals that have at least one relationship along a specified property to a specific individual. For example, when we wanted to predefine the dimensions of an individual object part we used a hasValue restriction (dimensions ∍ "40-50-80").

2.3 Ontology Reasoning

Ontology reasoning is achieved using the Jena OWL reasoner (Figure 4). The Jena OWL reasoner could be described as an instance-based reasoner that works by using rules to propogate the if- and only-if- implications of the OWL constructs on instance data. Reasoning about classes is done indirectly - for each declared class a prototypical instance is created and elaborated. The sub-class and sub-property lattices are cached using the embedded OWL reasoner. Each domain, range, subproperty and sub-class declaration is eagerly translated into a single query rewrite rule. The result of a query applied to the graph will be the union of the results from applying the query plus all the rewritten versions of the query to the underlying graph [11].

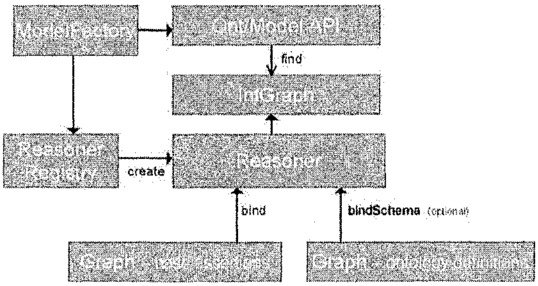


Fig. 4. The Jena Inference API layering [12]

3 3D Visualization Module

The 3D Visualization Module is realized on the Visualization and Animation Units. The structure of the individual 3D scenes supported by the 3D Visualization Module can be seeing at Figure 5.

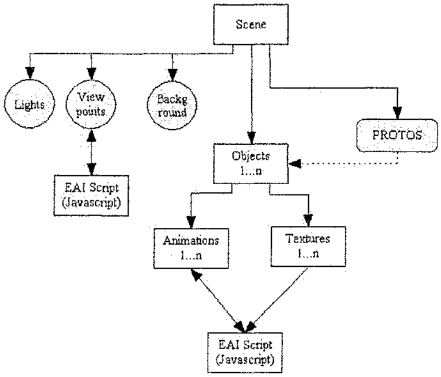


Fig. 5. The 3D scene structure of the 3D Visualization Module

The 3D scene contains the viewpoints, the lighting of the 3D world, the background, the 3D objects and the object functionalities (interactions) that are created dynamically according to the ontology specifications. The user interaction with the 3D scene is achieved by the use of predefined VRML Protos. For every 3D object that is inserted in the 3D scene, an animation representing its assembly process is dynamically generated.

4 The e-shopping platform in practice

The use of the INCOVIS platform "brings" advantages for both suppliers and buyers regarding (i) the cutback of transaction costs, (ii) the use of automated supply procedures, (iii) economy of scale, (iv) wide access on both local and international markets, (v) dynamic real-time price mechanisms/modules and (vi) the use of compatible/expandable technologies.

The requirements of the described INCOVIS platform for furniture products that together with the Intelligent Configurator Module and the 3D Visualization Module comprise the advanced 3D Shop system are:

- search and present all the available products, based on multi-criteria search engines
- group products into multilevel categories (set by the e-shop administrator)
- make offers/ sales and promote them
- update both the product catalogue and all items' availability (set by the e-shop administrator)
- create/use shop baskets (by the end buyers)
- provide several convenient pay/ receive methods
- provide a secure e-payment credit card transaction (with the use of HTTPS and SSL protocols).

However, the efficiency and overall quality of an e-commerce service depends "heavily" on its automatic connection with the existing ERP (Enterprise Resource Planning) system for the catalogue, prices, stock and product update. In order to integrate all the available ERP data with the e-shop database, a powerful staging mechanism is developed and securely transfers all necessary data. This staging process uses a smart "track changes" algorithm, to enhance the update speed.

There are two staging processes, Real Time Staging and Off Line Staging (that uses an automated batch process). The characteristics of the two staging "methods" are compared in the following table.

Table. 2 Staging Procedures Comparison

	Real Time Staging	**Off Line Staging**
Data Update	(+) All data are updated at all times	(-) All data are updated at specifically defined time periods
Infrastructure	(-) Reliable, high speed, technical infrastructure is necessary, available on a 24x7x365 basis	(+) Not so advanced technical infrastructure is necessary
Security	(-) The system can be secure but certain "protective" actions must be taken	(+) Security is obvious

The previous table shows that a real time staging procedure should be followed only if the nature of the commodity traded imposes the constant database update. In our case an every day off line procedure is chosen for both security and convenience reasons.

Yet, if we try to deduct a general case example we must note that each company's and product's needs, concerning the use of an e-market, are different; therefore the connectivity solutions (between an e-shop and an ERP) provided vary depending on: (i) the ERP used (it can be a widely used international ERP such as SAP, Oracle Applications, etc. or it can be a custom made system that fits to specific needs), (ii) the transaction volume and the form of the data transferred, (iii) the importance of the information transferred (regarding time, safety etc. aspects), (iv) the use of unilateral or bilateral communication and (v) whether it is an on-line or a batch transfer of data.

5 Experimental results and Conclusions

E-commerce services offered through a B2C (business to consumer) or B2B (business to business) system, provide the necessary infrastructure for real time e-business and an added value package of services that guarantee faster and more efficient buy and sell transactions, access to a broadened database of buyers/suppliers and business opportunities through the development of new partnerships.

In conclusion, in this paper we presented an interactive and user-friendly e-commerce solution for the furniture sector, but appropriate for other sectors as well. A furniture company called DIFROS S.A. has been the end-user responsible for using and testing the INCOVIS platform, so a number of its furniture products were integrated in the platform for evaluation and testing purposes (Figure 6). The evaluation procedure showed that the INCOVIS platform presented furniture products realistically and could be efficiently used as an on-line tool for their sales.

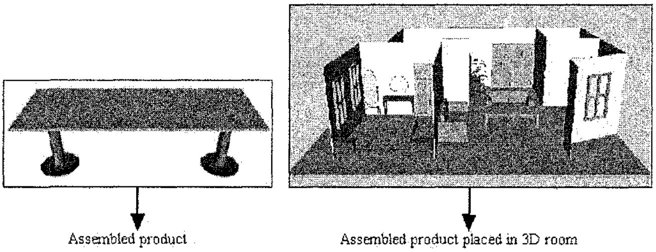


Fig. 6. The 3D product visualization in the INCOVIS platform

Finally, the main contribution is that it our approach adopts additional technologies such as an intelligent configurator module and a 3D visualization environment aiming at enabling a suitable representation of products in order to achieve the most realistic possible visualization and simulate an up to close shopping procedure.

Acknowledgment. Authors acknowledge the support given by the National General Secretariat of Research and Technology (http://www.gsrt.gr) to the INCOVIS project (http://www.incovis.gr) for carrying this research.

References

1. I. Kompatsiaris, V. Mezaris, and M. G. Strintzis, Multimedia content indexing and retrieval using an object ontology, Multimedia Content and the Semantic Web: Methods, Standards and Tools, G. Stamou and S. Kollias (Editors), ISBN 0-470-85753-6, Wiley, pp. 339-371 (May 2005).

2. I.Tsampoulatidis, G.Nikolakis, D.Tzovaras and M.G.Strintzis, Ontology Based Interactive Graphic Environment for Product Presentation, in Proc. CGI 2004, pp.644 - 647, Heraklion, Crete, Greece (June 2004).

3. V.Mezaris, I.Kompatsiaris, M.G.Strintzis, An Ontology Approach to Object-Based Image Retrieval, Proc. IEEE International Conference on Image Processing (ICIP 2003), Barcelona, Spain, vol. II, pp. 511-514 (September 2003).

4. Apache Jakarta; http://jakarta.apache.org/.

5. Jena, A Semantic Web Framework for Java; http://jena.sourceforge.net/.

6. Protégé, An Ontology Editor and Knowledge-base Framework; http://protege.stanford.edu.

7. Web 3D Consortium, VRML Standard; http://www.web3d.org/x3d/vrml/.

8. Protégé OWL Plugin ; http://protege.stanford.edu/plugins/owl/.

9. The OWL Web Ontology Language; http://www.w3.org/TR/owl-features/.

10. M. Horridge, H. Knublauch, A. Rector, R. Stevens, and C. Wroe, A Practical Guide To Building OWL Ontologies Using The Protégé-OWL Plugin and CO-ODE Tools, Edition 1.0, The University of Manchester (August 2004).

11. J. Carroll, I. Dickinson, C. Dollin, D. Reynolds, A. Seaborne, and K. Wilkinson, Jena: Implementing the Semantic Web Recommendations, Digital Media Systems Laboratory,HP Laboratories Bristol (2003).

12. Jena 2 Inference Support; http://jena.sourceforge.net/inference/.