ELSEVIER

Advances in Engineering Software 34 (2003) 621–631

ADVANCES IN ENGINEERING SOFTWARE

www.elsevier.com/locate/advengsoft

Distributed object model for collaborative CAD environments based on design history

C.A.M. Barbosa[a,*], Bruno Feijó[b], Marcelo Dreux[c], Rubens Melo[d], Sérgio Scheer[e]

[a]Institute of Computing/UFG, Campus 2, Block IMFI, Room 227, Goiania-GO 74001-970, Brazil
[b]Intelligent CAD Laboratory, Department of Computing, PUC-Rio, Rio de Janeiro, Brazil
[c]Department of Mechanical Engineering, PUC-Rio, Rio de jeneiro, Brazil
[d]Group of Technology in Database, Department of Computing, PUC-Rio, Rio de Janeiro, Brazil
[e]Center of Studies in Civil Engineering, Federal University of Paraná, Curitiba, Brazil

Received 24 April 2003; revised 12 May 2003; accepted 26 June 2003

Abstract

This paper presents new concepts for distributed dynamic design environments that deal with three aspects not effectively considered by the design literature in an integrated way: design history, CAD data model, and virtual prototyping. The present work details a design history model that gives support to views of higher levels than those usually found in design journals of CAD systems. The dynamic nature of the distributed environment is provided by an innovative mechanism that is not dependent on CORBA's DII or any other specialized middleware structure. Moreover, this paper presents an effective integration of a geometry bus on the top of a communication layer with strong support to mutant distributed objects at run time.

Keywords: Dynamic distributed object; Collaborative work; Design history; Product modeling; Engineering database

1. Introduction

The ultimate goal of design process management is to manage all aspects of the design process, including conceptual (or exploratory) design, problem decomposition, backtracking, consistency constraints, design history capture and recall, design rationale capture, feature-based modeling, design methodology enforcement and concurrent design. This definitive goal is far from being accomplished and many fundamental issues lack adequate treatment, particularly the questions of design history, CAD data models, and virtual prototyping in a collaborative environment.

Design history is not properly tackled by the industry and academy. CAD programs and design process manager software usually maintain a change history in their main data structure. This is essentially a feature that simply adds the dimension of time to design data through a kind of log mechanism (sometimes called design journal). The design journal tracks all changes made on a design: what has changed, the date and time of the change, the author, and comments about the reasons for the change. However, this feature merely represents temporal changes as an enumeration of discrete states. On the academic side, design research literature has few references on design history that go beyond simple design journals [1,2]. Also there are some efforts on design rationale capture [3] and semantics for feature-based design [4] that are higher-level issues related with design history. However, these references deal with specific domains (solid modeling, for example) and have no general model for collaborative CAD environments.

Another fundamental issue is the question of CAD data models. Data models for CAD are considerably more complex than those found in commercial applications, due to a number of reasons: (a) the deep hierarchical structure, (b) the multi-representational data aggregates, (c) the correlation across data representations, and (d) the connections across time. Moreover, the characteristics of the design process are quite unique, namely the iterative, exploratory, and collaborative nature of the design activities. In spite of

* Corresponding author. Tel.: +55-62-5211181; fax: +55-62-5211182.
E-mail addresses: barbosa@inf.ufg.br (C.A.M. Barbosa), bruno@inf.puc-rio.br (B. Feijó), dreux@mec.puc-rio.br (M. Dreux), rubens@inf.puc-rio.br (R. Melo), scheer@ufpr.br (S. Scheer).

0965-9978/$ - see front matter © 2003 Elsevier Ltd. All rights reserved.
doi:10.1016/S0965-9978(03)00095-4

the importance of CAD software (which is a multibillion dollar industry), research on CAD data models started only in the early 80s and is still not well-understood by the majority of the researchers in the database community. Firstly, research on this area attempted to extend the relational model [5] and, later, started investing resources on object-oriented models [6]. However, most of the literature still focuses on the issues of version modeling and propagation change. Few contributions present models for collaborative 3D CAD environments. Furthermore, database authors usually investigate object-sharing mechanisms or general aspects of heterogeneous systems, without considering the functional characteristics of the engineering design process and the complex requirements of handling distributed engineering data. On the other hand, researchers of collaborative CAD systems mostly concentrate on PDM (Product Data Management) systems based on web technology to provide groupware facility, without being in opposition to obsolete data models.

A third aspect of the computational view of design, which involves history and data models, is the question of virtual prototyping. The aim of virtual prototyping is to build a full virtual artifact in such a way that design and manufacturing problems are anticipated and discussed within a cooperative and distributed work environment, which embraces all the departments of a company. Virtual prototyping is much more than building a complete 3D model of an artifact. A virtual prototype works as a spatial and temporal object that can be built and queried by anyone in the enterprise on a heterogeneous computer network. The objects of a virtual prototype have several types of attributes, such as geometric values, design intents, manufacturing specifications, cost data, part numbers and references to documentation. Furthermore, these objects should be defined in the context of workflow management [7], organizational engineering [8] and requirements traceability [9,10]. From the implementation viewpoint, the objects of a distributed virtual prototype rest on two buses (Fig. 1): a geometry bus (e.g. ACIS [11]) and an object bus (e.g. CORBA [12]).

This paper deals with the three aspects above mentioned (design history, CAD data model, and virtual prototyping) in an integrated way. It presents a pragmatic approach to distributed object modeling for collaborative CAD environments that is strongly based on design history, independent of middleware, and has a special emphasis on the role of geometry buses for implementation purposes. This is a vast subject and the previous works by the authors concentrate on different aspects that complement the present paper [13,14]. The proposed model can be used as the basis for a new generation of CAD systems and can give support for future implementations of higher level manipulation of design objects, such as design rationale capture and feature-based modeling.

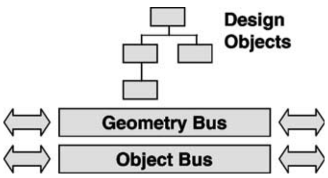

Fig. 1. Design objects on a distributed virtual prototype.

This paper is organized as follows. Firstly, it presents the object model, the concepts of Virtual Object Tree (VOT) and Interpreted Dynamic Objects (IDO), the mechanism for the propagation of changes in mutant objects, and a hierarchical definition of Design Entity. Secondly, the design history model is presented. Thirdly, the topics on the engineering database and the object communication mechanism are discussed. Fourthly, the component architecture, the distributed system, and an integration model are considered. Finally, the prototype is briefly presented and general conclusions are made.

2. The object model

In the proposed model, the link between two objects is itself an object defined by the following class [14]

```
class Link
  source //tag of the source object
  target //tag of the target object
  relation //any relation (IS_A, HAS_A)
  context //ELECTRICAL and/or MECHAN-
  ICAL and/or…
  start //the time when the link is
  created
  end //the current time or the time
  when the relation is removed
```

In the present paper, only `is-a` and `has-a` relations within a hierarchical tree are used. However, links and objects can also form cyclic graphs with generic constraint relations as proposed by Feijó et al. [15]. The class Link is particularly adequate to represent the dynamic nature of

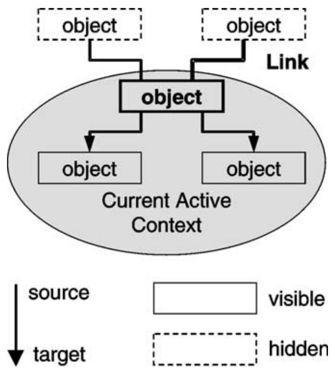

Fig. 2. An object, its neighborhood and context.